Unfitted Finite Elements for Fluid-Rigid Body Interaction Problems

Dissertation
zur Erlangung des akademischen Grades

doctor rerum naturalium
(Dr. rer. nat.)

von M.Sc. Henry Maximilian von Wahl

geb. am 27. April 1993 in Weinheim

genehmigt durch die Fakultät für Mathematik der

Otto-von-Guericke-Universität Magdeburg

Gutachter: Prof. Dr. Thomas Richter
Jun.-Prof. Dr. Jan Heiland
Prof. Dr. Maxim Olshanskii

eingereicht am: 29. Juni 2021
Verteidigung am: 8. Oktober 2021

Abstract

In this thesis, we develop unfitted finite element techniques to simulate fluid-rigid body interaction problems. The mathematical model for this consists of the incompressible Navier-Stokes equations and rigid body motion, coupled via the transfer of forces at the fluid-solid interface. Our method solves the resulting coupled moving domain problem in a fully Eulerian framework.

The discretisations considered in this thesis are based on unfitted finite element approaches (CutFEM). A fixed background mesh is considered in this approach such that moving domain problems can be solved in an Eulerian framework. The geometry is then described through a level set function, boundary conditions are implemented using Nitsche's method, and the Eulerian approach to the moving domain problem is enabled via an implicit extension of the solution with ghost-penalty stabilisation. As a result of the ghost-penalty stabilisation, the resulting linear systems have bounded condition numbers, independent of the cuts between the level set and the mesh.

We begin in the stationary case by analysing an isoparametric unfitted finite element method, with Taylor-Hood elements on the active mesh, applied to the Stokes equations on stationary domains. We then continue to the time-dependent case and analyse the isoparametric CutFEM method for the transient Oseen equation. We continue by using this method to compute a benchmark problem consisting of a freely rotating circular obstacle in a channel flow.

To advance to flow problems on moving domains, we develop an Eulerian time-stepping scheme for the time-dependent Stokes problem on moving domains using CutFEM. This method is then analysed with respect to stability and accuracy. In particular, we include in our analysis the geometry approximation error made in CutFEM by the approximation of the level set function. This Eulerian moving domain method is then applied to a flow problem, driven by the motion of a cylinder, to investigate the properties of the moving domain method in the full non-linear Navier-Stokes setting.

We progress to coupled fluid-rigid body interaction problems and investigate the stability and accuracy properties of a method, based on our Eulerian time-stepping scheme, for the coupled motion of a solid in free-fall with the fluid. Stability is shown for the time-dependent Stokes problem in the fluid domain, and the error is considered for the reduced case of the heat-equation in the fluid domain. Furthermore, we consider this method for a fluid-structure interaction problem, which is devised from experiential data. This problem includes solid contact and the solid rebounding of the fluid boundary wall.

Furthermore, we consider triangular shaped particles as a prototype of non-smooth geometries for the rigid body. These are then represented in CutFEM using multiple level set functions. In order to work with small triangular shaped particles which cannot be resolved sufficiently by the computational mesh, we develop an artificial deep neural network approach to predict the forces acting from the fluid onto the solid. This network predicts the forces more accurately than the direct evaluation of the forces from the FEM solution by using input data, which we can compute accurately on coarse meshes. As a result, the motion of the solid particles we obtain on an under-resolved mesh is comparable to that realised on highly resolved meshes.

Zusammenfassung

Wir entwickeln in dieser Arbeit eine Finite Elemente Methode um Fluid-Festkörper-Interaktionen zu simulieren. Das entsprechende mathematische Modell besteht aus den inkompressiblen Navier-Stokes- und Newton-Euler-Gleichgungen. Diese beiden Systeme sind durch die Kräfte am Interface zwischen dem Fluid und dem Festkörper gekoppelt. Unsere Methode löst das hieraus entstehende gekoppelte System in Eulerschen Koordinaten.

Die Diskretisierungen, die wir in dieser Arbeit betrachten, basieren auf einer nicht-angepassten Finite-Elemente-Methode (CutFEM). Hier verwenden wir dabei ein festes Gitter einer Hintergrundsgeometrie, wodurch es möglich ist, Probleme mit veränderlicher Geometrie in Eulerschen Koordinaten zu betrachten. Die Geometrie wird dann durch eine Levelset-Funktion beschrieben und Randbedingungen werden mit Nitsche's Methode erzwungen. Der Eulersche Ansatz für sich bewegende Gebiete wird durch eine implizite Erweiterung der Lösung ermöglicht. Diese Erweiterung wird durch die sogenannte "ghost-penalty" Stabilisierung realisiert. Durch diese Stabilisierung ist zudem gewährleistet, dass die jeweilige Konditionszahl der linearen Gleichungssysteme, die durch die Diskretisierung entsteht, unabhängig von den Schnitten zwischen dem Gitter und der Levelset-Funktion beschränkt bleibt.

Zunächst betrachten wir den stationären Fall und analysieren eine isoparametrische CutFEM Methode für die Stokes-Gleichung mit Taylor-Hood Elementen auf dem aktiven Teil des Gitters. Wir gehen dann in den zeitabhängigen Fall über und untersuchen die isoparametrische CutFEM Methode anhand der zeitabhängigen Oseen-Gleichung. Des Weiteren wenden wir die Methode auf ein Fluid-Struktur Problem an, welches aus einer sich frei drehenden Kugel in einer Kanalströmung besteht.

Um Stömungsmechanikprobleme auf sich bewegenden Gebieten zu betrachten, entwickeln wir ein Eulersches Zeitschrittverfahren für die zeitabhängige Stokes-Gleichung auf sich bewegenden Gebieten. Diese Methode analysieren wir bezüglich Stabilität und Genauigkeit. Insbesondere berücksichtigen wir hierbei den Geomtrieapproximationsfehler, der bei CutFEM durch die Approximation der Levelset-Funktion entsteht. Des Weiteren wenden wir diese Methode auf ein Beispiel an, in dem das Fluid durch die Bewegung eines Zylinders angetrieben wird, um die Eigenschaften unserer Methode im Falle der nicht-linearen Navier-Stokes Gleichungen zu untersuchen.

Wir fahren damit fort, gekoppelte Fluid-Festkörper-Probleme zu betrachten. Basierend auf dem Eulerschen Zeitschrittverfahren analysieren wir die Stabilität und Genauigkeit einer Methode für das gekoppelte Problem eines Festkörpers, der sich im freien Fall in einem Fluid befindet. Wir zeigen die Stabilität der Methode für den Fall der zeitabhängigen Stokes-Gleichung und der Fehler wird im reduzierten Fall der Wärmeleitungsgleichung betrachtet. Wir wenden dann das Eulersche Zeitschrittverfahren an einem Fluid-Struktur-Interaktion Problem an, welches anhand von experimentellen Daten konstruiert wurde. Dieses Problem beinhaltet das Abprallen der Struktur von der Wand des Fluidbehälters.

Darüber hinaus betrachten wir dreieckförmige Teilchen als Prototypen von Festkörpern mit nicht-glatter Geometrie. In der CutFEM Methode werden diese durch mehrere Levelset-Funktionen beschrieben. Damit die Bewegung solcher Teilchen auf sehr groben Gittern genau darstellen zu können, entwickeln wir ein künstliches neuronales Netz, das die Kräfte, die von dem Fluid auf die Teilchen wirken, vorhersagen soll. Da dieses Netz Werte als Eingabe erhält, welche auch in sehr grob aufgelösten Gittern genau zu berechnen sind, ist die Bewegung der Teilchen vergleichbar mit der Bewegung, die auf sehr hoch aufgelösten Gittern entsteht.

Contents

<div align="right">

CHAPTER 1

</div>

Introduction

Contents of Chapter

1.1 Aims and Motivation

This thesis is concerned with the numerical discretisation of the partial differential equations modelling the flows laden with particles. Particulate flows or particle settling in a fluid have many industrial and biological applications. Examples include the transport of blood cells in blood flows and the settling of sand in water. In particular, we focus on rigid body particles coupled to an incompressible fluid flow. While the mathematical theory of the motion of rigid particles in a liquid is one of the most classical problems in fluid dynamics, it remains an active area of mathematical research, c.f. [Gal02] and the references therein.

From the numerical perspective, there are many difficulties in the accurate discretisation of the fluid-structure interaction (FSI) problem posed by the dynamics of a rigid body coupled to an incompressible fluid. These include the coupled nature of the fluid and solid problems and topology changes in the geometry, which occur when particles collide with each other or with a fluid boundary wall. Furthermore, in the Eulerian Framework, which is the natural viewpoint for fluid problems, the moving fluid-solid interface poses additional problems. For one, the motion of this interface is not known a priori, and second, points that contain the fluid at one point in time may contain the solid in the next point in time and vice-versa. Any numerical scheme must, therefore, carefully deal with these issues.

The most well-established method for solving this type of problem is the Arbitrary Lagrangian-Eulerian (ALE) method [DGH82; Don+04]. In the ALE approach, the domain is mapped into a reference configuration, and then standard finite element methods (FEM) are used to solve the resulting problem. This results in very efficient and accurate discretisations for problems where the deformation with respect to the reference configuration remains moderate. However, if the deformation becomes too large or topology changes in the geometry occur, then the ALE method fails. In this case, the consideration of fully Eulerian approaches are appropriate to overcome these problems. This is the approach we will follow in this thesis.

In a fully Eulerian approach, a general background domain is discretised to define a set of potential fluid unknowns and the geometry is described separately on this background domain. We shall focus on a method known as CutFEM [Bur+14]. Here the geometry is represented by a level set function, and boundary conditions on the interface are implemented using Nitsche's method. While a significant amount of research has been devoted to *unfitted* finite element approaches for problems on stationary domains, the case of moving domains – one of the main goals of such methods – remains less well studied.

The goal of this thesis is to study and extend CutFEM to the coupled moving domain problem posed by fluid-rigid body interaction problems. The main achievements realised in this thesis are as follows:

- Most analysis of CutFEM ignores the geometry error made by the piecewise linear level set approximation used to define quadrature rules on elements cut by the level set. In this thesis, we extend the analysis of an isoparametric finite element method [Leh16; Leh17] to flow problems posed on stationary domains using Taylor-Hood elements. The analysis then includes the consistency error made through the geometry approximation. This approach uses a parametric mapping to deform the unfitted mesh such that the explicit, piecewise linear level set, which is used describing the domain boundary, is mapped onto a higher-order approximation of the boundary. As a result, optimal order convergence rates for high-order finite elements are recovered and illustrated numerically.

- We extend an unfitted Eulerian time-stepping scheme from [LO19; BFM19] for moving domain problems to the time-dependent Stokes problem on a moving domain, using unfitted Taylor-Hood elements. In contrast to the existing work in [BFM19], this only requires stabilisation in the vicinity of the unfitted, moving interface and the numerical analysis includes the geometry approximation resulting from the level set approximation in CutFEM. The method is validated numerically, and we present additional numerical studies, extending the approach to higher-order and examples including topology changes.

- We derive stability and error estimates for a version of the Eulerian moving domain discretisation applied to the coupled fluid-solid problem. The stability is shown for the case of the time-dependent Stokes equation in the fluid domain, while an error estimate is proven for the reduced case of the heat equation in the bulk domain. This can be seen as a restriction of the Stokes case to spaces of divergence-free functions.

- Using data from physical experiments presented in [HTR21], we devise a fluid-structure interaction set-up, which includes contact and rebound of an elastic particle. We then use this to validate our Eulerian time-stepping method applied to a coupled fluid-solid problem.

- Most methods for FSI problems rely on a sufficiently accurate discretisation of the fluid-solid interface such that the transfer of forces can be computed accurately. In order to apply the derived Eulerian time-stepping method in cases where the geometry of the particles is neither smooth nor resolved by the background mesh, we follow an idea from [MRS21] to train an artificial deep neural network to accurately predict the transfer of forces from the fluid onto triangular shaped particles. The accuracy of the predictions results from the network's input, which is available accurately in under-resolved situations.

1.2 Overview of Publications

As a result of the research of this thesis' author as a Ph.D. student, the author worked on several research papers intended for publication in international, peer-reviewed journals. In this section we give an overview of these publications, summarise their content and discuss their impact on this thesis. Furthermore, the contributions made by the author of this thesis to each of these publications is stated.

Publication I

[Wah+19b] H. von Wahl, T. Richter, C. Lehrenfeld, J. Heiland and P. Minakowski. 'Numerical benchmarking of fluid-rigid body interactions'. In: *Comput. & Fluids* 193, 104290 (30th Oct. 2019). DOI: 10.1016/j.compfluid.2019.104290

[Wah+19a] H. von Wahl, T. Richter, C. Lehrenfeld, J. Heiland and P. Minakowski. *Numerical benchmarking of fluid-rigid body interactions*. Zenodo repository, June 2019. DOI: 10.5281/zenodo.3253455

Abstract: We propose a fluid-rigid body interaction benchmark problem, consisting of a solid spherical obstacle in a Newtonian fluid, whose centre of mass is fixed but is free to rotate. A number of different problems are defined for both two and three spatial dimensions. The geometry is chosen specifically, such that the fluid-solid partition does not change over time and classical fluid solvers are able to solve the fluid-structure interaction problem. We summarise the different approaches used to handle the fluid-solid coupling and numerical methods used to solve the arising problems. The results obtained by the described methods are presented and we give reference intervals for the relevant quantities of interest.

Relevance for this thesis and author's contribution: This publication represents the first step from classical fluid dynamics to fluid-structure interactions by considering a geometry and FSI problem accessible to classical computational fluid dynamic techniques. We shall use the benchmark described in [Wah+19b] in section 3.4 as a basis for an initial exploration and evaluation of the unfitted finite element method used in this thesis in the context of fluid-structure interaction. The author's main contributions to this paper were the design and definition of the computational scenarios as well as the organisation of the project. The computations and the descriptions of the different discretisation techniques then split naturally among the authors. The complete numerical code for the simulations by this thesis' author was published in the supplementary open source data set [Wah+19a].

Publication II

[WRL21] H. von Wahl, T. Richter and C. Lehrenfeld. 'An unfitted Eulerian finite element method for the time-dependent Stokes problem on moving domains'. In: *IMA J. Numer. Anal.* (5th July 2021). DOI: 10.1093/imanum/drab044

[WRL20] H. von Wahl, T. Richter and C. Lehrenfeld. *An unfitted Eulerian finite element method for the time-dependent Stokes problem on moving domains.* Zenodo repository, Feb. 2020. DOI: 10.5281/zenodo.3647571

Abstract: We analyse a Eulerian Finite Element method, combining a Eulerian time-stepping scheme applied to the time-dependent Stokes equations with the CutFEM approach using inf-sup stable Taylor-Hood elements for the spatial discretisation. This is based on the method introduced by Lehrenfeld & Olshanskii [ESAIM: M2AN 53(2):585–614] in the context of a scalar convection-diffusion problems on moving domains, and extended to the non-stationary Stokes problem on moving domains by Burman, Frei & Massing [arXiv:1910.03054 [math.NA]] using stabilised equal-order elements. The analysis includes the geometrical error made by integrating over approximated level set domains in the discrete CutFEM setting. The method is implemented and the theoretical results are illustrated using numerical examples.

Relevance for this thesis and author's contribution: This publication represents a major contribution to this thesis, both in terms of the numerical method developed and the numerical analysed performed in that work. In [WRL21], we moved both to unfitted finite elements and moving domain problems. The critical step taken in this work was to consider an unfitted Eulerian time-stepping approach to fluid problems in time-dependent domains by considering the time-dependent Stokes equations. This step was vital to have robust numerical analysis for an Eulerian moving domain method. An Eulerian method was necessary to work on problems where the deformation of the geometry is very large or even solid contact occurs such that an Arbitrary Lagrangian-Eulerian approach would fail. We shall go into details of the method developed in this paper in chapter 4. Much of the analysis developed here forms the base for the analysis of the Eulerian time-stepping for the coupled fluid-solid problem in chapter 5. The majority of the work in this paper was done by the thesis' author. The scripts to reproduce all the numerical examples in the paper were published in the accompanying open-source data set [WRL21].

Publication III

[Wah+21] H. von Wahl, T. Richter, S. Frei and T. Hagemeier. 'Falling balls in a viscous fluid with contact: Comparing numerical simulations with experimental data'. In: *Phys. Fluids* 33.3, 033304 (2nd Mar. 2021). doi: 10.1063/5.0037971

[Wah+20] H. von Wahl, T. Richter, S. Frei and T. Hagemeier. *Falling balls in a viscous fluid with contact: Comparing numerical simulations with experimental data.* Zenodo repository, Nov. 2020. doi: 10.5281/zenodo.3989604

Abstract: We evaluate a number of different finite element approaches for fluid-structure (contact) interaction problems against data from physical experiments. This consists of trajectories of single particles falling through a highly viscous fluid and rebounding off the bottom fluid tank wall. The resulting flow is in the transitional regime between creeping and turbulent flows. This type of configuration is particularly challenging for numerical methods due to the large change of the fluid domain and the contact between the wall and particle. In the finite element simulations we consider both rigid body and linear elasticity models for the falling particles. In the first case, we compare results obtained with the well established Arbitrary Lagrangian Eulerian (ALE) approach and an unfitted moving domain method together with a simple and common approach for contact avoidance. For the full fluid-structure interaction (FSI) problem with contact, we use a fully Eulerian approach in combination with a unified FSI-contact treatment using Nitsche's method. For higher computational efficiency we use the geometrical symmetry of the experimental set-up to reformulate the FSI system into two spatial dimensions. Finally, we show full three-dimensional ALE computations to study the effects of small perturbations in the initial state of the particle to investigate deviations from a perfectly vertical fall observed in the experiment. The methods are implemented in open-source finite element libraries and the results are made freely available to aide reproducibility.

Relevance for this thesis and author's contribution: We moved to a real fluid-structure interaction problem with solid contact in this work, backed up by experimental data. Of particular interest for this thesis is the direct comparison of fluid-rigid body model with full elastic fluid-structure interaction models and with data from physical experiments. Here we were able to apply the method developed in [WRL21] to a full Navier-Stokes fluid model. It allowed us to investigate the impact of this reduced model, consisting of rigid body dynamics and a rotationally symmetric formulation of the PDE problem, onto the relevant quantities of interest. This thesis will go into the details of this work in chapter 6, beyond that published. The author of this thesis contributed to this work by extracting a computationally rigorous definition for the fluid-structure interaction problem and by implementing of the Eulerian time-stepping discretisation used to compute to the problem posed. Again, the scripts to fully reproduce the computational results presented in [Wah+21], were made available in the zenodo repository [Wah+20].

Publication IV

[WR21] H. von Wahl and T. Richter. 'Using a deep neural network to predict the motion of under-resolved triangular rigid bodies in an incompressible flow'. In: *Internat. J. Numer. Methods Fluids* (12th Aug. 2021). doi: 10.1002/fld.5037

Abstract: We consider non-spherical rigid body particles in an incompressible fluid in the regime where the particles are too large to assume that they are simply transported with the fluid without back-coupling and where the particles are also too small to make fully resolved direct numerical simulations feasible. Unfitted finite element methods with ghost-penalty stabilisation are well suited to fluid-structure-interaction problems as posed by this setting, due to the flexible and accurate geometry handling and for allowing topology changes in the geometry. In the computationally under resolved setting posed here, accurate computations of the forces by their boundary integral formulation are not viable. Furthermore,

analytical laws are not available due to the shape of the particles. However, accurate values of the forces are essential for realistic motion of the particles. To obtain these forces accurately, we train an artificial deep neural network using data from prototypical resolved simulations. This network is then able to predict the force values based on information which can be obtained accurately in an under-resolved setting. As a result, we obtain forces on very coarse and under-resolved meshes which are on average an order of magnitude more accurate compared to the direct boundary-integral computation from the Navier-Stokes solution, leading to solid motion comparable to that obtained on highly resolved meshes that would substantially increase the simulation costs.

Relevance for this thesis and author's contribution: We took two significant steps in this paper. First, we considered non-smooth geometries for the solid bodies, and secondly, we moved away from discretisations where the fluid-solid interface is fully resolved. The novel approach of training an artificial deep neural network to steer the solid particles by using information from the bulk of the fluid was an essential step in realising simulations with solid particles, where a resolved computational mesh of very small particles does not make sense. We shall make use of this approach in section 7.2. The majority of the work in [WR21] was done by the author of this thesis.

Publication V

[Leh+21] C. Lehrenfeld, F. Heimann, J. Preuss and H. von Wahl. 'ngsxfem: Add-on to NG-Solve for geometrically unfitted finite element discretizations'. In: *J. Open Source Softw.* 6.64 (10th Aug. 2021), p. 3237. DOI: 10.21105/joss.03237. URL: https://github.com/ngsxfem/ngsxfem

Summary: `ngsxfem` is an add-on library to `Netgen/NGSolve`, a general purpose, high performance finite element library for the numerical solution of partial differential equations. The add-on enables the use of geometrically unfitted finite element technologies known under different labels, e.g. XFEM, CutFEM, TraceFEM, Finite Cell, fictitious domain method or Cut-Cell methods, etc.. Both, `Netgen/NGSolve` and `ngsxfem` are written in C++ with a rich python interface through which it is typically used. `ngsxfem` is an academic software. Its primary intention is to facilitate the development and validation of new numerical methods.

Relevance for this thesis and author's contribution: All the unfitted numerical computations presented in this thesis were implemented using `ngsxfem`. This thesis' author's main contribution to this software was in the development of the multiple level set functionality of the software. The most significant part of this was to develop the python interface to enable the user to work flexibly and efficiently with level set geometries, described by multiple level sets.

1.3 Outline of Thesis

The remainder of this thesis is structured as follows:

- Chapter 2 gives a compact overview of the equations governing the motion of an incompressible fluid coupled to a rigid body. In addition, we state several related fluid and fluid-rigid body models which we use in this thesis to develop and study our discretisation techniques.

- In chapter 3, we establish the basics of the unfitted finite element discretisation techniques used in this thesis. Here we extend an established high-order unfitted finite element discretisation to the Stokes and time-dependent Oseen problems on smooth geometries, establishing both stability and energy error-estimates. We then validate these results with numerical examples. Furthermore,

we apply this discretisation to our rotating-rigid body benchmark [Wah+19b] to analyse the approximation properties of the unfitted discretisation in the context of a fluid-rigid body interaction problem.

- We develop the unfitted Eulerian time-stepping scheme, used to compute flow problems on changing geometries, in chapter 4. Using the time-dependent Stokes equations as the flow model on a domain with known motion, we prove the stability of the approach using unfitted Taylor-Hood elements and prove an error estimate, including the geometry error introduced by considering approximated level-set geometries, which is often ignored in the unfitted finite element error analysis in the literature. We present numerical examples illustrating the robustness and approximation properties of our method. Furthermore, we present several numerical examples using higher-order discretisations in space and time, illustrating the method's capabilities beyond that covered by our numerical analysis. Finally, we apply the method to a moving domain problem with the full Navier-Stokes equations governing the motion of the fluid. In order to establish the potential of our method in this context, we compare the results against other unfitted Eulerian time-stepping methods and a highly resolved, fitted discretisation using the Arbitrary Lagrangian-Eulerian (ALE) approach.

- We move further towards fluid-rigid body interaction problems in chapter 5 by analysing a version of our unfitted Eulerian time-stepping method, using Lagrange multipliers to implement the boundary condition on the moving interface, applied to the time-dependent Stokes coupled with translational rigid body motion. We show the stability of the resulting method and prove an error estimate for the reduced case of the heat equation in the bulk domain.

- In chapter 6, we work to apply our unfitted Eulerian discretisation for moving domain flow problems to a fluid-structure interaction with solid contact, for which we have experimental data to compare our results with. To reduce the computational complexity of this problem, we use the rotational symmetry of the geometry in cylindrical coordinates to derive a spatially reduced formulation of the problem. We give details of our solver applied to coupled fluid-solid problems and compute the fluid-structure interaction problem in the reduced formulation, using a basic contact avoidance algorithm to deal with the rebound problem posed by the set-up.

- Chapter 7 then deals with non-smooth geometries for the rigid body. Using triangles as the prototype for a rigid body with corners, resulting in a fluid domain with re-entrant corners, we investigate the ability of our method in a context where standard ALE approaches will fail. To efficiently simulate multiple triangular particles in a fluid in free-fall, we develop a deep neural network approach to predict the transmission forces acting on the particles. The neural network is trained to use volumetric information from the fluid as input such that we can then apply the network in highly under-resolved unfitted finite element simulations with multiple particles.

- We conclude this thesis in chapter 8 by summarising the results obtained and giving an overview of possible future research directions resulting from this work.

Incompressible Fluids with Rigid Body Motion

Structure of Chapter   The mathematical equations used to model the motion of rigid bodies in an incompressible fluid, and on which our numerical models and discretisations are based, are introduced. Furthermore, details of simplified equations used in this thesis to develop and analyse the presented methods are given.

Contents of Chapter

2.1 The Incompressible Navier-Stokes Equations and Rigid Body Motion

Let us consider a rigid body moving through a liquid. We denote the solid domain as $\mathcal{S} \subset \mathbb{R}^d$, with $d \in \{2, 3\}$, and assume that the fluid $\mathcal{F} = \mathbb{R}^d \setminus \overline{\mathcal{S}}$ fills the whole space. We assume that the rigid body $\mathcal{S}$ is an open, connected and bounded domain. Then the interface between the solid and the fluid region is denoted as $\mathcal{I} = \partial \mathcal{S} \subset \mathbb{R}^{d-1}$.

2.1.1 Fluid Model

We shall consider the *incompressible Navier-Stokes equations* as the model for the fluid. Let us therefore consider the fluid space-time domain, given by

$$\mathcal{Q} := \bigcup_{t \in (0, t^{\mathrm{end}})} \mathcal{F}(t) \times \{t\}.$$

In this domain and in an inertial frame, the fluid is described by a velocity $\boldsymbol{u}$ and a pressure p. Following the principles of conservation of linear momentum and mass, the velocity and pressure fulfil the incompressible Navier-Stokes equations given by

$$\rho_{\mathcal{F}} \frac{D\boldsymbol{u}}{Dt} - \operatorname{div}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = \rho_{\mathcal{F}} \boldsymbol{f} \tag{2.1a}$$

$$\operatorname{div}(\boldsymbol{u}) = 0. \tag{2.1b}$$

Here $\rho_{\mathcal{F}}$ is the homogeneous fluid density and

$$\frac{D}{Dt} := \partial_t + \boldsymbol{u} \cdot \nabla, \qquad \boldsymbol{\sigma}(\boldsymbol{u}, p) := \mu_{\mathcal{F}}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T) - \operatorname{Id} p$$

are the total material derivative and the Cauchy stress tensor, respectively, where Id is the identity tensor in $\mathbb{R}^d$ and $\boldsymbol{f}$ is an external body force acting on the fluid. Furthermore, $\mu_{\mathcal{F}} = \rho_{\mathcal{F}}\nu$ is the fluid's dynamic viscosity while $\nu > 0$ is the fluids kinematic viscosity. Taking the divergence constraint into account, the momentum balance equation in (2.1) can be rewritten as

$$\rho_{\mathcal{F}}\partial_t \boldsymbol{u} - \mu_{\mathcal{F}}\Delta\boldsymbol{u} + \rho_{\mathcal{F}}\boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla p = \boldsymbol{f}.$$

The system (2.1) is then completed with appropriate boundary conditions. On the temporal boundary, this is an initial condition $\boldsymbol{u}(\cdot,0) = \boldsymbol{u}_0(\cdot)$. On the fluid solid interface, we assume continuity of the velocity, i.e., we have the inhomogeneous Dirichlet condition $\boldsymbol{u}(\cdot,t) = U(t)$ on $\mathcal{I}(t)$ for the rigid velocity $U(t) \in \mathbb{R}^d$. Furthermore, we assume that the liquid is at rest at infinity, i.e., we impose

$$\lim_{|\boldsymbol{x}|\to\infty} \boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{0}. \tag{2.2}$$

2.1.2 Solid Model

Let $\mathcal{S}$ be a rigid body with homogeneous density $\rho_{\mathcal{S}}$. We define $c_{\mathcal{S}}$ to be the centre of mass of this body with respect to the origin of our inertial frame. The velocity of the solid is then given by

$$U = \boldsymbol{v} + \boldsymbol{\omega}\times\boldsymbol{r},$$

where $\boldsymbol{v} = \partial_t c_{\mathcal{S}}$ is the solid's translational velocity, $\boldsymbol{\omega}$ is the solid's angular velocity and $\boldsymbol{r} = \boldsymbol{x} - c_{\mathcal{S}}$ is a given point on the surface of the solid $\mathcal{I}$ relative to the bodies centre of mass. These two velocity components are then governed by the Newton-Euler equations

$$m_{\mathcal{S}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{v} = \boldsymbol{F} + \boldsymbol{F}_{\text{buoyant}} + \boldsymbol{F}_{\text{gravity}} \tag{2.3a}$$

$$\boldsymbol{I}_{\mathcal{S}}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\omega} + \boldsymbol{\omega}\times\boldsymbol{I}_{\mathcal{S}}\boldsymbol{\omega} = \boldsymbol{T} + \boldsymbol{T}_{\text{buoyant}} \tag{2.3b}$$

with the particles mass $m_{\mathcal{S}} = \rho_{\mathcal{S}}\,\text{vol}(\mathcal{S})$, the force and torque exerted by the fluid on the particle

$$\boldsymbol{F} = -\int_{\mathcal{I}}\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n}\,\mathrm{d}s \qquad\text{and}\qquad \boldsymbol{T} = -\int_{\mathcal{I}}\boldsymbol{r}\times\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n}\,\mathrm{d}s, \tag{2.4}$$

where $\boldsymbol{n}$ is the fluids outward pointing unit normal vector, and the solids moment of inertia tensor defined by

$$\boldsymbol{I}_{\mathcal{S}} = \rho_{\mathcal{S}}\int_{\mathcal{S}}(\|\boldsymbol{r}\|^2\,\text{Id}_3 - \boldsymbol{r}\otimes\boldsymbol{r})\,\mathrm{d}\boldsymbol{x}.$$

The pull due to gravity and the buoyancy force and torque are

$$\boldsymbol{F}_{\text{gravity}} = m_{\mathcal{S}}\boldsymbol{g}, \qquad \boldsymbol{F}_{\text{buoyant}} = -m_{\mathcal{F}}\boldsymbol{g} \qquad\text{and}\qquad \boldsymbol{T}_{\text{buoyant}} = -m_{\mathcal{F}}\boldsymbol{r}_{bo}\times\boldsymbol{g},$$

where $m_{\mathcal{F}} = \rho_{\mathcal{F}}\,\text{vol}(\mathcal{S})$ is the mass of the displaced fluid and $\boldsymbol{r}_{bo}$ is the vector from the centre of mass to the centre of buoyancy. The centre of buoyancy is defined as the centroid of the displaced fluid volume. Since the particles are completely submersed and both the fluid and solid have a constant density, the centre of mass and centre of buoyancy coincide, such that $\boldsymbol{r}_{bo} = 0$ and therefore $\boldsymbol{T}_{\text{buoyant}} = 0$.

Remark 2.1.   Note that due to the Helmholtz decomposition, c.f. [Gal11, Theorem III.1.1], the buoyancy effects can equivalently be included in the system by including the forces due to gravity on the right-hand side of (2.1) in the form $\boldsymbol{g}\nabla\boldsymbol{x}_d$. As the numerical methods in this thesis are not pressure-robust [Joh+17], including buoyancy in the solid and ignoring gravity on the fluid is more accurate on the discrete level.   ▲

Remark 2.2.   In two spatial dimensions, the non-linear, quadratic term $\boldsymbol{\omega}\times\boldsymbol{I}_{\mathcal{S}}\boldsymbol{\omega}$ in (2.3b) vanishes. Furthermore, the moment of inertia reduces to the scalar quantity $I_{\mathcal{S}} = \rho_{\mathcal{S}}\int_{\mathcal{S}}\|\boldsymbol{r}\|^2\,\mathrm{d}\boldsymbol{x}$ and the angular velocity is also a scalar quantity.   ▲

2.1.3 Full Model

The full model of an incompressible fluid contains a rigid body in free-fall is then given as follows: With the system starting at rest, i.e., $\boldsymbol{u}_0 = U_0 = \boldsymbol{0}$, we search for a fluid velocity and pressure $(\boldsymbol{u}, p)$ and a solid velocity U such that

$$\rho_{\mathcal{F}}(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}) - \operatorname{div}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = \boldsymbol{0} \qquad \text{in } \mathcal{Q} \qquad (2.5a)$$

$$\operatorname{div}(\boldsymbol{u}) = 0 \qquad \text{in } \mathcal{Q} \qquad (2.5b)$$

$$\boldsymbol{u}(\cdot, t) = U(\cdot, t) \qquad \text{on } \mathcal{I}(t) \qquad (2.5c)$$

$$U(\boldsymbol{x}, t) = \boldsymbol{v}(t) + \boldsymbol{\omega}(t) \times (\boldsymbol{x} - c_{\mathcal{S}}(t)) \qquad \text{on } \mathcal{I}(t) \qquad (2.5d)$$

$$m_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{v} = \boldsymbol{F} + \boldsymbol{F}_{\text{buoyant}} + \boldsymbol{F}_{\text{gravity}} \qquad \text{in } (0, t^{\text{end}}) \qquad (2.5e)$$

$$\boldsymbol{I}_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{\omega} + \boldsymbol{\omega} \times \boldsymbol{I}_{\mathcal{S}} \boldsymbol{\omega} = \boldsymbol{T} + \boldsymbol{T}_{\text{buoyant}} \qquad \text{in } (0, t^{\text{end}}), \qquad (2.5f)$$

where the fluid is at rest at infinity, c.f. (2.2).

