Temporal Multiscale Simulations for Multiphysics Problems

Dissertation

zur Erlangung des akademischen Grades

doctor rerum naturalium
(Dr. rer. nat.)

von Martyna Aleksandra Soszyńska, M.Sc.

geb. am 08.09.1992 in Poznań

genehmigt durch die Fakultät für Mathematik
der Otto-von-Guericke-Universität Magdeburg

Gutachter: Prof. Dr. Thomas Richter

eingereicht am: 20.02.2023

Verteidigung am: 14.06.2023

Abstract

We introduce novel time discretization schemes for coupled systems of partial differential equations. Our main focus is systems that are defined over spatially distinct domains with a common interface, where the coupling is enforced. We also look at volume coupled models defined over the same domain. Fluid-structure Interactions are one type of important application problem that fall into this framework.

Each of the physical problems can be governed by a different type of equations and therefore can exhibit different dynamics. In traditional methods, the time-step size has to be the same for both of the systems and adjusted to the subproblem with faster dynamics. As a result, the time-step size in the other one is unnecessarily small. Our aim is to develop time discretization schemes allowing for different time-step sizes without violating the coupling conditions. We introduce a hierarchy of time meshes - a common uniform coarse mesh and a second, finer mesh that can be chosen independently. The problems are formulated within the space-time framework which allows us access to the apparatus usually reserved for space discretization only. Although the formulation is monolithic, we solve the systems sequentially relying on a partitioned approach. To resolve the coupling conditions, special decoupling algorithms are introduced. Two such algorithms are discussed, namely a relaxation and a shooting method. We further develop an a posteriori error estimator based on the Dual Weighted Residual method and define the necessary adjoint formulations needed for this approach. The estimator is then used as an adaptivity criterion. We numerically test the performance of both of the decoupling strategies as well as the error estimator for a series of test problems.

We further expand our results by proving theoretical estimates and show error estimates for the implicit Euler time-stepping scheme. The starting point consists of a simple system of ordinary differential equations. It is then followed by the analysis of two heat equations coupled across a common interface scaled by different diffusivity constants. As the final problem, we look at an analogous Stokes equations system. We study both semi-discrete as well as fully discrete cases. For the Stokes problem, the estimates are proved in a special newly defined norm that incorporates the coupling conditions.

Zusammenfassung

Wir stellen ein neuartiges Zeitdiskretisierungsverfahren für gekoppelte Systeme von partiellen Differentialgleichungen vor. Wir untersuchen vornehmlich Systeme über räumlich getrennten Domänen mit einer gemeinsamen Schnittstelle, an welcher die Kopplungsbedingung forciert wird. Wir betrachten ebenfalls volumengekoppelte Modelle über der gleichen Domäne. Zu dieser Klasse von Problemen gehören unter anderem wichtige Anwendungsgebiete wie Flüssigkeits-Feststoff-Interaktion.

Jedes physikalische Problem kann durch eine andere Art von Gleichungen bestimmt werden und entsprechend unterschiedliche Dynamiken aufweisen. Bei traditionellen Methoden muss die Zeitschrittgröße für beide Systeme übereinstimmen und an das Unterproblem mit schnelleren Dynamiken angepasst werden. Als Ergebnis ist die Zeitschrittgröße im anderen Unterproblem unnötig klein. Unser Ziel ist es, Zeitschrittdiskretisierungsverfahren zu entwickeln, die unterschiedliche Zeitschrittgrößen ohne Verletzung der Kopplungsbedingungen ermöglichen. Wir stellen eine Zeitgitterhierarchie vor - ein grobes Gitter, das überall gleich ist, und ein zweites, feineres Gitter, das unabhängig gewählt werden kann. Wir formulieren die Probleme unter Verwendung des Orts-Zeit-Zugangs, der uns den Zugang zu Methoden ermöglicht, die normalerweise nur für die Raumdiskretisierung zur Verfügung stehen. Obwohl die Formulierung monolithisch ist, lösen wir die Systeme sequentiell unter Verwendung eines partitionierten Ansatzes. Um die Kopplungsbedingungen zu lösen, führen wir spezielle Entkopplungsalgorithmen ein. Wir schlagen zwei dieser Algorithmen vor – das *Relaxation*-und das *Shooting*-Verfahren. Wir entwickeln weiterhin einen a-posteriori Fehlerabschätzer, welcher auf der *Dual Weighted Residual*-Methode basiert. Wir definieren darüberhinaus die notwendigen dualen Probleme, die für diesen Ansatz erforderlich sind. Der Schätzer wird dann als Adaptivitätskriterium verwendet. Wir testen die Leistung beider Entkopplungsstrategien sowie des Fehlerschätzers für eine Reihe von Testproblemen numerisch.

Wir erweitern unsere Ergebnisse darüber hinaus mit theoretischen Abschätzungen und beweisen Fehlerschranken für die implizite Euler-Zeitschrittmethode. Wir beginnen mit einfachen Systemen gewöhnlicher Differentialgleichungen. Darauf folgt eine Analyse zweier Diffusionsgleichungen, die über eine gemeinsame Schnittstelle gekoppelt sind und über verschiedene Wärmeleitkoeffizienten skaliert werden. Als letztes Problem betrachten wir ein analoges System von Stokes-Gleichungen sowohl im semi-diskreten als auch im gänzlich diskreten Fall. Die Schätzungen für das Stokes-Problem werden mit Hilfe einer speziellen, neu definierten Norm bewiesen, welche auf den Kopplungsbedingungen aufbaut.

Contents

Introduction

In many engineering applications, we can encounter mathematical models where multiple physical phenomena simultaneously interact with each other. Such systems are called multiphysics problems. These might be fluid, structure, thermal, electromagnetic, acoustic, electric, and many other different models. Despite the differences, what these problems have in common is the importance of coupling capturing interdependencies between the individual parts of the entire model. This coupling can either happen across the volume of a physical domain (*volume coupling*) or only across a common interface in case the subsystems occupy distinct physical domains (*interface coupling*). It is a key component of multiphysics problems and therefore has to be accounted for with sufficient regularity. As a result, all of the interacting models have to be simulated simultaneously. This is one of the reasons why the simulation of multiphysics problems can be so challenging - each of the physical phenomena can have very distinct characteristics and can require completely different numerical approaches.

One special class of multiphysics problems consists of *Fluid-structure Interaction* (FSI) problems. There, we consider a solid structure immersed in a fluid. It is an example of an interface coupled problem. The fluid model is most commonly given by Navier-Stokes equations. The solid is usually modeled using either linear or nonlinear elastic equations. That results in a particularly demanding model. Indeed, even Navier-Stokes equations on their own are not fully understood and offer many open problems. Further, if we want to consider moving domains, each of the problems operates in different coordinate systems - the Eulerian framework is natural for fluid systems, while the Lagrangian point of view is more suitable for solid models. Each of the systems of equations is also of a different type. The fluid problem is an example of a parabolic system while the elastic structure is given by a hyperbolic system. The former, at least within the laminar regime, has smoothing properties whereas hyperbolic systems can be highly oscillatory.

This difference in dynamics is of particular interest to us. Obviously, it would be out of the scope of this thesis to consider all of the characteristics of multiphysics problems at once. Therefore, here we would like to mainly concentrate on time discretization. In traditional models, the choice of time-step sizes has to be adjusted to the problem with the fastest dynamics. This might lead to unnecessarily high resolution for the slower changing parts of the model. We would like to address this problem by introducing novel time discretization techniques that allow for using different time-step sizes for each of the subproblems. However, while doing so, we have to be careful not to violate the coupling conditions. Moreover, such
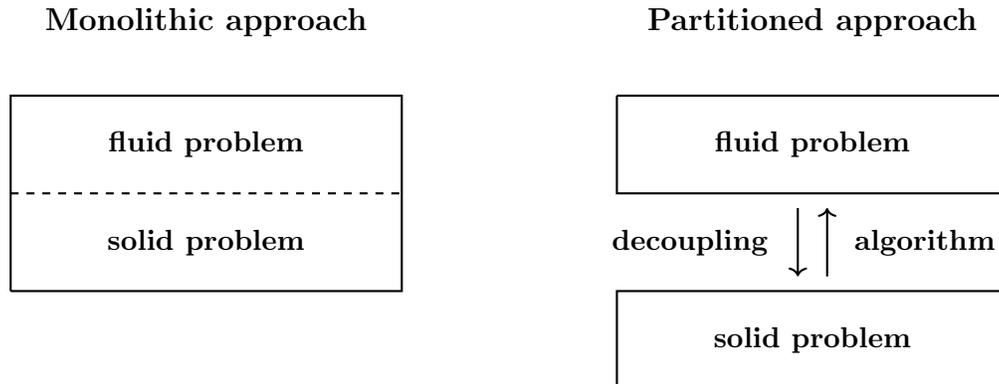
Figure 1: The difference between a monolithic and a partitioned approach.

a technique prohibits us from exploiting a traditionally used fully monolithic approach for solving algebraic systems arising from the discretization of underlying equations. In this approach, all of the systems are solved using one solver. The coupling conditions are guaranteed by the proper choice of test and trial function spaces. The discretization is therefore stable at the cost of low flexibility. An alternative is given by a partitioned approach. There, each of the systems is solved separately using its own solver. This flexibility allows for considering the special needs of each of the subproblems. The coupling is not accounted for within discretization and a special decoupling algorithm has to be introduced. Especially in the case of strong coupling, which may lead to many iterations within each time-step. In some circumstances, partitioned approaches are known to be unstable [1, 2]. The difference between the two approaches is sketched in Figure 1. In this thesis, we are going to use methods that borrow from both of the philosophies - formulations of the problems as well as discretization will be done monolithically, while we will use the partitioned framework to solve arising algebraic systems.

Although heuristically sometimes we can deduce which problems may need finer time discretization, it is highly beneficial to have access to an adaptivity criterion and therefore be able to choose time-steps automatically. As such a criterion, we will use an a posteriori error estimator based on a *Dual Weighted Residual* (DWR) method introduced by Becker and Rannacher in [3] and [4]. This method has been widely used for the space component however, it is less popular for time discretization. It requires embedding time-stepping, which is traditionally done using a finite difference method, within the finite element method framework. Fortunately, it can be done with relative ease and some of the most popular time-stepping schemes such as implicit Euler or Crank-Nicolson schemes have their finite element counterparts for specifically chosen trial and test spaces. To formulate a DWR estimator, one has to formulate, discretize, and solve an adjoint problem corresponding to the original one. This adjoint formulation is derived using a Gâteaux derivative and the procedure always yields linear problems. As a result, especially for complex nonlinear systems, the benefits of having

access to an adaptivity criterion outweigh the cost of computing an adjoint solution.

Another important property of novel discretization methods that has to be taken into consideration is their stability. It is a crucial property that ensures that inevitable roundoff errors are sufficiently damped and their accumulation is under control. Given the complexity of many physically meaningful multiphysics models, one usually has to simplify the system before performing any rigorous mathematical analysis. Assuming that some of the most important properties are preserved, such as for example incompressibility for incompressible flows, such simplified models can still provide important information about the dynamics of more realistic complex phenomena. We followed this route and analyzed the stability of multirate time discretization given by the implicit Euler scheme for a series of linear interface coupled problems. In the proofs, some of the key ingredients included a carefully defined operator transferring the interface values between the two time meshes as well as a modified inf-sup condition that was introduced by the authors in [5].

The thesis is split into two parts. The first four chapters together form the first part. There, we introduce the idea of multirate time-stepping methods, propose a way to solve such systems, and introduce an a posteriori estimator. The second part consists only of the last chapter, where we prove stability and error estimates. Although throughout the whole thesis, we discuss various aspects of multirate time discretization methods for coupled problems, each of the parts forms a separate cohesive story. Each of them considers a different set of model problems and proposes a different point of view – in the first part we are more interested in computationally feasible algorithms, and in the second we prove estimates depending on the unknown exact solutions.

In the first chapter, a set of examples is introduced that serves as a basis for our algorithms. In the first example, we couple heat and wave equations across a common interface. Despite its simplicity, this example presents a set of challenges typical for more complex problems. The heat equation is of parabolic type and therefore has smoothing properties. The wave equation, as an example of a hyperbolic equation, can be highly oscillatory. As the second example, we take a well-known challenging benchmark problem [6], where an elastic beam is submerged in a viscous fluid. The fluid material is modeled by Navier-Stokes equations, and the solid is described using the St. Venant-Kirchhoff model. Both of these models are nonlinear. They are also coupled across a common interface, where geometric, dynamic, and kinematic coupling conditions are established. All of these conditions are imposed weakly. For the conditions of the Neumann type, this is a standard procedure. The condition of the Dirichlet type is incorporated using the Nitsche method. The movement of the domains is resolved using the *Arbitrary Lagrangian Eulerian* (ALE) coordinates. As the last example, we take a thermoelasticity problem, where we couple linear elasticity equations with a linear equation describing temperature fluctuations. Conversely to the previous two examples, here the coupling happens across the whole volume of a common space domain. Each of these examples is described strongly as well as weakly using a continuous variational formulation.

In the next chapter, based on the previously described model problems, we discuss discretization methods. We start with time discretization and then establish links between com-

monly used time-stepping methods and space-time approaches. As it turns out, the implicit Euler method corresponds to a finite element method, where both trial and test functions are piecewise constant in time. The Crank-Nicolson scheme, on the other hand, can be formulated using continuous piecewise linear trial functions and piecewise constant test functions. We follow by introducing multirate time-stepping. The tension between the coupling conditions and flexible time-stepping for each subproblem is resolved by introducing a hierarchy of time meshes. We first define a coarse mesh, which is the same for both problems. Within this mesh, the coupling conditions are enforced. Each such time-step can be further split independently creating a finer time discretization. We then discretize all of the model problems and address the individual needs of each of them. Especially the second one requires some finesse in this regard. Then we shortly discuss space discretization. Here, we use standard commonly used elements. The only aspect which requires some additional attention is the transfer of interface values between the two meshes for the interface coupled problems.

The following chapter is devoted to decoupling methods. Since in each time-step we solve the algebraic systems using a partitioned approach, we need to introduce additional algorithms that take care of the coupling. We answer this need by presenting two solutions – a relaxation method and a shooting method. The former is given by a simple convex combination of the two previously computed solutions. In the latter, using the continuity of the solution on the interface, we introduce a shooting function. To find its root, a quasi-Newton method is defined. In each iteration, the method approximates the derivative of the shooting function using a finite difference quotient which in turn can be understood as a directional derivative. Since this process gives us access to the matrix-vector product only, we use an inner GMRES matrix-free solver. As it turns out, this method can be easily extended to volume coupled problems as well. We compare the performance of the two methods based on our model problems. For the heat and wave equation as well as the thermoelasticity ones, both of the methods perform well and we can observe the superiority of the shooting method. Due to the strong coupling and incompressibility of the fluid, the performance worsens significantly for the FSI problem.

In chapter four, we derive an a posteriori error estimator which is then used as an adaptivity criterion. Following the DWR method, we define adjoint variational formulations using the previously established embedding of time-stepping methods within the finite element method framework. Based on both the primal and adjoint solutions, the error estimator is defined. We reconstruct the exact solutions needed in the formulation by projecting the discrete solutions into higher dimensional spaces. The error is then split into contributions corresponding to different subproblems. Thanks to the good localizability properties of the estimator, one can easily distinguish between contributions coming from different time-steps. We use these localized values and compare them with a threshold value and mark time-steps for refinement. This adaptivity algorithm is then tested on the model problems. For the wave and heat equation and thermoelasticity problems, the algorithm is highly successful. For the FSI test case, we only test the estimator on fully uniform meshes.

In the last chapter, we derive stability and error estimates of the implicit Euler multirate

time-stepping scheme. In this portion of the thesis, a different set of test problems is considered. The simplest consists of a system of two coupled ordinary differential equations, then we look at two coupled heat equations scaled with different diffusivity constants. Finally, we proceed to two coupled Stokes equations. The two PDE problems are coupled across a common interface where standard kinematic and dynamic coupling conditions are imposed. For the last two problems, we consider both semi-discrete as well as fully discrete formulations. Also here, we impose Dirichlet conditions using the Nitsche method. The Nitsche terms are then incorporated into a new norm, in which we prove the estimates. Following [5], symmetrized variational formulations are used. From the same publication, we also borrow a modified version of a standard inf-sup condition. In the proofs, a proper definition of an operator that transfers the interface values between the two time meshes is crucial.

In both parts, we were able to produce new original results. The findings in the first part should be viewed holistically. Indeed, most of the individual parts were already introduced before. Multirate time-stepping was for example considered in [7, 8, 9]. The relaxation method was discussed in [10, 11, 12], and various versions of interface Newton methods were studied in [13, 14, 15, 16, 17, 18]. Representation of time-stepping schemes as finite element methods can be found in [19, 20]. Time adaptivity using the DWR method was shown in [21, 22]. However, the combination of all of these methods together forms a new cohesive framework. Starting from a strong formulation of a given problem, we show all of the steps needed to achieve a novel adaptive and flexible time discretization. Although we encountered some problems for strongly coupled problems, the methods presented here are general enough to be successfully implemented for a large variety of problems. The results restricted to the first model problem were published in [23]. The implementation was entirely done using the finite element software FEniCS in Python. The code is stored in the GitHub repository MSoszynska.

The novelty of the results presented in the second part is more straightforward. All of the proofs there form original work. Some of them use rather standard techniques such as for example the Gronwall lemma in the first one. Some of the other aspects required more originality. Especially challenging was a proper estimation of interface terms. As we established before, very helpful was work published in [5]. At the moment of writing this thesis, we are in the process of publishing these results.

Part I

Decoupling Strategies and Time Adaptivity

Chapter 1

Model Problems

Throughout this thesis, we will discuss adaptive time-stepping schemes for temporal multiscale problems. Instead of keeping the framework abstract, we will analyze three specific model problems. We will present them in detail in this chapter.

The first model consists of a linear system coupling a heat equation with a wave equation on a rectangular space domain. This simple model emulates some of the difficulties typical for FSI problems. Each of the equations is of a different type. The heat equation is an example of a parabolic problem, while the wave equation is a hyperbolic one. That in itself already poses a challenge akin to the ones appearing in the more complex problems.

The second one is an example of a true FSI problem - a popular benchmark problem first introduced by Turek and Hron [6] in 2007. There, one considers laminar incompressible flow around an elastic structure. The flow interacts with the elastic structure and leads to a large periodic displacement profile in the structure. That and the added mass effect make the simulation of this problem particularly challenging.

As our third and final model problem, we are going to consider a thermoelasticity problem. This almost linear system consists of an elasticity equation coupled with an equation describing temperature fluctuations. Also here, we are expecting oscillatory behavior of the elasticity model. Conversely to the previous problems, here the coupling happens across the whole space domain and not only the interface.

1.1   Strong Formulations

As the first step, we will define the problems by presenting the equations in a strong sense. Further in this chapter, we will proceed to their weak formulations using variational techniques.
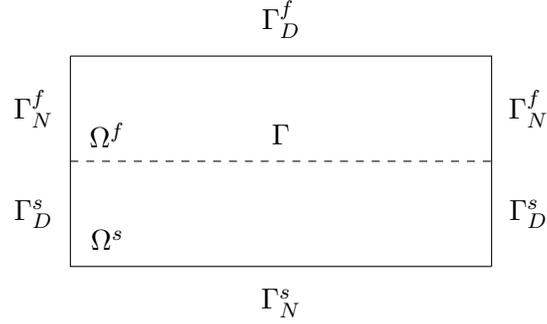
Figure 1.1: The domain Ω is split into two domains Ω^f and Ω^s which share a common interface Γ. On the boundaries Γ_D^f and Γ_D^s and we impose Dirichlet conditions while on Γ_N^f and Γ_N^s we prescribe Neumann conditions.

1.1.1 Heat and Wave Equation

This linear problem is too simplified to be considered an FSI problem. However, since we treat this example as an introduction to more complex models, we will use the corresponding nomenclature nevertheless. The heat equation will be associated with the fluid problem and the wave equation with the solid problem. Thus, we will use names such as *fluid problem*, *solid problem*, and *fluid domain, solid domain*, etc. In addition, the superscript f will refer to the fluid domain and we will connect s with the solid domain.

The problem is defined on a time interval $I = [0, T]$ on a space domain $\Omega := \overline{\Omega}^f \cup \overline{\Omega}^s$ consisting of two rectangular subdomains

$$\Omega^f := (0, 4) \times (0, 1), \quad \Omega^s := (0, 4) \times (0, -1).$$

The interface is defined as $\Gamma := \overline{\Omega}^f \cap \overline{\Omega}^s = (0, 4) \times \{0\}$. On the other boundaries, we either impose homogenous Dirichlet or free Neumann boundary conditions. A sketch illustrating the domains is shown in Figure 1.1. In the domain Ω^f we pose the heat equation

$$\partial_t v^f - \nu \Delta v^f + \boldsymbol{\beta} \cdot \nabla v^f = g^f \quad \text{in } I \times \Omega^f, \tag{1.1}$$

and in the domain Ω^s the wave equation

$$\partial_t v^s - \lambda \Delta u^s - \delta \Delta v^s = g^s, \quad \partial_t u^s = v^s \quad \text{in } I \times \Omega^s, \tag{1.2}$$

written as a first order system. By v^f and v^s we denote the velocities of fluid and solid and by u^s the solid displacement. $\nu > 0$ is the heat diffusion parameter, $\sqrt{\lambda}$ is the wave propagation speed, and $\delta \geqslant 0$ is a damping parameter. $\boldsymbol{\beta} \in \mathbb{R}^2$ refers to a transport direction. The two problems are coupled on the interface Γ by the transmission conditions

$$v^f = v^s, \quad \lambda \partial_{\vec{\mathbf{n}}_s} u^s + \nu \partial_{\vec{\mathbf{n}}_f} v^f = 0 \quad \text{on } I \times \Gamma. \tag{1.3}$$

These conditions mimic kinematic and dynamic coupling conditions commonly used in FSI problems. We distinguish between normal vectors corresponding to the fluid $\vec{\mathbf{n}}_f$ and the solid $\vec{\mathbf{n}}_s$ domain. Therefore it holds $\vec{\mathbf{n}}_f = -\vec{\mathbf{n}}_s$. The solid deformation u^s is harmonically extended to the fluid domain and denoted as u^f. That is analogous to tactics used in ALE coordinates. Here, the harmonic extension does not include any stiffening.

$$-\Delta u^f = 0 \text{ in } I \times \Omega^f, \quad u^f = u^s \quad \text{on } I \times \Gamma. \tag{1.4}$$

In the fluid domain, the left and right boundaries model free inflow and outflow, whereas the upper boundary describes a no-slip condition. In the solid domain, the left and right boundary model a fixed solid and the solid is free to move on the lower boundary, resulting in

$$
\begin{aligned}
u^f = v^f = 0 \text{ on } I \times \Gamma_D^f, \qquad & \partial_{\vec{\mathbf{n}}_f} v^f = \partial_{\vec{\mathbf{n}}_f} u^f = 0 \quad \text{on } I \times \Gamma_N^f \\
u^s = v^s = 0 \text{ on } I \times \Gamma_D^s, \qquad & \lambda \partial_{\vec{\mathbf{n}}_s} u^s + \delta \partial_{\vec{\mathbf{n}}_s} v^s = 0 \quad \text{on } I \times \Gamma_N^s.
\end{aligned} \tag{1.5}
$$

At time $t = 0$, all initial values are zero, i.e. $u^f(0) = u^s(0) = v^f(0) = v^s(0) = 0$. The exact values of the parameters read as

$$\nu = 0.001, \quad \boldsymbol{\beta} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \quad \lambda = 1000, \quad \delta = 0.1.$$

The external forces are set to be products of functions in space and time $g^f(\vec{\mathbf{x}}, t) := h^f(\vec{\mathbf{x}})f(t)$ and $g^s(\vec{\mathbf{x}}, t) := h^s(\vec{\mathbf{x}})f(t)$ where $h^f(\vec{\mathbf{x}})$, $h^s(\vec{\mathbf{x}})$ are space components and $f(t)$ is a time component that models a periodic pulse

$$f(t) = \begin{cases} 1 & z \leqslant t \leqslant z + 0.1 \\ 0 & z + 0.1 < t \leqslant z + 1, \end{cases} \quad \forall z \in \mathbb{Z}. \tag{1.6}$$

We will consider three different configurations of the right hand side. In Configuration 1.1, the right hand side is concentrated in Ω^f where the space component consists of an exponential function centered around $\left(\frac{1}{2}, \frac{1}{2}\right)$. For Configuration 1.2 we take a space component concentrated in Ω^s with an exponential function centered around $\left(\frac{1}{2}, -\frac{1}{2}\right)$. Configuration 1.3 is a combination of the two previous ones where the source terms are located both in the fluid and in the solid domain.

Configuration 1.1.

$$
\begin{aligned}
h^f(\vec{x}) &:= e^{-\left((x_1 - \frac{1}{2})^2 + (x_2 - \frac{1}{2})^2\right)}, & \vec{x} \in \Omega_f \\
h^s(\vec{x}) &:= 0, & \vec{x} \in \Omega_s
\end{aligned}
$$

Configuration 1.2.

$$
\begin{aligned}
h^f(\vec{x}) &:= 0, & \vec{x} \in \Omega_f \\
h^s(\vec{x}) &:= e^{-\left((x_1 - \frac{1}{2})^2 + (x_2 + \frac{1}{2})^2\right)}, & \vec{x} \in \Omega_s
\end{aligned}
$$

11
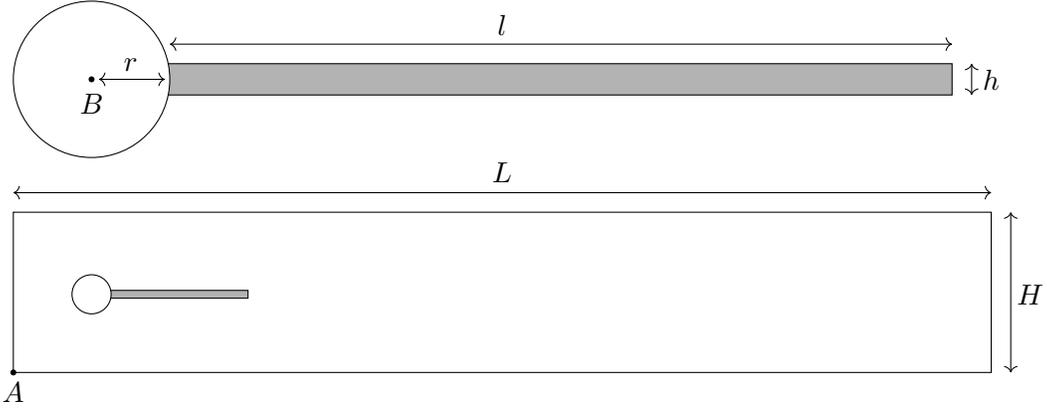
Figure 1.2: Geometric dimensions of the FSI benchmark. In the bottom figure, we can see the view of the whole domain. In the top figure, we zoom in on the cylinder and the elastic structure attached to it.

Configuration 1.3.

$$h_f(\vec{x}) := e^{-\left((x_1-\frac{1}{2})^2 + (x_2-\frac{1}{2})^2\right)}, \quad \vec{x} \in \Omega_f$$

$$h_s(\vec{x}) := e^{-\left((x_1-\frac{1}{2})^2 + (x_2+\frac{1}{2})^2\right)}, \quad \vec{x} \in \Omega_s$$

1.1.2 FSI Problem

The second problem we would like to consider is the FSI benchmark problem [6] where, the authors consider laminar incompressible flow around a cylinder, to which an elastic structure is attached. This is an extension of another popular benchmark problem featuring incompressible laminar flow around a cylinder [24].

This benchmark poses many challenges. Both the fluid and the elastic structure are modeled using nonlinear models. Additionally, the movement of the domains is included. Fluid and solid models operate in different coordinate systems which has to be accounted for. Finally, the ratio between the fluid and the solid densities $\frac{\rho^f}{\rho^s}$ is relatively high. That may introduce instabilities, especially if the discretized algebraic systems are solved using a partitioned approach. This problem is called *added mass effect* and has been explored in depth for example in [1] and [2].

In Figure 1.2, one can see the domain and in Table 1.1 the values of geometry parameters are listed. We would like to point out that the cylinder is not exactly in the middle. That allows the development of the oscillations. In Figure 1.3 we present and describe different boundary conditions. The inflow velocity continuously increases for the first two seconds

$$\mathbf{v}_{in}^f(\vec{x}, t) = \begin{cases} \mathbf{v}_0^f(\vec{x}) \frac{1-\cos\left(\frac{\pi}{2}t\right)}{2} & \text{for } t < 2.0 \\ \mathbf{v}_0^f(\vec{x}) & \text{for } t \geqslant 2.0 \end{cases}$$

geometry parameter	symbol	value
channel length	L	2.5
channel width	H	0.41
cylinder radius	r	0.05
elastic structure length	l	0.35
elastic structure thickness	h	0.02
reference point	A	$(0,0)$
reference point	B	$(0.2, 0.2)$

Table 1.1: Exact values of the geometry parameters mentioned in Figure 1.2. Note that this configuration is slightly asymmetrical.

and finally takes the form of a parabolic profile

$$\mathbf{v}_0^f(\vec{\mathbf{x}}) := \begin{pmatrix} 0 \\ 6\bar{\mathbf{U}} \frac{x_2(H-x_2)}{H^2} \end{pmatrix}.$$

$\bar{\mathbf{U}}$ denotes a prescribed mean velocity. The fluid problem is described using incompressible Navier-Stokes equations

$$\rho^f \partial_t \mathbf{v}^f + \rho^f (\mathbf{v}^f \cdot \nabla) \mathbf{v}^f = \operatorname{div} \boldsymbol{\sigma}^f$$
$$\operatorname{div} \mathbf{v}^f = 0. \tag{1.7}$$

Because of the upcoming coupling, we take the full symmetric tensor

$$\boldsymbol{\sigma}^f = -p^f \mathbf{I} + \rho^f \nu^f (\nabla \mathbf{v}^f + (\nabla \mathbf{v}^f)^T). \tag{1.8}$$

We consider constant density ρ^f and kinematic viscosity ν^f. On the outflow Γ_{out}^f, a modified natural outflow condition is imposed to avoid deflection at the end of the channel

$$(\rho^f \nu^f \nabla \mathbf{v}^f - p^f \mathbf{I}) \cdot \vec{\mathbf{n}}^f = 0.$$

The structure will be considered as an elastic compressible material

$$\rho^s \partial_{tt} \hat{\mathbf{u}}^s = \widehat{\operatorname{div}}(\hat{\mathbf{F}} \hat{\boldsymbol{\sigma}}^s). \tag{1.9}$$

The second time derivative is eliminated by introducing velocity $\hat{\mathbf{v}}^s = \partial_t \hat{\mathbf{u}}^s$. $\hat{\mathbf{F}}$ denotes a deformation gradient $\hat{\mathbf{F}} = \mathbf{I} + \nabla \hat{\mathbf{u}}$. As the material law, we choose St. Venant Kirchhoff model

$$\hat{\boldsymbol{\sigma}}^s = 2\mu^s \hat{\mathbf{E}} + \lambda^s \operatorname{tr}(\hat{\mathbf{E}}) \boldsymbol{I}, \tag{1.10}$$

where $\hat{\mathbf{E}}$ stands for Green-Lagrangian strain tensor $\hat{\mathbf{E}} = \frac{1}{2}(\hat{\mathbf{F}}^T \hat{\mathbf{F}} - \mathbf{I})$. The parameters λ^s and μ^s are called the first and the second Lamé parameters, respectively. They are related to the
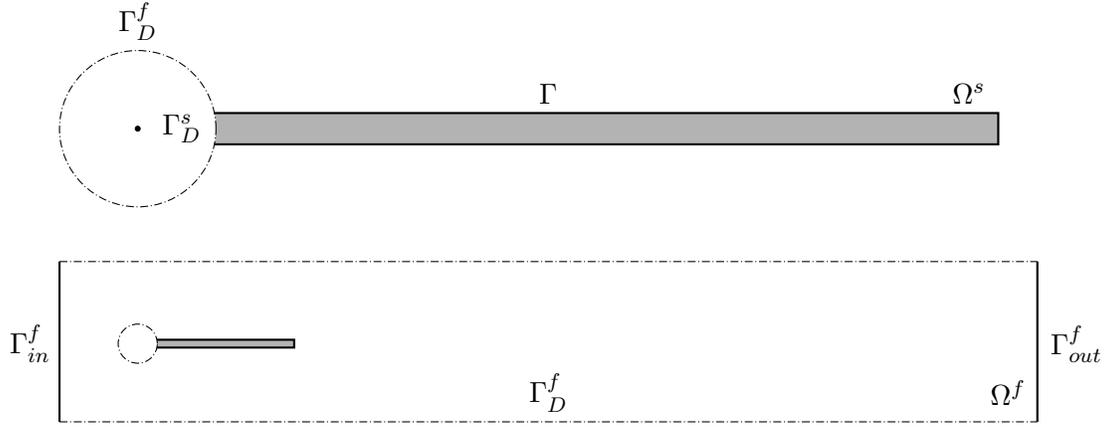
Figure 1.3: In this figure, we present the boundaries of the benchmark. The dashed lines correspond to no-slip boundary conditions. Depending on the domain, it is denoted by either Γ_D^f or Γ_D^s. On Γ_{in}^f we impose an inflow boundary condition. On Γ_{out}^f a free Neumann boundary condition is posed. Γ stands for an interface between the fluid and the solid models where coupling conditions act.

Poisson ratio ν^s describing compressibility and Young modulus E^s indicating stiffness of the material

$$\nu^s = \frac{\lambda^s}{2(\lambda^s + \mu^s)}, \quad E^s = \frac{\mu^s(3\lambda^s + 2\mu^s)}{\lambda^s + \mu^s}.$$

Any material with $\nu^s < \frac{1}{2}$ is considered compressible.

The fluid and the solid models operate in different coordinate systems. The Eulerian reference system is more appropriate for the fluid system, while the dynamics of the elastic structure are usually formulated in the Lagrangian system. Coupling these two configurations together requires the introduction of a common framework. There are two popular solutions to this problem. In the first one, called *Arbitrary Lagrangian Eulerian* (ALE) framework, one introduces an artificial displacement in the fluid domain $\hat{\mathbf{u}}^f$ and redefines the fluid problem in the Lagrangian coordinates. The biggest advantage of this approach is that the reference domain (including the interface) is fixed. However, the resulting system is highly nonlinear and the quality of the solution vastly depends on the regularity of the displacement $\hat{\mathbf{u}}^f$. In the second approach, the *Fully Eulerian* formulation, one instead formulates both of the subproblems in the Eulerian coordinates. The resulting system is simpler but additional techniques are needed to capture the movement of the interface. Here, we are going to consider the ALE approach.

We will define the fluid displacement as a continuous harmonic extension of the solid displacement with an additional stiffness operator S

$$S\Delta\hat{\mathbf{u}}^f = 0. \tag{1.11}$$

parameter	symbol	unit	FSI2	FSI3
fluid density	ρ^f	$\frac{kg}{m^3}$	$1 \cdot 10^3$	$1 \cdot 10^3$
kinematic viscosity	ν^f	$\frac{m^2}{s}$	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
mean velocity	$\bar{\mathbf{U}}$	$\frac{m}{s}$	$1 \cdot 10^0$	$2 \cdot 10^0$
solid density	ρ^s	$\frac{kg}{m^3}$	$1 \cdot 10^4$	$1 \cdot 10^3$
first Lamé parameter	λ^s	$\frac{kg}{ms^2}$	$2 \cdot 10^5$	$8 \cdot 10^6$
second Lamé parameter	μ^s	$\frac{kg}{ms^2}$	$5 \cdot 10^5$	$2 \cdot 10^6$

Table 1.2: Exact values for parameters for the two FSI test problems.

On the interface we have $\hat{\mathbf{u}}^f = \hat{\mathbf{u}}^s$ and on the other parts of the boundary, a no-slip boundary condition is set. We would like to emphasize that this displacement is indeed artificial and does not have any physical meaning, for example $\partial_t \hat{\mathbf{u}}^f \neq \hat{\mathbf{v}}^f$.

Using the extension above, one can as well extend the deformation gradient and its determinant $J = \det(\hat{\mathbf{F}})$. The The system after the ALE transformation takes the form

$$\rho^f \partial_t \mathbf{v}^f + \rho^f (\mathbf{v}^f \cdot \nabla) \mathbf{v}^f = \rho^f J \partial_t \hat{\mathbf{v}}^f + \rho^f J (\hat{\mathbf{F}}^{-1} (\hat{\mathbf{v}}^f - \partial_t \hat{\mathbf{u}}^f) \cdot \hat{\nabla} \hat{\mathbf{v}}^f).$$

The stress tensor equals to

$$\hat{\boldsymbol{\sigma}}^f = -\hat{p}^f \mathbf{I} + \rho^f \nu^f (\hat{\nabla} \hat{\mathbf{v}}^f \hat{\mathbf{F}}^{-1} + \hat{\mathbf{F}}^{-T} \hat{\nabla} \hat{\mathbf{v}}^T)$$

and the transformation of the incompressibility condition is defined by

$$\hat{J} \text{div } \mathbf{v}^f = \widehat{\text{div}}(\hat{J} \hat{\mathbf{F}}^{-1} \hat{\mathbf{v}}^f).$$

In the further parts of this thesis, we will not use hats to distinguish between quantities in the Eulerian and Lagrangian coordinate systems. For more details and a more rigorous analysis of the ALE approach, we refer to the book [25]. On the interface, the continuity coupling conditions in ALE coordinates stay unchanged

$$\mathbf{v}^f = \mathbf{v}^s, \quad \mathbf{u}^f = \mathbf{u}^s. \tag{1.12}$$

The balance of stress in the normal direction after the transformation is instead given by

$$J \boldsymbol{\sigma}^f \mathbf{F}^{-T} \cdot \vec{\mathbf{n}}^f = -\mathbf{F} \boldsymbol{\sigma}^s \cdot \vec{\mathbf{n}}^s. \tag{1.13}$$

On the left hand side, J comes from multiplying both sides of the fluid system by J and $\mathbf{F}^{-T}$ from integration by parts.

The authors in [6] analyze three test problems - FSI1, FSI2, and FSI3. Solving the FSI1 problem leads to a steady state and the other two test problems have periodic solutions. Since

the main focus of this thesis are time-depend problems, here we only analyze the last two configurations. The exact parameter values are displayed in Table 1.2. FSI2 leads to higher displacement profiles. That asks for a stronger stiffness operator (1.11), both of which are stated below

$$S_2(\vec{\mathbf{x}}) = 1 + 100\, e^{-75(x_1-0.6)^2} e^{-1000(x_2-0.21)^2}$$
$$+ 100\, e^{-75(x_1-0.6)^2} e^{-1000(x_2-0.19)^2},$$
$$S_3(\vec{\mathbf{x}}) = 1 + 20\, e^{-15(x_1-0.6)^2} e^{-200(x_2-0.21)^2}$$
$$+ 20\, e^{-15(x_1-0.6)^2} e^{-200(x_2-0.19)^2}.$$

Nevertheless, the FSI3 is considered the more challenging test case because of its density ratio $\frac{\rho^f}{\rho^s} = 1$ which is higher than $\frac{\rho^f}{\rho^s} = 0.1$ in the case of FSI2. The displacement profiles of both of these examples at the end of the elastic structure are shown in Figure 1.5.

1.1.3 Thermoelasticity

In our third and final model problem, we consider a thermoelasticity problem, where we look at an elastic material, whose material parameters depend on temperature. This problem will be decoupled into systems describing an elastic system and an equation governing temperature. Unlike the previous two examples, this problem is a volume coupled one. That means that both subproblems are defined over the same spatial domain.

We consider a T-shaped domain Ω^s which is presented in Figure 1.4. On all boundaries, we either set homogenous Dirichlet or homogenous Neumann boundary conditions. The geometry parameters are shown in Table 1.3. The elastic structure is modeled using damped linear elasticity equations

$$\rho\partial_t\mathbf{v} - \operatorname{div}\boldsymbol{\sigma} - \delta\Delta\mathbf{v} = \mathbf{f}, \quad \partial_t\mathbf{u} = \mathbf{v} \quad \text{in } I \times \Omega, \tag{1.14}$$

where the stress tensor is defined as

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\epsilon} + \lambda\operatorname{tr}\boldsymbol{\epsilon}\boldsymbol{I}, \quad \boldsymbol{\epsilon} = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right).$$

The Lamé parameters λ and μ depend on the temperature and are defined as

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

and $E = E_0(1 - \beta T)$, where T is the unknown temperature. This temperature is additionally governed by the equation

$$C\rho\partial_t T - \kappa\Delta T = \alpha\operatorname{div}\mathbf{u}. \tag{1.15}$$

All of the specific parameters are listed in Table 1.1.3. These values are matched with thermoelastic properties of steel. The external force is a pulse function that can be decomposed

geometry parameter	symbol	value
structure length	L	5.0
structure width	H	0.2
reference point	A	$(0,0)$
reference point	B	$(2.0, -2.0)$

Table 1.3: Values of the geometry parameters from Figure 1.4.
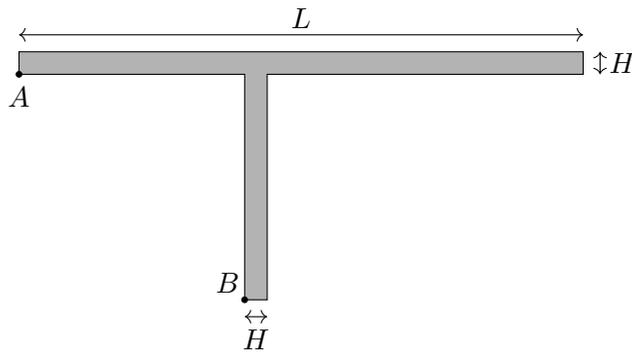
Figure 1.4: We present the T-shaped space domain and describe its geometry. As boundary conditions, on the very bottom of domain we take homogenous Dirichlet conditions for displacement and velocity. Otherwise we set homogenous Neumann conditions for all the unknowns.

as $\mathbf{f} = \mathbf{g}(t)f(t)$. The function f is defined in the same way as in the first model problem by (1.6). The function $\mathbf{g}$ is defined below

$$\mathbf{g}(t) = \left(2000\cos(10\pi t - 0.5\pi)^2, 0\right)^T.$$

1.2 Variational Formulations

The aim of this section is to transform the strong formulations of the model problems into variational equations suitable for the analysis from the perspective of the finite element method. This problem, given the defined variational forms and functions spaces, has the following variational form

parameter	symbol	unit	value
density	ρ	$\frac{kg}{m^3}$	7800
kinematic viscosity	ν	$\frac{m^2}{s}$	0.29
initial modulus of elasticity	E_0	$\frac{Gkg}{ms^2}$	210
damping parameter	δ	$\frac{kg}{ms}$	0.01
thermal conductivity	κ	$\frac{W}{m^\circ C}$	480
specific heat capacity	C	$\frac{J}{kg^\circ C}$	45
coefficient of thermal expansion	α	$\frac{G^\circ C}{m}$	20
modulus of elasticity scaling	β	$\frac{1}{^\circ C}$	0.25

Table 1.4: Material parameters for the thermoelasticity model problem. The values correspond to steel.

Problem 1.1. *Find $\vec{U} \in X$ such that*

$$B^f(\vec{U})(\mathbf{\Phi}^f) = F^f(\mathbf{\Phi}^f), \qquad \text{for all } \mathbf{\Phi}^f \in X^f,$$
$$B^s(\vec{U})(\mathbf{\Phi}^s) = F^s(\mathbf{\Phi}^s), \qquad \text{for all } \mathbf{\Phi}^s \in X^s$$

for the interface problems. The thermoelasticity variational problem is instead given by

Problem 1.2. *Find $(\vec{U}, T) \in X$ such that*

$$B^s(\vec{U}, T)(\mathbf{\Phi}) = F^s(\mathbf{\Phi}), \qquad \text{for all } \mathbf{\Phi} \in X^s,$$
$$B^T(\vec{U}, T)(\Lambda) = 0, \qquad \text{for all } \Lambda \in X^T.$$

Let us start with defining appropriate function spaces. As the first one, we will define a space akin to the space $H_0^1(\Omega)$, that is a space of functions with homogeneous Dirichlet boundary conditions whose first weak derivatives are in the space $L^2(\Omega)$. In our applications, we will define a similar space with the exception that instead of the whole boundary $\partial\Omega$, we will take only its subset $\Upsilon \subset \partial\Omega$

$$H_0^1(\Omega; \Upsilon) = \left\{ v \in H^1(\Omega) | \ v_{|\Upsilon} = 0 \right\}.$$

Similarly, one can define functions with weak L^2 derivatives, which on a subset of the boundary $\Upsilon \subset \partial\Omega$ are equal to a given function f

$$H^1(\Omega; \Upsilon; f) = \left\{ v \in H^1(\Omega) | \ v_{|\Upsilon} = f \right\}.$$

To define the whole space-time domain we will use the space

$$X(V) = \left\{ v \in L^2(I, V) | \ \partial_t v \in L^2(I, V^*) \right\},$$

where V^* denotes the dual space of V. On each of the domains Ω^f and Ω^s the L^2 products are defined using the notation

$$(\cdot,\cdot)_f := (\cdot,\cdot)_{H^1(\Omega^f)^d \times H^1(\Omega^f)^d}, \quad (\cdot,\cdot)_s := (\cdot,\cdot)_{H^1(\Omega^s)^d \times H^1(\Omega^s)^d}.$$

Using duality arguments, one can also define a similar product on the interface

$$\langle\cdot,\cdot\rangle_\Gamma := \langle\cdot,\cdot\rangle_{H^{-\frac{1}{2}}(\Gamma)^d \times H^{\frac{1}{2}}(\Gamma)^d}.$$

Finally, the products appropriate for expressing time derivatives are given by

$$\langle\cdot,\cdot\rangle_f := \langle\cdot,\cdot\rangle_{H^{-1}(\Omega^f)^d \times H^1(\Omega^f)^d}, \quad \langle\cdot,\cdot\rangle_s := \langle\cdot,\cdot\rangle_{H^{-1}(\Omega^s)^d \times H^1(\Omega^s)^d}.$$

d denotes dimensions of the spaces.

1.2.1 Heat and Wave Equation

In the case of our first example, where we couple the heat and wave equation, we choose $V^f := H_0^1(\Omega^f; \Gamma_D^f)$ and $V^s := H_0^1(\Omega^s; \Gamma_D^s)$ as the function spaces defined on the fluid and solid domain, respectively. For the fluid problem, we choose $\vec{\mathbf{U}}^f, \mathbf{\Phi}^f \in X^f := \left(X(V^f)\right)^2$ and for the solid $\vec{\mathbf{U}}^s, \mathbf{\Phi}^s \in X^s := \left(X(V^s)\right)^2$. These trial and test functions are defined as

$$\vec{\mathbf{U}}^f := \begin{pmatrix} v^f \\ u^f \end{pmatrix}, \quad \vec{\mathbf{U}}^s := \begin{pmatrix} v^s \\ u^s \end{pmatrix}, \quad \vec{\mathbf{U}} := \begin{pmatrix} \vec{\mathbf{U}}^f \\ \vec{\mathbf{U}}^s \end{pmatrix},$$

$$\mathbf{\Phi}^f := \begin{pmatrix} \varphi^f \\ \psi^f \end{pmatrix}, \quad \mathbf{\Phi}^s := \begin{pmatrix} \varphi^s \\ \psi^s \end{pmatrix}, \quad \mathbf{\Phi} := \begin{pmatrix} \mathbf{\Phi}^f \\ \mathbf{\Phi}^s \end{pmatrix}$$

and $\vec{\mathbf{U}}, \mathbf{\Phi} \in X := X^f \times X^s$. Given these functions, we can define the variational forms as

$$B^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := \int_I \langle \partial_t v^f, \varphi^f \rangle_f \, \mathrm{d}t + \int_I a^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) \, \mathrm{d}t + \left(v^f(0), \varphi^f(0) \right)_f,$$

$$B^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := \int_I \langle \partial_t v^s, \varphi^s \rangle_s \, \mathrm{d}t + \int_I \langle \partial_t u^s, \psi^s \rangle_s \, \mathrm{d}t + \int_I a^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) \, \mathrm{d}t$$
$$+ \left(v^s(0), \varphi^s(0) \right)_s + \left(u^s(0), \psi^s(0) \right)_s, \tag{1.16}$$

$$F^f(\mathbf{\Phi}^f) := \int_I \left(g^f, \varphi^f \right)_f \, \mathrm{d}t,$$

$$F^s(\mathbf{\Phi}^s) := \int_I \left(g^s, \varphi^s \right)_s \, \mathrm{d}t$$

with

$$a^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := \left(\nu \nabla v^f, \nabla \varphi^f \right)_f + \left(\boldsymbol{\beta} \cdot \nabla v^f, \varphi^f \right)_f + \left(\nabla u^f, \nabla \psi^f \right)_f - \left\langle \nu \partial_{\vec{\mathbf{n}}_f} v^f, \varphi^f \right\rangle_\Gamma$$
$$- \left\langle \partial_{\vec{\mathbf{n}}_f} u^f, \psi^f \right\rangle_\Gamma + \gamma\nu \left\langle v^f - v^s, \varphi^f \right\rangle_\Gamma + \gamma \left\langle u^f - u^s, \psi^f \right\rangle_\Gamma,$$

$$a^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := \left(\lambda \nabla u^s, \nabla \varphi^s \right)_s + \left(\delta \nabla v^s, \nabla \varphi^s \right)_s - \left(v^s, \psi^s \right)_s + \left\langle \nu \partial_{\vec{\mathbf{n}}_f} v^f, \varphi^s \right\rangle_\Gamma$$
$$- \left\langle \delta \partial_{\vec{\mathbf{n}}_s} v^s, \varphi^s \right\rangle_\Gamma. \tag{1.17}$$

To obtain this formulation, we first integrated by parts the Laplacian terms. The kinematic condition was incorporated into the fluid problem and the dynamic condition became a part of the solid problem. Both coupling conditions on the interface were imposed weakly. Since it is not a standard procedure for Dirichlet boundary conditions, we used the Nitsche method which was originally presented in [26]. The parameter γ can be seen as a penalization parameter enforcing $u^f = u^s$ and $v^f = v^s$ weakly. The parameter $\gamma > 0$ should be large enough to counter-balance different constants, like the one from the inverse estimate. We set $\gamma = 10$. Too small values for γ might cause a discrepancy from the Dirichlet condition, too large values worsen the conditioning of the resulting system. Normally, the Nitsche terms are scaled by γh^{-1}, where h stands for a space mesh size. We use this scaling in our simulations as well. However, for coherency, we will only include the parameter h in a fully discrete formulation. We refer to [27] for the analysis of a full fluid-structure interaction system with Nitsche coupling on the interface. We would also like to refer to Section 5.1, where different methods to enforce interface conditions are discussed.

1.2.2 FSI Problem

In the FSI example for the displacement and velocity, we again choose spaces of H^1 functions with appropriate Dirichlet boundary conditions. In the case of pressure, it is enough to assume L^2 regularity. Therefore

$$
\begin{aligned}
V_{\mathbf{v}}^f &:= H_0^1(\Omega^f; \Gamma_D^f), \cap H^1(\Omega^f; \Gamma_{in}^f; \mathbf{v}_{in}^f); \\
V_{\mathbf{u}}^f &:= H_0^1(\Omega^f; \partial\Omega^f \backslash \Gamma), \\
V_p^f &:= L^2(\Omega^f), \\
V_{\mathbf{v}}^s = V_{\mathbf{u}}^s &:= H_0^1(\Omega^s; \Gamma_D^s).
\end{aligned}
$$

The spaces are further specified by

$$
\begin{aligned}
X^f &:= \left(X(V_{\mathbf{v}}^f)\right)^d \times \left(X(V_{\mathbf{u}}^f)\right)^d \times L^2(I, V_p^f), \\
X^s &:= \left(X(V_{\mathbf{v}}^s)\right)^d \times \left(X(V_{\mathbf{u}}^s)\right)^d.
\end{aligned}
$$

We take trial and test functions defined over the fluid $\vec{\mathbf{U}}^f, \mathbf{\Phi}^s \in X^f := X(V^f)$ and solid $\vec{\mathbf{U}}^s, \mathbf{\Phi}^s \in X^s := X(V^s)$ domains. They are further split to

$$
\vec{\mathbf{U}}^f := \begin{pmatrix} \mathbf{v}^f \\ \mathbf{u}^f \\ p^f \end{pmatrix}, \quad \vec{\mathbf{U}}^s := \begin{pmatrix} \mathbf{v}^s \\ \mathbf{u}^s \end{pmatrix}, \quad \vec{\mathbf{U}} := \begin{pmatrix} \vec{\mathbf{U}}^f \\ \vec{\mathbf{U}}^s \end{pmatrix},
$$

$$
\mathbf{\Phi}^f := \begin{pmatrix} \boldsymbol{\varphi}^f \\ \psi^f \\ \omega^f \end{pmatrix}, \quad \mathbf{\Phi}^s := \begin{pmatrix} \boldsymbol{\varphi}^s \\ \psi^s \end{pmatrix}, \quad \mathbf{\Phi} := \begin{pmatrix} \mathbf{\Phi}^f \\ \mathbf{\Phi}^s \end{pmatrix}
$$

and $\vec{\mathbf{U}}, \mathbf{\Phi} \in X := X^f \times X^s$. After integrating the stress tensor by parts and applying the ALE transformation, we obtain the system

$$
\begin{aligned}
B^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := & \int_I \left\langle \rho^f J \partial_t \mathbf{v}^f, \boldsymbol{\varphi}^f \right\rangle_f \mathrm{d}t - \int_I \left\langle \rho^f J \mathbf{F}^{-1} \nabla \mathbf{v}^f \partial_t \mathbf{u}^f, \boldsymbol{\varphi}^f \right\rangle_f \mathrm{d}t \\
& + \int_I a^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) \, \mathrm{d}t + \left(\rho^f J \mathbf{v}^f(0), \boldsymbol{\varphi}^f(0) \right)_f \\
& - \left(\rho^f J \mathbf{F}^{-1} \nabla \mathbf{v}^f(0) \partial_t \mathbf{u}^f(0), \boldsymbol{\varphi}^f(0) \right)_f, \\
B^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := & \int_I \left\langle \rho^s \partial_t \mathbf{v}^s, \boldsymbol{\varphi}^s \right\rangle_s \mathrm{d}t + \int_I \left\langle \partial_t \mathbf{u}^s, \boldsymbol{\psi}^s \right\rangle_s \mathrm{d}t \\
& + \int_I a^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) \, \mathrm{d}t + \left(\rho^s \mathbf{v}^s(0), \boldsymbol{\varphi}^s(0) \right)_s + \left(\mathbf{u}^s(0), \boldsymbol{\psi}^s(0) \right)_s.
\end{aligned}
\tag{1.18}
$$

The forms used in the formulation above are defined below

$$
\begin{aligned}
a^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := & \left(J \boldsymbol{\sigma}^f \mathbf{F}^{-T}, \nabla \boldsymbol{\varphi}^f \right)_f + \left(\rho^f J \mathbf{F}^{-1} \mathbf{v}^f \cdot \nabla \mathbf{v}^f, \boldsymbol{\varphi}^f \right)_f + \left(J \mathrm{tr}(\mathbf{F}^{-1} \nabla \mathbf{v}^f), \omega^f \right)_f \\
& + \left(S \nabla \mathbf{u}^f, \nabla \boldsymbol{\psi}^f \right)_f - \left\langle J \boldsymbol{\sigma}^f \mathbf{F}^{-T} \cdot \vec{\mathbf{n}}_f, \boldsymbol{\varphi}^f \right\rangle_\Gamma - \left\langle S \partial_{\vec{\mathbf{n}}_f} \mathbf{u}^f, \boldsymbol{\psi}^f \right\rangle_\Gamma \\
& + \gamma \rho^f \nu \left\langle \mathbf{v}^f - \mathbf{v}^s, \boldsymbol{\varphi}^f \right\rangle_\Gamma + \gamma \left\langle \mathbf{u}^f - \mathbf{u}^s, \boldsymbol{\psi}^f \right\rangle_\Gamma, \\
a^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := & \left(\mathbf{F} \boldsymbol{\sigma}^s, \nabla \boldsymbol{\varphi}^s \right)_s - \left(\mathbf{v}^s, \boldsymbol{\psi}^s \right)_s + \left\langle J \boldsymbol{\sigma}^f, \mathbf{F}^{-T} \cdot \vec{\mathbf{n}}_f, \varphi^s \right\rangle_\Gamma.
\end{aligned}
\tag{1.19}
$$

In this variational formulation to avoid the second derivative in space of $\mathbf{u}^f$, we used the identity

$$
\mathrm{div}(J \mathbf{F}^{-1} \nabla \mathbf{v}^f) = J \mathrm{tr}(\mathbf{F}^{-1} \nabla \mathbf{v}^f).
$$

For the proof, we again refer to [25].