Changing the Frame of Reference

When considering the system (2.5), the most significant problem is that the fluid domain is an unknown function of time. Therefore, it is advantageous to rewrite the coupled fluid-solid problem in a frame attached to the solid $\mathcal{S}$, in which the fluid domain remains constant over time. To this end, we follow [Gal02]. Without loss of generality, consider a frame $\mathfrak{R}'$, whose origin coincides with the centre of mass of the solid $c_{\mathcal{S}}$ and which coincides with the inertial frame of reference at time $t = 0$. Then, if $\boldsymbol{x}$ is the position vector of a point P and $\boldsymbol{y}$ denotes that same point in $\mathfrak{R}'$, it holds that

$$\boldsymbol{x} = \boldsymbol{Q}(t)\boldsymbol{y} + c_{\mathcal{S}}(t), \qquad \boldsymbol{Q}(0) = \text{Id}, \qquad c_{\mathcal{S}}(0) = \boldsymbol{0}. \qquad (2.6)$$

Using that in the solid it holds that $\dot{c}_{\mathcal{S}}(t) = \boldsymbol{v}$ and $\dot{\boldsymbol{x}} = U$, it follows from (2.6) and (2.5d) that the angular velocity is related to $\boldsymbol{Q}$ via

$$\dot{\boldsymbol{Q}}(t)\boldsymbol{Q}^T(t)\boldsymbol{a} = \boldsymbol{\omega} \times \boldsymbol{a} \qquad \text{for all } \boldsymbol{a} \in \mathbb{R}^3. \qquad (2.7)$$

To rewrite the system (2.5) in the frame of reference $\mathfrak{R}'$, we define the following transformed fields for the fluid and solid equations. For the fluid, we set

$$\mathsf{u}(\boldsymbol{y}, t) := \boldsymbol{Q}^T(t)\boldsymbol{u}(\boldsymbol{Q}(t)\boldsymbol{y} + c_{\mathcal{S}}(t), t), \quad \mathsf{p}(\boldsymbol{y}, t) := p(\boldsymbol{Q}(t)\boldsymbol{y} + c_{\mathcal{S}}(t), t), \quad \mathsf{T}(\mathsf{u}, \mathsf{p}) := \boldsymbol{Q}^T \boldsymbol{\sigma}(\boldsymbol{Q}\mathsf{u}, \mathsf{p})\boldsymbol{Q}.$$

For the solid, we further define

$$\boldsymbol{\zeta}(t) := \boldsymbol{Q}^T(t)\boldsymbol{v} \qquad \text{and} \qquad \boldsymbol{\xi}(t) := \boldsymbol{Q}^T(t)\boldsymbol{\omega}.$$

It then follows from (2.7) that

$$\boldsymbol{Q}^T(t)\dot{\boldsymbol{Q}}(t)\boldsymbol{a} = \boldsymbol{\xi} \times \boldsymbol{a} \qquad \text{for all } \boldsymbol{a} \in \mathbb{R}^3. \qquad (2.8)$$

We now need to rewrite the fluid and solid equations in terms of these transformed fields. For the time-derivative of the fluid velocity, we have

$$\partial_t \boldsymbol{u} = \partial_t(\boldsymbol{Q}\mathsf{u}) = \dot{\boldsymbol{Q}} + \boldsymbol{Q}(\partial_t \mathsf{u} + \dot{\boldsymbol{y}} \cdot \nabla_{\boldsymbol{y}}\mathsf{u}).$$

Differentiating (2.6) with respect to time, taking the definitions of $\mathsf{u}, \boldsymbol{\zeta}$ and $\boldsymbol{\xi}$ into account and applying (2.7), we also have that

$$\dot{\boldsymbol{y}} = \boldsymbol{Q}^T(\boldsymbol{u} - \boldsymbol{v}) - \boldsymbol{Q}^T \dot{\boldsymbol{Q}}\boldsymbol{y} = \mathsf{u} - \boldsymbol{\zeta} - \boldsymbol{\xi} \times \boldsymbol{y}.$$

Combining the above equations then gives

$$\boldsymbol{Q}^T \partial_t \boldsymbol{u} = \boldsymbol{\xi} \times \mathsf{u} + \partial_t \mathsf{u} + (\mathsf{u} - \boldsymbol{\zeta} - \boldsymbol{\xi} \times \boldsymbol{y}) \cdot \nabla_{\boldsymbol{y}} \mathsf{u}.$$

Furthermore, by the definition of the transformed fields, it follows that

$$\operatorname{div}_{\boldsymbol{x}}(\boldsymbol{u}) = \operatorname{div}_{\boldsymbol{y}}(\mathsf{u}) \qquad \text{and} \qquad \boldsymbol{Q}^T \operatorname{div}_{\boldsymbol{x}}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = \operatorname{div}_{\boldsymbol{y}}(\mathsf{T}(\mathsf{u}, \mathsf{p})).$$

For the Cauchy stress-tensor, we observe that due to (2.6) and the definition of the transformed velocity it holds

$$\nabla_{\boldsymbol{x}} \boldsymbol{u} + (\nabla_{\boldsymbol{x}} \boldsymbol{u})^T = \boldsymbol{Q}^T \big(\nabla_{\boldsymbol{y}} \mathsf{u} + (\nabla_{\boldsymbol{y}} \mathsf{u})^T\big) \boldsymbol{Q}.$$

Then by the definition of the transformed pressure, it follows that

$$\boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{Q}(\nabla_{\boldsymbol{y}} \mathsf{u} + (\nabla_{\boldsymbol{y}} \mathsf{u})^T) \boldsymbol{Q}^T - \operatorname{Id} \mathsf{p}.$$

So by the definition of the transformed stress tensor, we see that the Cauchy stress tensor has the same form in both frames of reference, i.e., $\mathsf{T}(\mathsf{u}, \mathsf{p}) = \mu_{\mathcal{F}}(\nabla \mathsf{u} + \nabla \mathsf{u}^T) - \operatorname{Id} \mathsf{p}$.

Now, let $\mathcal{D}$ be the fixed fluid region in the frame $\mathfrak{R}'$. As a result of the above considerations, we have that the Navier-Stokes equations in the frame moving with the rigid body are given by

$$\rho_{\mathcal{F}}(\partial_t \mathsf{u} + (\mathsf{u} - \mathsf{U}) \cdot \nabla \mathsf{u} + \boldsymbol{\xi} \times \mathsf{u}) - \operatorname{div}(\mathsf{T}(\mathsf{u}, \mathsf{p})) = \boldsymbol{Q}^T \boldsymbol{f} \qquad \text{in } \mathcal{D} \times (0, t^{\text{end}})$$

$$\operatorname{div}(\mathsf{u}) = 0 \qquad \text{in } \mathcal{D} \times (0, t^{\text{end}}),$$

where

$$\mathsf{U}(\boldsymbol{y}, t) = \boldsymbol{\zeta}(t) + \boldsymbol{\xi}(t) \times \boldsymbol{y}$$

is the total velocity of the rigid body in the moving frame. The boundary conditions are then

$$\lim_{|\boldsymbol{y}| \to \infty} \boldsymbol{u}(\boldsymbol{y}, t) = 0 \qquad \text{and} \qquad \mathsf{u}(\boldsymbol{y}, t) = \mathsf{U}(\boldsymbol{y}, t) \text{ on } \mathcal{I}' \times (0, t^{\text{end}}),$$

where $\mathcal{I}'$ is the solid boundary in the moving frame.

To complete the system, we now rewrite the solid equations in $\mathfrak{R}'$. By the definition of the transformed solid translational velocity, we have with (2.8) and the orthogonality of $\boldsymbol{Q}$ that

$$m_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{v} = m_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} (\boldsymbol{Q}\boldsymbol{\zeta}) = m_{\mathcal{S}} (\dot{\boldsymbol{Q}}\boldsymbol{\zeta} + \boldsymbol{Q}\dot{\boldsymbol{\zeta}}) = m_{\mathcal{S}} \boldsymbol{Q}(\dot{\boldsymbol{\zeta}} + \boldsymbol{\xi} \times \boldsymbol{\zeta}).$$

Denoting n as the unit normal vector pointing out of the fluid, we have that $\boldsymbol{n} = \boldsymbol{Q}\mathsf{n}$, and for the forces acting from the fluid onto the solid that

$$\int_{\mathcal{I}} \boldsymbol{\sigma}(\boldsymbol{u}, p) \boldsymbol{n} \, \mathrm{d}s = \boldsymbol{Q} \int_{\mathcal{I}'} \mathsf{T}(\mathsf{u}, \mathsf{p}) \mathsf{n} \, \mathrm{d}s.$$

The equation of linear momentum (2.3a) governing the translational velocity component of the solid is therefore transformed as

$$m_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{\zeta} + m_{\mathcal{S}} \boldsymbol{\xi} \times \boldsymbol{\zeta} = \boldsymbol{Q}^T \boldsymbol{F}_{\text{ext}} - \int_{\mathcal{I}'} \mathsf{T}(\mathsf{u}, \mathsf{p}) \mathsf{n} \, \mathrm{d}s,$$

where $\boldsymbol{F}_{\text{ext}}$ is the sum of the external forces acting on the solid. To rewrite the equation of angular momentum (2.3b), we use the identity $(\boldsymbol{Q}\boldsymbol{a}) \times (\boldsymbol{Q}\boldsymbol{b}) = \boldsymbol{Q}(\boldsymbol{a} \times \boldsymbol{b})$, so that with $\mathsf{I} := \boldsymbol{Q}^T \boldsymbol{I}_{\mathcal{S}} \boldsymbol{Q}$ we find

$$\mathsf{I} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{\xi} + \boldsymbol{\xi} \times \mathsf{I} \boldsymbol{\xi} = \boldsymbol{Q}^T \boldsymbol{T}_{\text{ext}} - \int_{\mathcal{I}'} \boldsymbol{y} \times \mathsf{T}(\mathsf{u}, \mathsf{p}) \mathsf{n} \, \mathrm{d}s.$$

Now, in order to reformulate the free-fall fluid-rigid body problem (2.5) in the frame moving with the rigid body, we first note that since the motion of the solid is not known, we also do not know the direction of the vector $\boldsymbol{G}(t) \coloneqq \boldsymbol{Q}^T(t)\boldsymbol{g}$. Differentiating this with respect to time, and using the orthogonality of $\boldsymbol{Q}$ and that $\dot{\boldsymbol{g}} = \boldsymbol{0}$, we find with (2.8) that

$$\dot{\boldsymbol{G}} = \dot{\boldsymbol{Q}}^T \boldsymbol{g} = \dot{\boldsymbol{Q}}^T \boldsymbol{Q} \boldsymbol{G} = -\boldsymbol{Q}^T \dot{\boldsymbol{Q}} \boldsymbol{G} = \boldsymbol{G} \times \boldsymbol{\xi}.$$

As a result of the above considerations, we are now able to reformulate the full free fall fluid-rigid body system (2.5) in the frame moving with the solid body. Given the initial conditions $\mathbf{u}(\cdot, 0) = \boldsymbol{\zeta}(0) = \boldsymbol{\xi}(0) = \boldsymbol{0}$ and $\boldsymbol{G} = \boldsymbol{g}$, find $(\mathbf{u}, \mathsf{p}, \boldsymbol{\zeta}, \boldsymbol{\xi}, \boldsymbol{G})$ such that

$$\rho_\mathcal{S}\big(\partial_t \mathbf{u} + (\mathbf{u} - \mathsf{U}) \cdot \nabla \mathbf{u} + \boldsymbol{\xi} \times \mathbf{u}\big) - \mu_\mathcal{F} \Delta \mathbf{u} + \nabla \mathsf{p} = \rho_\mathcal{S} \boldsymbol{G} \qquad \text{in } \mathcal{D} \times (0, t^{\text{end}}) \quad (2.9\text{a})$$

$$\text{div}(\mathbf{u}) = 0 \qquad \text{in } \mathcal{D} \times (0, t^{\text{end}}) \quad (2.9\text{b})$$

$$\mathbf{u}(\cdot, t) = \mathsf{U}(\cdot, t) \qquad \text{on } \mathcal{I}' \quad (2.9\text{c})$$

$$\mathsf{U}(\boldsymbol{y}, t) = \boldsymbol{\zeta}(t) + \boldsymbol{\xi}(t) \times \boldsymbol{y} \qquad \text{on } \mathcal{I}' \quad (2.9\text{d})$$

$$m_\mathcal{S} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{\zeta} + m_\mathcal{S} \boldsymbol{\xi} \times \boldsymbol{\zeta} = m_\mathcal{S} \boldsymbol{G} - \int_{\mathcal{I}'} \mathbf{T}(\mathbf{u}, \mathsf{p}) \mathbf{n} \, \mathrm{d}s \qquad \text{in } (0, t^{\text{end}}) \quad (2.9\text{e})$$

$$\mathbf{I} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{\xi} + \boldsymbol{\xi} \times \mathbf{I}\boldsymbol{\xi} = - \int_{\mathcal{I}'} \boldsymbol{y} \times \mathbf{T}(\mathbf{u}, \mathsf{p}) \mathbf{n} \, \mathrm{d}s \qquad \text{in } (0, t^{\text{end}}) \quad (2.9\text{f})$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{G} = \boldsymbol{G} \times \boldsymbol{\xi} \qquad \text{in } (0, t^{\text{end}}), \quad (2.9\text{g})$$

where the fluid is at rest at infinity. Note that in this form, we do not include the buoyancy effect in the solid equations, since we include gravity in the fluid momentum balance equation.

Solvability

To arrive at a statement on the existence of solutions to (2.9), we need to derive a weak formulation of the problem. To this end, let $\mathcal{D}_T \coloneqq \mathcal{D} \times [0, t^{\text{end}})$ and define the space

$$\mathcal{C} \coloneqq \left\{ \boldsymbol{\varphi} \in \boldsymbol{C}^\infty(\overline{\mathcal{D}_T}) \; \middle| \; \begin{array}{c} \text{div}(\boldsymbol{\varphi}(\cdot, t)) = 0 \text{ for all } t \in [0, t^{\text{end}}), \text{ there exist } \boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2 \in C_0^\infty([0, t^{\text{end}})) \\ \text{such that } \boldsymbol{\varphi}(\boldsymbol{y}, t) = -(\boldsymbol{\varphi}_1(t) + \boldsymbol{\varphi}_2(t) \times \boldsymbol{y}) \\ \text{in a neighbourhood of } \mathcal{I}' \text{ and for all } t \in [0, t^{\text{end}}) \end{array} \right\}.$$

Multiplying the fluid momentum balance equation (2.9a) with a test-function $\boldsymbol{\varphi} \in \mathcal{C}$, integrating with respect to $\mathcal{D}_T$, integrating by parts with respect to time and space and inserting the balance of linear and angular momentum for the solid then leads to the weak form

$$\rho_\mathcal{F} \int_0^{t^{\text{end}}} \int_\mathcal{D} -\frac{\partial \boldsymbol{\varphi}}{\partial t} \cdot \boldsymbol{u} - ((\mathbf{u} - \mathsf{U}) \cdot \nabla \boldsymbol{\varphi}) \cdot \boldsymbol{u} + \boldsymbol{\varphi} \cdot \boldsymbol{\xi} \times \mathbf{u} + 2\nu \boldsymbol{D}(\boldsymbol{\varphi}) : \boldsymbol{D}(\boldsymbol{u}) \, \mathrm{d}\boldsymbol{y} \, \mathrm{d}t$$

$$= \int_0^{t^{\text{end}}} m_\mathcal{S} \frac{\mathrm{d}\boldsymbol{\varphi}_1}{\mathrm{d}t} \cdot \boldsymbol{\zeta} + \frac{\mathrm{d}\boldsymbol{\varphi}_2}{\mathrm{d}t} \cdot \mathbf{I}\boldsymbol{\xi} - m_\mathcal{S} \boldsymbol{\xi} \times \mathbf{u} \cdot \boldsymbol{\varphi}_1 - \boldsymbol{\xi} \times (\mathbf{I}\boldsymbol{\xi}) \cdot \boldsymbol{\varphi}_2 + (m_\mathcal{S} - m_\mathcal{F})\boldsymbol{G} \cdot \boldsymbol{\varphi}_1 \, \mathrm{d}t, \quad (2.10)$$

where $\boldsymbol{D}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$ is the velocity deformation tensor. Furthermore, for (2.9g) we derive with $\boldsymbol{\psi} \in C_0^\infty([0, t^{\text{end}}))$ the weak form

$$\boldsymbol{\psi}(0) \cdot \boldsymbol{g} + \int_0^{t^{\text{end}}} \frac{\mathrm{d}\boldsymbol{\psi}}{\mathrm{d}t} \cdot \boldsymbol{G} + \boldsymbol{G} + \boldsymbol{\xi} \times \boldsymbol{G} \cdot \boldsymbol{\psi} \, \mathrm{d}t = 0. \quad (2.11)$$

Notation   To formulate the statement on the existence, we require the notion of some function spaces. In the following, for $\Omega \subset \mathbb{R}^d$, we will use the standard notation for *Sobolev* spaces $\mathcal{W}^{k,p}(\Omega)$ for scalar valued, weakly differentiable functions with the associated norm $\|\cdot\|_{\mathcal{W}^{k,p}(\Omega)}$ and semi-norm $|\cdot|_{\mathcal{W}^{k,p}(\Omega)}$ for $k \in \mathbb{N}_0$ and $1 \leq p \leq \infty$, [Eva98; EG21a]. As above, spaces and norms of vector valued functions are indicated by using bold letters. From these spaces, we obtain the Hilbert space $\mathcal{W}^{k,2}(\Omega) = \mathcal{H}^k(\Omega)$ and the Lebesgue space by $\mathcal{W}^{0,p}(\Omega) = \mathcal{L}^p(\Omega)$. The $\mathcal{L}^2(\Omega)$-inner product is then denoted by $(\cdot, \cdot)_\Omega$ and the resulting natural $\mathcal{L}^2(\Omega)$-norm is abbreviated to $\|\cdot\|_\Omega$. Additionally, we denote the subspaces $\mathcal{L}_0^2(\Omega)$ as the space of $\mathcal{L}^2$-integrable functions with mean zero and $\mathcal{H}_0^1(\Omega)$ as the set of $\mathcal{H}^1$-functions with vanishing trace on $\partial\Omega$. We further note that notion of Sobolev spaces can be extended to so-called fractional order spaces for non-integer k. Finally, for time-dependent problems, we denote *Bochner* spaces by

$$\mathcal{L}^p(t_1, t_2; X) := \left\{ f(x, t) \mid \int_{t_1}^{t_2} \|f\|_X^p \, \mathrm{d}t < \infty \right\}$$

for a normed space X and $p \in [0, \infty]$, with the standard modification for $p = \infty$, [EG21c, Part XIII, Chapter 64]. Where there is no risk of ambiguity in t_1, t_2, we will use the shorter notation $\mathcal{L}^p(X)$ for $\mathcal{L}^p(t_1, t_2; X)$.

With these formulations we can therefore formulate the existence of Leray-Hopf weak solutions to the coupled free fall fluid-rigid body problem.

Theorem 2.3.   *There exists at least one weak solution $(\boldsymbol{u}, \boldsymbol{\zeta}, \boldsymbol{\xi}, \boldsymbol{G})$ of the free fall fluid-rigid body problem, such that $\boldsymbol{u} \in \mathcal{L}^2(0, t^{end}, \boldsymbol{\mathcal{H}}^1(\mathcal{D}))$, $\mathsf{u} = \mathsf{U} = \boldsymbol{\zeta} + \boldsymbol{\xi} \times \boldsymbol{y}$ on $\mathcal{I}' \times [0, t^{end}]$ in the sense of traces and (2.10) and (2.11) hold for all $\boldsymbol{\varphi} \in \mathcal{C}$ and $\boldsymbol{\psi} \in C_0^\infty([0, t^{end}))$, respectively.*

Proof. See [Gal02, Part I, section 4]. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

We further refer to [GGH12] for the the $\mathcal{L}^p$-theory for strong solutions of the coupled fluid-rigid body system (2.5).

Remark 2.4.   In the above formulation, the fluid domain fills the entire space, which is not filled by the solid, and we assumed that the fluid is at rest at infinity. The consideration of an unbounded fluid domain was chosen to avoid the possibility of contact between a part of the fluid boundary and the solid. In fact, the correct model for the transition to contact with a solid wall is not yet fully understood. For details on the current state of research for contact modelling from a theoretical perspective, see section 6.2 below. As this thesis deals with computational methods for fluid-rigid body interactions, we will only consider bounded fluid domains below. We, therefore, are mainly concerned with looking at the system before topological contact occurs between the fluid and a solid boundary of the fluid domains.   ▲

2.2   Related and Simplified Models

In the development of our numerical methods, we also consider a number of simplified models of the fluid-rigid body system, given by (2.5). In models where the motion of the fluid domain is a given quantity, and there are no solid equations to be considered, we shall refer to the fluid domain as Ω, in line with the standard notation in computational fluid dynamics (CFD).

Navier-Stokes on Moving Domains

As a simplification of the full fluid-rigid body system (2.5), we take the solid motion as a given quantity. The resulting equations are the Navier-Stokes equations on a moving domain. We consider a domain $\Omega(t) \subset \mathbb{R}^d$ with prescribed motion and define the space-time domain as $\mathcal{Q} := \bigcup_{t \in (0, t^{\text{end}})} \Omega(t) \times \{t\}$. In $\mathcal{Q}$

we then consider the Navier-Stokes equations: Given an initial condition $\boldsymbol{u}_0$, find a velocity and pressure $(\boldsymbol{u}, p)$ such that

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \tag{2.12a}$$

$$\nabla \cdot \boldsymbol{u} = 0, \tag{2.12b}$$

together with homogeneous Dirichlet boundary conditions on the spatial boundary $\Gamma(t) = \partial\Omega(t)$. We refer to [Boc77; Sal88; GS06] for details on the existence of solutions to this problem.

Transient Stokes with Translational Body Motion

As an alternative simplification of the full fluid-rigid body system, we ignore the non-linear transport term and restrict the solid motion to translational motion. The result is a system of linear and coupled fluid-solid equations. To this end, let $\Omega = \mathcal{F}(t) \dot\cup \mathcal{I} \dot\cup \mathcal{S}$ be a bounded and connected domain in $\mathbb{R}^d$, and let $\Gamma = \partial\Omega$ be the outer boundary of the fluid domain. Given a divergence-free initial fluid condition $\boldsymbol{u}_0$ with a compatible initial solid condition U_0, we search for a fluid velocity and pressure $(\boldsymbol{u}, p)$ and a solid velocity U such that

$$\rho_{\mathcal{F}} \partial_t \boldsymbol{u} - \operatorname{div}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = \rho_{\mathcal{F}} \boldsymbol{f} \qquad\qquad \text{in } \mathcal{Q} \tag{2.13a}$$

$$\operatorname{div}(\boldsymbol{u}) = 0 \qquad\qquad \text{in } \mathcal{Q} \tag{2.13b}$$

$$\boldsymbol{u}(\cdot, t) = 0 \qquad\qquad \text{on } \Gamma \tag{2.13c}$$

$$\boldsymbol{u}(\cdot, t) = \boldsymbol{v}(t) \qquad\qquad \text{on } \mathcal{I}(t) \tag{2.13d}$$

$$m_{\mathcal{S}} \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{v} = \boldsymbol{F} + \boldsymbol{F}_{\text{buoyant}} + \boldsymbol{F}_{\text{gravity}} \qquad\qquad \text{in } (0, t^{\text{end}}). \tag{2.13e}$$

See [Gal02, section 4.1] for details on the existence and uniqueness of the time-dependent Stokes equations coupled to rigid body motion of a solid body.

Transient Stokes Equations on Moving Domains

As a further simplifications, we consider the transient Stokes problem on a moving domain with prescribed motion. With a time-dependent domain $\Omega(t) \subset \mathbb{R}^d$ bounded in space, we take the space time-domain $\mathcal{Q} := \bigcup_{t \in (0, t^{\text{end}})} \Omega(t) \times \{t\}$. In $\mathcal{Q}$ we then consider the time-dependent Stokes problem: Find the velocity $\boldsymbol{u}$ and pressure p such that

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \tag{2.14a}$$

$$\nabla \cdot \boldsymbol{u} = 0, \tag{2.14b}$$

together with Dirichlet boundary conditions on the space-boundary $\Gamma(t) = \partial\Omega(t)$, an initial condition $\boldsymbol{u}(\cdot, 0) = \boldsymbol{u}_0(\cdot)$ and a forcing term $\boldsymbol{f}(t)$ with the viscosity $\nu > 0$. For the well-posedness of this problem, we refer to [BFM19, Section 2.1].

Remark 2.5. The time-dependent Stokes equations are not a natural model to consider. If we takes a creeping flow as the fluid model, such that the convective forces are negligible compared to the diffusion forces, then the time-derivative can also be ignored, i.e., the complete material derivative can be ignored. As a result, at every time t, the stationary Stokes equations should be considered in $\mathcal{F}(t)$; see for example [HB81]. We do not consider the time-dependent Stokes equations on moving domains as a physical model of rigid body motion, but rather a simplification of the Navier-Stokes equations to consider a linear model, easing the numerical analysis of discretisations, on the way to the full fluid-rigid body model. ▲

The Navier-Stokes Equations

As we take the incompressible Navier-Stokes equations as our fluid model, we shall also be interested in the time-dependent *Navier-Stokes* equations on a fixed fluid domain. To this end, let $\Omega \subset \mathbb{R}^d$ with $d \in \{2, 3\}$ be an connected, bounded and fixed domain with Lipschitz continuous boundary. The incompressible Navier-Stokes equations are then given by

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad \text{in } \Omega \times (0, t^{\text{end}}) \qquad (2.15\text{a})$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \times (0, t^{\text{end}}) \qquad (2.15\text{b})$$

$$\boldsymbol{u} = 0 \qquad \text{on } \Gamma \times (0, t^{\text{end}}) \qquad (2.15\text{c})$$

$$\boldsymbol{u}(\cdot, 0) = \boldsymbol{u}_0(\cdot) \qquad \text{in } \Omega. \qquad (2.15\text{d})$$

Provided sufficient regularity of the initial data $\boldsymbol{u}_0$ and the body force $\boldsymbol{f}$, we know existence of weak solutions $(\boldsymbol{u}, p)$ to the Navier-Stokes equations (2.15). Uniqueness is only known in the case of $d = 2$, while for $d = 3$ uniqueness of the solution is an open problem. For details, see for example [BF13, Chapter V, section 1.3] or [Gal00].

The Time-Dependent Oseen Equations

As a simplification of the Navier-Stokes equations, we shall also consider the *time-dependent Oseen equations*. Here the transport field is taken to be a given quantity, rather than the velocity itself. For a given, divergence-free convection field, the problem is to find a velocity and pressure $(\boldsymbol{u}, p)$ such that

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{b} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad \text{in } \Omega \times (0, t^{\text{end}}) \qquad (2.16\text{a})$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \times (0, t^{\text{end}}) \qquad (2.16\text{b})$$

$$\boldsymbol{u} = 0 \qquad \text{on } \Gamma \times (0, t^{\text{end}}) \qquad (2.16\text{c})$$

$$\boldsymbol{u}(\cdot, 0) = \boldsymbol{u}_0(\cdot) \qquad \text{in } \Omega. \qquad (2.16\text{d})$$

See [Deu21] and the references therein for the theoretical existence, uniqueness and long-time behaviour of the solution to (2.16).

The Stokes Equations

A further simplification is to remove the transport term and assume a stationary solution such that the time-derivative vanishes. This models a creeping flow where the diffusion forces dominate the inertia forces. In this case, we end up with the *Stokes problem*

$$-\nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad \text{in } \Omega \qquad (2.17\text{a})$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } \Omega \qquad (2.17\text{b})$$

$$\boldsymbol{u} = 0 \qquad \text{on } \Gamma. \qquad (2.17\text{c})$$

See for example [BF13, Chapter IV, section 5.1] or [Gal11, Chapter IV] for the existence and uniqueness of solutions to problem (2.17).

Unfitted Finite Elements for Flow Problems

Structure of Chapter   The unfitted finite element method used in this work is introduced. Central concepts relating to the high-order geometry approximation and stabilisation techniques are summarised from the available literature. The available analysis for the high-order geometry approximation of the isoparametric CutFEM approach is extended to the Stokes problem on fixed domains using Taylor-Hood elements. This analysis is further extended to the time-dependent flow problems on stationary domains using the transient Oseen problem. The theoretical results are illustrated and validated numerically using high-order finite elements. The first step towards fluid-structure interactions is made by applying the isoparametric CutFEM to a benchmark problem proposed in our work [Wah+19b], in which the fluid-solid geometry does not change in time.

Contents of Chapter

3.1   Introduction to CutFEM

CutFEM is an unfitted finite element method based on a description of the computational domain by a level set function on some larger background domain [BH10; BH12; BH14; Bur+14; Mas+14]. The idea is to integrate on the computational domain by generating custom quadrature rules on cut elements, while
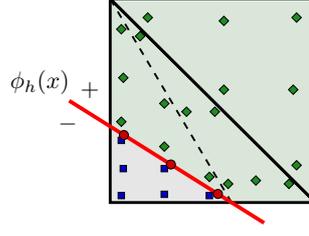
Figure 3.1: Quadrature points for an integration rule of order four on cut and uncut element with respect to a straight cut.

using the standard polynomial basis on the entire element, on which the computational domain has some contribution. Boundary conditions inside an unfitted finite element can then be implemented using a Lagrange multiplier approach [BH10] or more commonly by using Nitsche's method [Nit71] as presented in [BH12; BH14; Mas+14]. To ensure numerical stability of this method, independent of the mesh level set cut configuration, ghost-penalty stabilisation [Bur10] has become very popular.

Related *unfitted finite element methods*, known under a variety of names, include XFEM [MDB99; FB10], the Finite Cell method [PDR07], TraceFEM [ORG09] and ϕ-FEM [DL20]. We note that the concepts of these methods have been used before, for example in penalty methods [Bab73b; BE86], the fictitious domain method [GPP94a; GPP94b], the immersed boundary method [PM89]. Finally, we note that a CutFEM method without ghost-penalty stabilisation [BHL21] has been developed very recently.

3.1.1 Geometry Description with Level Sets

Let us consider a open bounded domain $\Omega \subset \mathbb{R}^d$, for $d \in \{2, 3\}$, with boundary $\Gamma = \partial\Omega$. We assume that this is embedded into a larger polygonal domain $\widetilde{\Omega} \supset \Omega$ in $\mathbb{R}^d$, which we will denote as the *background domain*. The boundary Γ is then described by a *level set* function $\phi \colon \widetilde{\Omega} \to \mathbb{R}$ so that

$$\Gamma = \{\boldsymbol{x} \in \widetilde{\Omega} \mid \phi(\boldsymbol{x}) = 0\} \qquad \text{and} \qquad \Omega = \{\boldsymbol{x} \in \widetilde{\Omega} \mid \phi(\boldsymbol{x}) < 0\}.$$

This level set function is taken to be sufficiently smooth, that is in a neighbourhood U_G of Γ we have that $\phi \in C^{m+1}(U_\Gamma)$ for some $m \in \mathbb{N}_{>1}$.

In order to avoid explicitly writing generic constants in inequalities, we will use the notation $a \lesssim b$, iff there exists a constant $c > 0$ independent of the mesh size and the mesh-interface cut position such that $a \leq cb$. Similarly, $a \gtrsim b$ denotes $a \geq cb$. If both $a \lesssim b$ and $a \gtrsim b$ hold, then we will write $a \simeq b$.

Now, we consider a simplicial, shape-regular and quasi-uniform triangulation $\widetilde{\mathcal{T}_h}$ of the background domain $\widetilde{\Omega}$. We will refer to this as the *background mesh*. The parameter h denotes the diameter of the mesh for which we we have $h \simeq h_T$ with the individual element diameter h_T for $T \in \widetilde{\mathcal{T}_h}$.

In order to achieve higher-order convergence of the method using the isoparametric approach [Leh16; Leh17] below, we assume that there is a sufficiently good approximation of the smooth level set function $\phi_h^{\text{ho}} \in \mathbb{P}^k$ in the space of continuous and piecewise polynomial functions of order k on $\widetilde{\mathcal{T}_h}$. This assumptions translates as

$$\|\phi - \phi_h^{\text{ho}}\|_{U_\Gamma,\infty} + h\|\nabla\phi - \nabla\phi_h^{\text{ho}}\|_{U_\Gamma,\infty} \lesssim h^{k+1},$$

where $\|\cdot\|_{X,\infty}$ denotes the $\mathcal{L}^\infty(X)$-norm and we assume that $m \geq k$, and k is the order of our finite element space.

Explicit Boundary Description   An explicit description of the domain is necessary for a robust generation of quadrature rules on cut elements, see [Bur+14, Section 5]. To this end, we consider a piecewise

linear level approximation $\phi_h \in \mathbb{P}^1$. With this we can then explicitly define the approximated domain boundary

$$\Gamma_h := \{\boldsymbol{x} \in \widetilde{\Omega} \mid \phi_h(\boldsymbol{x}) = 0\}.$$

This piecewise linear domain boundary results in straight cuts on each elements, such that with a simple sub-triangulation, quadrature rules can be generated to integrate over the approximated interface and computational domain. An example of the quadrature points generated by this procedure can be seen in Figure 3.1. The approximated computational domain is then given by

$$\Omega_h := \{\boldsymbol{x} \in \widetilde{\Omega} \mid \phi_h(\boldsymbol{x}) < 0\}.$$

The main disadvantage of this *geometry approximation* is that it introduces an error of order h^2 on the cut elements. As a result methods based on this can be at most of order two. The method we shall deploy to obtain a geometry error of the correct order will be presented bellow.

Cut Element Notation We introduce the notation we shall use to denote the different parts of the background mesh used in CutFEM. We define the *active mesh*

$$\mathscr{T}_h := \{T \in \widetilde{\mathscr{T}_h} \mid T \cap \Omega_h \neq \emptyset\}$$

as the set of elements which contain some contribution to the approximated domain Ω_h. We then take the set of *cut elements*

$$\mathscr{T}_{h,\Gamma} := \{T \in \mathscr{T}_h \mid T \cap \Gamma_h \neq \emptyset\}$$

as the elements containing the boundary approximation. The set of uncut, or interior, elements is then denoted as

$$\mathscr{T}_{h,\text{in}} := \mathscr{T}_h \setminus \mathscr{T}_{h,\Gamma}.$$

Furthermore, we consider the interior extension of cut elements by direct facet neighbours

$$\mathscr{T}_{h,\Gamma,+} := \{T \in \mathscr{T}_h \mid \exists T' \in \mathscr{T}_{h,\Gamma} \text{ with } \text{meas}_{d-1}(\overline{T} \cap \overline{T'}) \neq 0\}.$$

With this, we can then define the set of interior facets of cut elements as

$$\mathscr{F}_h := \{F = \overline{T}_1 \cap \overline{T}_2 \mid T_1, T_2 \in \mathscr{T}_{h,\Gamma,+} \text{ with } T_1 \neq T_2 \text{ and } \text{meas}_{d-1}(F) > 0\}.$$

Finally, we define the *active domain* as

$$\mathcal{O}_{\mathscr{T}} := \{\boldsymbol{x} \in T \mid T \in \mathscr{T}_h\} \subset \mathbb{R}^d$$

and the *domain of cut elements*

$$\mathcal{O}_{\mathscr{T}_\Gamma} := \{\boldsymbol{x} \in T \mid T \in \mathscr{T}_{h,\Gamma}\} \subset \mathbb{R}^d.$$

To achieve higher-order geometry approximations of the discrete level set domain, we assume that h is sufficiently small, so that ϕ is smooth enough in $\mathcal{O}_{\mathscr{T}_\Gamma}$, i.e., $\mathcal{O}_{\mathscr{T}_\Gamma} \subset U_\Gamma$. In Figure 3.2 we illustrate these sets of elements and facets for a given mesh and level set approximation.

For the analysis below, we assume that the interface Γ intersects the boundary of every element $T \in \mathscr{T}_{h,\Gamma}$ exactly twice and each cut open edge is intersected exactly once.

3.1.2 Ghost-Penalty Stabilisation

In fictitious domain methods, we integrate over a level set domain on a background mesh constructed independently of the level set geometry. As a result, it occurs that there are elements $T \in \mathscr{T}_h$ in the active mesh with very little contribution to the computational domain, i.e., $\text{meas}_d(\Omega_h \cap T) \leq \varepsilon$ for some
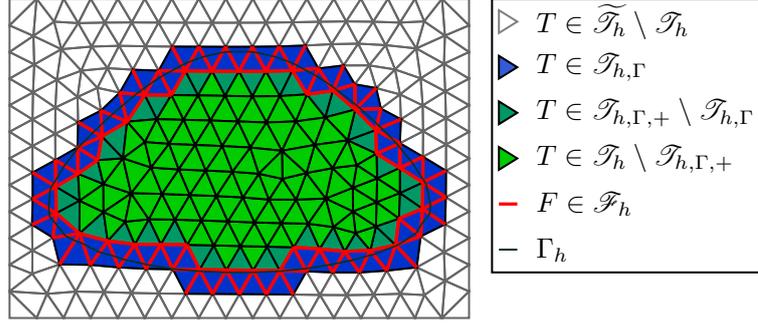
Figure 3.2: A background mesh, piecewise linear level set function and the element sets corresponding to the resulting cut configuration.

small $\varepsilon > 0$. We refer to such mesh-level set cut configurations as *bad cuts*. Elements with bad cuts in the active mesh do not affect the accuracy of the numerical integration over Ω_h. However, the condition number of the resulting finite element stiffness matrix is severely impacted by bad cuts, affecting the accuracy and convergence properties of solvers of the linear system. In fact, for $\varepsilon \to 0$, the condition number of the stiffness matrix becomes unbounded [Bur10].

To improve the robustness of fictitious domain methods with respect to the mesh-level set cut configuration, ghost-penalty stabilisation was introduced by BURMAN [Bur10]. The basic idea is to include a penalty term in the variational formulation of a given PDE problem, which controls the solution on cut elements, by taking the solution on neighbouring interior elements into account. We refer to [GM19; GSM20] for a more abstract and extensive approach to designing a ghost-penalty operator for a given problem. As a result, ghost-penalty stabilisation gives control over the solution in norms defined on the entire active domain. Additional benefits include that we are able to prove bounds for the condition number of the resulting linear systems and that it can provide the necessary pressure stabilisation on cut cells needed to guarantee solvability.

To the best of our knowledge, there are currently three different types of ghost-penalty operator used in the literature.

Local Projection Type Ghost-Penalties   This version of the ghost-penalty operator was introduced in the original paper [Bur10]. For $F \in \mathscr{F}_h$ consider the facet patch $\omega_F = T_1 \cup T_2$ for $\overline{T}_1 \cap \overline{T}_2 = F$. Then the ghost-penalty bilinear form is defined facet-wise as

$$j_h^{\mathrm{LPS}}(u, v) \coloneqq \gamma_{gp} \sum_{F \in \mathscr{F}_h} j_{h,F}^{\mathrm{LPS}}(u, v),$$

where

$$j_{h,F}^{\mathrm{LPS}}(u, v) \coloneqq \frac{1}{h^2} \int_{\omega_F} (u - \Pi_{\omega_F} u) \cdot (v - \Pi_{\omega_F} v) \, \mathrm{d}\boldsymbol{x} = \frac{1}{h^2} \int_{\omega_F} (u - \Pi_{\omega_F} u) v \, \mathrm{d}\boldsymbol{x}$$

for $u, v \in \mathcal{L}^2(\mathcal{O}_{\mathscr{T}_{h,\Gamma,+}})$. Here $\Pi_{\omega_F} : \mathcal{L}^2(\omega_F) \to \mathbb{P}^k(\omega_F)^d$ is the $\mathcal{L}^2$-projection onto the space of polynomials of order k. We shall not consider this form here.

Normal-Derivative Jump Ghost-Penalties   The most widely used ghost-penalty operator in the literature is the *derivative jump* version. A non exhaustive list of publications using this includes [BH12; BH14; Leh17; Mas+14; GO17; MSW18; SW14; GM19; BFM19]. This version has the form

$$j_h^{\mathrm{djump}}(u, v) = \sum_{F \in \mathscr{F}_h} \sum_{l=1}^{k} \gamma_{gp,l} \frac{h^{2l-1}}{l!} \int_F [\![\partial_{\boldsymbol{n}}^l u]\!] \, [\![\partial_{\boldsymbol{n}}^l v]\!] \, \mathrm{d}s$$

for $u, v \in V_h$, where $[\![w]\!] = w^+ - w^-$ with $w(x)^{\pm} = \lim_{t \to 0} w(x \pm t\boldsymbol{n})$ denotes the jump operator across a facet F. This version is not well suited for higher-order methods, i.e., $k > 1$, since this requires the computation of higher-order normal-derivatives, as well as a choice for each of the k stability parameters. We note however, that in the majority of the literature, only the case $k = 1$ is considered, in which the $\mathbb{P}^1$ geometry error also does not play a role.

Direct Version of Ghost-Penalties The so-called *direct version* of the ghost penalty operator, introduced by Preuss [Pre18] is the variant we shall focus on in our numerical analysis and use in our numerical computations. Works in the literature which use this version include [LO19; WRL21; Wah+21; OQS21; AB21]. Again let $\omega_F = T_1 \cup T_2$ again denote a facet patch around a facet $F \in \mathscr{F}_h$. For $u, v \in V_h$ we define the *direct* ghost-penalty operator as

$$j_h^{\mathrm{dir}}(u, v) := \gamma_{gp} \sum_{F \in \mathscr{F}_h} j_{h,F}^{\mathrm{dir}}(u, v), \tag{3.1}$$

where

$$j_{h,F}^{\mathrm{dir}}(u, v) := \frac{1}{h^2} \int_{\omega_F} (u_1 - u_2)(v_1 - v_2)\, \mathrm{d}\boldsymbol{x}.$$

Here we denote $u_i = \mathcal{E}^{\mathbb{P}}(u|_{T_i})$ and $v_i = \mathcal{E}^{\mathbb{P}}(v|_{T_i})$ for $i \in \{1, 2\}$, with the canonical extension of polynomials $\mathcal{E}^{\mathbb{P}} : \mathbb{P}(T) \to \mathbb{P}(\mathbb{R}^d)$ from an element T to $\mathbb{R}^d$. Compared to the other versions of the ghost penalty operator above, this formulation has the advantage of only being implicitly dependent on the polynomial order k (through the extension operator $\mathcal{E}^{\mathbb{P}}$) and only requiring a single choice for the penalty parameter γ_{gp}.

Properties of the Ghost-Penalty Operator

We cover the mechanism and the most important theoretical properties of the ghost-penalty stabilisation used here. The basic principle is to give control norms in the active domain $\mathcal{O}_{\mathscr{T}}$. Since we will focus on the direct version of the ghost-penalty operator, we shall define the volume based jump operator on a facet patch $\omega_F = T_1 \cup T_2$ as

$$[\![v]\!]_{\omega_F} := v_1 - v_2 \tag{3.2}$$

with $v_i = \mathcal{E}^{\mathbb{P}}(u|_{T_i})$ and the canonical extension of polynomials $\mathcal{E}^{\mathbb{P}}$ as above.

Lemma 3.1 (*Ghost-penalty mechanism*). *Let $\omega_F = T_1 \cup T_2$ be a facet patch over a facet $F = \overline{T}_1 \cap \overline{T}_2$. Let v be a piecewise polynomial function defined relative to the macro-element $\overline{T}_1 \cup \overline{T}_2$ and v_i be the restriction of v to T_i. Then there exists a constant $c_{L3.1} > 0$ dependent only on the shape-regularity of T and the polynomial order of v such that*

$$\|v\|_{T_1}^2 \leq c_{L3.1} \left(\|v\|_{T_2}^2 + \|[\![v]\!]_{\omega_F}\|_{T_1}^2 \right). \tag{3.3}$$

Proof. See [Pre18, Lemma 3.1]. $\qquad\qquad\square$

Lemma 3.2. *The direct ghost penalty operator can be bound from above by both the local projection and normal-derivative ghost-penalty operators*

$$j_h^{dir}(v, v) \lesssim j_h^{djump}(v, v) \qquad and \qquad j_{h,F}^{dir}(v, v) \lesssim j_{h,F}^{LPS}(v, v).$$

Proof. For the normal-derivative bound, see [Pre18, Ch. 3, Remark 6] and for the local projection bound, see the proof of [LO19, Lemma 5.2]. $\qquad\qquad\square$

Lemma 3.3. *For all $v \in V_h$ for $k \geq 1$, we have for the direct ghost-penalty operator that*

$$\|v\|^2_{\mathcal{O}_{\mathcal{F}}} \lesssim \left(\|v\|_{\Omega} + h^2 j_h^{dir}(v, v) \right) \lesssim \|v\|^2_{\mathcal{O}_{\mathcal{F}}} \tag{3.4}$$

$$\|\nabla v\|^2_{\mathcal{O}_{\mathcal{F}}} \lesssim \left(\|\nabla v_h\|_{\Omega} + j_h^{dir}(v, v) \right) \lesssim \|\nabla v\|^2_{\mathcal{O}_{\mathcal{F}}} \tag{3.5}$$

with constants independent of the mesh-level set cut position.

Proof. For the first inequality we split the norm into a sum over cut and uncut elements

$$\|v_h\|^2_{\mathcal{O}_{\mathcal{F}}} = \sum_{T \in \mathcal{T}_{h,\Gamma}} \|v_h\|^2_T + \sum_{T \in \mathcal{T}_{h,\text{in}}} \|v_h\|^2_T.$$

Now let $T \in \mathcal{T}_{h,\Gamma}$ be a cut element. We assume that the boundary Γ is sufficiently well resolved by the mesh such that we only have to cross a finite number of facets $F_i = \overline{T}_{i-1} \cap \overline{T}_i$ with $T_0 = T$ to reach an interior element $T_N \in \mathcal{T}_{h,\text{in}}$. We refer to [LO19, Remark 5.2] for a detailed explanation as to why this assumption is reasonable. Using the ghost-penalty mechanism, we then have

$$\|v\|^2_T \lesssim \|v\|^2_{T_N} + \sum_{i=0}^{N-1} \|[\![v]\!]_{\omega_F}\|^2_{T_i} \lesssim \|v\|^2_{T_N} + \sum_{i=1}^{N} \|[\![v]\!]_{\omega_F}\|^2_{\omega_F} \lesssim \|v\|^2_{T_N} + h^2 j_h^{dir}(v, v).$$

The second inequality follows from Lemma 3.2 and the properties of the other ghost-penalty operators. For the gradient bound we follow the same lines and use an inverse inequality to get $h^2 j_{h,F}^{\text{dir}}(\nabla v, \nabla v) \lesssim j_{h,F}^{\text{dir}}(v, v)$. □

In the later analysis, it will be necessary to insert general $\mathcal{L}^2$-functions into the ghost-penalty operator. In this case we take $u_i = \mathcal{E}^{\mathbb{P}}(\Pi_{T_i} u|_{T_i})$, where Π_{T_i} is the $\mathcal{L}^2(T_i)$-projection onto $\mathbb{P}^k(T)$.

Lemma 3.4 (*Consistency of direct ghost-penalties*). *Let $v \in \mathcal{H}^{k+1}(\mathcal{O}_{\mathcal{F}})$ and $w \in \mathcal{H}^k(\mathcal{O}_{\mathcal{F}})$ for some $k \geq 1$. Then it holds that*

$$j_h^{dir}(v, v) \lesssim h^{2k} \|v\|_{\mathcal{H}^{k+1}(\mathcal{O}_{\mathcal{F}})}$$

$$h^2 j_h^{dir}(w, w) \lesssim h^{2k} \|w\|_{\mathcal{H}^k(\mathcal{O}_{\mathcal{F}})}.$$

Furthermore, let $\mathcal{I}_h^*$ be the Scott-Zhang interpolation operator [SZ90] for the appropriate finite element space under consideration. Then we also have

$$j_h^{dir}(v - \mathcal{I}_h^* v, v - \mathcal{I}_h^* v) \lesssim h^{2k} \|v\|_{\mathcal{H}^{k+1}(\mathcal{O}_{\mathcal{F}})}$$

$$h^2 j_h^{dir}(w - \mathcal{I}_h^* w, w - \mathcal{I}_h^* w) \lesssim h^{2k} \|w\|_{\mathcal{H}^k(\mathcal{O}_{\mathcal{F}})}.$$

Proof. See [LO19, Lemma 5.8]. The result for $h^2 j_h^{\text{dir}}(\cdot, \cdot)$ follows by the same line of argument and taking the h-scaling into account. □

3.1.3 Higher-Order Geometry Approximation

As noted above, the $\mathbb{P}^1$ level set approximation, used in CutFEM to generating quadrature rules on cut elements, leads to a geometry approximation error of order h^2, in cases where the level set function is not already (piecewise) linear. As a result, higher-order finite element methods, i.e., where the polynomial order of the finite element space is chosen as $k \geq 2$, optimal order error convergence in the $\mathcal{L}^2$-norm cannot be expected. In order to recover optimal convergence for higher-order finite elements, including the lowest order inf-sup stable Taylor-Hood elements $\mathbb{P}^2/\mathbb{P}^1$ [TH73], we shall use an isoparametric mapping approach introduced by LEHRENFELD [Leh16], of which we will give a brief overview in this section.

We follow [Leh17] for this overview, where the method was described and analysed for the Poisson problem.

To construct a high order domain approximation, we consider a special parametric mapping $\Theta_h \in (\mathbb{P}^k)^d$ on the mesh $\widetilde{\mathscr{T}_h}$, which we take to be a bijection on $\widetilde{\Omega}$. For details on the construction of this mapping we refer to [Leh16]. With this explicit mesh transformation, we can then explicitly construct a high order domain approximation

$$\Omega_h^{\mathrm{ho}} \coloneqq \{\Theta_h(\boldsymbol{x}) \mid \boldsymbol{x} \in \Omega_h\} \qquad \text{and} \qquad \Gamma_h^{\mathrm{ho}} \coloneqq \{\Theta_h(\boldsymbol{x}) \mid \boldsymbol{x} \in \Gamma_h\}.$$

Furthermore, we assume to have an ideal one-to-one transformation $\Psi \colon \widetilde{\Omega} \to \widetilde{\Omega}$, which maps the piecewise linear domain approximation onto the *exact* domain, i.e.,

$$\Omega \coloneqq \{\Psi(\boldsymbol{x}) \mid \boldsymbol{x} \in \Omega_h\} \qquad \text{and} \qquad \Gamma \coloneqq \{\Psi(\boldsymbol{x}) \mid \boldsymbol{x} \in \Gamma_h\}.$$

For the parametric mapping we then have the following results.

Lemma 3.5. *For sufficiently small $h > 0$ we have for the parametric mapping Θ_h and the exact transformation Ψ that*

$$\Theta_h(\boldsymbol{x}) = x \text{ for } \boldsymbol{x} = \boldsymbol{x}_V \text{ a vertex in } \mathscr{T}_h \text{ or } \boldsymbol{x} \in \mathcal{O}_{\mathscr{T}} \setminus \mathcal{O}_{\mathscr{T}_{\Gamma,+}}$$
$$\|\Theta_h - \mathrm{id}\|_\infty \lesssim h^2 \qquad \|D\Theta_h - I\|_\infty \lesssim h$$
$$\|\Theta_h - \Psi\|_{\mathcal{L}^\infty(\mathcal{O}_{\mathscr{T}_\Gamma})} + h\|D(\Theta_h - \Psi)\|_{\mathcal{L}^\infty(\mathcal{O}_{\mathscr{T}_\Gamma})} \lesssim h^{k+1} \qquad \text{i.e.,} \quad \mathrm{dist}(\Gamma, \Gamma_h^{ho}) \lesssim h^{k+1}.$$

Proof. See [LR17, Lemmata 3.5, 3.7 and 3.8]. $\qquad\square$

From this lemma, we can see that Θ_h is a small perturbation of the identity. Since the mapping essentially repairs the geometry approximation error made by the piecewise linear level set approximation, this perturbation vanishes for $h \to 0$. Additionally, we have that the mapping and its inverse are bounded in higher order derivatives.

Lemma 3.6. *For $T \in \mathscr{T}_h$ we have for the parametric mapping that*

$$\|\Theta_h\|_{m,\infty,T} \lesssim 1 \qquad and \qquad \|\Theta_h^{-1}\|_{m,\infty,\Theta_h(T)} \lesssim 1 \qquad for \quad m \in \{1,\ldots,k+1\}$$

with the norm $\|\cdot\|_{m,\infty,T} \coloneqq \max_{l \le m} \|D^l \cdot\|_{\mathcal{L}^\infty(T)}$.

Proof. See [Leh17, Lemma 2]. $\qquad\square$

Also, the $\mathcal{L}^2$- and $\mathcal{H}^1$-norms on the discrete domain and the higher-order domain are equivalent.

Lemma 3.7. *For $v, w \in \mathcal{H}^1(\Omega_h^{ho})$ with $\mathrm{tr}(\nabla w) \in \mathcal{L}^2(\Gamma_h^{ho})$ it holds*

$$\|v \circ \Theta_h\|_{\Omega_h} \simeq \|v\|_{\Omega_h^{ho}} \qquad and \qquad \|v \circ \Theta_h\|_{\Gamma_h} \simeq \|v\|_{\Gamma_h^{ho}}$$
$$\|\nabla(w \circ \Theta_h)\|_{\Omega_h} \simeq \|\nabla w\|_{\Omega_h^{ho}} \qquad and \qquad \|\nabla(w \circ \Theta_h)\|_{\Gamma_h} \simeq \|\nabla w\|_{\Gamma_h^{ho}}.$$

Proof. See [Leh17, Lemma 3]. $\qquad\square$

In the isoparametric finite element discretisation, the discrete problem will be posed on the discrete domain Ω_h^{ho}, while the original problem is posed on the domain Ω. For the analysis we therefore require a smooth bijective mapping Φ_h between these two domains, such that $\Phi_h(\Gamma_h^{\mathrm{ho}}) = \Gamma$. The piecewise smooth function $\Phi_h \coloneqq \Psi \circ \Theta_h^{-1}$ has this property and is again a small perturbation of the identity.

Lemma 3.8. *For $h > 0$ sufficiently small, $\Phi_h := \Psi \circ \Theta_h^{-1} \colon \widetilde{\Omega} \to \widetilde{\Omega}$ defines a homeomorphism with $\Phi_h \in C(\widetilde{\Omega}) \cap C^{k+1}(\Theta_h(\widetilde{\mathscr{T}_h}))$, which fulfils*

$$\Phi_h(\Omega_h^{ho}) = \Omega$$

$$\|\Phi_h - \mathrm{id}\|_{\mathcal{L}^\infty(\widetilde{\Omega})} + h\|D\Phi_h - I\|_{\mathcal{L}^\infty(\widetilde{\Omega})} \lesssim h^{k+1}.$$

Proof. See [LR17, Lemma 5.5]. □

3.2 CutFEM for the Stokes Problem

We now consider the isoparametric CutFEM method applied to the Stokes equation (2.17). CutFEM was first applied to the Stokes equation by BURMAN and HANSBO [BH14] and MASSING et al. [Mas+14]. For stability of this method, pressure-stabilisation with ghost-penalties is necessary, even for elements which are inf-sup stable on uncut elements. In this context, GUZMÁN and OLSHANSKII [GO17] presented the stability of this discretisation with various inf-sup stable elements. The analysis in these papers assumed that the discrete domain Ω_h and the smooth domain Ω match exactly. In the following section we will present and analyse the CutFEM finite element method with ghost-penalty stabilisation using Taylor-Hood elements conjunction with the parametric mapping Θ_h as introduced by LEHRENFELD [Leh16].

Before we begin with the discretisation, we give the weak formulation for the Stokes problem (2.17).

Problem P1. Find a velocity and pressure $(\boldsymbol{u}, p) \in \boldsymbol{\mathcal{H}}_0^1(\Omega) \times \mathcal{L}_0^2(\Omega)$ such that for all $(\boldsymbol{v}, q) \in \boldsymbol{\mathcal{H}}_0^1(\Omega) \times \mathcal{L}_0^2(\Omega)$ it holds

$$a(\boldsymbol{u}, \boldsymbol{v}) + b(p, \boldsymbol{v}) + b(q, \boldsymbol{u}) = f(\boldsymbol{v}). \tag{3.6}$$

The bilinear and linear forms are thereby given by

$$a(\boldsymbol{u}, \boldsymbol{v}) := \nu \int_\Omega \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}, \qquad b(q, \boldsymbol{v}) := -\int_\Omega q \nabla \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} \quad \text{and} \quad f(\boldsymbol{v}) := \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}. \tag{3.7}$$

The bilinear form $a(\cdot, \cdot)$ is continuous and coercive and the bilinear form $b(\cdot, \cdot)$ is continuous and fulfils an inf-sup condition, c.f. [Joh16]. For a detailed study of these equations, we refer to [Gal11, Chapter IV].

3.2.1 Discretisation

For the isoparametric CutFEM discretisation of the Stokes equations, we begin by defining the relevant spaces on the background mesh. With the discrete mapping discussed in subsection 3.1.3, consider the following isoparametric finite element spaces

$$\boldsymbol{V}_h^{\mathrm{iso}} := \{\boldsymbol{v}_h \circ \Theta_h^{-1} \mid \boldsymbol{v}_h \in [\mathbb{P}^k]^d\} \qquad \text{and} \qquad Q_h^{\mathrm{iso}} := \{q_h \circ \Theta_h^{-1} \mid q_h \in \mathbb{P}^{k-1}\}$$

for $k \geq 2$ on the active part of the mesh. To allow for the evaluation of the normal-derivative of velocity functions and the normal of pressure functions on the discrete boundary, we further introduce the following infinite dimensional spaces

$$\boldsymbol{V}_h^{\mathrm{reg}} := \boldsymbol{\mathcal{H}}^2(\Omega_h^{\mathrm{ho}}) \cap \{\boldsymbol{v} \circ \Phi_h \mid \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^2(\Omega)\} \qquad \text{and} \qquad Q_h^{\mathrm{reg}} := \mathcal{H}^1(\Omega_h^{\mathrm{ho}}) \cap \{q \circ \Phi_h \mid q \in \mathcal{H}^1(\Omega)\}.$$

For the unfitted, isoparametric finite element method we use a Nitsche variational formulation. The discrete problem then reads:

Problem P2. Find $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ such that for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ there holds that

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + \nu i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + b_h(p_h, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h) - {}^1\!/\!\nu j_h(p_h, q_h) = f_h(\boldsymbol{v}_h). \tag{3.8}$$

Here, the diffusion bilinear form is

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) := \nu \int_{\Omega_h^{\mathrm{ho}}} \nabla \boldsymbol{u}_h : \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \nu N_h(\boldsymbol{u}_h, \boldsymbol{v}_h)$$

with the Nitsche terms

$$N_h(\boldsymbol{u}_h, \boldsymbol{v}_h) := N_h^c(\boldsymbol{u}_h, \boldsymbol{v}_h) + N_h^c(\boldsymbol{v}_h, \boldsymbol{u}_h) + N_h^s(\boldsymbol{u}_h, \boldsymbol{v}_h), \qquad \text{where}$$

$$N_h^c(\boldsymbol{u}_h, \boldsymbol{v}_h) := -\int_{\Gamma_h^{\mathrm{ho}}} \partial_{\boldsymbol{n}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}s \qquad \text{and} \qquad N_h^s(\boldsymbol{u}_h, \boldsymbol{v}_h) := \frac{\sigma}{h} \int_{\Gamma_h^{\mathrm{ho}}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}s.$$

Here, $\sigma > 0$ is the penalty parameter which must scale with k^2 due to an inverse trace estimate [WH03]. We will specify the exact value later. The pressure bilinear form is

$$b_h(q_h, \boldsymbol{v}_h) := -\int_{\Omega_h^{\mathrm{ho}}} q_h \nabla \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + N_h^c(q_h, \boldsymbol{v}_h), \qquad \text{where} \qquad N_h^c(q_h, \boldsymbol{v}_h) := \int_{\Gamma_h^{\mathrm{ho}}} (q_h \boldsymbol{n}) \cdot \boldsymbol{v}_h \, \mathrm{d}s.$$

Note that in our notation, we do not distinguish between the viscosity and pressure Nitsche consistency terms. The arguments make it clear which is meant. For the stabilising form, we use the direct ghost-penalty operator (3.1). The diffusion and pressure stabilisation terms are then

$$i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) := j_h^{\mathrm{dir}}(\boldsymbol{u}_h, \boldsymbol{v}_h) \qquad \text{and} \qquad j_h(p_h, q_h) := h^2 j_h^{\mathrm{dir}}(\boldsymbol{u}_h, \boldsymbol{v}_h).$$

The different h-scaling in the velocity and pressure ghost-penalties, is due to the fact that the velocity is stabilised in an $\mathcal{H}^1$-sense, while the pressure is stabilised in the $\mathcal{L}^2$-sense.

We then collect all the bilinear forms from the partial differential equation together into

$$A_h((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) := a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + b_h(p_h, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h)$$

and the stabilising bilinear forms into

$$J_h((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) := \nu i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) - 1/\nu j_h(p_h, q_h).$$

As a final note, in case of non-homogenous Dirichlet boundary data $\boldsymbol{u}_D$, we modify the right-hand side

$$f_h(\boldsymbol{v}_h, q_h) := \int_{\Omega_h^{\mathrm{ho}}} \mathcal{E} \boldsymbol{f} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \nu N_h^c(\boldsymbol{v}_h, \boldsymbol{u}_D) + \nu N_h^s(\boldsymbol{u}_D, \boldsymbol{v}_h) + N_h^c(q_h, \boldsymbol{u}_D),$$

where $\mathcal{E} \colon \Omega \to \Omega_h^{\mathrm{ho}}$ is a smooth extension operator, the details of which we discuss below.

3.2.2 Notation and Basic Results

Norms For the analysis of the numerical method, we require some further notation. With the discrete bilinear forms defined above, we define the natural norms resulting thereof as

$$\|\|\boldsymbol{v}\|\|_a^2 := \|\nabla \boldsymbol{v}\|_{\Omega_h^{\mathrm{ho}}}^2 + \|h^{1/2} \partial_{\boldsymbol{n}} \boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}}^2 + \|h^{-1/2} \boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}}^2 \qquad \text{and} \qquad \|\|q\|\|_b^2 := \|q\|_{\Omega_h^{\mathrm{ho}}}^2 + \|h^{1/2} q\|_{\Gamma_h^{\mathrm{ho}}}^2,$$

as well as norms defined via the entire active domain as

$$\|\|\boldsymbol{v}\|\|_*^2 := \|\nabla \boldsymbol{v}\|_{\mathcal{O}_{\mathscr{T}}}^2 + \|h^{-1/2} \boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}}^2, \qquad \|\|v\|\|_{*,-1} := \sup_{\boldsymbol{v} \in \boldsymbol{V}_h^{\mathrm{iso}}} \frac{(\boldsymbol{v}, \boldsymbol{w})_{\Omega_h^{\mathrm{ho}}}}{\|\|\boldsymbol{w}\|\|_*} \qquad \text{and} \qquad \|\|q\|\|_*^2 := \|q\|_{\mathcal{O}_{\mathscr{T}}}^2.$$

We again do not distinguish the notation between the velocity and pressure. The argument makes it clear which is meant. For the compound velocity-pressure space we then further define

$$\|\|(\boldsymbol{v}, q)\|\|_A^2 := \|\|\boldsymbol{v}\|\|_a^2 + \|\|q\|\|_b^2 \qquad \text{and} \qquad \|\|(\boldsymbol{v}, q)\|\|_*^2 := \|\|\boldsymbol{v}\|\|_*^2 + \|\|q\|\|_*^2.$$

Estimates   We collect a number of cut versions of inverse and trace estimates. For $v_h \in \mathbb{P}^k(T)$, $T \in \widetilde{\mathscr{T}_h}$, we have the inverse and trace estimates:

$$\|\nabla v_h\|_T \lesssim h_T^{-1}\|v_h\|_T, \tag{3.9a}$$

$$\|h^{1/2}\partial_{\boldsymbol{n}} v_h\|_F \lesssim \|\nabla v_h\|_T, \tag{3.9b}$$

$$\|h^{1/2}\partial_{\boldsymbol{n}} v_h\|_{T \cap \Gamma_h} \lesssim \|\nabla v_h\|_T. \tag{3.9c}$$

For (3.9a) and (3.9b) see for example [Qua17]. For a proof of (3.9c), see [HH02]. For $v \in \mathcal{H}^1(T)$, $T \in \mathscr{T}_h$, we also have the following trace inequality from [HH02]

$$\|v_h\|_{T \cap \Gamma_h} \lesssim h_T^{-1/2}\|v_h\|_T + h_T^{1/2}\|\nabla v_h\|_T. \tag{3.10}$$

Furthermore, we have a discrete version of the Poincaré inequality

$$\|\boldsymbol{v}_h\|_{\mathcal{O}_{\mathscr{T}}} \leq c_{P,h}|\!|\!|\boldsymbol{v}_h|\!|\!|_* \tag{3.11}$$

for all $\boldsymbol{v}_h \in \boldsymbol{V}_h$, see [Mas+14, Lemma 7.2] for a proof thereof. Due to the equivalence of norms in Lemma 3.7, the same results hold true for functions from the isomaparemtric spaces $\boldsymbol{V}_h^{\mathrm{iso}}, Q_h^{\mathrm{iso}}$ and on the deformed mesh elements $\Theta_h(T)$, with the high-order level set boundary Γ_h^{ho}. Therefore, it follows that

$$|\!|\!|\boldsymbol{v}_h|\!|\!|_a \lesssim |\!|\!|\boldsymbol{v}_h|\!|\!|_* \qquad \text{and} \qquad |\!|\!|q_h|\!|\!|_b \lesssim |\!|\!|q_h|\!|\!|_*. \tag{3.12}$$

Additionally, (3.10), (3.9c) and (3.9a) give us the inverse estimates

$$|\!|\!|\boldsymbol{v}_h|\!|\!|_* \lesssim h^{-1}\|\boldsymbol{v}_h\|_{\mathcal{O}_{\mathscr{T}}} \qquad \text{and} \qquad |\!|\!|(\boldsymbol{v}_h, q_h)|\!|\!|_* \lesssim h^{-1}\|(\boldsymbol{v}_h, q_h)\|_{\mathcal{O}_{\mathscr{T}}} \tag{3.13}$$

for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$, where $\|(\boldsymbol{v}_h, q_h)\|_{\mathcal{O}_{\mathscr{T}}}$ denotes the $\mathcal{L}^2(\mathcal{O}_{\mathscr{T}})$-norm on the product space, c.f. [Mas+14, Lemma 7.1].

Interpolation and Extension   Let $\mathcal{I}_h^*$ be the standard Scott-Zhang interpolation operator [SZ90] for the appropriate finite element space under consideration. From [SZ90; Ape99], we recall the following standard interpolation result.