The implementation of the coupling conditions is very similar to the previous problem. All of the coupling conditions are imposed weakly which in the case of the kinematic condition is done with the help of the Nitsche method. During simulations we discovered that this problem requires a higher value of the parameter γ and therefore we chose $\gamma = 1000$. Also here, the scaling γh^{-1} will be added in front of the Nitsche terms in the fully discrete formulation only. The Dirichlet boundary conditions are added to the fluid and the Neumann conditions to the solid problem. In the Neumann conditions, we had to account for the ALE transformation and apply the stiffness operator.

1.2.3 Thermoelasticity

In this last model problem, we take the space $V^s := H_0^1(\Omega^s; \Gamma_D)$ to model the solution to the elasticity problem. Γ_D denotes the bottom part of the boundary, where the homogeneous Dirichlet boundary conditions are imposed. For the temperature equation, we take $V^T := H_0^1(\Omega^s)$. Further, we will use solid trial and test functions $\vec{\mathbf{U}}, \mathbf{\Phi} \in X^s := \left(X(V^s) \right)^{2d}$ and the temperature functions $T, \Lambda \in X^T := X(V^T)$

$$\vec{\mathbf{U}} := \begin{pmatrix} \mathbf{v} \\ \mathbf{u} \end{pmatrix}, \quad \boldsymbol{\Phi} := \begin{pmatrix} \boldsymbol{\varphi} \\ \boldsymbol{\psi} \end{pmatrix}.$$

That leads us to the problem

$$
\begin{aligned}
B^s(\vec{\mathbf{U}}, T)(\boldsymbol{\Phi}) &:= \int_I \left\langle \rho \partial_t \mathbf{v}, \boldsymbol{\varphi} \right\rangle \mathrm{d}t + \int_I \left\langle \partial_t \mathbf{u}, \boldsymbol{\psi} \right\rangle \mathrm{d}t + \int_I a^s(\vec{\mathbf{U}}, T)(\boldsymbol{\Phi}) \, \mathrm{d}t \\
&\quad + \left(\rho \mathbf{v}(0), \boldsymbol{\varphi}(0) \right) + \left(\mathbf{u}(0), \boldsymbol{\psi}(0) \right), \\
B^T(\vec{\mathbf{U}}, T)(\Lambda) &:= \int_I \left\langle C\rho \partial_t T, \Lambda \right\rangle \mathrm{d}t + \int_I a^T(\vec{\mathbf{U}}, T)(\Lambda) \, \mathrm{d}t + \left(C\rho T(0), \Lambda(0) \right), \\
F^s(\boldsymbol{\Phi}) &:= \int_I \left(\mathbf{g}, \boldsymbol{\varphi} \right) \mathrm{d}t
\end{aligned}
\tag{1.20}
$$

with

$$
\begin{aligned}
a^s(\vec{\mathbf{U}}, T)(\boldsymbol{\Phi}) &:= \left(\boldsymbol{\sigma}, \nabla\boldsymbol{\varphi} \right) + \left(\delta\nabla\mathbf{v}, \nabla\boldsymbol{\varphi} \right) - \left(\mathbf{v}, \boldsymbol{\psi} \right) \\
a^T(\vec{\mathbf{U}}, T)(\Lambda) &:= \left(\kappa\nabla T, \nabla\Lambda \right) - \left(\alpha \mathrm{div}\, \mathbf{u}, \Lambda \right).
\end{aligned}
$$

We arrived at this formulation by simple integration by parts. Note that this problem in general is a nonlinear one. However, given our decoupled splitting, each of the subsystems is reduced to a set of linear equations. Since all of the terms are integrated over the same space domain, we omitted the subscript in the L^2 product.

Here we would like to pay a bit of attention to the indices "s" and "T". They will be used throughout this thesis while referring to the thermoelasticity problem. However, whenever we will discuss general concepts concerning coupled problems, where the volume coupling nature of this problem is not important, we will instead use indices "f" and "s".
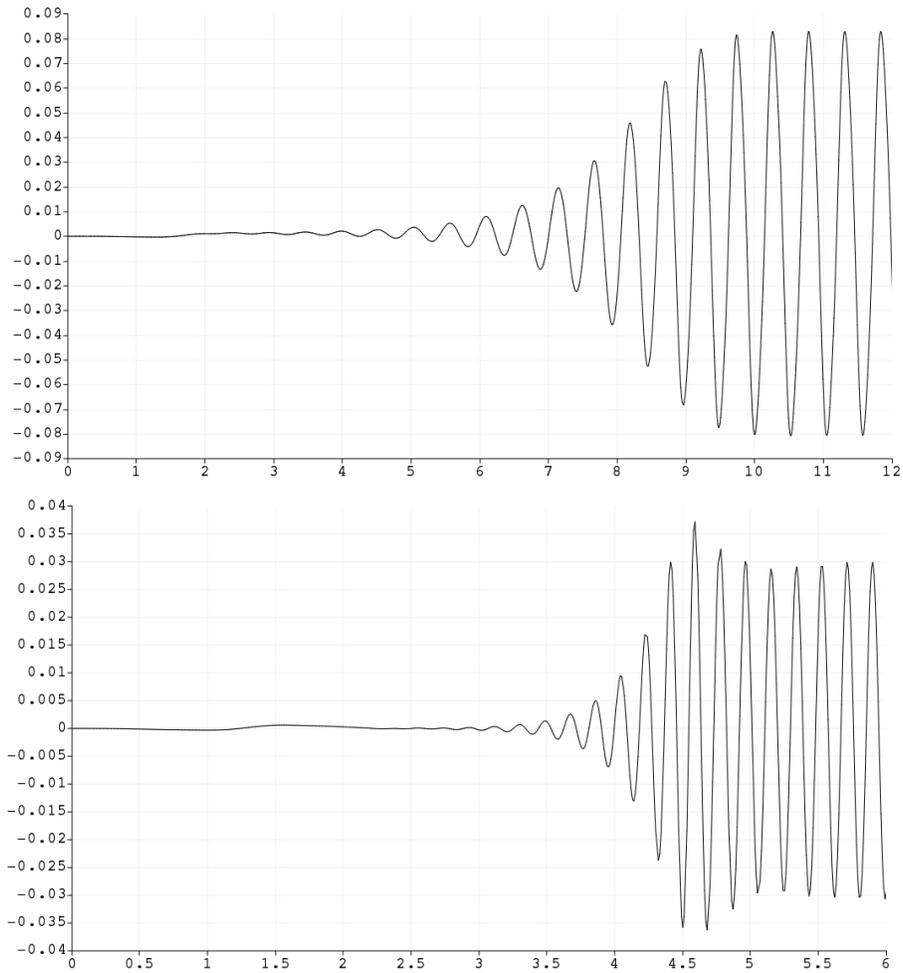
Figure 1.5: In this figure, we show the vertical displacement of FSI2 (top) and FSI3 (down) at the end of the elastic structure over time. In both examples we can see that the oscillations are proceeded by a long stagnation phase. FSI2 example needs a longer time to develop the oscillations and once they develop, the vertical displacement reaches higher values with larger periods.

Chapter 2

Discretization of Coupled Problems

A suitable choice of time and space discretizations is crucial for the stability and accuracy of finite element formulations. This can be especially challenging for coupled problems. The introduction of adaptive time-stepping procedures can further complexify these problems. One of the tools that may help with the analysis, is looking at classical time-stepping schemes from the perspective of the space-time framework. This way, we can have access to tools from the finite element community more commonly used for space discretization. This will be especially important in later chapters where we will derive an error estimator used as a criterion in an adaptive time-stepping algorithm. In the case of space discretization, although we are going to employ classical approaches, handling the transfer of interface values between different meshes is non-trivial.

We will start this chapter by tackling time discretization and will first discuss the specific needs of coupled problems. Then we will explain how to embed time-stepping methods in a space-time formulation and present ways to cope with different time meshes for each of the coupled systems. A crucial part will be the introduction of an interpolation operator which will enable a transfer of values between the two meshes. Finally, we will derive semi-discrete formulations of our model problems including the FSI problem which will require the most care. Then we will consider the space discretization and shortly present methods for reassigning the interface coupling conditions between different space meshes.

2.1   Time Discretization

Choosing an appropriate time-stepping scheme for coupled systems is particularly challenging since each of the problems calls for different properties in the scheme. Indeed, time discretization of parabolic equations leads to highly stiff systems which need preferably implicit stable schemes. Hyperbolic problems have the feature of energy conservation and thus require schemes with little numerical dissipation. This tension can be illustrated with the help of the class of time-stepping schemes called θ-schemes. First, let us define a simple coupled

system consisting of a parabolic and a hyperbolic equation

$$\int_I \left\{ \left\langle \partial_t u^1, \varphi^1 \right\rangle + a^1 \left(\vec{\mathbf{U}}, \mathbf{\Phi}^1 \right) \right\} \mathrm{d}t = \int_I \left(f^1(\vec{\mathbf{U}}), \mathbf{\Phi}^1 \right) \mathrm{d}t,$$

$$\int_I \left\{ \left\langle \partial_t u^2, \varphi^2 \right\rangle + \left\langle \partial_t v^2, \psi^2 \right\rangle + a^2 \left(\vec{\mathbf{U}}, \mathbf{\Phi}^2 \right) \right\} \mathrm{d}t = \int_I \left(f^2(\vec{\mathbf{U}}), \mathbf{\Phi}^2 \right) \mathrm{d}t,$$

(2.1)

where the trial and test functions are defined as

$$\vec{\mathbf{U}}^1 := \begin{pmatrix} u^1 \\ v^1 \end{pmatrix}, \quad \vec{\mathbf{U}}^2 := \begin{pmatrix} v^2 \\ u^2 \end{pmatrix}, \quad \vec{\mathbf{U}} := \begin{pmatrix} \vec{\mathbf{U}}^1 \\ \vec{\mathbf{U}}^2 \end{pmatrix},$$

$$\mathbf{\Phi}^1 := \begin{pmatrix} \varphi^1 \\ \psi^1 \end{pmatrix}, \quad \mathbf{\Phi}^2 := \begin{pmatrix} \varphi^2 \\ \psi^2 \end{pmatrix}, \quad \mathbf{\Phi} := \begin{pmatrix} \mathbf{\Phi}^1 \\ \mathbf{\Phi}^2 \end{pmatrix}$$

For now, we assume that these problems are defined over the same space domain Ω with homogeneous Dirichlet boundary conditions. We can define the space of solutions of this system as (2.1)

$$X := \left\{ \phi \in L^2(I, V), \partial_t \varphi \in L^2(I, V^*) \right\},$$

where as space V we choose $V = H_0^1(\Omega)$ and its dual space is given by $V^* = H^{-1}(\Omega)$. One can introduce discretization of the time interval $I = [0, T]$

$$0 = t_0 < t_1 < ... < t_N = T, \quad k_n := t_n - t_{n-1}, \quad I_n := (t_{n-1}, t_n]$$

with a grid size $k := \max_{n=1,...,N} k_n$. Then, for a given $\theta \in [0, 1]$ we obtain the scheme

$$
\begin{aligned}
&\left(u_k^1(t_n) - u_k^1(t_{n-1}), \varphi_k^1(t_n) \right) + \left(u_k^2(t_n) - u_k^2(t_{n-1}), \varphi_k^2(t_n) \right) \\
&+ \left(v_k^2(t_n) - v_k^2(t_{n-1}), \psi_k^2(t_n) \right) + \theta k_n a^1 \left(\vec{\mathbf{U}}_k(t_n), \mathbf{\Phi}_k^1(t_n) \right) \\
&+ (1-\theta) k_n a^1 \left(\vec{\mathbf{U}}_k(t_{n-1}), \mathbf{\Phi}_k^1(t_n) \right) + \theta k_n a^2 \left(\vec{\mathbf{U}}_k(t_n), \mathbf{\Phi}_k^2(t_n) \right) \\
&+ (1-\theta) k_n a^2 \left(\vec{\mathbf{U}}_k(t_{n-1}), \mathbf{\Phi}_k^2(t_{n-1}) \right) \\
&\quad = \theta k_n \left(f^1(\vec{\mathbf{U}}_k(t_n)), \mathbf{\Phi}_k^1(t_n) \right) + (1-\theta) k_n \left(f^1(\vec{\mathbf{U}}_k(t_{n-1})), \mathbf{\Phi}_k^1(t_n) \right) \\
&\quad + \theta k_n \left(f^2(\vec{\mathbf{U}}_k(t_n)), \mathbf{\Phi}_k^2(t_n) \right) + (1-\theta) k_n \left(f^2(\vec{\mathbf{U}}_k(t_{n-1})), \mathbf{\Phi}_k^2(t_n) \right)
\end{aligned}
$$

(2.2)

By setting $\theta = 0$ we get the explicit Euler scheme, $\theta = \frac{1}{2}$ leads to the Crank-Nicolson and $\theta = 1$ corresponds to the implicit Euler. Based on a deeper study of this class of schemes, one can deduce three basic observations:

1. **Convergence rate** - only the Crank-Nicolson scheme has the second order of convergence. Other schemes converge linearly.

2. **Stability** - the more implicit the scheme is, the more stable it is. Consequently, the implicit Euler has the best stability properties.

3. **Numerical energy dissipation** - Crank-Nicolson has the lowest energy dissipation.

For details, we refer to [25]. From this perspective, the most sensible way to resolve these tensions is to choose the Crank-Nicolson scheme. Its A-stability is enough for most of the applications. However, for more challenging problems it may not be able to deal with errors accumulated over time. Further, it may not sufficiently smooth the solution for the irregular initial data. Thus, one can try to slightly shift the Crank-Nicolson to the implicit side. Indeed, in [28] and [29] it was discovered that by replacing some of the Crank-Nicolson time-steps with smaller implicit Euler ones, we can recover the second order convergence rate. This scheme, after its author, is called the *Rannacher time marching scheme.* Further, in [30] the authors were able to prove second order error estimates for nonstationary Navier-Stokes equations. Interestingly, for a classic version of the Crank-Nicolson scheme, the authors had to assume $k \leqslant ch^{\frac{2}{3}}$, whereas, for the modified version of the scheme, this assumption was not necessary.

A similar idea stands behind *shifted Crank-Nicolson scheme* [25]. There, instead of introducing implicit Euler steps one uniformly shifts θ using

$$\theta := \frac{1}{2} + \mathcal{O}(k)$$

It is enough to achieve a strongly A-stable scheme which still has a second order convergence rate and very low numerical energy dissipation. Because of its simplicity, this is also the scheme that we decided to use to discretize our FSI model problem.

Another way to construct a time-stepping scheme with good properties is by using a so-called, *fractional θ-scheme*, whereby carefully splitting a time-step into smaller parts and choosing appropriate θ parameters for each of them one can recover second order convergence and low numerical energy dissipation. Since the chosen θ values belong to $\theta \in (\frac{1}{2}, 1]$, the resulting scheme is strongly A-stable. This approach has been introduced in [31] and further extended in [32].

2.1.1 Space-time Approach

So far, we have been discussing time discretization methods, where the time component was discretized first before the space component. It is also possible to use Galerkin discretization simultaneously for both space and time, so-called *space-time discretization.* This approach allows access to tools from the finite element community such as robust error estimates and adjoint formulations. However, the computational costs of space-time methods, caused by an additional dimension in the discretization, can be prohibitive. Fortunately, up to quadratures of nonlinear terms, one can express some of the classic single-step time-stepping methods within the space-time framework.

This idea has been explored in [19] and then further extended in [20]. By embedding the definition of time-stepping schemes in the Galerkin method framework, the authors were

able to derive both a priori and a posteriori estimates in the energy and L^2 norms. We will demonstrate this idea using both discontinuous and continuous approaches. The first step is defining polynomial spaces over the time interval I similar to spaces conventionally used for the finite element method

$$X_k^r = \left\{ \varphi \in C(\bar{I}, V) \mid \varphi|_{I_n} \in \mathcal{P}_r(V) \text{ for all } n = 1, \ldots, N \right\},$$

$$Y_k^r = \left\{ \varphi \in L^2(\bar{I}, V) \mid \varphi|_{I_n} \in \mathcal{P}_r(V) \text{ for all } n = 1, \ldots, N \text{ and } \varphi(0) \in L^2(\Omega) \right\},$$

where $\mathcal{P}_r(V)$ denotes a space of piecewise polynomials of degree r and value in a Sobolev space V, which here is chosen as $V = H_0^1(\Omega)$.

2.1.1.1 Discontinuous Galerkin Methods

In discontinuous methods, often denoted as *dG(r)*, both the trial and test functions belong to Y_k^r. In a special case of this approach, dG(0), we consider the space Y_k^0 that contains piecewise constant functions. Then, on every interval I_n:

$$\vec{\mathbf{U}}_k(t)\Big|_{I_n} = \vec{\mathbf{U}}_k(t_n), \qquad \boldsymbol{\Phi}_k(t)\Big|_{I_n} = \boldsymbol{\Phi}_k(t_n). \tag{2.3}$$

A corresponding projection operator is defined by

$$\left(\mathbf{i}_k^{IE} \vec{\mathbf{U}} \right)(t)\Big|_{I_n} = \vec{\mathbf{U}}(t_n).$$

The space Y_k^r is the image of this operator and therefore for any $\vec{\mathbf{U}}_k \in Y_k^r$ we have

$$\left(\mathbf{i}_k^{IE} \vec{\mathbf{U}}_k \right) = \vec{\mathbf{U}}_k.$$

At the initial time, we set

$$\left(\mathbf{i}_k^{IE} \vec{\mathbf{U}} \right)(0) = \vec{\mathbf{U}}(0).$$

This choice of Galerkin spaces leads to a nonconforming formulation, where $Y_k^r \notin X$. Nevertheless, one can introduce a conforming semi-discrete formulation of the problem (2.1). We introduce jump terms typical for discontinuous Galerkin methods

$$\vec{\mathbf{U}}_k^+(t) := \lim_{s \searrow t} \vec{\mathbf{U}}_k(s), \qquad \vec{\mathbf{U}}_k^-(t) := \lim_{s \nearrow t} \vec{\mathbf{U}}_k(s), \qquad [\vec{\mathbf{U}}_k](t) := \vec{\mathbf{U}}_k^+(t) - \mathbf{U}_k^-(t),$$

which in turn can be used in the following conforming problem over each of the intervals I_n

$$\int_{I_n} \left\{ \left\langle \partial_t u_k^1, \varphi_k^1 \right\rangle + a^1 \left(\vec{\mathbf{U}}_k, \mathbf{\Phi}_k^1 \right) \right\} \mathrm{d}t + \left([u_k^1](t_{n-1}), \varphi_k^{1,+}(t_{n-1}) \right)$$

$$= \int_{I_n} \left(f^1(\vec{\mathbf{U}}_k), \mathbf{\Phi}_k^1 \right) \mathrm{d}t,$$

$$\int_{I_n} \left\{ \left\langle \partial_t u_k^2, \varphi_k^2 \right\rangle + \left\langle \partial_t v_k^2, \psi_k^2 \right\rangle + a^2 \left(\vec{\mathbf{U}}_k, \mathbf{\Phi}_k^2 \right) \right\} \mathrm{d}t + \left([u_k^2](t_{n-1}), \varphi_k^{2,+}(t_{n-1}) \right) \qquad (2.4)$$

$$+ \left([v_k^2](t_{n-1}), \psi_k^{2,+}(t_{n-1}) \right)$$

$$= \int_{I_n} \left(f^2(\vec{\mathbf{U}}_k), \mathbf{\Phi}_k^2 \right) \mathrm{d}t.$$

If we assume that the forms a^1, a^2 as well as the functions f^1 and f^2 are linear in $\vec{\mathbf{U}}$ and do not explicitly depend on time t, then the discretization of the problem (2.1) using the formulation above is equivalent to the implicit Euler scheme. Although on every interval I_n, the time derivative is zero

$$\int_{I_n} \left\langle \partial_t \vec{\mathbf{U}}_k, \mathbf{\Phi}_k \right\rangle \mathrm{d}t = 0,$$

we can use the jump term to derive the finite difference quotient conventionally associated with time-stepping

$$\left([\vec{\mathbf{U}}_k](t_{n-1}), \mathbf{\Phi}_k^+(t_{n-1}) \right) = \left(\vec{\mathbf{U}}_k(t_n) - \vec{\mathbf{U}}_k(t_{n-1}), \mathbf{\Phi}_k(t_n)) \right).$$

In the case of nonlinearities, we lose the equivalence between the implicit Euler and dG(0) method. However, the error can be split

$$\left\| \vec{\mathbf{U}} - \vec{\mathbf{U}}_k \right\| \leqslant \left\| \vec{\mathbf{U}} - \vec{\mathbf{U}}_G \right\| + \left\| \vec{\mathbf{U}}_G - \vec{\mathbf{U}}_k \right\|, \qquad (2.5)$$

where $\vec{\mathbf{U}}_k$ denotes the solution coming from the implicit Euler scheme and we refer to $\mathbf{U}^G$ as the result of the Galerkin scheme given by (2.3). In general, these two errors are of the same order and thus, have to be accounted for separately. If the nonlinearity only comes from the right hand side, one can solve this problem by approximating the integrals

$$\int_{I_n} \left(f^1(\vec{\mathbf{U}}_k), \mathbf{\Phi}_k^1 \right) \mathrm{d}t, \quad \int_{I_n} \left(f^2(\vec{\mathbf{U}}_k), \mathbf{\Phi}_k^2 \right) \mathrm{d}t$$

using higher order quadrature rules.

2.1.1.2 Continuous Galerkin Methods

Similarly, we can consider continuous setup, where we pick continuous trial functions from space X_k^r and discontinuous test functions of one order lower from Y_k^{r-1}. That method is

referred to as $cG(r)$. Here as well one can find one of the θ-schemes discussed earlier. Indeed, for cG(1) trial functions are piecewise continuous in time

$$\vec{\mathbf{U}}_k(t)\Big|_{I_n} = \frac{t_n - t}{t_n - t_{n-1}}\vec{\mathbf{U}}_k(t_{n-1}) + \frac{t - t_{n-1}}{t_n - t_{n-1}}\vec{\mathbf{U}}_k(t_n) \tag{2.6}$$

and test functions, the same as in the previous example, are piecewise constant in time

$$\boldsymbol{\Phi}_k(t)\Big|_{I_n} = \boldsymbol{\Phi}_k(t_n). \tag{2.7}$$

The following projection operator will be associated with this method

$$\left(\mathbf{i}_k^{CN}\vec{\mathbf{U}}\right)(t)\Big|_{I_n} = \frac{t_n - t}{t_n - t_{n-1}}\vec{\mathbf{U}}(t_{n-1}) + \frac{t - t_{n-1}}{t_n - t_{n-1}}\vec{\mathbf{U}}(t_n),$$

where for any $\vec{\mathbf{U}}_k \in X_k^r$ we have $\left(\mathbf{i}_k^{CN}\vec{\mathbf{U}}_k\right) = \vec{\mathbf{U}}_k$ and at the initial time we set

$$\left(\mathbf{i}_k^{CN}\vec{\mathbf{U}}_k\right)(0) = \vec{\mathbf{U}}_k(0).$$

This choice of trial and test spaces leads to a conforming method, where the problem (2.1) does not have to be adjusted other that replacing the continuous functions with semi-discrete ones. After performing time integration using these functions we obtain

$$\int_{I_n} \left\langle \partial_t \vec{\mathbf{U}}_k, \boldsymbol{\Phi}^k \right\rangle \mathrm{d}t = \int_{I_n} \left(\frac{\vec{\mathbf{U}}_k(t_n)}{t_n - t_{n-1}} - \frac{\vec{\mathbf{U}}_k(t_{n-1})}{t_n - t_{n-1}}, \boldsymbol{\Phi}_k(t_n)\right)\mathrm{d}t$$

$$= \left(\vec{\mathbf{U}}_k(t_n) - \vec{\mathbf{U}}_k(t_{n-1}), \boldsymbol{\Phi}_k(t_n)\right).$$

Similarly, assuming linearity of the form a

$$\int_{I_n} a\left(\vec{\mathbf{U}}_k, \boldsymbol{\Phi}_k\right)\mathrm{d}t$$

$$= \int_{I_n} \frac{t_n - t}{t_n - t_{n-1}} a\left(\vec{\mathbf{U}}_k(t_{n-1}), \boldsymbol{\Phi}_k(t_n)\right)\mathrm{d}t + \int_{I_n} \frac{t - t_{n-1}}{t_n - t_{n-1}} a\left(\vec{\mathbf{U}}_k(t_n), \boldsymbol{\Phi}_k(t_n)\right)\mathrm{d}t$$

$$= \frac{1}{2}k_n a\left(\vec{\mathbf{U}}_k(t_{n-1}), \boldsymbol{\Phi}_k(t_n)\right) + \frac{1}{2}k_n a\left(\vec{\mathbf{U}}_k(t_n), \boldsymbol{\Phi}_k(t_n)\right).$$

From these computations, one can clearly spot a connection between this scheme and the Crank-Nicolson scheme. In truth, under the same assumptions on linearity as in the case of the previously discussed discontinuous Galerkin method, the Crank-Nicolson method and the cG(1) method are equivalent. The inconsistencies arising from the nonlinear terms can be dealt with in a similar way using (2.5). Finally, we would like to point out that by choosing appropriate test and trial spaces along with quadrature rules, one can recover θ-schemes for all values of θ. For details, we would like to refer to [25].

2.1.2 Multirate Time-stepping

We can further extend the idea of time-stepping methods for coupled problems by introducing multirate time-stepping, where each of the problems has a different time mesh. It is not a new concept and has its roots in domain decomposition methods. One of the first articles on the matter is [7]. A detailed overview of the domain decomposition methods one can find in the book [8]. In [9] the authors present methods more specifically designed for coupled systems. For another outlook on the literature, we would like to refer to the introduction of Chapter 3 devoted to decoupling algorithms. These two topics are closely related since decoupling a coupled problem is necessary to introduce independent time-stepping strategies.

Establishing multirate time-stepping methods for coupled systems requires resolving the tension between two factors. First, we have to guarantee that the solutions fulfill the coupling conditions. Second, we need to provide enough flexibility to enable independent time-stepping in each of the problems. We decided to solve this problem by introducing a hierarchy of time meshes. The coarse mesh, which is the same for both of the problems, assures that the coupling is enforced. The fine meshes introduce independent time-step sizes in each of the domains. The coarse mesh is defined by

$$0 = t_0 < t_1 < ... < t_N = T, \quad k_n := t_n - t_{n-1}, \quad I_n := (t_{n-1}, t_n].$$

This mesh will be referred to as *macro time mesh* and to its time-steps as *macro time-steps*. Each of the intervals I_n is further independently split in the fluid domain

$$t_{n-1} = t_{n,0}^f < ... < t_{N_n^f}^f = t_n, \quad k_{n,m}^f := t_{n,m}^f - t_{n,m-1}^f, \quad I_{n,m}^f := (t_{n,m}^f, t_{n,m-1}^f]$$

and in the solid domain

$$t_{n-1} = t_{n,0}^s < ... < t_{N_n^s}^s = t_n, \quad k_{n,m}^s := t_{n,m}^s - t_{n,m-1}^s, \quad I_{n,m}^s := (t_{n,m}^s, t_{n,m-1}^s].$$

We will use the names *micro time mesh* and *micro time-steps* while referring to these meshes. An example of this kind of partitioning is sjowm in Figure 2.1. The global mesh sizes are defined by

$$k^f := \max k_{n,m}^f, \quad k^s := \max k_{n,m}^s, \quad k := \max\{k^f, k^s\}.$$

One can also define a global number of time-steps in each subdomain

$$N^f := \sum_{n=1}^{N} N_n^f, \quad N^s := \sum_{n=1}^{N} N_n^s.$$

We assume that micro time-steps are introduced only when necessary. That means that for every interval I_n the only points that coincide in the fluid and solid micro mesh are the ends of the interval t_{n-1} and t_n. Further, if we assume that these meshes are a result of an adaptive algorithm, where the time-steps can be refined only in the middle, then for each I_n there is no micro time-stepping in either the fluid or the solid domain. This mechanism is explained
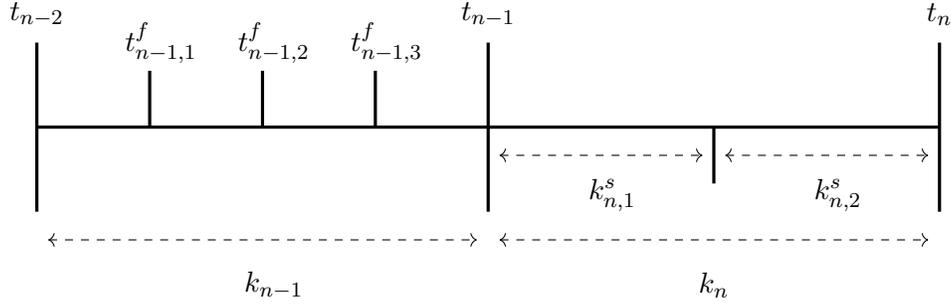
Figure 2.1: We show a snapshot of time partitioning with two macro time-steps. In the first one, we introduce four micro time-steps in the fluid domain. In the second one, we have two micro time-steps in the solid domain.
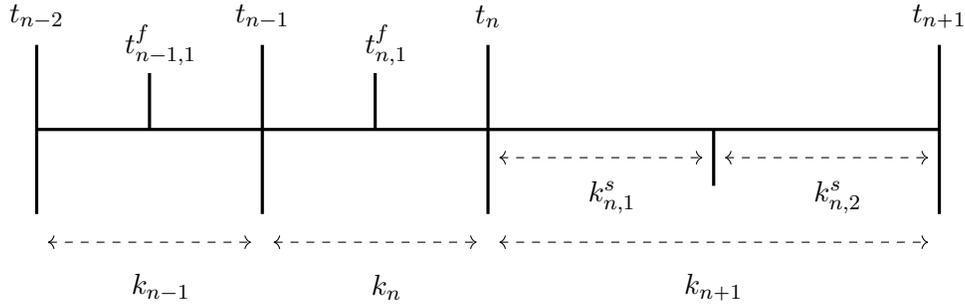

Figure 2.2: Here we show the refinement of the time mesh presented in Figure 2.1 by splitting I_n in the solid subproblem. Since the time mesh corresponding to the fluid subproblem was already split at this point (point $t^f_{n-1,2}$ in Figure 2.1), we can introduce a new macro time-step.

in Figures 2.1 and 2.2. This adaptive procedure is further discussed in Section 4.2.5, where we introduce an adaptivity algorithm.

To enable the transfer of values between the time meshes, which is necessary to evaluate the coupling conditions, we need to introduce appropriate operators. Based on our needs, we decided to take average operators

$$\left(\mathbf{I}^f_k\vec{\mathbf{U}}^f\right)(t)\Big|_{I^f_{n,m}} := \frac{1}{k^f_{n,m}}\int_{I^f_{n,m}}\vec{\mathbf{U}}^f\,\mathrm{d}t, \quad \left(\mathbf{I}^s_k\vec{\mathbf{U}}^s\right)(t)\Big|_{I^s_{n,m}} := \frac{1}{k^s_{n,m}}\int_{I^s_{n,m}}\vec{\mathbf{U}}^s\,\mathrm{d}t. \quad (2.8)$$

We will also use the shorter notation $\mathbf{I}_k := (\mathbf{I}^f_k, \mathbf{I}^s_k)^T$. At the initial time, we set

$$\left(\mathbf{I}_k\vec{\mathbf{U}}\right)(0) = \vec{\mathbf{U}}(0).$$

The key property of this operator is that the error given by this operator has an average equal

to zero

$$\int_{I_n} \left(\vec{\mathbf{U}} - \mathbf{I}_k \vec{\mathbf{U}} \right) \mathrm{d}t = \mathbf{0}.$$

It will be crucial in Chapter 5, where stability estimates will be derived.

2.1.3 Semi-discrete Formulations

Using the information above one can derive semi-discrete formulations for each of our model problems. In all of them, we either use the Crank-Nicolson or the shifted Crank-Nicolson time-stepping scheme. In both cases, we will use the correspondence to the Cg(1) scheme. Therefore, as the trial spaces for the interface coupled problem, we take

$$X_k^{f,1} = \left\{ \varphi \in C(\bar{I}, V^f) \middle| \varphi|_{I_{n,m}^f} \in \mathcal{P}_1(V^f) \text{ for all } I_{n,m}^f \subset I \right\},$$

$$X_k^{s,1} = \left\{ \varphi \in C(\bar{I}, V^s) \middle| \varphi|_{I_{n,m}^s} \in \mathcal{P}_1(V^s) \text{ for all } I_{n,m}^s \subset I \right\}.$$

The test spaces for the interface problems are given by

$$Y_k^{f,0} = \left\{ \varphi \in L^2(\bar{I}, V^f) \middle| \varphi|_{I_{n,m}^f} \in \mathcal{P}_0(V^f) \text{ for all } I_{n,m}^f \subset I \text{ and } \varphi(0) \in L^2(\Omega^f) \right\},$$

$$Y_k^{s,0} = \left\{ \varphi \in L^2(\bar{I}, V^s) \middle| \varphi|_{I_{n,m}^s} \in \mathcal{P}_0(V^s) \text{ for all } I_{n,m}^s \subset I \text{ and } \varphi(0) \in L^2(\Omega^s) \right\}.$$

Similarly, for the thermoelasticity problem, we will replace the $X_k^{f,1}$ by

$$X_k^{T,1} = \left\{ \varphi \in C(\bar{I}, V^T) \middle| \varphi|_{I_{n,m}^T} \in \mathcal{P}_1(V^T) \text{ for all } I_{n,m}^s \subset I \right\}.$$

Instead of $Y_k^{f,0}$, we will use

$$Y_k^{T,0} = \left\{ \varphi \in L^2(\bar{I}, V^T) \middle| \varphi|_{I_{n,m}^T} \in \mathcal{P}_0(V^T) \text{ for all } I_{n,m}^T \subset I \text{ and } \varphi(0) \in L^2(\Omega^s) \right\}.$$

Examples of functions given by these spaces are given in Figure 2.3.

As the result of this time discretization procedure, we will obtain the following variational problem for the interface problem

Problem 2.1. *Find* $\vec{U}_k \in X_k$ *such that*

$$\begin{aligned} B_k^f(\vec{U}_k)(\mathbf{\Phi}_k^f) &= F_k^f(\mathbf{\Phi}_k^f), && \text{for all } \mathbf{\Phi}_k^f \in Y_k^f, \\ B_k^s(\vec{U}_k)(\mathbf{\Phi}_k^s) &= F_k^s(\mathbf{\Phi}_k^s), && \text{for all } \mathbf{\Phi}_k^s \in Y_k^s. \end{aligned}$$

and the formulation

Problem 2.2. *Find* $\vec{U}_k \in X_k$ *such that*

$$\begin{aligned} B_k^s(\vec{U}_k, T_k)(\mathbf{\Phi}_k) &= F_k^s(\mathbf{\Phi}_k), && \text{for all } \mathbf{\Phi}_k \in X_k^s, \\ B_k^T(\vec{U}_k, T_k)(\Lambda_k) &= 0, && \text{for all } \Lambda_k \in X_k^T. \end{aligned}$$
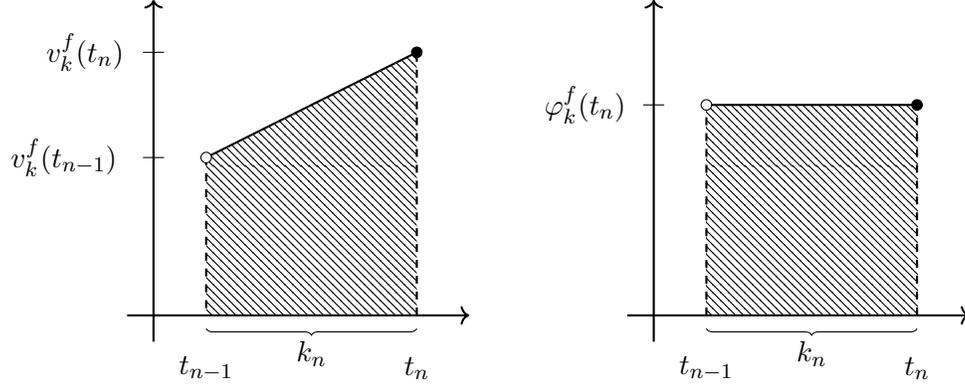
for the thermoelsticity problem.

Figure 2.3: We show an example of a piecewise linear trial function v_k^f (left) and a piecewise constant test function φ_k^f (right).

2.1.3.1   Heat and Wave Equation

We will start with the coupling of the heat and wave equations and discretize this problem using the classical Crank-Nicolson time-stepping scheme. Because this problem is linear in time, the Cg(1) time-stepping scheme and the Crank-Nicolson scheme are equivalent up to the quadrature of the right hand side. We will use notation

$$X_k^f := \left(X_k^{f,1}\right)^2, \quad X_k^s := \left(X_k^{s,1}\right)^2, \quad X_k := X_k^f \times X_k^s$$

while referring to the trial spaces. Similarly, we will denote

$$Y_k^f := \left(Y_k^{f,0}\right)^2, \quad Y_k^s := \left(Y_k^{s,0}\right)^2, \quad Y_k := Y_k^f \times Y_k^s.$$

As trial and test functions we take $\vec{\mathbf{U}}_k \in X_k$ and $\boldsymbol{\Phi}_k \in Y_k$, respectively. Then, we define the bilinear forms as

$$B_k^f(\vec{\mathbf{U}}_k)(\boldsymbol{\Phi}_k^f) := \int_I \left\{ \left(\partial_t u_k^f, \varphi_k^f\right)_f + a^f\left((\mathbf{U}_k^f, \mathbf{I}_k^f \mathbf{U}_k^s)^T, \boldsymbol{\Phi}_k^f\right) \right\} \mathrm{d}t,$$

$$B_k^s(\vec{\mathbf{U}}_k)(\boldsymbol{\Phi}_k^s) := \int_I \left\{ \left(\partial_t u_k^s, \varphi_k^s\right)_s + \left(\partial_t v_k^s, \psi_k^s\right)_s + a^s\left((\mathbf{I}_k^s \mathbf{U}_k^f, \mathbf{U}_k^s)^T, \boldsymbol{\Phi}_k^s\right) \right\} \mathrm{d}t,$$

The right hand side is integrated in time using 2-point Gaussian quadrature. Assuming there is no micro-time-stepping in I_n, this quadrature rule is given by

$$\int_{I_n} g(t)\,\mathrm{d}t \approx \frac{k_n}{2}\left(g\left(\frac{k_n}{2\sqrt{3}} + \bar{t}_n\right) + g\left(-\frac{k_n}{2\sqrt{3}} + \bar{t}_n\right)\right),$$

where $\bar{t}_n = \frac{t_n - t_{n_1}}{2}$. Since this quadrature has one order higher convergence rate than the Crank-Nicolson method, we can ensure that the quadrature error is also of a higher order than the Galerkin error. Thus, it can be neglected.

2.1.3.2 FSI Problem

To discretize in time the FSI example, we will use the shifted Crank-Nicolson time-stepping scheme. Since this is a highly nonlinear system, its discretization introduces a quadrature error, whose order of convergence is the same as the order of convergence of the scheme itself. This example also carries some specific difficulties. One of them is incompressibility. The pressure in the systems describing incompressible fluids can be understood as a Lagrange multiplier to a corresponding system solved in the space of divergence-free functions

$$V_0 := \left\{ \varphi \in H^1(\Omega) \mid \text{div } \varphi = 0 \right\}.$$

As a consequence, we have two possible ways to discretize the pressure components. The first option is to discretize them using the shifted Crank-Nicolson scheme. Then, before applying the ALE transformation, the incompressibility condition is discretized using

$$\left(\frac{1}{2} - \mathcal{O}(k) \right) k_n \big(\text{div } \mathbf{v}_k^f(t_{n-1}), \omega_k^f(t_n) \big)_f + \left(\frac{1}{2} + \mathcal{O}(k) \right) k_n \big(\text{div } \mathbf{v}_k^f(t_n), \omega_k^f(t_n) \big)_f.$$

By testing this formulation with a function of the form

$$\omega_k^f(t_n) := \text{div } \mathbf{v}_k^f(t_n) - \text{div } \mathbf{v}_k^f(t_{n-1}),$$

we obtain

$$\left\| \text{div } \mathbf{v}_k^f(t_N) \right\|^2 = \left\| \text{div } \mathbf{v}_k^f(0) \right\|^2 + \mathcal{O}(k).$$

Even though in our example we set $\mathbf{v}_k^f(0) = \mathbf{0}$, which does not violate the incompressibility condition, this identity still shows that any inevitable truncation errors are not damped and instead carried through the whole time interval. A remedy to this problem is using the second approach, where we first restrict the equations to the space V_0, discretize it there using the shifted Crank-Nicolson scheme and then apply the Lagrange multiplier. This way leads to a discretization, where all the terms involving the pressure p_k^f or its test function ω_k^f are fully implicit. Therefore, we take $p_k^f \in Y_k^{f,0}$ as well as

$$X_k^f := \left(X_k^{f,1} \right)^{2d} \times Y_k^{f,0}, \quad X_k^s := \left(X_k^{s,1} \right)^{2d}, \quad X_k := X_k^f \times X_k^s$$

and

$$Y_k^f := \left(Y_k^{f,0} \right)^{2d+1}, \quad Y_k^s := \left(Y_k^{s,0} \right)^{2d}, \quad Y_k := Y_k^f \times Y_k^s.$$

To distinguish between the terms discretized using the shifted Crank-Nicolson and the implicit Euler schemes, we will split the forms

$$a^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) = a_{CN}^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) + a_{IE}^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f),$$
$$a^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) = a_{CN}^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) + a_{IE}^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s).$$

The form $a_{IE}^f(\cdot)(\cdot)$ contains all the terms including either the pressure or the corresponding test function. The form $a_{IE}^s(\cdot)(\cdot)$ contains the parts of the coupling conditions including pressure. All of these forms are stated below

$$
\begin{aligned}
a_{CN}^f(\vec{\mathbf{U}})(\boldsymbol{\Phi}^f) :=& \left(2J\boldsymbol{\epsilon}^f\mathbf{F}^{-T}, \nabla\boldsymbol{\varphi}^f\right)_f + \left(\rho^f J\mathbf{F}^{-1}\mathbf{v}^f \cdot \nabla\mathbf{v}^f, \boldsymbol{\varphi}^f\right)_f + \left(S\nabla\mathbf{u}^f, \nabla\boldsymbol{\psi}^f\right)_f \\
& - \left\langle 2J\boldsymbol{\epsilon}^f\mathbf{F}^{-T}\cdot\vec{\mathbf{n}}_f, \boldsymbol{\varphi}^f\right\rangle_\Gamma - \left\langle S\partial_{\vec{\mathbf{n}}_f}\mathbf{u}^f, \boldsymbol{\psi}^f\right\rangle_\Gamma + \gamma\rho^f\nu\left\langle\mathbf{v}^f - \mathbf{v}^s, \boldsymbol{\varphi}^f\right\rangle_\Gamma \\
& + \gamma\left\langle\mathbf{u}^f - \mathbf{u}^s, \boldsymbol{\psi}^f\right\rangle_\Gamma, \\
a_{IE}^f(\vec{\mathbf{U}})(\boldsymbol{\Phi}^f) :=& -\left(Jp^f\mathbf{F}^{-T}, \nabla\boldsymbol{\varphi}^f\right)_f + \left(J\mathrm{tr}(\mathbf{F}^{-1}\nabla\mathbf{v}^f), \omega^f\right)_f + \left\langle Jp^f\mathbf{F}^{-T}\cdot\vec{\mathbf{n}}_f, \boldsymbol{\varphi}^f\right\rangle_\Gamma, \\
a_{CN}^s(\vec{\mathbf{U}})(\boldsymbol{\Phi}^s) :=& \left(\mathbf{F}\boldsymbol{\sigma}^s, \nabla\boldsymbol{\varphi}^s\right)_s - \left(\mathbf{v}^s, \boldsymbol{\psi}^s\right)_s + \left\langle 2J\boldsymbol{\epsilon}^f, \mathbf{F}^{-T}\cdot\vec{\mathbf{n}}_f, \varphi^s\right\rangle_\Gamma, \\
a_{IE}^s(\vec{\mathbf{U}})(\boldsymbol{\Phi}^s) :=& -\left\langle Jp^f, \mathbf{F}^{-T}\cdot\vec{\mathbf{n}}_f, \boldsymbol{\varphi}^s\right\rangle_\Gamma
\end{aligned}
\tag{2.9}
$$

for

$$
\boldsymbol{\epsilon} := \rho^f\nu^f\frac{1}{2}\left(\nabla\mathbf{v}^f\mathbf{F}^{-1} + \mathbf{F}^{-T}(\nabla\mathbf{v}^f)^T\right).
$$

Discretization in time of an FSI problem in ALE coordinates brings its own set of challenges. One has to be especially careful while discretizing the time derivative terms, namely the terms

$$
\int_{I_n}\left\langle\rho^f J\partial_t\mathbf{v}^f, \boldsymbol{\varphi}^f\right\rangle_f \mathrm{d}t \quad\text{and}\quad \int_{I_n}\left\langle\rho^f J\mathbf{F}^{-1}\nabla\mathbf{v}^f\partial_t\mathbf{u}^f, \boldsymbol{\varphi}^f\right\rangle_f \mathrm{d}t. \tag{2.10}
$$

Time discretization of the latter can especially cause problems. As far as we know, there does not exist a strict study on this issue and existing approaches are instead based on experience. We will present the two most common choices. Both rely on taking an average between the old t_{n-1} and the new t_n time-steps while computing the terms in front of the time derivative. In the first, we take the average expression

$$
\begin{aligned}
&\frac{1}{2}\left(\rho^f J_k(t_n)\mathbf{F}_k^{-1}(t_n)\nabla\mathbf{v}_k^f(t_n)(\mathbf{u}_k^f(t_n) - \mathbf{u}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f \\
&+ \frac{1}{2}\left(\rho^f J_k(t_{n-1})\mathbf{F}_k^{-1}(t_{n-1})\nabla\mathbf{v}_k^f(t_{n-1})(\mathbf{u}_k^f(t_n) - \mathbf{u}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f.
\end{aligned}
$$

In the second, we choose as the time argument the time average $\bar{t}_n = \frac{t_n - t_{n-1}}{2}$ and obtain

$$
\left(\rho^f J_k(\bar{t}_n)\mathbf{F}_k^{-1}(\bar{t}_n)\nabla\mathbf{v}_k^f(\bar{t}_n)(\mathbf{u}_k^f(t_n) - \mathbf{u}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f.
$$

Both of these possibilities are further explored in [33] and [34]. According to these publications, both treatments of time derivatives yield similar results. We have decided to use the first approach because its implementation in our case proved to be less troublesome. In the first term from (2.10), we decided to treat the determinant J in a fully implicit way and therefore obtain

$$
\left(\rho^f J_k(t_n)(\mathbf{v}_k^f(t_n) - \mathbf{v}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f.
$$

With these issues explained, we can proceed and define the semi-discrete forms. Since in this example we do not have the correspondence between the formulations given by the space-time and traditional time-stepping, we present already integrated forms over a time subinterval I_n

$$
\begin{aligned}
B_k^f(\vec{\mathbf{U}}_k)(\mathbf{\Phi}_k^f)\Big|_{I_n} :=& \left(\rho^f J_k(t_n)(\mathbf{v}_k^f(t_n) - \mathbf{v}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f \\
&- \frac{1}{2}\left(\rho^f J_k(t_n)\mathbf{F}_k^{-1}(t_n)\nabla\mathbf{v}_k^f(t_n)(\mathbf{u}_k^f(t_n) - \mathbf{u}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f \\
&- \frac{1}{2}\left(\rho^f J_k(t_{n-1})\mathbf{F}_k^{-1}(t_{n-1})\nabla\mathbf{v}_k^f(t_{n-1})(\mathbf{u}_k^f(t_n) - \mathbf{u}_k^f(t_{n-1})), \boldsymbol{\varphi}_k^f(t_n)\right)_f \\
&+ \left(\frac{1}{2} + k_n\right)k_n a_{CN}^f\left(\vec{\mathbf{U}}_k^f(t_n), \mathbf{I}_k^f\vec{\mathbf{U}}_k^s(t_n)\right)^T\left(\mathbf{\Phi}_k^f(t_n)\right) \\
&+ \left(\frac{1}{2} - k_n\right)k_n a_{CN}^f\left(\vec{\mathbf{U}}_k^f(t_{n-1}), \mathbf{I}_k^f\vec{\mathbf{U}}_k^s(t_{n-1})\right)^T\left(\mathbf{\Phi}_k^f(t_n)\right) \\
&+ k_n a_{IE}^f\left(\vec{\mathbf{U}}_k^f(t_n), \mathbf{I}_k^f\vec{\mathbf{U}}_k^s(t_n)\right)^T\left(\mathbf{\Phi}_k^f(t_n)\right),
\end{aligned}
$$

$$
\begin{aligned}
B_k^s(\vec{\mathbf{U}}_k)(\mathbf{\Phi}_k^s)\Big|_{I_n} :=& \left(\rho^s(\mathbf{v}_k^s(t_n) - \mathbf{v}_k^s(t_{n-1})), \boldsymbol{\varphi}_k^s(t_n)\right)_s + \left(\mathbf{u}_k^s(t_n) - \mathbf{u}_k^s(t_{n-1}), \boldsymbol{\psi}_k^s(t_n)\right)_s \\
&+ \left(\frac{1}{2} + k_n\right)k_n a_{CN}^s\left(\mathbf{I}_k^s\vec{\mathbf{U}}_k^f(t_n), \vec{\mathbf{U}}_k^s(t_n)\right)^T\left(\mathbf{\Phi}_k^s(t_n)\right) \\
&+ \left(\frac{1}{2} - k_n\right)k_n a_{CN}^s\left(\mathbf{I}_k^s\vec{\mathbf{U}}_k^f(t_{n-1}), \vec{\mathbf{U}}_k^s(t_{n-1})\right)^T\left(\mathbf{\Phi}_k^s(t_n)\right) \\
&+ k_n a_{IE}^s\left(\mathbf{I}_k^s\vec{\mathbf{U}}_k^f(t_n), \vec{\mathbf{U}}_k^s(t_n)\right)^T\left(\mathbf{\Phi}_k^s(t_n)\right).
\end{aligned}
$$

We used the operator $\mathbf{I}^k$ to transfer the interface values between the two domains. To make it more readable, this formulation assumes that there is no micro time-stepping in I_n. Otherwise, one would need to replace I_n by either $I_{n,m}^f$ or $I_{n,m}^s$ and use the corresponding time discretization points

2.1.4 Thermoelasticity

This problem too will be discretized using the Crank-Nicolson method. That results in a slight inconsistency between the time-stepping scheme and the discretization following from the space-time approach. We take

$$
X_k^s := \left(X_k^{s,1}\right)^{2d}, \quad X_k^T := X_k^{T,1}, \quad X_k := X_k^s \times X_k^T
$$

as trial spaces as well as

$$
Y_k^s := \left(Y_k^{s,0}\right)^{2d}, \quad Y_k^T := Y_k^{T,0}, \quad Y_k := Y_k^s \times Y_k^T
$$

as the test spaces. Below we present the discretization given by the Crank-Nicolson scheme. For simplicity, we omit indices corresponding to the micro time-steps

$$
\begin{aligned}
B_k^s(\vec{\mathbf{U}}, T_k)(\mathbf{\Phi})\Big|_{I_n} &:= \Big(\rho(\mathbf{v}_k(t_n) - \mathbf{v}_k(t_{n-1})), \boldsymbol{\varphi}_k(t_n)\Big) + \Big(\mathbf{u}_k(t_n) - \mathbf{u}_k(t_{n-1}), \boldsymbol{\psi}_k(t_n)\Big) \\
&\quad + \frac{1}{2} k_n a^s \Big(\vec{\mathbf{U}}_k(t_n), I_k^s T_k(t_n)\Big)^T \Big(\mathbf{\Phi}_k(t_n)\Big) \\
&\quad + \frac{1}{2} k_n a^s \Big(\vec{\mathbf{U}}_k(t_{n-1}), I_k^s T_k(t_{n-1})\Big)^T \Big(\mathbf{\Phi}_k(t_n)\Big), \\
B_k^T(\vec{\mathbf{U}}_k, T_k)(\Lambda)\Big|_{I_n} &:= \Big(\rho C(T_k(t_n) - T_k(t_{n-1})), \Lambda(t_n)\Big) \\
&\quad + \frac{1}{2} k_n a^T \Big(\mathbf{I}_k^T \vec{\mathbf{U}}_k(t_n), T_k(t_n)\Big)^T \Big(\Lambda_k(t_n)\Big) \\
&\quad + \frac{1}{2} k_n a^T \Big(\mathbf{I}_k^T \vec{\mathbf{U}}_k(t_{n-1}), T_k(t_{n-1})\Big)^T \Big(\Lambda_k(t_n)\Big).
\end{aligned}
$$

In this formulation, the superscripts switch from the interface operators to the ones corresponding to the thermoelasticity problem.

2.2 Space Discretization

The discretization in space is performed using classical approaches. For every problem, we use triangular elements continuous in space. However, in the case of interface coupled problems, we can not guarantee the continuity across the interface neither of the trial nor the test space because of our partitioned approach. Further, handling the transfer of interface values between different meshes does require additional attention. It will be explored in the next section.