Lemma 3.9.   *For $T \in \widetilde{\mathscr{T}_h}$ and $w \in \mathcal{H}^k(T)$, $k \geq 1$, it holds that*

$$\|D^k(w - \mathcal{I}_h^* w)\|_T \leq h^{k-m}\|D^m w\|_T, \qquad 0 \leq m \leq k.$$

Furthermore, on the boundary of an element it holds

$$\|w - \mathcal{I}_h^* w\|_{\partial T} \leq h^{k-1/2}\|D^k w\|_T.$$

To construct an interpolation $\mathcal{L}^2(\mathcal{O}_{\mathscr{T}}) \to \boldsymbol{V}_h$, i.e, on the entire active mesh, we recall that there is a linear extension operator $\mathcal{E}\colon \mathcal{H}^m(\Omega) \to \mathcal{H}^m(\mathcal{O}_{\mathscr{T}})$ for $m \geq 0$ for which it holds that

$$\|\mathcal{E}v\|_{\mathcal{H}^m(\mathcal{O}_{\mathscr{T}})} \lesssim \|v\|_{\mathcal{H}^m(\Omega)}. \tag{3.14}$$

See for example [Ste70, Section VI.3]. With this extension we define $\mathcal{I}_h\colon \mathcal{L}^2(\mathcal{O}_{\mathscr{T}}) \to \boldsymbol{V}_h$ as

$$\mathcal{I}_h v \coloneqq \mathcal{I}_h^* \mathcal{E} v.$$

With the stability of the extension (3.14), the same interpolation estimates hold for $\mathcal{I}_h$ as for $\mathcal{I}_h^*$. We then have the optimal order interpolation estimates

$$\|\nabla(\mathcal{E}v - \mathcal{I}_h v))\|_{\Omega_h^{\mathrm{ho}}} \lesssim h^k R(v), \quad h^{-1/2}\|\mathcal{E}v - \mathcal{I}_h v\|_{\Gamma_h^{\mathrm{ho}}} \lesssim h^k R(v), \quad h^{1/2}\|\partial_{\boldsymbol{n}}(\mathcal{E}v - \mathcal{I}_h v)\|_{\Gamma_h^{\mathrm{ho}}} \lesssim h^k R(v),$$

where $R(u) = \|v\|_{\mathcal{H}^{3,\infty}(\Omega)}$ if $k = 2$ and $R(u) = \|v\|_{\mathcal{H}^{k+1}(\Omega)}$ if $k \geq 3$; see [LR17; Leh17] for details. We note that these estimates are primarily due to the regularity of the parametric mapping from Lemma 3.6. An immediate consequence from the optimal order interpolation estimates is an interpolation estimate in our triple norms for $\boldsymbol{v} \in \mathcal{H}^{k+1}(\Omega)$ and $q \in \mathcal{H}^k(\Omega)$

$$\|\mathcal{E}\boldsymbol{v} - \mathcal{I}_h\boldsymbol{v}\|_* \lesssim h^k R(\boldsymbol{v}) \qquad \text{and} \qquad \|\mathcal{E}q - \mathcal{I}_h q\|_* \lesssim h^k \|q\|_{\mathcal{H}^k(\Omega)}. \tag{3.15}$$

As we have seen above, the extension operator is used to make sense of function on Ω_h^{ho}, which themselves are defined on Ω. Note that the mapping Φ_h, defined in Lemma 3.8, also defines such an extension. However, this is not sufficiently regular, as the extension is only in $\mathcal{H}^{1,\infty}(\Omega_h^{\mathrm{ho}})$. We therefore quantify, in which sense the mapped function $\boldsymbol{u} \circ \Phi_h$ are close to the extended function $\mathcal{E}\boldsymbol{u}$, based on the smoothness of the boundary $\Gamma \in C^{k+1}$.

Lemma 3.10. *For all $\boldsymbol{u} \in \mathcal{H}^3(\Omega)$ it holds*

$$\|\boldsymbol{u} \circ \Phi_h - \mathcal{E}\boldsymbol{u}\|_{\Omega_h^{ho}} \lesssim h^{k+1} \|\boldsymbol{u}\|_{\mathcal{H}^1(\Omega)} \tag{3.16}$$

$$\|\nabla\boldsymbol{u} \circ \Phi_h - \nabla\mathcal{E}\boldsymbol{u}\|_{\Omega_h^{ho}} \lesssim h^{k+1} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} \tag{3.17}$$

$$\|\boldsymbol{u} \circ \Phi_h - \mathcal{E}\boldsymbol{u}\|_{\Gamma_h^{ho}} \lesssim h^{k+1} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} \tag{3.18}$$

$$\|\partial_{\boldsymbol{n}}\boldsymbol{u} \circ \Phi_h - \mathcal{E}\partial_{\boldsymbol{n}}\boldsymbol{u}\|_{\Gamma_h^{ho}} \lesssim h^{k+1} \|\boldsymbol{u}\|_{\mathcal{H}^3(\Omega)}. \tag{3.19}$$

Proof. See [GOR15, Lemma 7.3] for (3.16), (3.17) and (3.18). The estimate (3.19) then immediately follows by replacing $\boldsymbol{u}$ with $\partial_{\boldsymbol{n}}\boldsymbol{u}$ in (3.18). $\qquad\square$

Remark 3.11. In [WRL21], we treated the approximation properties of the unfitted finite element space and the geometrical approximation separately. In this paper, we therefore proved an additional Lemma for the difference between the mapped and extended exact solution on the boundary, both for the normal derivative and the trace, where we traded a power of h in the convergence rate on the right-hand side, for lower regularity of the solution, plus an additional additive higher-order term. Since we require the regularity $\boldsymbol{u} \in \mathcal{H}^{k+1}(\Omega)$ and $p \in \mathcal{H}^k(\Omega)$ for the exact solution of the Stokes equations with $k \geq 2$ to obtain optimal order error estimates for the Taylor-Hood elements, see for example [Joh16, Corollary 4.30], the above Lemma is sufficient. $\qquad\blacktriangle$

3.2.3 Stability and Solvability

We begin with the numerical analysis of the method (3.8). The following results show that the bilinear form from the stabilised CutFEM discretisation fulfils an inf-sup on the product space of the velocity and pressure space, similar to the spatially smooth case, c.f. [Joh16, Lemma 4.11]. Most importantly, this property is shown in norms, which are independent of the mesh-interface cut position. As a result we then obtain the unique solvability of the problem and stability of the solution, independent of the mesh-interface cut position.

Lemma 3.12 (*Continuity*). *For the diffusion bilinear form, we have that for all $\boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{V}_h^{iso} + \boldsymbol{V}_h^{reg}$ that*

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}) \lesssim \|\|\boldsymbol{u}\|\|_a \|\|\boldsymbol{v}\|\|_a, \tag{3.20a}$$

and for all $\boldsymbol{u}_h, \boldsymbol{v}_h \in \boldsymbol{V}_h^{iso}$ that

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + \nu i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) \lesssim \|\|\boldsymbol{u}_h\|\|_* \|\|\boldsymbol{v}_h\|\|_*. \tag{3.20b}$$

For the pressure bilinear form, we have that for all $\boldsymbol{v} \in \boldsymbol{V}_h^{iso} + \boldsymbol{V}_h^{reg}$ and $q \in Q_h^{iso} + Q_h^{reg}$ that

$$b_h(q, \boldsymbol{v}) \lesssim \|\|q\|\|_b \|\|\boldsymbol{v}\|\|_a, \tag{3.21a}$$

and for all $\boldsymbol{v}_h \in \boldsymbol{V}_h^{iso}$ and $q_h \in Q_h^{iso}$ that

$$b_h(q_h, \boldsymbol{v}_h) \lesssim \|\|q_h\|\|_* \|\|\boldsymbol{v}_h\|\|_*. \tag{3.21b}$$

Proof. The estimates (3.20a) and (3.21a) are an immediate consequence of the Cauchy-Schwarz inequality, while (3.21b) then immediately follows by (3.12).

For (3.20b), we see that with repeated use of the Cauchy-Schwarz inequality, we have a Cauchy-Schwarz inequality for the direct ghost-penalties

$$
\begin{aligned}
i_h(\boldsymbol{u},\boldsymbol{v}) = \sum_{F\in\mathscr{F}_h}\frac{\gamma_{gp}}{h^2}\int_{\omega_F}[\![\boldsymbol{u}]\!]\cdot[\![\boldsymbol{v}]\!]\,\mathrm{d}\boldsymbol{x} &\le \sum_{F\in\mathscr{F}_h}\left[\frac{\gamma_{gp}}{h^2}\int_{\omega_F}[\![\boldsymbol{u}]\!]\cdot[\![\boldsymbol{u}]\!]\,\mathrm{d}\boldsymbol{x}\right]^{1/2}\left[\frac{\gamma_{gp}}{h^2}\int_{\omega_F}[\![\boldsymbol{v}]\!]\cdot[\![\boldsymbol{v}]\!]\,\mathrm{d}\boldsymbol{x}\right]^{1/2}\\
&\le \left[\sum_{F\in\mathscr{F}_h}\frac{\gamma_{gp}}{h^2}\int_{\omega_F}[\![\boldsymbol{u}]\!]\cdot[\![\boldsymbol{u}]\!]\,\mathrm{d}\boldsymbol{x}\right]^{1/2}\left[\sum_{F\in\mathscr{F}_h}\frac{\gamma_{gp}}{h^2}\int_{\omega_F}[\![\boldsymbol{v}]\!]\cdot[\![\boldsymbol{v}]\!]\,\mathrm{d}\boldsymbol{x}\right]^{1/2}\\
&= i_h(\boldsymbol{u},\boldsymbol{u})^{1/2}i_h(\boldsymbol{v},\boldsymbol{v})^{1/2}. \tag{3.22}
\end{aligned}
$$

With the trace estimate (3.10), we have

$$
a_h(\boldsymbol{u},\boldsymbol{v}) \lesssim \|\nabla\boldsymbol{u}\|_{\Omega_h^{\mathrm{ho}}}\|\nabla\boldsymbol{v}\|_{\Omega_h^{\mathrm{ho}}} + \|\nabla\boldsymbol{u}\|_{\Omega_h^{\mathrm{ho}}}\|h^{-1/2}\boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}} + \|h^{-1/2}\boldsymbol{u}\|_{\Gamma_h^{\mathrm{ho}}}\|\boldsymbol{v}\|_{\Omega_h^{\mathrm{ho}}}
$$
$$
+ \sigma\|h^{-1/2}\boldsymbol{u}\|_{\Gamma_h^{\mathrm{ho}}}\|h^{-1/2}\boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}}.
$$

Using this in combination with Young's inequality and (3.22), we then get

$$
\begin{aligned}
(a_h(\boldsymbol{u},\boldsymbol{v}) + i_h(\boldsymbol{u},\boldsymbol{v}))^2 &\lesssim \left(\|\nabla\boldsymbol{u}\|_{\Omega_h^{\mathrm{ho}}}^2 + i_h(\boldsymbol{u},\boldsymbol{u}) + \|h^{-1/2}\boldsymbol{u}\|_{\Gamma_h^{\mathrm{ho}}}\right)\left(\|\nabla\boldsymbol{v}\|_{\Omega}^2 + i_h(\boldsymbol{v},\boldsymbol{v}) + \|h^{-1/2}\boldsymbol{v}\|_{\Gamma_h^{\mathrm{ho}}}^2\right)\\
&\lesssim \|\!|\!|\boldsymbol{u}|\!|\!|_*^2\|\!|\!|\boldsymbol{v}|\!|\!|_*^2.
\end{aligned}
$$

The last inequality follows by an application of the ghost-penalty norm equivalence (3.5). $\qquad\square$

Corollary 3.13. *For all $(\boldsymbol{u}_h,p_h),(\boldsymbol{v}_h,q_h) \in \boldsymbol{V}_h^{iso}\times Q_h^{iso}$ it holds that*

$$
(A_h + J_h)((\boldsymbol{u}_h,p_h),(\boldsymbol{v}_h,q_h)) \lesssim |\!|\!|(\boldsymbol{u}_h,p_h)|\!|\!|_* |\!|\!|(\boldsymbol{v}_h,q_h)|\!|\!|_*.
$$

Proof. Let $\omega_F = T_1 \cup T_2$ be a facet patch. Due to shape-regularity of the mesh $\mathscr{T}_h$, we have that $\|p_i\|_{T_1}$ is equivalent to $\|p_i\|_{T_2}$. We therefore have that

$$
\begin{aligned}
j_h(p_h,p_h) = \gamma_{gp}\sum_{F\in\mathscr{F}_h}\int_{\omega_F}[\![p_h]\!]^2\,\mathrm{d}\boldsymbol{x} &\lesssim \sum_{F\in\mathscr{F}_h}\|p_{h,1}\|_{\omega_F}^2 + \|p_{h,2}\|_{\omega_F}^2\\
&\lesssim \sum_{F\in\mathscr{F}_h}\|p_{h,1}\|_{T_1}^2 + \|p_{h,1}\|_{T_2}^2 + \|p_{h,2}\|_{T_1}^2 + \|p_{h,2}\|_{T_2}^2 \lesssim \sum_{T\in\mathscr{T}_h^{\Gamma,+}}\|p_h\|_T^2 \lesssim \|p_h\|_{\mathcal{O}_{\mathscr{T}}}^2.
\end{aligned}
$$

Using the Cauchy-Schwarz inequality for the ghost-penalty operator (3.22) for the pressure stabilisation, the claim follows by Lemma 3.12. $\qquad\square$

Lemma 3.14 (*Coercivity of the diffusion operator*). *For σ sufficiently large and h sufficiently small, it holds for all $\boldsymbol{u}_h \in \boldsymbol{V}_h^{iso}$ that*

$$
a_h(\boldsymbol{u}_h,\boldsymbol{u}_h) + i_h(\boldsymbol{u}_h,\boldsymbol{u}_h) \gtrsim |\!|\!|\boldsymbol{u}_h|\!|\!|_*^2.
$$

Proof. For h sufficiently small, shape-regularity of the transformed mesh follows by the mesh regularity of the original mesh. As a result, Lemma 3.1 and therefore Lemma 3.3 also hold on the transformed mesh. The claim then follows by [Leh17, Lemma 8]. $\qquad\square$

To formulate the stability result for the pressure coupling bilinear form, we need to introduce the *interior domain*, defined as $\mathcal{O}_{\mathrm{int}} := \mathcal{O}_{\mathscr{T}} \setminus \mathcal{O}_{\mathscr{T}_\Gamma}$. This is the domain consisting of active and uncut elements.

Lemma 3.15 (*Disturbed inf-sup result*). *If h is sufficiently small, then there exists a constant $\beta > 0$ such that*

$$\beta \|q_h\|_{\Omega_h^{ho}} \leq \sup_{\boldsymbol{v}_h \in \boldsymbol{V}_h^{iso}} \frac{b_h(q_h, \boldsymbol{v}_h)}{\|\|\boldsymbol{v}_h\|\|_*} + j_h(q_h, q_h)^{1/2}$$

for all $q_h \in Q_h^{iso}$ with $q_h|_{\mathcal{O}_{int}} \in \mathcal{L}_0^2(\mathcal{O}_{int})$. The constant β is independent of p and h.

Proof. See. [GO17] for the proof on undeformed meshes. The statement then follows from the regularity of the mesh deformation Θ_h. $\qquad\square$

Lemma 3.16 (*Inf-sup result*). *For $\sigma > 0$ sufficiently large $h > 0$ and sufficiently small, there exists a constant $c_{L3.16} > 0$, independent of h and the mesh-interface cut position, such that for all $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}$ it holds*

$$c_{L3.16} \|\|(\boldsymbol{u}_h, p_h)\|\|_* \leq \sup_{(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}} \frac{(A_h + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h))}{\|\|(\boldsymbol{v}_h, q_h)\|\|_*}. \tag{3.23}$$

Proof. The proof follows standard arguments, c.f. [GO17, section 5]. We give a brief overview due to the significance of the result. The proof is done for $\nu = 1$. The scaling with respect to arbitrary $\nu > 0$ then follows by a rescaling argument. For a given $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}$, we can find a $\boldsymbol{w}_h \in \boldsymbol{V}_h^{iso}$ such that $\|\|\boldsymbol{w}_h\|\|_* = \|p_h\|_{\Omega_h^{ho}}$. Then due to Lemma 3.15, we have with the continuity of the diffusion bilinear form and the weighted Young's inequality that

$$\beta \|p_h\|_{\Omega_h^{ho}}^2 \leq b_h(p_h, \boldsymbol{w}_h) + j_h(p_h, p_h) \|p_h\|_{\Omega_h^{ho}} = A_h((\boldsymbol{u}_h, p_h), (\boldsymbol{w}_h, 0)) - a_h(\boldsymbol{u}_h, \boldsymbol{w}_h) + j_h(p_h, p_h) \|p_h\|_{\Omega_h^{ho}}$$

$$\leq (A_h + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{w}_h, 0)) + \frac{c_{(3.20b)}^2}{4\beta} \|\|\boldsymbol{u}_h\|\|_*^2 + \frac{\beta}{4} \|p_h\|_{\Omega_h^{ho}}^2 + \frac{1}{\beta} j_h(p_h, p_h) + \frac{\beta}{4} \|p_h\|_{\Omega_h^{ho}}^2.$$

With the coercivity of the diffusion bilinear form, we also have

$$(A_h + J_h)((\boldsymbol{u}_h, p_h, \boldsymbol{u}_h, -p_h)) \geq c_{L3.14} \|\|\boldsymbol{u}_h\|\|_*^2 + j_h(p_h, p_h).$$

With the appropriate choice of $\alpha > 0$, we get that

$$(A_h + J_h)((\boldsymbol{u}_h, p_h, \boldsymbol{u}_h + \alpha \boldsymbol{w}_h, -p_h)) \gtrsim \|\|(\boldsymbol{u}_h, p_h)\|\|_*^2,$$

where the constant only depends on β, $c_{(3.20b)}$ and $c_{L3.14}$, each of which are independent of h and the mesh-interface cut position. The claim follows by the observation

$$\|\|(\boldsymbol{u}_h - \alpha \boldsymbol{w}_h, -p_h)\|\|_*^2 = \|\|\boldsymbol{u}_h - \alpha \boldsymbol{w}_h\|\|_*^2 + \|\|p_h\|\|_*^2 \leq \|\|\boldsymbol{u}_h\|\|_*^2 + \alpha^2 \|\|\boldsymbol{w}_h\|\|_*^2 + \|\|p_h\|\|_*^2$$

$$\leq \|\|\boldsymbol{u}_h\|\|_*^2 + (1 + \alpha^2) \|\|p_h\|\|_*^2 \lesssim \|\|(\boldsymbol{u}_h, p_h)\|\|_*^2.$$

$\qquad\square$

Corollary 3.17 (*Unique solvability*). *The unfitted, isoparametric finite element discretisation* (3.8) *of the Stokes problem admits a unique solution $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}$ bounded by the data*

$$\|\|(\boldsymbol{u}_h, p_h)\|\|_* \leq c_{L3.16}^{-1} \|\|\boldsymbol{f}_h\|\|_{*,-1}.$$

Proof. This is an immediate consequence of the Banach-Nečas-Babuška theorem, see for example [EG21b, Theorem 25.9], applicable due to the continuity and stability of the stabilised bilinear form $A_h + J_h$ in the product space. $\qquad\square$

3.2.4 Condition Number

We now investigate the condition number of the stiffness matrix, resulting from the isoparametric unfitted CutFEM discretisation. In fact, due to the regularity of the parametric mapping for sufficiently small h and the ghost-penalty stabilisation providing stability in norms independent of the mesh-interface cut position, we shall recover the standard condition number estimate for second-order problems, c.f. [EG04, Theorem 9.11].

Definition 3.18. Let $\mathscr{A} \in \mathbb{R}^{n \times n}$ be an invertible matrix. The *condition number* of $\mathscr{A}$ is defined as

$$\kappa(\mathscr{A}) := \|\mathscr{A}\| \|\mathscr{A}^{-1}\|$$

with the matrix norm $\|\mathscr{A}\| := \sup_{\boldsymbol{x} \in \mathbb{R}^n \setminus \{0\}} \|\mathscr{A}\boldsymbol{x}\|_2 / \|\boldsymbol{x}\|_2$, where $(\cdot, \cdot)_2$ denoted the Euclidean inner product and $\|\cdot\|_2$ the corresponding induced norm.

As noted in subsection 3.1.2, ghost-penalty stabilisation enables the proof of bounds on the condition number of the system matrix, independent of the mesh interface cut position. Conversely, it is known for the system matrix $\mathscr{A}$ resulting from a CutFEM discretisation without ghost-penalty stabilisation, that if there are cut elements with only a very small region inside the domain of interest, i.e., there exists an element $T \in \mathscr{T}_{h,\Gamma}$ with $\mathrm{meas}_d(T \cap \Omega_h) < \varepsilon$ and $\mathrm{meas}_{d-1}(T \cap \Gamma_h) < \varepsilon$, then the condition number is bounded from below by $\kappa(\mathscr{A}) \gtrsim \varepsilon^{-1/2}$, c.f. [Bur10].

Following [Mas+14], we now show that the condition number of the system matrix resulting from the stabilised method can be bound by ch^{-2}, independently of the mesh-interface cut position. To prove this bound, we need to introduce some further notation. Let $\{\psi_i\}_i^n$ be a basis of the finite element space $\boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$. Then $\mathcal{B} \to \mathbb{R}^n$, $(\boldsymbol{u}_h, q_h) \mapsto U$ where $(\boldsymbol{u}_h, q_h) = \sum_{i=1}^n U_i \psi_i$ defines an isomorphism. We then have from [EG04, Lemma 9.7] that for a quasi-uniform mesh $\widetilde{\mathscr{T}_h}$ it holds

$$\|U\|_2^2 \simeq h^d \|(\boldsymbol{u}_h, p_h)\|_{\mathcal{O}_{\mathscr{T}}}^2. \tag{3.24}$$

Now, we define the stiffness matrix $\mathscr{A}$ of (3.8) such that

$$(\mathscr{A}U, V)_2 = (A_h + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h))$$

for all $(\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$, where we denote $U = \mathcal{B}((\boldsymbol{u}_h, p_h))$ and $V = \mathcal{B}((\boldsymbol{v}_h, q_h))$. Now, since we are considering the Stokes problem with Dirichlet boundary conditions for the velocity on the entire boundary $\partial \Omega_h^{\mathrm{ho}} = \Gamma_h^{\mathrm{ho}}$, the pressure is only unique up to an additive constant. The stiffness matrix matrix $\mathscr{A}$ is therefore singular with kernel $\ker(\mathscr{A}) = \mathrm{span}\{\mathcal{B}((\boldsymbol{0}, 1))\}$. For the rest of this section we will therefore work with $\mathscr{A}$ as a bijective mapping between $\widehat{\mathbb{R}}^n := \mathbb{R}^n / \ker(\mathscr{A})$ and $\widetilde{\mathbb{R}}^n := \mathrm{im}(\mathscr{A})$.

Theorem 3.19 (*Condition number estimate*). *For the stiffness matrix of the isoparametric CutFEM discretisation* (3.8) *of the Stokes problem, it holds that*

$$\kappa(\mathscr{A}) \lesssim h^{-2}.$$

Before we proceed with the proof, we note that the matrix norm can be equivalently defined as

$$\|\mathscr{B}\| = \sup_{U,V \in \mathbb{R}^n \setminus \{0\}} \frac{(\mathscr{B}U, V)_2}{\|U\|_2 \|V\|_2}.$$

Proof. With Lemma 3.6 and Lemma 3.7, we have that norms on the deformed mesh are equivalent to the norm on the untransformed mesh. Using the continuity of the bilinear form in Corollary 3.13, the inverse estimate (3.13) and the norm equivalence (3.24), we see that for all $U, V \in \widehat{\mathbb{R}}^n \setminus \{0\}$ it holds

$$(\mathscr{A}U, V)_2 = (A_h + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h))$$
$$\lesssim \|\|(\boldsymbol{u}_h, p_h)\|\|_* \|\|(\boldsymbol{v}_h, q_h)\|\|_* \lesssim h^{-2} \|(\boldsymbol{u}_h, p_h)\|_{\mathcal{O}_{\mathscr{T}}} \|(\boldsymbol{v}_h, q_h)\|_{\mathcal{O}_{\mathscr{T}}} \lesssim h^{d-2} \|U\|_2 \|V\|_2$$

such that $\|\mathscr{A}\|_2 \lesssim h^{d-2}$. Similarly, we have with the inf-sup stability of the system in Lemma 3.16, the Poincaré inequality (3.11) and the norm equivalence (3.24), that for all $U \in \widehat{\mathbb{R}}^n \setminus \{0\}$ it holds

$$
\begin{aligned}
(\mathscr{A}U, V)_2 &= (A_h + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) \\
&\gtrsim \|\|(\boldsymbol{u}_h, p_h)\|\|_* \|\|(\boldsymbol{v}_h, q_h)\|\|_* \gtrsim \|(\boldsymbol{u}_h, p_h)\|_{\mathcal{O}_{\mathscr{T}}} \|(\boldsymbol{v}_h, q_h)\|_{\mathcal{O}_{\mathscr{T}}} \gtrsim h^d \|U\|_2 \|V\|_2.
\end{aligned}
$$

With this we have that

$$
\|\mathscr{A}U\| = \sup_{W \in \widetilde{\mathbb{R}}^n \setminus \{0\}} \frac{(\mathscr{A}U, W)_2}{\|W\|_2} \geq \frac{(\mathscr{A}U, V)_2}{\|V\|_2} \gtrsim h^d \|U\|_2.
$$

With $U = \mathscr{A}^{-1}W$, which is permissible by the regularity if $\mathscr{A}$ on the reduced space, the above inequality implies that

$$
\|\mathscr{A}^{-1}W\|_2 \lesssim h^{-d}\|W\|_2 \qquad \Rightarrow \qquad \|\mathscr{A}^{-1}\|_2 \lesssim h^{-d}.
$$

As a result we have that

$$
\kappa(\mathscr{A}) = \|\mathscr{A}\|_2 \|\mathscr{A}^{-1}\|_2 \lesssim h^{-2}.
$$

$\square$

3.2.5 A Priori Error Analysis

We shall now show, that the isoparametric CutFEM discretisation of the Stokes problem admits optimal order error estimates in the natural norm, under the assumption that the exact level set geometry is sufficiently smooth and that the mesh resolves this geometry sufficiently well. This is to be expected, as the parametric mapping recovers the correct order in the geometry approximation so that this does not dominate the approximation properties of the finite element space. However, since $\boldsymbol{V}_h^{\mathrm{iso}} \not\subset \boldsymbol{\mathcal{H}}_0^1(\Omega)$ and $Q_h^{\mathrm{iso}} \not\subset \mathcal{L}_0^2(\Omega)$, we do not have standard Galerkin orthogonality. As a result, the most effort is needed to show the approximate Galerkin orthogonality/consistency error of the correct order in h.

For the error analysis, we introduce the bilinear and linear forms for the exact geometry. In order to then test with discrete functions, we make use of the mapping Φ_h. For $\boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^2(\Omega) + \{\boldsymbol{v}_h \circ \Phi_h^{-1} \mid \boldsymbol{v}_h \in \boldsymbol{V}_h^{\mathrm{iso}}\}$ and $p, q \in \mathcal{H}^1(\Omega) + \{q_h \circ \Phi_h^{-1} \mid q_h \in Q_h^{\mathrm{iso}}\}$, we define

$$
a^1(\boldsymbol{u}, \boldsymbol{v}) := a(\boldsymbol{u}, \boldsymbol{v}) - \int_\Gamma \partial_{\boldsymbol{n}} \boldsymbol{u} \cdot \boldsymbol{v} \, \mathrm{d}s, \qquad\qquad b^1(q, \boldsymbol{v}) := b(q, \boldsymbol{v}) + \int_\Gamma (q\boldsymbol{n}) \cdot \boldsymbol{v} \, \mathrm{d}s,
$$

$$
A((\boldsymbol{u}, p), (\boldsymbol{v}, q)) := a^1(\boldsymbol{u}, \boldsymbol{v}) + b^1(p, \boldsymbol{v}) + b^1(q, \boldsymbol{u}) \qquad \text{and} \qquad f(\boldsymbol{v}) := \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}.
$$

Lemma 3.20 (*Weak Galerkin orthogonality*). *Let* $(\boldsymbol{u}, p) \in \boldsymbol{\mathcal{H}}^2(\Omega) \times \mathcal{H}^1(\Omega)$ *be the solution to the Stokes problem* (2.17)*. Then for all* $\boldsymbol{v}_h, q_h \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}$ *it holds that*

$$
A((\boldsymbol{u}, p), (\boldsymbol{v}_h^\ell, q_h^\ell)) = f(\boldsymbol{v}_h^\ell) \tag{3.25}
$$

with $(\boldsymbol{v}_h^\ell, q_h^\ell) = (\boldsymbol{v}_h \circ \Phi_h^{-1}, q_h \circ \Phi_h^{-1})$.

Proof. This follows by applying integration by parts to the volume terms in $A(\cdot, \cdot)$. $\square$

In the literature on the CutFEM method for the Stokes problem [Mas+14; BH14; GO17], the authors assumed exact geometry handling such that $\Omega_h = \Omega$. As a result, the authors have the standard Galerkin orthogonality $A_h((\boldsymbol{u}, p), (\boldsymbol{v}_h, q_h)) = f_h(\boldsymbol{v}_h)$.

For the error analysis, we introduce some further notation. Let $(\boldsymbol{u}, p)$ be the solution of the Stokes problem (2.17) and $(\boldsymbol{u}_h, p_h)$ the solution of the isoparametric CutFEM discretisation of the Stokes problem (3.8).

In order for the error to be well-defined on the entire active domain $\mathcal{O}_{\mathscr{T}}$, we use the extension operator $\mathcal{E}$ to define the velocity and pressure errors as

$$\mathbb{E} := \boldsymbol{u}^e - \boldsymbol{u}_h \qquad \text{and} \qquad \mathbb{D} := p^e - p_h,$$

where $\boldsymbol{u}^e := \mathcal{E}\boldsymbol{u}$ and $p^e := \mathcal{E}p$, to ease the notation. We then split this into an interpolation and discretisation error

$$\mathbb{E} = (\boldsymbol{u}^e - \mathcal{I}_h\boldsymbol{u}) + (\mathcal{I}_h\boldsymbol{u} - \boldsymbol{u}_h) =: \boldsymbol{\eta} + \mathbf{e}_h \tag{3.26a}$$

$$\mathbb{D} = (p^e - \mathcal{I}_h p) + (\mathcal{I}_h p - p_h) =: \zeta + \mathrm{d}_h, \tag{3.26b}$$

see subsection 3.2.2 for details on the interpolation.

Now, subtracting (3.25) from (3.8), as well as adding and subtracting appropriate terms, we get the error equation

$$\begin{aligned}
(A_h + J_h)((\mathbb{E}, \mathbb{D}), (\boldsymbol{v}_h, q_h)) &= f(\boldsymbol{v}_h^\ell) - f_h(\boldsymbol{v}_h) + a_h(\boldsymbol{u}, \boldsymbol{v}_h) - a^1(\boldsymbol{u}, \boldsymbol{v}_h^\ell) + b_h(p, \boldsymbol{v}_h) - b^1(p, \boldsymbol{v}_h^\ell) \\
&\qquad + b_h(q_h, \boldsymbol{u}) - b^1(q_h^\ell, \boldsymbol{u}) + i_h(\boldsymbol{u}^e, \boldsymbol{v}_h) - j_h(p^e, q_h) \\
&=: \mathfrak{T}_1 + \mathfrak{T}_2 + \mathfrak{T}_3 + \mathfrak{T}_4 + \mathfrak{T}_5 \\
&= \mathfrak{E}_c(\boldsymbol{v}_h, q_h).
\end{aligned} \tag{3.27}$$

The five terms on the right-hand side of this are the forcing, diffusion, pressure, divergence constraint and ghost-penalty contributions to the consistency error. For the consistency error, we then have the following result.

Lemma 3.21 (*Consistency error*). *For the consistency error of the isoparametric CutFEM discretisation of the Stokes equation, it holds that*

$$|\mathfrak{E}_c(\boldsymbol{v}_h, q_h)| \lesssim h^k R_c(\boldsymbol{u}, p, \boldsymbol{f}) \|\|(\boldsymbol{v}_h, q_h)\|\|_*$$

with $R_c(\boldsymbol{u}, p, \boldsymbol{f}) := \|\boldsymbol{f}\|_{\boldsymbol{\mathcal{H}}^1(\Omega)} + \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega)} + \|p\|_{\mathcal{H}^k(\Omega)}.$

Proof. We deal with the five consistency error contributions separately. The proof follows the same lines as [WRL21, Lemma 5.14].

Consistency term 1: $\mathfrak{T}_1 = f(\boldsymbol{v}_h^\ell) - f_h(\boldsymbol{v}_h)$. Using a chain rule to change the integral over Ω to Ω_h^{ho}, Lemma 3.8, the stability of the extension and denoting $J = \det(D\Phi_h)$ yields

$$|\mathfrak{T}_1| = |f(\boldsymbol{v}_h^\ell) - f_h(\boldsymbol{v}_h)| = |(\boldsymbol{f} \circ \Phi_h, (J-1)\boldsymbol{v}_h)_{\Omega_h^{\mathrm{ho}}} + (\boldsymbol{f} \circ \Phi_h - \boldsymbol{f}^e, \boldsymbol{v}_h)_{\Omega_h^{\mathrm{ho}}}| \lesssim h^k \|\boldsymbol{f}\|_{\boldsymbol{\mathcal{H}}^1(\Omega)} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

See also [LR17, Lemma 5.10] and [GOR14, Lemma 7.5].

Consistency term 2: $\mathfrak{T}_2 = a_h(\boldsymbol{u}, \boldsymbol{v}_h) - a^1(\boldsymbol{u}, \boldsymbol{v}_h^\ell)$. First we observe that it follows from the chain rule that

$$\nabla \boldsymbol{v}_h^\ell(\hat{\boldsymbol{x}}) = \nabla(\boldsymbol{v}_h \circ \Phi^{-1})(\hat{\boldsymbol{x}}) = D\Phi(\boldsymbol{x})^{-T} \nabla \boldsymbol{v}_h(\boldsymbol{x}) \qquad \text{for } \hat{\boldsymbol{x}} \in \Omega, \ \boldsymbol{x} := \Phi^{-1}(\hat{\boldsymbol{x}}). \tag{3.28}$$

Now we split the diffusion consistency error term into a the volume contribution, the consistency term, symmetry term and penalty term. For the volume term we have

$$\begin{aligned}
\mathfrak{T}_2^1 &:= \nu \int_{\Omega_h^{\mathrm{ho}}} \nabla \boldsymbol{u}^e : \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \nu \int_{\Omega} \nabla \boldsymbol{u} : \nabla \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} \\
&= \nu \int_{\Omega_h^{\mathrm{ho}}} \nabla \boldsymbol{u}^e : \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \nu \int_{\Omega_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}) \circ \Phi_h : D\Phi_h^{-T} \nabla \boldsymbol{v}_h J \, \mathrm{d}\boldsymbol{x} \\
&= \nu \int_{\Omega_h^{\mathrm{ho}}} \nabla \boldsymbol{u}^e - (\nabla \boldsymbol{u}) \circ \Phi_h : \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \nu \int_{\Omega_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}) \circ \Phi_h : (I - JD\Phi_h^{-T}) \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \\
&\lesssim \|\nabla \boldsymbol{u}^e - (\nabla \boldsymbol{u}) \circ \Phi_h\|_{\Omega_h^{\mathrm{ho}}} \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}} + \|I - JD\Phi_h^{-T}\|_{\mathcal{L}^\infty(\Omega_h^{\mathrm{ho}})} \|(\nabla \boldsymbol{u}) \circ \Phi_h\|_{\Omega_h^{\mathrm{ho}}} \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.
\end{aligned}$$

Now, with the triangle inequality and Lemma 3.8 we find that

$$\|I - JD\Phi_h^{-T}\|_{\mathcal{L}^\infty(\Omega_h^{\mathrm{ho}})} \le \|(I - IJ)D\Phi_h^{-T}\|_{\mathcal{L}^\infty(\Omega_h^{\mathrm{ho}})} + \|I - D\Phi_h^{-T}\|_{\mathcal{L}^\infty(\Omega_h^{\mathrm{ho}})} \lesssim h^k. \tag{3.29}$$

So together with (3.17) it holds

$$|\mathfrak{T}_2^1| \lesssim h^k \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

The boundary terms are a little more involved, since we do not only have to take the different domains into account, but also the different normal vectors on these different boundaries. Let $\hat{\boldsymbol{n}}$ denote the unit normal vector on Γ and $\boldsymbol{n}$ the normal on Γ_h^{ho}. We then have from [LR17] that

$$\int_\Gamma (\nabla \boldsymbol{u})\hat{\boldsymbol{n}} \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{s} = \int_{\Gamma_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}) \circ \Phi_h \, D\Phi_h^{-T} \boldsymbol{n} \cdot \boldsymbol{v}_h J \,\mathrm{d}s.$$

Then, together with (3.29) and (3.19) it holds for the consistency term that

$$
\begin{aligned}
\mathfrak{T}_2^2 &= \int_{\Gamma_h^{\mathrm{ho}}} -\partial_{\boldsymbol{n}} \boldsymbol{u}^e \cdot \boldsymbol{v}_h \,\mathrm{d}s - \int_\Gamma -\partial_{\hat{\boldsymbol{n}}} \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{s} \\
&= \int_{\Gamma_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}) \circ \Phi_h \, D\Phi_h^{-T} \boldsymbol{n} \cdot \boldsymbol{v}_h J \,\mathrm{d}s - \int_{\Gamma_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}^e) \boldsymbol{n} \cdot \boldsymbol{v}_h \,\mathrm{d}s \\
&= \int_{\Gamma_h^{\mathrm{ho}}} (\nabla \boldsymbol{u}) \circ \Phi_h \cdot (JD\Phi_h^{-T} - I)\boldsymbol{n} \cdot \boldsymbol{v}_h \,\mathrm{d}s + \int_{\Gamma_h^{\mathrm{ho}}} ((\nabla \boldsymbol{u}) \circ \Phi_h - \nabla \boldsymbol{u}^e)\boldsymbol{n} \cdot \boldsymbol{v}_h \,\mathrm{d}s \\
&\lesssim \|JD\Phi_h^{-T} - I\|_{\mathcal{L}^\infty(\Omega_h^{\mathrm{ho}})} \|h^{1/2}(\nabla \boldsymbol{u}) \circ \Phi_h \boldsymbol{n}\|_{\Gamma_h^{\mathrm{ho}}} \|h^{-1/2}\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} + h^{k+1}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^3(\Omega)} \|\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} \\
&\lesssim h^{k+1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^3(\Omega)} \|h^{-1/2}\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}}.
\end{aligned}
$$

For the symmetry term, we use the fact that $\boldsymbol{u} \circ \Phi_h = 0$ on Γ_h^{ho}, so that with (3.18) we have

$$
\begin{aligned}
\mathfrak{T}_2^3 &= \int_{\Gamma_h^{\mathrm{ho}}} \partial_{\boldsymbol{n}} \boldsymbol{v}_h \cdot (\underbrace{\boldsymbol{u} \circ \Phi_h}_{=0} - \boldsymbol{u}^e) \,\mathrm{d}s \lesssim \|\partial_{\boldsymbol{n}} \boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} \|\boldsymbol{u} \circ \Phi_h - \boldsymbol{u}^e\|_{\Gamma_h^{\mathrm{ho}}} \\
&\qquad\qquad \lesssim h^{k+1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|h^{1/2}\partial_{\boldsymbol{n}} \boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} \lesssim h^{k+1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|\nabla \boldsymbol{v}_h\|_{\mathcal{O}_{\mathscr{T}}},
\end{aligned}
$$

where the last inequality follows from the trace inequality (3.9c). Very similarly, we have for the penalty term that

$$\mathfrak{T}_2^4 = \frac{1}{h}\int_{\Gamma_h^{\mathrm{ho}}} \boldsymbol{u}^e - \underbrace{\boldsymbol{u} \circ \Phi_h}_{=0} \,\mathrm{d}s \le h^{1/2}\|\boldsymbol{u}^2 - \boldsymbol{u} \circ \Phi_h\|_{\Gamma_h^{\mathrm{ho}}} \|h^{-1/2}\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} \lesssim h^{k+1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|h^{-1/2}\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}}.$$

In summary we therefore have

$$|\mathfrak{T}_2| \le |\mathfrak{T}_2^1| + |\mathfrak{T}_2^2| + |\mathfrak{T}_2^3| + |\mathfrak{T}_2^4| \lesssim h^k \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^3(\Omega)} \vert\!\vert\!\vert \boldsymbol{v}_h \vert\!\vert\!\vert_*.$$

<u>Consistency term 3:</u> $\mathfrak{T}_3 = b_h(p, \boldsymbol{v}_h) - b^1(p, \boldsymbol{v}_h^\ell)$. As in (3.28) we have with the chain rule that

$$\nabla \cdot \boldsymbol{v}_h^\ell(\hat{\boldsymbol{x}}) = \mathrm{tr}(\nabla(\boldsymbol{v}_h \circ \Phi^{-1})(\hat{\boldsymbol{x}})) = \mathrm{tr}(D\Phi(\boldsymbol{x})^{-T}\nabla \boldsymbol{v}_h(\boldsymbol{x})) \qquad \text{for } \hat{\boldsymbol{x}} \in \Omega, \ \boldsymbol{x} := \Phi^{-1}(\hat{\boldsymbol{x}}). \tag{3.30}$$

We again treat the volume and boundary terms separately. It follows using (3.30), (3.29) and (3.16) that

$$
\begin{aligned}
\mathfrak{T}_3^1 &= \int_\Omega p\nabla \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{\boldsymbol{x}} - \int_{\Omega_h^{\mathrm{ho}}} p^e\nabla \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} = \int_{\Omega_h^{\mathrm{ho}}} p \circ \Phi_h \, \mathrm{tr}(D\Phi(\boldsymbol{x})^{-T}\nabla \boldsymbol{v}_h)J - p^e\nabla \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_h^{\mathrm{ho}}} p \circ \Phi_h \, \mathrm{tr}((JD\Phi(\boldsymbol{x})^{-T} - I)\nabla \boldsymbol{v}_h) \,\mathrm{d}\boldsymbol{x} + \int_{\Omega_h^{\mathrm{ho}}} (p \circ \Phi_h - p^e) \,\mathrm{tr}(\nabla \boldsymbol{v}_h) \\
&\lesssim h^k \|p\|_\Omega \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}} + h^{k+1}\|p\|_{\mathcal{H}^1(\Omega)} \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}} \lesssim h^k \|p\|_{\mathcal{H}^1(\Omega)} \|\nabla \boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.
\end{aligned}
$$

For the boundary term, we use the same change of normal direction and measure as for the diffusion consistency boundary term, so that

$$
\begin{aligned}
\mathfrak{T}_3^1 &= \int_{\Gamma_h^{\mathrm{ho}}} p^e \boldsymbol{n} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma} p \hat{\boldsymbol{n}} \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} = \int_{\Gamma_h^{\mathrm{ho}}} p^e \boldsymbol{n} \cdot \boldsymbol{v}_h - p \circ \Phi_h D \Phi_h^{-T} \boldsymbol{n} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \\
&= \int_{\Gamma_h^{\mathrm{ho}}} (p^e - p \circ \Phi_h) \boldsymbol{n} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma_h^{\mathrm{ho}}} p \circ \Phi_h (I - D\Phi_h^{-T}) \boldsymbol{n} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \\
&\lesssim h^{k+3/2} \|p\|_{\mathcal{H}^2(\Omega)} \|h^{-1/2} \boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}} + h^k \|p \circ \Phi_h\|_{\Gamma_h^{\mathrm{ho}}} \|\boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}}.
\end{aligned}
$$

For the pressure norm on the approximated interface we use a trace inequality so that we can estimate $\|p \circ \Phi_h\|_{\Gamma_h^{\mathrm{ho}}} \simeq \|p\|_\Gamma \lesssim \|p\|_{\mathcal{H}^1(\Omega)}$. In total we therefore have for the boundary term that

$$
|\mathfrak{T}_3^2| \le h^{k+1/2} \|p\|_{\mathcal{H}^2(\Omega)} \|h^{-1/2} \boldsymbol{v}_h\|_{\Gamma_h^{\mathrm{ho}}}.
$$

Overall this give for the pressure consistency error

$$
|\mathfrak{T}_3| \le |\mathfrak{T}_3^1| + |\mathfrak{T}_3^2| \lesssim h^k \|p\|_{\mathcal{H}^2(\Omega)} \|\|\boldsymbol{v}_h\|\|_*.
$$

Consistency term 4: $\mathfrak{T}_4 = b_h(q_h, \boldsymbol{u}) - b^1(q_h^\ell, \boldsymbol{u})$. As before we have for the volume term

$$
\begin{aligned}
\mathfrak{T}_4^1 &= \int_{\Omega_h^{\mathrm{ho}}} q_h^\ell \nabla \cdot \boldsymbol{u}^e \, \mathrm{d}\hat{\boldsymbol{x}} - \int_{\Omega_h^{\mathrm{ho}}} q_h \nabla \cdot \boldsymbol{u}^e \, \mathrm{d}\boldsymbol{x} = \int_{\Omega_h^{\mathrm{ho}}} q_h (\nabla \cdot \boldsymbol{u}) \circ \Phi_h J \, \mathrm{d}\boldsymbol{x} - \int_{\Omega_h^{\mathrm{ho}}} q_h \nabla \cdot \boldsymbol{u}^e \, \mathrm{d}\boldsymbol{x} \\
&= \int_{\Omega_h^{\mathrm{ho}}} q_h ((\nabla \cdot \boldsymbol{u}) \circ \Phi_h - \nabla \cdot \boldsymbol{u}^e) \, \mathrm{d}\boldsymbol{x} + \int_{\Omega_h^{\mathrm{ho}}} q_h (\nabla \cdot \boldsymbol{u}) \circ \Phi_h (J - 1) \, \mathrm{d}\boldsymbol{x} \\
&\lesssim h^{k+1} \|q_h\|_{\Omega_h^{\mathrm{ho}}} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} + h^k \|q_h\|_{\Omega_h^{\mathrm{ho}}} \|\nabla \boldsymbol{u}\|_\Omega \lesssim h^k \|q_h\|_{\Omega_h^{\mathrm{ho}}} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)}.
\end{aligned}
$$

For the boundary contribution we proceed as before

$$
\begin{aligned}
\mathfrak{T}_4^2 &= \int_{\Gamma_h^{\mathrm{ho}}} q_h \boldsymbol{n} \cdot \boldsymbol{u}^e \, \mathrm{d}s - \int_{\Gamma} q_h^\ell \hat{\boldsymbol{n}} \cdot \boldsymbol{u} \, \mathrm{d}\hat{s} = \int_{\Gamma_h^{\mathrm{ho}}} q_h \boldsymbol{n} \cdot \boldsymbol{u}^e \, \mathrm{d}s - \int_{\Gamma_h^{\mathrm{ho}}} q_h D\Phi_h^{-T} \boldsymbol{n} \cdot \boldsymbol{u} \circ \Phi_h J \, \mathrm{d}s \\
&= \int_{\Gamma_h^{\mathrm{ho}}} q_h \boldsymbol{n} \cdot (\boldsymbol{u}^e - \boldsymbol{u} \circ \Phi_h) \, \mathrm{d}s + \int_{\Gamma_h^{\mathrm{ho}}} q_h (I - J D\Phi_h^{-T}) \boldsymbol{n} \cdot \boldsymbol{u} \circ \Phi_h \, \mathrm{d}s \\
&\lesssim h^{k+1} \|q_h\|_{\Gamma_h^{\mathrm{ho}}} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} + h^k \|q_h\|_{\Gamma_h^{\mathrm{ho}}} \|\boldsymbol{u}\|_\Gamma \lesssim h^{k+1/2} \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} \|q_h\|_{\mathcal{O}_{\mathcal{T}}}.
\end{aligned}
$$

In the final step we have used the trace estimate (3.10) and the inverse estimate (3.9a) to bound $\|q_h\|_{\Gamma_h^{\mathrm{ho}}} \lesssim h^{-1/2} \|q_h\|_{\mathcal{O}_{\mathcal{T}}}$ and the fact that $\boldsymbol{u}$ vanishes on Γ. The divergence constraint consistency term is therefore bound by

$$
|\mathfrak{T}_4| \le |\mathfrak{T}_4^1| + |\mathfrak{T}_4^2| \lesssim h^k \|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega)} \|\|q_h\|\|_*.
$$

Consistency term 5: $\mathfrak{T}_5 = i_h(\boldsymbol{u}^e, \boldsymbol{v}_h^\ell) - j_h(p^e, q_h^\ell)$. We use the Cauchy-Schwarz inequality for the ghost-penalty operator (3.22), the consistency of the ghost-penalty operator in Lemma 3.4, as well as the stability of the extension operator to estimate

$$
\begin{aligned}
|\mathfrak{T}_5| &= |\gamma_{gp} \nu j_h^{\mathrm{dir}}(\boldsymbol{u}^e, \boldsymbol{v}_h) + \gamma_{gp} \frac{1}{\nu} h^2 j_h^{\mathrm{dir}}(p^e, q_h)| \\
&\lesssim j_h^{\mathrm{dir}}(\boldsymbol{u}^e, \boldsymbol{u}^e)^{1/2} j_h^{\mathrm{dir}}(\boldsymbol{v}_h, \boldsymbol{v}_h)^{1/2} + (h^2 j_h^{\mathrm{dir}}(p^e, p^e))^{1/2} (h^2 j_h^{\mathrm{dir}}(q_h, q_h))^{1/2} \\
&\lesssim h^k \|\boldsymbol{u}^e\|_{\mathcal{H}^{k+1}(\mathcal{O}_{\mathcal{T}})} \|\boldsymbol{v}_h\|_{\mathcal{H}^1(\mathcal{O}_{\mathcal{T}})} + h^k \|p^e\|_{\mathcal{H}^k(\mathcal{O}_{\mathcal{T}})} \|q_h\|_{\mathcal{O}_{\mathcal{T}}} \\
&\lesssim h^k \|\boldsymbol{u}\|_{\mathcal{H}^{k+1}(\Omega)} \|\|\boldsymbol{v}_h\|\|_* + h^k \|p\|_{\mathcal{H}^k(\Omega)} \|\|q_h\|\|_*.
\end{aligned}
$$

Combining the above estimates for the five error contributions then completes the proof. $\qquad\square$

The result shown in Lemma 3.21 can be seen as a disturbed Galerkin orthogonality result and it quantifies the consistency error made through the approximated geometry.

Theorem 3.22 (*Energy error estimate*). *Assume that $\sigma > 0$ is appropriately large, $h > 0$ is sufficiently small and $\gamma_{gp} > 0$ appropriate such that Lemma 3.16 holds true. For the exact solution of the Stokes problem, we assume $(\boldsymbol{u}, p) \in \mathcal{H}^{3,\infty}(\Omega) \times \mathcal{H}^2(\Omega)$ for $k = 2$ or $(\boldsymbol{u}, p) \in \mathcal{H}^{k+1}(\Omega) \times \mathcal{H}^k(\Omega)$ for $k \geq 3$ and for the forcing term, we assume that $\boldsymbol{f} \in \mathcal{H}^1(\Omega)$ is extended sufficiently smooth into Ω_h^{ho} such that Lemma 3.21 holds. Then it holds for the solution $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}$ of (3.8) that*

$$\||(\mathbb{E}, \mathbb{D})\||_A \lesssim h^k R(\boldsymbol{u}, p, \boldsymbol{f})$$

with

$$R(\boldsymbol{u}, p, \boldsymbol{f}) = \begin{cases} \|\boldsymbol{u}\|_{\mathcal{H}^{3,\infty}(\Omega)} + \|p\|_{\mathcal{H}^2(\Omega)} + \|\boldsymbol{f}\|_{\mathcal{H}^1(\Omega)} & \text{for } k = 2 \\ \|\boldsymbol{u}\|_{\mathcal{H}^{k+1}(\Omega)} + \|p\|_{\mathcal{H}^k(\Omega)} + \|\boldsymbol{f}\|_{\mathcal{H}^1(\Omega)} & \text{for } k \geq 3. \end{cases}$$

Proof. Using the triangle inequality, we split the error into the interpolation and approximation parts

$$\||(\mathbb{E}, \mathbb{D})\||_A \leq \||(\boldsymbol{\eta}, \zeta)\||_A + \||(\mathbf{e}_h, \mathrm{d}_h)\||_A.$$

For the discretisation error we use the bound (3.12) and the inf-sup stability of the stabilised bilinear form in Lemma 3.16 to get

$$\||(\mathbf{e}_h, \mathrm{d}_h)\||_A \lesssim \||(\mathbf{e}_h, \mathrm{d}_h)\||_* \lesssim \sup_{(\boldsymbol{v}, q) \in \boldsymbol{V}_h^{iso} \times Q_h^{iso}} \frac{(A_h + J_h)((\mathbf{e}_h, \mathrm{d}_h), (\boldsymbol{v}, q))}{\||(\boldsymbol{v}, q)\||_*}.$$

Adding and subtracting the extended exact solution we then have

$$(A_h + J_h)((\mathbf{e}_h, \mathrm{d}_h), (\boldsymbol{v}, q)) = (A_h + J_h)((\boldsymbol{\eta}, \zeta), (\boldsymbol{v}, q)) + (A_h + J_h)((\mathbb{E}, \mathbb{D}), (\boldsymbol{v}, q)).$$

For the first term we use the continuity of the stabilised form and the interpolation properties (3.15). The second term is the consistency error for which we can apply Lemma 3.21 to conclude the proof. $\square$
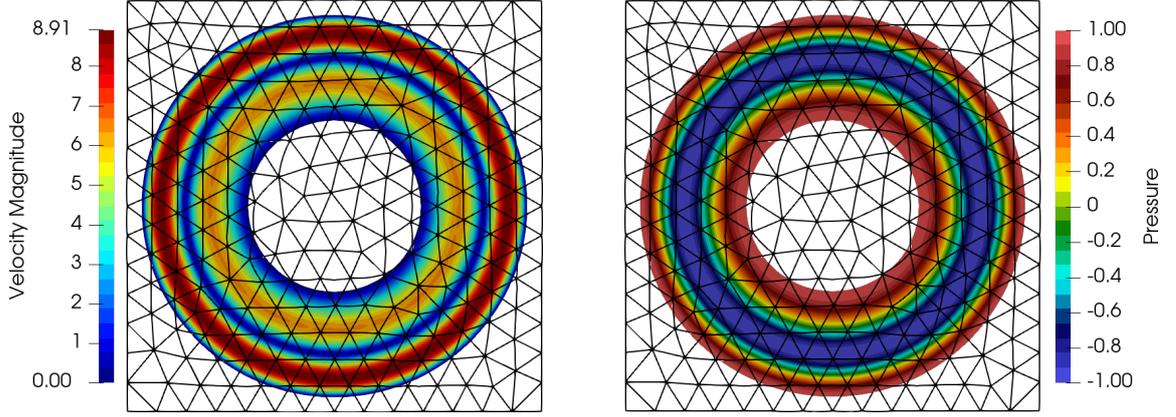
3.2.6 Numerical Examples

We consider a number of numerical examples to investigate the theoretical results in practice. To be able to compute errors, we construct a right-hand side such that the exact solution is known. This is slightly more involved than for fitted FEM, because we do not want to set the Dirichlet condition $\boldsymbol{u}_D = \boldsymbol{u}$ on the approximated boundary, as this would hide the geometry approximation error. We therefore need to set the condition $\boldsymbol{u}_D = \boldsymbol{u}|_\Gamma$ on Γ_h^{ho}. This is easiest, if this is constant. To this end, let us consider the background domain $\widetilde{\Omega} = (-1.7, 1.7)^2$. On this we define the computational domain and corresponding level set function as

$$\Omega := \{\boldsymbol{x} \in \mathbb{R}^2 \mid {}^1\!/{}_2 < \|\boldsymbol{x}\|_2 < {}^5\!/{}_2\} \qquad \text{and} \qquad \phi(\boldsymbol{x}) := \sin(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)).$$

Taking this level set as the stream function, we then construct the velocity field as

$$\boldsymbol{u} = \begin{pmatrix} \partial_{\boldsymbol{x}_2}\phi \\ -\partial_{\boldsymbol{x}_1}\phi \end{pmatrix} = \begin{pmatrix} 2\pi\boldsymbol{x}_2 \cos(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)) \\ -2\pi\boldsymbol{x}_1 \cos(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)) \end{pmatrix},$$

which is divergence-free due to the Schwarz theorem and fulfils homogeneous Dirichlet boundary conditions on Γ. The forcing term is then defined as $\boldsymbol{f} := -\nu\Delta\boldsymbol{u} + \nabla\phi$. Due to the Helmholtz decomposition, changes in the forcing-term by gradient fields are absorbed by the pressure. The pressure solution is therefore $p = \phi$. Furthermore, we then als have that $p \in \mathcal{L}_0^2(\Omega)$. An example numerical solution of this can be seen in Figure 3.3.

Figure 3.3: Velocity and pressure solution of the Stokes problem using CutFEM $\texttt{TH}_4^{\text{iso}}$ elements with the deformed mesh with $h_{\text{max}} = 0.25$, using $\texttt{TH}_4^{\text{iso}}$ elements, Nitsche parameter $\sigma = 40k^2$ and ghost-penalty parameter $\gamma_{gp} = 0.01$.

Set-Up

All the numerical examples presented in this thesis are computed using $\texttt{ngsxfem}$ [Leh+21], an add-on to the high-order finite element library $\texttt{netgen}/\texttt{NGSolve}$ [Sch97; Sch14]. On the active mesh we use Taylor-Hood elements $\mathbb{P}^k/\mathbb{P}^{k-1}$ for $k \geq 2$ [TH73; BF91] which we shall abbreviate as $\texttt{TH}_k$. In case the parametric mapping of order k is applied, we denote this as $\texttt{TH}_k^{\text{iso}}$. We shall consider the viscosity $\nu = 10^{-3}$ and to fix the pressure constant, we use a Lagrange multiplier.

Implementational Efficiency Aspects

Continuous finite element spaces $\mathbb{P}^k$ of order $k \geq 3$ have degrees of freedom which are internal to an element. As a result, these can be removed from the global linear system, by a procedure known as *static condensation* [Guy65; Iro65]. Here, the local unknowns are removed from the global system via a *Schur complement* procedure. Unfortunately, this is incompatible with ghost-penalty stabilisation, which is inherently local on cut elements. We therefore only apply the static condensation procedure to degrees of freedom which stem from elements $T \in \mathscr{T}_h \setminus \mathscr{T}_{h,\Gamma,+}$ and do not participate in the ghost-penalty stabilisation. Since the number of elements in $\mathscr{T}_{h,\Gamma,+}$ is small in comparison to the total number of active elements, we still obtain significant efficiency gains through this procedure, as size of the linear systems that need to be solved is reduced.

Condition Number Estimate

We investigate the condition number estimate in Theorem 3.19 with respect to the ghost-penalty parameter γ_{gp}, for the polynomial orders $k = 2, 5$ with fixed Nitsche penalty parameter $\sigma = 40k^2$. To this end, we move the centre of the ring domain between $(-0.1, 0)$ and $(0.1, 0)$ in 201 equal steps on a fixed mesh with $h_{\text{max}} = 0.2$. As a result we get 201 different mesh-interface cut positions. The ghost-penalty parameter is taken as $\gamma_{gp} \in \{0, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. For simplicity, we take the same parameter for both the velocity and pressure stabilisation. The condition number of the resulting linear system is then estimated by exporting the resulting sparse matrix to $\texttt{MATLAB}$, where the condition number is estimated by $\texttt{condest}$. Note that we consider the full system here, rather than the condensed Schur compliment. The resulting condition number for the different ghost-penalty parameters and cut positions, depending on the centre of the ring domain, can be seen in Figure 3.4.

As we can see in Figure 3.4, the condition number estimate of the unstabilised system is very large and is highly dependent on the mesh-interface cut position. We also see that even a very small ghost-penalty
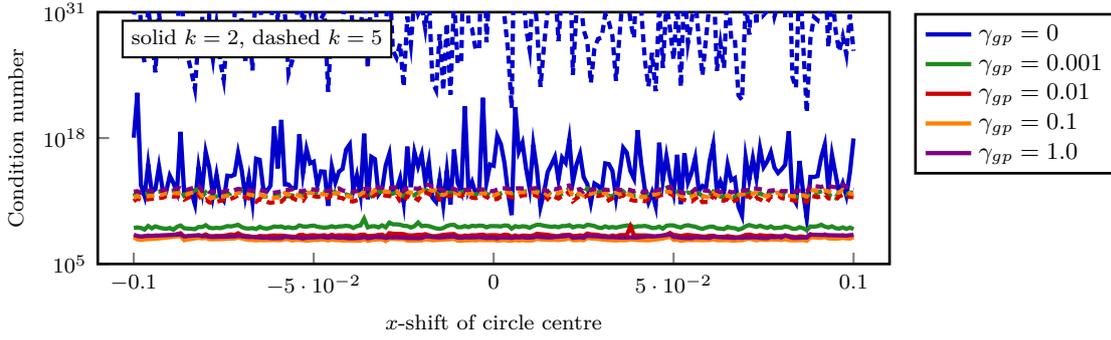
Figure 3.4: Condition number estimate of the linear system resulting from the isoparametric CutFEM Taylor-Hood discretisation of the Stokes problem on a mesh with $h_{\max} = 0.2$ and different mesh-interface cut configurations as well as different ghost-penalty stabilisation parameters.

parameter $\gamma_{gp} = 10^{-3}$ improves the condition number significantly and with $\gamma_{gp} \geq 10^{-1}$, the dependence an the mesh-interface cut position as almost entirely removed. Finally, we note that the condition number increased slightly with the largest testest ghost-penalty parameter $\gamma_{gp} = 1$.

Parameter Dependence

As a second preliminary investigation, we look into the dependence of the error on the Nitsche and ghost-penalty parameters. To this end, we consider a fixed mesh of the background domain with mesh size $h_{\max} = 0.2$. On this mesh, we consider TH_2^{iso} and TH_5^{iso} elements. The ghost-penalty parameter γ_{gp} is chosen between 10^{-1} and 10^{-5} and the Nitsche penalty parameter σ is taken between $4k^2$ and $400k^2$. The resulting velocity errors in the $\mathcal{L}^2(\Omega_h^{\text{ho}})$- and $\mathcal{H}^2(\Omega_h^{\text{ho}})$-norms and the pressure error in the $\mathcal{L}^2(\Omega_h^{\text{ho}})$-norm are shown in Figure 3.5.

Looking at the results, we see that in general, a smaller ghost-penalty parameter is beneficial for the error. However, for $k = 2$, the decrease of the ghost-penalty by four order of magnitude only lead to a decrease in the error by factor of two. For $k = 5$ this decrease is more significant by a factor of ten. Taking the condition number results into account, we see that a balance must be struck when choosing the ghost-penalty parameter. One the one hand, the stabilisation is necessary in order to obtain reasonably conditioned linear systems, on the other, choosing the parameter too large deteriorates the resulting error of the solution.

With respect to the Nitsche parameter, $\sigma = 4k^2$ appears to be too small. However, over the rest of the considered parameter range, this did not affect the results.

Convergence Study

Finally, we investigate the error convergence with respect to the mesh size and the polynomial order of the isoparametric finite element space. To this end, we consider a series of meshes of the background domain, constructed with $0.016 \leq h_{\max} \leq 0.4$. On these meshes we consider TH_k^{iso} elements with $k = 2, 3, 4, 5$. The Nitsche parameter is chosen as $\sigma = 40k^2$ and the ghost penalty parameter is taken as $\gamma_{gp} = 0.01$. The resulting velocity and pressure errors in the relevant norms can be seen in Figure 3.6.

Figure 3.6 shows that we obtain optimal order error convergence for the velocity of order $k + 1$ in the $\mathcal{L}^2$-norm and of order k in the $\mathcal{H}^1$-semi-norm. Furthermore, the pressure error in the $\mathcal{L}^2$-norm also converges optimally with order k, thus illustrating the error estimate in Theorem 3.22.
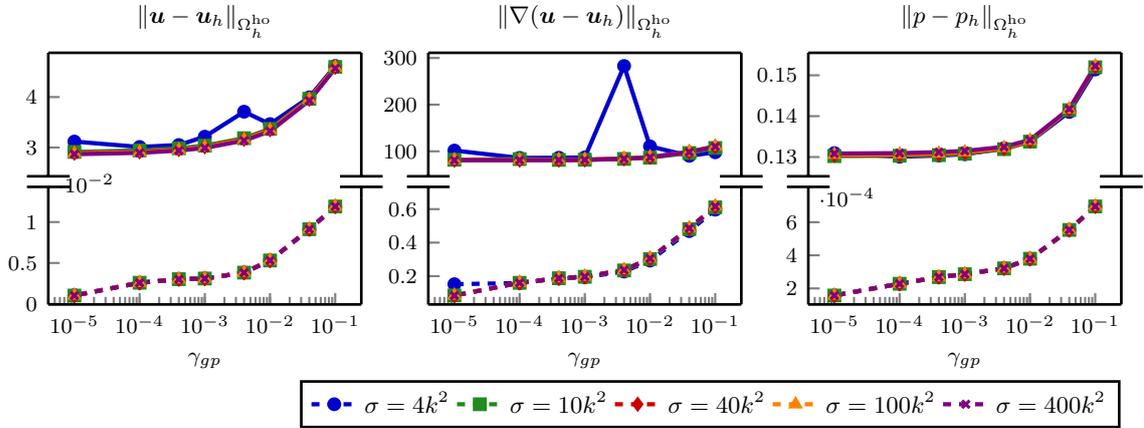
Figure 3.5: Velocity and pressure error resulting from the isoparametric Taylor-Hood CutFEM discretisation for the Stokes problem with different ghost-penalty and Nitsche parameters on a mesh with $h_{\max} = 0.2$. Solid lines indicate $k = 2$ and dashed lines indicate $k = 5$.
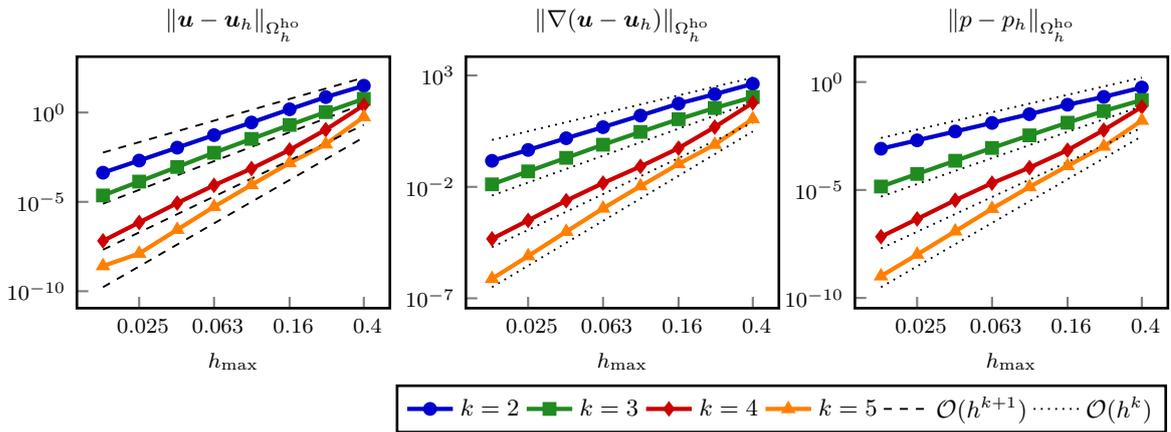
Figure 3.6: Mesh convergence of the velocity- and pressure-error for the isoparametric CutFEM method for the Stokes problem using $\mathtt{TH}_k^{\mathrm{iso}}$ elements.