To discretize the problems, we introduce regular triangulations $\mathcal{T}_h^f$ and $\mathcal{T}_h^s$. We assume that they match across the interface Γ. K^f denotes an element of the mesh $\mathcal{T}_h^f$ and K^s an element in $\mathcal{T}_h^s$. Their sizes are denoted by h_K^f and h_K^s, respectively. Further,

$$
h^f := \max_{K^f \in \mathcal{T}_h^f} h_K^f, \quad h^s := \max_{K^s \in \mathcal{T}_h^s} h_K^s, \quad h := \max\left\{h^f, h^s\right\}.
$$

As the function spaces, for the interface coupled problems, we take space of continuous polynomials of order r

$$
\begin{aligned}
X_{k,h}^{f,1}(r) &= \left\{\boldsymbol{\varphi} \in C(X_k^{f,1}) \,\Big|\, \varphi|_{K^f} \in \mathcal{P}_r(\Omega^f) \text{ for all } K^f \in \mathcal{T}_h^f\right\}, \\
X_{k,h}^{s,1}(r) &= \left\{\boldsymbol{\varphi} \in C(X_k^{s,1}) \,\Big|\, \varphi|_{K^s} \in \mathcal{P}_r(\Omega^s) \text{ for all } K^s \in \mathcal{T}_h^s\right\}, \\
Y_{k,h}^{f,0}(r) &= \left\{\boldsymbol{\varphi} \in C(Y_k^{f,0}) \,\Big|\, \varphi|_{K^f} \in \mathcal{P}_r(\Omega^f) \text{ for all } K^f \in \mathcal{T}_h^f\right\}, \\
Y_{k,h}^{s,0}(r) &= \left\{\boldsymbol{\varphi} \in C(Y_k^{s,0}) \,\Big|\, \varphi|_{K^s} \in \mathcal{P}_r(\Omega^s) \text{ for all } K^s \in \mathcal{T}_h^s\right\}.
\end{aligned}
$$

For our volume coupled problem, we will replace the spaces corresponding to the fluid problem by

$$X_{k,h}^{T,1}(r) = \left\{ \varphi \in C(X_k^{T,1}) \ \middle| \ \varphi|_{K^s} \in \mathcal{P}_r(\Omega^s) \text{ for all } K^s \in \mathcal{T}_h^s \right\},$$

$$Y_{k,h}^{T,0}(r) = \left\{ \varphi \in C(Y_k^{T,0}) \ \middle| \ \varphi|_{K^s} \in \mathcal{P}_r(\Omega^s) \text{ for all } K^s \in \mathcal{T}_h^s \right\}.$$

Given these function spaces, one can define the fully discrete problem either as

Problem 2.3. *Find* $\vec{U}_{k,h} \in X_{k,h} := X_{k,h}^f \times X_{k,h}^s$ *such that*

$$B_{k,h}^f(\vec{U}_{k,h})(\mathbf{\Phi}_{k,h}^f) = F_k^f(\mathbf{\Phi}_{k,h}^f), \qquad \text{for all } \mathbf{\Phi}_{k,h}^f \in Y_{k,h}^f,$$

$$B_{k,h}^s(\vec{U}_{k,h})(\mathbf{\Phi}_{k,h}^s) = F_k^s(\mathbf{\Phi}_{k,h}^s), \qquad \text{for all } \mathbf{\Phi}_{k,h}^s \in Y_{k,h}^s$$

or, in the case of thermoelasticity problem, as

Problem 2.4. *Find* $(\vec{U}_{k,h}, T_{k,h}) \in X_{k,h} := X_{k,h}^f \times X_{k,h}^s$ *such that*

$$B_k^s(\vec{U}_{k,h}, T_{k,h})(\mathbf{\Phi}_{k,h}) = F_k^s(\mathbf{\Phi}_{k,h}), \qquad \text{for all } \mathbf{\Phi}_{k,h} \in Y_{k,h}^s,$$

$$B_k^T(\vec{U}_{k,h}, T_{k,h})(\Lambda_{k,h}) = 0, \qquad \text{for all } \Lambda_{k,h} \in Y_{k,h}^T.$$

As we can see, only the forms defining the interface coupled problems change. This is due to the inclusion of the fully discrete Nitsche terms on the interface.

2.2.1   Transfer of Interface Values

Transferring the coupling conditions across the interface poses a technical difficulty worth discussing. Particularly challenging is shifting normal derivatives between the meshes. Indeed, computing a normal derivative of a function on a single triangular element depends on all the nodes and not only the ones touching the interface. We propose two possible approaches to solve this problem:

1. **First transfer then assemble** - assuming that the fluid and solid meshes are symmetric with respect to the interface, we first reassign nodal values from one mesh to the other, then assemble them on the new mesh. The process is illustrated in Figure 2.4. This approach was used for the model coupling heat and wave equation.

2. **First assemble then transfer** - we first assemble the system and then transfer the nodal values between the meshes. The advantage of this approach is that once the system is assembled, the important information is only stored in the nodes directly on the interface. Therefore, we can relax the requirements on the fluid and solid meshes. Here, it is enough to assume that the interface nodes on both meshes coincide. We used this approach for the FSI problem.
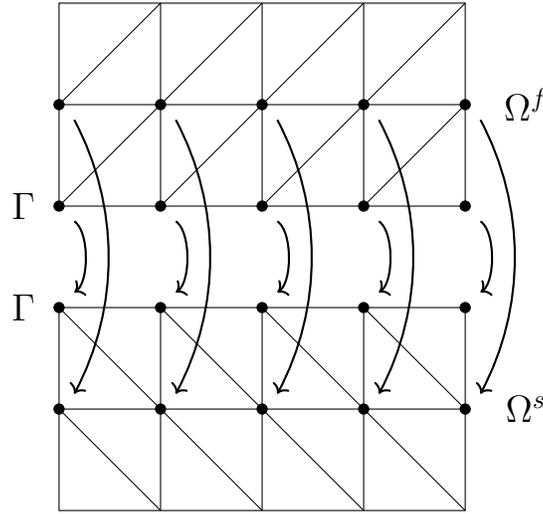
Figure 2.4: In this figure we show how nodal values are mirrored across the interface from the fluid to the solid domain. Triangular linear elements are considered.

We can clearly see that because of its flexibility, the second approach is superior. However, from the implementation standpoint, it can be quite troublesome. In the first approach, the function can be treated as a global entity and as such can be mirrored with respect to a known axis of symmetry. It is not necessary to perform independent transformations on a node level. The second way is inherently local, where the transformations have to be performed node by node. Therefore, we need to know the exact correspondence between the coordinates of a node and its position in the assembled system. It is further complicated for mixed multidimensional finite element spaces and meshes which are not regular. Therefore, this approach requires quite deep knowledge of the finite element software.

2.2.2 Heat and Wave Equation

In this example, as the space mesh, we take a regular triangular mesh. To obtain symmetry across the interface, the cells in the fluid domain have a right diagonal, and the cells in the solid domain have a left diagonal. For an illustration, we again refer to Figure 2.4. As the function spaces, we take elements that are continuous and piecewise linear in space. Therefore, we choose

$$X_{k,h}^f = \left(X_{k,h}^{f,1}(1)\right)^2, \quad X_{k,h}^s = \left(X_{k,h}^{s,1}(1)\right)^2, \quad Y_{k,h}^f = \left(Y_{k,h}^{f,0}(1)\right)^2, \quad Y_{k,h}^s = \left(Y_{k,h}^{s,0}(1)\right)^2.$$
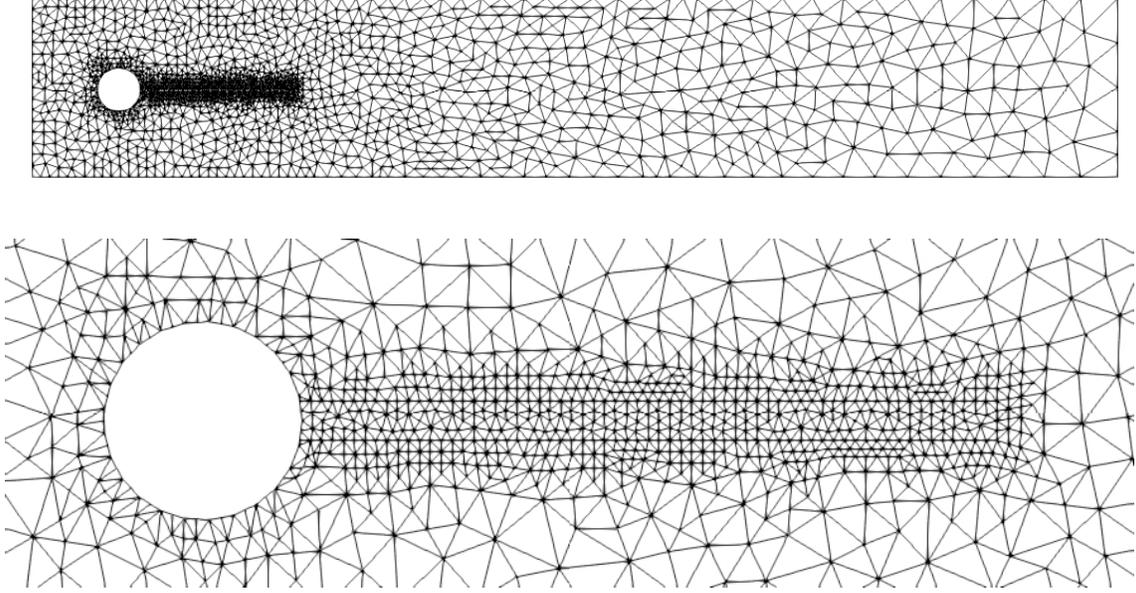
Figure 2.5: In this figure, we present the space mesh of the FSI model problem. In the top figure, the whole mesh is shown, in the bottom, we zoom on the cylinder and attached elastic structure. The solid mesh as well as the fluid cells close to the obstacle are highly refined. One can also see some refinement around the cylinder. In the direction of the outflow, the cell size continuously increases.

We change the forms a^f and a^s in (1.17) to include the fully discrete version of the Nitsche coupling

$$
\begin{aligned}
a_h^f(\vec{\mathbf{U}})(\boldsymbol{\Phi}^f) := & \left(\nu\nabla v^f, \nabla\varphi^f\right)_f + \left(\boldsymbol{\beta}\cdot\nabla v^f, \varphi^f\right)_f + \left(\nabla u^f, \nabla\psi^f\right)_f - \left\langle\nu\partial_{\vec{\mathbf{n}}_f} v^f, \varphi^f\right\rangle_\Gamma \\
& - \left\langle\partial_{\vec{\mathbf{n}}_f} u^f, \psi^f\right\rangle_\Gamma + \frac{\gamma\nu}{h}\left\langle v^f - v^s, \varphi^f\right\rangle_\Gamma + \frac{\gamma}{h}\left\langle u^f - u^s, \psi^f\right\rangle_\Gamma, \\
a_h^s(\vec{\mathbf{U}})(\boldsymbol{\Phi}^s) := & \left(\lambda\nabla u^s, \nabla\varphi^s\right)_s + \left(\delta\nabla v^s, \nabla\varphi^s\right)_s - \left(v^s, \psi^s\right)_s + \left\langle\nu\partial_{\vec{\mathbf{n}}_f} v^f, \varphi^s\right\rangle_\Gamma \\
& - \left\langle\delta\partial_{\vec{\mathbf{n}}_s} v^s, \varphi^s\right\rangle_\Gamma.
\end{aligned}
$$

2.2.3 FSI Problem

Since this domain is more complicated, the domains are discretized using unstructured meshes. The fluid mesh is finely refined around the elastic structure. To better approximate the curvature of the cylinder, the elements around it are also refined. To save some computational effort, we consider bigger cells close to the outflow. the nodes of both the fluid and the solid mesh coincide on the interface. The mesh is shown in Figure 2.5.

As the function spaces in the fluid domain, we take the classical inf-sup stable Tylor-Hood elements, where the space for velocity is given by continuous piecewise quadratic functions and for the pressure, we take piecewise linear functions. Similarly, the velocity in the elastic structure, as well as the displacement in both of the domains, are approximated using the piecewise quadratic functions

$$X_{k,h}^f = \left(X_{k,h}^{f,1}(2)\right)^{2d} \times Y_{k,h}^{f,0}(1), \quad X_{k,h}^s = \left(X_{k,h}^{s,1}(2)\right)^{2d},$$
$$Y_{k,h}^f = \left(Y_{k,h}^{f,0}(2)\right)^{2d} \times Y_{k,h}^{f,0}(1), \quad Y_{k,h}^s = \left(Y_{k,h}^{s,0}(2)\right)^{2d}.$$

After including the Nitsche terms, the discrete version of the forms given by (1.19) reads as

$$a_h^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := \left(J\boldsymbol{\sigma}^f\mathbf{F}^{-T}, \nabla\boldsymbol{\varphi}^f\right)_f + \left(\rho^f J\mathbf{F}^{-1}\mathbf{v}^f \cdot \nabla\mathbf{v}^f, \boldsymbol{\varphi}^f\right)_f + \left(J\mathrm{tr}(\mathbf{F}^{-1}\nabla\mathbf{v}^f), \omega^f\right)_f$$
$$+ \left(S\nabla\mathbf{u}^f, \nabla\boldsymbol{\psi}^f\right)_f - \left\langle J\boldsymbol{\sigma}^f\mathbf{F}^{-T} \cdot \vec{\mathbf{n}}_f, \boldsymbol{\varphi}^f\right\rangle_\Gamma - \left\langle S\partial_{\vec{\mathbf{n}}_f}\mathbf{u}^f, \boldsymbol{\psi}^f\right\rangle_\Gamma$$
$$+ \frac{\gamma\rho^f\nu}{h}\left\langle\mathbf{v}^f - \mathbf{v}^s, \boldsymbol{\varphi}^f\right\rangle_\Gamma + \frac{\gamma}{h}\left\langle\mathbf{u}^f - \mathbf{u}^s, \boldsymbol{\psi}^f\right\rangle_\Gamma,$$
$$a_h^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := \left(\mathbf{F}\boldsymbol{\sigma}^s, \nabla\boldsymbol{\varphi}^s\right)_s - \left(\mathbf{v}^s, \boldsymbol{\psi}^s\right)_s + \left\langle J\boldsymbol{\sigma}^f, \mathbf{F}^{-T} \cdot \vec{\mathbf{n}}_f, \boldsymbol{\varphi}^s\right\rangle_\Gamma.$$

For the time-stepping, these forms are then split analogously as in (2.9).

2.2.4 Thermoelasticity

Since this is an example of a volume coupled problem where we do not need to tackle the issue of transferring the interface values between the two domains, its space discretization is the most straightforward. Also here, we use a collection of triangular linear function spaces

$$X_{k,h}^s = \left(X_{k,h}^{s,1}(1)\right)^{2d}, \quad X_{k,h}^T = X_{k,h}^{T,1}(1), \quad Y_{k,h}^s = \left(Y_{k,h}^{s,0}(1)\right)^{2d}, \quad Y_{k,h}^T = Y_{k,h}^{T,0}(1).$$

Chapter 3

Decoupling Algorithms

As it was already stated in the introduction, there exist two major approaches designed to solve algebraic systems arising from the discretization of coupled problems - a monolithic and a partitioned approach. In the former, one incorporates all subproblems into one larger system. In the latter, each system is solved separately. In a monolithic approach, through a proper choice of test and trial spaces, the coupling conditions are incorporated within a variational formulation. That leads to stable numerical schemes at a cost of lack of flexibility. Thus, our focus on temporal multiscale problems forces us to consider partitioned approaches. This, on the other hand, requires the introduction of decoupling algorithms to properly enforce coupling conditions which is the main topic of this chapter.

We will start with explaining the outline of weakly coupled schemes and justify why they are insufficient for our applications. Then we will proceed to strongly coupled schemes. First, we will present the general idea, then propose two different approaches, namely a relaxation and a shooting method. We will finish by evaluation of their performance based on our model problems. Their limitations will be discussed in the case of strongly coupled FSI problems due to the added mass effect.

3.1 Types of Coupled Schemes

In this section, we will present two types of coupled schemes, namely weakly and strongly coupled ones. We will shortly discuss their characteristics and list their advantages and drawbacks.

3.1.1 Weakly Coupled Schemes

Possibly the most straightforward way to numerically solve coupled systems is by using *weakly coupled schemes*. These are semi-explicit schemes that do not require any iterative procedure. To obtain a solution at a new time-step, in the coupling, one uses the solution from the previous time-step. That means, given the solution $\vec{\mathbf{U}}(t_n)$, one performs the following

steps to obtain $\vec{\mathbf{U}}(t_{n+1})$:

1. Using the solution $\vec{\mathbf{U}}^s(t_n)$, solve the fluid problem and obtain $\vec{\mathbf{U}}^f(t_{n+1})$.

2. With the help of $\vec{\mathbf{U}}^f(t_{n+1})$, solve the solid problem to get $\vec{\mathbf{U}}^s(t_{n+1})$.

In the case of interface coupled problems, this procedure can be understood as using solutions of the other subproblem to set boundary conditions on the interface to fully define and then solve a given subproblem. This algorithm is further conceptualized by

$$\vec{\mathbf{U}}^s(t_n) \xrightarrow[\text{problem}]{\text{fluid}} \vec{\mathbf{U}}^f(t_{n+1}) \xrightarrow[\text{problem}]{\text{solid}} \vec{\mathbf{U}}^s(t_{n+1}).$$

In both of these steps, if necessary, micro time-stepping is performed. We stress that for every time-step these steps are performed only once without any subcycling. To avoid unnecessary indices, here and throughout this chapter, we will omit the index k since we only consider semi-discrete solutions anyway.

This procedure might seem attractive given its low computational costs and simplicity. Indeed, an example of a successful application in the context of aeroelasticity is given in [35]. Unfortunately, such a method is only suitable for loosely coupled problems, otherwise, it can suffer from stability issues. A rather pessimistic analysis is performed in [1] and [2], where authors consider systems that exhibit added mass instabilities.

3.1.2 Strongly Coupled Schemes

The weakly coupled scheme described above does not guarantee that coupling conditions are satisfied exactly. To overcome that, a frequently used solution is to iteratively solve both of the subproblems until the coupling conditions are met. Such schemes hold the name of *strongly coupled schemes.* To accelerate and stabilize this procedure, after each iteration an additional step is performed by applying a *decoupling function.* This term will be further explored in the next sections.

From this description emerges the following scheme - until convergence, given the solution $\vec{\mathbf{U}}_i(t_{n+1})$, the i-th iteration of a decoupling method results from the following steps:

1. Using $\vec{\mathbf{U}}_i^s(t_{n+1})$, solve the fluid problem and obtain $\vec{\mathbf{U}}_{i+1}^f(t_{n+1})$.

2. From $\vec{\mathbf{U}}_{i+1}^f(t_{n+1})$ by solving the solid problem acquire the intermediary solution $\widetilde{\vec{\mathbf{U}}}_{i+1}^s(t_{n+1})$.

3. Apply a decoupling function to $\widetilde{\vec{\mathbf{U}}}_{i+1}^s(t_{n+1})$ and get $\vec{\mathbf{U}}_{i+1}^s(t_{n+1})$.

A concise version of this algorithm is written below

$$\vec{\mathbf{U}}_i^s(t_{n+1}) \xrightarrow[\text{problem}]{\text{fluid}} \vec{\mathbf{U}}_{i+1}^f(t_{n+1}) \xrightarrow[\text{problem}]{\text{solid}} \widetilde{\vec{\mathbf{U}}}_{i+1}^s(t_{n+1}) \xrightarrow[\text{function}]{\text{decoupling}} \vec{\mathbf{U}}_{i+1}^s(t_{n+1}).$$

Although this procedure might be costly, its stability properties are far superior compared to its weakly coupled counterpart. Below we formalize the first two steps of the decoupling procedure

Problem 3.1. *For a given $\vec{U}_i^s \in X_k^s$, find $\vec{U}_{i+1}^f \in X_k^f$ and $\widetilde{\vec{U}}_{i+1}^s \in X_k^s$ such that:*

$$
B_{n+1}^f \left(\begin{array}{c} \vec{U}_{i+1}^f \\ \vec{U}_i^s \end{array} \right) (\mathbf{\Phi}^f) = F_{n+1}^f(\mathbf{\Phi}^f)
$$

$$
B_{n+1}^s \left(\begin{array}{c} \vec{U}_{i+1}^f \\ \widetilde{\vec{U}}_{i+1}^s \end{array} \right) (\mathbf{\Phi}^s) = F_{n+1}^s(\mathbf{\Phi}^s)
$$

for all $\mathbf{\Phi}^f \in Y_k^f$ and $\mathbf{\Phi}^s \in Y_k^s$. By B_n^f and B_n^s we denote restrictions of forms B^f and B^s to I_n. Forms F_n^f and F_n^s are defined accordingly.

In our volume coupled model problem, where we couple an elastic structure and temperature evolution, in each iteration we will start with solving the solid problem and then proceed to the temperature one. Hence, the volume coupled version of this problem would be given by

Problem 3.2. *For a given $T_i \in X_k^T$, find $\vec{U}_{i+1} \in X_k^s$ and $\widetilde{T}_{i+1} \in X_k^T$ such that :*

$$
B_{n+1}^s \left(\begin{array}{c} \vec{U}_{i+1} \\ T_i \end{array} \right) (\mathbf{\Phi}) = F_{n+1}^s(\mathbf{\Phi})
$$

$$
B_{n+1}^T \left(\begin{array}{c} \vec{U}_{i+1} \\ \widetilde{T}_{i+1} \end{array} \right) (\Lambda) = F_{n+1}^T(\Lambda)
$$

for all $\mathbf{\Phi} \in Y_k^s$ and $\Lambda \in Y_k^T$.

3.2 Coupled Schemes for FSI Problems

Numerical simulation of FSI problems in a regime that is not fully monolithic is particularly challenging. That is especially true for problems with a high ratio between the fluid and solid densities $\frac{\rho^f}{\rho^s}$ where the fluid is incompressible. Heuristically, the movement of the elastic structure acts on the fluid close to the interface. Since the fluid is not compressible, it has to move along with the solid. Therefore, it acts as an additional artificial mass on the structure at the interface. For a given density of the structure, this interaction is stronger, the higher the density of the fluid is. This phenomenon does not happen if the problem is solved fully within the monolithic regime.

The description above of this effect is in line with the literature. In [1] the authors explored the correlation between this instability and time-stepping schemes. They deduced that this problem occurs independently of the choice of a time-stepping scheme. Moreover, the onset happened earlier when a smaller time-step size was used. They concluded that the instability is

inherent in the scheme itself and is caused by too large eigenvalues of the added mass operator on the interface. In [2] the influence of geometry was further explored. Numerical tests lead to an observation that lengthening of a domain worsens the instability. The authors also compared the performance of weakly and strongly coupled schemes for numerical problems where the added mass effect was present. They noted that weakly coupled schemes were not able to yield any results and the strongly coupled ones needed more iterations than usually. A different aspect of this issue was studied in [36] and [37]. There, the added mass effect was studied from the perspective of comparison between compressible and incompressible fluid models. The instability turned out to have a much stronger effect on incompressible fluids. In line with our heuristic explanation, due to the incompressibility condition, the displacement of the interface induces a global perturbation in the fluid. For compressible fluids, this perturbation is only local. It was further confirmed that decreasing time-step sizes worsens the effect.

3.3 Decoupling Algorithms

We will define in detail two decoupling algorithms that we will use for numerical simulations. Both of them are examples of strongly coupled schemes

3.3.1 Relaxation Method

The first method we would like present is a simple fixed-point method first presented in [38]. For a fixed damping parameter $\tau \in [0, 1]$ the new solution is defined by a convex combination of the old and the previously computed intermediate solution.

Definition 3.1 (Relaxation Function). *Let $\vec{U}_i^s \in X_k^s$ and $\widetilde{\vec{U}}_{i+1}^s \in X_k^s$ be the solid intermediate solution defined in Problem 3.1. Then for $\tau \in [0, 1]$ the relaxation function $R : X_k^s \to X_k^s$ is given by:*

$$R(\vec{U}_i^s) := \tau \widetilde{\vec{U}}_{i+1}^s + (1 - \tau) \vec{U}_i^s$$

Assuming that we already know the value $\vec{U}^s(t_n)$, we pose

$$\vec{U}_0^s(t_n + 1) := \vec{U}^s(t_n),$$
$$\vec{U}_{i+1}^s(t_{n+1}) := R(\vec{U}_i^s)(t_{n+1}) \text{ for } i = 1, 2, \dots.$$

Even though this method was introduced over 60 years ago, it is widely used up to this day. For example in [10] authors praised its convergence properties compared with low computational cost. The importance of the right choice of the damping parameter was underlined. In a more recent paper [11], the relaxation method was used to simulate a nonlinear flexible beam. Nevertheless, this method appears to be inferior to more elaborate quasi-Newton approaches which we will discuss in the next section. A comparison between the two was presented for example in [12]. Especially in the case of strong coupling between the subproblems the relaxation method might not be sufficient.

46

3.3.2   Shooting Method

Multirate time-stepping can be implemented in a more sophisticated way by using a shooting method. Then the resulting root-finding problem can be solved by the Newton method. This method has been popular and widely used in literature. To successfully employ this approach one has to make two crucial decisions. The first one is the definition of the root-finding problem itself. The second one is an approximation of a Jacobian needed in each Newton iteration. We will start with the former.

Definition 3.2 (Interface Shooting Function)**.** *Let* $\vec{U}_i^s \in X_k^s$ *and* $\widetilde{\vec{U}}_{i+1}^s \in X_k^s$ *be the solid solution of Problem 3.1. Then the shooting function* $S : X_k^s \to (L^2(\Gamma))^2$ *is defined as:*

$$S(\vec{U}_i^s) := \left(\vec{U}_i^s(t_n) - \widetilde{\vec{U}}_{i+1}^s(t_n) \right)\Bigg|_\Gamma \tag{3.1}$$

The function above is only appropriate for interface coupled problems. Below we present the volume coupled version of this definition.

Definition 3.3 (Volume Shooting Function)**.** *Let* $T_i \in X_k^T$ *and* $\widetilde{T}_{i+1} \in X_k^T$ *be the temperature solution of Problem 3.2. Then the shooting function* $S : X_k^T \to L^2(\Omega)$ *is defined as:*

$$S(T_i) := T_i(t_n) - \widetilde{T}_{i+1}(t_n) \tag{3.2}$$

Interestingly, none of these definitions uses the coupling conditions explicitly. Instead, they rely on reaching the point where the solution stabilizes. This way, the method becomes more flexible and can be used in a variety of situations. It is one of our new contributions to the application of this method.

Given these functions, in each iteration of the shooting procedure, we want to find a solution $\vec{U}_i^s$ to $S(\vec{U}_i^s) = 0$. By discretizing this equation using the Newton method, we obtain

$$S'(\vec{U}_i^s)\vec{d}_{i+1} = -S(\vec{U}_i^s),$$

where $\vec{d}_{i+1}$ is an unknown vector. This problem is either defined on the common interface in the case of interface coupled problems or the whole domain for volume coupled ones.

An appropriate approximation of the Jacobian S' is one of the most challenging aspects of the quasi-Newton approaches presented here. There exists a range of possible solutions to tackle this problem. An extended survey on Jacobian-free methods is presented in [13]. In [14] authors compute the Jacobians exactly using shape derivative calculus. Strongly coupled problems including an added mass effect are explored in [15], where the Jacobian is handled using model order reduction techniques. Another possible solution is presented in [16] where the approximation is done using the least square method. In [17] a matrix-free approach using a Jacobian of a simplified model is considered for applications coming from medical imaging. In [18] authors explore a variety of different solutions such as a direct solution with block-triangular approximate Jacobians as well as matrix-free approaches with
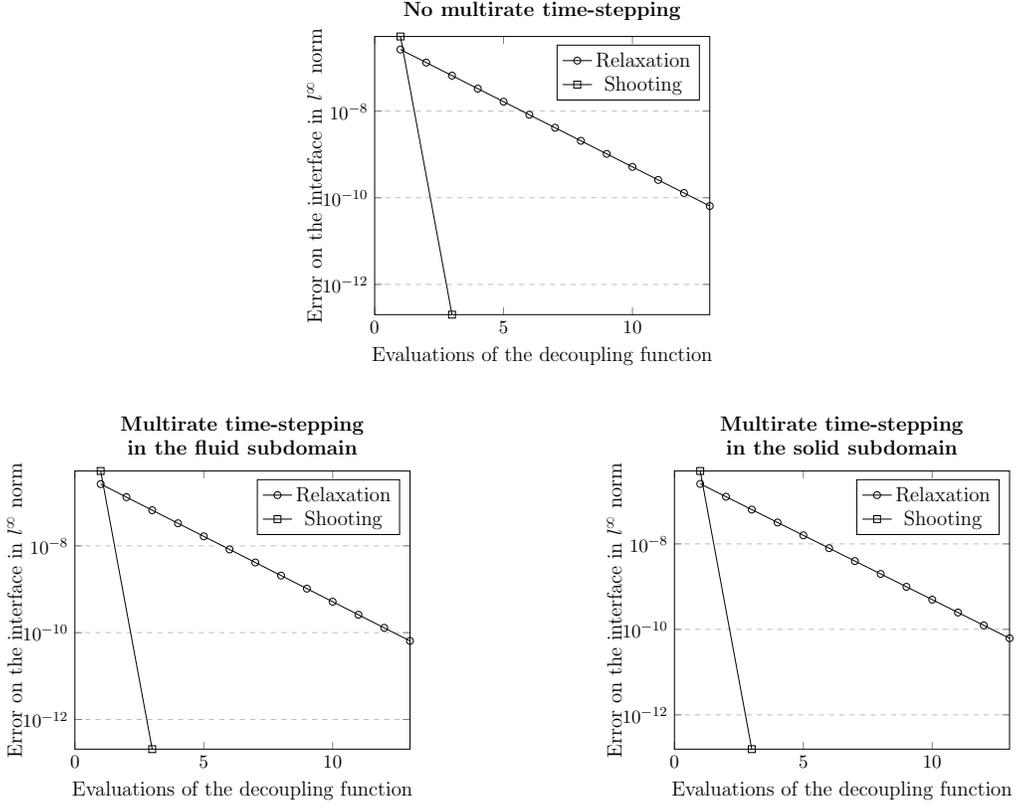
Figure 3.1: Performance of decoupling methods for Configuration 1 in one macro time-step in the case of $N_2^f = N_2^s = 1$ (top), $N_2^f = 10$ and $N_2^s = 1$ (left), $N_2^f = 1$ and $N_2^s = 10$ (right).

approximate Jacobians and a global pressure Schur complement.

Here as well we decided to use a matrix-free approach, where the Jacobian-vector product is approximated using a finite difference method. The first needed step is a choice of a suitable matrix-free Krylov subspace linear solver such as matrix-free GMRES. Thanks to that, we do not need all the entries of the Jacobian, only a suitable definition of its vector-matrix product. This quantity can be interpreted as a directional derivative and approximated using a finite difference quotient

$$S'(\vec{\mathbf{U}}_i^s)\vec{\mathbf{d}}_{i+1} \approx \frac{S(\vec{\mathbf{U}}_i^s + \varepsilon\vec{\mathbf{d}}_{i+1}) - S(\vec{\mathbf{U}}_i^s)}{\varepsilon} \tag{3.3}$$

with a suitable value of ε. This way, we achieve a nested iterative algorithm, where for each macro time-step the outer loop is defied by the Newton and the inner one by the GMRES method. Once the increment vector $\vec{\mathbf{d}}_{i+1}$ is computed, we can define the Newton iteration by
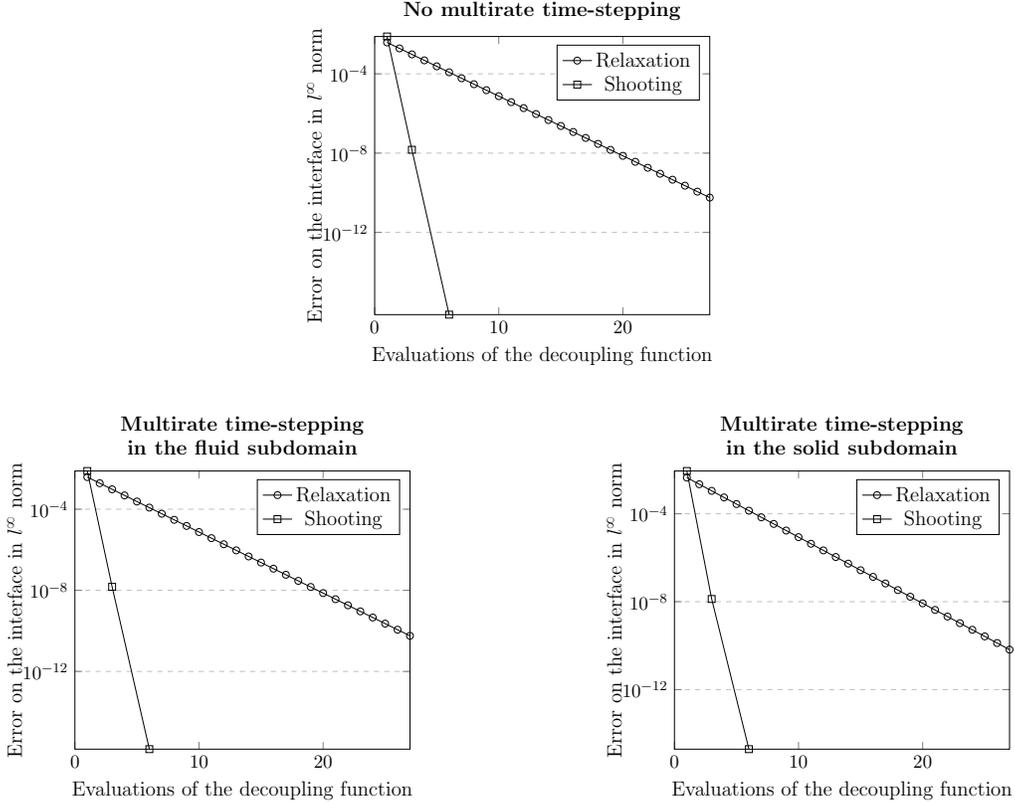
Figure 3.2: Performance of decoupling methods for Configuration 2 in one macro time-step in the case of $N_2^f = N_2^s = 1$ (top), $N_2^f = 10$ and $N_2^s = 1$ (left), $N_2^f = 1$ and $N_2^s = 10$ (right).

$$\vec{\mathbf{U}}_0^s(t_n)\big|_\Gamma := \vec{\mathbf{U}}^s(t_{n-1})\big|_\Gamma,$$
$$\vec{\mathbf{U}}_{i+1}^s(t_n)\big|_\Gamma := \vec{\mathbf{U}}_i^s(t_n)\big|_\Gamma + \vec{\mathbf{d}}_{i+1}. \tag{3.4}$$

The solution on the previously computed time-step $\vec{\mathbf{U}}^s(t_{n-1})$ serves here as an initial guess. It is well known that a good choice of an initial guess for the Newton method is especially important because of its limited convergence radius. Thus, we also considered an alternative initial guess additionally consisting of the solution $\vec{\mathbf{U}}^s(t_{n-2})$ given by

$$\vec{\mathbf{U}}_0^s(t_n)\big|_\Gamma := \left(\vec{\mathbf{U}}^s(t_{n-1}) + \frac{k_n}{k_{n-1}}(\vec{\mathbf{U}}^s(t_{n-1}) - \vec{\mathbf{U}}^s(t_{n-2}))\right)\Big|_\Gamma. \tag{3.5}$$

Although matrix-free quasi-Newton methods are not new, the usage of a directional derivative (3.3) to approximate the Jacobian is a novelty.

49
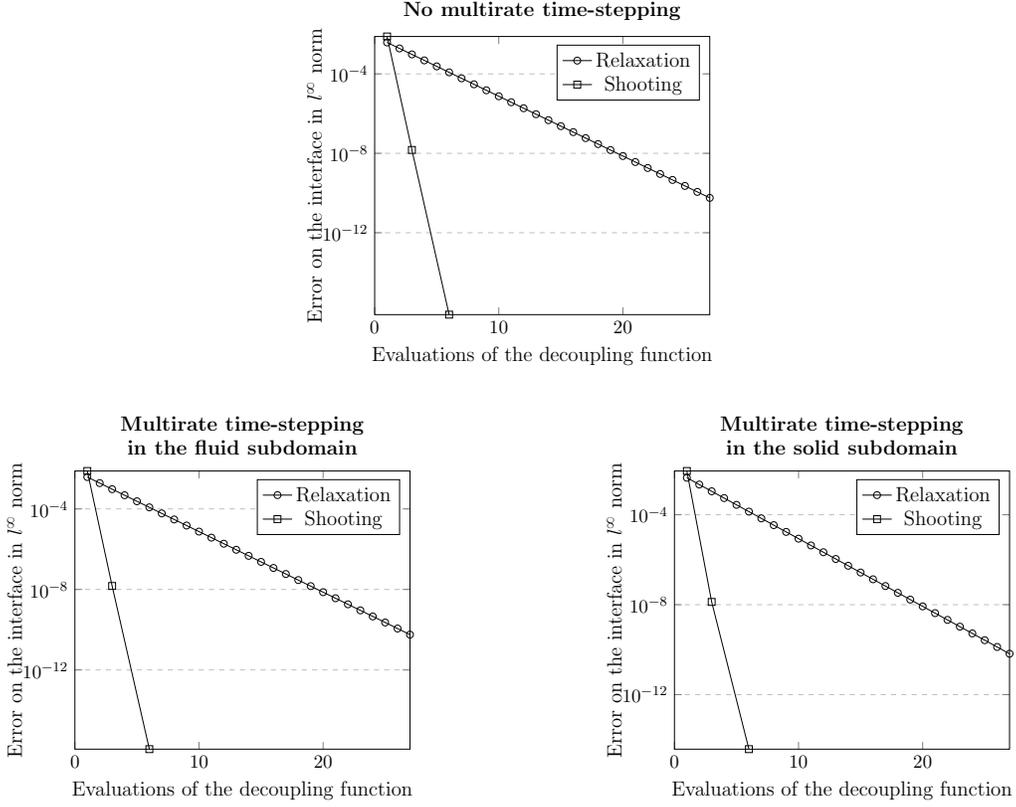
Figure 3.3: Performance of decoupling methods for Configuration 3 in one macro time-step in the case of $N_2^f = N_2^s = 1$ (top), $N_2^f = 10$ and $N_2^s = 1$ (left), $N_2^f = 1$ and $N_2^s = 10$ (right).

3.3.3 Stopping Criteria

As stopping criteria we use a mixture of a relative and an absolute tolerance. Regardless of the decoupling algorithm, we use the shooting function to establish stopping criteria. The absolute tolerance stopping criterion is given by

$$||S(\vec{\mathbf{U}}_i^t)||_\infty \leqslant tol$$

and the relative tolerance criterion by

$$\frac{||S(\vec{\mathbf{U}}_i^s)||_\infty}{||S(\vec{\mathbf{U}}_0^s)||_\infty} \leqslant tol.$$

$|| \cdot ||_\infty$ denotes l^∞ norm. In the case of the shooting method, we have to set the tolerance for both the outer Newton method as well as the inner GMRES method. We can save some

Figure 3.4: Number of evaluations of the decoupling functions for Configuration 1 needed for convergence on the time interval $I = [0, 1]$ for $N = 50$ in the case of $N^f = N^s = N$ (top), $N^f = 10N$ and $N^s = N$ (left), $N^f = N$ and $N^s = 10N$ (right).

computational resources by setting the GMRES tolerance slightly higher than the Newton method tolerance. However, by doing so we run the risk of not obtaining the accuracy required to reach the tolerance of the outer loop. Therefore, we will either use constant values of tolerances shown in Table 3.1 or take GMRES tolerance equal to

$$tol := \max\left(||S(\vec{\mathbf{U}}_i^t)||_\infty, 10^{-10}\right).$$

We would also like to go back to ε from the finite difference quotient (3.3). We have discovered that it is beneficial to set dependency between ε, the residual and the absolute tolerance of the Newton method. After this consideration, we arrived at the epsilon given by the same formula as the tolerance

$$\varepsilon := \max\left(||S(\vec{\mathbf{U}}_i^t)||_\infty, 10^{-10}\right).$$
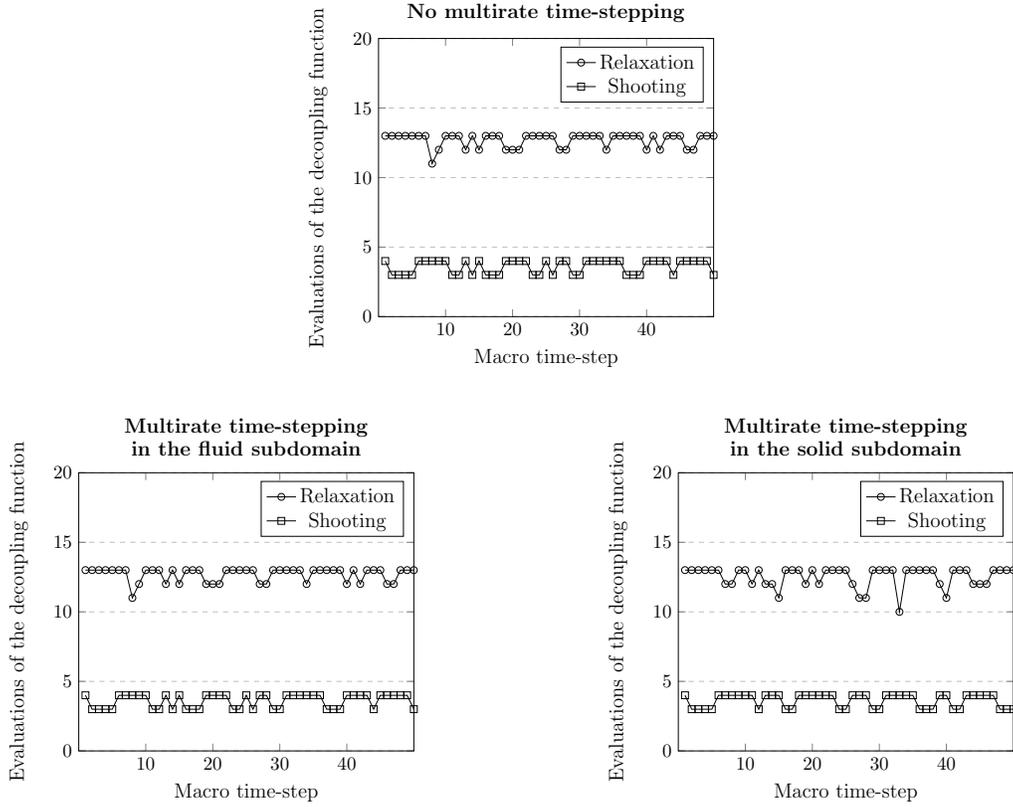
Figure 3.5: Number of evaluations of the decoupling functions for Configuration 2 needed for convergence on the time interval $I = [0, 1]$ for $N = 50$ in the case of $N^f = N^s = N$ (top), $N^f = 10N$ and $N^s = N$ (left), $N^f = N$ and $N^s = 10N$ (right).

Although, the tolerance for the GMRES method as well as ϵ are given by the same formula, the reasons for that are different. By setting the GMRES tolerance depending on the shooting function S, we can only save some computational resources. On the other hand, choosing ϵ this way positively influences the stability of the shooting method, especially in the case of the FSI test case.

3.4 Numerical Results

In this section, we are going to compare the performance of the presented methods on our model problems. In each case, we will look at the convergence rate of the residual of decoupling methods and the number of iterations needed to converge. Whenever possible, we will introduce micro time-stepping and analyze its influence on the performance.
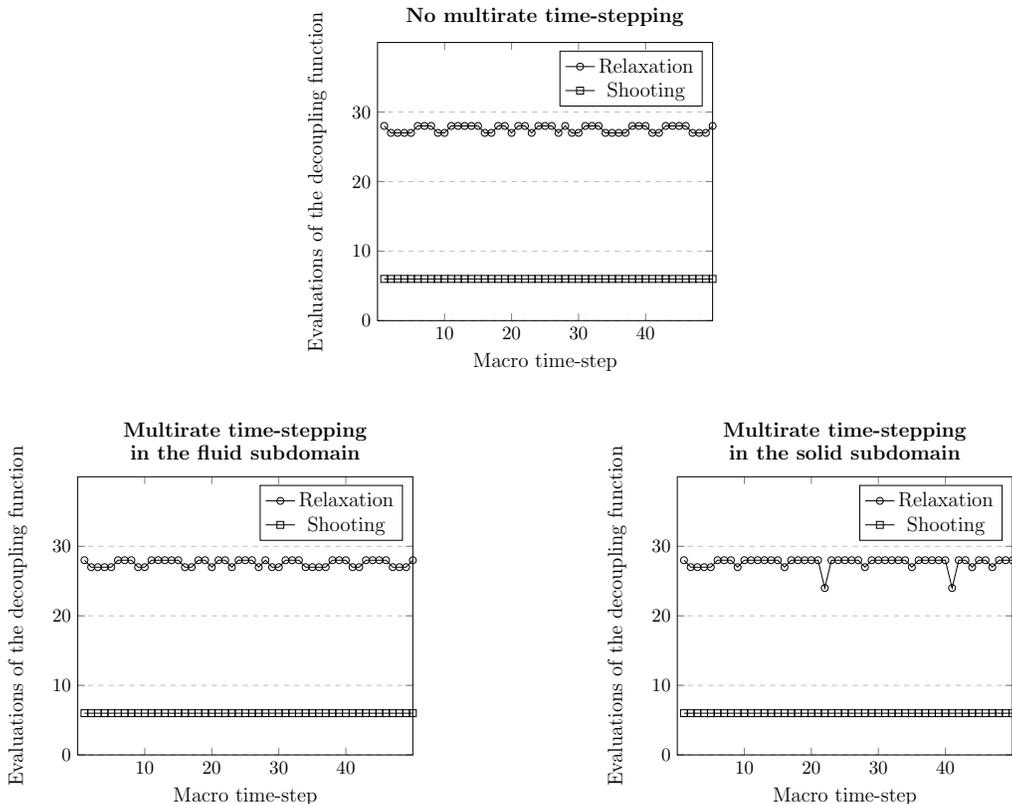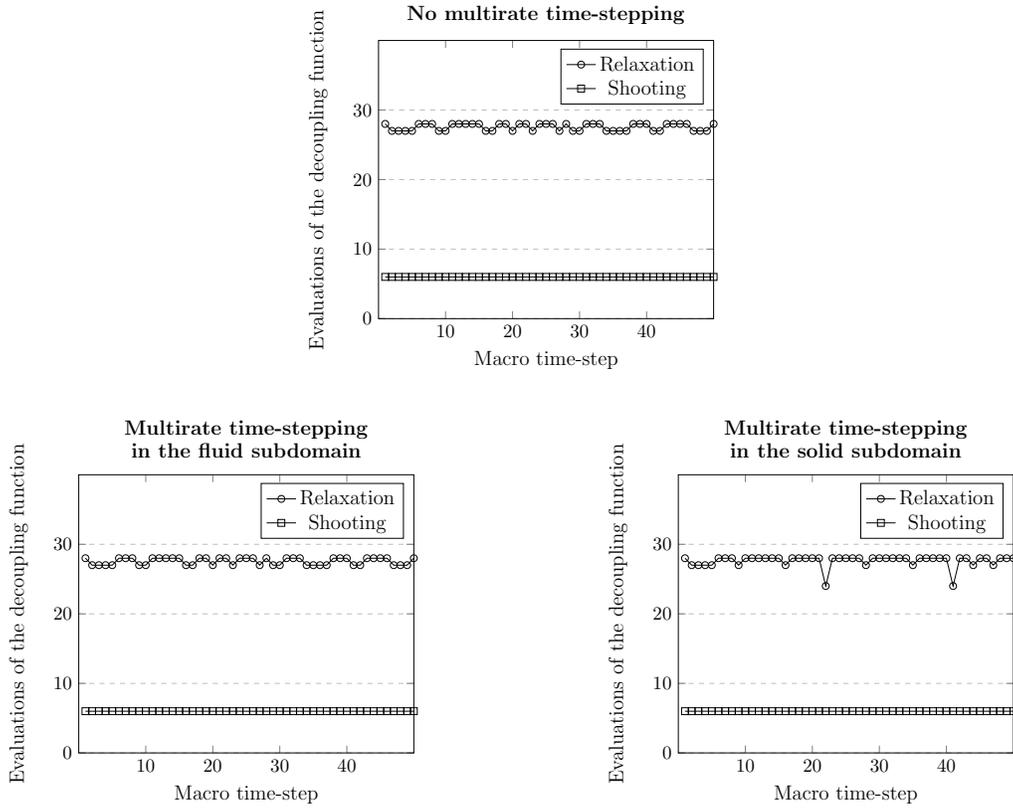
Figure 3.6: Number of evaluations of the decoupling functions for Configuration 3 needed for convergence on the time interval $I = [0,1]$ for $N = 50$ in the case of $N^f = N^s = N$ (top), $N^f = 10N$ and $N^s = N$ (left), $N^f = N$ and $N^s = 10N$ (right).

3.4.1 Heat and Wave Equation

For our simple case of coupling of the heat and wave equations, our approach works very well. The shooting method is clearly superior to the relaxation method. Further, introducing even many micro time-steps does not harm the performance.

In Figures 3.1, 3.2, and 3.3 we looked at the performance of the decoupling methods for every version of the source terms (Configuration 1, 2 and 3). In each case, we looked at the values of the residuals on the single macro time-step $I_2 = [0.02, 0.04]$. For each of the right hand sides, we performed simulations in the instance of no micro time-stepping ($N_2^f = N_2^s = 1$), 10 micro time-steps in the fluid domain ($N_2^f = 10N_2^s = 10$), and 10 micro time-steps in the solid domain ($10N_2^f = N_2^s = 10$). For all of these examples, we chose only uniform time-step sizes. No matter which source term we considered, the introduction
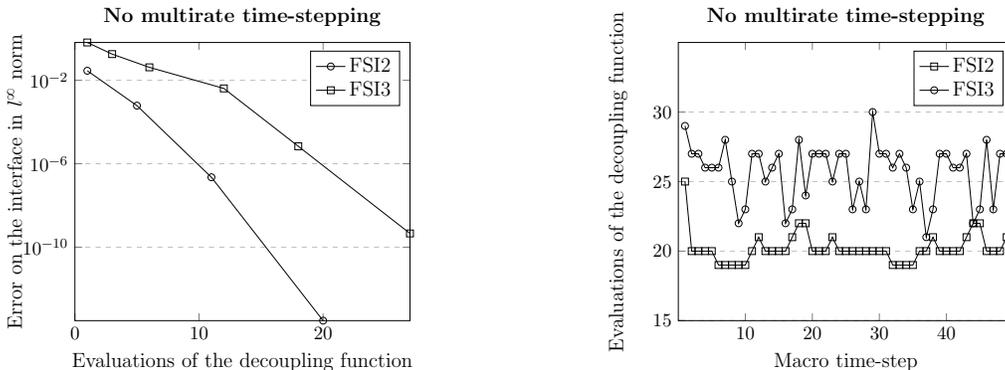
Figure 3.7: Performance of the shooting function one macro time-step (left) and number of evaluations needed for convergence on the time interval I (right).

tolerance	relaxation	Newton	GMRES
absolute tolerance	10^{-10}	10^{-10}	10^{-10}
relative tolerance	10^{-8}	10^{-8}	10^{-8}

Table 3.1: Here we present the tolerances. We decided to set the same tolerances for the Newton and GMRES methods.

of the micro time-steps had very little influence on the performance. Clearly, the shooting method outperforms the relaxation method. The relaxation method has a linear convergence rate. Because of the nested GMRES method as well very low number of iterations needed to converge, it is difficult to draw any assumptions on the convergence rate of the shooting method. In Figure 3.1, the relaxation method needed 13 iterations, whereas the shooting method required only 2 iterations of the outer Newton method and overall 3 linear systems. The results in Figure 3.2 and Figure 3.3 are very similar. The relaxation method required 27 iterations instead of 3 iterations and overall 6 linear systems were needed to reach the prescribed tolerance in the case of the shooting method. Based on these pieces of information, we can claim that adding the source term to the wave equation results in a more challenging system. That should not be surprising knowing its hyperbolic nature. This difference in difficulty is big enough that once we have the source term in the wave equation, adding a similar term to the heat equation does not have a visible impact on how challenging the resulting system becomes.

In Figures 3.4, 3.5, and 3.6 we show the results over the whole time interval $I = [0, 1]$ for each of the configurations of the right hand side. We looked at the uniform time-stepping with $k = 0.02$ and therefore $N = 50$. In every case, again, we considered an example without any
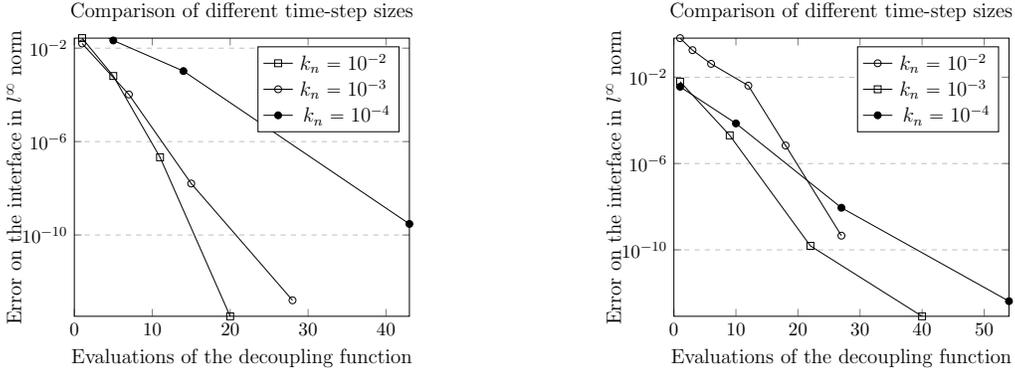
Figure 3.8: Number of evaluations of the decoupling functions needed for convergence depending on the size of time-steps for the FSI2 example (left) and FSI3 (right).

micro time-stepping ($N^f = N^s = N$), micro time-stepping in the fluid domain ($N^f = 10N$ and $N^s = N$), and micro time-stepping in the solid domain ($N^f = N$ and $N^s = 10N$). In all of these examples, the performance of both of the decoupling methods is stable over the whole time interval and does not depend on the number of micro time-steps. The number of iterations needed for convergence was the lowest in the examples, where the external force was applied to the heat equation only. There, the relaxation method required between 10 and 13 linear systems per time-step to reach the tolerance. The shooting method converged after solving only 3 or 4 systems. Similarly as in our analysis of a single time-step, here as well the simulations corresponding to Configuration 2 and 3 yield the same results. The relaxation method required there mostly 27 or 28 iterations with occasional exceptions where only 24 were sufficient. The performance of the shooting method was very stable, where universally 6 systems per time-step were enough to converge. All in all, the superiority of the shooting method extends to the whole interval.