3.2.7 Summary

The above numerical computations show that the ghost-penalty stabilisation is necessary to obtain stiffness-matrices with condition numbers independent of the interface-mesh cut position and illustrate that the bound in Theorem 3.19 is independent on the interface-mesh cut position. Small stabilisation parameters are sufficient, and the condition number can even increase if the parameter is chosen too large. We have seen that the velocity and pressure errors depend on the choice of ghost-penalty parameters, but this dependence is negligible over a wide range of penalty parameters. Furthermore, we observed that once the Nitsche parameter is sufficiently large, the choice of this parameter has no noticeable effect on the solution error.

With respect to the convergence properties of the method, we saw that the isoparametric CutFEM method with Taylor-Hood elements displays optimal order convergence for both the velocity and pressure. The geometry approximation in the isoparametric CutFEM approach successfully hides the geometry approximation error. Furthermore, this illustrates that our error estimate in Theorem 3.22 is sharp in the order k.

3.3 CutFEM for the Time-Dependent Oseen Problem

Moving towards time-dependent fluid problems, on the way to considering fluid-structure interaction problems, we consider the Oseen problem (2.16) in the context of the isoparametric CutFEM discretisation. The weak formulation of the Oseen problem over a finite time interval $[0, t^{\mathrm{end}}]$ is then given as follows:

Problem P3. Find a velocity and pressure $(\boldsymbol{u}, p) : (0, t^{\mathrm{end}}) \to \boldsymbol{\mathcal{H}}_0^1(\Omega) \times \mathcal{L}_0^2(\Omega)$ with $\boldsymbol{u}(0) = \boldsymbol{u}_0$ such that for all $(\boldsymbol{v}, q) \in \boldsymbol{\mathcal{H}}_0^1(\Omega) \times \mathcal{L}_0^2(\Omega)$ it holds

$$(\partial_t \boldsymbol{u}, \boldsymbol{v})_\Omega + a(\boldsymbol{u}, \boldsymbol{v}) + c(\boldsymbol{b}; \boldsymbol{u}, \boldsymbol{v}) + b(p, \boldsymbol{v}) + b(q, \boldsymbol{u}) = f(\boldsymbol{v}) \tag{3.31}$$

with a given convection field $\boldsymbol{b} \in \mathcal{L}^\infty(0, t^{\mathrm{end}}; \boldsymbol{\mathcal{H}}^{1,\infty}(\Omega))$ with $\mathrm{div}(\boldsymbol{b}) = 0$ and initial condition $\boldsymbol{u}_0 \in \boldsymbol{\mathcal{H}}_0^1(\Omega)$ with $\mathrm{div}(\boldsymbol{u}_0) = 0$.

The assumed spatial regularity of the transport field is standard in the literature for the Oseen equation, see for example [Bra+07]. The (bi-)linear forms $a(\cdot, \cdot)$, $b(\cdot, \cdot)$ and $f(\cdot)$ are defined as for the Stokes problem in (3.7) and the trilinear form $c(\cdot, \cdot, \cdot)$ is given by

$$c(\boldsymbol{b}; \boldsymbol{u}, \boldsymbol{v}) := \int_\Omega \boldsymbol{b} \cdot \nabla \boldsymbol{u} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}.$$

For this form, we have the following properties.

Lemma 3.23. *The trilinear form $c(\cdot, \cdot, \cdot)$ is continuous on $\boldsymbol{\mathcal{H}}_0^1(\Omega) \times \boldsymbol{\mathcal{H}}_0^1(\Omega) \times \boldsymbol{\mathcal{H}}_0^1(\Omega)$ and skew symmetric. That is for $1/p + 1/q + 1/r = 1$ and $\boldsymbol{b} \in \boldsymbol{\mathcal{H}}^1(\Omega)$ with $\mathrm{div}(\boldsymbol{b}) = 0$ and $\boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{\mathcal{H}}_0^1(\Omega)$*

$$|c(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{v})| \leq \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{L}}^r(\Omega)} \|\nabla \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^p(\Omega)} \|\boldsymbol{v}\|_{\boldsymbol{\mathcal{L}}^q(\Omega)}, \quad c(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{v}) = -c(\boldsymbol{b}, \boldsymbol{v}, \boldsymbol{u}) \quad and \quad c(\boldsymbol{b}, \boldsymbol{u}, \boldsymbol{u}) = 0. \tag{3.32}$$

Proof. See for example [BF13, Lemma V.1.1] or [Joh16, Remark 5.6]. □

3.3.1 Discretisation

Let us consider the isoparametric Taylor-Hood spaces $\boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ for a polynomial order $k \geq 2$ for a sufficiently smooth level set domain Ω and a finite time-interval $[0, t^{\mathrm{end}}]$. We the consider a uniform time-step $\Delta t = t^{\mathrm{end}}/N$ for some fixed $N \in \mathbb{N}$ and denote $t^n = n\Delta t$. For simplicity, we take the first-order BDF1 discretisation of the time derivative to formulate the fully discrete scheme. Superscript indices denote the time-step at which the function is evaluated.

Problem P4. With initial data $\boldsymbol{u}_h^0 \in \boldsymbol{V}_h^{\mathrm{iso}}$, for $n = 1, \ldots, N$, find $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ such that for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ it holds

$$\frac{1}{\Delta t} m_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + a_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + \nu i_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + c_h(\boldsymbol{b}, \boldsymbol{u}_h^n, \boldsymbol{v}_h) + b(p_h^n, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h^n) - {}^{1}\!/\!{}_{\nu} j_h(p_h^n, q_h)$$
$$= \frac{1}{\Delta t} m_h(\boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h) + f_h(\boldsymbol{v}_h). \quad (3.33)$$

The bilinear and linear forms $a_h(\cdot, \cdot)$, $b_h(\cdot, \cdot)$ and $f_h(\boldsymbol{v}_h)$ as well as the stabilisation bilinear forms $i_h(\cdot, \cdot)$ and $j_h(\cdot, \cdot)$ are as in the Stokes discretisation (3.8). The remaining bilinear forms are

$$m_h(\boldsymbol{u}_h, \boldsymbol{u}_h) := \int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \qquad \text{and} \qquad c_h(\boldsymbol{b}; \boldsymbol{u}_h, \boldsymbol{v}_h) := \int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{b} \cdot \nabla \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \boldsymbol{b} \cdot \boldsymbol{n} \, \mathrm{d}s$$

with the inflow boundary $\Gamma_{h,\mathrm{in}}^{\mathrm{ho}} := \{\boldsymbol{x} \in \Gamma_h^{\mathrm{ho}} \mid \boldsymbol{b} \cdot \boldsymbol{n} < 0\}$ for a given, divergence-free convective field $\boldsymbol{b} \in \mathcal{H}^{1,\infty}(\Omega_h^{\mathrm{ho}})$. In cases of a non-homogeneous Dirichlet condition for the velocity $\boldsymbol{u}$ on the unfitted inflow boundary $\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}$, we modify the right-hand side appropriately for a consistent formulation, see for example [BH07]. For these mass and transport bilinear forms, we have the following properties.

Lemma 3.24. *For all $\boldsymbol{u}_h, \boldsymbol{v}_h \in \boldsymbol{V}_h^{iso}$ it holds for the mass bilinear forms that*

$$m_h(\boldsymbol{u}_h, \boldsymbol{u}_h) \geq 0 \qquad \text{and} \qquad m_h(\boldsymbol{u}_h, \boldsymbol{v}_h) \lesssim \interleave \boldsymbol{u}_h \interleave_* \interleave \boldsymbol{v}_h \interleave_*,$$

and for the transport bilinear form it holds

$$c_h(\boldsymbol{b}, \boldsymbol{u}_h, \boldsymbol{u}_h) \geq 0 \qquad \text{and} \qquad c_h(\boldsymbol{b}, \boldsymbol{u}_h, \boldsymbol{v}_h) \lesssim \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{L}}^\infty(\Omega_h^{ho})} \interleave \boldsymbol{u}_h \interleave_* \interleave \boldsymbol{v}_h \interleave_*.$$

Proof. The first inequality for the mass bilinear form is trivial, since $m_h(\boldsymbol{u}_h, \boldsymbol{u}_h) = \|\boldsymbol{u}_h\|_{\Omega_h^{\mathrm{ho}}}^2 \geq 0$. The second inequality follows by the Cauchy-Schwarz inequality and the Poincaré inequality (3.11).

For the first inequality for the transport term, we see that with integration by parts that

$$\int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{b} \cdot \nabla \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} = -\int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{div}(\boldsymbol{b}) + \boldsymbol{b} \cdot \nabla \boldsymbol{v}_h \cdot \boldsymbol{u}_h \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma_h^{\mathrm{ho}}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \boldsymbol{b} \cdot \boldsymbol{n} \, \mathrm{d}s.$$

Due to the divergence-free nature of $\boldsymbol{b}$, it holds

$$c_h(\boldsymbol{b}; \boldsymbol{u}_h, \boldsymbol{v}_h) = \frac{1}{2} \| |\boldsymbol{b} \cdot \boldsymbol{n}|^{1/2} \boldsymbol{u}_h \|_{\Gamma_h^{\mathrm{ho}}}^2$$

by which the claim follows. The second inequality is a simple application of the generalised Hölder inequality, as well as the Poincaré inequality (3.11). $\qquad \square$

Lemma 3.25 (*Unique solvability*). *For every time-step $n = 1, \ldots, N$, the isoparametric CutFEM discretisation (3.33) of the Oseen equations admits a unique solution, bounded by the data.*

Proof. Due to Lemma 3.24, the bilinear form

$$A_h^{\mathrm{os}}((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) := \frac{1}{\Delta t} m_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + a_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + \nu i_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + c_h(\boldsymbol{b}, \boldsymbol{u}_h^n, \boldsymbol{v}_h)$$
$$+ b(p_h^n, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h^n) - {}^{1}\!/\!{}_{\nu} j_h(p_h^n, q_h)$$

is bounded on the product space $\boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$ in the $\interleave(\cdot, \cdot)\interleave_*$-norm. Also due to Lemma 3.24, the inf-sup property (3.23) can be proven completely analogously for $A_h^{\mathrm{os}}(\cdot, \cdot)$. Therefore the claim follows by applying the Banach-Nečas-Babuška theorem recursively. $\qquad \square$

Lemma 3.26 (*Condition number estimate*). *For sufficiently small $h > 0$, the stiffness matrix $\mathscr{A}^{os}$ of the isoparametric CutFEM discretisation (3.33) of the Oseen problem, has a condition number that is independent of the mesh-interface cut position and is bounded by*

$$\kappa(\mathscr{A}^{os}) \lesssim h^{-2}.$$

Proof. Due to Lemma 3.24, all the necessary properties needed for the bilinear form in the Stokes case, also hold for the Oseen bilinear form. The proof is therefore identical to that of Theorem 3.19. □

A Priori Error Estimate

Before we come to the energy estimate, we see that with integration by parts it holds for the strong solution of the Oseen problem the weak Galerkin orthogonality

$$\int_\Omega \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} + A((\boldsymbol{u}, p), (\boldsymbol{v}_h^\ell, q_h^\ell)) + c(\boldsymbol{b}; \boldsymbol{u}, \boldsymbol{v}_h^\ell) = f(\boldsymbol{v}_h^\ell) \qquad \text{for all } (\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$$

with the Stokes bilinear form $A(\cdot, \cdot)$ and $\boldsymbol{v}_h^\ell = \boldsymbol{v}_h \circ \Phi_h^{-1}$, $q_h^\ell = q_h \circ \Phi_h^{-1}$. Subtracting this from the discrete formulation (3.33) and rearranging terms, we then get the error equation for the isoparametric CutFEM BDF1 discretisation

$$\frac{1}{\Delta t} \int_{\Omega_h^{\mathrm{ho}}} (\mathbb{E}^n - \mathbb{E}^{n-1}) \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + (A_h + J_h)((\mathbb{E}^n, \mathbb{D}^n), (\boldsymbol{v}_h, q_h)) + c_h(\boldsymbol{b}; \mathbb{E}^n, \boldsymbol{v}_h)$$

$$= f(\boldsymbol{v}_h^\ell) - f_h(\boldsymbol{v}_h) + \int_{\Omega_h^{\mathrm{ho}}} \frac{1}{\Delta t}(\boldsymbol{u}^n - \boldsymbol{u}^{n-1}) \cdot \boldsymbol{v}_h \, \mathrm{d}x - \int_\Omega \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}}$$

$$+ a_h(\boldsymbol{u}, \boldsymbol{v}_h) - a^1(\boldsymbol{u}, \boldsymbol{v}_h^\ell) + b_h(p, \boldsymbol{v}_h) - b^1(p, \boldsymbol{v}_h^\ell) + b_h(q_h, \boldsymbol{u}) - b^1(q_h^\ell, \boldsymbol{u})$$

$$+ i_h(\boldsymbol{u}^e, \boldsymbol{v}_h) - j_h(p^e, q_h) + c_h(\boldsymbol{b}; \boldsymbol{u}_h^n, \boldsymbol{v}_h) - c(\boldsymbol{b}; \boldsymbol{u}, \boldsymbol{v}_h^\ell)$$

$$=: \mathfrak{T}_1 + \mathfrak{T}_2 + \mathfrak{T}_3 + \mathfrak{T}_4 + \mathfrak{T}_5 + \mathfrak{T}_6 + \mathfrak{T}_7$$

$$= \mathfrak{E}_c^{\mathrm{os}}(\boldsymbol{v}_h, q_h). \tag{3.34}$$

Remark 3.27. In contrast to standard methods with exact geometry handling, where the only contribution would result from the discretisation of the time derivative $\mathfrak{T}_2$, we have the geometry consistency errors from each of the terms in the original equation and the consistency error introduced by the direct ghost-penalty stabilisation. ▲

Lemma 3.28 (*Consistency estimates*). *Assume that the strong solution to the Oseen equation has the spatial regularity $(\boldsymbol{u}(t), p(t)) \in \boldsymbol{\mathcal{H}}^{k+1}(\Omega) \times \mathcal{H}^k(\Omega)$ as well as the temporal regularity*

$$\partial_t^2 \boldsymbol{u} \in \mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{L}}^2(\Omega)) \qquad and \qquad \partial_t \boldsymbol{u} \in \mathcal{L}^2(0, t^{end}; \boldsymbol{\mathcal{H}}^1(\Omega)).$$

Furthermore, we have that the divergence-free transport field has the regularity $\boldsymbol{b} \in \boldsymbol{\mathcal{H}}^{1,\infty}(\Omega)$. Then for the consistency error of the isoparametric CutFEM BDF1 discretisation of the Oseen problem, we have that

$$|\mathfrak{E}_c^{os}(\boldsymbol{v}_h, q_h)| \lesssim (\Delta t + h^k) R_c^1(\boldsymbol{u}, p, \boldsymbol{f}, \boldsymbol{b}) \|\|\boldsymbol{v}_h\|\|_* + h^k R_c^2(\boldsymbol{u}, p) \|\|q_h\|\|_* \tag{3.35}$$

with $R_c^1(\boldsymbol{u}, p, \boldsymbol{f}, \boldsymbol{b}) = \|\partial_t^2 \boldsymbol{u}\|_{\mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{L}}^2(\Omega))} + \|\partial_t \boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^1(\Omega)} + \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega)} + \|p\|_{\mathcal{H}^2(\Omega)} + \|\boldsymbol{f}\|_{\boldsymbol{\mathcal{H}}^1(\Omega)} + \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{H}}^{1,\infty}(\Omega)}$ and $R_c^2(\boldsymbol{u}, p) = \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} + \|p\|_{\mathcal{H}^k(\Omega)}$.

Proof. The consistency components $\mathfrak{T}_1$ and $\mathfrak{T}_3 - \mathfrak{T}_6$ correspond to the five consistency components of the Stokes discretisation in (3.27). We therefore only have to deal with the temporal and transport terms. We omit the explicit notation of the extension for better readability, by identifying $\boldsymbol{u}$ with its extension $\mathcal{E}\boldsymbol{u}$. As in Lemma 3.21, the proof is a combination of the use of the chain rule and Lemma 3.10.

<u>Consistency term 2:</u> $\mathfrak{T}_2 = \int_{\Omega_h^{\mathrm{ho}}} \frac{1}{\Delta t}(\boldsymbol{u}^n - \boldsymbol{u}^{n-1}) \cdot \boldsymbol{v}_h \,\mathrm{d}x - \int_\Omega \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{\boldsymbol{x}}$. Using partial integration (product rule for integrals), we have

$$\mathfrak{T}_2 = -\int_{\Omega_h^{\mathrm{ho}}} \int_{t^{n-1}}^{t^n} \frac{t - t^{n-1}}{\Delta t} \partial_t^2 \boldsymbol{u} \,\mathrm{d}t \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} + \int_{\Omega_h^{\mathrm{ho}}} \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} - \int_\Omega \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{\boldsymbol{x}}.$$

The first term can be bound by

$$\Big| \int_{\Omega_h^{\mathrm{ho}}} \int_{t^{n-1}}^{t^n} \frac{t - t^{n-1}}{\Delta t} \partial_t^2 \boldsymbol{u} \,\mathrm{d}t \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} \Big| \leq \frac{1}{2} \Delta t \|\partial_t^2 \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^\infty(t^{n-1},t^n;\Omega_h^{\mathrm{ho}})} \|\boldsymbol{v}_h\|_{\boldsymbol{\mathcal{L}}^1(\Omega_h^{\mathrm{ho}})}$$
$$\lesssim \Delta t \|\partial_t^2 \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^\infty(0,t^{\mathrm{end}};\Omega)} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

For the remaining part, we have

$$\int_{\Omega_h^{\mathrm{ho}}} \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} - \int_\Omega \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{\boldsymbol{x}} = \int_{\Omega_h^{\mathrm{ho}}} (\partial_t \boldsymbol{u} - (\partial_t \boldsymbol{u}) \circ \Phi_h) \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} + \int_{\Omega_h^{\mathrm{ho}}} (\partial_t \boldsymbol{u}) \circ \Phi_h \cdot \boldsymbol{v}_h (1 - J) \,\mathrm{d}\boldsymbol{x}$$
$$\lesssim h^k \|\partial_t \boldsymbol{u}\|_{\mathcal{H}^1(\Omega)} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}} + h^k \|\partial_t \boldsymbol{u}\|_\Omega \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}},$$

where $J = \det(D\Phi_h)$. In sum we therefore have

$$|\mathfrak{T}_2| \lesssim (\Delta t + h^k)(\|\partial_t^2 \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^\infty(0,t^{\mathrm{end}};\boldsymbol{\mathcal{L}}^2(\Omega))} + \|\partial_t \boldsymbol{u}\|_{\mathcal{H}^1(\Omega)}) \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

<u>Consistency term 7:</u> $\mathfrak{T}_7 = c_h(\boldsymbol{b}; \boldsymbol{u}^n, \boldsymbol{v}_h) - c(\boldsymbol{b}; \boldsymbol{u}^n, \boldsymbol{v}_h^\ell)$. For the volume integral part, we have with the regularity of $\boldsymbol{b}$ that

$$\int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{b} \cdot \nabla \boldsymbol{u}^n \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} - \int_\Omega \boldsymbol{b} \cdot \hat{\nabla} \boldsymbol{u}^n \cdot \boldsymbol{v}_h^\ell \,\mathrm{d}\hat{\boldsymbol{x}} = \int_{\Omega_h^{\mathrm{ho}}} (\boldsymbol{b} - \boldsymbol{b} \circ \Phi_h) \cdot \nabla \boldsymbol{u}^n \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x}$$
$$+ \int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{b} \circ \Phi_h \cdot \nabla(\boldsymbol{u}^n - \boldsymbol{u}^n \circ \Phi_h) \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x}$$
$$+ \int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{b} \circ \Phi_h \cdot (I - J D\Phi_h^{-T}) \nabla(\boldsymbol{u} \circ \Phi_h) \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x}$$
$$\lesssim h^k \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{H}}^{1,\infty}(\Omega)} \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

For the boundary part, we have with the trace estimate (3.10) and the inverse estimate (3.9a) that

$$-\int_{\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}} \boldsymbol{u}^n \cdot \boldsymbol{v}_h \, \boldsymbol{b} \cdot \boldsymbol{n} \,\mathrm{d}s + \int_{\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}} \underbrace{\boldsymbol{u}^n \circ \Phi_h}_{=0} \cdot \boldsymbol{v}_h \, \boldsymbol{b} \cdot \boldsymbol{n} \,\mathrm{d}s \leq \|\boldsymbol{u}^n - \boldsymbol{u}^n \circ \Phi_h\|_{\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}} \|\boldsymbol{v}_h\|_{\Gamma_{h,\mathrm{in}}^{\mathrm{ho}}} \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{L}}^\infty(\Gamma_{h,\mathrm{in}}^{\mathrm{ho}})}$$
$$\lesssim h^{k+1/2} \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|\boldsymbol{v}_h\|_{\mathcal{O}_{\mathcal{T}}} \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{L}}^\infty(\Omega)}.$$

This means that we can estimate the transport consistency error by

$$|\mathfrak{T}_7| \lesssim h^k \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{H}}^{1,\infty}(\Omega)} \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\Omega)} \|\boldsymbol{v}_h\|_{\mathcal{O}_{\mathcal{T}}}.$$

Combining the above estimates for the individual consistency error contributions then completes the proof. □

As in (3.26), we split the velocity and pressure errors into an interpolation error and a discretisation error. Inserting this into (3.34) and rearranging terms then gives the following error equation for the discretisation error

$$\frac{1}{\Delta t} \int_{\Omega_h^{\mathrm{ho}}} (\mathbf{e}_h^n - \mathbf{e}_h^{n-1}) \cdot \boldsymbol{v}_h \,\mathrm{d}\boldsymbol{x} + a_h(\mathbf{e}_h^n, \boldsymbol{v}_h) + c_h(\boldsymbol{b}, \mathbf{e}_h^n, \boldsymbol{v}_h) + b_h(\mathrm{d}_h^n, \boldsymbol{v}_h) + b_h(q_h, \mathbf{e}_h^n)$$
$$+ \nu i_h(\mathbf{e}_h^n, \boldsymbol{v}_h) - \frac{1}{\nu} j_h(\mathrm{d}_h^n, q_h) = \mathfrak{E}_c^{\mathrm{os}}(\boldsymbol{v}_h, q_h) + \mathfrak{E}_i^{\mathrm{os}}(\boldsymbol{v}_h, q_h) \quad (3.36)$$

with the interpolation error component

$$\mathfrak{E}_i^{\mathrm{os}}(\boldsymbol{v}_h, q_h) = -\frac{1}{\Delta t} \int_{\Omega_h^{\mathrm{ho}}} (\boldsymbol{\eta}^n - \boldsymbol{\eta}^{n-1}) \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - a_h(\boldsymbol{\eta}^n, \boldsymbol{v}_h) - c_h(\boldsymbol{b}, \boldsymbol{\eta}^n, \boldsymbol{v}_h)$$

$$- b_h(\zeta^n, \boldsymbol{v}_h) - b_h(q_h, \boldsymbol{\eta}^n) - \nu i_h(\boldsymbol{\eta}^n, \boldsymbol{v}_h) + \frac{1}{\nu} j_h(\zeta^n, q_h). \quad (3.37)$$

Lemma 3.29. *Assume that the exact solution to the Oseen problem fulfils the regularity assumptions $\boldsymbol{u} \in \mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{H}}^{k+1}(\Omega))$, $\partial_t \boldsymbol{u} \in \mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{H}}^k(\Omega))$ and $p \in \mathcal{L}^\infty(0, t^{end}; \mathcal{H}^k(\Omega))$. For the interpolation error contribution it then holds*

$$|\mathfrak{E}_i^{os}(\boldsymbol{v}_h, q_h)| \lesssim h^k R_i(\boldsymbol{u}, p) \|\|(\boldsymbol{v}_h, q_h)\|\|_*$$

with $R_i(\boldsymbol{u}, p) := \|\boldsymbol{u}\|_{\mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{H}}^{k+1}(\Omega))} + \|\partial_t \boldsymbol{u}\|_{\mathcal{L}^\infty(0, t^{end}; \boldsymbol{\mathcal{H}}^k(\Omega))} + \|p\|_{\mathcal{L}^\infty(0, t^{end}; \mathcal{H}^k(\Omega))}.$

Proof. Let us consider the individual contributions in (3.37). The estimates for the diffusion, transport, pressure coupling and mass-conservation terms follow by their continuity and the interpolation properties (3.15). For the ghost-penalty terms, the estimate follows from Lemma 3.4 and the stability of the (implicitly included) extension operator. For the time-derivative term, we observe that by the stability of the extension it holds that

$$\|\partial_t \boldsymbol{\eta}^n\|_{\Omega_h^{\mathrm{ho}}} \lesssim h^k \|\partial_t \boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^k(\mathcal{O}_{\mathscr{T}})} \lesssim h^k \|\partial_t \boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^k(\Omega)}.$$

Then with the Cauchy-Schwarz inequality we have

$$\left| \frac{1}{\Delta t} \int_{\Omega_h^{\mathrm{ho}}} (\boldsymbol{\eta}^n - \boldsymbol{\eta}^{n-1}) \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \right| \leq \frac{1}{\Delta t} \left\| \int_{t^{n-1}}^{t^n} \partial_t \boldsymbol{\eta}(t) \, \mathrm{d}t \right\|_{\Omega_h^{\mathrm{ho}}} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}$$

$$\leq \Delta t^{-1/2} \left(\int_{t^{n-1}}^{t^n} \|\partial_t \boldsymbol{\eta}^n\|_{\Omega_h^{\mathrm{ho}}}^2 \, \mathrm{d}t \right)^{1/2} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}} \lesssim h^k \sup_{t \in [t^{n-1}, t^n]} \|\partial_t \boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^k(\mathcal{O}_{\mathscr{T}})} \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}$$

$$\lesssim h^k \sup_{t \in [0, t^{\mathrm{end}}]} \|\partial_t \boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^k(\Omega)}) \|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}. \quad (3.38)$$

$\square$

Theorem 3.30 (*Energy error estimate*). *Under the above regularity assumptions on the data and exact solution, such that Lemma 3.28 and Lemma 3.29 hold true, we have for the velocity error of the isoparametric CutFEM discretisation with a BDF1 discretisation of the time-derivative, that*

$$\|\mathbb{E}^n\|_{\Omega_h^{ho}}^2 + \Delta t \sum_{k=1}^n \nu \|\|\mathbb{E}^n\|\|_*^2 \lesssim \|\mathbb{E}^0\|_{\Omega_h^{ho}}^2 + t^n (\Delta t + h^{2k}/\Delta t + \Delta t h^{2k}) R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f})$$

with

$$R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f}) := \sup_{t \in [0, t^n]} \left[\|\partial_t^2 \boldsymbol{u}\|_\Omega^2 + \|\partial_t \boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^k(\Omega)}^2 + \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega)}^2 + \|p\|_{\mathcal{H}^k(\Omega)}^2 + \|\boldsymbol{f}\|_{\boldsymbol{\mathcal{H}}^1(\Omega)}^2 + \|\boldsymbol{b}\|_{\boldsymbol{\mathcal{H}}^{1, \infty}(\Omega)}^2 \right].$$

Proof. We show the claim for the discretisation error. The result for the full error follows then by the optimal interpolation properties of the finite element space. We test the error equation (3.36) with $2\Delta t(\mathbf{e}_h^n - \mathrm{d}_h^n)$, which gives

$$\|\mathbf{e}_h^n\|_{\Omega_h^{ho}}^2 + \|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{ho}}^2 - \|\mathbf{e}_h^{n-1}\|_{\Omega_h^{ho}} + 2\Delta t(a_h(\mathbf{e}_h^n, \mathbf{e}_h^n) + \nu i_h(\mathbf{e}_h^n, \mathbf{e}_h^n) + c_h(\boldsymbol{b}, \mathbf{e}_h^n, \mathbf{e}_h^n))$$

$$+ 2\Delta t \frac{1}{\nu} j_h(\mathrm{d}_h^n, \mathrm{d}_h^n) = 2\Delta t \left(\mathfrak{E}_c^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n) + \mathfrak{E}_i^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n) \right). \quad (3.39)$$

Using the coercivity of the stabilised diffusion form in Lemma 3.14, the skew symmetry of the transport term in Lemma 3.24 and the positivity of the pressure ghost-penalty term, we then have

$$\|\mathbf{e}_h^n\|_{\Omega_h^{\mathrm{ho}}}^2 + \|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}^2 + 2\Delta t\|\mathbf{e}_h^n\|_* \lesssim \|\mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}} + 2\Delta t\left(|\mathfrak{E}_c^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n)| + |\mathfrak{E}_i^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n)|\right). \tag{3.40}$$

With the estimates of Lemma 3.28 and Lemma 3.29 and after an application of Young's inequality and a weighted Young's inequality, we then have for the right-hand side terms that

$$2\Delta t\left(\mathfrak{E}_c^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n) + \mathfrak{E}_i^{\mathrm{os}}(\mathbf{e}_h^n, \mathrm{d}_h^n)\right) \lesssim \Delta t\left(1/\varepsilon(\Delta t^2 + h^{2k})R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f}) + \varepsilon\||(\mathbf{e}_h^n, \mathrm{d}_h^n)\||_*^2\right) \tag{3.41}$$

with $R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f})$ as defined above. Now, it is clear from Lemma 3.16 and Lemma 3.24 that the inf-sup condition (3.23) also holds if we add the transport term to the bilinear form. That is, for

$$A_{h,1}^{\mathrm{os}}((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) \coloneqq a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + c_h(\boldsymbol{b}, \boldsymbol{u}_h, \boldsymbol{v}_h) + b(p_h, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h)$$

we have

$$\||(\boldsymbol{u}_h, p_h)\||_* \lesssim \sup_{(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}} \frac{(A_{h,1}^{\mathrm{os}} + J_h)((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h))}{\||(\boldsymbol{v}_h, q_h)\||_*}$$

for all $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}$. So with (3.34) it holds

$$(A_{h,1}^{\mathrm{os}} + J_h)((\mathbf{e}_h^n, \mathrm{d}_h^n), (\boldsymbol{v}_h, q_h)) = (A_{h,1}^{\mathrm{os}} + J_h)((\boldsymbol{\eta}^n, \zeta^n), (\boldsymbol{v}_h, q_h)) + (A_{h,1}^{\mathrm{os}} + J_h)((\mathbb{E}^n, \mathbb{D}^n), (\boldsymbol{v}_h, q_h))$$

$$\lesssim \||(\boldsymbol{\eta}^n, \zeta^n)\||_*\||\boldsymbol{v}_h, q_h)\||_* + |\mathfrak{E}_c^{\mathrm{os}}(\boldsymbol{v}_h, q_h)| + \frac{1}{\Delta t}\|\mathbb{E}^n - \mathbb{E}^{n-1}\|_{\Omega_h^{\mathrm{ho}}}\|\boldsymbol{v}_h\|_{\Omega_h^{\mathrm{ho}}}.$$

So by splitting the error term $1/\Delta t\|\mathbb{E}^n - \mathbb{E}^{n-1}\|_{\Omega_h^{\mathrm{ho}}}$ into the interpolation and discretisation error parts, it then follows with (3.38), the consistency error estimate (3.35) and the interpolation properties (3.15), that it holds

$$\||(\mathbf{e}_h^n, \mathrm{d}_h^n)\|| \lesssim \sup_{(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\mathrm{iso}} \times Q_h^{\mathrm{iso}}} \frac{(A_{h,1}^{\mathrm{os}} + J_h)((\mathbf{e}_h^n, \mathrm{d}_h^n), (\boldsymbol{v}_h, q_h))}{\||(\boldsymbol{v}_h, q_h)\||_*}$$

$$\lesssim h^k(R_i(\boldsymbol{u}, p) + R_c(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f})) + \frac{1}{\Delta t}\|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}.$$

So that with the weighted Young's inequality, we have

$$\Delta t\varepsilon\||(\mathbf{e}_h^n, \mathrm{d}_h^n)\||^2 \lesssim \varepsilon\Delta t h^{2k}R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f}) + \frac{\varepsilon}{\Delta t}\|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}^2.$$

With ε sufficiently small, that is $\varepsilon \lesssim \Delta t$ we get that $c\varepsilon/\Delta t\|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}^2 \leq \|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}^2$. Inserting this into (3.41), we get in (3.40) that

$$\|\mathbf{e}_h^n\|_{\Omega_h^{\mathrm{ho}}}^2 + 2\Delta t\|\mathbf{e}_h^n\|_* \lesssim \|\mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}} + (\Delta t^2 + h^{2k} + \Delta t^2 h^{2k})R(\boldsymbol{u}, p, \boldsymbol{b}, \boldsymbol{f}).$$

Summing over $n = 1, \ldots, N$ and factoring out a factor Δt on the right-hand side then gives the result. □

Remark 3.31. We observe that the above error estimate is not optimal in time. This is due to a suboptimal bound for $\|\frac{1}{\Delta t}(\mathbf{e}_h^n - \mathbf{e}_h^{n-1})\|_{\Omega_h^{\mathrm{ho}}}$. In fact, we lost half an order in the temporal convergence due to the necessary choice of $\varepsilon \sim \Delta t$ after the applications of the weighted Youngs inequality, to absorb the $\|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}^2$ term in the left hand side. However, since we can expect that $\|\mathbf{e}_h^n - \mathbf{e}_h^{n-1}\|_{\Omega_h^{\mathrm{ho}}}$ scales with a factor Δt, we do not expect that the above estimate is sharp in Δt.