3.4.2 FSI Problem

Unfortunately, the performance of the shooting method for the more complicated FSI problem is not fully satisfactory. Even though the method can converge consistently in the case of no micro time-stepping, it requires many iterations to achieve the prescribed tolerance. Moreover, it is only capable of dealing with micro time-stepping in the fluid domain. Introducing it in the solid domain leads to a volatile instability in pressure which after a few time-steps causes the method to crash. Since the instability occurs in the pressure, we would attribute it to the incompressibility condition, more precisely, the added mass effect. Although disappointing, this course of events should not be very surprising. By design, performing micro time-stepping inevitably introduces explicit elements into time discretization which in turn worsens stability. For this model problem, we do not present results for the

No multirate time-stepping

Multirate time-stepping
in the structure problem

Multirate time-stepping
in the temperature problem
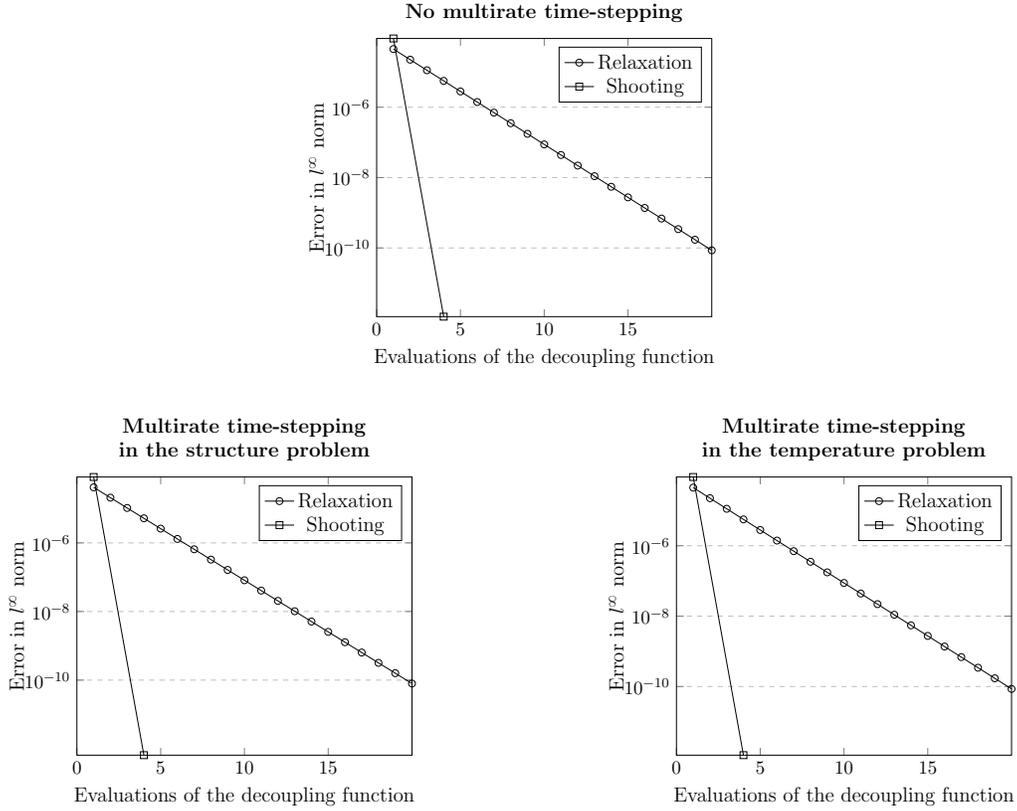
Figure 3.9: Performance of decoupling methods for the thermoelasticity model problem in one macro time-step in the case of $N_2^s = N_2^T = 1$ (top), $N_2^s = 10$ and $N_2^T = 1$ (left), $N_2^s = 1$ and $N_2^T = 10$ (right).

relaxation method because it was not able to converge at all.

The left picture in Figure 3.7 shows the results on a single macro time-step for both variants of the FSI problem. We are interested in time-steps where the oscillations are already fully developed. The development of these oscillations is slower for the FSI2 test problem (for the displacement profile, see Figure 1.5). Therefore, for each of the model problems we choose a different time-step. For the FSI2 problem, we take $I_2 = [12.01, 12.02]$, for the FSI3 we take $I_2 = [6.01, 6.02]$. Either way, the time-step size is equal to $k_n = 0.01$. We can clearly see that the method struggled more with the FSI3 variant of the model. One can notice that in the number of iterations of the Newton method, the number of GMRES iterations as well as the magnitude of the error itself. For the FSI2 test case, the shooting method needed 4 iterations of the Newton method and 20 GMRES iterations. For FSI3 these values are equal to 6 and 27, respectively. In both cases, we can observe the best error reduction in the final iteration
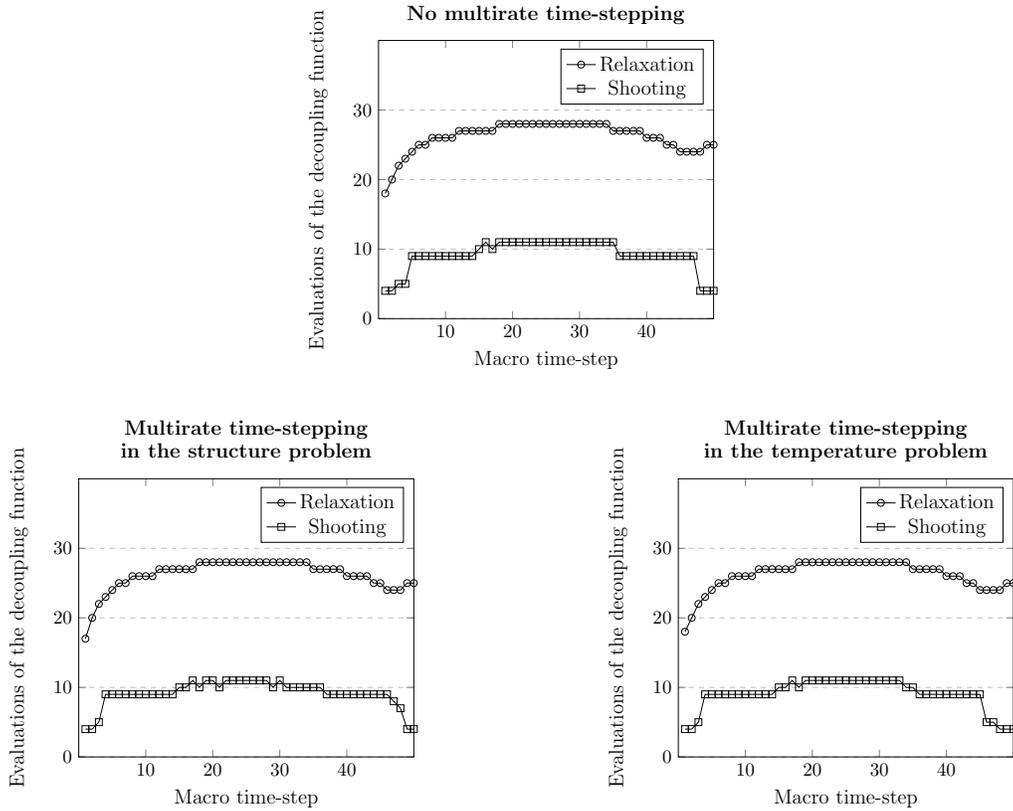
Figure 3.10: Number of evaluations of the decoupling functions for the thermoelasticity model problem needed for convergence on the time interval $I = [0.0, 0.1]$ for $N = 50$ in the case of $N^s = N^T = N$ (top), $N^s = 10N$ and $N^T = N$ (left), $N^s = N$ and $N^T = 10N$ (right).

of the Newton method. It is expected given the nature of the Newton method as well as the varying tolerances set for the GMRES method, which were discussed in Section 3.3.3. We would also like to point out that the comparison of simulations of the two model problems with the same time-step size might not be fair since the dynamics of the two problems are different. More precisely, the period of the oscillations for the FSI3 is much smaller than for the FSI2 model. Either way, given that we can not perform fully adaptive multirate time-stepping, the computational costs of the shooting method do not make it competitive with a more traditional fully monolithic approach.

The right picture in Figure 3.7 represents a a plot illustrating the performance of the shooting method during a longer simulation without any micro time-stepping for $N = 50$. Also here, different time intervals are used for each of the test models. For FSI2 we choose $I = [12, 12.5]$ and for FSI3 $I = [6, 6.5]$. The FSI2 model requires fewer iterations to reach

convergence. For FSI2 the number of evaluations of the shooting functions varies between 19 and 25 whereas for FSI3 this number is between 22 and 30. In the case of FSI2 the number of iterations changes less volatilely which might suggest better stability properties. However, for both test cases, we can notice that the overall behavior is stable throughout the simulation. Finally, we would like to point out that the jump in the first time-step is due to the initial guess that we applied given by (3.5), which we could not use at the very beginning of the simulation due to the lack of access to the previous solutions. That in itself also shows the effectiveness of this initial guess.

In the process of finalizing these simulations, we have noticed that there is a dependence between the size of a time-step and the number of Newton as well as GMRES iterations needed to reach the prescribed tolerance. Unsurprisingly, the smaller the time-step size, the number of Newton iterations needed decreases. However, at the same time, for the smaller time-steps, the GMRES method needs more iterations, especially in the final iterations of the Newton method. Given the fact that the bulk of computations lies in the GMRES method, overall the shooting method converges faster for bigger time-steps. It is illustrated in Figure 3.8. There, we compare the performance of different time-step sizes for both the FSI2 and FSI3 model problems. We examine three time-step sizes: $k_n = 0.01$, $k_n = 0.001$, and $k_n = 0.0001$. Each mark represents one iteration of a Newton method. For the FSI2 example, for all the three time-step sizes, the shooting method needed 4 Newton iterations. For the biggest time-step, we needed 20 evaluations of the shooting function, for the smaller time-step this number increased to 28 and to 43 for the smallest time-step. For the FSI3 problem, we can observe more variety in the number of Newton iterations, where for the biggest time-step 6 iterations were required instead of 4. From the biggest to the smallest time-step, the number of overall evaluations of the decoupling function is equal to 27, 40, and 54, respectively. That gives a clear increase in comparison with similar results for the FSI2 example.

3.4.3 Thermoelasticity

Similarly, as in the previous examples, we start with the analysis of the performance over a single time interval. These results are shown in Figure 3.9. Given the oscillatory nature of the thermoelasticity problem, we choose a smaller time-step of $k_2 = 0.002$ and look at the interval $I_2 = [0.002, 0.004]$. Again, we look at the case where there is not micro time-stepping with $N_2^s = N_2^T = 1$, micro time-stepping in the structure problem $N_2^s = 10$, $N_2^T = 1$, and micro time-stepping in the thermal problem $N_2^s = 1$, $N_2^T = 10$. Also here, micro time-stepping does not have any significant influence on the performance. The improvement from using the shooting method is significant. The shooting method requires 2 Newton iterations and overall 4 linear systems to reach the tolerance. The relaxation method needs 20 iterations.

Figure 3.10 shows what happens throughout the whole simulation over the time interval $I = [0.0, 0.1]$. We take a coarse mesh with $N = 50$ and $k_n = 0.002$. We similarly consider a case without any micro time-stepping at all ($N_2^s = N_2^T = 1$), uniform micro time-stepping in the structure problem ($N_2^s = 10$, $N_2^T = 1$), and micro time-stepping in the thermal problem ($N_2^s = 1$, $N_2^T = 10$). The performance of neither of the decoupling methods deteriorates

due to micro time-stepping. The shooting method needs between 4 and 11 linear systems. The relaxation requires between 17 and 28 iterations. Again, the shooting method performs much better than the relaxation method. For this example, the number of linear systems needed to converge varies across the time interval. Although the specific values for each of the methods are different, the tendencies are similar, where the number of iterations needed to reach convergence increases until the middle of the interval and then decreases. That most likely reflects the dynamics of the solution.

Chapter 4

A Posteriori Error Estimation and Adaptivity

In this chapter, we will discuss the concept of a posteriori error estimation for the temporal component applied to the finite element method. As in the previous chapters, our variational formulations will be treated as space-time problems. This way, we will be able to extend the usage of tools normally reserved only for the space component to the time component as well. To our knowledge, there are no flexible error estimators resulting in strict error bounds for complex FSI problems. Thus, we will relax the requirement of rigorous bounds and instead concentrate on flexibility and practical application. From this perspective, the *Dual Weighted Residual* (DWR) method introduced by Becker and Rannacher [3], [4] is a very good candidate.

We will start this chapter by presenting the concept of error estimation in general terms and then introduce goal oriented estimation and proceed to the DWR method. After that, the DWR method will be used for temporal estimation. Linear and then nonlinear problems will be discussed. Finally, we will present numerical experiments exemplifying both of these cases. We will derive and discretize adjoint problems needed to obtain the estimator.

4.1 Introduction to Error Estimation

Error estimation is an essential tool when analyzing most numerical methods. Thanks to such estimates, one can examine both the accuracy and stability of a given method. In the finite element method, estimates deal with a quantity $||\vec{U} - \vec{U}_{k,h}||$, where $\vec{U}$ is an exact solution to a given problem and $\vec{U}_{k,h}$ is its discrete counterpart. There exist two basic types of error estimates - *a priori* and *a posteriori* error estimates. The former depends on the exact solution $\vec{U}$ while the latter is a function of the approximate solution $\vec{U}_{k,h}$ and can be evaluated once a numerical simulation is complete. In practice, we usually do not know the exact solution $\vec{U}$ and thus, only an a posteriori error estimate is computable.

An a posteriori error estimate has one additional advantage. As long as it is possible

to localize different contributions to an estimate, one can use it as a tool in adaptive mesh refinement. With such an error indicator at hand, one can aim to reach an optimal solution. Here, an optimal solution can be understood in two ways. This can either mean that for a given tolerance ϵ, the numerical solution is reached using the minimal amount of computational resources or that for given computational resources one gets the best possible solution. To clarify, computational resources may include the number of nodes in discretization, number of time-steps in case of nonstationary problems, number of solved linear systems, or, more broadly, time and memory needed to finalize a computer simulation. Of course, in most cases reaching such optimality is not feasible. Nevertheless, applying adaptive techniques can vastly improve the performance of numerical methods. For these reasons, we will concentrate on the a posteriori error estimation techniques.

The characteristics of an ideal a posteriori error estimator include:

- **Rigorous bounds** - strict bounds grant control and information about fringe cases.

- **Asymptotic behaviour** - good error estimates must have the same convergence rate as the estimated error itself.

- **Computability** - the estimator must depend on quantities that we can actually compute.

- **Localizability** - only if the estimator is easily localizable, one can identitify the cells that should be refined.

- **Computational costs** - the cost of computing the estimator should not exceed the savings granted by the adaptive procedure.

- **Flexibility** - it is highly desirable that the estimator can be successfully used for a wide variety of problems.

We cannot expect an estimator to fulfill all of these conditions at once. Mainly the tension occurs between computability and strict error bounds. It is especially true in the case of complex applications where there are few theoretical results available. Because of that, we will consider estimators which yield good results in practical applications at the cost of error bounds. One could argue that providing these bounds is a necessary condition to call an adaptive strategy an error estimator. For simplicity, we will use this phrase nevertheless. However, we would like to stress that we are aware of this discrepancy.

Finally, it is well beyond the scope of this work to extensively present a posteriori error estimation methods. For that reason, we would like to refer to the literature to complement the overview of possible approaches. Detailed reviews can be found in [39] and [40]. We would also like to mention [41] containing a concise and technical survey on the topic.

4.1.1 Error Estimation and Adaptive Time-Stepping

In standard approaches, the discretization of the time component in partial differential equations is usually performed using the finite difference method. However, it is also possible

to use the finite element method instead. That enables access to a wider variety of error estimation tools, which in turn can be used as time-stepping adaptivity criteria. The formulation of time-stepping schemes using the space-time framework was described in detail in Chapter 2 devoted to time discretization. Unfortunately, the literature on adaptive time-stepping using tools derived from the space-time framework is still lacking. To our knowledge, the most applicable to our case is the work of Dominik Meidner, who developed in [22] and [21] a goal oriented error estimation method for the fractional step θ-method. Another worthwhile article [42] presents adaptivity for nonlinear parabolic equations in the space-time framework.

4.1.2 Goal Oriented Error Estimation

There are situations where the main interest is not the computation of a global error $||\vec{\mathbf{U}} - \vec{\mathbf{U}}_{k,h}||$. Instead, one is focused on, so-called, *quantities of interest*. That might be deformation, stress, or drag and lift coefficients. All of these quantities can be expressed in terms of some functional $Q : X \to \mathbb{R}$ acting from the space of solutions to the real numbers. In this instance, we are interested in the estimation of $|Q(\vec{\mathbf{U}}) - Q(\vec{\mathbf{U}}_{k,h})|$. This kind of approach is called *goal oriented error estimation*. Since the main topic of this thesis is the temporal aspect of a discrete solution, we will concentrate on the error given by the semi-discrete solution $|Q(\vec{\mathbf{U}}) - Q(\vec{\mathbf{U}}_k)|$.

4.2 Dual Weighted Residual Method

Traditional techniques of a posteriori error estimation usually give estimates in energy norm derived from a specific differential operator defining a given problem. They are often determined globally and include unknown constants that may depend on a problem itself. As such, they can not reliably be used as adaptivity criteria. *Dual Weighted Residual* (DWR) method extends this idea resulting in a more practically applicable method. There, one solves an additional adjoint problem, which is then used as weights capturing the influence of local variations of the residual on the global error.

DWR method has been commonly used in literature. In [43] the authors present an overview of adjoint methods in the a posteriori error estimation and postprocessing. Adaptivity for multiple target quantities in aerodynamics is explored in [44]. This method is used in the analysis of linear-elastic perfect plasticity [45] and anisotropic elements for viscous flows [46]. It is used in the context of phase-field fracture propagation in [47]. Authors in [48] apply it to complex periodic gratings. In [49] it is implemented within a neuronal network. Just from these few examples, one can see how versatile this method is.

Now we will explain the main idea behind the method based on articles [3] and [4]. We assume here a simplified set-up where the forms describing the continuous and semi-discrete problems are the same. We also assume conformity of the function spaces $X_k, Y_k \in X$. Then, the exact solution $\vec{\mathbf{U}} \in X$ is given by solving the problem

$$B(\vec{\mathbf{U}})(\mathbf{\Phi}) = F(\mathbf{\Phi}), \qquad \text{for all } \mathbf{\Phi} \in X$$

and its semi-discrete counterpart $\vec{\mathbf{U}}_k \in X_k$ comes from the solution of

$$B(\vec{\mathbf{U}}_k)(\mathbf{\Phi}_k) = F(\mathbf{\Phi}_k), \qquad \text{for all } \mathbf{\Phi}_k \in Y_k.$$

One can embed evaluation of the goal functional in the framework of optimal control. Then, computing $Q(\vec{\mathbf{U}})$ is equivalent to solving this trivial optimization problem

$$Q(\vec{\mathbf{U}}) = \text{min!} \quad \text{such that} \quad B(\vec{\mathbf{U}})(\mathbf{\Phi}) = F(\mathbf{\Phi}) \text{ for all } \mathbf{\Phi} \in X.$$

A natural way of proceeding is the introduction of a Lagrangian $\mathcal{L}(\cdot)(\cdot)$ and searching its stationary point

$$\mathcal{L}(\vec{\mathbf{U}}, \vec{\mathbf{Z}}) := Q(\vec{\mathbf{U}}) + F(\vec{\mathbf{Z}}) - B(\vec{\mathbf{U}})(\vec{\mathbf{Z}}). \tag{4.1}$$

4.2.1 Linear Case

If we assume that the form $B(\cdot)(\cdot)$ is bilinear and the functional $Q(\cdot)$ is linear, then finding the stationary point of the Lagrangian above corresponds to finding a solution $\vec{\mathbf{Z}} \in X$ to the problem

Problem 4.1. *Find $\vec{\mathbf{Z}} \in X$ such that*

$$B(\mathbf{\Xi})(\vec{\mathbf{Z}}) = Q(\mathbf{\Xi}), \qquad \text{for all } \mathbf{\Xi} \in X$$

The solution $\vec{\mathbf{Z}}$ is called an *adjoint solution*. To distinguish between $\vec{\mathbf{U}}$ and $\vec{\mathbf{Z}}$, we will call the solution $\vec{\mathbf{U}}$ a *primal solution*. The correspondence between both of them can be seen in the identity

$$F(\vec{\mathbf{U}}) = B(\vec{\mathbf{U}})(\vec{\mathbf{Z}}) = Q(\vec{\mathbf{Z}}).$$

Similarly, the semi-discrete adjoint solution $\vec{\mathbf{Z}}_k \in Y_k$ is given by

Problem 4.2. *Find $\vec{\mathbf{Z}}_k \in Y_k$ such that*

$$B(\mathbf{\Xi}_k)(\vec{\mathbf{Z}}_k) = Q(\mathbf{\Xi}_k), \qquad \text{for all } \mathbf{\Xi}_k \in X_k.$$

We can use the connection between primal and adjoint solution and apply it to errors $\mathbf{e} := \vec{\mathbf{U}} - \vec{\mathbf{U}}_k$, $\mathbf{e}^* := \vec{\mathbf{Z}} - \vec{\mathbf{Z}}_k$. With the help of Galerkin orthogonality we have

$$Q(\mathbf{e}) = B(\mathbf{e})(\vec{\mathbf{Z}}) = B(\mathbf{e})(\mathbf{e}^*) = B(\vec{\mathbf{U}})(\mathbf{e}^*) = F(\mathbf{e}^*).$$

To simplify the notation, we define primal and adjoint residuals

$$\rho(\vec{\mathbf{U}})(\mathbf{\Phi}) := F(\mathbf{\Phi}) - B(\vec{\mathbf{U}})(\mathbf{\Phi}),$$
$$\rho^*(\vec{\mathbf{Z}})(\mathbf{\Xi}) := Q(\mathbf{\Xi}) - B(\mathbf{\Xi})(\vec{\mathbf{Z}}).$$

For any $\mathbf{\Phi}_k \in Y_k$ and $\mathbf{\Xi} \in X_k$ it holds

$$\rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \mathbf{\Phi}_k) = B(\mathbf{e})(\mathbf{e}^*) = \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \mathbf{\Xi}_k),$$

which in turn leads to

$$Q(e) = \min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) = \min_{\boldsymbol{\Xi} \in X_k} \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k) = F(\mathbf{e}^*).$$

Equivalently

$$Q(\vec{\mathbf{U}}) - Q(\vec{\mathbf{U}}_k) = \frac{1}{2} \min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) + \frac{1}{2} \min_{\boldsymbol{\Xi} \in X_k} \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k). \qquad (4.2)$$

This identity serves as the basis of the a posteriori error representation discussed here. For a linear case, using both the primal and the adjoint residuals in the error estimator is not necessary. It is sufficient to use the residual of the primal problem only. However, a full formulation of the estimator will become relevant for nonlinear problems.

4.2.2 Nonlinear Case

The error representation above can be easily extended to nonlinear problems as well. Then, the adjoint solution corresponding to (4.1) is given by

Problem 4.3. *Find $\vec{Z} \in X$ such that*

$$B'_{\vec{U}}(\boldsymbol{\Xi})(\vec{Z}) = Q'_{\vec{U}}(\boldsymbol{\Xi}), \qquad \text{for all } \boldsymbol{\Xi} \in X,$$

where by $B'_{\vec{U}}(\boldsymbol{\Xi})(\cdot)$ and $Q'_{\vec{U}}(\boldsymbol{\Xi})$ we denote Gâteaux derivatives at the primal solution $\vec{U}$ in the direction of the test function $\boldsymbol{\Xi}$.

The analogous semi-discrete system reads as

Problem 4.4. *Find $\vec{Z}_k \in Y_k$ such that*

$$B'_{\vec{U}}(\boldsymbol{\Xi}_k)(\vec{Z}_k) = Q'_{\vec{U}}(\boldsymbol{\Xi}_k), \qquad \text{for all } \boldsymbol{\Xi} \in X_k.$$

Again, we associate primal and adjoint residuals with errors $\mathbf{e}$, and $\mathbf{e}^*$, respectively

$$\rho(\vec{\mathbf{U}})(\boldsymbol{\Phi}) := F(\boldsymbol{\Phi}) - B(\vec{\mathbf{U}})(\boldsymbol{\Phi}),$$
$$\rho^*(\vec{\mathbf{Z}})(\boldsymbol{\Xi}) := Q'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi}) - B'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi})(\vec{\mathbf{Z}}).$$

After some simple analysis of the Lagrangian $\mathcal{L}(\cdot)(\cdot)$, it is possible to represent the error of the goal functional as

$$Q(\vec{\mathbf{U}}) - Q(\vec{\mathbf{U}}_k) = \frac{1}{2} \min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) + \frac{1}{2} \min_{\boldsymbol{\Xi} \in X_k} \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k) + R. \qquad (4.3)$$

The remainder term $R = \mathcal{O}(|\mathbf{e}|^3, |\mathbf{e}^*|^3)$ is cubic in the errors $\mathbf{e}$, $\mathbf{e}^*$ and vanishes if $B(\cdot)(\cdot)$ is bilinear and $Q(\cdot)$ is quadratic. In this error approximation technique, the *weights* are given by the terms $\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k$ and $\vec{\mathbf{U}} - \boldsymbol{\Xi}_k$. They capture the influence of local fluctuations on the

global error.

For nonlinear problems using both of the residuals is significant. It can be seen in the relation between both of them given by

$$\rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k) = \min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) + \Delta\rho,$$

where $\Delta\rho = \mathcal{O}(|\mathbf{e}|^2, |\mathbf{e}^*|^2)$. That means that by omitting the adjoint residual $\rho^*(\cdot)(\cdot)$ we lose one order of convergence. The term $\Delta\rho$ can be understood as an indicator of the influence of nonlinearity on the error. For a more detailed analysis, again, we would like to refer to articles [3] and [4].

Remark 4.2.1. One might ask whether it is too costly to solve an additional variational problem just to obtain an adaptivity criterion. And that is a valid question. Especially in the case of linear problems where computing the adjoint solution doubles computational costs. However, by construction, the adjoint problem is always linear. So for nonlinear primal problems, solving one additional linear system is a fraction of needed computational resources.

4.2.3 Approximation of Exact Solutions

Evaluation of the error estimator in the current form of (4.3) requires exact solutions $\vec{\mathbf{U}}$ and $\vec{\mathbf{Z}}$. Since we do not have access to them, we will need to use appropriate approximations instead. Here we would like to present some of the existing approaches.

The first, and probably the most obvious one, is to compute more accurate primal $\hat{\vec{\mathbf{U}}}_k \in \hat{X}_k$ and adjoint $\hat{\vec{\mathbf{Z}}}_k \in \hat{Y}_k$ solutions and use them to substitute exact solutions from (4.3). This can be done by either choosing $\hat{X}_k$ and $\hat{Y}_k$ as higher order polynomial degree spaces or by refining mesh. Then, the weights are approximated by

$$\min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) \approx \rho(\vec{\mathbf{U}}_k)(\hat{\vec{\mathbf{Z}}}_k - i_k^Y \vec{\mathbf{Z}}_k),$$

$$\min_{\boldsymbol{\Xi} \in X_k} \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k) \approx \rho^*(\vec{\mathbf{Z}}_k)(\hat{\vec{\mathbf{U}}}_k - i_k^X \vec{\mathbf{U}}_k),$$

where by i_k^Y and i_k^X we denote interpolation operators $i_k^Y : Y_k \to \hat{Y}_k$ and $i_k^X : X_k \to \hat{X}_k$. This method yields very good results. Unfortunately, it is also computationally expensive and rarely used in practise. It was proposed in one of the original papers [4].

The method has been improved in the context of the hp-finite element method in [50]. Instead of computing the solutions $\hat{\vec{\mathbf{U}}}_k$ and $\hat{\vec{\mathbf{Z}}}_k$ globally, the authors define them locally. As a result, for every element K in the mesh, the approximate solution is only computed on a patch consisting of neighboring elements with either homogenous Neumann or Dirichlet boundary conditions. This method has the advantage of being easily parallelizable.

The final approach we would like to discuss here is the reconstruction method. There, we try to obtain a higher order solution without solving an additional problem. One can

do that by partitioning the time mesh into patches and then reinterpreting the solutions as solutions from higher polynomial degree spaces. The validity of this approach is based on the assumption of superconvergence at the nodes given sufficient regularity of the mesh. The weights are therefore approximated by

$$
\begin{aligned}
\min_{\boldsymbol{\Phi} \in Y_k} \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}} - \boldsymbol{\Phi}_k) &\approx \rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^{(1)} - \vec{\mathbf{Z}}_k), \\
\min_{\boldsymbol{\Xi} \in X_k} \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}} - \boldsymbol{\Xi}_k) &\approx \rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}}_k^{(2)} - \vec{\mathbf{U}}_k),
\end{aligned}
\tag{4.4}
$$

We will use reconstructions of one order higher than the original solutions. Assuming that the time discretization is given by the cG(1) method, the reconstructions of the primal solution $\vec{\mathbf{U}}_k^{(2)}$ and the adjoint solution $\vec{\mathbf{Z}}_k^{(1)}$ will be quadratic and linear in time, respectively. The exact formulas are given below

$$
\begin{aligned}
\vec{\mathbf{U}}_k^{(2)}(t)\Big|_{I_n} &:= \frac{(t_n - t)(t_{n+1} - t)}{k_n(k_n + k_{n+1})} \vec{\mathbf{U}}_k(t_{n-1}) + \frac{(t - t_{n-1})(t_{n+1} - t)}{k_n k_{n+1}} \vec{\mathbf{U}}_k(t_n) \\
&\quad + \frac{(t - t_{n-1})(t - t_n)}{k_n(k_n + k_{n+1})} \vec{\mathbf{U}}_k(t_{n-1}), \\
\vec{\mathbf{Z}}_k^{(1)}(t)\Big|_{I_n} &:= \frac{t - \bar{t}_{n+1}}{\bar{t}_{n-1} - \bar{t}_{n+1}} \vec{\mathbf{Z}}_k(t_{n-1}) + \frac{t - \bar{t}_{n-1}}{\bar{t}_{n+1} - \bar{t}_{n-1}} \vec{\mathbf{Z}}_k(t_{n+1})
\end{aligned}
\tag{4.5}
$$

with the interval midpoints $\bar{t}_n := \frac{t_n + t_{n-1}}{2}$. For the dG(0) method, the reconstruction of the primal solution $\vec{\mathbf{U}}_k^{(1)}$ is given by the same formulation as $\vec{\mathbf{Z}}_k^{(1)}$. For the sake of simplicity, we omitted the indices corresponding to the micro time-steps. In our algorithm, we use patches consisting of micro time-steps instead. We would like to point out that the patch structure does not necessarily coincide with the time mesh structure. It is possible for two micro time-steps from the same patch to belong to two different macro time-steps. Additionally, to make the primal reconstruction more robust, we demand two micro time-steps from the same local patch to have the same length. The reconstructions are illustrated in Figure 4.1. This technique was as well for the first time introduced in one of the original papers on the DWR method [4]. The extension to the time component was presented in the papers mention before on the topic [22], [21], and [42]. This is the approach we choose for our applications.

4.2.4 Construction of the Error Estimator

Finally, we are ready to define the error estimator. We use the identity (4.2) for linear and (4.2) for nonlinear problems. In the nonlinear case, remainder term R is omitted. The weights are approximated using the reconstruction method. We replace the exact solutions with (4.5) and use (4.4). Once we implement these steps, we get

$$
\eta(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k) := \frac{1}{2}\rho(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^{(1)} - \vec{\mathbf{Z}}_k) + \frac{1}{2}\rho^*(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}}_k^{(2)} - \vec{\mathbf{U}}_k).
\tag{4.6}
$$
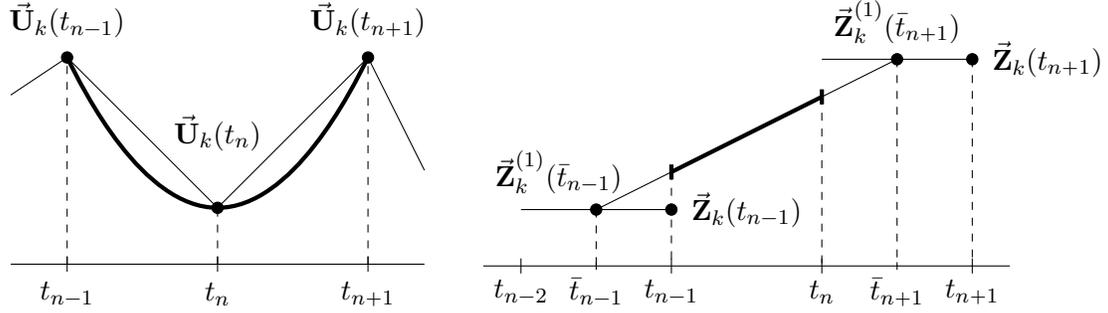
Figure 4.1: Reconstruction of the primal solution $\vec{\mathbf{U}}_k^{(2)}$ (left) and the adjoint solution $\vec{\mathbf{Z}}_k^{(1)}$ (right).

Although we have already mentioned that in the introduction to this chapter, we would like to repeat that both of these steps introduce inaccuracies. Therefore, the DWR method does not result in strict error bounds. Hence, in a strict sense, it is more of an error approximation technique rather than a true error estimator. We use this name for convenience.

In the current formulation of the estimator (4.6), it is not possible to distinguish between contributions from the fluid and the solid subproblem. Moreover, this formulation is not localized, that is, it is not possible to tell apart contributions from different time-steps. Thus, we will now split it into corresponding parts. The extended version of the adjoint continuous Problem 4.1 reads as

Problem 4.5. *Find $\vec{\mathbf{Z}} \in X$ such that*

$$B'_{\vec{\mathbf{U}}^f}(\mathbf{\Xi}^f)(\vec{\mathbf{Z}}) = Q'_{\mathbf{U}^f}(\mathbf{\Xi}^f), \qquad \text{for all } \mathbf{\Xi}^f \in X^f,$$
$$B'_{\vec{\mathbf{U}}^s}(\mathbf{\Xi}^s)(\vec{\mathbf{Z}}) = Q'_{\mathbf{U}^s}(\mathbf{\Xi}^s), \qquad \text{for all } \mathbf{\Xi}^s \in X^s.$$

The primal residual is split into

$$\rho(\vec{\mathbf{U}})(\mathbf{\Phi}) := \rho^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) + \rho^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s)$$

where

$$\rho^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f) := F^f(\mathbf{\Phi}^f) - B^f(\vec{\mathbf{U}})(\mathbf{\Phi}^f),$$
$$\rho^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s) := F^s(\mathbf{\Phi}^s) - B^s(\vec{\mathbf{U}})(\mathbf{\Phi}^s).$$

The adjoint residual is split in a similar fashion

$$\rho^*(\vec{\mathbf{Z}})(\mathbf{\Xi}) := \rho^{f,*}(\vec{\mathbf{Z}})(\mathbf{\Xi}^f) + \rho^{s,*}(\vec{\mathbf{Z}})(\mathbf{\Xi}^s)$$

and

$$\rho^{f,*}(\vec{\mathbf{Z}})(\mathbf{\Xi}^f) := Q'_{\vec{\mathbf{U}}^f}(\mathbf{\Xi}^f) - B'_{\vec{\mathbf{U}}^f}(\mathbf{\Xi}^f)(\vec{\mathbf{Z}}),$$
$$\rho^{s,*}(\vec{\mathbf{Z}})(\mathbf{\Xi}^s) := Q'_{\vec{\mathbf{U}}^s}(\mathbf{\Xi}^s) - B'_{\vec{\mathbf{U}}^s}(\mathbf{\Xi}^s)(\vec{\mathbf{Z}}).$$

Finally, we introduce a divided version of the estimator (4.6), where a division based on different subproblems is considered. Each of these contributions is further split between the primal and adjoint residuals

$$\eta_k(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k) := \eta_k^f(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^f) + \eta_k^s(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^s) + \eta_k^{f,*}(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^f) + \eta_k^{s,*}(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^s)$$

with

$$
\begin{aligned}
\eta_k^f(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^f) &:= \frac{1}{2}\rho^f(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^{f,(1)} - \vec{\mathbf{Z}}_k^f), \\
\eta_k^s(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^s) &:= \frac{1}{2}\rho^s(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^{s,(1)} - \vec{\mathbf{Z}}_k^s), \\
\eta_k^{f,*}(\vec{\mathbf{Z}}_k^f)(\vec{\mathbf{U}}_k) &:= \frac{1}{2}\rho^{f,*}(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}}_k^{f,(2)} - \vec{\mathbf{U}}_k^f), \\
\eta_k^{s,*}(\vec{\mathbf{Z}}_k^s)(\vec{\mathbf{U}}_k) &:= \frac{1}{2}\rho^{s,*}(\vec{\mathbf{Z}}_k)(\vec{\mathbf{U}}_k^{s,(2)} - \vec{\mathbf{U}}_k^s).
\end{aligned}
\tag{4.7}
$$

The localization of the error estimator (4.7) is simply given by

$$
\begin{aligned}
\eta_{n,m}^f &:= \eta^f(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^f)\Big|_{I_{n,m}^f}, & \eta_{n,m}^s &:= \eta^s(\vec{\mathbf{U}}_k)(\vec{\mathbf{Z}}_k^s)\Big|_{I_{n,m}^s}, \\
\eta_{n,m}^{f,*} &:= \eta^{f,*}(\vec{\mathbf{Z}}_k^f)(\vec{\mathbf{U}}_k)\Big|_{I_{n,m}^f}, & \eta_{n,m}^{s,*} &:= \eta^{s,*}(\vec{\mathbf{Z}}_k^s)(\vec{\mathbf{U}}_k)\Big|_{I_{n,m}^s}.
\end{aligned}
\tag{4.8}
$$

The same technique is used for the thermoelasticity problem.

4.2.5 Adaptivity

Once we have successfully defined an error indicator, we can use it to propose an adaptivity strategy. In other words, based on the error estimators (4.8), one can decide which time-steps will be refined on every refinement level. Generally, the most efficient discretizations have balanced error indicators. Thus, the most natural choice is to pick the time-steps with the highest error indicators. However, it is unclear how many of these time-steps should be picked on every refinement level. Guidelines to answer this question are usually highly heuristic. According to [25], three most common strategies are

1. **Fixed number strategy** - refine $p\%$ of the elements with the highest error indicator value.

2. **Fixed fraction strategy** - refine all the elements, whose error indicators sum up to $p\%$ of the overall error.

3. **Equilibration strategy** - refine all the elements, whose error indicators are larger than an average scaled by α.
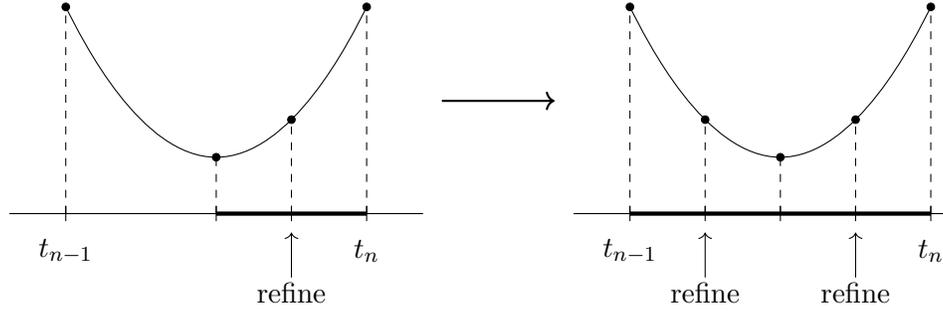
Figure 4.2: Preservation of the local patch structure. We either refine none or both time-steps within one patch.
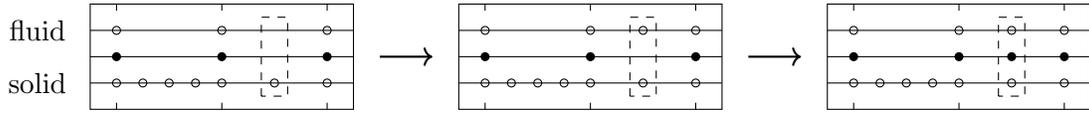
Figure 4.3: From top to bottom we present fluid nodes, global nodes, and solid nodes. On the left, we can see a mesh before refinement. In the middle drawing, due to refinement, one node is added to the fluid mesh. Since the node also belongs to the solid mesh, on the right, it is added to the global mesh as well.

Here, we are going to use the equilibration strategy adjusted to our specific case. We define the average by

$$
\bar{\eta}_k := \frac{1}{N^f} \sum_{n=1}^{N} \sum_{m=1}^{N_n^f} \left| \eta_{n,m}^f + \eta_{n,m}^{f,*} \right| + \frac{1}{N^s} \sum_{n=1}^{N} \sum_{m=1}^{N_n^s} \left| \eta_{n,m}^s + \eta_{n,m}^{s,*} \right|
$$

and propose refining criteria using

$$
\begin{aligned}
\left(\left| \eta_{n,m}^f + \eta_{n,m}^{f,*} \right| \geq \bar{\eta}_k \right) &\implies \text{refine } I_{n,m}^f, \\
\left(\left| \eta_{n,m}^s + \eta_{n,m}^{s,*} \right| \geq \bar{\eta}_k \right) &\implies \text{refine } I_{n,m}^s.
\end{aligned}
\tag{4.9}
$$

The adaptivity algorithm is supplemented by two additional objectives:

1. Preserve symmetry of the patches used for quadratic reconstruction. If only one time-step in a specific patch is refined, refine the other one as well, see Figure 4.2.

2. Introduce micro time-stepping only when necessary. An example is given in Figure 4.3.

4.3 Discretization of the Adjoint Problem

In this section, we will discuss in detail the formulation of adjoint problems. We will tackle both the linear and nonlinear cases of our model problems. For all of them, we will first present continuous, then semi-discrete, and fully discrete variational formulations.

While analyzing adjoint formulations, one encounters a dilemma similar to the one, which is very well known in the optimal control community. This correspondence is not surprising since we already mentioned a connection between the optimal control framework and the DWR method. There, one has to decide between two available approaches arising from problem optimization. In the first one, called *first-discretize-then-optimize*, the problem is first discretized and then the optimality system is solved. In the other one, *first-optimize-then-discretize* the order is reversed - one first formulates the optimality system and then applies discretization. In our case, we have to decide whether we want to consider the adjoint problem of the continuous primal problem and then discretize it, or whether we want to study the adjoint of the discrete primal problem. These two pathways can lead to the same destination but not necessarily. For linear problems, there is a have full correspondence between space-time formulations and traditional time-stepping schemes such as implicit Euler and Crank-Nicolson. As a result, one can obtain the same scheme by either discretization of the adjoint of the continuous primal variational problem or by directly taking the adjoint of the discretized primal problem. In the discretization of nonlinear problems, we introduce quadrature errors (dealing with nonlinear terms, time derivatives, and making the pressure terms fully implicit). Therefore, the first-discretize-then-optimize and first-optimize-then-discretize approaches lead to different results. In this thesis, we decided to use the former.

4.3.1 Continuous Variational Formulation

We are going to present continuous adjoint variational problems needed to successfully compute the error estimator. We will present exact formulations for each of the model problems and define the following system given by the Gâteaux derivatives of the primal formulation

Problem 4.6. *Find $\vec{Z} \in X$ such that for a given solution $\vec{U} \in X$*

$$B'_{\vec{U}^f}(\mathbf{\Xi}^f)(\vec{Z}) = Q'_{U^f}(\mathbf{\Xi}^f), \qquad \text{for all } \mathbf{\Xi}^f \in X^f,$$
$$B'_{\vec{U}^s}(\mathbf{\Xi}^s)(\vec{Z}) = Q'_{U^s}(\mathbf{\Xi}^s), \qquad \text{for all } \mathbf{\Xi}^s \in X^s.$$

The variant of this problem suitable for the volume coupled approach is given by

Problem 4.7. *Find $(\vec{Z}, S) \in X$ such that for a given solution $(\vec{U}, T) \in X$*

$$B'_{\vec{U}}(\mathbf{\Xi})(\vec{Z}, S) = Q'_U(\mathbf{\Xi}), \qquad \text{for all } \mathbf{\Xi} \in X^s,$$
$$B'_T(\Theta)(\vec{Z}, S) = Q'_T(\Theta), \qquad \text{for all } \Theta \in X^T.$$

4.3.1.1 Heat and Wave Equation

We will start with defining the adjoint test and trial functions

$$\vec{\mathbf{Z}}^f := \begin{pmatrix} z^f \\ y^f \end{pmatrix}, \quad \vec{\mathbf{Z}}^s := \begin{pmatrix} z^s \\ y^s \end{pmatrix}, \quad \vec{\mathbf{Z}} := \begin{pmatrix} \vec{\mathbf{Z}}^f \\ \vec{\mathbf{Z}}^s \end{pmatrix},$$

$$\boldsymbol{\Xi}^f := \begin{pmatrix} \zeta^f \\ \xi^f \end{pmatrix}, \quad \boldsymbol{\Xi}^s := \begin{pmatrix} \zeta^s \\ \xi^s \end{pmatrix}, \quad \boldsymbol{\Xi} := \begin{pmatrix} \boldsymbol{\Xi}^f \\ \boldsymbol{\Xi}^s \end{pmatrix}.$$

Because the forms are linear, the derivatives are created simply by switching the positions between the trial and test functions. Integration by parts was applied to the time derivatives to remove derivatives of test functions

$$B'_{\vec{\mathbf{U}}^f}(\boldsymbol{\Xi}^f)(\vec{\mathbf{Z}}) = -\int_I \langle \zeta^f, \partial_t z^f \rangle_f \, \mathrm{d}t + \int_I a'_{\vec{\mathbf{U}}^f}(\boldsymbol{\Xi}^f)(\vec{\mathbf{Z}}) \, \mathrm{d}t + (\zeta^f(T), z^f(T))_f,$$

$$B'_{\vec{\mathbf{U}}^s}(\boldsymbol{\Xi}^s)(\vec{\mathbf{Z}}) = -\int_I \langle \zeta^s, \partial_t z^s \rangle_s \, \mathrm{d}t - \int_I \langle \xi^s, \partial_t y^s \rangle_s \, \mathrm{d}t + \int_I a'_{\vec{\mathbf{U}}^s}(\boldsymbol{\Xi}^s)(\vec{\mathbf{Z}}) \, \mathrm{d}t$$
$$+ (\zeta^s(T), z^s(T))_s + (\xi^s(T), y^s(T))_s.$$

The derivatives of the forms a^f and a^s are created similarly. Note that the derivatives of some of the coupling conditions from the primal problem are shifted from the fluid to the solid form and vice versa. It happens because the forms have to be sorted regarding the test functions.

$$\begin{aligned}
a'_{\vec{\mathbf{U}}^f}(\boldsymbol{\Xi}^f)(\vec{\mathbf{Z}}) &= (\nu\nabla\zeta^f, \nabla z^f)_f + (\boldsymbol{\beta}\cdot\nabla\zeta^f, z^f)_f + (\nabla\xi^f, \nabla y^f)_f - \langle \partial_{\vec{\mathbf{n}}_f}\xi^f, y^f \rangle_\Gamma \\
&\quad + \gamma\langle \xi^f, y^f \rangle_\Gamma - \langle \nu\partial_{\vec{\mathbf{n}}_f}\zeta^f, z^f \rangle_\Gamma + \gamma\langle \nu\zeta^f, z^f \rangle_\Gamma + \langle \nu\partial_{\vec{\mathbf{n}}_f}\zeta^f, z^s \rangle_\Gamma, \\
a'_{\vec{\mathbf{U}}^s}(\boldsymbol{\Xi}^s)(\vec{\mathbf{Z}}) &= (\lambda\nabla\xi^s, \nabla z^s)_s + (\delta\nabla\zeta^s, \nabla z^s)_s - (\zeta^s, y^s)_s - \gamma\langle \xi^s, y^f \rangle_\Gamma \\
&\quad - \gamma\langle \nu\zeta^s, z^f \rangle_\Gamma - \langle \delta\partial_{\vec{\mathbf{n}}_s}\zeta^s, z^s \rangle_\Gamma.
\end{aligned} \tag{4.10}$$

4.3.1.2 FSI Problem

Here, we present the adjoint continuous formulation of the nonlinear model problem. The definitions of the test and trial functions are given by

$$\vec{\mathbf{Z}}^f := \begin{pmatrix} \mathbf{z}^f \\ \mathbf{y}^f \\ q^f \end{pmatrix}, \quad \vec{\mathbf{Z}}^s := \begin{pmatrix} \mathbf{z}^s \\ \mathbf{y}^s \end{pmatrix}, \quad \vec{\mathbf{Z}} := \begin{pmatrix} \vec{\mathbf{Z}}^f \\ \vec{\mathbf{Z}}^s \end{pmatrix},$$

$$\boldsymbol{\Xi}^f := \begin{pmatrix} \zeta^f \\ \xi^f \\ \chi^f \end{pmatrix}, \quad \boldsymbol{\Xi}^s := \begin{pmatrix} \zeta^s \\ \xi^s \end{pmatrix}, \quad \boldsymbol{\Xi} := \begin{pmatrix} \boldsymbol{\Xi}^f \\ \boldsymbol{\Xi}^s \end{pmatrix}.$$

As we will see, the formulation is arguably much more complicated than the linear case. Nevertheless, despite nonlinearities in the primal problem, the adjoint problem is always

linear. We will start the derivation of the problem with time derivatives. The fluid problem is defined as

$$\langle \rho^f J \partial_t \mathbf{v}^f, \boldsymbol{\varphi}^f \rangle_f \implies \langle \rho^f J \mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f)\partial_t \mathbf{v}^f, \mathbf{z}^f \rangle_f + \langle \rho^f J \partial_t \boldsymbol{\zeta}^f, \mathbf{z}^f \rangle_f,$$

$$\langle \rho^f J \nabla\mathbf{v}^f \mathbf{F}^{-1}\partial_t\mathbf{u}^f, \boldsymbol{\varphi}^f \rangle_f \implies \langle \rho^f J \mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f)\nabla\mathbf{v}^f\mathbf{F}^{-1}\partial_t\mathbf{u}^f, \mathbf{z}^f \rangle_f$$
$$+ \langle \rho^f J \nabla\boldsymbol{\zeta}^f\mathbf{F}^{-1}\partial_t\mathbf{u}^f, \mathbf{z}^f \rangle_f$$
$$- \langle \rho^f J \nabla\mathbf{v}^f\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f\mathbf{F}^{-1}\partial_t\mathbf{u}^f, \mathbf{z}^f \rangle_f$$
$$+ \langle \rho^f J \nabla\mathbf{v}^f\mathbf{F}^{-1}\partial_t\boldsymbol{\xi}^f, \mathbf{z}^f \rangle_f$$

To get rid of the time derivative of a test function, one can use integration by parts

$$\int_I \langle \rho^f J \partial_t\boldsymbol{\zeta}^f, \mathbf{z}^f \rangle_f \, \mathrm{d}t = - \int_I \langle \rho^f \partial_t J \boldsymbol{\zeta}^f, \mathbf{z}^f \rangle_f \, \mathrm{d}t - \int_I \langle \rho^f J \boldsymbol{\zeta}^f, \partial_t\mathbf{z}^f \rangle_f \, \mathrm{d}t$$
$$+ (\rho^f J \boldsymbol{\zeta}^f(T), \mathbf{z}^f)_f(T),$$

$$\int_I \langle \rho^f J \nabla\mathbf{v}^f\mathbf{F}^{-1}\partial_t\boldsymbol{\xi}^f, \mathbf{z}^f \rangle_f \, \mathrm{d}t = - \int_I \langle \rho^f \partial_t(J\nabla\mathbf{v}^f\mathbf{F}^{-1})\boldsymbol{\xi}^f, \mathbf{z}^f \rangle_f \, \mathrm{d}t - \int_I \langle \rho^f J\nabla\mathbf{v}^f\mathbf{F}^{-1}\boldsymbol{\xi}^f, \partial_t\mathbf{z}^f \rangle_f \, \mathrm{d}t$$
$$+ (\rho^f J\nabla\mathbf{v}^f\mathbf{F}^{-1}\boldsymbol{\xi}^f(T), \mathbf{z}^f(T))_f.$$

Now we will define $a'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi})(\vec{\mathbf{Z}})$ and show adjoints of the relevant terms. We will start with the stress term

$$(J\boldsymbol{\sigma}^f\mathbf{F}^{-T}, \nabla\boldsymbol{\varphi}^f)_f \implies (J\mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f)\boldsymbol{\sigma}^f\mathbf{F}^{-T}, \nabla\mathbf{z}^f)_f + (J(\boldsymbol{\sigma}^f)'_{\vec{\mathbf{U}}^f}(\boldsymbol{\Xi}^f)\mathbf{F}^{-T}, \nabla\mathbf{z}^f)_f$$
$$- (J\boldsymbol{\sigma}^f\mathbf{F}^{-T}(\nabla\boldsymbol{\xi}^f)^T\mathbf{F}^{-T}, \nabla\mathbf{z}^f)_f.$$

That is followed by the adjoint of the convective term

$$(J\nabla\mathbf{v}^f\mathbf{F}^{-1}\mathbf{v}^f, \boldsymbol{\varphi}^f)_f \implies (J\mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f)\nabla\mathbf{v}^f\mathbf{F}^{-1}\mathbf{v}^f, \mathbf{z}^f)_f + (J\nabla\boldsymbol{\zeta}^f\mathbf{F}^{-1}\mathbf{v}^f, \mathbf{z}^f)_f$$
$$- (J\nabla\mathbf{v}^f\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f\mathbf{F}^{-1}\mathbf{v}^f, \mathbf{z}^f)_f + (J\nabla\mathbf{v}^f\mathbf{F}^{-1}\boldsymbol{\zeta}^f, \mathbf{z}^f)_f$$

as well as the incompressibility condition

$$(J\mathrm{tr}(\mathbf{F}^{-1}\nabla\mathbf{v}^f), \omega^f)_f \implies (J\mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f)\mathrm{tr}(\mathbf{F}^{-1}\nabla\mathbf{v}^f), q^f)_f - (J\mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f\mathbf{F}^{-1}\nabla\mathbf{v}^f), q^f)_f$$
$$+ (J\mathrm{tr}(\mathbf{F}^{-1}\nabla\boldsymbol{\zeta}^f), q^f)_f.$$

The derivatives of the stress tensors are defined as

$$(\boldsymbol{\sigma}^f)'_{\vec{\mathbf{U}}^f}(\boldsymbol{\Xi}^f) = \rho^f\nu^f\left(\nabla\boldsymbol{\zeta}^f\mathbf{F}^{-1} - \nabla\mathbf{v}^f\mathbf{F}^{-1}\nabla\boldsymbol{\xi}^f\mathbf{F}^{-1} - \mathbf{F}^{-T}(\nabla\boldsymbol{\xi}^f)^T\mathbf{F}^{-T}(\nabla\mathbf{v}^f)^T\right.$$
$$\left. + \mathbf{F}^{-T}(\nabla\boldsymbol{\zeta}^f)^T\right) - \chi^f\mathbf{I},$$
$$(\boldsymbol{\sigma}^s)'_{\vec{\mathbf{U}}^s}(\boldsymbol{\Xi}^s) = \mu^s(\nabla\boldsymbol{\xi}^s)^T\mathbf{F} + \mu^s\mathbf{F}^T\nabla\boldsymbol{\xi}^s + \lambda^s\mathrm{tr}\left((\nabla\boldsymbol{\xi}^s)^T\mathbf{F} + \mathbf{F}^T\nabla\boldsymbol{\xi}^s\right).$$

4.3.1.3 Thermoelasticity

Following the previously establish adjoint problems, we start by defining the trial and test functions, which for the structure problem are given by

$$\vec{\mathbf{Z}} := \begin{pmatrix} \mathbf{z} \\ \mathbf{y} \end{pmatrix}, \quad \vec{\mathbf{\Xi}} := \begin{pmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\xi} \end{pmatrix}.$$

The trial and test function for the temperature problem will be denoted by S and Θ, respectively. The resulting adjoint variational forms are listed below

$$
\begin{aligned}
B'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}})(\vec{\mathbf{Z}}, S) = &- \int_I \langle \rho \boldsymbol{\zeta}, \partial_t \mathbf{z} \rangle \, \mathrm{d}t - \int_I \langle \boldsymbol{\xi}, \partial_t \mathbf{y} \rangle \, \mathrm{d}t + \int_I a'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}})(\vec{\mathbf{Z}}, S) \, \mathrm{d}t \\
&+ (\rho \boldsymbol{\zeta}(T), \mathbf{z}(T)) + (\boldsymbol{\xi}(T), \mathbf{y}(T)), \\
B'_T(\Theta)(\vec{\mathbf{Z}}, S) = &- \int_I \langle C\rho\Theta, \partial_t S \rangle \, \mathrm{d}t + \int_I a'_T(\Theta)(\vec{\mathbf{Z}}, S) \, \mathrm{d}t + (C\rho\Theta(T), S(T)).
\end{aligned}
$$

In these equations unfortunately we were not able to avoid overlapping notations. Therefore, we would like to point out that the notation T denotes here both the primal solution of the temperature equation as well as the final time of the time interval $I = [0, T]$. The adjoint forms used in the formulation above, are equal to

$$
\begin{aligned}
a'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}})(\vec{\mathbf{Z}}, S) =& \left(\boldsymbol{\sigma}'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}}), \nabla \mathbf{z} \right) + \left(\delta \nabla \boldsymbol{\zeta}, \nabla \mathbf{z} \right) - \left(\boldsymbol{\xi}, \mathbf{y} \right) - \left(\alpha \mathrm{div}\, \boldsymbol{\xi}, S \right) \\
a'_T(\Theta)(\vec{\mathbf{Z}}, S) =& \left(\boldsymbol{\sigma}'_T(\Theta), \nabla \mathbf{z} \right) + \left(\kappa \nabla \Theta, \nabla S \right),
\end{aligned}
$$

where

$$
\begin{aligned}
\boldsymbol{\sigma}'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}}) =& 2\mu \boldsymbol{\epsilon}'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}}) + \lambda \mathrm{tr}\, \boldsymbol{\epsilon}'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}}) \boldsymbol{I}, \\
\boldsymbol{\sigma}'_T(\Theta) =& 2\mu'_T(\Theta)\boldsymbol{\epsilon} + \lambda'_T(\Theta) \mathrm{tr}\, \boldsymbol{\epsilon} \mathbf{I}.
\end{aligned}
$$

Below we list derivatives of the material parameters

$$
\boldsymbol{\epsilon}'_{\vec{\mathbf{U}}}(\vec{\mathbf{\Xi}}) = \frac{1}{2} \left(\nabla \boldsymbol{\xi} + (\nabla \boldsymbol{\xi})^T \right), \quad \lambda'_T(\Theta) = \frac{-\nu E_0 \beta S}{(1+\nu)(1-2\nu)}, \quad \mu'_T(\Theta) = \frac{-E_0 \beta S}{2(1+\nu)}.
$$

In the formulation of these equations, we can see a slight nonlinearity of this problem.

4.3.2 Semi-discrete Variational Formulation

Let us remind that the primal solution $\vec{\mathbf{U}}_k$ was computed using the trial space X_k consisting of piecewise linear functions in time and the test space Y_k of piecewise constant functions in time. As we already stated, for a linear case, the Gâteaux derivative is given by swapping the positions of the test and trial functions. As a consequence, in the discretization, the trial and the test spaces are swapped as well. That in turn directly impacts the time-stepping

scheme for the adjoint problem. This way, we can acquire a scheme similar to the Crank-Nicolson scheme. The main difference is that the adjoint semi-discrete problem has to be solved backward in time - first we acquire the solution $\vec{Z}_N$, then $\vec{Z}_{N-1}$, and so on. Here as well we skip the indices corresponding to micro time-stepping.

We start with presenting the semi-discrete adjoint variational problems

Problem 4.8. *Find* $\vec{Z}_k \in Y_k$ *such that for a given solution* $\vec{U}_k \in X_k$

$$(B_k)'_{\vec{U}_k^f}(\Xi_k^f)(\vec{Z}_k) = Q'_{U_k^f}(\Xi_k^f), \qquad \text{for all } \Xi_k^f \in X_k^f,$$

$$(B_k)'_{\vec{U}_k^s}(\Xi_k^s)(\vec{Z}_k) = Q'_{U_k^s}(\Xi_k^s), \qquad \text{for all } \Xi_k^s \in X_k^s.$$

Problem 4.9. *Find* $(\vec{Z}_k, S_k) \in Y_k$ *such that for a given solution* $(\vec{U}_k, T_k) \in X_k$

$$(B_k)'_{\vec{U}_k}(\Xi_k)(\vec{Z}_k, S_k) = Q'_{U_k}(\Xi_k), \qquad \text{for all } \Xi_k \in X_k^s,$$

$$(B_k)'_{T_k}(\Theta_k)(\vec{Z}_k, S_k) = Q'_{T_k}(\Theta_k), \qquad \text{for all } \Theta_k \in X_k^T.$$

4.3.2.1   Heat and Wave Equation

Let us remind that the linear forms can be obtained by either discretization of the continuous adjoint problem or by considering an adjoint problem to the semi-discrete primal problem. As a result, we obtain a scheme given by

$$\left(\zeta_k^f(t_n), z_k^f(t_n)\right)_f - \left(\zeta_k^f(t_n), z_k^f(t_{n+1})\right)_f$$

$$+ \frac{1}{2}k_n a'_{\vec{U}_k^f}(\Xi_k^f(t_n))(\vec{Z}_k^f(t_n), \mathbf{I}_k^f \vec{Z}_k^s(t_n))$$

$$+ \frac{1}{2}k_{n+1} a'_{\vec{U}_k^f}(\Xi_k^f(t_n))(\vec{Z}_k^f(t_{n+1}), \mathbf{I}_k^f \vec{Z}_k^s(t_{n+1})) = Q'_{\vec{U}_k^f}(\Xi_k^f)\Big|_{I_n},$$

$$\left(\zeta_k^s(t_n), z_k^s(t_n)\right)_s - \left(\zeta_k^s(t_n), z_k^s(t_{n+1})\right)_s$$

$$+ \left(\xi_k^s(t_n), y_k^s(t_n)\right)_s - \left(\xi_k^s(t_n), y_k^s(t_{n+1})\right)_s$$

$$+ \frac{1}{2}k_n a'_{\vec{U}_k^s}(\Xi_k^s(t_n))(\mathbf{I}_k^s \vec{Z}_k^f(t_n), \vec{Z}_k^s(t_n))$$

$$+ \frac{1}{2}k_{n+1} a'_{\vec{U}_k^s}(\Xi_k^s(t_n))(\mathbf{I}_k^s \vec{Z}_k^f(t_{n+1}), \vec{Z}_k^s(t_{n+1})) = Q'_{\vec{U}_k^s}(\Xi_k^s)\Big|_{I_n}.$$

The solution at the initial time instead follows from

$$-\left(\zeta_k^f(t_0), z_k^f(t_1)\right)_f + \frac{1}{2}k_1 a'_{\vec{U}_k^f}(\Xi_k^f(t_0))(\vec{Z}_k^f(t_1), \mathbf{I}_k^f \vec{Z}_k^s(t_1)) = Q'_{\vec{U}_k^f}(\Xi_k^f)\Big|_{I_1},$$

$$-\left(\zeta_k^s(t_0), z_k^s(t_1)\right)_s - \left(\xi_k^s(t_0), y_k^s(t_1)\right)_s + \frac{1}{2}k_1 a'_{\vec{U}_k^s}(\Xi_k^s(t_0))(\mathbf{I}_k^s \vec{Z}_k^f(t_1), \vec{Z}_k^s(t_1)) = Q'_{\vec{U}_k^s}(\Xi_k^s)\Big|_{I_1}.$$

We obtain the last solution by

$$\left(\zeta_k^f(t_N), z_k^f(t_N)\right)_f + \frac{1}{2}k_N a'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f(t_N))(\vec{\mathbf{Z}}_k^f(t_N), \mathbf{I}_k^f\vec{\mathbf{Z}}_k^s(t_N)) = Q'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f)\Big|_{I_N},$$

$$\left(\zeta_k^s(t_N), z_k^s(t_N)\right)_s + \left(\xi_k^s(t_N), y_k^s(t_N)\right)_s + \frac{1}{2}k_N a'_{\vec{\mathbf{U}}_k^s}(\mathbf{\Xi}_k^s(t_N))(\mathbf{I}_k^s\vec{\mathbf{Z}}_k^f(t_N), \vec{\mathbf{Z}}_k^s(t_N)) = Q'_{\vec{\mathbf{U}}_k^s}(\mathbf{\Xi}_k^s)\Big|_{I_N}.$$

This scheme is similar to the one given by standard θ-schemes. The difference lies in the fact that here the test functions are piecewise linear and the terms in the time-stepping scheme have to be sorted with regard to the indices of the test functions.

4.3.2.2 FSI Problem

Using Gâteaux derivative on the semi-discrete primal problem we can obtain a semi-discrete adjoint formulation. The discretization of the time derivatives in the fluid problem is given by

$$\langle \rho^f J_k \partial_t \mathbf{v}_k^f, \boldsymbol{\varphi}_k^f \rangle_f \Longrightarrow \left(\rho^f J'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f(t_n))(\mathbf{v}_k^f(t_n) - \mathbf{v}_k^f(t_{n-1})), z_k^f(t_n)\right)_f$$
$$+ \left(\rho^f J_k(t_n)\boldsymbol{\zeta}_k^f(t_n), (\mathbf{z}_k^f(t_n) - \mathbf{z}_k^f(t_{n+1})\right)_f,$$

$$\langle \rho^f J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1} \partial_t \mathbf{u}_k^f, \boldsymbol{\varphi}_k^f \rangle_f \Longrightarrow -\frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})'_{\mathbf{U}_k^f}(\mathbf{\Xi}_k^f(t_n))\mathbf{u}_k^f(t_n), \mathbf{z}_k^f(t_n)\right)_f$$
$$+ \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})'_{\mathbf{U}_k^f}(\mathbf{\Xi}_k^f(t_n))\mathbf{u}_k^f(t_{n-1}), \mathbf{z}_k^f(t_n)\right)_f$$
$$- \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})'_{\mathbf{U}_k^f}(\mathbf{\Xi}_k^f(t_n))\mathbf{u}_k^f(t_{n+1}), \mathbf{z}_k^f(t_{n+1})\right)_f$$
$$+ \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})'_{\mathbf{U}_k^f}(\mathbf{\Xi}_k^f(t_n))\mathbf{u}_k^f(t_n), \mathbf{z}_k^f(t_{n+1})\right)_f$$
$$- \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})(t_n)\boldsymbol{\xi}_k^f(t_n), \mathbf{z}_k^f(t_n)\right)_f$$
$$- \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})(t_{n-1})\boldsymbol{\xi}_k^f(t_n), \mathbf{z}_k^f(t_n)\right)_f$$
$$+ \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})(t_{n+1})\boldsymbol{\xi}_k^f(t_n), \mathbf{z}_k^f(t_{n+1})\right)_f$$
$$+ \frac{1}{2}\left(\rho^f (J_k \nabla \mathbf{v}_k^f \mathbf{F}_k^{-1})(t_n)\boldsymbol{\xi}_k^f(t_n), \mathbf{z}_k^f(t_{n+1})\right)_f.$$

Here, we assumed that $n \neq 1$ and $n \neq N$. In these cases, the scheme has to be adjusted analogously to the linear case. The discretization of the time-derivatives in the solid problem, up to multiplication by the solid density ρ^s, is the same as in the previously presented linear

example. Finally, we account for the fully implicit terms

$$
\begin{aligned}
a^f(\mathbf{U}_k)(\mathbf{\Phi}_k^f) \Longrightarrow \; & (\theta + k_n)k_n(a_{CN})'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f(t_n))(\vec{\mathbf{Z}}_k^f(t_n), \mathbf{I}_k^f \vec{\mathbf{Z}}_k^s(t_n)) \\
& + (\theta - k_{n+1})k_{n+1}(a_{CN})'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f(t_n))(\vec{\mathbf{Z}}_k^f(t_{n+1}), \mathbf{I}_k^f \vec{\mathbf{Z}}_k^s(t_{n+1})) \\
& + k_n(a_{IE})'_{\vec{\mathbf{U}}_k^f}(\mathbf{\Xi}_k^f(t_n))(\vec{\mathbf{Z}}_k^f(t_n), \mathbf{I}_k^f \vec{\mathbf{Z}}_k^s(t_n)), \\
a^s(\mathbf{U}_k)(\mathbf{\Phi}_k^s) \Longrightarrow \; & (\theta + k_n)k_n a'_{\vec{\mathbf{U}}_k^s}(\mathbf{\Xi}_k^s(t_n))(\mathbf{I}_k^s \vec{\mathbf{Z}}_k^f(t_n), \vec{\mathbf{Z}}_k^s(t_n)) \\
& + (\theta - k_{n+1})k_{n+1} a'_{\vec{\mathbf{U}}_k^s}(\mathbf{\Xi}_k^s(t_n))(\mathbf{I}_k^s \vec{\mathbf{Z}}_k^f(t_{n+1}), \vec{\mathbf{Z}}_k^s(t_{n+1})).
\end{aligned}
$$

4.3.2.3 Thermoelasticity

The semi-discrete forms describing the thermoelasticity problem are very similar to the ones from the first model problem, where we coupled together heat and wave equations. In both cases, we arrive at an adjoint version of the Crank-Nicolson time-stepping scheme. The only difference lies in the way this scheme is acquired. As we remember, the first model problem is fully linear, and therefore taking the adjoint of the continuous problem and then discretizing it as well as deriving the adjoint of the semi-discrete primal problem lead to the same destination. This model problem is slightly nonlinear which means that the equivalence described above is lost. As a result, the scheme that we are going to present comes only from directly applying the Gâteaux derivative to the semi-discrete primal variational formulation

$$
\begin{aligned}
& \big(\rho\boldsymbol{\zeta}_k(t_n), \mathbf{z}_k(t_n)\big) - \big(\rho\boldsymbol{\zeta}_k(t_n), \mathbf{z}_k(t_{n+1})\big) \\
& + \big(\boldsymbol{\xi}_k(t_n), \mathbf{y}_k(t_n)\big) - \big(\boldsymbol{\xi}_k(t_n), \mathbf{y}_k(t_{n+1})\big) \\
& + \frac{1}{2}k_n a'_{\vec{\mathbf{U}}_k}(\mathbf{\Xi}_k(t_n))(\vec{\mathbf{Z}}_k(t_n), I_k^s S_k(t_n)) \\
& + \frac{1}{2}k_{n+1} a'_{\vec{\mathbf{U}}_k}(\mathbf{\Xi}_k(t_n))(\vec{\mathbf{Z}}_k(t_{n+1}), I_k^s S_k(t_{n+1})) = \int_{I_n} Q'_{\vec{\mathbf{U}}_k}(\mathbf{\Xi}_k)\,\mathrm{d}t, \\
& \big(C\rho\Theta_k(t_n), S_k(t_n)\big) - \big(C\rho\Theta_k(t_n), S_k(t_{n+1})\big) \\
& + \frac{1}{2}k_n a'_{T_k}(\Theta_k(t_n))(\mathbf{I}_k^T \vec{\mathbf{Z}}_k(t_n), S_k(t_n)) \\
& + \frac{1}{2}k_{n+1} a'_{T_k}(\Theta_k(t_n))(\mathbf{I}_k^T \vec{\mathbf{Z}}_k(t_{n+1}), S_k(t_{n+1})) = \int_{I_n} Q'_{T_k}(\Theta_k)\,\mathrm{d}t.
\end{aligned}
$$

We omit the equations corresponding to the fringe cases of the very first and very last time-step given how similar they are to the previously shown examples.

4.3.3 Fully Discrete Variational Formulation

For the space discretization, we use the same classical approaches and function spaces already discussed in Section 2.2 devoted to the space discretization of the primal problems.

This leads us to one of the following fully discrete adjoint formulations

Problem 4.10. *Find $\vec{Z}_{k,h} \in Y_{k,h}$ such that for a given solution $\vec{U}_{k,h} \in X_{k,h}$*

$$(B_{k,h})'_{\vec{U}^f_{k,h}}(\Xi^f_{k,h})(\vec{Z}_{k,h}) = Q'_{U^f_{k,h}}(\Xi^f_{k,h}), \qquad \text{for all } \Xi^f_{k,h} \in X^f_{k,h},$$

$$(B_{k,h})'_{\vec{U}^s_{k,h}}(\Xi^s_{k,h})(\vec{Z}_{k,h}) = Q'_{U^s_{k,h}}(\Xi^s_{k,h}), \qquad \text{for all } \Xi^s_{k,h} \in X^s_{k,h}.$$

Problem 4.11. *Find $(\vec{Z}_{k,h}, S_{k,h}) \in Y_{k,h}$ such that for a given solution $(\vec{U}_{k,h}, T_{k,h}) \in X_{k,h}$*

$$(B_k)'_{\vec{U}_{k,h}}(\Xi_{k,h})(\vec{Z}_{k,h}, S_{k,h}) = Q'_{U_{k,h}}(\Xi_{k,h}), \qquad \text{for all } \Xi_{k,h} \in X^s_{k,h},$$

$$(B_k)'_{T_{k,h}}(\Theta_{k,h})(\vec{Z}_{k,h}, S_{k,h}) = Q'_{T_{k,h}}(\Theta_{k,h}), \qquad \text{for all } \Theta_{k,h} \in X^T_{k,h}.$$

In the first two model problems, we have to account for the fully discrete version of the Nitsche coupling conditions.

4.4 Numerical Results

In this section, we will test the error estimator given by the DWR method on our model problems. To evaluate the performance of our method on each of them, we introduce the notion of effectivity given by

$$\text{eff}_k := \frac{\eta_k}{\widetilde{Q}(\vec{U}) - Q(\vec{U}_k)},$$

where $\widetilde{Q}(\vec{U})$ stands for an appropriate approximation of the exact value of the functional $Q(\vec{U})$. This quantity can be computed using extrapolation in time. Indeed, let us assume that the functional Q is computed on a sequence of time meshes. Given these values, we can formulate a system of equations

$$\begin{cases} Q_1 := Q(\vec{U}_k) = \widetilde{Q}(\vec{U}) + Ck^q & = \widetilde{Q}(\vec{U}) + \widetilde{C}, \\ Q_2 := Q(\vec{U}_{\frac{k}{2}}) = \widetilde{Q}(\vec{U}) + C\left(\frac{k}{2}\right)^q & = \widetilde{Q}(\vec{U}) + \widetilde{C}\left(\frac{1}{2}\right)^q, \\ Q_3 := Q(\vec{U}_{\frac{k}{4}}) = \widetilde{Q}(\vec{U}) + C\left(\frac{k}{4}\right)^q & = \widetilde{Q}(\vec{U}) + \widetilde{C}\left(\frac{1}{4}\right)^q, \end{cases}$$

where q stands for convergence rate and $\widetilde{C}$ for an extrapolation constant. Below we state the solutions to this system

$$\widetilde{Q} = \frac{Q_1 Q_3 - (Q_2)^2}{Q_1 - 2Q_2 + Q_3}, \quad q = \frac{\ln\left(\frac{Q_2 - Q_3}{Q_1 - Q_2}\right)}{-\ln 2}, \quad \widetilde{C} = \frac{(Q_1 - Q_2)^2}{Q_1 - 2Q_2 + Q_3}.$$

For each model problem, we will examine different goal functionals in the effort to extract the characteristics of each of the subproblems. We will first look at uniform time-steppping and its residuals and effectivities. The three finest meshes from the uniform time-stepping will be used to approximate the exact value of the goal functional. Then we will either proceed to asymmetrical uniform time-stepping, where one subsystem has twice finer time mesh than the other or to fully adaptive time-stepping.
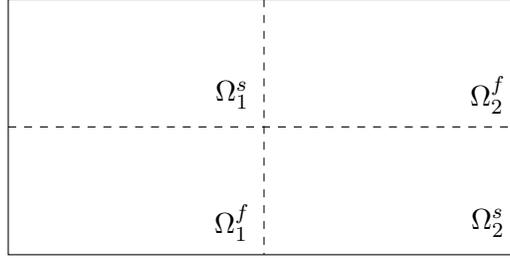
Figure 4.4: In this figure we show the splitting of the domains used in the computation of functionals, where we only use the right half of the domains, that is Ω_2^f for the functionals concentrated on the fluid domain and Ω_2^s in case the functional is applied to the solid problem.

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$3.62 \cdot 10^{-8}$	$5.01 \cdot 10^{-10}$	$1.05 \cdot 10^{-7}$	$5.03 \cdot 10^{-10}$	$1.42 \cdot 10^{-7}$	$8.06 \cdot 10^{-8}$	1.76
100	$9.66 \cdot 10^{-9}$	$1.37 \cdot 10^{-10}$	$9.96 \cdot 10^{-9}$	$1.40 \cdot 10^{-10}$	$1.99 \cdot 10^{-8}$	$2.05 \cdot 10^{-8}$	0.97
200	$2.48 \cdot 10^{-9}$	$3.00 \cdot 10^{-11}$	$2.52 \cdot 10^{-9}$	$3.02 \cdot 10^{-11}$	$5.07 \cdot 10^{-9}$	$5.22 \cdot 10^{-9}$	0.97
400	$6.28 \cdot 10^{-10}$	$9.44 \cdot 10^{-12}$	$6.33 \cdot 10^{-10}$	$9.56 \cdot 10^{-12}$	$1.28 \cdot 10^{-9}$	$1.31 \cdot 10^{-9}$	0.98
800	$1.58 \cdot 10^{-10}$	$2.02 \cdot 10^{-12}$	$1.58 \cdot 10^{-10}$	$2.06 \cdot 10^{-12}$	$3.20 \cdot 10^{-10}$	$3.28 \cdot 10^{-10}$	0.98

Table 4.1: Residuals and effectivities for the fluid subdomain functional in case of uniform time-stepping $N^f = N^s = N$.

4.4.1 Heat and Wave Equation

We start by examining our first linear example. We will look at three different goal functionals here - one concentrated in the fluid domain, one concentrated in the solid domain, and the third being the sum of the first two. Each of the functionals is associated with a different source term, that is a different configuration of the right hand side. For each of these functionals, we will only use the right half of each of the domain. The sketch of this partitioning is presented in Figure 4.4. As the time interval, we take $I = [0, 1]$.

4.4.1.1 First Test Case

The first functional corresponds to Configuration 1, where we apply an external force only to the fluid problem. The functional itself is also nonzero only on the fluid domain, where we look at the gradient of velocity scaled by the diffusive parameter ν over a domain $\Omega_2^f = (2, 4) \times (0, 1)$. We use the notation $v_2^f := v^f|_{\Omega_2^f}$

$$Q^f(\vec{\mathbf{U}}) := \int_0^T \nu \|\nabla v_2^f\|_f^2 \, \mathrm{d}t, \quad Q^s(\vec{\mathbf{U}}) := 0.$$

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$9.66 \cdot 10^{-9}$	$4.99 \cdot 10^{-10}$	$9.96 \cdot 10^{-9}$	$5.01 \cdot 10^{-10}$	$2.06 \cdot 10^{-8}$	$2.17 \cdot 10^{-8}$	0.95
100	$2.48 \cdot 10^{-9}$	$1.37 \cdot 10^{-10}$	$2.52 \cdot 10^{-9}$	$1.39 \cdot 10^{-10}$	$5.28 \cdot 10^{-9}$	$5.45 \cdot 10^{-9}$	0.97
200	$6.28 \cdot 10^{-10}$	$2.99 \cdot 10^{-11}$	$6.33 \cdot 10^{-10}$	$3.01 \cdot 10^{-11}$	$1.32 \cdot 10^{-9}$	$1.43 \cdot 10^{-9}$	0.92
400	$1.58 \cdot 10^{-10}$	$9.44 \cdot 10^{-12}$	$1.58 \cdot 10^{-10}$	$9.56 \cdot 10^{-12}$	$3.35 \cdot 10^{-10}$	$3.58 \cdot 10^{-10}$	0.94

Table 4.2: Residuals and effectivities for the fluid subdomain functional in case of uniform refinement in the fluid domain $N^f = 2N$ and $N^s = N$.

N	N^f	N^s	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	56	50	$3.08 \cdot 10^{-8}$	$5.01 \cdot 10^{-10}$	$3.16 \cdot 10^{-8}$	$5.04 \cdot 10^{-10}$	$6.34 \cdot 10^{-8}$	$6.64 \cdot 10^{-8}$	0.95
50	100	50	$9.66 \cdot 10^{-9}$	$4.99 \cdot 10^{-10}$	$9.96 \cdot 10^{-9}$	$5.01 \cdot 10^{-10}$	$2.06 \cdot 10^{-8}$	$2.17 \cdot 10^{-8}$	0.95
50	110	50	$8.21 \cdot 10^{-9}$	$4.99 \cdot 10^{-10}$	$8.32 \cdot 10^{-9}$	$5.02 \cdot 10^{-10}$	$1.75 \cdot 10^{-8}$	$1.84 \cdot 10^{-8}$	0.95
50	156	50	$5.08 \cdot 10^{-9}$	$4.99 \cdot 10^{-10}$	$5.18 \cdot 10^{-9}$	$4.97 \cdot 10^{-10}$	$1.13 \cdot 10^{-8}$	$1.20 \cdot 10^{-8}$	0.94

Table 4.3: Residuals and effectivities for the fluid subdomain functional in case of adaptive time-stepping.

Its integration in time is computed using a 2-point Gaussian quadrature rule. The Gâteaux derivative is equal to

$$(Q^f)'_{\vec{\mathbf{U}}}(\mathbf{\Xi}^f) = \int_0^T 2\nu \big(\nabla v_2^f, \nabla \zeta_2^f \big)_f \, \mathrm{d}t,$$

where similarly $\zeta_2^f := \zeta^f \big|_{\Omega_2^f}$.

Table 4.1 shows the values of partial residuals, as well as the effectivities on a series of five uniform time meshes, without any micro time-stepping with $N^f = N^s = N$. On the coarsest mesh, we have $N = 50$ and $k_n = 0.02$. On the finest mesh, these values are equal to $N = 800$ and $k_n = 0.00125$. The exact value of the goal functional is reconstructed using the previously described extrapolation in time procedure. Using the three finest available meshes, we obtained the value $\widetilde{Q} = 6.029469 \cdot 10^{-5}$. The corresponding convergence rate is almost perfectly quadratic with $q = 1.997127$. In the table, we can see that the effectivities, except the coarsest mesh, are very close to one. That convincingly proves the excellent performance of the error estimator in this example. We would also like to note that the primal and adjoint residuals corresponding to each of the domains are very close to each other. That can be easily explained by the linear nature of this problem. On each level of refinement, the residuals in the fluid domain are roughly two orders of magnitude higher. That suggests introducing further refinements in this domain.

In Table 4.2 we show the results where the time-steps in the fluid domain were once uniformly refined and therefore $N^f = 2N$ and $N^s = N$. As the reference value $\bar{Q}$, we take the same number as in the previous, fully uniform, example. In order not to exceed its accuracy,
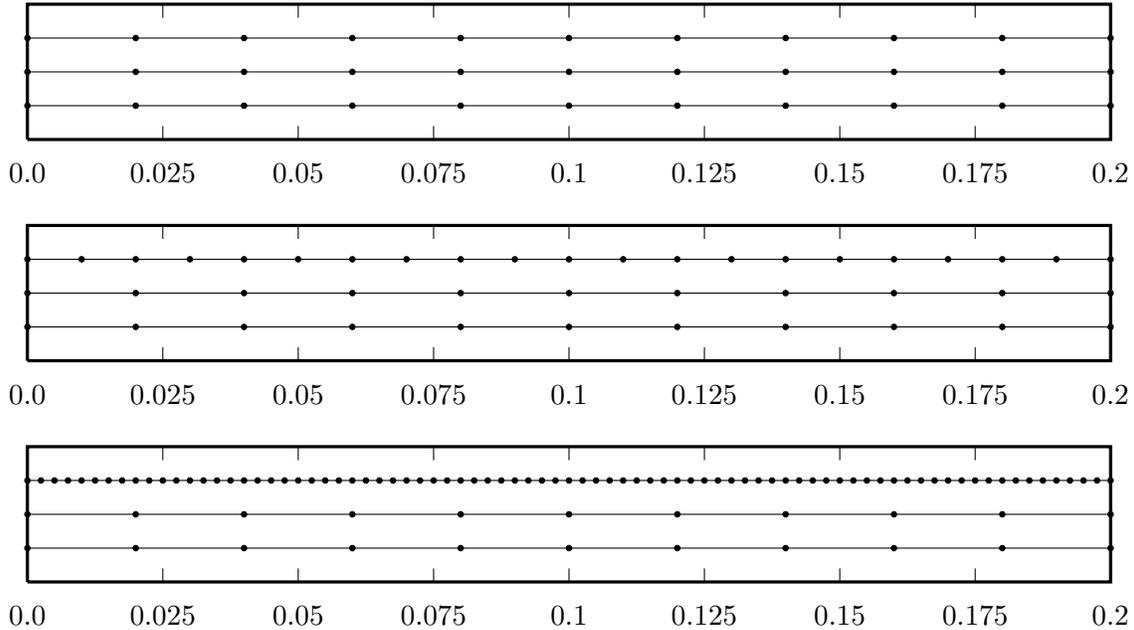
Figure 4.5: This figure shows the adaptive time mesh for the fluid functional.

only four simulations are performed. The deterioration of the effectivity is very minimal and we can achieve a significant reduction of the extrapolated error $\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$. In fact, based on the extrapolated error, we can deduce that partial refinement in the fluid domain is just as effective in its reduction as a uniform refinement in both of the domains. Partial residuals indicate that we should continue refining the time mesh in the fluid domain.

In Table 4.3 we collect the numbers and consider fully adaptive time-stepping. Our procedure starts with the coarsest mesh $N^f = N^s = N = 50$. From there, we run four iterations of the adaptive time-stepping. The total sum of time-steps in both of the domains is equal to $N^f + N^s$. Again, we obtain very good performance with only slight deterioration of the effectivities. Because of the concentration of errors in the fluid domain, only the fluid time-steps were refined. At the end of this adaptive procedure, we obtain the extrapolated error equal to $1.20 \cdot 10^{-8}$ with a total number of time-steps equal to 206. The closest corresponding value from Table 4.1 is $1.99 \cdot 10^{-8}$ in the case of $N = 100$ and therefore 200 time-steps in both domains. That is a significant reduction with only 6 additional time-steps.

Finally, Figure 4.5 represents a sequence of adaptive meshes that result from the previously discussed refinement strategy. From top to bottom, we show the initial time mesh, mesh after two steps and four steps of our adaptive procedure. In the top row, we show the initial mesh with 50 macro time-steps and no further splitting in fluid and solid. For a better presentation, we only take a small subset of the temporal interval $[0.0, 0.2]$. In the middle plot, one can see the mesh after 2 steps of adaptive refinement. The bottom plot illustrates the mesh

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$2.03 \cdot 10^{-10}$	$2.66 \cdot 10^{-6}$	$1.93 \cdot 10^{-10}$	$1.03 \cdot 10^{-5}$	$1.30 \cdot 10^{-5}$	$2.49 \cdot 10^{-5}$	0.52
100	$4.53 \cdot 10^{-11}$	$2.59 \cdot 10^{-6}$	$4.26 \cdot 10^{-11}$	$2.67 \cdot 10^{-6}$	$5.26 \cdot 10^{-6}$	$4.77 \cdot 10^{-6}$	1.10
200	$1.28 \cdot 10^{-11}$	$5.18 \cdot 10^{-7}$	$1.26 \cdot 10^{-11}$	$5.21 \cdot 10^{-7}$	$1.04 \cdot 10^{-6}$	$9.80 \cdot 10^{-7}$	1.06
400	$3.30 \cdot 10^{-12}$	$1.17 \cdot 10^{-7}$	$3.29 \cdot 10^{-12}$	$1.17 \cdot 10^{-7}$	$2.34 \cdot 10^{-7}$	$2.23 \cdot 10^{-7}$	1.05
800	$8.32 \cdot 10^{-13}$	$2.82 \cdot 10^{-8}$	$8.32 \cdot 10^{-13}$	$2.80 \cdot 10^{-8}$	$5.62 \cdot 10^{-8}$	$5.07 \cdot 10^{-8}$	1.11

Table 4.4: Residuals and effectivities for the solid subdomain functional in case of uniform time-stepping $N^f = N^s = N$.

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$4.13 \cdot 10^{-10}$	$2.61 \cdot 10^{-6}$	$1.91 \cdot 10^{-9}$	$2.68 \cdot 10^{-6}$	$5.29 \cdot 10^{-6}$	$4.68 \cdot 10^{-6}$	1.13
100	$8.69 \cdot 10^{-11}$	$5.20 \cdot 10^{-7}$	$-3.72 \cdot 10^{-11}$	$5.23 \cdot 10^{-7}$	$1.04 \cdot 10^{-6}$	$9.54 \cdot 10^{-7}$	1.09
200	$1.80 \cdot 10^{-11}$	$1.17 \cdot 10^{-7}$	$1.40 \cdot 10^{-12}$	$1.17 \cdot 10^{-7}$	$2.34 \cdot 10^{-7}$	$2.16 \cdot 10^{-7}$	1.08
400	$3.94 \cdot 10^{-12}$	$2.82 \cdot 10^{-8}$	$1.87 \cdot 10^{-12}$	$2.80 \cdot 10^{-8}$	$5.62 \cdot 10^{-8}$	$4.90 \cdot 10^{-8}$	1.15

Table 4.5: Residuals and effectivities for the solid subdomain functional in case of uniform refinement in the solid domain $N^f = N$ and $N^s = 2N$.

after 4 steps of refinement. In each separate graph, the middle line corresponds to the macro time mesh, the top line represents the fluid micro mesh and the bottom line the solid micro mesh. As expected, this example leads to sub-cycling within the fluid domain and there is no refinement in the solid time mesh. Refinement in the fluid domain is fully uniform.

4.4.1.2 Second Test Case

As our second test problem, we choose a set-up somewhat symmetric to the previous one. This time, we take a functional which is concentrated only on the right half of the solid domain $\Omega_2^s = (2, 4) \times (-1, 0)$ with Configuration 2 as the right hand side in the primal problem. Here, instead of the velocity, we look at a gradient of displacement scaled by parameter λ and therefore the functional is given by

$$Q^f(\vec{\mathbf{U}}) = 0, \quad Q^s(\vec{\mathbf{U}}) = \int_0^T \lambda \|\nabla u_2^s\|_s^2 \, \mathrm{d}t$$

and its Gâteaux derivative is equal to

$$(Q^s)'_{\vec{\mathbf{U}}}(\mathbf{\Xi}^s) = \int_0^T 2\lambda \left(\nabla u_2^s, \nabla \xi_2^s \right)_s \mathrm{d}t,$$

N	N^f	N^s	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	50	88	$3.77 \cdot 10^{-10}$	$6.57 \cdot 10^{-6}$	$6.72 \cdot 10^{-8}$	$6.91 \cdot 10^{-6}$	$1.35 \cdot 10^{-5}$	$1.06 \cdot 10^{-5}$	1.28
50	50	166	$5.17 \cdot 10^{-10}$	$1.35 \cdot 10^{-6}$	$7.16 \cdot 10^{-8}$	$1.38 \cdot 10^{-6}$	$2.80 \cdot 10^{-6}$	$2.52 \cdot 10^{-6}$	1.11
50	50	286	$5.80 \cdot 10^{-10}$	$4.54 \cdot 10^{-7}$	$4.16 \cdot 10^{-8}$	$4.56 \cdot 10^{-7}$	$9.52 \cdot 10^{-7}$	$7.34 \cdot 10^{-7}$	1.30
54	54	400	$5.70 \cdot 10^{-10}$	$1.19 \cdot 10^{-7}$	$4.12 \cdot 10^{-8}$	$1.19 \cdot 10^{-7}$	$2.81 \cdot 10^{-7}$	$1.10 \cdot 10^{-7}$	2.55

Table 4.6: Residuals and effectivities for the solid subdomain functional in case of adaptive time-stepping.

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$3.74 \cdot 10^{-8}$	$2.26 \cdot 10^{-6}$	$8.73 \cdot 10^{-8}$	$1.03 \cdot 10^{-5}$	$1.27 \cdot 10^{-5}$	$2.58 \cdot 10^{-5}$	0.49
100	$9.87 \cdot 10^{-9}$	$2.30 \cdot 10^{-6}$	$7.93 \cdot 10^{-9}$	$2.43 \cdot 10^{-6}$	$4.76 \cdot 10^{-6}$	$4.51 \cdot 10^{-6}$	1.06
200	$2.52 \cdot 10^{-9}$	$4.53 \cdot 10^{-7}$	$2.38 \cdot 10^{-9}$	$4.70 \cdot 10^{-7}$	$9.27 \cdot 10^{-7}$	$9.23 \cdot 10^{-7}$	1.00
400	$6.36 \cdot 10^{-10}$	$1.08 \cdot 10^{-7}$	$6.28 \cdot 10^{-10}$	$1.11 \cdot 10^{-7}$	$2.20 \cdot 10^{-7}$	$2.12 \cdot 10^{-7}$	1.04
800	$1.60 \cdot 10^{-10}$	$2.68 \cdot 10^{-8}$	$1.59 \cdot 10^{-10}$	$2.75 \cdot 10^{-8}$	$5.47 \cdot 10^{-8}$	$4.87 \cdot 10^{-8}$	1.12

Table 4.7: Residuals and effectivities for the combined functional in case of uniform time-stepping $N^f = N^s = N$.

where again $u_2^s, \xi_2^s := u^s\big|_{\Omega_2^s}, \xi^s\big|_{\Omega_2^s}$.

Similarly, as in the previous example, we start with simulations that do not contain any micro time-stepping ($N^f = N^s = N$). The same sequence of uniform time-meshes is used. The extrapolation in time of the results from the three last simulations yields the exact value $\widetilde{Q} = 3.458826 \cdot 10^{-4}$ and a surprisingly high convergence rate of $q = 2.135897$. Although the effectivities are still highly satisfactory, one can observe a slight deterioration in comparison with the previous test case. That seems to be analogous to our numerical tests of decoupling strategies in Section 3.4.1, where we also discovered that Configuration 2 of the right hand side requires more iterations to obtain the prescribed tolerance. That might be explained by the more challenging hyperbolic character of the wave equation compared to the parabolic nature of the heat equation. We can also see that the extrapolated errors and consequently the residuals are significantly higher than in the first test case. Of course, partially it can be explained by the higher value of the functional itself. However, even after taking that into account, we can still see that the error on the coarsest mesh is only one order of magnitude lower than the value of the functional. Also here we can observe that the primal and adjoint residuals have almost identical values. The residuals are highly concentrated in the solid domain which justifies why in the following simulation we decided to introduce micro time-stepping in this subdomain.

In Table 4.5 we collect the numbers for a simulation where the time-steps in the solid domain were once uniformly refined , that is $N^f = N$ and $N^s = 2N$. This refinement does
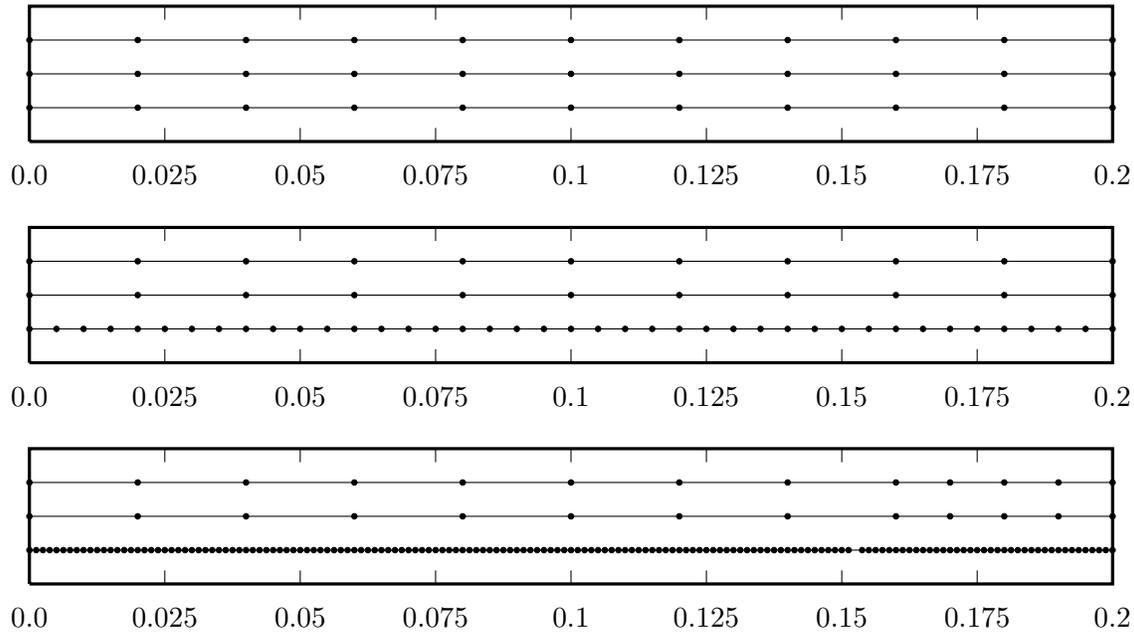
Figure 4.6: This figure shows the adaptive time mesh for the solid functional.

not have a significant impact on the effectivity values and yet it can considerably reduce the extrapolated error $\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$. Also here, partial refinement yields similar results as global uniform refinement. The numbers encourage even more aggressive refinement in the solid domain.

In Table 4.6 we present the outcomes of a fully adaptive procedure starting from the uniform simulation on the coarsest mesh for $N = 50$. In comparison to a similar simulation for the fluid functional, we can make a few observations. First, the refinement in the solid domain is indeed much more aggressive with a total number of 400 time-steps at the last level of refinement. In the final simulation, there is also some refinement in the fluid domain. We can however observe a worsening of effectivities, especially in the last simulation. That being said, the reduction of extrapolation error is still impressive. After these simulations with the total number of 454 time-steps the extrapolated error is equal to $1.10 \cdot 10^{-7}$. For comparison, in Table 4.4 for 800 total number time-steps we have $\widetilde{Q} - Q(\vec{\mathbf{U}}_k) = 2.34 \cdot 10^{-7}$.

In Figure 4.6 we give a glimpse into the adaptive refinement procedure for the solid functional. The same time interval $[0.0, 0.2]$ is examined. Unsurprisingly, the vast majority of refinements happen in the solid domain. Compared to Figure 4.6, refinement is much more aggressive. On the last two levels, the fluid problem also requires some local refinement. Whenever possible, we avoid excessive sub-cycling by refining the macro mesh as described in Section 4.2.5.

N	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$3.74 \cdot 10^{-8}$	$2.51 \cdot 10^{-6}$	$8.13 \cdot 10^{-8}$	$2.63 \cdot 10^{-6}$	$5.26 \cdot 10^{-6}$	$5.10 \cdot 10^{-6}$	1.03
100	$9.84 \cdot 10^{-9}$	$4.75 \cdot 10^{-7}$	$7.87 \cdot 10^{-9}$	$4.91 \cdot 10^{-7}$	$9.84 \cdot 10^{-7}$	$1.08 \cdot 10^{-6}$	0.91
200	$2.52 \cdot 10^{-9}$	$1.11 \cdot 10^{-7}$	$2.39 \cdot 10^{-9}$	$1.14 \cdot 10^{-7}$	$2.30 \cdot 10^{-7}$	$2.52 \cdot 10^{-7}$	0.91
400	$6.36 \cdot 10^{-10}$	$2.72 \cdot 10^{-8}$	$6.31 \cdot 10^{-10}$	$2.79 \cdot 10^{-8}$	$5.64 \cdot 10^{-8}$	$5.87 \cdot 10^{-8}$	0.96

Table 4.8: Residuals and effectivities for the combined functional in case of uniform refinement in the solid domain $N^f = N$ and $N^s = 2N$.

N	N^f	N^s	η_k^f	η_k^s	$\eta_k^{f,*}$	$\eta_k^{s,*}$	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	50	88	$3.77 \cdot 10^{-8}$	$6.21 \cdot 10^{-6}$	$1.57 \cdot 10^{-7}$	$6.63 \cdot 10^{-6}$	$1.30 \cdot 10^{-5}$	$1.08 \cdot 10^{-5}$	1.21
50	50	166	$3.73 \cdot 10^{-8}$	$1.28 \cdot 10^{-6}$	$1.60 \cdot 10^{-7}$	$1.33 \cdot 10^{-6}$	$2.80 \cdot 10^{-6}$	$3.15 \cdot 10^{-6}$	0.89
52	52	280	$3.57 \cdot 10^{-8}$	$4.74 \cdot 10^{-7}$	$5.96 \cdot 10^{-8}$	$4.85 \cdot 10^{-7}$	$1.054 \cdot 10^{-6}$	$1.59 \cdot 10^{-6}$	0.66
56	56	396	$3.36 \cdot 10^{-8}$	$1.32 \cdot 10^{-7}$	$5.44 \cdot 10^{-8}$	$1.36 \cdot 10^{-7}$	$3.55 \cdot 10^{-7}$	$8.66 \cdot 10^{-7}$	0.41

Table 4.9: Residuals and effectivities for the combined functional in case of adaptive time-stepping.

4.4.1.3   Third Test Case

In our third and final test case for this model problem, we examine a functional which is a sum of the two previously studied functionals

$$Q^f(\vec{\mathbf{U}}) := \int_0^T \nu \|\nabla v_2^f\|_f^2 \, \mathrm{d}t, \quad Q^s(\vec{\mathbf{U}}) = \int_0^T \lambda \|\nabla u_2^s\|_s^2 \, \mathrm{d}t.$$

In this case, we take Configuration 3 of the source term, which in turn is a sum of the right hand sides in Configuration 1 and 2.

In Table 4.7 we look at a simulation consisting of a series of uniform meshes. Using the last three simulations, we obtain an extrapolated exact value of the functional $\widetilde{Q} = 4.206957 \cdot 10^{-4}$ with a convergence rate $q = 2.122432$. The resulting functional value is close to the sum of the two previously examined functionals. The effectivities in this example are again excellent. For $N = 200$ we were even able to reach exactly 1.00 (up to rounding to two decimal places). Similarly as in the case of the functional, the partial fluid and solid residuals are close to sums of the corresponding terms from the two previous examples. Since the contribution from the solid problem is dominant, we will introduce micro time-stepping there.

In Table 4.8 we gather numbers for a uniformly refined case with micro time-stepping in the solid domain. The effectivities worsen only slightly. Here one can also observe a comparable reduction of the overall extrapolated error as in the case of a full uniform refinement in both domains. Further, by comparing Table 4.8 and 4.5 we can see similar performance. Indeed,
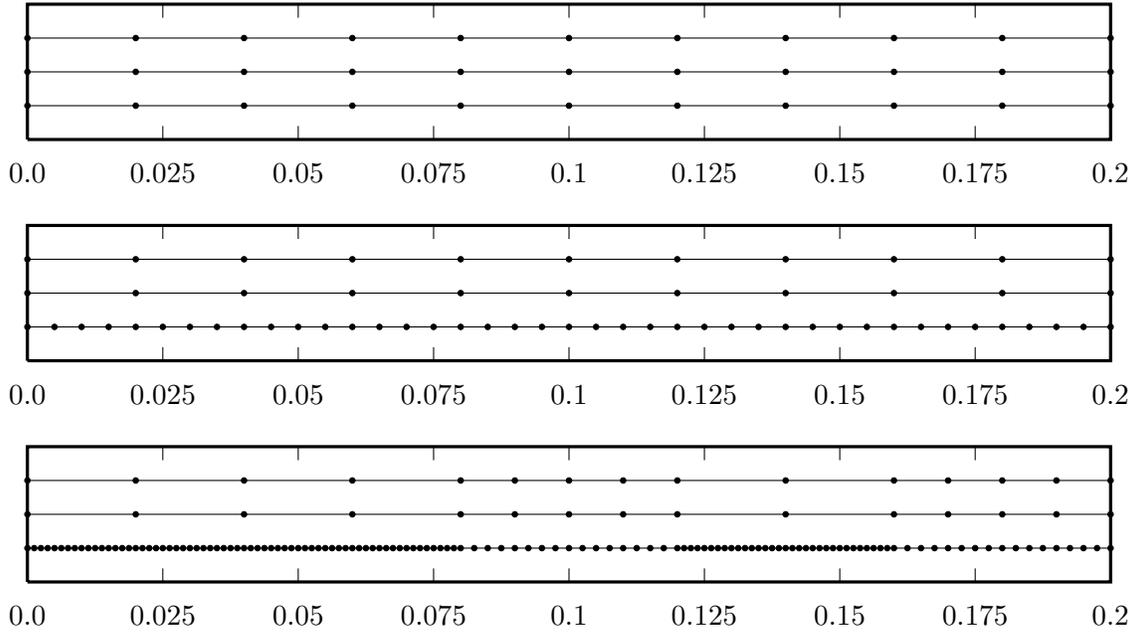
Figure 4.7: This figure shows the adaptive time mesh for the combined functional.

after the last refinement, the extrapolated error is equal to $5.87 \cdot 10^{-8}$ compared to a similar value of $5.62 \cdot 10^{-8}$ in Table 4.5. Although the number here is a bit higher, the difference is not significant.

In Table 4.9 we look at a fully adaptive case. Here we can observe a noticeable reduction in the accuracy of our error estimator. Although the drop in effectivity is similar to the fully adaptive simulations of the second test case, this configuration also exhibits a lesser reduction of the extrapolated error. Even though the distribution of refined time-steps is similar with the total number of time-steps in the fluid domain equal to 56, and in the solid domain to 396, the extrapolated error is reduced to $8.66 \cdot 10^{-7}$ compared to $1.10 \cdot 10^{-7}$ in Table 4.6. That suggests that our adaptive algorithm is the most effective when we consider functionals concentrated on only one subproblem.

Finally, in Figure 4.7 we present the adaptive time mesh. Again, the same time interval $[0.0, 0.2]$ is considered. Also here, there is extensive refinement in the solid domain. Interestingly, in spots where the solid time mesh is coarser, we can see some refinement in the fluid domain as well. To avoid unnecessary micro time-stepping, refinement in the fluid domain also leads to refinement of the macro time mesh.

N	η_k^f	η_k^s	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$2.98 \cdot 10^{-3}$	$1.79 \cdot 10^{-2}$	$2.09 \cdot 10^{-2}$	$3.96 \cdot 10^{-2}$	0.53
100	$6.76 \cdot 10^{-4}$	$1.90 \cdot 10^{-3}$	$2.57 \cdot 10^{-3}$	$9.38 \cdot 10^{-3}$	0.27
200	$1.68 \cdot 10^{-4}$	$4.53 \cdot 10^{-4}$	$6.22 \cdot 10^{-4}$	$2.15 \cdot 10^{-3}$	0.29
400	$4.38 \cdot 10^{-5}$	$1.16 \cdot 10^{-4}$	$1.60 \cdot 10^{-4}$	$4.93 \cdot 10^{-4}$	0.33

Table 4.10: Residuals and effectivities for the FSI2 benchmark in case of uniform time-stepping $N^f = N^s = N$.

N	η_k^f	η_k^s	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$2.97 \cdot 10^{-2}$	$1.12 \cdot 10^{-1}$	$1.42 \cdot 10^{-1}$	$9.87 \cdot 10^{-2}$	1.44
100	$9.36 \cdot 10^{-3}$	$8.87 \cdot 10^{-3}$	$1.82 \cdot 10^{-2}$	$2.76 \cdot 10^{-2}$	0.66
200	$2.46 \cdot 10^{-3}$	$2.05 \cdot 10^{-3}$	$4.51 \cdot 10^{-3}$	$7.79 \cdot 10^{-3}$	0.58
400	$6.30 \cdot 10^{-4}$	$5.37 \cdot 10^{-4}$	$1.17 \cdot 10^{-3}$	$2.20 \cdot 10^{-3}$	0.53

Table 4.11: Residuals and effectivities for the FSI3 benchmark in case of uniform time-stepping $N^f = N^s = N$.

4.4.2 FSI Problem

In the following numerical tests, we are going to consider the FSI benchmark. For this example, a shorter time interval $I = [0, 0.5]$ is chosen. Since the movement of the elastic structure is the most interesting and challenging part of these simulations, we examine a goal functional consisting of solid displacement over the whole solid domain. To achieve a better regularity, we look at its norm instead of its gradient and therefore

$$Q^f(\vec{\mathbf{U}}) := 0, \quad Q^s(\vec{\mathbf{U}}) := \int_0^T \lambda^s \|\mathbf{u}^s\|_s^2 \, \mathrm{d}t.$$

The Gâteaux derivative is given by

$$(Q^s)'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi}^s) = \int_0^T 2\lambda^s \left(\mathbf{u}^s, \boldsymbol{\xi}^s\right)_s \mathrm{d}t.$$

Because of the oscillatory character of this problem, we decided to slightly modify how the residuals are computed. Let us recall the formula which we used previously

$$\eta_k = \sum_{n=1}^{N} \sum_{m=1}^{N_n^f} \left(\eta_{n,m}^f + \eta_{n,m}^{f,*}\right) + \sum_{n=1}^{N} \sum_{m=1}^{N_n^s} \left(\eta_{n,m}^s + \eta_{n,m}^{s,*}\right).$$

N	η_k^s	η_k^T	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$1.57 \cdot 10^{-3}$	$6.08 \cdot 10^{-6}$	$1.57 \cdot 10^{-3}$	$1.15 \cdot 10^{-2}$	0.14
100	$5.08 \cdot 10^{-4}$	$1.52 \cdot 10^{-6}$	$5.09 \cdot 10^{-4}$	$2.76 \cdot 10^{-3}$	0.18
200	$1.23 \cdot 10^{-4}$	$3.80 \cdot 10^{-7}$	$1.23 \cdot 10^{-4}$	$6.90 \cdot 10^{-4}$	0.18
400	$3.05 \cdot 10^{-5}$	$9.51 \cdot 10^{-8}$	$3.06 \cdot 10^{-5}$	$1.72 \cdot 10^{-4}$	0.18
800	$7.60 \cdot 10^{-6}$	$2.38 \cdot 10^{-8}$	$7.62 \cdot 10^{-6}$	$4.31 \cdot 10^{-5}$	0.18

Table 4.12: Residuals and effectivities for the thermoelasticity model problem in case of uniform time-stepping $N^s = N^T = N$.

N	η_k^s	η_k^T	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	$5.08 \cdot 10^{-4}$	$6.08 \cdot 10^{-6}$	$5.14 \cdot 10^{-4}$	$2.75 \cdot 10^{-3}$	0.19
100	$1.24 \cdot 10^{-4}$	$1.52 \cdot 10^{-6}$	$1.25 \cdot 10^{-4}$	$6.89 \cdot 10^{-4}$	0.18
200	$3.05 \cdot 10^{-5}$	$3.81 \cdot 10^{-7}$	$3.09 \cdot 10^{-5}$	$1.72 \cdot 10^{-4}$	0.18
400	$7.60 \cdot 10^{-6}$	$9.52 \cdot 10^{-8}$	$7.69 \cdot 10^{-6}$	$4.31 \cdot 10^{-5}$	0.18

Table 4.13: Residuals and effectivities for the thermoelasticity model problem in case of uniform refinement in the structure problem $N^s = 2N$ and $N^T = N$.

For the FSI example, we will absolute values inside the sums and instead consider

$$\eta_k = \sum_{n=1}^{N} \sum_{m=1}^{N_n^f} \left| \eta_{n,m}^f + \eta_{n,m}^{f,*} \right| + \sum_{n=1}^{N} \sum_{m=1}^{N_n^s} \left| \eta_{n,m}^s + \eta_{n,m}^{s,*} \right|. \tag{4.11}$$

This new formulation can be justified by the fact that a similar formula was already present in the adaptive time-stepping criterion (4.9).

Due to the added mass effect, we, unfortunately, were not able to introduce micro time-stepping in the solid domain. Because of that, here we present the results only for the fully uniform time interval partitioning. We would also like to recall that due to nonlinearities in the equations governing this example, the time discretization does not fully correspond to the formulation given by the space-time approach. In addition, the resulting quadrature errors are of the same order as the error estimator itself. Since these additional errors were not accounted for, we should not expect as good effectivities as in the previous linear test model. Given these reasons, as we will see, the numbers achieved by us are highly satisfactory.