In standard fitted and conforming finite element methods, an optimal bound for $\|\frac{1}{\Delta t}(\mathbf{e}_h^n - \mathbf{e}_h^{n-1})\|_{\Omega_h^{\mathrm{ho}}}$ is obtained by testing the error equation with $\boldsymbol{v}_h = \frac{1}{\Delta t}(\mathbf{e}_h^n - \mathbf{e}_h^{n-1}) =: \delta\mathbf{e}_h^n$ and using the fact that $\delta\mathbf{e}_h^n$, with the interpolation defined via a Stokes projector, is weakly divergence-free to remove the pressure from the resulting equation, c.f. [EG21c]. Following the same techniques in our setting did not lead to a satisfactory result since the pressure discretisation error remains part of the equation such that additional terms remain for which a bound is not clear. ▲

Remark 3.32 (BDF2 time-discretisation). To extend the above analysis to the BDF2 discretisation of the time-derivative $\partial_t \boldsymbol{u} \approx \frac{1}{\Delta t} \left(\frac{3}{2} \boldsymbol{u}^n - 2 \boldsymbol{u}^{n-1} + \frac{1}{2} \boldsymbol{u}^{n-2} \right)$, we require the following modifications. First, we replace the polarisation identity used in (3.39) with the polarisation identity

$$2(3\boldsymbol{u}_h^n - 4\boldsymbol{u}_h^{n-1} + \boldsymbol{u}_h^{n-2}, \boldsymbol{u}_h^n)_{\Omega_h} = \|\boldsymbol{u}_h^n\|_{\Omega_h}^2 + \|2\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_{\Omega_h}^2 + \|\boldsymbol{u}_h^n - 2\boldsymbol{u}_h^{n-1} + \boldsymbol{u}_h^{n-2}\|_{\Omega_h}^2$$
$$- \|\boldsymbol{u}_h^{n-1}\|_{\Omega_h}^2 - \|2\boldsymbol{u}_h^{n-1} - \boldsymbol{u}_h^{n-2}\|_{\Omega_h}^2, \quad (3.42)$$

c.f. (6.33) in [EG04]. For an equivalent of the estimate (3.38), we observe that

$$\frac{3}{2}\boldsymbol{u}_h^n - 2\boldsymbol{u}_h^{n-1} + \frac{1}{2}\boldsymbol{u}_h^{n-2} = \frac{3}{2}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}) - \frac{1}{2}(\boldsymbol{u}_h^{n-1} - \boldsymbol{u}_h^{n-2})$$

so that the proof for (3.38) can be applied to each of these two components. Furthermore, the have the improved consistency estimate

$$(1/2\Delta t (3\boldsymbol{u}_h^n - 4\boldsymbol{u}_h^{n-2} + \boldsymbol{u}_h^{n-2}), \boldsymbol{v}_h)_{\Omega_h} \lesssim \Delta t^2 \|\partial_t^3 \boldsymbol{u}_h\|_{\mathcal{L}^\infty(0, t^{\mathrm{end}}; \boldsymbol{\mathcal{L}}^2(\Omega_h))} \|\boldsymbol{v}_h\|_{\Omega_h}$$

[HNW93]. The final component is then a sufficiently accurate approximation of $\boldsymbol{u}_h^1$, in order to get second-order convergence. ▲

3.3.2 Numerical Examples

To construct an analytical solution on a suitable level set domain, we take the same domain as for the Stokes computations in subsection 3.2.6 and construct the exact solution based on that considered for Stokes. To not affect the boundary conditions, we add a multiplicative temporal component to the exact solution, which gives

$$\boldsymbol{u}^{\mathrm{os}} = \sin(\pi t) \begin{pmatrix} 2\pi \boldsymbol{x}_2 \cos(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)) \\ -2\pi \boldsymbol{x}_1 \cos(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)) \end{pmatrix} \quad \text{and} \quad p^{\mathrm{os}} = \sin(\pi t) \sin(\pi(\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2)).$$

We then take the transport field to be the exact solution, i.e., $\boldsymbol{b} = \boldsymbol{u}^{\mathrm{os}}$ and consider the time interval $[0, 1]$. Unless otherwise indicated, we consider $\nu = 10^{-3}$, $\sigma = 40k^2$ and $\gamma_{gp} = 0.01$. The right-hand side $\boldsymbol{f}$ is then constructed by plugging in the exact solution into the Oseen equation (2.16). Finally, to measure the space-time norms we use the time-discrete norm

$$\|\cdot\|_{\ell^2(X)}^2 := \Delta t \sum_{i=1}^n \|\cdot\|_X^2$$

for a given normed space X.

Condition Number Estimate

We consider the identical set-up as in subsection 3.2.6. The resulting condition number estimates for the system from a BDF1 time-discretisation in the first time-step with a time-step size of $\Delta t = 1/100$ can be seen in Figure 3.7.

We observe that the results are very similar to the Stokes case in subsection 3.2.6. Indeed, the ghost-penalty stabilisation results in condition numbers that are almost independent of the mesh-interface cut configuration. Furthermore, we also observe that as in the Stokes case, the condition number increased again for $\gamma_{gp} = 1$. Therefore, in the sense of optimal condition number for the system, the ghost-penalty parameter should not be chosen too large.

Convergence

We consider the temporal and spatial convergence properties of the discretisation separately.
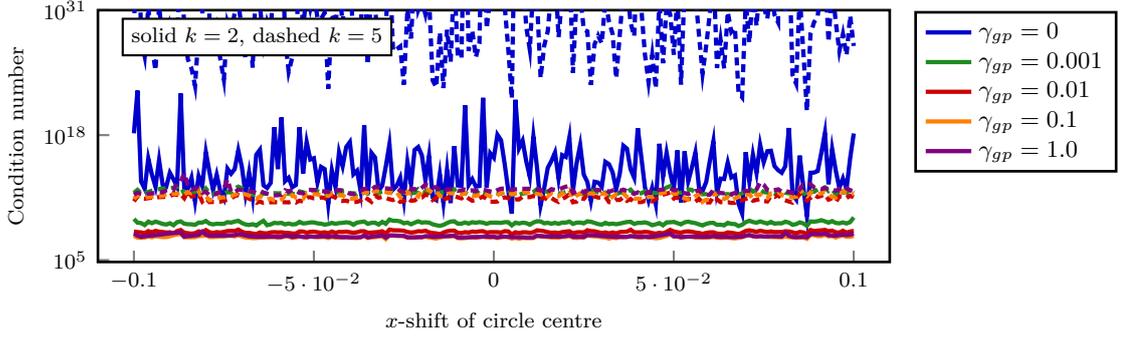
Figure 3.7: Condition number estimate of the linear system resulting from the isoparametric CutFEM Taylor-Hood discretisation for the Oseen problem on a mesh with $h_{\max} = 0.2$ with different mesh-interface cut configurations and ghost-penalty stabilisation parameters. The time-discretisation is with the BDF1 method and the time=step is chosen as $\Delta t = 1/100$.
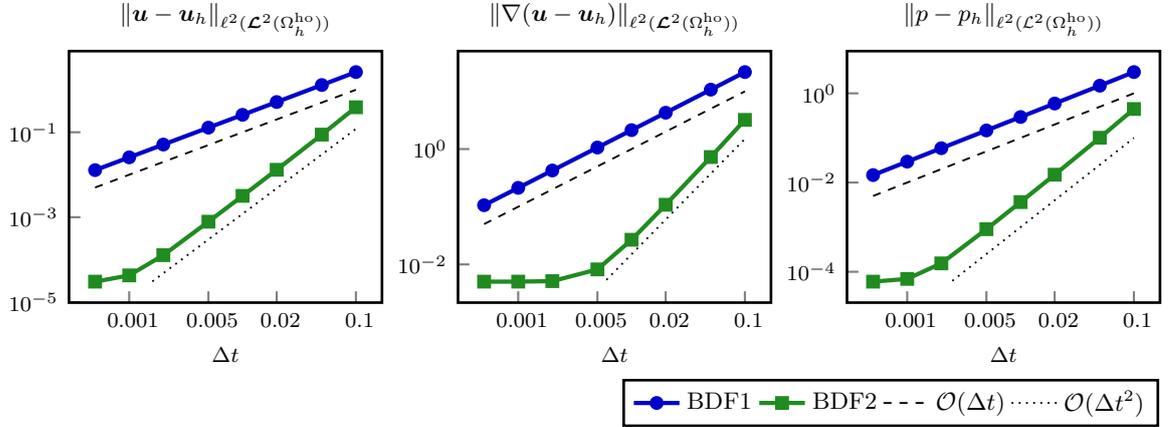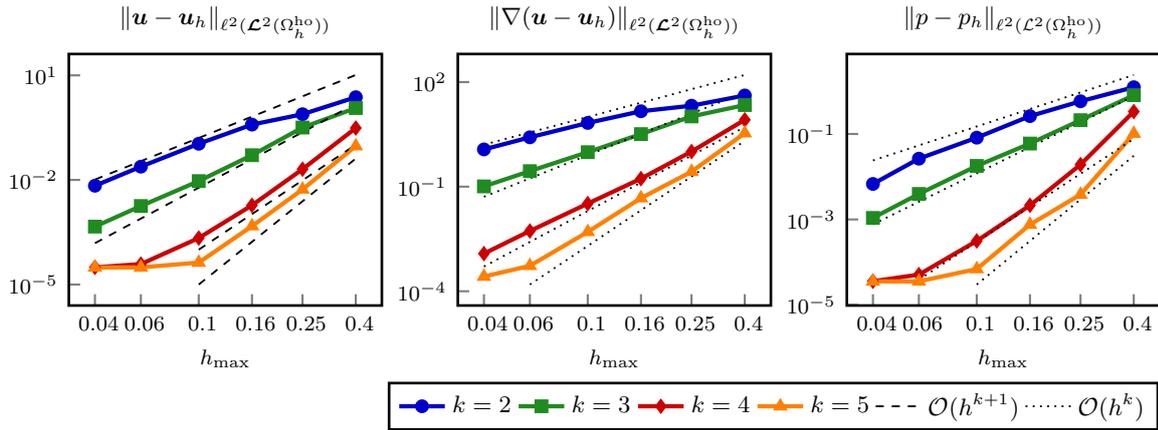

Figure 3.8: Time-step convergence of the velocity- and pressure-errors with BDF1 and BDF2 time-stepping for the Oseen problem with the isoparametric CutFEM method for the spatial discretisation on a mesh with $h_{\max} = 0.1$, using $\mathtt{TH}_5^{\mathrm{iso}}$ elements.

Time-Step Convergence   We consider a background mesh with $h_{\max} = 0.1$ together with $\mathtt{TH}_5^{\mathrm{iso}}$ elements. We then take both the BDF1 and the BDF2 discretisations of the time-derivative and consider time-steps between $\Delta t = 1/10$ and $\Delta t = 1/2000$. For the BDF2 discretisation, we use a single step of the BDF1 discretisation, to initialise the BDF2 scheme. The resulting errors of the velocity in the $\ell^2(\boldsymbol{\mathcal{L}}^2(\Omega_h^{\mathrm{ho}}))$- and $\ell^2(\boldsymbol{\mathcal{H}}^1(\Omega_h^{\mathrm{ho}}))$-norm and the pressure in the $\ell^2(\mathcal{L}^2(\Omega_h^{\mathrm{ho}}))$-norm can be seen in Figure 3.8.

Looking at the convergence rates, we see optimal order convergence with respect to the time-step both for schemes, i.e., first-order convergence for BDF1 and second-order convergence for BDF2 in all considered norms, until step spatial error begins to dominate in the BDF2 case. As a result of this, we note that the scheme has not realised the expected loss of half an order of convergence, and the error estimate derived in Theorem 3.30 is therefore most likely not sharp, as noted upon in Remark 3.31.

Mesh Convergence   In order to see the spatial error, rather than the temporal error, we use the BDF2 discretisation for the time-derivative and take the time-step $\Delta t = 1/1000$. We then consider a series of meshes of the background domain, constructed using mesh parameters $0.04 \leq h_{\max} \leq 0.4$. On these meshes, we consider $\mathtt{TH}_k^{\mathrm{iso}}$ elements with $k = 2, 3, 4, 5$. The resulting space-time errors of the velocity and pressure can be seen in Figure 3.9.

Figure 3.9: Mesh convergence of the velocity- and pressure-errors for the isoparametric CutFEM method for the Oseen problem using $\texttt{TH}_k^{\text{iso}}$ elements. The time discretisation is done with the BDF2 method and the time-step $\Delta t = 1/1000$.

We again observe optimal order convergence in all three norms considered, while the spatial error appears to dominate. For higher-order polynomials on the finish meshes, we observe that the temporal error dominates. For $k = 2$ we note that we do not have optimal convergence on the coarsest meshes, which we take to be pre-asymptotic behaviour. Furthermore, it appears that for $k = 3$, that we have a slight loss in the convergence in the velocity error, although the errors are still better than in the case for $k = 2$.

3.3.3 Summary

In the above numerical computations, we have seen that the dependence of the condition number of the stiffness matrix on the ghost-penalty parameter is very similar to that in the Stokes case. This illustrates that the bound in Lemma 3.26 is independent of the interface-mesh cut position. Concerning the asymptotic convergence, we have seen that the isoparametric CutFEM discretisation converges optimally with the time-step, even though our error analysis did not show this. With respect to the mesh size, the convergence is not as monotone as in the Stokes case, but we retain optimal order convergence. This suggests, that Theorem 3.30 is not sharp, and that a better estimate of the term $\|\frac{1}{\Delta t}(\mathbf{e}_h^n - \mathbf{e}_h^{n-1})\|_{\Omega_h^{\text{ho}}}$ should also be possible in the unfitted case with the addition velocity-pressure coupling introduced by the geometry consistency error.

3.4 A Fluid-Rigid Body Interaction Benchmark

In [Wah+19b], we published a benchmark for fluid-rigid body interactions for purely rotational motion of the solid in both two and three spatial dimensions. The basic idea here was to take the well-established *flow around a cylinder* benchmark by SCHÄFER and TUREK [ST96] and add additional degrees of freedom by allowing the rotation of the solid around its centre of mass. In 3d, we modified the spatial setup slightly to consider a sphere rather than a cylinder to avoid solid-solid contact between the wall and the rotating solid. As a result of this design, the fluid-solid partition does not change over time, so that standard fluid solvers can access this benchmark, as the interaction is essentially an implicit defined non-homogeneous Dirichlet boundary condition on the fluid-solid interface.

We shall use this benchmark to investigate the approximation properties of the isoparametric Cut-FEM method we have derived and analysed in section 3.2 and section 3.3 when applied to a fluid-rigid body interaction problem.
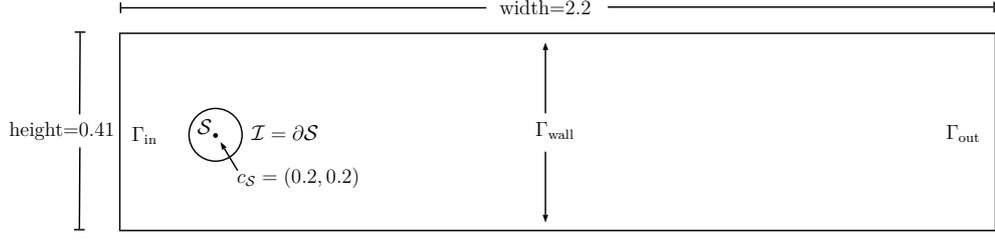
Figure 3.10: Domain for the two-dimensional freely-rotating sphere benchmark.

3.4.1 Benchmark Description

We give brief overview of the definition of the benchmark we shall consider here. In particular, we shall focus on the stationary *Rot2d-1* and the unsteady and fixed time scenario *Rot2d-3*.

Domain and Equations   The domain of interest is given by $\Omega = (0, 2.2) \times (0, 0.41)$. The solid domain is given by $\mathcal{S} = \{\boldsymbol{x} \in \Omega \mid \|\boldsymbol{x} - (0.2, 0.2)^T\|_2 < 0.05\}$ and the fluid domain is then $\mathcal{F} = \Omega \setminus \overline{\mathcal{S}}$. An illustration of this can be seen in Figure 3.10.

In this domain, we then consider the fluid-rigid body system (2.5). As the rigid body can only rotate around its centre of mass, and since there are no external forces acting on the fluid, the system reduces to

$$\rho_\mathcal{F}(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}) - \operatorname{div}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = 0 \qquad \text{in } \mathcal{F} \times (0, t^{\text{end}})$$
$$\operatorname{div}(\boldsymbol{u}) = 0 \qquad \text{in } \mathcal{F} \times (0, t^{\text{end}})$$

for the fluid and

$$I_\mathcal{S}\partial_t \omega = T \qquad \text{in } (0, t^{\text{end}})$$

for the solid. The boundary conditions for the fluid are

$$\boldsymbol{u} = \omega(t) r \vec{\boldsymbol{t}} \qquad \text{on } \mathcal{I} \times (0, t^{\text{end}})$$
$$\boldsymbol{u} = \boldsymbol{u}_{\text{in}} \qquad \text{on } \Gamma_{\text{in}} \times (0, t^{\text{end}})$$
$$\boldsymbol{u} = 0 \qquad \text{on } \Gamma_{\text{wall}} \times (0, t^{\text{end}})$$
$$\rho_\mathcal{F}\nu(\nabla\boldsymbol{u})\boldsymbol{n} - Ip\boldsymbol{n} = 0 \qquad \text{on } \Gamma_{\text{out}} \times (0, t^{\text{end}}),$$

i.e, continuity of the velocity at the fluid solid interface, a Dirichlet inflow condition given by

$$\boldsymbol{u}_{\text{in}}(\boldsymbol{x}, t) = \frac{4U(t)\boldsymbol{x}_2(0.41 - \boldsymbol{x}_2)}{0.41^2} \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \tag{3.43}$$

no-slip conditions on the walls of the channel, and a parallel outflow "do-nothing" condition on the outflow boundary, see [Ric17, Section 2.4.2]. In these equations, $\omega(t)$ is the scalar angular velocity (in a counterclockwise sense), $\vec{\boldsymbol{t}} = (-\boldsymbol{n}_2, \boldsymbol{n}_1)^T$ is the tangential vector (counter-clockwise) on the solid and r is the radius of the circular rigid body. In two spatial dimensions, the moment of inertia $I_\mathcal{S}$ of inertia of the circular solid body and the torque exerted by the fluid onto the solid are given by

$$I_\mathcal{S} = \rho_\mathcal{S}\int_\mathcal{S} \|\boldsymbol{x} - c_\mathcal{S}\|_2^2 \,\mathrm{d}\boldsymbol{x} = \rho_\mathcal{S}\frac{\pi}{2}r^4 \qquad \text{and} \qquad T = \int_\mathcal{I} r\vec{\boldsymbol{t}} \cdot \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} \,\mathrm{d}s.$$

C_D	C_L	Δp	ω^*
5.57955881	0.004714193	0.1175202	0.0012629346

Table 3.1: Reference values for the benchmark problem "Rot2d-1" taken from [Wah+19b].

$C_{D,\max}$	$t_{D,\max}$	$C_{L,\max}$	$t_{D,\max}$	$C_{T,\max}$	$t_{D,\max}$	$\Delta p(8)$	$\omega^*_{\max}$	$t_{\omega,\max}$
2.95089	3.3962	0.4656	5.6939	0.009688	5.8430	-0.11187	0.003347	5.9563

Table 3.2: Reference values for the benchmark problem "Rot2d-3" taken from [Wah+19b].

Benchmark Quantities Amongst the quantities of interest in the benchmarks are the forces acting on the solid body. The forces acting on the solid in the Euclidean coordinate axes are given by $\boldsymbol{F} = \int_{\mathcal{I}} \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}\,\mathrm{d}s$. As benchmark quantities, we are then interested in the dimensionless drag-lift and torque coefficients, given by

$$C_D = \frac{2}{u_m^2 \rho_{\mathcal{F}} L} \boldsymbol{F}_1, \qquad C_L = \frac{2}{u_m^2 \rho_{\mathcal{F}} L} \boldsymbol{F}_2 \qquad \text{and} \qquad C_T = \frac{4}{u_m^2 \rho_{\mathcal{F}} L^2} T, \tag{3.44}$$

where u_m is a characteristic velocity and L is a characteristic length, which in this case is taken as $L = 0.1$, the diameter of the obstacle. The characteristic velocity is however problem dependent. As further quantities of interest, as defined in [Wah+19b], we have the dimensionless angular velocity of the solid body and the pressure difference between the front and back of the solid. These are given by

$$\omega^* = \frac{\omega L}{2u_m} \qquad \text{and} \qquad \Delta p := p((0.15, 0.2)) - p((0.25, 0.2)).$$

Problem Set-Up

We consider two set-ups from [Wah+19b]. The stationary problem *Rot2d-1* and the time-dependent problem over a fixed time-interval denoted as *Rot2d-3*. In both cases, the fluid's density is $\rho_{\mathcal{F}} = 1$ and its viscosity is $\nu = 0.001$. The solid density is $\rho_{\mathcal{S}} = 10$.

Stationary Problem: Rot2d-1 The inflow speed is $U = 0.3$, which gives a mean inflow speed of $u_m = 2 \cdot 0.3/3 = 0.2$, which we take as the characteristic speed of the problem. The resulting flow has the Reynolds number $Re = 20$ and is stationary. The reference values for this problem are given in Table 3.1.

Time-Dependent Problem Over a Fixed Time-Interval: Rot2d-3 The inflow speed is given by $U(t) = 1.5\sin(\pi t/8)$ over the time-interval $[0, 8]$. The characteristic speed is the maximal mean inflow speed, which is $u_m = 1.0$. As a result, the maximal Reynolds number of the flow is $Re = 100$. The quantities of interest here are the maximal force coefficients and the maximal dimensionless angular velocity and the respective times at which these are realised. The reference values for this problem are given in Table 3.2.

3.4.2 Unfitted Discretisation

The discretisation of the Navier-Stokes equation is based on the isoparametric CutFEM method we developed and analysed for the Oseen problem in section 3.3. However, in the non-linear case, we change the formulation of the convective term from the convective formulation used before to the *skew-symmetric* form

$$c_h(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) := \frac{1}{2}\Big[\int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{u} \cdot \nabla \boldsymbol{v} \cdot \boldsymbol{w}\,\mathrm{d}\boldsymbol{x} - \int_{\Omega_h^{\mathrm{ho}}} \boldsymbol{u} \cdot \nabla \boldsymbol{w} \cdot \boldsymbol{v}\,\mathrm{d}\boldsymbol{x} \Big].$$

This form is equivalent to the convective form and has the advantage of vanishing for $\boldsymbol{w} = \boldsymbol{v}$, for all $\boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\Omega_h^{\text{ho}})$, see [Joh16, section 6.1.2]. Therefore, we do not need any additional boundary contributions for the stability of the form. We compute the forces by evaluating the boundary integrals which define them, since approaches such as the Babuška-Miller trick [BM84] do not work in the present unfitted setting.

Stationary Problem

As described in [Wah+19b, section 3.1.1], the stationary problem can be solved directly by applying the Newton method to solve the problem

$$j(\omega) := T(\boldsymbol{u}(\omega)) \overset{!}{=} 0,$$

where $\boldsymbol{u}(\omega)$ is the stationary Navier-Stokes solution with angular velocity ω and the solid interface. To approximate the Jacobean of the functional $j(\cdot)$, we then use a finite difference approximation

$$j'(\omega)(\delta\omega) \approx \frac{T(\boldsymbol{u}(\omega + \delta)) - T(\boldsymbol{u}(\omega))}{\delta}.$$

To solve the necessary stationary Navier-Stokes problems to approximate the torque Jacobean, we use the following isoparametric CutFEM discretisation.

Problem P5. Find $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h^{\text{iso}} \times Q_h^{\text{iso}}$ such that for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^{\text{iso}} \times Q_h^{\text{iso}}$ it holds

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + \nu i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) + c_h(\boldsymbol{u}_h, \boldsymbol{u}_h, \boldsymbol{v}_h) + b_h(p_h, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h) + \frac{1}{\nu} j_h(p_h, q_h) = 0.$$

The resulting non-linear system is then solved by a Newton iteration.

Time-Dependent Problem

To decouple the Navier-Stokes fluid equation from the Newton-Euler solid equation, we use a semi-implicit time-marching scheme for the fluid equation and an explicit scheme for the solid equation.

We discretise the Navier-Stokes equations in time by using the second-order implicit-explicit (IMEX) SBDF2 scheme [ARW95]. Here, the less stiff convective term is treated completely explicitly such that we need to solve the same linear system in each time-step. This comes at the cost of a CFL time-step restriction of order established experimentally as $\Delta t \lesssim {}^h\!/\!{k^{3/2}}$ [Wah18]. In conjunction with this scheme, we use the explicit part, i.e., a second-order extrapolation, to advance the solid ODE problem in time. Due to the necessarily small time-step for the fluid problem, the explicit treatment of the solid problem does not result in any stability issues. The discrete scheme then reads in every time-step

$$\frac{1}{\Delta t} \frac{3}{2} m_h(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + (a_h + \nu i_h)(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + b_h(p_h, \boldsymbol{v}_h) + b_h(q_h, \boldsymbol{u}_h) + \frac{1}{\nu} j_h(p_h, q_h)$$
$$= \frac{1}{\Delta t} \left(2 m_h(\boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h) - \frac{1}{2} m_h(\boldsymbol{u}_h^{n-2}, \boldsymbol{v}_h) \right) - 2 c_h(\boldsymbol{u}_h^{n-1}, \boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h) + c_h(\boldsymbol{u}_h^{n-2}, \boldsymbol{u}_h^{n-2}, \boldsymbol{v}_h)$$

for the fluid problem, and

$$\frac{1}{\Delta t} \frac{3}{2} \omega^n = \frac{1}{\Delta t} \left(2\omega^{n-1} - \frac{1}{2} \omega^{n-2} \right) + I_{\mathcal{S}}^{-1} (2 T(\boldsymbol{u}_h^{n-1}) - T(\boldsymbol{u}_h^{n-2}))$$

for the solid. To initialise the system we use the corresponding first-order IMEX SBDF1 scheme, which is sufficient to realise second-order time-step convergence of the method [Wah18].

Discretisation		Results (Convergence Rate)				
k	$h_{\max}$	C_D	C_L	C_T	Δp	ω^*
2	0.08	5.58379747	-0.007768625	$8.4 \cdot 10^{-15}$	0.1162891	0.0043527636
2	0.04	5.57685210 (0.6)	0.010236941 (1.2)	$2.6 \cdot 10^{-11}$	0.1173639 (3.0)	-0.0002105188 (1.1)
2	0.02	5.57903095 (2.4)	0.006133097 (2.0)	$1.6 \cdot 10^{-15}$	0.1173366 (-0.2)	0.0010200395 (2.6)
2	0.01	5.57929045 (1.0)	0.004716603 (9.2)	$7.9 \cdot 10^{-12}$	0.1174920 (2.7)	0.0012697050 (5.2)
Reference		5.57955881	0.004714193	0.0	0.1175202	0.0012629346

Table 3.3: Results for the *Rot-2d1* problem, computed using the isoparametric CutFEM method with $\mathtt{TH}_2^{\mathrm{iso}}$ elements over a series of meshes.

Discretisation			Results (Convergence rate)				
k	$h_{\max}$	n_{ref}	C_D	C_L	C_T	Δp	ω^*
2	0.04	0	5.57685210	0.010236941	$2.6 \cdot 10^{-11}$	0.1173639	-0.0002105188
2	0.04	1	5.57944586 (4.6)	0.005804510 (2.3)	$3.9 \cdot 10^{-13}$	0.1173899 (0.3)	0.0008787381 (1.9)
2	0.04	2	5.57904214 (-2.2)	0.004199299 (1.1)	$1.2 \cdot 10^{-12}$	0.1174909 (2.2)	0.0013357075 (2.4)
2	0.04	3	5.57954550 (5.3)	0.004666192 (3.4)	$1.1 \cdot 10^{-12}$	0.1175193 (5.1)	0.0012671340 (4.1)
Reference			5.57955881	0.004714193	0.0	0.1175202	0.0012629346

Table 3.4: Results for the *Rot-2d1* problem, computed using the isoparametric CutFEM method with $\mathtt{TH}_2^{\mathrm{iso}}$ elements on a mesh with $h_{\max} = 0.04$ after multiple bisections of cut elements.

3.4.3 Numerical Results

To compute both the *Rot-2d1* and *Rot-2d3* problems, we begin with a mesh constructed with $h = h_{\max}/8$ in the front of the channel where $\boldsymbol{x}_1 \leq 0.7$. In conjunction with our previous results, we take the ghost-penalty parameter to be $\gamma_{gp} = 0.01$, and the Nitsche penalty parameter is $\sigma = 40k^2$.

To compute the convergence behaviour of the method in the functional values of interest, we assume a form for the error of $err(h) = |err| \sim ch^r$. For two errors, we can then compute the convergence rate as $r = \big(\log(err(h)) - \log(err(h/2))\big)/\log(2)$. To capture the true maxima of time-dependent functional values, we interpolate a fourth-order spline to the data and then take the maximum of the spline as the reported value.

Rot-2d1 We begin by considering the case $k = 2$ over a series of meshes. The results of these computations can be seen in Table 3.3. Here, we see that while all quantities of interest converge to the reverence values, the lift and angular-velocity functions have the wrong sign on the first two meshes. This can, in part, be attributed to the fact that these quantities are small in absolute value, and therefore difficult to approximate.

As the quantities of interest only begin to be accurate on the finer meshes, we consider the mesh with $h_{\max} = 0.04$ and bisect the elements cut by the interface. This can be expected to lead to improved results, as highly accurate force values on the interface require highly resolved meshes at the interface, even in the fitted setting [Wah+19b]. We consider up to three levels of mesh refinement of cut elements. The results can be seen in Table 3.4. Since we observe mesh convergence in the functionals, even though only cut elements are iteratively refined, we see that most of the error in the previous global mesh convergence study was in the approximation of the solution at the interface.

To consider the effect of the isoparametric mapping, we consider the same sequence of iterative bisection of elements cut by the solid interface but without the parametric mapping. This corresponds to the choice of $\Theta_h = \mathrm{id}$, as a result of which the geometry approximation error is of order h^2. These results can be seen in Table 3.5. Here we see that while we still observe convergence of the quantities of interest, the

Discretisation			Results (Convergence rate)				
k	$h_{\max}$	n_{ref}	C_D	C_L	C_T	Δp	ω^*
2	0.04	0	5.56379624	0.018412451	$-3.1 \cdot 10^{-12}$	0.1172884	-0.0017422572
2	0.04	1	5.56966202 (0.7)	0.007492112 (2.3)	$1.1 \cdot 10^{-13}$	0.1173697 (0.6)	0.0005777901 (2.1)
2	0.04	2	5.57417707 (0.9)	0.000668006 (-0.5)	$3.5 \cdot 10^{-12}$	0.1174811 (1.9)	0.0019070896 (0.1)
2	0.04	3	5.57738077 (1.3)	0.004808180 (5.4)	$9.1 \cdot 10^{-13}$	0.1175166 (3.5)	0.0012118283 (3.7)
Reference			5.57955881	0.004714193	0.0	0.1175202	0.0012629346

Table 3.5: Results for the *Rot-2d1* problem, computed using the CutFEM method with TH$_2$ elements on a mesh with $h_{\max} = 0.04$ after multiple bisections of cut elements.