In Table 4.10 we show residuals and effetcivities for the FSI2 benchmark test. We take a series of four fully uniform time meshes, where the time-step sizes ranging from $k_n = 0.01$ up to $k_n = 0.00125$ were considered. Again, using extrapolation in time we obtained the exact functional value $\widetilde{Q} = 5.627679$ and a high convergence rate of $q = 2.125132$. This is quite surprising knowing that some of the terms were discretized using the implicit Euler scheme,

N	N^s	N^T	η_k^s	η_k^T	η_k	$\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$	eff_k
50	78	50	$4.47 \cdot 10^{-3}$	$6.40 \cdot 10^{-6}$	$4.48 \cdot 10^{-3}$	$5.23 \cdot 10^{-3}$	0.86
50	114	50	$4.74 \cdot 10^{-3}$	$8.96 \cdot 10^{-6}$	$4.75 \cdot 10^{-3}$	$1.98 \cdot 10^{-3}$	2.39
50	142	50	$1.45 \cdot 10^{-3}$	$8.29 \cdot 10^{-6}$	$1.46 \cdot 10^{-3}$	$1.36 \cdot 10^{-3}$	1.08
50	180	50	$1.12 \cdot 10^{-3}$	$7.36 \cdot 10^{-6}$	$1.13 \cdot 10^{-3}$	$1.20 \cdot 10^{-3}$	0.94
50	226	50	$2.80 \cdot 10^{-3}$	$9.86 \cdot 10^{-6}$	$2.81 \cdot 10^{-3}$	$5.79 \cdot 10^{-4}$	4.86

Table 4.14: Residuals and effectivities for the thermoelasticity model problem in case of adaptive time-stepping.

which has only a linear convergence rate. The effectivities are not as close to one as in the previous example. However, they consistently reach similar values which in turn means that the residuals and the extrapolated errors have a similar convergence rate. We can also see in this example the dominance of contributions coming from the solid problem over the ones from the fluid problem. Noteworthy, however, this difference is smaller than in the results of the previous test model.

In Table 4.11 we gather our findings for the FSI3 benchmark problem. We obtained $\widetilde{Q} = 2.179523$ and $q = 1.824982$. The effectivities are higher than in the previous problem. Except for the coarsest mesh, they consistently reach similar values. Having said that, one can observe a modest deterioration of the effectivities here which indicates a small mismatch in the convergence rates of the extrapolated error and the residual. Further, the extrapolated errors have higher values than in the previous FSI2 test case. That can be easily explained by the shorter period of the displacement oscillations. For visualization of this fact, see Figure 1.5. Here we can also see the dominance of the solid residuals but the difference is even smaller than in the previous FSI2 problem.

4.4.3 Thermoelasticity

We continue our analysis and proceed to the thermoelasticity problem. Due to the oscillatory nature of this problem, we consider a very short time interval $I = [0, 0.1]$. For this example, only one functional is examined. It takes into account contributions from both the structure as well as the temperature variables and is defined by a squared norm of the stress tensor scaled by a Lamé parameter. As a result, the functional is given by

$$Q(\vec{\mathbf{U}}, T) := \int_0^T \lambda \|\boldsymbol{\epsilon}\|^2 \, \mathrm{d}t.$$

Since this functional depends on variables from both of the subproblems, we need to consider both of the partial Gâteaux derivatives

$$Q'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi}) = \int_0^T 2\lambda \boldsymbol{\epsilon} \boldsymbol{\epsilon}'_{\vec{\mathbf{U}}}(\boldsymbol{\Xi}) \, \mathrm{d}t, \qquad Q'_T(\Theta) = \int_0^T \lambda'_T(\Theta) \|\boldsymbol{\epsilon}\|^2 \, \mathrm{d}t.$$
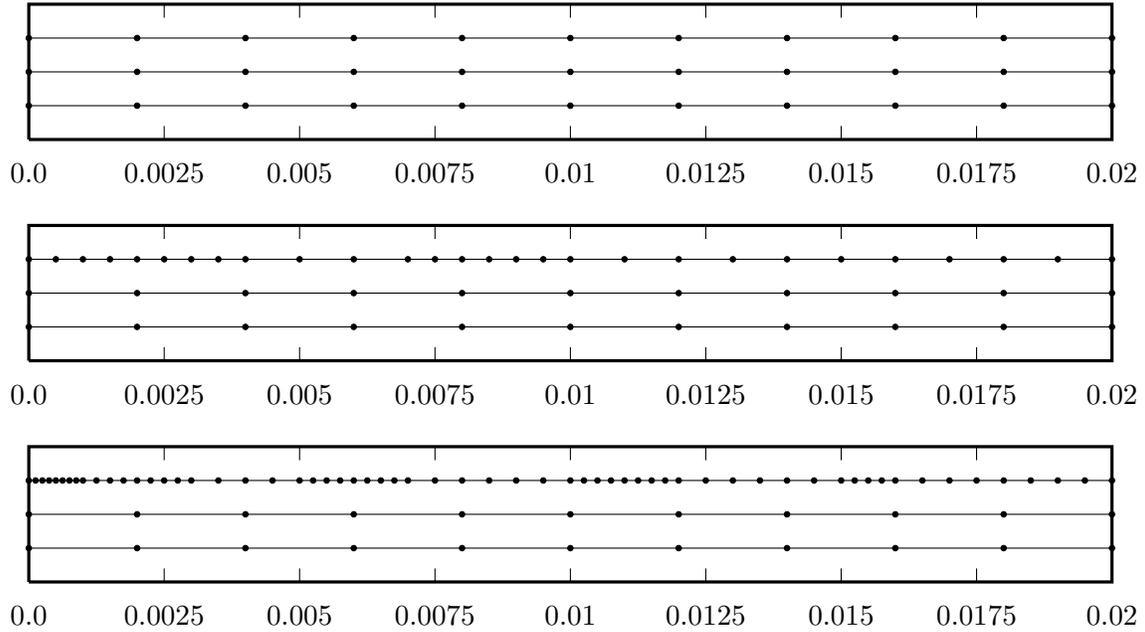
Figure 4.8: This figure shows the adaptive time mesh for the thermoelasticity example.

Let us recall

$$\epsilon'_{\vec{\mathbf{U}}}(\mathbf{\Xi}) = \frac{1}{2}\left(\nabla\boldsymbol{\zeta} + (\nabla\boldsymbol{\zeta})^T\right), \qquad \lambda'_T(\Theta) = \frac{-\nu E_0 \beta S}{(1+\nu)(1-2\nu)}.$$

Also here, small modifications will be implemented in the computation of partial residuals. We will use the same algorithm as in the previous FSI problem (4.11) with a difference that we will consider contributions from the structure and thermal problems instead of the fluid and structure ones.

In Table 4.12 we explore the results for the uniform time-stepping. In each simulation, the computations were performed over the same interval I. On the coarsest mesh, we divided it into $N = 50$ time-steps, where for each of them $k_n = 0.002$. We then introduced a sequence of binary refinements until $N = 800$ and $k_n = 0.000125$. Using the extrapolation in time procedure with the last three most accurate results, we obtained $\widetilde{Q} = 13.108507$ and $q = 2.000106$. Also here, the convergence rate here is almost perfectly quadratic. Although the effectivities achieved here are not as good as in the first linear example, they are very stable and except for the coarsest level, stay at $\text{eff}_k = 0.18$. That means that we managed to achieve an excellent alignment of the convergence rates between the estimator and the extrapolated error.

Following the same path as in the previous examples, in Table 4.13 we show computation results for a case, where an additional level of uniform refinement was introduced in variables corresponding to the structure problem. We decided to introduce micro time-stepping there

based on values of partial residuals which are not present in this thesis. Similarly as in the first test problem, introducing partial refinement is just as effective in reducing the extrapolated error as uniform refinement. For example, the extrapolated error $\widetilde{Q} - Q(\vec{\mathbf{U}}_k)$ in Table 4.13 for $N = 400$ and in Table 4.12 for $N = 800$ are equal. A similar phenomenon happens on coarser levels as well. The effectivities have very stable values and stay close to $\text{eff}_k = 0.18$.

In Table 4.14 we finally present the fully adaptive case. As our initial set-up, we took the coarsest solution in Table 4.12. Unsurprisingly, given the results in Table 4.13, only the variables corresponding to the structure subproblem are refined. The effectivities values are more volatile with the worst value on the finest mesh $\text{eff}_k = 4.86$. That being said, this deterioration is acceptable and typical of adaptive refinement. More importantly, we achieved a significant reduction in the extrapolation error. On the finest level, we have $N^s = 226$, $N^T = 50$, and $\widetilde{Q} - Q(\vec{\mathbf{U}}_k) = 5.79 \cdot 10^{-4}$. The closest value from Table 4.13 is for $N^s = 200$, and $N^T = 100$ with $\widetilde{Q} - Q(\vec{\mathbf{U}}_k) = 5.79 \cdot 10^{-4}$. That indicates a significant improvement given that the performance of the partial refinement is already much better than fully uniform time-stepping. Such extensive refinement only in the structure problem indicates a limited influence of the Lamé parameter λ on the values of the goal functional Q.

We show the final adaptive time mesh in Figure 4.8. For the thermoelasticity example, a shorter time interval $[0.0, 0.02]$ was considered. We can notice intense refinement in the structure problem. Compared to the previously shown adaptive meshes, this one is the least regular with distinct coarser and finer regions. Particularly the onset region is highly refined.

Part II

Theoretical Error and Stability Estimates

Chapter 5

A Priori Estimates for Interface Coupled Problems

In the previous part of this thesis, we presented a multirate adaptive time-stepping algorithm equipped with an a posteriori error estimator. However, despite its flexibility, robustness, and other attractive attributes, this estimator did not yield any rigorous error bounds. We will address this issue in this chapter. Since a strict analysis of physically meaningful models is exceedingly difficult, we will instead choose a set of simplified linear problems.

After a brief general introduction to interface coupled problems, we will look at a simple system of ordinary differential equations. Then we will proceed to two interface coupled systems of partial differential equations. The first one will consist of two heat equations scaled by different diffusivity constants. In the second one, we will additionally impose an incompressibility condition resulting in two Stokes systems. All of them will be discretized using the implicit Euler time-stepping scheme. The scheme will allow for independent time-step sizes in each of the subproblems. The transfer of values between the non-matching time meshes will be handled using an average operator. We will prove a priori estimates with mostly optimal linear convergence rates. Only the convergence rate of pressure in the Stokes problem will be reduced by half an order. To prove this last estimate, a modified version of the inf-sup condition will be used. The estimates for the fully discrete solutions of the heat and Stokes equations will be proved in a modified energy norm that takes into account a Nitsche jump term on the interface. We are in the process of publishing excerpts from this chapter.

5.1 Interface Coupled Problems

Resolving the interface for coupled problems is a crucial step in deriving successful discretization techniques. As we briefly discussed in the introduction to this thesis, there exist two main approaches to solving such systems. In a monolithic approach, one solves the whole coupled system using one solver. That allows defining trial and test spaces globally. In a continuous setting, that means that trial and test functions are continuous across the interface.

From the continuity of the former, one directly obtains a global continuity of the solution. The latter condition implies continuity of a stress tensor in the normal direction. This way, we can embed the coupling conditions in the proper choice of trial and test spaces. That results in a stable scheme. Unfortunately, the lack of flexibility prohibits us from using different time-step sizes for each of the problems and therefore this approach is not available to us.

In a partitioned approach, each of the subproblems is considered separately and has its own set of test and trial spaces. The continuity of neither trial nor test functions on the interface is automatically fulfilled. In this set of methods, we can distinguish between two different techniques to enforce the coupling conditions. The first solution is to enrich the trial space by introducing additional basis functions to the approximation. This approach is especially useful when one considers unfitted space meshes, where the interface does not have to align with the facets of mesh cells. It is called either *extended* or *generalized finite element method*. The enrichment is based on the concept of partition of unity. It is often combined with level set functions for tracking the interface. In this case, the absolute value of the level set function is used as the enrichment. If we enrich the solution space with functions with compact support, the resulting system is sparse, making the computations cost effective. A review on this topic can be found in [51]. A similar idea stands behind the *immersed finite element method*. There, not only the internal interface but also outer boundaries do not have align with the space mesh structure. The interface is resolved using modified function spaces. The method was first developed in [52, 53], then extended to interface problems [54, 55]. In the *weak Galerkin finite element method* [56], functions from a solution space are allowed to have different values in the interior and on the boundary of elements. This method proved to be especially advantageous in the case of complex interfaces.

Alternatively, one can enforce conditions on the interface by introducing additional terms to a variational formulation. Most of these techniques are based on the Nitsche method [26]. According to this method, one introduces an additional stabilization term scaled by h^{-1} to enforce the continuity of the solution across the interface. The term h^{-1} is used to balance the normal derivative on the interface. It provides the necessary power of h to use an inverse discrete inequality of the form

$$h^{\frac{1}{2}}||\partial_{\vec{\mathbf{n}}}\mathbf{u}^{k,h}||_\Gamma \leqslant ||\nabla\mathbf{u}^{k,h}||_\Omega.$$

This is the approach we used until now and will continue to use it in this chapter. Nitsche method is used in a wide variety of contexts. Its generality allows for different polynomial degrees, space meshes, and physical models on each side of the interface. In [57] the authors present multiple applications. A discontinuous elasticity model has been analyzed in [58]. A study on a full FSI model in ALE coordinates with Nitsche boundary is shown in [59]. This method is also consistent and stable with optimal convergence rate in a mesh-dependent energy norm and L^2 norm. A priori and a posteriori estimates are proved in [60]. Convergence of an adaptive algorithm for elliptic interface problems is presented in [61].

5.2 Coupling of Ordinary Differential Equations

As our first example we chose a standard system of two ordinary equations where we look for a solution $\mathbf{u} = (u_1, u_2)^T : I \times I \to \mathbb{R} \times \mathbb{R}$ with $\mathbf{u} \in L^2(\bar{I})^2$ to a system

$$d_t u_1(t) = f_1\big(t, u_1(t), u_2(t)\big), \quad d_t u_2(t) = f_2\big(t, u_1(t), u_2(t)\big), \quad \mathbf{u}(0) = \mathbf{0}. \tag{5.1}$$

We assume that the function $\mathbf{f} = (f_1, f_2)$ is Lipschitz continuous. Next, Follows a semi-discrete formulation. The two problems are coupled across the macro mesh over the time interval $I = [0, T]$

$$0 = t^0 < t^1 ... < t^N = T, \ \ k^n = t^n - t^{n-1}, \ \ I^n = (t^{n-1}, t^n].$$

Every time subinterval I^n has its own time interval partitioning corresponding to each of the subproblems

$$t^{n-1} = t_1^{n,0} < t_1^{n,1} ... < t_1^{n,N_1^n} = t^n, \quad t^{n-1} = t_2^{n,0} < t_2^{n,1} ... < t_2^{n,N_2^n} = t^n.$$

On top of that,

$$k_1^{n,m} := t_1^{n,m} - t_1^{n,m-1}, \ \ I_1^{n,m} := (t_1^{n,m-1}, t_1^{n,m}],$$
$$k_2^{n,m} := t_2^{n,m} - t_2^{n,m-1}, \ \ I_2^{n,m} := (t_2^{n,m-1}, t_2^{n,m}]$$

as well as

$$k_1 := \max_{n,m} k_1^{n,m}, \quad k_2 := \max_{n,m} k_2^{n,m}, \quad k := \max\{k_1, k_2\}.$$

An example of this kind of time mesh is shown in Figure 2.1.

We assume that in this time partitioning, micro time-steps are introduced only when necessary, see Figures 2.1 and 2.2 in Chapter 2. Besides, we assume that these time meshes are a result of an adaptive time-stepping procedure where time-steps are refined only in the middle. Based on these two assumptions for every macro time-step I^n we either have $N_1^n = 1$ or $N_2^n = 1$. The space of discrete solutions consists of piecewise constant functions defined over each of the meshes

$$X_j^k = \big\{ u \in L^2(\bar{I}) \big| \ u|_{I_j^{n,m}} \in \mathbb{R} \text{ for all } I_j^{n,m} \subset I \text{ and } u(0) = 0 \big\}, \quad j = 1, 2 \tag{5.2}$$

We take discrete solutions $\mathbf{u}^k \in X^k$, where $\mathbf{u}^k = (u_1^k, u_2^k)^T$ and $X^k = X_1^k \times X_2^k$. To further specify the time-stepping scheme, we define the operator $\mathbf{i}^k = (i_1^k, i_2^k)^T$ with $\mathbf{i}^k : L^2(\bar{I})^2 \to \mathbb{R}^2$ by

$$i_j^k u \big|_{I_j^{n,m}} := u(t_j^{n,m}), \quad j = 1, 2. \tag{5.3}$$

The initial condition is given by $(\mathbf{i}^k \mathbf{u})(0) = \mathbf{u}(0)$. This choice of projection operators indicates the implicit Euler method. We follow this route and introduce a finite difference quotient d_t^k typical for this time-stepping scheme

$$d_t^k u_j^k \big|_{I_j^{n,m}} = \frac{u_j^k(t_j^{n,m}) - u_j^k(t_j^{n,m-1})}{k_j^{n,m}}, \quad j = 1, 2. \tag{5.4}$$

Since we are interested in coupled problems, we also need an apparatus to deal with the transfer of solutions between the non-matching time meshes. To resolve this issue, we additionally introduce an operator $\mathbf{I}^k = (I_1^k, I_2^k)^T$ with $\mathbf{I}^k : L^2(\bar{I})^2 \to \mathbb{R}^2$ given by an average over each partitioning, where on every interval $I_1^{n,m}$ and $I_2^{n,m}$ it holds

$$I_j^k u\big|_{I_j^{n,m}} := \frac{1}{k_j^{n,m}} \int_{I_j^{n,m}} u \, \mathrm{d}t, \quad j = 1, 2. \tag{5.5}$$

A similar operator $\bar{I}^k$ will be used which is defined accordingly on the coarse time mesh consisting of the macro time-steps

$$\bar{I}^k u\big|_{I^n} := \frac{1}{k^n} \int_{I^n} u \, \mathrm{d}t. \tag{5.6}$$

At the initial point, we impose $(\mathbf{I}^k \mathbf{u})(0) = \mathbf{u}(0)$ and $(\bar{I}^k \mathbf{u})(0) = \mathbf{u}(0)$. Further information is given in Figure 5.1. The key property of the operator $\mathbf{I}^k$ is that its error has average zero over each macro time-step

$$\int_{I^n} \left(\mathbf{u} - \mathbf{I}^k \mathbf{u} \right) \mathrm{d}t = 0.$$

Moreover, for any $u_1^k \in X_1^k$ and $u_2^k \in X_2^k$

$$\int_{I^n} \left(u_1^k - I_2^k u_1^k \right) \mathrm{d}t = \int_{I^n} \left(u_2^k - I_1^k u_2^k \right) \mathrm{d}t = 0. \tag{5.7}$$

This identity directly follows from the hierarchical structure in the discretization, where we know that nodes t^n and t^{n-1} belong to both of the discretizations.

We would like to reiterate the difference between operators $\mathbf{i}^k$ and $\mathbf{I}^k$. The former is our primary operator used in the implicit Euler time-stepping scheme and will be a part of a projection error that will be estimated in each of the following proofs. The latter is used exclusively to transfer the solutions between different time meshes. The difference between the two is further illustrated in Figure 5.2. Finally, both $\mathbf{i}^k$ and $\mathbf{I}^k$ are properly defined projection operators and therefore, for any $\mathbf{u}^k \in X^k$, we have

$$\mathbf{i}^k \mathbf{u}^k = \mathbf{I}^k \mathbf{u}^k = \mathbf{u}^k. \tag{5.8}$$

Given this preliminary information, the semi-discrete problem is defined as

$$d_t^k u_1^k = f_1(i_1^k t, u_1^k, I_1^k u_2^k), \quad d_t^k u_2^k = f_2(i_2^k t, I_2^k u_1^k, u_2^k). \tag{5.9}$$

We will now prove a stability estimate of this semi-discrete system using Gronwall's lemma.

Theorem 1. *Let $\mathbf{u}$ be a continuous solution to (5.1) and $\mathbf{u}^k \in X^k$ its discrete counterpart and a solution to (5.9). Further, let us assume that $\mathbf{f} \in C^1(\bar{I})^2$ with Lipschitz constants L_1 and L_2, respectively. If we further assume that $(k_1^N + k_2^N)(L_1 + L_2) \leqslant \frac{1}{2}$, where $k_1^N := k_1^{N,N_1^N}$*
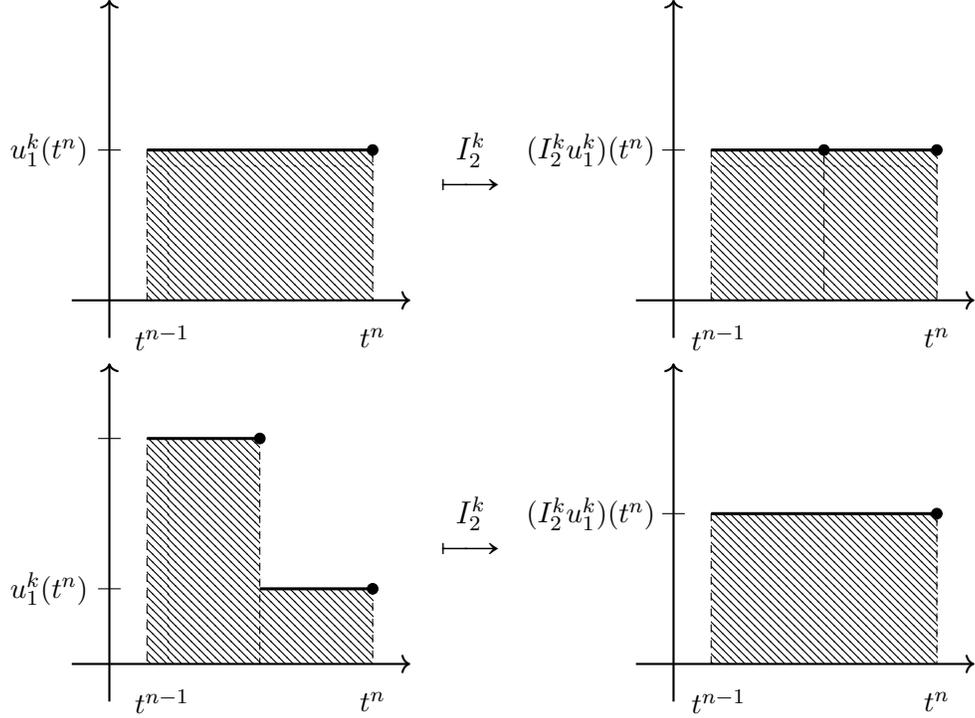
Figure 5.1: We present two examples of the transformation given by the projection operator $\mathbf{I}^k$. In the top sketch, one macro time-step is split into two smaller micro time-steps and $(I_2^k u_1^k)(t_2^{n,1}) = (I_2^k u_1^k)(t^n) = u_1^k(t^n)$. In the bottom sketch, two smaller micro time-steps are merged together with $(I_2^k u_1^k)(t^n) = \frac{k_1^{n,1}}{k^n} u_1^k(t_1^{n,1}) + \frac{k_1^{n,2}}{k^n} u_1^k(t^n)$.

and $k_2^N := k_2^{N,N_2^N}$ that is the sizes of the last time-steps in each of the timelines, then the following estimate holds

$$\left\| e_1^k(t^N) \right\| + \left\| e_2^k(t^N) \right\| \leqslant e^{2T(L_1+L_2)} \left(2\|\tau_1^k\| + 2\|\tau_2^k\| \right),$$

with the truncation errors (for $j = 1, 2$ and using the notation $\hat{j} = 3 - j$)

$$
\|\tau_j^k\| \leqslant \sum_{n=1}^{N} \sum_{m=1}^{N_j^n} \left\{ \frac{1}{2}(k_j^{n,m})^2 \max_{t \in I} \|d_t f_j(t, u_1, u_2)\| + L_j k^n k_j^{n,m} \max_{t \in I} \|f_j(t, u_1, u_2)\| \right.
$$
$$
\left. + L_{\hat{j}}(k_j^{n,m})^2 \max_{t \in I} \|f_j(t, u_1, u_2)\| \right\} \tag{5.10}
$$

and where the errors $\boldsymbol{e}^k = (e_1^k, e_2^k)^T$ are defined as

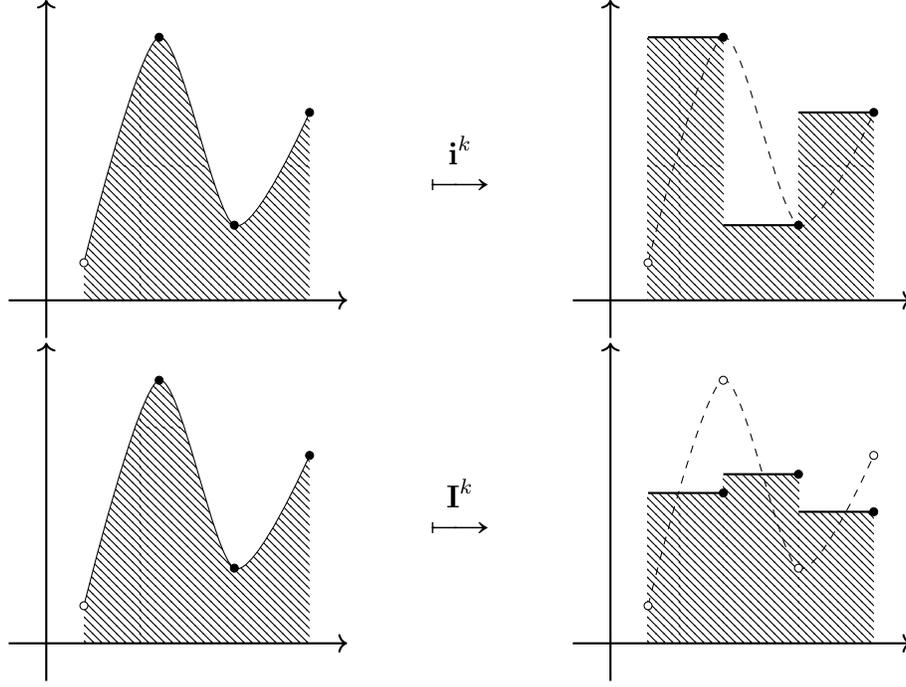$$e_j^k := u_j^k - i_j^k u_j, \quad j = 1, 2.$$

99

Figure 5.2: An example showing a difference between projection operators $\mathbf{i}^k$ and $\mathbf{I}^k$. The top figure shows the projection given by the $\mathbf{i}^k$ operator and in the bottom one, we instead have a look at the $\mathbf{I}^k$ operator.

Proof. Since the analysis of both the errors e_1^k and e_2^k is analogous, in this proof we will only estimate e_1^k. Using both the continuous (5.1) as well as the discrete (5.9) formulation we have

$$e_1^k(t_1^{n,m}) = e_1^k(t_1^{n,m-1}) + \int_{I_1^{n,m}} \left\{ f_1(t_1^{n,m}, u_1^k, I_1^k u_2^k) - f_1(t_1^{n,m}, u_1(t_1^{n,m}), \bar{I}^k u_2) \right\} \mathrm{d}t - \tau_{1,k}^{n,m},$$

where

$$\tau_{1,k}^{n,m} := u_1(t_1^{n,m}) - u_1(t_1^{n,m-1}) - \int_{I_1^{n,m}} f_1(t_1^{n,m}, u_1(t_1^{n,m}), \bar{I}^k u_2) \, \mathrm{d}t.$$

We sum up the values of the errors over the whole time interval

$$
\begin{aligned}
e_1^k(t^N) = \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} & \left\{ f_1(t_1^{n,m}, u_1^k, I_1^k u_2^k) \right. \\
& \left. - f_1(t_1^{n,m}, u_1(t_1^{n,m}), \bar{I}^k u_2) \right\} \mathrm{d}t - \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} \tau_{1,k}^{n,m} \, \mathrm{d}t.
\end{aligned}
\tag{5.11}
$$

Note that $e_1^k(0) = 0$. We will use the notation

$$\tau_1^k := \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} \tau_{1,k}^{n,m} \, \mathrm{d}t.$$

After applying the triangle inequality to equation (5.11) as well as the Lipschitz continuity of the function f_1

$$\|e_1^k(t^N)\| \leqslant \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} k_1^{n,m} L_1 \|e_1^k(t_1^{n,m})\| + \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} L_1 \|I_1^k u_2^k - \bar{I}^k u_2\| \, \mathrm{d}t - \tau_1^k. \tag{5.12}$$

We proceed with the estimation of the term $\sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} L_1 \|I_1^k u_2^k - \bar{I}^k u_2\| \, \mathrm{d}t$. Based on the definition of our time meshes, for each macro time-step I^n, there is micro time-stepping in only one of the submeshes. Therefore, it is sufficient to only consider the following possibilities:

1. There is no micro time-stepping in I_1^k, from which follows that $N_1^n = 1$ and $I_1^k\big|_{I^n} = \bar{I}^k\big|_{I^n}$

$$\sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} L_1 \|I_1^k u_2^k - \bar{I}^k u_2\| \, \mathrm{d}t = \int_{I^n} L_1 \|\bar{I}^k u_2^k - \bar{I}^k u_2\| \, \mathrm{d}t$$

$$= L_1 \left\| \sum_{m=1}^{N_2^n} k_2^{n,m} u_2^k(t_2^{n,m}) - \int_{I^n} u_2(s) \, \mathrm{d}s \right\|$$

$$\leqslant L_1 \sum_{m=1}^{N_2^n} k_2^{n,m} \left\{ \|e_2^k(t_2^{n,m})\| + \frac{1}{k_2^{n,m}} \int_{I_2^{n,m}} \|u_2(t_2^{n,m}) - u_2(s)\| \, \mathrm{d}s \right\}$$

$$\leqslant \sum_{m=1}^{N_2^n} L_1 k_2^{n,m} \|e_2^k(t_2^{n,m})\| + \sum_{m=1}^{N_2^n} L_1 (k_2^{n,m})^2 \max_{t \in I} \|f_2(t, u_1, u_2)\|.$$

2. There is no micro time-stepping in I_2^k, from which follows that $N_2^n = 1$ and $I_2^k\big|_{I^n} = \bar{I}^k\big|_{I^n}$

$$\sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} L_1 \|I_1^k u_2^k - \bar{I}^k u_2\| \, \mathrm{d}t = \sum_{m=1}^{N_1^n} L_1 k_1^{n,m} \left\| u_2^k(t^n) - \frac{1}{k^n} \int_{I^n} u_2(s) \, \mathrm{d}s \right\|$$

$$= L_1 \left\| k^n u_2^k(t^n) - \int_{I^n} u_2(s) \, \mathrm{d}s \right\| \leqslant L_1 k^n \|e_2^k(t^n)\| + L_1 (k^n)^2 \max_{t \in I} \|f_2(t, u_1, u_2)\|$$

$$= \sum_{m=1}^{N_2^n} L_1 k_2^{n,m} \|e_2^k(t_2^{n,m})\| + \sum_{m=1}^{N_2^n} L_1 (k_2^{n,m})^2 \max_{t \in I} \|f_2(t, u_1, u_2)\|.$$

Let us continue with the estimation of the remaining term in the error equation (5.11)

$$\|\tau_1^k\| \leqslant \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \left\| u_1(t_1^{n,m}) - u_1(t_1^{n,m-1}) - k_1^{n,m} f_1(t_1^{n,m}, u_1(t_1^{n,m}), u_2(t_1^{n,m})) \right\|$$

$$+ \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \int_{I_1^{n,m}} \left\| f_1(t_1^{n,m}, u_1(t_1^{n,m}), u_2(t_1^{n,m})) - f_1(t_1^{n,m}, u_1(t_1^{n,m}), \bar{I}^k u_2) \right\| \mathrm{d}t.$$

It holds

$$\left\| u_1(t_1^{n,m}) - u_1(t_1^{n,m-1}) - k_1^{n,m} f_1(t_1^{n,m}, u_1(t_1^{n,m}), u_2(t_1^{n,m})) \right\|$$

$$\leqslant \frac{1}{2} (k_1^{n,m})^2 \max_{t \in I} \| d_t f_1(t, u_1, u_2) \|$$

as well as

$$\int_{I_1^{n,m}} \left\| f_1(t_1^{n,m}, u_1(t_1^{n,m}), u_2(t_1^{n,m})) - f_1(t_1^{n,m}, u_1(t_1^{n,m}), \bar{I}^k u_2) \right\| \mathrm{d}t$$

$$\leqslant L_1 \frac{k_1^{n,m}}{k^n} \left\| \int_{I^n} (u_2(t_1^{n,m}) - u_2(s)) \, \mathrm{d}s \right\| \leqslant L_1 k_1^{n,m} k^n \max_{t \in I} \| f_1(t, u_1, u_2) \|.$$

Analogously, one can analyze e_2^k. Once it is done, we can proceed with Gronwall's lemma. The terms corresponding to the last time-step in the inequality (5.12) are then transferred from the right to the left side. Assuming that $(k_1^N + k_2^N)(L_1 + L_2) \leqslant \frac{1}{2}$, then it holds

$$\frac{1}{2} \left(\|e_1^k(t^N)\| + \|e_2^k(t^N)\| \right) \leqslant \{1 - k_1^N(L_1 + L_2)\} \|e_1^k(t^N)\| + \{1 - k_2^N(L_1 + L_2)\} \|e_2^k(t^N)\|.$$

Applying Gronwall's lemma yields the results. □

Remark 5.2.1 (Separation of the time scales)**.** *In a simplified form, we were able to prove*

$$\left\| e^k(t^N) \right\| = \mathcal{O}\left(k_1 \|d_t f_1(t, u_1, u_2)\|\right) + \mathcal{O}\left(k \|f_1(t, u_1, u_2)\|\right)$$
$$+ \mathcal{O}\left(k_2 \|d_t f_2(t, u_1, u_2)\|\right) + \mathcal{O}\left(k \|f_2(t, u_1, u_2)\|\right).$$

Based on that, let us make a few observations. First, we obtained linear convergence in time typical for the implicit Euler scheme. Second, while we were not able to fully decouple the two problems, the macro time step k only acts on the lower order term f but not its derivative. This indicates that the oscillations of functions f_1, and f_2 are localizable to each of the two subproblems and an efficient discretization by a multirate method is possible. This is in agreement with the a posteriori error estimate and the numerical results demonstrated in [23].
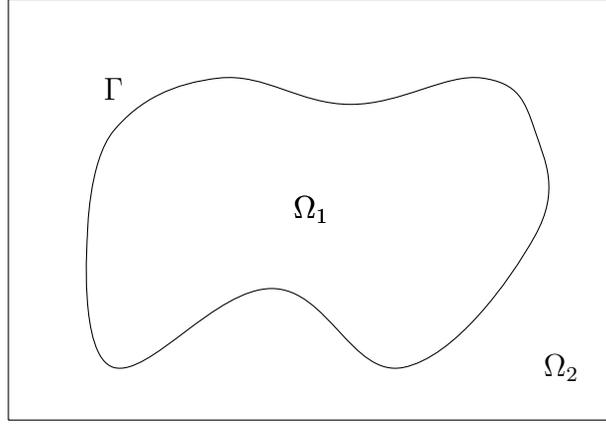
Figure 5.3: We show the splitting of the domain Ω into Ω_1 and Ω_2 with a common interface Γ.

5.3 Coupling of Heat Equations

In this section, we consider a heat equation prescribed on two domains $\bar{\Omega}_1 \cup \bar{\Omega}_2 = \Omega \subset \mathbb{R}^d$ for $d \in \{2,3\}$ with a common interface Γ. The domains are illustrated in Figure 5.3. On each of the domains, we choose different diffusivity constants $\boldsymbol{\nu} = (\nu_1, \nu_2)^T$ and external forces $\mathbf{f} = (\mathbf{f}_1, \mathbf{f}_2)^T \in L^2(\Omega)^d$. We define a space-time domain for any arbitrary function space V

$$X(V) := \left\{ v \in L^2(I, V) \mid \partial_t v \in L^2(I, V^*) \right\}, \tag{5.13}$$

and take $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2)^T$ with $\mathbf{u}_j \in X(H^1(\Omega_j))^d$ for $j = 1, 2$. The solution $\mathbf{u} : \Omega \times I \to \mathbb{R}^d$ is governed by the equations

$$\partial_t \mathbf{u}_1 - \nu_1 \Delta \mathbf{u}_1 = \mathbf{f}_1 \ \text{ in } \Omega_1 \text{ and } \partial_t \mathbf{u}_2 - \nu_2 \Delta \mathbf{u}_2 = \mathbf{f}_2 \ \text{ in } \Omega_2. \tag{5.14}$$

On the interface $\Gamma = \partial\Omega_1 \cap \partial\Omega_2$ we impose coupling conditions typical for continuous two-phase flow problems, see, for instance [62], that is continuity of the solutions (kinematic condition) and balance of stress in the normal direction (dynamic condition)

$$\begin{aligned}
\mathbf{u}_1 &= \mathbf{u}_2 & \text{on } \Gamma, \\
\nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 &= -\nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2 & \text{on } \Gamma.
\end{aligned} \tag{5.15}$$

$\vec{\mathbf{n}}_1$ and $\vec{\mathbf{n}}_2$ denote normal vectors corresponding to each of the domains. In particular, on the interface $\vec{\mathbf{n}}_1 = -\vec{\mathbf{n}}_2$. On the outer boundary $\partial\Omega$ we choose a no-slip boundary condition $\mathbf{u}_1 = \mathbf{u}_2 = \mathbf{0}$. Similarly, at initial time we set $\mathbf{u}_1(0) = \mathbf{u}_2(0) = \mathbf{0}$. After integration by parts using test functions $\boldsymbol{\varphi} = (\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2)^T$ with $\boldsymbol{\varphi}_j \in X(H^1(\Omega_j))^d$ for $j = 1, 2$, we obtain

$$\begin{aligned}
&\int_I \left\{ (\mathbf{d}_t \mathbf{u}, \boldsymbol{\varphi})_\Omega + \boldsymbol{\nu}(\nabla \mathbf{u}, \nabla \boldsymbol{\varphi})_\Omega - \langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1, \boldsymbol{\varphi}_1 \rangle_\Gamma - \langle \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \boldsymbol{\varphi}_2 \rangle_\Gamma \right\} \mathrm{d}t \\
&= \int_I (\mathbf{f}, \boldsymbol{\varphi})_\Omega \, \mathrm{d}t.
\end{aligned} \tag{5.16}$$

Given the coupling conditions (5.15), we have

$$\nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 = \frac{1}{2} \left(\nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2 \right) = -\nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2 \tag{5.17}$$

and therefore the interface terms are equal to

$$- \langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1, \boldsymbol{\varphi}_1 \rangle_\Gamma - \langle \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \boldsymbol{\varphi}_2 \rangle_\Gamma = \frac{1}{2} \langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma .$$

To symmetrize the formulation, we subtract

$$\frac{1}{2} \langle \mathbf{u}_2 - \mathbf{u}_1, \nu_1 \partial_{\vec{\mathbf{n}}_1} \boldsymbol{\varphi}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \boldsymbol{\varphi}_2 \rangle_\Gamma , \tag{5.18}$$

which vanishes once $\mathbf{u}_1 = \mathbf{u}_2$ on the interface. Further a Nitsche term is added

$$\gamma \langle \mathbf{u}_2 - \mathbf{u}_1, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma .$$

γ denotes the Nitsche constant. Traditionally, this interface term is scaled by $\frac{\gamma}{h}$ instead of γ, where h stands for the space mesh size. Since we consider both semi-discrete and fully discrete cases, h will be included in fully discrete formulations only. For more details on the Nitsche method, we refer to the original paper [26] and to [63], where similar approaches are applied to two-phase flow problems and fluid-structure interactions. Since the exact solution fulfills coupling conditions and therefore the interface terms are equal to zero, we obtain a consistent and, as we will later see, coercive formulation

$$\begin{aligned} a(\mathbf{u}, \boldsymbol{\varphi}) &:= \int_I \bigg\{ (\mathbf{d}_t \mathbf{u}, \boldsymbol{\varphi})_\Omega + \boldsymbol{\nu} \left(\nabla \mathbf{u}, \nabla \boldsymbol{\varphi} \right)_\Omega + \frac{1}{2} \langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma \\ &\quad - \frac{1}{2} \langle \mathbf{u}_2 - \mathbf{u}_1, \nu_1 \partial_{\vec{\mathbf{n}}_1} \boldsymbol{\varphi}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \boldsymbol{\varphi}_2 \rangle_\Gamma + \gamma \langle \mathbf{u}_2 - \mathbf{u}_1, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma \bigg\} \, \mathrm{d}t \\ &= \int_I (\mathbf{f}, \boldsymbol{\varphi})_\Omega \, \mathrm{d}t. \end{aligned} \tag{5.19}$$

This formulation guarantees the fulfillment of coupling conditions (5.15) even without any assumptions on the continuity of trial and test functions on the interface. Due to the arbitrariness of test functions, condition (5.18) leads to the continuity across the interface of solutions. Additional interface terms coming from integration by parts and returning to the strong formulation (5.14) as well as property (5.17) guarantee the balance of stress. This variational treatment of the interface conditions was first proposed by P. Hansbo and M. G. Larson in [5].

We will use the notations $(\cdot, \cdot)_{\Omega_1}$, $(\cdot, \cdot)_{\Omega_2}$ and $(\cdot, \cdot)_\Omega$ to indicate the L^2-product over a corresponding domain. The norms over each of the domains will be denoted in a similar way. On the interface, using Riesz representation theorem, we then define

$$\langle \mathbf{u}, \boldsymbol{\varphi} \rangle_\Gamma := \langle \mathbf{u}, \boldsymbol{\varphi} \rangle_{H^{-\frac{1}{2}}(\Gamma)^d \times H^{\frac{1}{2}}(\Gamma)^d} , \qquad \|\mathbf{u}\|_\Gamma := \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle_\Gamma} .$$

5.3.1 Semi-discrete Case

With the help of the projection operators $\mathbf{I}_1^k$ and $\mathbf{I}_2^k$ (we use bold letters to indicate that projections are performed on multidimensional functions), we are ready to formulate a semi-discrete variational problem again using the implicit Euler time-stepping scheme

$$
\begin{aligned}
a^k(\mathbf{u}^k, \boldsymbol{\varphi}^k) &:= \int_I \Big\{ \left(\mathbf{d}_t^k \mathbf{u}^k, \boldsymbol{\varphi}^k \right)_\Omega + \boldsymbol{\nu} \left(\nabla \mathbf{u}^k, \nabla \boldsymbol{\varphi}^k \right)_\Omega - \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{I}_1^k \mathbf{u}_2^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \\
&\quad + \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{I}_2^k \mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma - \frac{1}{2} \left\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k, \nu_1 \partial_{\vec{\mathbf{n}}_1} \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \\
&\quad + \frac{1}{2} \left\langle \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k, \nu_2 \partial_{\vec{\mathbf{n}}_2} \boldsymbol{\varphi}_2^k \right\rangle_\Gamma - \gamma \left\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \\
&\quad + \gamma \left\langle \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \Big\} \, \mathrm{d}t \\
&= \int_I \left(\mathbf{f}, \boldsymbol{\varphi}^k \right)_\Omega \mathrm{d}t.
\end{aligned}
\tag{5.20}
$$

The corresponding function spaces are defined as $(j = 1, 2)$

$$
X_j^k := \left\{ \boldsymbol{\varphi} \in L^2(\bar{I}, H^1(\Omega_j)) \,\middle|\, \varphi|_{I_j^{n,m}} \in \mathcal{P}_0(H^1(\Omega_j)) \text{for all } I_j^{n,m} \subset I, \ \boldsymbol{\varphi}(0) = \mathbf{0} \right\}
\tag{5.21}
$$

and $X^k := X_1^k \times X_2^k$. We also introduce a similar space defined over the whole domain Ω

$$
Y^k := \left\{ \boldsymbol{\varphi} = (\varphi_1, \varphi_2)^T \in L^2(\bar{I}, H^1(\Omega)) \,\middle|\, \varphi_1|_{\Omega_1} \in X_1^k, \varphi_2|_{\Omega_2} \in X_2^k \right\},
$$

By testing the semi-discrete formulation with functions concentrated only on the interface, on each macro time-step I_n, we have for any $\boldsymbol{\varphi}_1^k \in (X_1^k)^d$ and $\boldsymbol{\varphi}_2^k \in (X_2^k)^d$

$$
0 = \int_{I_n} \left\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t = \int_{I_n} \left\langle \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t,
$$

$$
0 = \int_{I_n} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1^k + \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{I}_1^k \mathbf{u}_2^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t = \int_{I_n} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{I}_2^k \mathbf{u}_1^k + \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t.
$$

These identities together with the property (5.7) allow us to recover the coupling conditions in a weak form, that is for any $\boldsymbol{\varphi}_1^k \in (X_1^k)^d$ and $\boldsymbol{\varphi}_2^k \in (X_2^k)^d$

$$
0 = \int_{I_n} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t = \int_{I_n} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t,
$$

$$
0 = \int_{I_n} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1^k + \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t = \int_{I_n} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1^k + \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t.
\tag{5.22}
$$

Now we will prove an error estimate given by the following estimation

Theorem 2. *Let $\boldsymbol{u} \in X(H_0^1(\Omega))^d$, $\boldsymbol{u}_j \in W^{1,\infty}(H^2(\Omega_j))^d$ for $j = 1, 2$ be continuous solutions to (5.19) and $\boldsymbol{u}^k \in (X^k)^d$ their semi-discrete counterpart and a solution to (5.20), then the following estimate holds*

$$\left\| \boldsymbol{e}^k(t^N) \right\|_\Omega^2 + \int_I \boldsymbol{\nu}^2 \left\| \nabla \boldsymbol{e}^k \right\|_\Omega^2 \mathrm{d}t$$

$$\leqslant C_1 \sum_{n=1}^N \sum_{m=1}^{N_1^n} \left\{ (k_1^{n,m})^3 \max_{t \in I} \|\boldsymbol{d}_t \nabla \boldsymbol{u}_1\|_{\Omega_1}^2 + (k_1^{n,m})^3 \max_{t \in I} \|\boldsymbol{d}_t \partial_{\vec{n}_1} \boldsymbol{u}_1\|_\Gamma^2 \right\}$$

$$+ C_2 \sum_{n=1}^N \sum_{m=1}^{N_2^n} \left\{ (k_2^{n,m})^3 \max_{t \in I} \|\boldsymbol{d}_t \nabla \boldsymbol{u}_2\|_{\Omega_2}^2 + (k_2^{n,m})^3 \max_{t \in I} \|\boldsymbol{d}_t \partial_{\vec{n}_2} \boldsymbol{u}_2\|_\Gamma^2 \right\},$$

where the errors $\boldsymbol{e}^k = (\boldsymbol{e}_1^k, \boldsymbol{e}_2^k)^T$ are defined as

$$\boldsymbol{e}_1^k := \boldsymbol{u}_1^k - \boldsymbol{i}_1^k \boldsymbol{u}_1, \qquad \boldsymbol{e}_2^k := \boldsymbol{u}_2^k - \boldsymbol{i}_2^k \boldsymbol{u}_2.$$

Proof. Using Galerkin orthogonality, we have $a^k(\mathbf{u}^k, \mathbf{e}^k) = a(\mathbf{u}, \mathbf{e}^k)$ and therefore it holds

$$a^k(\mathbf{u}^k, \mathbf{e}^k) - \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k \right)_\Omega + \boldsymbol{\nu}\left(\nabla \mathbf{i}^k \mathbf{u}, \nabla \mathbf{e}^k \right)_\Omega \right\} \mathrm{d}t$$

$$= a(\mathbf{u}, \mathbf{e}^k) - \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k \right)_\Omega + \boldsymbol{\nu}\left(\nabla \mathbf{i}^k \mathbf{u}, \nabla \mathbf{e}^k \right)_\Omega \right\} \mathrm{d}t. \tag{5.23}$$

By adding and subtracting terms, the lest side of this identity can be rewritten as

$$a^k(\mathbf{u}^k, \mathbf{e}^k) - \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k \right)_\Omega + \boldsymbol{\nu}\left(\nabla \mathbf{i}^k \mathbf{u}, \nabla \mathbf{e}^k \right)_\Omega \right\} \mathrm{d}t$$

$$= \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{e}^k, \mathbf{e}^k \right)_\Omega + \boldsymbol{\nu}\|\nabla \mathbf{e}^k\|_\Omega^2 + \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2^k, \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma \right.$$

$$- \frac{1}{2} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^k \right\rangle_\Gamma + \gamma \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma$$

$$- \frac{1}{2} \left\langle \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2^k - \mathbf{I}_1^k \mathbf{u}_2^k), \mathbf{e}_1^k \right\rangle_\Gamma + \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{I}_2^k \mathbf{u}_1^k - \mathbf{u}_1^k), \mathbf{e}_2^k \right\rangle_\Gamma \tag{5.24}$$

$$- \frac{1}{2} \left\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_2^k, \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^k \right\rangle_\Gamma + \frac{1}{2} \left\langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^k \right\rangle_\Gamma$$

$$- \gamma \left\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_2^k, \mathbf{e}_1^k \right\rangle_\Gamma + \gamma \left\langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \mathbf{e}_2^k \right\rangle_\Gamma \left. \right\} \mathrm{d}t.$$

Since the semi-discrete solution fulfills the coupling conditions in the sense of (5.22), specifically the Dirichlet condition, we have

$$\int_I \left\{ -\frac{1}{2} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^k \right\rangle_\Gamma + \gamma \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma \right\} \mathrm{d}t = 0.$$

One can further simplify expression (5.24) by noticing, that on every macro time-step I^n

$$\int_{I^n} \Big\{ -\frac{1}{2}\langle \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2^k - \mathbf{I}_1^k \mathbf{u}_2^k), \mathbf{e}_1^k\rangle_\Gamma + \frac{1}{2}\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{I}_2^k \mathbf{u}_1^k - \mathbf{u}_1^k), \mathbf{e}_2^k\rangle_\Gamma$$
$$-\frac{1}{2}\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_2^k, \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{e}_1^k\rangle_\Gamma + \frac{1}{2}\langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \nu_2 \partial_{\vec{\mathbf{n}}_2}\mathbf{e}_2^k\rangle_\Gamma \tag{5.25}$$
$$-\gamma\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_2^k, \mathbf{e}_1^k\rangle_\Gamma + \gamma\langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \mathbf{e}_2^k\rangle_\Gamma \Big\}\,\mathrm{d}t = 0.$$

To explain that, let us look at the integral $\int_{I^n} \langle \mathbf{u}_1^k - I_2^k \mathbf{u}_1^k, \mathbf{e}_2^k\rangle_\Gamma\,\mathrm{d}t$. From the construction of our time meshes, there are two possibilities. According to the first one, there is no micro time-stepping in the domain Ω_1, in other words, $N_1^n = 1$ and $\mathbf{I}_1^k\big|_{I^n} = \bar{\mathbf{I}}^k\big|_{I^n}$. Then $\mathbf{I}_2^k \mathbf{u}_1^k = \mathbf{u}_1^k$ and therefore $\langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \mathbf{e}_2^k\rangle_\Gamma = 0$. Otherwise, we have no micro time-stepping in the domain Ω_2 ($N_2^n = 1$ and $\mathbf{I}_2^k\big|_{I^n} = \bar{\mathbf{I}}^k\big|_{I^n}$). In this case, knowing that the test function $\mathbf{e}_2^k$ is a continuous constant over the interval I^n and using the property (5.7) of the projection operator $\mathbf{I}_2^k$, we can write

$$\int_{I^n} \langle \mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k, \mathbf{e}_2^k\rangle_\Gamma\,\mathrm{d}t = \Big\langle \int_{I^n}(\mathbf{u}_1^k - \mathbf{I}_2^k \mathbf{u}_1^k)\,\mathrm{d}t, \mathbf{e}_2^k\Big\rangle_\Gamma = 0.$$

The reasoning corresponding to the remaining terms in (5.25) is analogous. Let us look at the unresolved interface term. The weak coupling conditions can be further exploited and lead to

$$\int_{I^n} \frac{1}{2}\Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2}\mathbf{u}_2^k, \mathbf{e}_2^k - \mathbf{e}_1^k\Big\rangle_\Gamma\,\mathrm{d}t = \int_{I^n} \frac{1}{2}\Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2}\mathbf{u}_2^k, \mathbf{i}_1^k \mathbf{u}_1 - \mathbf{i}_2^k \mathbf{u}_2\Big\rangle_\Gamma\,\mathrm{d}t$$

Let us look closely at the implications of our time mesh structure. Because of the symmetry of this expression, without loss of generality, we can assume that $N_1^n = 1$ and $\mathbf{I}_1^k\big|_{I^n} = \bar{\mathbf{I}}^k\big|_{I^n}$. The use of both the weak (5.22) and the strong (5.15) coupling conditions leads us to

$$\int_{I^n} \frac{1}{2}\Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1^k - \nu_2 \partial_{\vec{\mathbf{n}}_2}\mathbf{u}_2^k, \mathbf{i}_1^k \mathbf{u}_1 - \mathbf{i}_2^k \mathbf{u}_2\Big\rangle_\Gamma\,\mathrm{d}t$$
$$= \Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1^k, \int_{I^n}(\mathbf{i}_1^k \mathbf{u}_1 - \mathbf{u}_1)\,\mathrm{d}t\Big\rangle_\Gamma + \Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1^k, \int_{I^n}(\mathbf{u}_2 - \mathbf{i}_2^k \mathbf{u}_2)\,\mathrm{d}t\Big\rangle_\Gamma = 0.$$

The left side of the identity (5.24), with the help of the strong coupling conditions, is equal to

$$a(\mathbf{u}, \mathbf{e}^k) - \int_I \Big\{ (\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k)_\Omega + \boldsymbol{\nu}(\nabla \mathbf{i}^k \mathbf{u}, \nabla \mathbf{e}^k)_\Omega \Big\}\,\mathrm{d}t$$
$$= \int_I \Big\{ (\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k)_\Omega + \boldsymbol{\nu}(\nabla(\mathbf{u} - \mathbf{i}^k \mathbf{u}), \nabla \mathbf{e}^k)_\Omega$$
$$+ \frac{1}{2}\Big\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}\mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2}\mathbf{u}_2, \mathbf{e}_2^k - \mathbf{e}_1^k\Big\rangle_\Gamma \Big\}\,\mathrm{d}t.$$

While analyzing the remaining terms in (5.24), we will concentrate on the error $\mathbf{e}_1^k$ since the estimations corresponding to the second error are very similar. Therefore, a single interval $I_1^{n,m}$ will be considered. Let us start with the time discretization error

$$\int_{I_1^{n,m}} \mathbf{d}_t \mathbf{u}_1 \, dt = \mathbf{u}_1(t_1^{n,m}) - \mathbf{u}_1(t_1^{n,m-1}) = \int_{I_1^{n,m}} \mathbf{d}_t^k \mathbf{i}_1^k \mathbf{u}_1 \, dt.$$

As a result, we have

$$\int_{I_1^{n,m}} \left(\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}_1^k \mathbf{u}, \mathbf{e}^k \right)_\Omega dt = \mathbf{0}. \tag{5.26}$$

We will now examine the Laplacian terms on the right side of (5.24). Knowing that the error $\mathbf{e}_1^k$ is constant in time on every interval $I_1^{n,m}$, we have

$$\left| \int_{I_1^{n,m}} \nu_1 \left(\nabla(\mathbf{u}_1 - \mathbf{i}_1^k \mathbf{u}_1), \nabla \mathbf{e}_1^k \right)_{\Omega_1} dt \right|$$
$$= \left| \int_{I_1^{n,m}} \int_t^{t_1^{n,m}} \nu_1 \left(\mathbf{d}_t \nabla \mathbf{u}_1(s), \nabla \mathbf{e}_1^k(t_1^{n,m}) \right)_{\Omega_1} ds \, dt \right| \tag{5.27}$$
$$\leqslant c(k_1^{n,m})^3 \max_{t \in I} \|\mathbf{d}_t \nabla \mathbf{u}_1\|_{\Omega_1}^2 + \frac{1}{8} \nu_1^2 \int_{I_1^{n,m}} \left\| \nabla \mathbf{e}_1^k(t_1^{n,m}) \right\|_{\Omega_1}^2 dt.$$

The remaining time discretization term in (5.24) can be rewritten as

$$\int_{I_1^{n,m}} \left(\mathbf{d}_t^k \mathbf{e}_1^k, \mathbf{e}_1^k \right)_{\Omega_1} dt = \frac{1}{2} \left\| \mathbf{e}_1^k(t_1^{n,m}) \right\|_{\Omega_1}^2 - \frac{1}{2} \left\| \mathbf{e}_1^k(t_1^{n,m-1}) \right\|_{\Omega_1}^2$$
$$+ \frac{1}{2} \left\| \mathbf{e}_1^k(t_1^{n,m}) - \mathbf{e}_1^k(t_1^{n,m-1}) \right\|_{\Omega_1}^2. \tag{5.28}$$

Summing these terms over the whole time interval, we obtain

$$\int_I \left(\mathbf{d}_t^k \mathbf{e}_1^k, \mathbf{e}_1^k \right)_{\Omega_1} dt = \frac{1}{2} \left\| \mathbf{e}_1^k(t^N) \right\|_{\Omega_1}^2 + \frac{1}{2} \sum_{n=1}^N \sum_{m=1}^{N_1^n} \left\| \mathbf{e}_1^k(t_1^{n,m}) - \mathbf{e}_1^k(t_1^{n,m-1}) \right\|_{\Omega_1}^2$$

and therefore

$$\left\| \mathbf{e}^k(t^N) \right\|_\Omega^2 \leqslant \int_I 2 \left(\mathbf{d}_t^k \mathbf{e}^k, \mathbf{e}^k \right)_\Omega dt.$$

Let us proceed to the last interface term on the right side of (5.23). Implementing very similar solutions as in the analysis of the previous interface terms, we can show that

$$\int_{I^n} \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma dt$$
$$= \int_{I^n} \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{u}_2), \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma dt \tag{5.29}$$
$$\leqslant \int_{I^n} \frac{1}{2} \left\| \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{u}_2) \right\|_\Gamma \left\| \mathbf{e}_2^k - \mathbf{e}_1^k \right\|_\Gamma dt.$$

The first term in (5.29) can be estimated using the fundamental theorem of calculus, from which follows

$$\nu_j \left\| \partial_{\vec{\mathbf{n}}_j}(\mathbf{u}_j - \mathbf{I}_j^k \mathbf{u}_j) \right\|_\Gamma \leqslant \nu_j k_j^{n,m} \max_{t \in I} \left\| \mathbf{d}_t \partial_{\vec{\mathbf{n}}_j} \mathbf{u}_j \right\|_\Gamma, \quad j = 1, 2.$$

The other term can be dealt with by using the trace inequality

$$\left\| \mathbf{e}_2^k - \mathbf{e}_1^k \right\|_\Gamma \leqslant \left\| \mathbf{e}_1^k \right\|_\Gamma + \left\| \mathbf{e}_2^k \right\|_\Gamma \leqslant c_1 \left\| \nabla \mathbf{e}_1^k \right\|_{\Omega_1} + c_2 \left\| \nabla \mathbf{e}_2^k \right\|_{\Omega_2}.$$

After combining these estimates using the Young and Poincaré inequalities

$$\left| \int_{I^n} \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2, \mathbf{e}_2^k - \mathbf{e}_1^k \right\rangle_\Gamma \mathrm{d}t \right|$$

$$\leqslant \sum_{m=1}^{N_1^n} \left\{ c_1 (k_1^{n,m})^3 \max_{t \in I} \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1\|_\Gamma^2 + \frac{1}{8} \nu_1^2 \int_{I_1^{n,m}} \left\| \nabla \mathbf{e}_1^k(t_1^{n,m}) \right\|_{\Omega_1}^2 \mathrm{d}t \right\} \quad (5.30)$$

$$+ \sum_{m=1}^{N_2^n} \left\{ c_2 (k_2^{n,m})^3 \max_{t \in I} \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2\|_\Gamma^2 + \frac{1}{8} \nu_2^2 \int_{I_2^{n,m}} \left\| \nabla \mathbf{e}_2^k(t_2^{n,m}) \right\|_{\Omega_2}^2 \mathrm{d}t \right\}$$

Performing the same steps for the solution $\mathbf{u}_2$ and summing these terms over the whole interval I will lead to the final result. $\qquad\square$

In a more compact way, we just proved that

$$\|\mathbf{e}^k(t^N)\| + \int_I \boldsymbol{\nu} \|\mathbf{e}^k\| \, \mathrm{d}t = \mathcal{O}\left(k_1 \|\mathbf{d}_t \nabla \mathbf{u}_1\|_{\Omega_1}\right) + \mathcal{O}\left(k_1 \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1\|_\Gamma\right)$$

$$+ \mathcal{O}\left(k_2 \|\mathbf{d}_t \nabla \mathbf{u}_2\|_{\Omega_2}\right) + \mathcal{O}\left(k_2 \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2\|_\Gamma\right).$$

The convergence is linear which is expected for the implicit Euler method. Here, we were able to fully decouple the system. Due to the interface coupling conditions, we avoided interdependencies between contributions from different time discretizations.

5.3.2 Fully Discrete Case

To discretize the problem in space, we introduce regular triangulations $\mathcal{T}_1^h$ and $\mathcal{T}_2^h$. We assume that they match across the interface Γ. K_1 denotes an element of the mesh $\mathcal{T}_1^h$ and by K_2 an element of $\mathcal{T}_2^h$. Their sizes are denoted by h_1^K and h_2^K, respectively. Further,

$$h_1 := \max_{K_1 \in \mathcal{T}_1^h} h_1^K, \quad h_2 := \max_{K_2 \in \mathcal{T}_2^h} h_2^K, \quad h := \max\{h_1, h_2\}.$$

As function spaces, we take the space of continuous polynomials of order r

$$X_1^{k,h}(r) = \left\{ \boldsymbol{\varphi} \in X_1^k \middle| \varphi|_{K_1} \in \mathcal{P}_r(\Omega_1) \text{ for all } K_1 \in \mathcal{T}_1^h \text{ and } \varphi|_{\partial \Omega_1 \backslash \Gamma} = \mathbf{0} \right\},$$

$$X_2^{k,h}(r) = \left\{ \boldsymbol{\varphi} \in X_2^k \middle| \varphi|_{K_2} \in \mathcal{P}_r(\Omega_2) \text{ for all } K_2 \in \mathcal{T}_2^h \text{ and } \varphi|_{\partial \Omega_2 \backslash \Gamma} = \mathbf{0} \right\}.$$

We similarly define the function space

$$Y^{k,h}(r) := \left\{ \boldsymbol{\varphi} \in Y^k \middle| \varphi|_{\Omega_1} \in X_1^{k,h}(r) \text{ and } \varphi|_{\Omega_2} \in X_2^{k,h}(r) \right\}.$$

Let us introduce a Ritz projection operator. To ensure continuity over the interface, we will define it over the space $Y^{k,h}(r)$ instead of

$$X^{k,h}(r) := X_1^{k,h}(r) \times X_2^{k,h}(r).$$

As a consequence, we take

$$\mathbf{R}^h \mathbf{u} = (\mathbf{R}_1^h \mathbf{u}_1, \mathbf{R}_2^h \mathbf{u}_2)^T \in (Y^{k,h}(r))^d$$

defined by

$$\left(\nabla \mathbf{R}^h \mathbf{u}, \nabla \boldsymbol{\varphi}^{k,h} \right)_{\Omega} = \left(\nabla \mathbf{u}, \nabla \boldsymbol{\varphi}^{k,h} \right)_{\Omega} \quad \text{for all} \quad \boldsymbol{\varphi}^{k,h} \in (Y^{k,h}(r))^d. \tag{5.31}$$

Again, since the Ritz projection operator is imposed on a function space consisting only of functions continuous across the interface, we have

$$\mathbf{R}_1^h \mathbf{u}_1 \Big|_{\Gamma} = \mathbf{R}_2^h \mathbf{u}_2 \Big|_{\Gamma}. \tag{5.32}$$

We will now list some of the useful properties of the Ritz operator.

Corollary 1. *Given $\mathbf{u} \in X(H_0^1(\Omega))^d$, $\mathbf{u}_j^k \in X(H^{r+1}(\Omega_j))^d$ for $j = 1, 2$, the Ritz projection operator defined by (5.31) has the following properties:*

(i) $\left\| \mathbf{u} - \mathbf{R}^h \mathbf{u} \right\|_{\Omega} \leqslant c_1 h^{r+1} \left\| \nabla^{r+1} \mathbf{u}_1 \right\|_{\Omega_1} + c_2 h^{r+1} \left\| \nabla^{r+1} \mathbf{u}_2 \right\|_{\Omega_2},$

(ii) $\left\| \nabla(\mathbf{u} - \mathbf{R}^h \mathbf{u}) \right\|_{\Omega} \leqslant c_1 h^r \left\| \nabla^{r+1} \mathbf{u}_1 \right\|_{\Omega_1} + c_2 h^r \left\| \nabla^{r+1} \mathbf{u}_2 \right\|_{\Omega_2},$

(iii) $\left\| \nabla(\mathbf{u}_1 - \mathbf{R}_1^h \mathbf{u}_1) \cdot \vec{\boldsymbol{n}}_1 \right\|_{\Gamma} + \left\| \nabla(\mathbf{u}_2 - \mathbf{R}_2^h \mathbf{u}_2) \cdot \vec{\boldsymbol{n}}_2 \right\|_{\Gamma} \leqslant c_1 h^{r-\frac{1}{2}} \left\| \nabla^{r+1} \mathbf{u}_1 \right\|_{\Omega_1} + c_2 h^{r-\frac{1}{2}} \left\| \nabla^{r+1} \mathbf{u}_2 \right\|_{\Omega_2}.$

Then, our variational problem is given by

$$a^{k,h}(\mathbf{u}^{k,h}, \boldsymbol{\varphi}^{k,h}) = \int_I \left\{ (\mathbf{d}_t^k \mathbf{u}^{k,h}, \boldsymbol{\varphi}^{k,h})_{\Omega} + \nu \left(\nabla \mathbf{u}^{k,h}, \nabla \boldsymbol{\varphi}^{k,h} \right)_{\Omega} \right.$$

$$- \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\boldsymbol{n}}_1} \mathbf{u}_1^{k,h} - \nu_2 \partial_{\vec{\boldsymbol{n}}_2} \mathbf{I}_1^k \mathbf{u}_2^{k,h}, \boldsymbol{\varphi}_1^{k,h} \right\rangle_{\Gamma} + \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\boldsymbol{n}}_1} \mathbf{I}_2^k \mathbf{u}_1^{k,h} - \nu_2 \partial_{\vec{\boldsymbol{n}}_2} \mathbf{u}_2^{k,h}, \boldsymbol{\varphi}_2^{k,h} \right\rangle_{\Gamma}$$

$$- \frac{1}{2} \left\langle \mathbf{I}_1^k \mathbf{u}_2^{k,h} - \mathbf{u}_1^{k,h}, \nu_1 \partial_{\vec{\boldsymbol{n}}_1} \boldsymbol{\varphi}_1^{k,h} \right\rangle_{\Gamma} + \frac{1}{2} \left\langle \mathbf{u}_2^{k,h} - \mathbf{I}_2^k \mathbf{u}_1^{k,h}, \nu_2 \partial_{\vec{\boldsymbol{n}}_2} \boldsymbol{\varphi}_2^{k,h} \right\rangle_{\Gamma} \tag{5.33}$$

$$\left. - \frac{\gamma}{h} \left\langle \mathbf{I}_1^k \mathbf{u}_2^{k,h} - \mathbf{u}_1^{k,h}, \boldsymbol{\varphi}_1^{k,h} \right\rangle_{\Gamma} + \frac{\gamma}{h} \left\langle \mathbf{u}_2^{k,h} - \mathbf{I}_2^k \mathbf{u}_1^{k,h}, \boldsymbol{\varphi}_2^{k,h} \right\rangle_{\Gamma} \right\} dt$$

$$= \int_I \left(\mathbf{f}, \boldsymbol{\varphi}^k \right)_{\Omega} dt.$$

Moreover, we introduce a new norm

$$|||\mathbf{u}|||_G := \left(\boldsymbol{\nu}^2 \, ||\nabla \mathbf{u}||_G^2 + \frac{\gamma}{h} \, ||\mathbf{u}_2 - \mathbf{u}_1||_\Gamma^2 \right)^{\frac{1}{2}},$$

where G can be substituted with Ω_1, Ω_2, or Ω and use the appropriate component of $\boldsymbol{\nu}$. Let us proceed with the error estimation for the fully discrete case.

Theorem 3. *Let $\boldsymbol{u} \in X(H_0^1(\Omega))^d$, $\boldsymbol{u}_j \in W^{1,\infty}(H^{r+1}(\Omega_j))^d$ for $j = 1, 2$ be continuous solutions to (5.19) and $\boldsymbol{u}^k \in (X^k)^d$ their discrete counterpart and a solution to (5.33), then the following estimate holds*

$$\left\| \boldsymbol{e}^{k,h}(t^N) \right\|_\Omega^2 + \int_I \left\| \left\| \boldsymbol{e}^{k,h} \right\| \right\|_\Omega^2 \mathrm{d}t$$

$$\leqslant C \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \left\{ (k_1^{n,m})^3 \max_{t \in I} || \boldsymbol{d}_t \nabla \boldsymbol{u}_1 ||_{\Omega_1}^2 + (k_1^{n,m})^3 h \max_{t \in I} || \boldsymbol{d}_t \partial_{\vec{\boldsymbol{n}}_1} \boldsymbol{u}_1 ||_\Gamma^2 \right.$$

$$\left. + k_1^{n,m} h^{2r+2} \max_{t \in I} \left\| \boldsymbol{d}_t \nabla^{r+1} \boldsymbol{u}_1 \right\|_{\Omega_1}^2 + k_1^{n,m} h^{2r} \left\| \nabla^{r+1} \boldsymbol{u}_1(t_1^{n,m}) \right\|_{\Omega_1}^2 \right\}$$

$$+ C \sum_{n=1}^{N} \sum_{m=1}^{N_2^n} \left\{ (k_2^{n,m})^3 \max_{t \in I} || \boldsymbol{d}_t \nabla \boldsymbol{u}_2 ||_{\Omega_2}^2 + (k_2^{n,m})^3 h \max_{t \in I} || \boldsymbol{d}_t \partial_{\vec{\boldsymbol{n}}_2} \boldsymbol{u}_2 ||_\Gamma^2 \right.$$

$$\left. + k_2^{n,m} h^{2r+2} \max_{t \in I} \left\| \boldsymbol{d}_t \nabla^{r+1} \boldsymbol{u}_2 \right\|_{\Omega_2}^2 + k_2^{n,m} h^{2r} \left\| \nabla^{r+1} \boldsymbol{u}_2(t_2^{n,m}) \right\|_{\Omega_2}^2 \right\},$$

where the errors $\boldsymbol{e}^{k,h} = (\boldsymbol{e}_1^{k,h}, \boldsymbol{e}_2^{k,h})^T$ are defined as

$$\boldsymbol{e}_1^{k,h} := \boldsymbol{u}_1^{k,h} - \boldsymbol{i}_1^k \boldsymbol{R}_1^h \boldsymbol{u}_1, \qquad \boldsymbol{e}_2^{k,h} := \boldsymbol{u}_2^{k,h} - \boldsymbol{i}_2^k \boldsymbol{R}_2^h \boldsymbol{u}_2.$$

Proof. Here we also start with the Galerkin orthogonality

$$a^k(\mathbf{u}^{k,h}, \mathbf{e}^{k,h}) - a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}) = a(\mathbf{u}, \mathbf{e}^{k,h}) - a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}). \tag{5.34}$$

On the left side of this equation, using the symmetry of the interface terms, we have

$$a^k(\mathbf{u}^{k,h}, \mathbf{e}^{k,h}) - a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}) = \int_I \left\{ (\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h})_\Omega + |||\mathbf{e}^{k,h}|||_\Omega \right.$$

$$- \frac{1}{2} \langle \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{e}_2^{k,h} - \mathbf{I}_1^k \mathbf{e}_2^{k,h}), \mathbf{e}_1^{k,h} \rangle_\Gamma + \frac{1}{2} \langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{I}_2^k \mathbf{e}_1^{k,h} - \mathbf{e}_1^{k,h}), \mathbf{e}_2^{k,h} \rangle_\Gamma$$

$$- \frac{1}{2} \langle \mathbf{I}_1^k \mathbf{e}_2^{k,h} - \mathbf{e}_2^{k,h}, \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^{k,h} \rangle_\Gamma + \frac{1}{2} \langle \mathbf{e}_1^{k,h} - \mathbf{I}_2^k \mathbf{e}_1^{k,h}, \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^{k,h} \rangle_\Gamma$$

$$- \frac{\gamma}{h} \langle \mathbf{I}_1^k \mathbf{e}_2^{k,h} - \mathbf{e}_2^{k,h}, \mathbf{e}_1^{k,h} \rangle_\Gamma + \frac{\gamma}{h} \langle \mathbf{e}_1^{k,h} - \mathbf{I}_2^k \mathbf{e}_1^{k,h}, \mathbf{e}_2^{k,h} \rangle_\Gamma \right\} \mathrm{d}t.$$

Carrying out identical reasoning as in the previous proof leads us to

$$a^k(\mathbf{u}^{k,h}, \mathbf{e}^{k,h}) - a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}) = \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h} \right)_\Omega + |||\mathbf{e}^{k,h}|||_\Omega \right\} \mathrm{d}t.$$

To explain the disappearance of the interface terms, we refer to the equation (5.25). We continue by analyzing the right side of the orthogonality identity (5.34). Here we already omit the unnecessary interface terms including the projection operators

$$
\begin{aligned}
a(\mathbf{u}, \mathbf{e}^{k,h}) &- a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}) \\
&= \int_I \Big\{ \left(\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h} \right)_\Omega + \nu \left(\nabla(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}), \nabla \mathbf{e}^{k,h} \right)_\Omega \\
&\quad + \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2), \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \qquad (5.35) \\
&\quad - \frac{1}{2} \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \nu_1 \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^{k,h} - \nu_2 \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^{k,h} \right\rangle_\Gamma \\
&\quad + \frac{\gamma}{h} \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \Big\} \mathrm{d}t.
\end{aligned}
$$

We will again estimate these terms by taking $\mathbf{u}_1$ over the interval $I_1^{n,m}$. Starting with the time discretization error, it can be split between the contributions coming from the time and space meshes

$$\int_I \left\{ \left(\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h} \right)_\Omega \right\} \mathrm{d}t = \int_I \left\{ \left(\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^{k,h} \right)_\Omega + \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h} \right)_\Omega \right\} \mathrm{d}t.$$

The first term we already examined in the previous proof and based on (5.26) is equal to zero. Estimation of the second term directly follows from property (i) in Corollary 1 with the help of the Young and Poincaré inequalities.

$$
\begin{aligned}
\Bigg| \int_{I_1^{n,m}} &\left(\mathbf{d}_t^k i_1^k \mathbf{u}_1 - \mathbf{d}_t^k i_1^k \mathbf{R}_1^h \mathbf{u}_1, \mathbf{e}_1^{k,h} \right)_{\Omega_1} \mathrm{d}t \Bigg| \\
&\leqslant k_1^{n,m} \max_{t \in I} \left\| \mathbf{d}_t(\mathbf{u}_1 - \mathbf{R}_1^h \mathbf{u}_1) \right\|_{\Omega_1} \left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) \right\|_{\Omega_1} \qquad (5.36) \\
&\leqslant c k_1^{n,m} h^{2r+2} \max_{t \in I} \left\| \mathbf{d}_t \nabla^{r+1} \mathbf{u}_1 \right\|_{\Omega_1}^2 + \frac{1}{8} \int_{I_1^{n,m}} \nu_1^2 \left\| \nabla \mathbf{e}_1^{k,h} \right\|_{\Omega_1}^2 \mathrm{d}t.
\end{aligned}
$$

The Laplacian term in (5.35) can be split similarly

$$
\begin{aligned}
\int_I \nu &\left(\nabla(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}), \nabla \mathbf{e}^{k,h} \right)_\Omega \mathrm{d}t \\
&= \int_I \left\{ \nu \left(\nabla(\mathbf{u} - \mathbf{i}^k \mathbf{u}), \nabla \mathbf{e}^{k,h} \right)_\Omega + \nu \left(\nabla(\mathbf{i}^k \mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}), \nabla \mathbf{e}^{k,h} \right)_\Omega \right\} \mathrm{d}t.
\end{aligned}
$$

The first term was estimated in the previous proof by (5.27). The second term in the identity above can be also estimated by relying on property property (ii) in Corollary 1

$$\left| \int_{I_1^{n,m}} \nu_1 \left(\nabla (\mathbf{i}_1^k \mathbf{u}_1 - \mathbf{i}_1^k \mathbf{R}_1^h \mathbf{u}_1), \nabla \mathbf{e}_1^{k,h} \right)_{\Omega_1} \mathrm{d}t \right|$$
$$\leqslant c k_1^{n,m} h^{2r} \left\| \nabla^{r+1} \mathbf{u}_1(t_1^{n,m}) \right\|_{\Omega_1}^2 + \frac{1}{8} \int_{I_1^{n,m}} \nu_1^2 \left\| \nabla \mathbf{e}_1^{k,h} \right\|_{\Omega_1}^2 \mathrm{d}t. \tag{5.37}$$

We can proceed to the coupling conditions in (5.35). We estimate the first one on each macro time-step I^n

$$\left| \int_{I^n} \frac{1}{2} \left\langle \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2), \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t \right|$$
$$\leqslant \int_{I^n} \left\{ \frac{h}{2\gamma} \left\| \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) \right\|_\Gamma^2 + \frac{\gamma}{8h} \left\| \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\|_\Gamma^2 \right\} \mathrm{d}t.$$

The normal derivatives in (5.35) can be estimated by splitting the errors similarly

$$\int_{I^n} \left\| \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - I_1^k \mathbf{R}_1^h \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - I_2^k \mathbf{R}_2^h \mathbf{u}_2) \right\|_\Gamma^2 \mathrm{d}t$$
$$\leqslant \sum_{m=1}^{N_1^n} c_1 k_1^{n,m} (\nu_1)^2 \left\{ (k_1^{n,m})^2 \max_{t \in I} \| \mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1 \|_\Gamma^2 + h^{2r-1} \left\| \nabla^{r+1} \mathbf{u}_1(t_1^{n,m}) \right\|_{\Omega_1}^2 \right\} \tag{5.38}$$
$$+ \sum_{m=1}^{N_2^n} c_2 k_2^{n,m} (\nu_2)^2 \left\{ (k_2^{n,m})^2 \max_{t \in I} \| \mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2 \|_\Gamma^2 + h^{2r-1} \left\| \nabla^{r+1} \mathbf{u}_2(t_2^{n,m}) \right\|_{\Omega_2}^2 \right\}.$$

Estimation of the space component follows from property (iii) in Corollary 1. The time component was estimated in (5.30). The analysis of the remaining two interface terms is very similar. Since the last one is slightly simpler, we will take it as an example. We have, given the continuity of $\mathbf{u}$

$$\int_{I^n} \frac{\gamma}{h} \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t$$
$$= \int_{I^n} \frac{\gamma}{h} \left\langle \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t.$$

Using an identical set of arguments as in the previous proof, we can show that

$$\int_{I^n} \frac{\gamma}{h} \left\langle \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t = \int_{I^n} \frac{\gamma}{h} \left\langle \mathbf{R}_1^h \mathbf{u}_1 - \mathbf{R}_2^h \mathbf{u}_2, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t.$$

Finally, given the continuity of the Ritz operator (5.32), it holds

$$\int_{I^n} \frac{\gamma}{h} \left\langle \mathbf{R}_1^h \mathbf{u}_1 - \mathbf{R}_2^h \mathbf{u}_2, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \mathrm{d}t = \mathbf{0}. \tag{5.39}$$

The last interface term in (5.35) can be estimated similarly. That ends the proof. $\qquad \square$

The theorem is equivalent to

$$
\begin{aligned}
\left\|\mathbf{e}^{k,h}(t^N)\right\| + \int_I \left\|\left\|\mathbf{e}^k\right\|\right\| \mathrm{d}t & \\
&= \mathcal{O}\big(k_1 \left\|\mathbf{d}_t \nabla \mathbf{u}_1\right\|_{\Omega_1}\big) + \mathcal{O}\big(k_1 h^{\frac{1}{2}} \left\|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1\right\|_\Gamma\big) + \mathcal{O}\big(h^{r+1} \left\|\mathbf{d}_t \nabla^{r+1} \mathbf{u}_1\right\|_{\Omega_1}\big) \\
&\quad + \mathcal{O}\big(h^r \left\|\nabla^{r+1} \mathbf{u}_1\right\|_{\Omega_1}\big) + \mathcal{O}\big(k_2 \left\|\mathbf{d}_t \nabla \mathbf{u}_2\right\|_{\Omega_2}\big) + \mathcal{O}\big(k_2 h^{\frac{1}{2}} \left\|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2\right\|_\Gamma\big) \\
&\quad + \mathcal{O}\big(h^{r+1} \left\|\mathbf{d}_t \nabla^{r+1} \mathbf{u}_2\right\|_{\Omega_2}\big) + \mathcal{O}\big(h^r \left\|\nabla^{r+1} \mathbf{u}_2\right\|_{\Omega_2}\big).
\end{aligned}
$$

Also here, we were able to fully decouple this system. In fact, we were able to equip the the terms $\left\|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1\right\|_\Gamma$ and $\left\|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2\right\|_\Gamma$ with an additional half an order of convergence in space compared to the semi-discrete case. That being said, a comprehensive comparison between the fully discrete and semi-discrete cases is not possible since both of the inequalities are proved in different norms. Overall, we preserved the linear convergence in time.

5.4 Coupling of Stokes Equations

As our third and final problem, we consider a system composed of two time-dependent Stokes equations. Each of them has a separate kinematic viscosity $(\nu_1, \nu_2)^T = \boldsymbol{\nu}$. Velocity $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2)^T : \Omega \times I \to \mathbb{R}^d$, $\mathbf{u}_j \in X(H^1(\Omega_j))^d$ and pressure $\mathbf{p} = (p_1, p_2)^T : \Omega \times I \to \mathbb{R}$, $p_j \in X(L^2(\Omega_j))$ for $j = 1, 2$ are solutions to the system

$$
\begin{aligned}
\partial_t \mathbf{u}_1 - 2\nu_1 \mathrm{div}\ \dot{\boldsymbol{\epsilon}}(\mathbf{u}_1) + \nabla p_1 &= \mathbf{f}_1 & &\text{in } \Omega_1, \\
\mathrm{div}\ \mathbf{u}_1 &= 0 & &\text{in } \Omega_1, \\
\partial_t \mathbf{u}_2 - 2\nu_2 \mathrm{div}\ \dot{\boldsymbol{\epsilon}}(\mathbf{u}_2) + \nabla p_2 &= \mathbf{f}_2 & &\text{in } \Omega_2, \\
\mathrm{div}\ \mathbf{u}_2 &= 0 & &\text{in } \Omega_2,
\end{aligned}
$$

where

$$
\dot{\boldsymbol{\epsilon}}(\mathbf{u}) = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right).
$$

On the outer boundary we set $\mathbf{u}_1 = \mathbf{u}_2 = \mathbf{0}$. Also at the initial time, we impose $\mathbf{u}_1(0) = \mathbf{u}_2(0) = \mathbf{0}$. On the interface we set

$$
\begin{aligned}
\mathbf{u}_1 &= \mathbf{u}_2 & &\text{on } \Gamma, \\
\sigma_1(\mathbf{u}_1, p_1) \cdot \vec{\mathbf{n}}_1 &= -\sigma_2(\mathbf{u}_2, p_2) \cdot \vec{\mathbf{n}}_2 & &\text{on } \Gamma,
\end{aligned}
$$

where the stress tensors $\boldsymbol{\sigma} = (\sigma_1, \sigma_2)^T$ are given by

$$
\sigma_1(\mathbf{u}_1, p_1) = 2\nu_1 \dot{\boldsymbol{\epsilon}}(\mathbf{u}_1) - p_1 I, \qquad \sigma_2(\mathbf{u}_2, p_2) = 2\nu_2 \dot{\boldsymbol{\epsilon}}(\mathbf{u}_2) - p_2 I.
$$

As test functions, we take $\boldsymbol{\varphi} = (\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2)^T$, $\boldsymbol{\varphi}_j \in X(H^1(\Omega_j))^d$ and $\boldsymbol{\psi} = (\psi_1, \psi_2)^T$, $\psi_j \in X(L^2(\Omega_j))$ for $j = 1, 2$. The incompressibility form is defined as

$$
b(\mathbf{u}, \boldsymbol{\psi}) = \int_I \left\{ -(\mathrm{div}\ \mathbf{u}, \boldsymbol{\psi})_\Omega + \frac{1}{2}\langle \psi_2 \vec{\mathbf{n}}_2 - \psi_1 \vec{\mathbf{n}}_1, \mathbf{u}_2 - \mathbf{u}_1\rangle_\Gamma \right\} \mathrm{d}t. \tag{5.40}
$$

In this formulation, we again follow the work presented in [5]. We note that the additional interface term does not violate the incompressibility condition since the exact solution is continuous across the interface and therefore

$$\langle \psi_2 \vec{\mathbf{n}}_2 - \psi_1 \vec{\mathbf{n}}_1, \mathbf{u}_2 - \mathbf{u}_1 \rangle_\Gamma = 0.$$

Given that, the variational problem is defined as

$$a(\mathbf{u}, \boldsymbol{\varphi}) + b(\boldsymbol{\varphi}, \mathbf{p}) - b(\mathbf{u}, \boldsymbol{\psi}) = \int_I (\mathbf{f}, \boldsymbol{\varphi})_\Omega \, \mathrm{d}t, \tag{5.41}$$

where form $a(\cdot, \cdot)$ has a similar definition as in the case of the heat equation

$$a(\mathbf{u}, \boldsymbol{\varphi}) := \int_I \Big\{ (d_t \mathbf{u}, \boldsymbol{\varphi})_\Omega + 2\boldsymbol{\nu} \, (\dot{\boldsymbol{\epsilon}}(\mathbf{u}), \nabla \boldsymbol{\varphi})_\Omega + \langle \nu_1 \dot{\epsilon}(\mathbf{u}_1) \cdot \vec{\mathbf{n}}_1 - \nu_2 \dot{\epsilon}(\mathbf{u}_2) \cdot \vec{\mathbf{n}}_2, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma$$
$$- \langle \mathbf{u}_2 - \mathbf{u}_1, \nu_1 \dot{\epsilon}(\boldsymbol{\varphi}_1) \cdot \vec{\mathbf{n}}_1 - \nu_2 \dot{\epsilon}(\boldsymbol{\varphi}_2) \cdot \vec{\mathbf{n}}_2 \rangle_\Gamma + \gamma \langle \mathbf{u}_2 - \mathbf{u}_1, \boldsymbol{\varphi}_2 - \boldsymbol{\varphi}_1 \rangle_\Gamma \Big\} \, \mathrm{d}t.$$

5.4.1 Semi-discrete Case

Similarly, one can construct a discrete incompressibility form. We have to pay attention to the position of the trial and test functions. Therefore we define two versions of this form

$$\bar{b}^k(\mathbf{u}^k, \boldsymbol{\psi}^k) := \int_I \Big\{ -\big(\mathrm{div}\ \mathbf{u}^k, \boldsymbol{\psi}^k \big)_\Omega - \frac{1}{2} \Big\langle \vec{\mathbf{n}}_1 \psi_1^k, \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k \Big\rangle_\Gamma$$
$$+ \frac{1}{2} \Big\langle \vec{\mathbf{n}}_2 \psi_2^k, \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k \Big\rangle_\Gamma \Big\} \, \mathrm{d}t,$$

$$b^k(\boldsymbol{\varphi}^k, \mathbf{p}^k) := \int_I \Big\{ -\big(\mathrm{div}\ \boldsymbol{\varphi}^k, \mathbf{p}^k \big)_\Omega - \frac{1}{2} \Big\langle \vec{\mathbf{n}}_2 I_1^k p_2^k - \vec{\mathbf{n}}_1 p_1^k, \boldsymbol{\varphi}_1^k \Big\rangle_\Gamma$$
$$+ \frac{1}{2} \Big\langle \vec{\mathbf{n}}_2 p_2^k - \vec{\mathbf{n}}_1 I_2^k p_1, \boldsymbol{\varphi}_2^k \Big\rangle_\Gamma \Big\} \, \mathrm{d}t. \tag{5.42}$$

These forms together with the form

$$a^k(\mathbf{u}^k, \boldsymbol{\varphi}^k) := \int_I \Big\{ \big(d_t^k \mathbf{u}^k, \boldsymbol{\varphi}^k \big)_\Omega + 2\boldsymbol{\nu} \, \big(\dot{\boldsymbol{\epsilon}}(\mathbf{u}^k), \nabla \boldsymbol{\varphi}^k \big)_\Omega - \Big\langle \nu_1 \dot{\epsilon}(\mathbf{u}_1^k) \cdot \vec{\mathbf{n}}_1 - \nu_2 \dot{\epsilon}(\mathbf{I}_1^k \mathbf{u}_2^k) \cdot \vec{\mathbf{n}}_2, \boldsymbol{\varphi}_1^k \Big\rangle_\Gamma$$
$$+ \Big\langle \nu_1 \dot{\epsilon}(\mathbf{I}_2^k \mathbf{u}_1^k) \cdot \vec{\mathbf{n}}_1 - \nu_2 \dot{\epsilon}(\mathbf{u}_2^k) \cdot \vec{\mathbf{n}}_2, \boldsymbol{\varphi}_2^k \Big\rangle_\Gamma - \Big\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k, \nu_1 \dot{\epsilon}(\boldsymbol{\varphi}_1^k) \cdot \vec{\mathbf{n}}_1 \Big\rangle_\Gamma$$
$$+ \Big\langle \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k, \nu_2 \dot{\epsilon}(\boldsymbol{\varphi}_2^k) \cdot \vec{\mathbf{n}}_2 \Big\rangle_\Gamma - \gamma \Big\langle \mathbf{I}_1^k \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_1^k \Big\rangle_\Gamma$$
$$+ \gamma \Big\langle \mathbf{u}_2^k - \mathbf{I}_2^k \mathbf{u}_1^k, \boldsymbol{\varphi}_2^k \Big\rangle_\Gamma \Big\} \, \mathrm{d}t.$$

define the semi-discrete variational problem

$$a^k(\mathbf{u}^k, \boldsymbol{\varphi}^k) + b^k(\boldsymbol{\varphi}^k, \mathbf{p}^k) - \bar{b}^k(\mathbf{u}^k, \boldsymbol{\psi}^k) = \int_I (\mathbf{f}, \boldsymbol{\varphi}^k)_\Omega \, \mathrm{d}t. \tag{5.43}$$

From this formulation, we can derive semi-discrete coupling conditions

$$0 = \int_{I_n} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t = \int_{I_n} \left\langle \mathbf{u}_2^k - \mathbf{u}_1^k, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t,$$

$$0 = \int_{I_n} \left\langle \sigma_1(\mathbf{u}_1, p_1) \cdot \vec{\mathbf{n}}_1 + \sigma_2(\mathbf{u}_2, p_2) \cdot \vec{\mathbf{n}}_2, \boldsymbol{\varphi}_1^k \right\rangle_\Gamma \mathrm{d}t \tag{5.44}$$

$$= \int_{I_n} \left\langle \sigma_1(\mathbf{u}_1, p_1) \cdot \vec{\mathbf{n}}_1 + \sigma_2(\mathbf{u}_2, p_2) \cdot \vec{\mathbf{n}}_2, \boldsymbol{\varphi}_2^k \right\rangle_\Gamma \mathrm{d}t$$

for any $\boldsymbol{\varphi}_1^k \in (X_1^k)^d$ and $\boldsymbol{\varphi}_2^k \in (X_2^k)^d$. As it turns out, the theorem from the previous sections can be easily extended to the Stokes equation as well

Theorem 4. *Let* $\mathbf{u} \in X(H_0^1(\Omega))^d$, $\mathbf{u}_j \in W^{1,\infty}(H^2(\Omega_j))^d$, $p_j \in W^{1,\infty}(L^2(\Omega_j))$ *for* $j = 1, 2$ *be continuous solutions to (5.41) and* $\mathbf{u}^k \times \mathbf{p}^k \in \left(X^k\right)^{d+1}$ *their semi-discrete counterparts and solutions to (5.43), then the following estimate holds*

$$\left\| \boldsymbol{e}^k(t^N) \right\|_\Omega^2 + \int_I \boldsymbol{\nu}^2 \left\| \nabla \boldsymbol{e}^k \right\|_\Omega^2 \mathrm{d}t$$

$$\leqslant C \sum_{n=1}^{N} \sum_{m=1}^{N_1^n} \left\{ (k_1^{n,m})^3 \max_{t \in I} \| \boldsymbol{d}_t \dot{\epsilon}(\boldsymbol{u}_1) \|_{\Omega_1}^2 + (k_1^{n,m})^3 \max_{t \in I} \| d_t p_1 \|_{\Omega_1}^2 \right.$$

$$\left. + (k_1^{n,m})^3 \max_{t \in I} \| \boldsymbol{d}_t \sigma_1(\boldsymbol{u}_1, p_1) \cdot \vec{\boldsymbol{n}}_1 \|_\Gamma^2 \right\}$$

$$+ C \sum_{n=1}^{N} \sum_{m=1}^{N_2^n} \left\{ (k_2^{n,m})^3 \max_{t \in I} \| \boldsymbol{d}_t \dot{\epsilon}(\boldsymbol{u}_2) \|_{\Omega_2}^2 + (k_2^{n,m})^3 \max_{t \in I} \| d_t p_2 \|_{\Omega_2}^2 \right.$$

$$\left. + (k_2^{n,m})^3 \max_{t \in I} \| \boldsymbol{d}_t \sigma_2(\boldsymbol{u}_2, p_2) \cdot \vec{\boldsymbol{n}}_2 \|_\Gamma^2 \right\},$$

where the errors $\boldsymbol{e}^k = (\boldsymbol{e}_1^k, \boldsymbol{e}_2^k)^T$ *are defined as*

$$\boldsymbol{e}_1^k := \boldsymbol{u}_1^k - \boldsymbol{i}_1^k \boldsymbol{u}_1, \qquad \boldsymbol{e}_2^k := \boldsymbol{u}_2^k - \boldsymbol{i}_2^k \boldsymbol{u}_2.$$

Proof. By again using Galerkin orthogonality

$$a^k(\mathbf{u}^k, \mathbf{e}^k) + b^k(\mathbf{e}^k, \mathbf{p}^k) - \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k \right)_\Omega + 2\boldsymbol{\nu} \left(\dot{\epsilon}(\mathbf{i}^k \mathbf{u}), \nabla \mathbf{e}^k \right)_\Omega \right\} \mathrm{d}t$$

$$= a(\mathbf{u}, \mathbf{e}^k) + b(\mathbf{e}^k, \mathbf{p}) - \int_I \left\{ \left(\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k \right)_\Omega + 2\boldsymbol{\nu} \left(\dot{\epsilon}(\mathbf{i}^k \mathbf{u}), \nabla \mathbf{e}^k \right)_\Omega \right\} \mathrm{d}t. \tag{5.45}$$

For any $\boldsymbol{\psi}^k \in X^k$, we have

$$\int_I \left(\mathrm{div}\, \mathbf{u}, \boldsymbol{\psi}^k \right)_\Omega \mathrm{d}t = \int_I \left(\mathrm{div}\, \mathbf{u}^k, \boldsymbol{\psi}^k \right)_\Omega \mathrm{d}t = 0.$$

Knowing that the semi-discrete pressure is piecewise constant in time, we can claim that

$$\int_I \left(\operatorname{div} \mathbf{e}^k, \mathbf{p}^k \right)_\Omega dt = \int_I \left\{ \left(\operatorname{div} \mathbf{u}^k, \mathbf{p}^k \right)_\Omega - \left(\operatorname{div} \mathbf{i}^k \mathbf{u}, \mathbf{p}^k \right)_\Omega \right\} dt$$
$$= \int_I \left\{ \left(\operatorname{div} \mathbf{u}^k, \mathbf{p}^k \right)_\Omega - \mathbf{i}^k \left(\operatorname{div} \mathbf{u}, \mathbf{p}^k \right)_\Omega \right\} dt = 0. \tag{5.46}$$

Therefore, on the left side of (5.45), we have

$$a^k(\mathbf{u}^k, \mathbf{e}^k) + b^k(\mathbf{e}^k, \mathbf{p}^k) - \int_I \left\{ (\mathbf{d}_t^k \mathbf{i}^k \mathbf{u}, \mathbf{e}^k)_\Omega + 2\boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{i}^k \mathbf{u}), \nabla \mathbf{e}^k \right)_\Omega \right\} dt$$
$$= \int_I (\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h})_\Omega dt + \int_I 2\boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), \nabla \mathbf{e}^{k,h} \right)_\Omega dt.$$

We can similarly show that

$$\int_I \left(\operatorname{div} \mathbf{e}^k, \mathbf{p} \right)_\Omega dt = \int_I \left(\operatorname{div} \mathbf{e}^k, \mathbf{p} - \mathbf{i}^k \mathbf{p} \right)_\Omega dt.$$

Indeed, it holds

$$\int_I \left(\operatorname{div} \mathbf{e}^k, \mathbf{i}^k \mathbf{p} \right)_\Omega dt = \int_I \left\{ \left(\operatorname{div} \mathbf{u}^k, \mathbf{i}^k \mathbf{p}^k \right)_\Omega - \left(\operatorname{div} \mathbf{i}^k \mathbf{u}, \mathbf{i}^k \mathbf{p}^k \right)_\Omega \right\} dt$$
$$= -\int_I \mathbf{i}^k \left(\operatorname{div} \mathbf{u}, \mathbf{i}^k \mathbf{p}^k \right)_\Omega dt = 0.$$

Further, on each interval $I_1^{n,m}$ we have

$$\left| \int_{I_1^{n,m}} \left(\operatorname{div} \mathbf{e}_1^k, p_1 - i_1^k p_1 \right)_{\Omega_1} dt \right| \leqslant c(k_1^{n,m})^3 \max_{t\in I} \|d_t p_1\|_{\Omega_1}^2 + \frac{1}{8}\nu_1 \int_{I_1^{n,m}} \left\| \nabla \mathbf{e}_1^k \right\|_{\Omega_1}^2 dt.$$

The Laplacian term in (5.45) can be symmetrized and using the first Korn inequality leads to

$$\int_I 2\boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), \nabla \mathbf{e}^{k,h} \right)_\Omega dt$$
$$= \int_I \boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), \nabla \mathbf{e}^{k,h} \right)_\Omega dt + \int_I \boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), \nabla \mathbf{e}^{k,h} \right)_\Omega dt$$
$$= \int_I \boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), \nabla \mathbf{e}^{k,h} \right)_\Omega dt + \int_I \boldsymbol{\nu}\left(\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h}), (\nabla \mathbf{e}^{k,h})^T \right)_\Omega dt$$
$$= \int_I 2\boldsymbol{\nu}\|\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h})\|_\Omega^2 dt \geqslant c_K \int_I \boldsymbol{\nu}\|\nabla \mathbf{e}^{k,h}\|_\Omega^2 dt.$$

c_K denotes the constant from Korn inequality. The rest directly follows from Theorem 2 simply by using the appropriate Neumann coupling conditions on the interface (5.30) and replacing $\nabla \mathbf{e}^{k,h}$ with $\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h})$ in (5.27). $\qquad\square$

We just showed that

$$
\begin{aligned}
||\mathbf{e}^k(t^N)|| + \int_I &\boldsymbol{\nu}||\mathbf{e}^k||\,\mathrm{d}t \\
&= \mathcal{O}\big(k_1\,||\mathbf{d}_t\dot{\epsilon}(\mathbf{u}_1)||_{\Omega_1}\big) + \mathcal{O}\big(k_1\,||d_tp_1||_{\Omega_1}\big) + \mathcal{O}\big(k_1\,||\mathbf{d}_t\sigma_1(\mathbf{u}_1,p_1)\cdot\vec{\mathbf{n}}_1||_\Gamma\big) \\
&+ \mathcal{O}\big(k_2\,||\mathbf{d}_t\dot{\epsilon}(\mathbf{u}_2)||_{\Omega_2}\big) + \mathcal{O}\big(k_2\,||d_tp_2||_{\Omega_2}\big) + \mathcal{O}\big(k_2\,||\mathbf{d}_t\sigma_2(\mathbf{u}_2,p_2)\cdot\vec{\mathbf{n}}_2||_\Gamma\big).
\end{aligned}
$$

These results are analogous to what we were able to show in Theorem 2. The differences include replacing $\nabla\mathbf{e}^{k,h}$ with $\dot{\boldsymbol{\epsilon}}(\mathbf{e}^{k,h})$ and adding new pressure terms. The newly introduced volume pressure terms are also decoupled. We managed to preserve the optimal linear convergence rate.

5.4.2 Fully Discrete Case

One can establish a similar estimate for the fully discrete coupled Stokes equations. Since the main focus of this paper is time discretization, we consider classical inf-sup stable Taylor-Hood elements, where $\mathbf{u}^{k,h} \times \mathbf{p}^{k,h} \in \big(X^{k,h}(r)\big)^d \times X^{k,h}(r-1)$ for $r \geqslant 2$. The fully discrete variational formulation reads as

$$
a^{k,h}(\mathbf{u}^{k,h},\boldsymbol{\varphi}^{k,h}) + b^k(\boldsymbol{\varphi}^{k,h},\mathbf{p}^{k,h}) - \bar{b}^k(\mathbf{u}^{k,h},\boldsymbol{\psi}^{k,h}) = \int_I \big(\mathbf{f},\boldsymbol{\varphi}^{k,h}\big)_\Omega\,\mathrm{d}t, \tag{5.47}
$$

where

$$
\begin{aligned}
a^{k,h}(\mathbf{u}^{k,h},\boldsymbol{\varphi}^{k,h}) := \int_I \Big\{ &\big(\mathbf{d}_t^k\mathbf{u}^{k,h},\boldsymbol{\varphi}^{k,h}\big)_\Omega + 2\boldsymbol{\nu}\Big(\dot{\boldsymbol{\epsilon}}(\mathbf{u}^{k,h}),\nabla\boldsymbol{\varphi}^{k,h}\Big)_\Omega \\
&- \Big\langle \nu_1\dot{\epsilon}(\mathbf{u}_1^{k,h})\cdot\vec{\mathbf{n}}_1 - \nu_2\dot{\epsilon}(\mathbf{I}_1^k\mathbf{u}_2^{k,h})\cdot\vec{\mathbf{n}}_2, \boldsymbol{\varphi}_1^{k,h}\Big\rangle_\Gamma \\
&+ \Big\langle \nu_1\dot{\epsilon}(\mathbf{I}_2^k\mathbf{u}_1^{k,h})\cdot\vec{\mathbf{n}}_1 - \nu_2\dot{\epsilon}(\mathbf{u}_2^{k,h})\cdot\vec{\mathbf{n}}_2, \boldsymbol{\varphi}_2^{k,h}\Big\rangle_\Gamma \\
&- \Big\langle \mathbf{I}_1^k\mathbf{u}_2^{k,h} - \mathbf{u}_1^{k,h}, \nu_1\dot{\epsilon}(\boldsymbol{\varphi}_1^{k,h})\cdot\vec{\mathbf{n}}_1\Big\rangle_\Gamma + \Big\langle \mathbf{u}_2^{k,h} - \mathbf{I}_2^k\mathbf{u}_1^{k,h}, \nu_2\dot{\epsilon}(\boldsymbol{\varphi}_2^{k,h})\cdot\vec{\mathbf{n}}_2\Big\rangle_\Gamma \\
&- \frac{\gamma}{h}\Big\langle \mathbf{I}_1^k\mathbf{u}_2^{k,h} - \mathbf{u}_1^{k,h}, \boldsymbol{\varphi}_1^{k,h}\Big\rangle_\Gamma + \frac{\gamma}{h}\Big\langle \mathbf{u}_2^{k,h} - \mathbf{I}_2^k\mathbf{u}_1^{k,h}, \boldsymbol{\varphi}_2^{k,h}\Big\rangle_\Gamma \Big\}\,\mathrm{d}t.
\end{aligned}
$$

For the Stokes problem, we will use a modified version of the Ritz projection operator (5.31) to account for the incompressibility condition

$$
\begin{aligned}
\big(\nabla\mathbf{R}^h\mathbf{u},\nabla\boldsymbol{\varphi}^{k,h}\big)_\Omega - \big(\mathbf{q},\mathrm{div}\,\boldsymbol{\varphi}^{k,h}\big)_\Omega &= \big(\nabla\mathbf{u},\nabla\boldsymbol{\varphi}^{k,h}\big)_\Omega, \quad \boldsymbol{\varphi}^{k,h}\in(Y^{k,h}(r))^d \\
\big(\mathrm{div}\,\mathbf{R}^h\mathbf{u},\boldsymbol{\psi}^{k,h}\big)_\Omega &= 0, \qquad\qquad\quad \boldsymbol{\psi}^{k,h}\in Y^{k,h}(r-1).
\end{aligned}
$$

All of the properties established in Corollary 1 still hold. The newly introduced pressure term $\mathbf{q}$ is only a Lagrange multiplier needed to project the solution $\mathbf{u}$ into the space of divergence-free functions. We will not come back to it in the proofs. However, for the pressure $\mathbf{p}$, we will use an additional projection operator $\mathbf{I}^h = (I_1^h, I_2^h)^T$ given by

$$
\big(\mathbf{I}^h\mathbf{p},\boldsymbol{\psi}^{k,h}\big)_\Omega = \big(\mathbf{p},\boldsymbol{\psi}^{k,h}\big)_\Omega \quad \text{for all} \quad \boldsymbol{\psi}^{k,h}\in X^{k,h}(r-1). \tag{5.48}
$$

Below are listed some of the useful properties.

118

Corollary 2. *Given $p_j \in L(\bar{I}, H^r(\Omega_j))$ for $j = 1, 2$, the projection operator given by (5.48) has the following properties:*

(i) $\left\| \boldsymbol{p} - \boldsymbol{I}^h \boldsymbol{p} \right\|_\Omega \leqslant c_1 h^r \left\| \nabla^r \boldsymbol{p}_1 \right\|_{\Omega_1} + c_2 h^r \left\| \nabla^r \boldsymbol{p}_2 \right\|_{\Omega_2},$

(ii) $\left\| (p_1 - I_1^h p_1) \vec{\boldsymbol{n}}_1 \right\|_\Gamma + \left\| (p_2 - I_2^h p_2) \vec{\boldsymbol{n}}_2 \right\|_\Gamma \leqslant c_1 h^{r-\frac{1}{2}} \left\| \nabla^r p_1 \right\|_{\Omega_1} + c_2 h^{r-\frac{1}{2}} \left\| \nabla^r p_2 \right\|_{\Omega_2}.$

We proceed to the velocity error estimation for the fully discrete problem.

Theorem 5. *Let $\boldsymbol{u} \in X(H_0^1(\Omega))^d$, $\boldsymbol{u}_j \in W^{1,\infty}(H^{r+1}(\Omega_j))^d$, $p_j \in W^{1,\infty}(H^r(\Omega_j))$ for $j = 1, 2$ be continuous solutions to (5.41) and $\boldsymbol{u}^{k,h} \times \boldsymbol{p}^{k,h} \in \left(X^{k,h}(r) \right)^d \times X^{k,h}(r-1)$ their discrete counterparts and solutions to (5.47), then the following estimate holds*

$$
\left\| \boldsymbol{e}^{k,h}(t^N) \right\|_\Omega^2 + \int_I \left\| \left\| \boldsymbol{e}^{k,h} \right\| \right\|_\Omega^2 \, \mathrm{d}t
$$

$$
\leqslant C \sum_{n=1}^N \sum_{m=1}^{N_1^n} \left\{ (k_1^{n,m})^3 \max_{t\in I} \| \boldsymbol{d}_t \dot{\epsilon}(\boldsymbol{u}_1) \|_{\Omega_1}^2 + (k_1^{n,m})^3 \max_{t\in I} \| d_t p_1 \|_{\Omega_1}^2 \right.
$$

$$
+ (k_1^{n,m})^3 h \max_{t\in I} \| \boldsymbol{d}_t \sigma_1(\boldsymbol{u}_1, p_1) \cdot \vec{\boldsymbol{n}}_1 \|_\Gamma^2 + k_1^{n,m} h^{2r+2} \max_{t\in I} \left\| \boldsymbol{d}_t \nabla^{r+1} \boldsymbol{u}_1 \right\|_{\Omega_1}^2
$$

$$
\left. + k_1^{n,m} h^{2r} \left\| \nabla^{r+1} \boldsymbol{u}_1(t_1^{n,m}) \right\|_{\Omega_1}^2 + k_1^{n,m} h^{2r} \left\| \nabla^r p_1(t_1^{n,m}) \right\|_{\Omega_1}^2 \right\}
$$

$$
+ C \sum_{n=1}^N \sum_{m=1}^{N_2^n} \left\{ (k_2^{n,m})^3 \max_{t\in I} \| \boldsymbol{d}_t \dot{\epsilon}(\boldsymbol{u}_2) \|_{\Omega_2}^2 + (k_2^{n,m})^3 \max_{t\in I} \| d_t p_2 \|_{\Omega_2}^2 \right.
$$

$$
+ (k_2^{n,m})^3 h \max_{t\in I} \| \boldsymbol{d}_t \sigma_2(\boldsymbol{u}_2, p_2) \cdot \vec{\boldsymbol{n}}_2 \|_\Gamma^2 + k_2^{n,m} h^{2r+2} \max_{t\in I} \left\| \boldsymbol{d}_t \nabla^{r+1} \boldsymbol{u}_2 \right\|_{\Omega_2}^2
$$

$$
\left. + k_2^{n,m} h^{2r} \left\| \nabla^{r+1} \boldsymbol{u}_2(t_2^{n,m}) \right\|_{\Omega_2}^2 + k_2^{n,m} h^{2r} \left\| \nabla^r p_2(t_2^{n,m}) \right\|_{\Omega_2}^2 \right\},
$$

where the errors $\boldsymbol{e}^k = (\boldsymbol{e}_1^k, \boldsymbol{e}_2^k)^T$, $\boldsymbol{\eta}^k = (\eta_1^k, \eta_2^k)^T$ are defined as

$$
\boldsymbol{e}_1^{k,h} := \boldsymbol{u}_1^{k,h} - \boldsymbol{i}_1^k \boldsymbol{R}_1^h \boldsymbol{u}_1, \qquad \boldsymbol{e}_2^{k,h} := \boldsymbol{u}_2^{k,h} - \boldsymbol{i}_2^k \boldsymbol{R}_2^h \boldsymbol{u}_2,
$$
$$
\eta_1^{k,h} := p_1^{k,h} - i_1^k I_1^h p_1, \qquad \eta_2^{k,h} := p_2^{k,h} - i_2^k I_2^h p_2.
$$

Proof. The Galerkin orthogonality gives us

$$
a^k(\mathbf{e}^{k,h}, \mathbf{e}^{k,h}) + b^k(\mathbf{e}^{k,h}, \boldsymbol{\eta}^{k,h}) - \bar{b}^k(\mathbf{e}^{k,h}, \boldsymbol{\eta}^{k,h})
$$
$$
= a(\mathbf{u}, \mathbf{e}^{k,h}) - a^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h}) + b(\mathbf{e}^{k,h}, \mathbf{p}) - b^k(\mathbf{e}^{k,h}, \mathbf{i}^k \mathbf{I}^h \mathbf{p}) \qquad (5.49)
$$
$$
- b(\mathbf{u}, \boldsymbol{\eta}^{k,h}) + \bar{b}^k(\mathbf{i}^k \mathbf{R}^h \mathbf{u}, \boldsymbol{\eta}^{k,h}).
$$

119

One can show that the left side is equal to

$$a^k(\mathbf{e}^{k,h}, \mathbf{e}^{k,h}) + b^k(\mathbf{e}^{k,h}, \boldsymbol{\eta}^{k,h}) - \bar{b}^k(\mathbf{e}^{k,h}, \boldsymbol{\eta}^{k,h}) = a^k(\mathbf{e}^{k,h}, \mathbf{e}^{k,h}).$$

On the right side of (5.49), we have

$$
\begin{aligned}
a^k(\mathbf{e}^{k,h}, \mathbf{e}^{k,h}) = \int_I \Big\{ &\left(\mathbf{d}_t \mathbf{u} - \mathbf{d}_t^k \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \mathbf{e}^{k,h} \right)_\Omega + 2\nu \left(\dot{\boldsymbol{\epsilon}}(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}), \nabla \mathbf{e}^{k,h} \right)_\Omega \\
&- \left(\operatorname{div} \mathbf{e}^{k,h}, \mathbf{p} - \mathbf{i}^k \mathbf{I}^h \mathbf{p} \right)_\Omega + \left(\operatorname{div}(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}), \boldsymbol{\eta}^{k,h} \right)_\Omega \\
&+ \left\langle \sigma_1(\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1, p_1 - I_1^k I_1^h p_1) \cdot \vec{\mathbf{n}}_1, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \\
&- \left\langle \sigma_2(\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2, p_2 - I_2^k I_2^h p_2) \cdot \vec{\mathbf{n}}_2, \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \\
&- \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \sigma_1(\mathbf{e}_1^{k,h}, \eta_1^{k,h}) \cdot \vec{\mathbf{n}}_1 \right\rangle_\Gamma \\
&+ \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \sigma_2(\mathbf{e}_2^{k,h}, \eta_2^{k,h}) \cdot \vec{\mathbf{n}}_2 \right\rangle_\Gamma \\
&+ \frac{\gamma}{h} \left\langle (\mathbf{u}_2 - \mathbf{I}_2^k \mathbf{R}_2^h \mathbf{u}_2) - (\mathbf{u}_1 - \mathbf{I}_1^k \mathbf{R}_1^h \mathbf{u}_1), \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\rangle_\Gamma \Big\} \, \mathrm{d}t.
\end{aligned}
$$

Most of these terms were already estimated in previous proofs. We dealt with the time contributions of the time derivative (5.26), the Laplacian terms (5.27), and normal derivatives (5.30) in Theorem 2. We looked at the space components in (5.36) and (5.37) in Theorem 3. The interface terms were considered in equations (5.38) and (5.39). In (5.38) we have to additionally account for the interpolation in space of the pressure

$$\int_{I_1^{n,m}} h \| (i_1^k p_1 - i_1^k I_1^h p_1) \cdot \vec{\mathbf{n}}_1 \|_\Gamma^2 \, \mathrm{d}t \leqslant k_1^{n,m} h^{2r} \| \nabla^r p_1(t_1^{n,m}) \|_\Gamma^2.$$

The remaining divergence terms are equal to zero

$$\int_{I_1^{n,m}} \left(\operatorname{div} \mathbf{e}_1^{k,h}, p_1 - i_1^k I_1^h p_1 \right)_{\Omega_1} \mathrm{d}t = 0,$$

$$\int_{I_1^{n,m}} \left(\operatorname{div}(\mathbf{u}_1 - \mathbf{i}_1^k \mathbf{R}_1^h \mathbf{u}_1), \eta_1^{k,h} \right)_{\Omega_1} \mathrm{d}t = 0.$$

Indeed, the exact solution $\mathbf{u}$ and the fully discrete solution $\mathbf{u}^{k,h}$ are divergence-free by definition. The Ritz projection $\mathbf{R}^h \mathbf{u}$ is also divergence-free and the time projection operator i^k does not violate this property, see equation (5.46). That ends the proof. $\qquad \square$

Equivalently, we obtained

$$
\begin{aligned}
||\mathbf{e}^{k,h}&(t^N)|| + \int_I ||| \mathbf{e}^k ||| \, \mathrm{d}t \\
&= \mathcal{O}\big(k_1 \, ||\mathbf{d}_t \dot\epsilon(\mathbf{u}_1)||_{\Omega_1}\big) + \mathcal{O}\big(k_1 \, ||d_t p_1||_{\Omega_1}\big) + \mathcal{O}\big(k_1 h^{\frac{1}{2}} \, ||\mathbf{d}_t \sigma_1(\mathbf{u}_1, p_1) \cdot \vec{\mathbf{n}}_1||_\Gamma\big) \\
&\quad + \mathcal{O}\big(h^{r+1} \, ||\mathbf{d}_t \nabla^{r+1} \mathbf{u}_1||_{\Omega_1}\big) + \mathcal{O}\big(h^r \, ||\nabla^{r+1} \mathbf{u}_1||_{\Omega_1}\big) + \mathcal{O}\big(h^r \, ||\nabla^r p_1||_{\Omega_1}\big) \\
&\quad + \mathcal{O}\big(k_2 \, ||\mathbf{d}_t \dot\epsilon(\mathbf{u}_2)||_{\Omega_2}\big) + \mathcal{O}\big(k_2 \, ||d_t p_2||_{\Omega_2}\big) + \mathcal{O}\big(h^{r+1} \, ||\mathbf{d}_t \nabla^{r+1} \mathbf{u}_2||_{\Omega_2}\big) \\
&\quad + \mathcal{O}\big(h^r \, ||\nabla^{r+1} \mathbf{u}_2||_{\Omega_2}\big) + \mathcal{O}\big(k_2 h^{\frac{1}{2}} \, ||\mathbf{d}_t \sigma_2(\mathbf{u}_2, p_2) \cdot \vec{\mathbf{n}}_2||_\Gamma\big) + \mathcal{O}\big(h^r \, ||\nabla^r p_2||_{\Omega_2}\big)
\end{aligned}
$$

It is another example of an optimal estimate. We again were able to fully decouple time-step dependence. We will finish with a suboptimal estimate of the pressure error.

Theorem 6. *Let* $\mathbf{u} \in X(H_0^1(\Omega))^d$, $\mathbf{u}_j \in W^{1,\infty}(H^{r+1}(\Omega_j))^d$, $p_j \in W^{1,\infty}(H^r(\Omega_j))$ *for* $j = 1, 2$ *be continuous solutions to (5.41) and* $\mathbf{u}^{k,h} \times \mathbf{p}^{k,h} \in \big(X^{k,h}(r)\big)^d \times X^{k,h}(r-1)$ *their discrete counterparts and solutions to (5.47), then the following estimate holds*

$$
\begin{aligned}
\int_I &||\boldsymbol{\eta}^{k,h}||_\Omega^2 \, \mathrm{d}t \\
&\leqslant C \sum_{n=1}^N \sum_{m=1}^{N_1^n} \Bigg\{ (k_1^{n,m})^2 \max_{t \in I} ||\mathbf{d}_t \dot\epsilon(\mathbf{u}_1)||_{\Omega_1}^2 + (k_1^{n,m})^2 \max_{t \in I} ||d_t p_1||_{\Omega_1}^2 \\
&\quad + (k_1^{n,m})^2 h \max_{t \in I} ||\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1||_\Gamma^2 + (k_1^{n,m})^2 h \max_{t \in I} ||d_t p_1||_\Gamma^2 + h^{2r+2} \max_{t \in I} \left|\left| \mathbf{d}_t \nabla^{r+1} \mathbf{u}_1 \right|\right|_{\Omega_1}^2 \\
&\quad + h^{2r} ||\nabla^{r+1} \mathbf{u}_1(t_1^{n,m})||_{\Omega_1}^2 + h^{2r} ||\nabla^r p_1(t_1^{n,m})||_{\Omega_1}^2 \Bigg\} \\
&\quad + C \sum_{n=1}^N \sum_{m=1}^{N_2^n} \Bigg\{ (k_2^{n,m})^2 \max_{t \in I} ||\mathbf{d}_t \dot\epsilon(\mathbf{u}_2)||_{\Omega_2}^2 + (k_2^{n,m})^2 \max_{t \in I} ||d_t p_2||_{\Omega_2}^2 \\
&\quad + (k_2^{n,m})^2 h \max_{t \in I} ||\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2||_\Gamma^2 + (k_2^{n,m})^2 h \max_{t \in I} ||d_t p_2||_\Gamma^2 + h^{2r+2} \max_{t \in I} \left|\left| \mathbf{d}_t \nabla^{r+1} \mathbf{u}_2 \right|\right|_{\Omega_2}^2 \\
&\quad + h^{2r} ||\nabla^{r+1} \mathbf{u}_2(t_2^{n,m})||_{\Omega_2}^2 + h^{2r} ||\nabla^r p_2(t_2^{n,m})||_{\Omega_2}^2 \Bigg\},
\end{aligned}
$$

where the errors $\mathbf{e}^k = (e_1^k, e_2^k)^T$, $\boldsymbol{\eta}^k = (\eta_1^k, \eta_2^k)^T$ *are defined as*

$$
\begin{aligned}
e_1^{k,h} &:= \mathbf{u}_1^{k,h} - i_1^k \mathbf{R}_1^h \mathbf{u}_1, & e_2^{k,h} &:= \mathbf{u}_2^{k,h} - i_2^k \mathbf{R}_2^h \mathbf{u}_2, \\
\eta_1^{k,h} &:= p_1^{k,h} - i_1^k I_1^h p_1, & \eta_2^{k,h} &:= p_2^{k,h} - i_2^k I_2^h p_2.
\end{aligned}
$$

Proof. We would like to obtain an estimate of the form

$$c \int_{I_1^{n,m}} k_1^{n,m} \left\| \boldsymbol{\eta}_1^{k,h} \right\|_\Omega^2 \, \mathrm{d}t \leqslant \int_{I_1^{n,m}} (\mathbf{d}_t^k \mathbf{e}_1^{k,h}, \mathbf{e}_1^{k,h}) \, \mathrm{d}t + \int_{I_1^{n,m}} \left\| \left| \mathbf{e}_1^{k,h} \right| \right\|_\Omega^2 \, \mathrm{d}t \qquad (5.50)$$

on each $I_1^{n,m}$ for $\mathbf{e}_1^{k,h}$ as well as an analogous set of estimates for $\mathbf{e}_2^{k,h}$. We will be then able to use Theorem 5 on the right side of this identity. To achieve this goal, we need to use the inf-sup stability of our trial space. We are going to use a generalized version of the inf-sup stability condition proved in [5], from which follows that there exists a constant β such that for every $\mathbf{q}^{k,h} \in X^{k,h}(r-1)$, we have

$$\int_I \| \mathbf{q}^{k,h} \|_\Omega \, \mathrm{d}t \leqslant \beta \sup_{\boldsymbol{\varphi}^{k,h} \in V^{k,h}} \frac{b(\boldsymbol{\varphi}^{k,h}, \mathbf{q}^{k,h})}{\| | \boldsymbol{\varphi}^{k,h} | \|_\Omega},$$

where $V^{k,h} := \left(X^{k,h}(r) \right)^d$. In particular, we can claim that

$$\int_I \| \boldsymbol{\eta}^{k,h} \|_\Omega \, \mathrm{d}t \leqslant \beta \sup_{\boldsymbol{\varphi}^{k,h} \in V^{k,h}} \frac{b(\boldsymbol{\varphi}^{k,h}, \boldsymbol{\eta}^{k,h})}{\| | \boldsymbol{\varphi}^{k,h} | \|_\Omega}.$$

After using again the Galerkin orthogonality

$$b(\boldsymbol{\varphi}^{k,h}, \boldsymbol{\eta}^{k,h}) = - a^k(\mathbf{e}^{k,h}, \boldsymbol{\varphi}^{k,h}) + a^k(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \boldsymbol{\varphi}^{k,h}) + b(\boldsymbol{\varphi}^{k,h}, \mathbf{p} - \mathbf{i}^k \mathbf{I}^h \mathbf{p}).$$

Let us start with an estimation of the first term

$$\sup_{\boldsymbol{\varphi}^{k,h} \in V^{k,h}} \frac{a^k(\mathbf{e}^{k,h}, \boldsymbol{\varphi}^{k,h})}{\| | \boldsymbol{\varphi}^{k,h} | \|_\Omega}$$

$$\leqslant c \int_I \left\{ \left\| \mathbf{d}_t^k \mathbf{e}^{k,h} \right\|_\Omega + \boldsymbol{\nu} \left\| \nabla \mathbf{e}^{k,h} \right\|_\Omega + \nu_1 h^{\frac{1}{2}} \left\| \partial_{\vec{\mathbf{n}}_1} \mathbf{e}_1^{k,h} \right\|_\Gamma + \nu_2 h^{\frac{1}{2}} \left\| \partial_{\vec{\mathbf{n}}_2} \mathbf{e}_2^{k,h} \right\|_\Gamma \right. \qquad (5.51)$$

$$\left. + h^{-\frac{1}{2}} (1 + \gamma) \left\| \mathbf{e}_2^{k,h} - \mathbf{e}_1^{k,h} \right\|_\Gamma \right\} \mathrm{d}t \leqslant c \int_I \left\{ \left\| \mathbf{d}_t^k \mathbf{e}^{k,h} \right\|_\Omega + \left\| \left| \mathbf{e}^{k,h} \right| \right\|_\Omega \right\} \mathrm{d}t.$$

The time derivative is equal to

$$\int_{I_1^{n,m}} \left\| \mathbf{d}_t^k \mathbf{e}_1^{k,h} \right\|_{\Omega_1} \mathrm{d}t = \left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) - \mathbf{e}_1^{k,h}(t_1^{n,m-1}) \right\|_{\Omega_1}.$$

That leads to an estimation

$$\int_{I_1^{n,m}} \| \boldsymbol{\eta}^{k,h} \|_{\Omega_1} \, \mathrm{d}t \leqslant \left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) - \mathbf{e}_1^{k,h}(t_1^{n,m-1}) \right\|_{\Omega_1} + \int_{I_1^{n,m}} \left\| \left| \mathbf{e}_1^{k,h} \right| \right\|_{\Omega_1} \mathrm{d}t.$$

Knowing that all of these functions are piecewise constant in time, the inequality is equivalent to

$$k_1^{n,m} \| \boldsymbol{\eta}^{k,h}(t_1^{n,m}) \|_{\Omega_1} \leqslant \left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) - \mathbf{e}_1^{k,h}(t_1^{n,m-1}) \right\|_{\Omega_1} + k_1^{n,m} \left\| \left| \mathbf{e}_1^{k,h}(t_1^{n,m}) \right| \right\|_{\Omega_1}.$$

By squaring both sides of the inequality and going back to the integral form, we get

$$\int_{I_1^{n,m}} k_1^{n,m} \|\boldsymbol{\eta}^{k,h}\|_{\Omega_1}^2 \, dt \leq 2 \left(\left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) - \mathbf{e}_1^{k,h}(t_1^{n,m-1}) \right\|_{\Omega_1}^2 + \int_{I_1^{n,m}} k_1^{n,m} \left\| \left| \mathbf{e}_1^{k,h} \right| \right\|_{\Omega_1}^2 \, dt \right).$$

Then, based on (5.28), we have

$$\left\| \mathbf{e}_1^{k,h}(t_1^{n,m}) - \mathbf{e}_1^{k,h}(t_1^{n,m-1}) \right\|_{\Omega}^2 \leq \int_{I_1^{n,m}} 2(\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h})_{\Omega_1}.$$

Assuming that $k_1^{n,m} \leq 1$, we have

$$\int_{I_1^{n,m}} k_1^{n,m} \left\| \left| \mathbf{e}_1^{k,h} \right| \right\|_{\Omega_1}^2 \, dt \leq \int_{I_1^{n,m}} \left\| \left| \mathbf{e}_1^{k,h} \right| \right\|_{\Omega_1}^2 \, dt.$$

This way, the estimate (5.50) can be acquired. We continue with the remaining terms

$$\sup_{\boldsymbol{\varphi}^{k,h} \in V^{k,h}} \frac{a^k(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}, \boldsymbol{\varphi}^{k,h})}{\|\|\boldsymbol{\varphi}^{k,h}\|\|_{\Omega}} \leq c \int_I \left\{ \left\| \mathbf{d}_t^k(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}) \right\|_{\Omega} + \nu \left\| \dot{\boldsymbol{\epsilon}}(\mathbf{u} - \mathbf{i}^k \mathbf{R}^h \mathbf{u}) \right\|_{\Omega} \right.$$
$$\left. + h^{\frac{1}{2}} \left\| \nu_1 \partial_{\vec{\mathbf{n}}_1}(\mathbf{u}_1 - \mathbf{i}_1^k \mathbf{R}_1^h \mathbf{u}_1) - \nu_2 \partial_{\vec{\mathbf{n}}_2}(\mathbf{u}_2 - \mathbf{i}_2^k \mathbf{R}_2^h \mathbf{u}_2) \right\|_{\Gamma} \right\} \, dt.$$

All of these terms were estimated in the previous proofs. For a detailed recollection, we refer to the previous proof. The last term can be estimated using

$$\sup_{\boldsymbol{\varphi}^{k,h} \in V^{k,h}} \frac{b(\boldsymbol{\varphi}^{k,h}, \mathbf{p} - \mathbf{i}^k \mathbf{I}^h \mathbf{p})}{\|\|\boldsymbol{\varphi}^{k,h}\|\|_{\Omega}} \leq c \int_I \left\{ \left\| \mathbf{p} - \mathbf{i}^k \mathbf{I}^h \mathbf{p} \right\|_{\Omega} + h^{\frac{1}{2}} \left\| (p_1 - i_1^k I_1^h p_1) \vec{\mathbf{n}}_1 \right\|_{\Gamma} \right.$$
$$\left. + h^{\frac{1}{2}} \left\| (p_2 - i_2^k I_2^h p_2) \vec{\mathbf{n}}_2 \right\|_{\Gamma} \right\} \, dt.$$

For the interface terms, we have

$$\int_{I_1^{n,m}} \left\| (p_1 - i_1^k I_1^h p_1) \vec{\mathbf{n}}_1 \right\|_{\Gamma}^2 \, dt \leq (k_1^{n,m})^3 \max_{t \in I} \|d_t p_1\|_{\Gamma}^2 + k_1^{n,m} h^{2r-1} \|\nabla^r p_1(t_1^{n,m})\|_{\Gamma}^2.$$

Combining together all of the steps ends the proof. $\qquad\square$

We showed a suboptimal estimate of the form

$$\int_I \|\boldsymbol{\eta}^{k,h}\|_{\Omega} \, dt = \mathcal{O}\left(k_1^{\frac{1}{2}} \|\mathbf{d}_t \dot{\epsilon}(\mathbf{u}_1)\|_{\Omega_1}\right) + \mathcal{O}\left(k_1^{\frac{1}{2}} \|d_t p_1\|_{\Omega_1}\right) + \mathcal{O}\left(k_1^{\frac{1}{2}} h^{\frac{1}{2}} \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_1} \mathbf{u}_1\|_{\Gamma}\right)$$
$$+ \mathcal{O}\left(k_1^{\frac{1}{2}} h^{\frac{1}{2}} \|d_t p_1\|_{\Gamma}\right) + \mathcal{O}\left(k_1^{-\frac{1}{2}} h^{r+1} \|\mathbf{d}_t \nabla^{r+1} \mathbf{u}_1\|_{\Omega_1}\right) + \mathcal{O}\left(k_1^{-\frac{1}{2}} h^r \|\nabla^{r+1} \mathbf{u}_1\|_{\Omega_1}\right)$$
$$+ \mathcal{O}\left(k_1^{-\frac{1}{2}} h^r \|\nabla^r p_1\|_{\Omega_1}\right) + \mathcal{O}\left(k_2^{\frac{1}{2}} \|\mathbf{d}_t \dot{\epsilon}(\mathbf{u}_2)\|_{\Omega_2}\right) + \mathcal{O}\left(k_2^{\frac{1}{2}} \|d_t p_2\|_{\Omega_2}\right)$$
$$+ \mathcal{O}\left(k_2^{\frac{1}{2}} h^{\frac{1}{2}} \|\mathbf{d}_t \partial_{\vec{\mathbf{n}}_2} \mathbf{u}_2\|_{\Gamma}\right) + \mathcal{O}\left(k_2^{\frac{1}{2}} h^{\frac{1}{2}} \|d_t p_2\|_{\Gamma}\right) + \mathcal{O}\left(k_2^{-\frac{1}{2}} h^{r+1} \|\mathbf{d}_t \nabla^{r+1} \mathbf{u}_2\|_{\Omega_2}\right)$$
$$+ \mathcal{O}\left(k_2^{-\frac{1}{2}} h^r \|\nabla^{r+1} \mathbf{u}_2\|_{\Omega_1}\right) + \mathcal{O}\left(k_2^{-\frac{1}{2}} h^r \|\nabla^r p_2\|_{\Omega_2}\right)$$
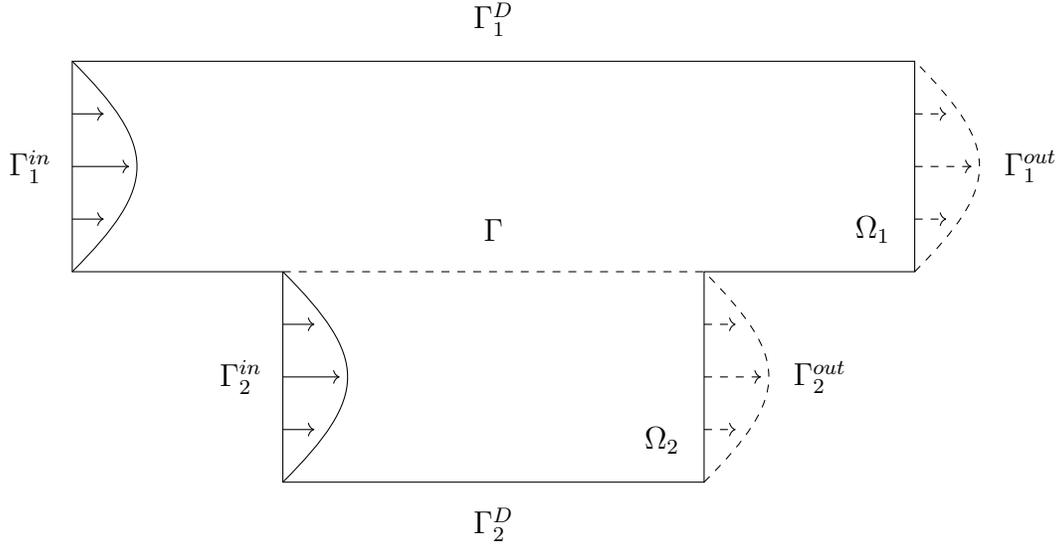
Figure 5.4: We show a sketch of the domains for our Stokes numerical example. The interface is denoted by Γ. Parabolic inflows are prescribed on the inlets Γ_1^{in} and Γ_2^{in} and free Neumann conditions on the outlets Γ_1^{out} and Γ_2^{out}. Otherwise, we take no-slip boundary conditions on Γ_1^D and Γ_2^D.

Due to the inf-sup estimation (5.51), half an order of convergence in time was lost. Namely, the source of this loss is the time derivative. We obtained the term $\|\mathbf{d}_t^k \mathbf{e}^{k,h}\|_\Omega^2 = (\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{d}_t^k \mathbf{e}^{k,h})_\Omega$, whereas, on the left side of the orthogonality identity (5.49), we have $(\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h})_\Omega$. We were only able to show the estimate

$$k(\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{d}_t^k \mathbf{e}^{k,h})_\Omega \leqslant (\mathbf{d}_t^k \mathbf{e}^{k,h}, \mathbf{e}^{k,h})_\Omega.$$

This is a nontrivial problem and was encountered for example in the series of articles [64, 65, 66, 67] about the Navier-Stokes equations. Specifically, in [67] the authors comment on the difficulties that come with the optimal estimation of the time derivative term. This issue has been successfully circumvented in [68], where the optimal convergence rate of pressure was retrieved. Indeed, the authors were able to show optimality for the Crank-Nicolson time-stepping scheme in L^2

$$\left\| I^k p - p^k \right\|_{L^2(I, H^1(\Omega))} \leqslant Ck^2 \tag{5.52}$$

and L^∞

$$\left\| J^k p - p^k \right\|_{L^\infty(I, H^1(\Omega))} \leqslant Ck^2 \tag{5.53}$$

norms. The operator J^k is given by

$$J^k p\big|_{I^n} := u(\bar{t}_n),$$

124

Uniform		Refinement in water				Refinement in oil			
-		Uniform		Asymmetric		Uniform		Asymmetric	
N_1	N_2	N_1	N_2	N_1	N_2	N_1	N_2	N_1	N_2
4	4	-	-	-	-	-	-	-	-
8	8	8	4	8	4	4	8	4	8
16	16	16	8	16	4	8	16	4	16
32	32	32	16	32	4	16	32	4	32
64	64	64	32	64	4	32	64	4	64

Table 5.1: We show different ways to refine the time mesh. N_1 and N_2 denote the total number of time-steps in each of the domains. We either use fully uniform time-stepping, uniform refinement where time-steps in one domain are twice smaller than in the other, or fully asymmetric refinement where time-steps in only one domain are refined.

where $\bar{t}_n := \frac{t_n + t_{n-1}}{2}$. This publication considered neither coupled problems nor multirate time-stepping.

5.4.3 Numerical Example

As a numerical example, we take a coupled Stokes problem. The configuration emulates a coupling of water and oil, therefore $\nu_1 = 1$ and $\nu_2 = 56$. The domain consists of two connected pipelines. Each of them is given by rectangular domains $\Omega_1 = [0,4] \times [0,1]$ and $\Omega_2 = [1,3] \times [-1,0]$. On each of the inlets parabolic inflows are prescribed

$$\mathbf{u}_1^{in} = \sin(\pi t)x_1(1-x_1) \quad \text{and} \quad \mathbf{u}_2^{in} = \sin(\pi t)x_2(1+x_2)$$

for $t \in I = [0,1]$. On the outlets, we choose free Neumann boundary conditions. Otherwise, we take no-slip Dirichlet boundary conditions. A sketch of the domain is shown in Figure 5.4. Since the flow is fully driven by the boundary conditions, we take $\mathbf{f}_1 = \mathbf{f}_2 = \mathbf{0}$.

In Figure 5.6 we show the results of a velocity convergence study. The problem is simulated on a constant fine space mesh and perform refinement in the time component only. We start with a very coarse time mesh with only 4 time-steps in both of the domains. Then we either refine both time meshes or one of them only. Different types of time refinement are described in more detail in Table 5.1 and Figure 5.5. The top figure in Figure 5.6 shows the convergence rate of the total error appearing as the left hand side in the inequality proved in Theorem 5. Since the error is squared, we expect a quadratic convergence rate and it is exactly what we were able to achieve. In this figure as well as all the others, we included a dashed triangle to illustrate quadratic convergence for comparison. In the figures below, we present convergence rates of total and H^1 errors corresponding to either the water or the oil problem. There, only time-steps in one domain were refined, that is time-steps in the water
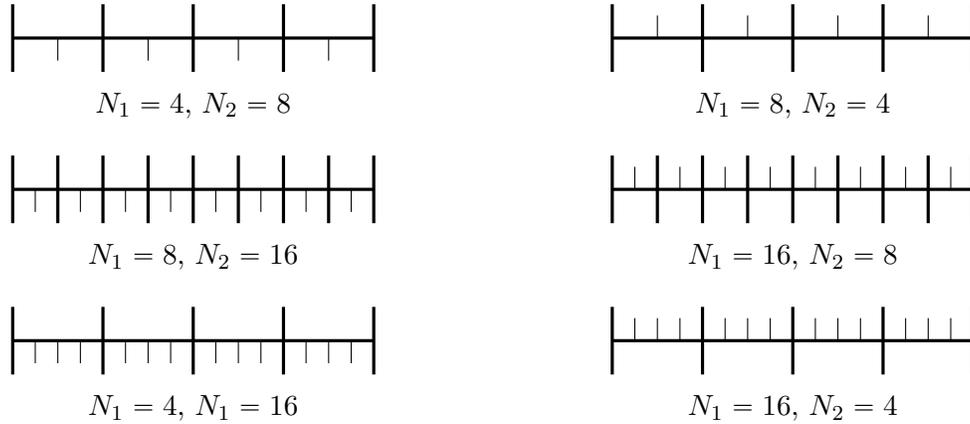
Figure 5.5: We present different instances of time meshes described in Table 5.1 to illustrate the differences between uniform and asymmetric meshes.

problem in the middle row, or in the oil problem in the bottom row. In both cases, the left graphs correspond to the total errors and on the right, we show partial H^1 errors. Since the total error is heavily dominated by contributions from the oil problem, only partial refinement in this subproblem leads to convergence of the overall error. This partial refinement does not affect the convergence rate of partial H^1 errors negatively in any way. It is especially apparent in the water problem. That in turn, validates our findings in Theorem 5 where we were able to fully decouple time contributions from different subproblems. In these figures, we did not include the refinement where the time-steps in one domain were twice the size as in the other because the corresponding curves fully overlap with the ones presented here.

In Figure 5.7 we present graphs corresponding to pressure. In the top figure, we show the convergence rate of the joint L^2 error. Again, we were able to achieve quadratic convergence (we remind the readers that the plot includes a squared error). This way, our conjecture was confirmed that the error estimate proved in Theorem 6 is suboptimal. However, in the graphs of partial errors below we can see that they are not fully decoupled. There, we compare asymmetric meshes and uniform meshes (time-steps in one domain are twice smaller as in the other). Indeed, especially in the case of the water problem, one can see a deterioration of the convergence rate in the case of asymmetric time meshes. The graphs suggest that this deterioration is driven by the number of micro time-steps. This effect is much more pronounced in the water problem. That might be due to the difference in viscosity between the two problems.
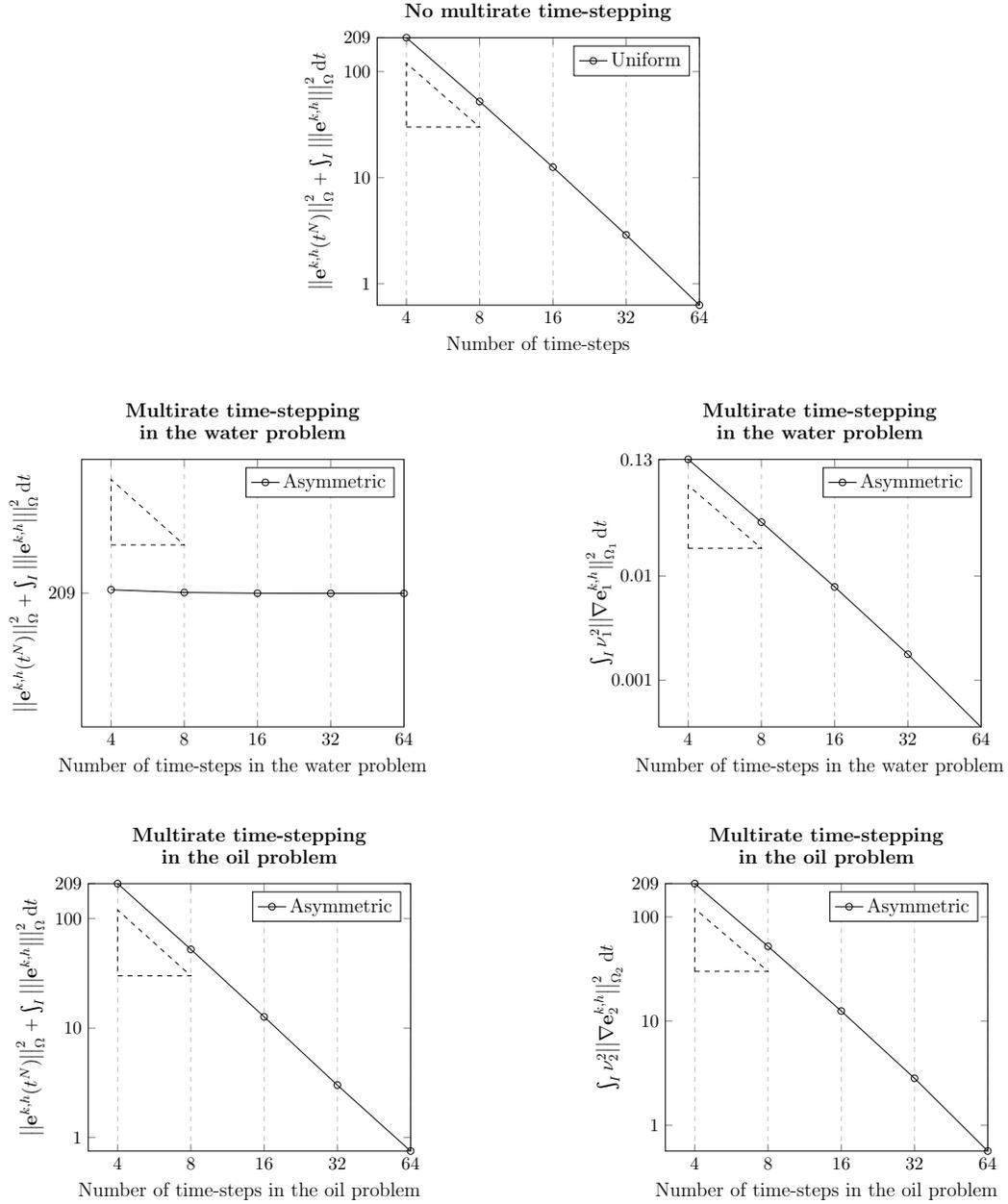
Figure 5.6: Convergence rate of the total velocity error on a uniform mesh (top), total and H^1 water errors for refinement in the water mesh only (middle row), and total and H^1 oil errors for refinement in the oil mesh only (bottom row) with respect to the number of time-steps.
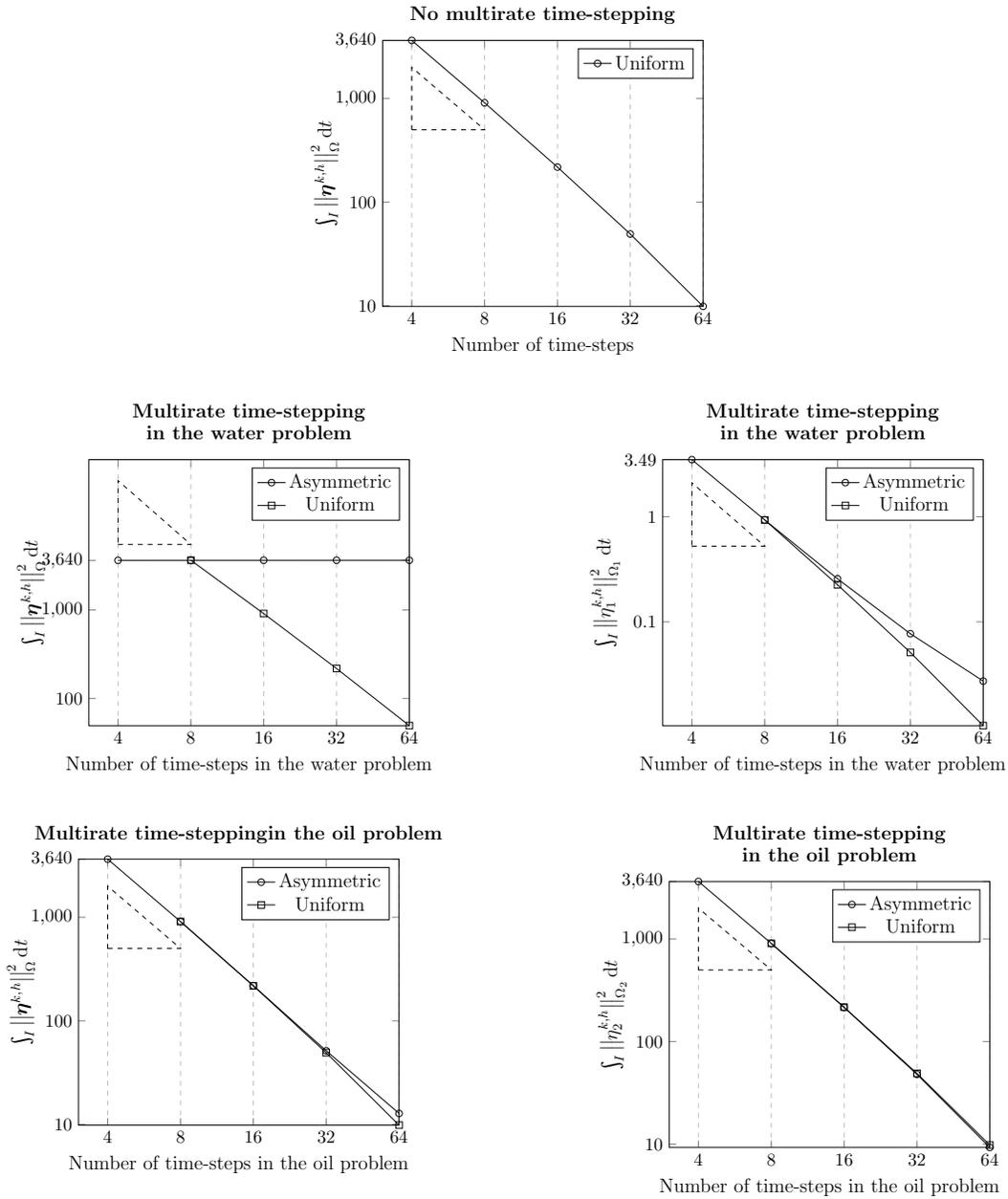
Figure 5.7: Convergence rate of the total pressure error on a uniform mesh (top), total and water errors for refinement in the water mesh (middle row), and total and oil errors for refinement in the oil mesh (bottom row) with respect to the number of time-steps.

Conclusion

In this study, we looked at various aspects of multirate time-stepping methods for coupled problems. In the first part, we were mostly concerned with the practical side of handling such systems and introduced a special time partitioning that allowed for an independent choice of time-step sizes without violations of coupling conditions. The time discretization was embedded in a space-time framework. We examined decoupling algorithms that enabled solving strongly coupled systems within the partitioned regime. After deriving and solving adjoint formulations, we computed a DWR estimator and used it as an adaptivity criterion.

All of these considerations were supported by numerical results for three model problems - a simple linear interface coupled problem, a highly complex nonlinear FSI system on moving domains and a slightly nonlinear volume coupled one. Except for the FSI problem, we were able to achieve highly satisfactory results. Indeed, the newly proposed shooting method performed much better than a commonly used relaxation method. Adaptive time-stepping yielded significantly more efficient error reduction than the fully uniform refinement. The results for the FSI system were only a partial success. Although conversely to the relaxation method, the shooting method was capable of reaching convergence, it required a substantial number of iterations. Additionally, introducing micro time-stepping in the solid domain gave rise to oscillations in pressure which in turn caused the simulation to crash after just a few time-steps. As a result, we only tested the error estimator on fully uniform time meshes. We did not take into account the quadrature errors coming from the mismatch between the space-time discretization and the one we actually used. Despite that, the effectivity values were promising. We suspect that the failure of multirate time-stepping for the FSI problem comes from the shift of the scheme to the explicit side whenever micro time-stepping is introduced. Indeed, the intermediate solution is computed as a linear combination between the previously computed one and the solution of a current system. As we discussed in Section 3.2, these instabilities are only present when one considers a partitioned approach. Unfortunately, standard monolithic discretizations are not flexible enough to allow for different time-step sizes for each of the problems.

This issue can be circumvented by leaning more into the space-time direction. The cost of solving algebraic systems arising from full space-time discretizations can be prohibitive. There exists however an intermediate solution using, so-called, *time slabs*. There, one sequentially solves full space-time formulations defined on patches consisting of a small number of time-steps. This method has been presented in more detail for example in [69, 70]. Our time

partitioning with macro and micro time-step seems to perfectly fit into this framework. At the moment of writing this thesis, our research in this direction is at the very initial stage and we can not yet say whether this method can indeed solve our problems, nor can we discuss any implementation details. Before closing this paragraph, we would like to mention that although we are disappointed that our approach did not provide us with a fully satisfactory simulation of the FSI problem, we should remember that this is a very challenging benchmark. It strongly couples two nonlinear systems with an additional incompressibility condition in the fluid. The ALE coordinates introduce additional highly nonlinear terms. Therefore, the limitations of our approach illustrated by this benchmark should not discredit it in other contexts.

In the last chapter of this thesis, we were instead concentrated on more theoretical aspects of multirate time-stepping. We proved a priori error estimates for a series of linear coupled problems. All of them were discretized using the implicit Euler time-stepping scheme. We looked at a standard system of coupled ordinary differential equations and then proceeded to two systems of partial differential equations. The first one was given by two coupled heat equations with different diffusivity constants. The second one was an extension of the former and consisted of two Stokes equations. Except for the pressure, where the order of convergence was reduced by one half, our estimates yielded an optimal linear convergence rate. To handle the transfer of interface conditions between different domains, we introduced a new interpolation operator. It was specifically designed to have an average zero over each time interval. Due to using the dG(0) representation of the implicit Euler scheme, this property is not violated after multiplying an average term with any trial or test function since both of them are piecewise constant in time. This trick was used frequently in the proofs. The estimates for the fully discrete heat and Stokes equation were proved in a modified energy norm that incorporated the Nitsche jump over the interface. To prove the estimate for pressure, we used a modified inf-sup condition [5] which is compatible with this new norm.

The suboptimality of the pressure estimate comes from the estimation of the time derivative term in the inf-sup condition. This term is known to be difficult to estimate [67]. Moreover, this problem has been successfully solved in [68], where the authors were able to derive optimal pressure estimates. Given these circumstances, we do not view this suboptimality as a significant failure on our part. In fact, we are quite pleased with the results presented here. They form a good complement to the previous part which was clearly more application oriented. That being said, one can clearly spot ways, in which these findings could be extended. One could consider problems coupling different physical models, for example, the heat and wave equation. We could also extend the proofs to account for the Crank-Nicolson and other time-stepping methods. Another possibility would be to introduce nonlinear terms.

Acknowledgements

I would not be able to write this thesis without the help of kind people around me as well as institutional support. I am incredibly grateful for this opportunity. I am sure that I will remember the time spent at OVGU fondly. The PhD program, that I had a pleasure to be a part of, has been a positive force in my life and awarded me with many resources that otherwise I would not have access to.

First of all, I would like to thank my supervisor prof. Thomas Richter. Most likely, it was up to him that I was hired in the first place. He was nothing but respectful towards me. I had the freedom to ask all the questions I needed to, also the stupid ones. He responded to my emails without delays, I could meet him in person whenever I asked for it. He was supportive of me getting an extension of my contract when I requested for one. I received all the guidance that I needed. I would like to mention my colleagues as well and thank them for their support, especially with editing of this thesis and helping me with the PhD defense presentation.

I would also like to acknowledge the graduate college MathCoRe for funding my PhD. Without its financial support, this project would not be possible. I want to thank for the aforementioned extension and the access to international conferences. On top of that, the college made me a part of the MathCoRe community, where I could meet other PhD students more easily. That is especially helpful to international students such as me.

Last but not least I want to express my appreciation to all the people close to my heart. Most of all, I would like to thank my partner Martin. He was essential in soothing the emotional ups and downs that typically come with research. My friends Lamprini, Shaimaa and Ali listened to me patiently. Back in Poland, I could always count on Łukasz, Ewelina and Wojtek.

Bibliography

[1] Christiane Förster, Wolfgang 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.

[2] 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*, 194:4506–4527, 10 2005.

[3] Roland Becker and Rolf Rannacher. Weighted a posteriori error control in fe methods. 03 1997.

[4] Roland Becker and Rolf Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica 2001*, 10:1 – 102, 05 2001.

[5] Peter Hansbo and Mats Larson. Discontinuous galerkin methods for incompressible and nearly incompressible elasticity by nitsche's method. *Computer Methods in Applied Mechanics and Engineering*, 191:1895–1908, 02 2002.

[6] Stefan Turek and Jaroslav Hron. Proposal for numerical benchmarking of fluid–structure interaction between an elastic object and laminar incompressible flow. *Fluid-Structure Interaction: Modelling, Simulation, Optimisation*, 53:371–385, 06 2007.

[7] David Keyes, Tony Chan, Gérard Meurant, Jeffrey Scroggs, and Robert Voigt. Domain decomposition methods for partial differential equations. *Mathematics of Computation*, 62:941, 04 1994.

[8] Andrea Toselli and Olof Widlund. *Domain Decomposition Methods – Algorithms and Theory*, volume 34. Springer Series in Computational Mathematics, 01 2005.

[9] Carlos Felippa and Thomas Geers. Partitioned analysis of coupled mechanical systems. *Engineering Computations*, 5:123–133, 12 1988.

[10] Ulrich Küttler and Wolfgang Wall. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43:61–72, 01 2008.

[11] Mingliang Yu, Xueyuan Nie, Guowei Yang, and Peinan Zhong. Fixed-point fluid structure interaction analysis based on geometrically exact approach. *Scientific Reports*, 10:10322, 06 2020.

[12] Joris Degroote, Rob Haelterman, Sebastiaan Annerel, Peter Bruggeman, and Jan Vierendeels. Performance of partitioned procedures in fluid-structure interaction. *Computers & Structures*, 88, 04 2010.

[13] Dana Knoll and David Keyes. Jacobian-free newton-krylov methods: A survey of approaches and applications. *Journal of Computational Physics*, 193:357–397, 01 2004.

[14] Miguel Fernández and Marwan Moubachir. A newton method using exact jacobians for solving fluid-structure coupling. *Computers Structures*, 83:127–142, 01 2005.

[15] Jean-Frédéric Gerbeau and Marina Vidrascu. A quasi-newton algorithm based on a reduced model for fluid-structure interaction problems in blood flows. *ESAIM Mathematical Modelling and Numerical Analysis*, 37, 07 2003.

[16] Joris Degroote, Klaus-Jürgen Bathe, and Jan Vierendeels. Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Computers & Structures*, 87:793–801, 06 2009.

[17] Jean-Frédéric Gerbeau, Marina Vidrascu, and Pascal Frey. Fluid-structure interaction in blood flows on geometries coming from medical imaging. *Computers & Structures*, 83:155–165, 01 2005.

[18] Matthias Heil. An efficient solver for the fully coupled solution of large-displacement fluid-structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 193:1–23, 01 2004.

[19] Kenneth Eriksson, Donald Estep, Peter Hansbo, and Claes Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, 4:105–158, 01 1995.

[20] Kenneth Eriksson, Donald Estep, Peter Hansbo, and Claes Johnson. *Computational Differential Equation.* Cambridge University Press, 01 1996.

[21] Dominik Meidner and Thomas Richter. A posteriori error estimation for the fractional step theta discretization of the incompressible navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 288, 05 2015.

[22] Dominik Meidner and Thomas Richter. Goal-oriented error estimation for the fractional step theta scheme. *Computational Methods in Applied Mathematics*, 14, 04 2014.

[23] Martyna Soszyńska and Thomas Richter. Adaptive time-step control for a monolithic multirate scheme coupling the heat and wave equation. *BIT Numerical Mathematics*, 61, 04 2021.

[24] Stefan Turek. Benchmark computations of laminar flow around a cylinder. *Notes on Numerical Fluid Mechanics*, 52, 01 1996.

[25] Thomas Richter. *Fluid-Structure Interactions: Models, Analysis and Finite Elements.* Springer Publishing Company, Incorporated, 1st edition, 2017.

[26] Joachim Nitsche. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15, 1971.

[27] Erik Burman and Miguel Fernández. An unfitted nitsche method for incompressible fluid-structure interaction using overlapping meshes. *Computer Methods in Applied Mechanics and Engineering*, 279, 09 2014.

[28] Mitchell Luskin, Rolf Rannacher, and Wolfgang Wendland. On the smoothing properties of crank-nicholson scheme. *Applicable Analysis*, 14:117–135, 01 1982.

[29] Rolf Rannacher. Finite element solution of diffusion problems with irregular data. *Numerische Mathematik*, 43:309–327, 06 1984.

[30] John Heywood and Rolf Rannacher. Finite-element approximation of the nonstationary navier–stokes problem. part iv: Error analysis for second-order time discretization. *SIAM Journal on Numerical Analysis*, 27, 04 1990.

[31] Roland Glowinski. Finite element methods for incompressible viscous flow. *Handbook of Numerical Analysis*, 9:3–1176, 12 2003.

[32] Stefan Turek, Ludmila Rivkind, Jaroslav Hron, and Roland Glowinski. Numerical study of a modified time-stepping -scheme for incompressible flow simulations. *Journal of Scientific Computing*, 28:533–547, 01 2006.

[33] Thomas Wick. Adaptive finite element simulation of fluid-structure interaction with application to heart-valve dynamics. 01 2011.

[34] Thomas Wick. Stability estimates and numerical comparison of second order time-stepping schemes for fluid-structure interactions. 01 2013.

[35] Charbel Farhat, Kristoffer van der Zee, and Philippe Geuzaine. Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 195:1973–2001, 03 2006.

[36] Harald Van Brummelen. Added mass effects of compressible and incompressible flows in fluid-structure interaction. *Journal of Applied Mechanics*, 76:021206–, 03 2009.

[37] Harald Van Brummelen. Partitioned iterative solution methods for fluid–structure interaction. *International Journal for Numerical Methods in Fluids*, 65:3 – 27, 01 2011.

[38] Bruce Irons and Robert Tuck. A version of the aitken accelerator for computer iteration. *International Journal for Numerical Methods in Engineering*, 1:275 – 277, 07 1969.

[39] Mark Ainsworth and J.Tinsley Oden. A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 142(1):1–88, 1997.

[40] Rüdiger Verführt. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley, 01 1996.

[41] Thomas Grätsch and Klaus-Jürgen Bathe. A posteriori error estimation techniques in practical finite element analysis. *Computers & Structures*, 83:235–265, 01 2005.

[42] Michael Schmich and Boris Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM J. Scientific Computing*, 30:369–393, 01 2008.

[43] Mike Giles and Endre Süli. *Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality*, pages 145–236. Acta Numerica, 07 2002.

[44] Rolf Hartmann. Multitarget error estimation and adaptivity in aerodynamic flow simulations. *SIAM Journal on Scientific Computing*, 31:708–731, 01 2008.

[45] Rolf Rannacher and Franz-Theo Suttmeier. A posteriori error control in finite element methods via duality techniques: Application to perfect plasticity. *Computational Mechanics*, 21:123–133, 03 1998.

[46] David Venditti and David Darmofal. Anisotropic grid adaptation for functional outputs: Application to two-dimensional viscous flows. *Journal of Computational Physics*, 187:22–46, 05 2003.

[47] Thomas Wick. Dual-weighted residual adaptivity for phase-field fracture propagation. *PAMM*, 15:619–620, 10 2015.

[48] Natacha Lord and Anthony Mulholland. A dual weighted residual method applied to complex periodic gratings. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 469, 12 2013.

[49] Julian Roth, Max Schröder, and Thomas Wick. Neural network guided adjoint computations in dual weighted residual error estimation. *SN Applied Sciences*, 4, 02 2022.

[50] Vincent Heuveline and Rolf Rannacher. Duality-based adaptivity in the hp-finite element method. *Journal of Numerical Mathematics*, 11:95–113, 06 2003.

[51] Ted Belytschko, Robert Gracie, and Giulio Ventura. A review of extended/generalized finite element methods for material modelling. *Modelling and Simulation in Materials Science and Engineering*, 17:043001, 04 2009.

[52] Lucy Zhang, Axel Gerstenberger, Xiaodong Wang, and Wing Liu. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering*, 193:2051–2067, 05 2004.

[53] Zhilin Li, Tao Lin, Yanping Lin, and Robert Rogers. An immersed finite element space and its approximation capability. *Numerical Methods for Partial Differential Equations*, 20:338 – 367, 05 2004.

[54] Tao Lin, Dongwoo Sheen, and Xu Zhang. A nonconforming immersed finite element method for elliptic interface problems. *Journal of Scientific Computing*, 79, 04 2019.

[55] Haifeng Ji, Jinru Chen, and Zhilin Li. A symmetric and consistent immersed finite element method for interface problems. *Journal of Scientific Computing*, 61, 12 2014.

[56] Lin Mu, Junping Wang, Xiu Ye, and Shan Zhao. A new weak galerkin finite element method for elliptic interface problems. *Journal of Computational Physics*, 325, 08 2016.

[57] Peter Hansbo. Nitsche's method for interface problems in computational mechanics. *GAMM-Mitteilungen*, 28, 11 2005.

[58] Anita Hansbo and Peter Hansbo. A finite element method for the simulation of strong and weak discontinuities in elasticity. *Computer Methods in Applied Mechanics and Engineering*, 193, 01 2004.

[59] Peter Hansbo, Joakim Hermansson, and Thomas Svedberg. Nitsche's method combined with space-time finite elements for ale fluid-structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 193:4195–4206, 10 2004.

[60] Zhiqiang Cai, Xiu Ye, and Shun Zhang. Discontinuous galerkin finite element methods for interface problems: A priori and a posteriori error estimations. *SIAM J. Numerical Analysis*, 49:1761–1787, 01 2011.

[61] Andrea Cangiani, Emmanuil Georgoulis, and Younis Sabawi. Convergence of an adaptive discontinuous galerkin method for elliptic interface problems. *Journal of Computational and Applied Mathematics*, 367:112397, 08 2019.

[62] Sven Groß, Volker Reichelt, and Arnold Reusken. A finite element based level set method for two-phase incompressible flows. *Computing and Visualization in Science*, 9(4):239–257, oct 2006.

[63] Erik Burman and Miguel Fernández. Explicit strategies for incompressible fluid-structure interaction problems: Nitsche type mortaring versus robin-robin coupling. *International Journal for Numerical Methods in Engineering*, 97, 02 2014.

[64] John Heywood and Rolf Rannacher. Finite element approximation of the nonstationary navier–stokes problem. i. regularity of solutions and second-order error estimates for spatial discretization. *SIAM Journal on Numerical Analysis*, 19:275–311, 04 1982.

[65] John Heywood and Rolf Rannacher. Finite element approximation of the nonstationary navier–stokes problem, part ii: Stability of solutions and error estimates uniform in time. *SIAM Journal on Numerical Analysis*, 23:750–777, 08 1986.

[66] John Heywood and Rolf Rannacher. Finite element approximation of the nonstationary navier–stokes problem iii. smoothing property and higher order error estimates for spatial discretization. *SIAM Journal on Numerical Analysis*, 25:489–512, 06 1988.

[67] John Heywood and Rolf Rannacher. Finite-element approximation of the nonstationary navier–stokes problem. part iv: Error analysis for second-order time discretization. *SIAM Journal on Numerical Analysis*, 27, 04 1990.

[68] Florian Sonner and Thomas Richter. Second order pressure estimates for the crank–nicolson discretization of the incompressible navier–stokes equations. *SIAM Journal on Numerical Analysis*, 58:375–409, 01 2020.

[69] Marius Paul Bruchhäuser, Uwe Köcher, and Markus Bause. On the implementation of an adaptive multirate framework for coupled transport and flow, 2022.

[70] Jan Philipp Thiele and Thomas Wick. Variational partition-of-unity localizations of space-time dual weighted residual estimators for parabolic problems, 2022.