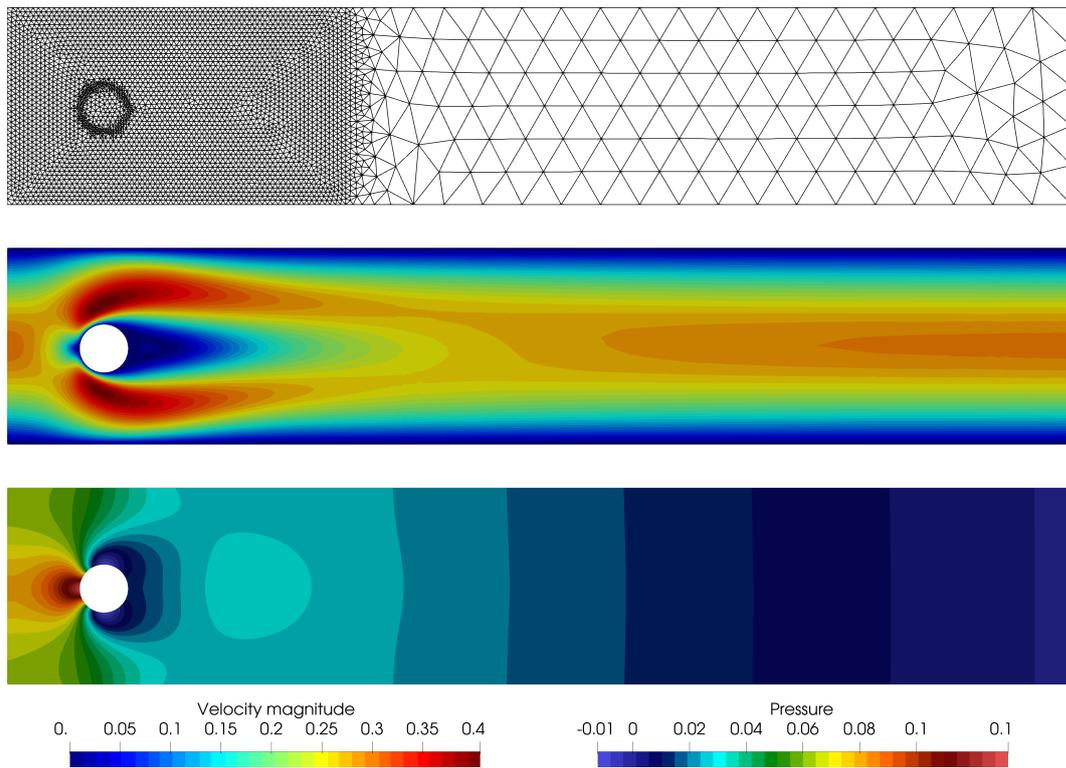| Discretisation			Results (Convergence rate)						
k	$h_{\max}$	n_{ref}	C_D		C_L		C_T	Δp	ω^*
3	0.08	3	5.57957643		0.004701057		$1.0 \cdot 10^{-12}$	0.1175097	0.0012636435
3	0.04	3	5.57955715	(3.4)	0.004711938	(2.5)	$1.0 \cdot 10^{-12}$	0.1175187 (2.8)	0.0012631640 (1.6)
3	0.02	3	5.57955873	(4.3)	0.004713665	(2.1)	$9.9 \cdot 10^{-13}$	0.1175197 (1.7)	0.0012630115 (1.6)
4	0.08	3	5.57955948		0.004714107		$1.0 \cdot 10^{-12}$	0.1175111	0.0012629250
4	0.04	3	5.57955878	(4.5)	0.004714148	(0.9)	$1.1 \cdot 10^{-12}$	0.1175187 (2.6)	0.0012629416 (0.5)
4	0.02	3	5.57955881	(3.7)	0.004714193	(6.7)	$7.6 \cdot 10^{-13}$	0.1175197 (1.6)	0.0012629357 (2.7)
5	0.08	3	5.57955879		0.004714195		$7.1 \cdot 10^{-13}$	0.1175111	0.0012629341
5	0.04	3	5.57955881	(4.9)	0.004714194	(1.6)	$2.1 \cdot 10^{-12}$	0.1175187 (2.6)	0.0012629357 (-1.2)
5	0.02	3	5.57955881	(-0.8)	0.004714195	(-1.3)	$2.0 \cdot 10^{-12}$	0.1175197 (1.7)	0.0012629356 (0.2)
5	0.08	0	5.57955289		0.004695598		$-3.1 \cdot 10^{-12}$	0.1171413	0.0012674014
5	0.08	1	5.57954771	(-0.9)	0.004688067	(-0.5)	$-3.5 \cdot 10^{-12}$	0.1172984 (0.8)	0.0012642677 (1.7)
5	0.08	2	5.57955808	(3.9)	0.004714269	(8.4)	$-2.0 \cdot 10^{-13}$	0.1174582 (1.8)	0.0012628177 (3.5)
5	0.08	3	5.57955879	(5.1)	0.004714195	(5.1)	$7.1 \cdot 10^{-13}$	0.1175111 (2.8)	0.0012629341 (7.9)
Reference			5.57955881		0.004714193		0.0	0.1175202	0.0012629346

Table 3.6: Results for the *Rot-2d1* problem, computed using the isoparametric CutFEM method with TH$_k^{\text{iso}}$ elements.

convergence rate is significantly reduced compared to the isoparametric results in Table 3.4. Indeed, even the drag coefficient, which is the easiest functional to compute, is only accurate up to three significant figures.

Finally, we consider higher-order spaces, that is, $k \geq 3$. As we have seen above, the mesh resolution at the interface is paramount. Therefore, we consider meshes with three levels of iterative bisections of elements cut by the solid interface. The results from these computations can be seen in Table 3.6. Here we see that the higher order of the finite element spaces lead to better convergence rates and significantly more accurate functional values. In fact, for $k = 5$, we see little convergence, as the resulting values with $h_{\max} = 0.08$ and three levels of interface element refinement are already extremely close to the reference values. However, we still see that even for $k = 5$, the very fine mesh at the interface is necessary, as the resulting values with fewer refinements at the interface are significantly worse.

Rot2d-3   As for the stationary case, we consider a number of different mesh configurations. Due to the stability needs of the IMEX time-stepping scheme, the necessary time-steps are very small. As a result, we shall keep close to the stability limit as we can expect the spatial error to dominate here. All the results of our numerical computations for the time-dependent problem Rot2d-3 are reported in Table 3.7.

Looking at the results for $k = 2$ under uniform mesh refinement, we see that while the drag lift and torque peaks are identified at the correct time, the maximum of the angular velocity is realised a the wrong time. Taking the mesh with $h_{\max} = 0.04$ and considering iterative refinement of cut elements, we see that one

Figure 3.11: Velocity and pressure solution for the *Rot2d-1* problem, computed using $\texttt{TH}_3^{\text{iso}}$ elements on a mesh with $h_{\text{max}} = 0.08$, $h = h_{\text{max}}/8$ in the front of the channel and one bisection of cut elements.

Discretisation				Results (Convergence rate)								
k	$h_{\max}$	n_{ref}	Δt	$C_{D,\max}$	$t_{D,\max}$	$C_{L,\max}$	$t_{L,\max}$	$C_{T,\max}$	$t_{T,\max}$	Δp	$\omega^*_{L,\max}$	$t_{\omega,\max}$
2	0.08	0	$1/1200$	2.9552047	3.93682	0.4788927	5.69388	0.0111239	5.84923	−0.1116168	0.0020097	7.37870
2	0.04	0	$1/2200$	2.9496983 (1.9)	3.93608	0.4657261 (6.7)	5.69465	0.0097047 (6.4)	5.84474	−0.1118929 (3.5)	0.0025218 (0.7)	7.36837
2	0.02	0	$1/4400$	2.9506717 (2.4)	3.93624	0.4667328 (−3.2)	5.69394	0.0097136 (−0.6)	5.84376	−0.1118075 (−1.4)	0.0029698 (1.1)	5.95713
2	0.01	0	$1/8800$	2.9506867 (0.1)	3.93620	0.4655993 (10.7)	5.69390	0.0096875 (5.8)	5.84315	−0.1118769 (3.2)	0.0033600 (4.9)	5.95641
2	0.04	0	$1/2500$	2.9496982	3.93608	0.4656920	5.69469	0.0097041	5.84474	−0.1118957	0.0025219	7.36855
2	0.04	1	$1/2500$	2.9509427 (4.5)	3.93634	0.4636937 (−4.4)	5.69404	0.0096925 (1.8)	5.84406	−0.1119765 (−2.1)	0.0037934 (0.9)	5.95673
2	0.04	2	$1/2500$	2.9505026 (−2.9)	3.93627	0.4642352 (0.5)	5.69376	0.0096544 (−2.9)	5.84361	−0.1118376 (1.7)	0.0037990 (−0.0)	5.95621
2	0.04	3	$1/2500$	2.9509784 (2.1)	3.93623	0.4650435 (1.3)	5.69377	0.0096694 (0.9)	5.84334	−0.1118738 (3.1)	0.0035001 (1.6)	5.95611
3	0.08	3	$1/2000$	2.9509056	3.93622	0.4656604	5.69389	0.0096888	5.84316	−0.1118570	0.0033580	5.95601
3	0.04	3	$1/4000$	2.9508918 (3.1)	3.93619	0.4656855 (−0.5)	5.69385	0.0096889 (−0.2)	5.84314	−0.1118711 (3.5)	0.0033506 (1.6)	5.95626
3	0.02	3	$1/8000$	2.9508932 (−0.8)	3.93619	0.4656606 (0.5)	5.69389	0.0096887 (0.4)	5.84306	−0.1118743 (−1.9)	0.0033485 (1.2)	5.95631
4	0.08	3	$1/3100$	2.9508943	3.93620	0.4657102	5.69380	0.0096895	5.84316	−0.1118667	0.0033492	5.95621
4	0.04	3	$1/6200$	2.9508935 (0.3)	3.93619	0.4656678 (0.7)	5.69388	0.0096888 (0.9)	5.84309	−0.1118731 (0.1)	0.0033485 (0.5)	5.95630
4	0.02	3	$1/9300$	2.9508936 (−0.0)	3.93619	0.4656577 (0.2)	5.69390	0.0096886 (0.3)	5.84305	−0.1118746 (−0.6)	0.0033484 (0.1)	5.95632
5	0.08	3	$1/4400$	2.9508935	3.93619	0.4656846	5.69386	0.0096891	5.84313	−0.1118700	0.0033485	5.95627
5	0.04	3	$1/8800$	2.9508936 (−0.0)	3.93619	0.4656587 (0.5)	5.69390	0.0096887 (0.7)	5.84305	−0.1118739 (−7.8)	0.0033484 (0.0)	5.95631
5	0.02	3	$1/13200$	2.9508936 (0.0)	3.93619	0.4656535 (0.1)	5.69390	0.0096886 (0.2)	5.84302	−0.1118750 (−0.3)	0.0033484 (0.0)	5.95632
Reference				2.95089	3.3962	0.4656	5.6939	0.009688	5.8430	−0.11187	0.003347	5.9563

Table 3.7: Results for the *Rot-2d3* problem, computed using the isoparametric CutFEM method with higher-order elements.

level of refinement is sufficient to identify the correct peak of the angular velocity. This illustrates that a sufficiently well-resolved interface is vital here to capture the dynamics accurately.

Looking at the mesh convergence results for the higher-order methods with three levels of refinement of cut elements, we see that the resulting quantities of interest are immediately very close to the known reference values such that further convergence cannot be observed. However, we also note that the reference values, as taken from [Wah+19b], only give between four and six significant figures.

3.4.4 Summary

The numerical simulations illustrate the importance of a highly resolved interface and the necessity of the parametric mapping in order to accurately compute the benchmark quantities of interest, even in this relatively simple fluid-rigid body interaction example. It is not surprising that a smaller mesh parameter at the interface is necessary to compute the reference values accurately since this is also the case for fitted FEM approaches to obtain accurate values [Wah+19b].

While we did not observe monotone convergence in $h_{\max}$ for the functional values, we saw that higher-order methods generally resulted in higher-order convergence and significantly more accurate values compared to the lowest order Taylor-Hood element. Furthermore, we have seen that once the interface is sufficiently resolved, high-order unfitted methods compute the relevant reference values as accurately as the fitted finite element methods in [Wah+19b].

An Unfitted Eulerian Time-Stepping Method for the Transient Stokes Problem on Moving Domains

Structure of Chapter The main unfitted Finite Element method using an Eulerian time-stepping approach for time-dependent flow problems on moving domains is developed. Using the transient Stokes equations on a moving domain with prescribed motion, the method is developed and analysed numerically. The numerical analysis is complemented with numerical experiments to illustrate the convergence behaviour of the scheme. This follows our work from [WRL21] and builds on [LO19; BFM19]. Further numerical studies, using the Navier-Stokes equations for the fluid model, are presented to compare the developed methods against other unfitted Finite Element schemes for moving domain flow problems from the literature. Examples including topology changes in the fluid geometry are presented.

Contents of Chapter

4.1 Problem Description

To move towards fluid-rigid body interaction, including motion of the solid, such that the fluid-solid partition changes over time, we consider the time-dependent Stokes problem on a moving domain (2.14). This consists of two significant simplifications in comparison to the full fluid-rigid body interaction system (2.5). The first simplification is to only consider the fluid part of the problem, with the motion of the

fluid domain $\mathcal{F}(t)$ given externally. The second is to remove the convection term, thereby assuming a flow with small Reynolds number. Since the equations are posed entirely in the fluid domain, we shall refer to the spatial domain as $\Omega(t)$.

Following [BFM19], we assume that the domain motion is described by a smooth mapping

$$\boldsymbol{\Psi}(t)\colon \Omega_0 \to \Omega(T) \qquad \text{with} \qquad \boldsymbol{\Psi} \in \boldsymbol{\mathcal{L}}^\infty(0, t^{\text{end}}; \boldsymbol{\mathcal{W}}^{k+1,\infty}(\Omega_0)) \cap \boldsymbol{\mathcal{W}}^{1,\infty}(0, t^{\text{end}}; \boldsymbol{\mathcal{W}}^{k,\infty}(\Omega_0)),$$

a smooth reference domain Ω_0 and $\text{div}(\partial_t \boldsymbol{\Psi}(\boldsymbol{\Psi}^{-1})) = 0$. Due to our context of fluid-rigid body dynamics, we take the continuity of the velocity at the fluid boundary as the Dirichlet boundary condition, i.e.,

$$\boldsymbol{u} = \partial_t \boldsymbol{\Psi}(\boldsymbol{\Psi}^{-1}) \quad \text{on } \Gamma(t) \coloneqq \partial\Omega(t).$$

Defining $\tilde{\boldsymbol{u}} = \boldsymbol{u} - \partial_t \boldsymbol{\Psi}(\boldsymbol{\Psi}^{-1})$, we can therefore homogenise the boundary condition in the system (2.14) to obtain

$$\partial_t \tilde{\boldsymbol{u}} - \nu\Delta\tilde{\boldsymbol{u}} + \nabla p = \tilde{\boldsymbol{f}} \qquad\qquad \text{in } \mathcal{Q} \qquad\qquad (4.1\text{a})$$
$$\text{div}(\tilde{\boldsymbol{u}}) = 0 \qquad\qquad \text{in } \mathcal{Q} \qquad\qquad (4.1\text{b})$$
$$\boldsymbol{u}(\cdot, t) = 0 \qquad\qquad \text{on } \Gamma(t), \qquad\qquad (4.1\text{c})$$

where $\tilde{\boldsymbol{f}} = \boldsymbol{f} + \partial_t^2 \boldsymbol{\Psi}(\boldsymbol{\Psi}^{-1}) - \nu\Delta(\partial_t \boldsymbol{\Psi}(\boldsymbol{\Psi}^{-1}))$. To keep the notation simple, we shall drop the tildes and identify $(\boldsymbol{u}, p)$ as the solution to the homogenised problem (4.1).

Now, let $I = [0, t^{\text{end}}]$, $\boldsymbol{V}(t) = \boldsymbol{\mathcal{H}}_0^1(\Omega(t))$, $\boldsymbol{L}(t) = \boldsymbol{\mathcal{L}}^2(\Omega(t))$ and $Q(t) \coloneqq \mathcal{L}_0^2(\Omega(t))$. We then define the spaces

$$\boldsymbol{V}_I \coloneqq \{\boldsymbol{u} \in \boldsymbol{\mathcal{L}}^2(0, t^{\text{end}}; \boldsymbol{V}(t)) \mid \partial_t \boldsymbol{u} \in \boldsymbol{\mathcal{L}}^2(0, t^{\text{end}}; \boldsymbol{L}(t))\} \qquad \text{and} \qquad Q_I \coloneqq \mathcal{L}^2(0, t^{\text{end}}; Q(t)).$$

We assume that for the data, we have $\boldsymbol{f} \in \boldsymbol{\mathcal{L}}^\infty(0, t^{\text{end}}; \boldsymbol{L}(t))$ and $\boldsymbol{u}_0 \in \boldsymbol{\mathcal{H}}^1(\Omega(0))$. Then the weak formulation of (4.1) is as follows: Find $(\boldsymbol{u}, p) \in \boldsymbol{V}_I \times Q_I$ such that

$$(\partial_t \boldsymbol{u}, \boldsymbol{v})_{\Omega(t)} + \nu(\boldsymbol{u}, \boldsymbol{v})_{\Omega(t)} - (p, \nabla\cdot\boldsymbol{v})_{\Omega(t)} - (q, \nabla\cdot\boldsymbol{u})_{\Omega(t)} = (\boldsymbol{f}, \boldsymbol{v})_{\Omega(t)} \qquad\qquad (4.2)$$

for all $(\boldsymbol{v}, q) \in \boldsymbol{V}_I \times Q_I$ almost everywhere in $t \in I$, with $\boldsymbol{u}(\cdot, 0) = \boldsymbol{u}(0)$ almost everywhere in $\Omega(0)$.

4.2 Discretisation

Most methods for solving problems on moving domains, such as (4.22), are based on an at least partially Lagrangian or purely Eulerian description of the moving boundary. The most common approach is the Arbitrary Eulerian-Langrangian (ALE) method [HLZ81; DGH82; Don+04]. The ALE approach works by mapping the domain and equations into a reference configuration in which the problem is solved, usually using fitted finite element approaches with standard time-stepping schemes [Ric17] or using space-time Galerkin formulations [Beh01; Beh08; KVV06; Neu13]. A drawback of the ALE approach is that the deformation with respect to the reference configuration must be small, and topology changes must not occur. For large deformations, the ALE method can be extended by considering regular mesh updates or space-time meshing. Purely Eulerian approaches avoid these issues, and we will follow this approach here. In a purely Eulerian framework, a fixed background mesh is considered to define a set of potential unknowns. The domain geometry is then described separately; see section 3.1 for an overview of such unfitted finite element approaches.

When considering time-dependent problems on moving domains in a purely Eulerian framework, the main challenge is the approximation of the time-derivate $\partial_t \boldsymbol{u} \approx 1/\Delta t(\boldsymbol{u}^n - \boldsymbol{u}^{n-1})$ because $\boldsymbol{u}^n$ and $\boldsymbol{u}^{n-1}$ are defined on different domains such that the difference of the two is not well-defined. An approach that has been proven to work in the unfitted Eulerian setting is a class of space-time Galerkin formulations. This has been considered, for example, in the context of scalar bulk problems [LR13; Leh15; Zah18;

Pre18; Hei20], problems on moving surfaces [GOR15; ORX14; OR14] or surface bulk problems [HLZ16]. Recently, initial steps have also been taken towards two-phase flow problems [VR18; AB21]. However, the fundamental problem with space-time Galerkin approaches is that $(d+1)$-dimensional FEM problems have to be solved, which comes at a high computational cost. An alternative to space-time methods is to use modified quadrature rules to reduce the space-time formulation into a classical time-stepping scheme [FR17b]. However, this approach then requires the expensive computation of projections between discrete function spaces at times t^n and t^{n-1}.

Our approach, as presented in [WRL21], follows a different approach to the aforementioned space-time methods. The idea of our approach is to use an implicit extension, based on ghost-penalty stabilisation, such that the approximation of the time-derivative $1/\Delta t(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})$ is again well-defined. This idea to enable the Eulerian time-stepping for problems on a moving surface by extended finite element functions was introduced by OLSHANSKII and XU [OX17]. Extending the discrete functions through ghost-penalty stabilisation was then analysed by LEHRENFELD and OLSHANSKII [LO19] for a scalar convection-diffusion problem and extended to the transient Stokes problem on a moving domain (2.14) by BURMAN, FREI and MASSING [BFM19].

In the context of flow problem, the idea of using ghost-penalties to define an extension of the velocity in order to apply standard time-discretisations for moving domain problems can be traced back to SCHOTT [Sch17]. However, the extension in [Sch17] is computed in a separate sub-step, and the extension is limited to a vertex patch of active elements such that the time-step must obey a Courant–Friedrichs–Lewy (CFL) type condition $\Delta t \lesssim h$. Furthermore, no numerical analysis of the method was provided in [Sch17] for this approach.

Our work can therefore be seen as an extension of that in [BFM19]. In contrast to the equal-order spatial discretisation with pressure stabilisation in [BFM19], we shall consider the use of unfitted Taylor-Hood elements, as in chapter 3, so that the stabilisation is only necessary in the vicinity of the domain boundary. Furthermore, our analysis focuses on the geometry approximation error inherent in CutFEM and the robustness in the viscosity ν. Both of these aspects were not covered in [BFM19].

4.2.1 Temporal Semi-Discretisation

Let us consider the uniform time-step $\Delta t = t^{\mathrm{end}}/N$ for some $N \in \mathbb{N}$. As before, we denote $t^n = n\Delta t$. We further introduce $I^n = [t^{n-1}, t^n)$, $\Omega^n = \Omega(t^n)$ and $\Gamma^n = \Gamma(t^n) = \partial\Omega(t^n)$. The δ-*neighbourhood* of $\Omega(t)$ is defined as
$$\mathcal{O}_\delta(\Omega(t)) := \{\boldsymbol{x} \in \mathbb{R}^d \mid \mathrm{dist}(\boldsymbol{x}, \Omega(t)) \leq \delta\}.$$
For the method, we then require δ to be sufficiently large for the domain Ω^n to be in the δ-neighbourhood of the previous domain, i.e.,
$$\Omega^n \subset \mathcal{O}_\delta(\Omega^{n-1}), \qquad \text{for } n = 1, \dots, N.$$
This is achieved by the choice
$$\delta = c_\delta \boldsymbol{w}_\infty^n \Delta t,$$
where $\boldsymbol{w}_\infty^n$ is the maximal normal speed of the domain interface and $c_\delta > 0$. The time-discretisation is then based on a combination of the *method of lines* approach, in combination with an extension operator for Sobolev functions to the δ-neighbourhood, which ensures that the solution defined on the domain at the previous time-step is well-defined on the current domain. In the context of unfitted finite element methods, an extension of the smooth solution is standard, as we have seen in subsection 3.2.2. In this context of moving domains, we have to define the extension slightly differently, which we discuss below; see also [LO19; WRL21].

Variational Formulation

We take the weak formulation (4.2) as the basis of our discretisation. We then discretise the time deriv-ative with the implicit Euler (or BDF1) method combined with the extension operator. The variational

formulation of the temporally semi-discrete problem then reads as follows:

Problem P6. For $n = 1, \ldots, N$, given $\boldsymbol{u}^{n-1} \in \boldsymbol{\mathcal{H}}_0^1(\Omega^{n-1})$ and $\boldsymbol{f}^n \in \boldsymbol{\mathcal{H}}^{-1}(\Omega^n)$, find $(\boldsymbol{u}^n, p^n) \in \boldsymbol{V}^n \times Q^n :=$ $\boldsymbol{\mathcal{H}}^1(\Omega^n) \times \mathcal{L}_0^2(\Omega^n)$ such that

$$\frac{1}{\Delta t}(\boldsymbol{u}^n, \boldsymbol{v})_{\Omega^n} + a^n(\boldsymbol{u}^n, \boldsymbol{v}) + b^n(p^n, \boldsymbol{v}) + b^n(q, \boldsymbol{u}^n) = \langle \boldsymbol{f}^n, \boldsymbol{v} \rangle_{(\boldsymbol{V}^n)', \boldsymbol{V}^n} + \frac{1}{\Delta t}(\mathcal{E}\boldsymbol{u}^{n-1}, \boldsymbol{v})_{\Omega^n} \qquad (4.3)$$

for all $(\boldsymbol{v}, q) \in \boldsymbol{V}^n \times Q^n$.

The bilinear forms therein are given by

$$a^n(\boldsymbol{u}, \boldsymbol{v}) = \nu \int_{\Omega^n} \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} \qquad \text{and} \qquad b^n(q, \boldsymbol{v}) = -\int_{\Omega^n} q \nabla \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} \qquad (4.4)$$

for the diffusion term and the velocity-pressure coupling, respectively. The extension operator $\mathcal{E} :$ $\boldsymbol{\mathcal{H}}^1(\Omega^{n-1}) \to \boldsymbol{\mathcal{H}}^1(\mathcal{O}_\delta(\Omega^{n-1}))$ allows us to make sense of the "initial value" $\boldsymbol{u}^{n-1} \in \boldsymbol{\mathcal{H}}_0^1(\Omega^{n-1})$ in $\Omega^n \subset \mathcal{O}_\delta(\Omega^{n-1})$. We discuss the details of this extension below.

Extension Operator

For the extension operator, we require the following family of space-time anisotropic spaces

$$\boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^m(\Omega(t))) := \left\{ \boldsymbol{v} \in \boldsymbol{\mathcal{L}}^2(\mathcal{Q}) \;\middle|\; \begin{array}{l} \boldsymbol{v}(\cdot, t) \in \boldsymbol{\mathcal{H}}^m(\Omega(t)) \text{ for a.e. } t \in (0, T) \\ \text{and } \operatorname*{ess\,sup}_{t \in (0,T)} \|\boldsymbol{v}(\cdot, t)\|_{\boldsymbol{\mathcal{H}}^k(\Omega(t))} < \infty \end{array} \right\},$$

for $m = 0, \ldots, k + 1$. We then denote $\partial_t \boldsymbol{v} = \boldsymbol{v}_t$ as the weak partial derivative with respect to the time variable, if this exists as an element of the space-time space $\boldsymbol{\mathcal{L}}^2(\mathcal{Q})$.

We now assume the existence of a spatial extension operator

$$\mathcal{E} : \boldsymbol{\mathcal{L}}^2(\Omega(t)) \to \boldsymbol{\mathcal{L}}^2(\mathcal{O}_\delta(\Omega(t))),$$

which fulfils the following properties:

Assumption A1. Let $\boldsymbol{v} \in \boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))) \cap \boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})$. There exist positive constants c_{A1a}, c_{A1b} and c_{A1c} that are uniform in t such that

$$\|\mathcal{E}\boldsymbol{v}\|_{\boldsymbol{\mathcal{H}}^k(\mathcal{O}_\delta(\Omega(t)))} \le c_{A1a} \|\boldsymbol{v}\|_{\boldsymbol{\mathcal{H}}^k(\Omega(t))} \qquad (4.5a)$$

$$\|\nabla(\mathcal{E}\boldsymbol{v})\|_{\mathcal{O}_\delta(\Omega(t))} \le c_{A1b} \|\nabla\boldsymbol{v}\|_{\Omega(t)} \qquad (4.5b)$$

$$\|\mathcal{E}\boldsymbol{v}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{O}_\delta(\mathcal{Q}))} \le c_{A1c} \|\boldsymbol{v}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \qquad (4.5c)$$

holds. Furthermore, if for $\boldsymbol{v} \in \boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^{k+1}(\Omega(t)))$ it holds for the weak partial time-derivative that $\boldsymbol{v}_t \in \boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^k(\Omega(t)))$, then

$$\|(\mathcal{E}\boldsymbol{v})_t\|_{\boldsymbol{\mathcal{H}}^k(\mathcal{O}_\delta(\Omega(t)))} \le c_{A1d} \left[\|\boldsymbol{v}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|\boldsymbol{v}_t\|_{\boldsymbol{\mathcal{H}}^k(\Omega(t))} \right], \qquad (4.6)$$

where the constant $c_{A1d} > 0$ again only depends on the motion of the spatial domain.

Such an extension operator can be constructed explicitly from the classical linear and continuous *universal* extension operator for Sobolev spaces (see, e.g., [Ste70, Section VI.3]), when the motion of the domain is described by a diffeomorphism $\boldsymbol{\Psi}(t) \colon \Omega_0 \to \Omega(t)$ for each $t \in [0, T]$ from the reference domain Ω_0 that is smooth in time. See [LO19] for details thereof.

Assuming sufficient regularity of the domain Ω^n, e.g., connected with Lipschitz-boundary, the well-posedness of (4.3) is given for every time-step by the standard theory of the Stokes-Brinkman problem or equivalently the Oseen problem with vanishing convective velocity, see for example [Joh16, Theorem 5.7].

Stability

We have the following stability estimates, for the semi-discrete scheme in (4.3).

Lemma 4.1. *Let $\{\boldsymbol{u}^n\}_{n=1}^N$ be the velocity solution to (4.3) with initial data $\boldsymbol{u}^0 \in \mathcal{H}^1(\Omega^0)$. Then for $n = 1, \ldots, N$, it holds that*

$$\|\boldsymbol{u}^n\|_{\Omega^n}^2 + \Delta t \sum_{i=1}^n \frac{\nu}{2} \|\nabla \boldsymbol{u}^i\|_{\Omega^i} \leq \exp(c_{L4.1} t^n) \left[\|\boldsymbol{u}^0\|_{\Omega^0}^2 + \frac{\nu \Delta t}{2} \|\nabla \boldsymbol{u}^0\|_{\Omega^0}^2 + \frac{c_P^2 \Delta t}{\nu} \sum_{i=1}^n \|\boldsymbol{f}^i\|_{\Omega^i}^2 \right],$$

with a positive constant $c_{L4.1}$ independent of the time-step size and the number of time-steps n and the constant c_P from the Poincaré-inequality.

Proof. The proof is analogue to that of [LO19, Lemma 3.6], using the test function $2\Delta t(\boldsymbol{u}^n, -p^n) \in \boldsymbol{V}^n \times Q^n$ to remove the pressure from the equation. $\qquad\square$

Lemma 4.2. *Let $\{p^n\}_{n=1}^N$ be the pressure solution to (4.3). Then it holds*

$$\|p^n\|_{\Omega^n} \leq c_{L4.2} \left[c_P \|\boldsymbol{f}^n\|_{\Omega^n} + \frac{1}{\Delta t} \|\boldsymbol{u}^n - \mathcal{E}\boldsymbol{u}^{n-1}\|_{\mathcal{H}^{-1}(\Omega^n)} + \nu \|\nabla \boldsymbol{u}^n\|_{\Omega^n} \right]$$

with a constant $c_{L4.2} > 0$ independent of the time-step size Δt and the time-step n.

Proof. The claim follows by the inf-sup stability of the form $b^n(\cdot, \cdot)$. We refer to [WRL21] for the details of the proof. $\qquad\square$

Remark 4.3. It is not immediately clear that Lemma 4.2 gives us the stability of the pressure, independent of the time-step size Δt. The explanation of why we do not expect a scaling of $\|p^n\|_{\Omega^n}$ with Δt^{-1} is based on an argument with the relation for the discretisation error. Using integration by parts, we have that the exact solution $(\boldsymbol{u}(t^n), p(t^n))$ fulfils for all $(\boldsymbol{v}, q) \in \boldsymbol{V}^n \times Q^n$ the variational problem

$$(\partial_t \boldsymbol{u}(t^n), \boldsymbol{v})_{\Omega^n} + a^n(\boldsymbol{u}(t^n), \boldsymbol{v}) + b^n(p(t^n), \boldsymbol{v}) + b^n(q, \boldsymbol{u}(t^n)) = \langle \boldsymbol{f}^n, \boldsymbol{v} \rangle_{(\boldsymbol{V}^n)', \boldsymbol{V}^n}.$$

In line with our previous notation, we define the temporal discretisation errors as $\mathbb{E}^n := \boldsymbol{u}(t^n) - \boldsymbol{u}^n$ and $\mathbb{D}^k := p(t^n) - p^n$. For $n = 1, .., N$ it then holds that

$$\left(\frac{\mathbb{E}^n - \mathcal{E}\mathbb{E}^{n-1}}{\Delta t}, \boldsymbol{v} \right)_{\Omega^n} + a^n(\mathbb{E}^n, \boldsymbol{v}) + b^n(\mathbb{D}^n, \boldsymbol{v}) + b^n(q, \mathbb{E}^n) = \left(\frac{\boldsymbol{u}(t_n) - \mathcal{E}\boldsymbol{u}(t^{n-1})}{\Delta t} - \partial_t \boldsymbol{u}(t_n), \boldsymbol{v} \right)_{\Omega^n}$$

for all $(\boldsymbol{v}, q) \in \boldsymbol{V}^n \times Q^n$. Now, assuming sufficient regularity, i.e. $\boldsymbol{u} \in \mathcal{W}^{2,\infty}(\mathcal{Q})$, we obtain the bound $c_{R4.3a} \Delta t \|\boldsymbol{u}\|_{\mathcal{W}^{2,\infty}(\mathcal{Q})} \|\boldsymbol{v}\|_{\Omega^n}$ for the right-hand side with a constant $c_{R4.3a}$ independent of n, $\boldsymbol{u}$ and Δt. Here, we also made use of (4.5c). As the left side of the equation is the same as in (4.3), we can apply Lemma 4.1 (using $\mathbb{E}^0 = 0$) to obtain the bound

$$\|\mathbb{E}^n\|_{\Omega^n} \leq c_{R4.3b} \exp(c_{R4.3c} t^n) \Delta t \|\boldsymbol{u}\|_{\mathcal{W}^{2,\infty}(\mathcal{Q})}.$$

Hence, the triangle inequality yields

$$\frac{1}{\Delta t} \|\boldsymbol{u}^n - \mathcal{E}\boldsymbol{u}^{n-1}\|_{\mathcal{H}^{-1}(\Omega^n)} \leq \frac{1}{\Delta t} \|\boldsymbol{u}(t^n) - \mathcal{E}\boldsymbol{u}(t^{n-1})\|_{\mathcal{H}^{-1}(\Omega^n)} + \frac{1}{\Delta t} \|\mathbb{E}^n\|_{\mathcal{H}^{-1}(\Omega^n)} + \frac{1}{\Delta t} \|\mathcal{E}\mathbb{E}^{n-1}\|_{\mathcal{H}^{-1}(\Omega^n)}$$

$$\leq \|\partial_t \boldsymbol{u}(t^n)\|_{\mathcal{H}^{-1}(\Omega^n)} + c_{R4.3d} \|\boldsymbol{u}\|_{\mathcal{W}^{2,\infty}(\mathcal{Q})},$$

and hence a bound on the norm of p^n that is independent on Δt^{-1}. $\qquad\blacktriangle$

4.2.2 Fully Discrete Method

The spatial discretisation is based on the CutFEM approach discussed in chapter 3. We take a simplicial, shape-regular and quasi-uniform mesh of the $\widetilde{\mathscr{T}_h}$ background domain $\widetilde{\Omega} \subset \mathbb{R}^d$, for which we assume that $\Omega(t) \subset \widetilde{\Omega}$ for all $t \in [0, t^{\text{end}}]$. The domain is defined via a level set function ϕ, c.f. subsection 3.1.1. The discrete level set domain at time t^n is then denoted as Ω_h^n and its boundary by Γ_h^n. Arbitrary cuts between the mesh and the level set function are stabilised via ghost-penalties, c.f. subsection 3.1.2. The only difference is in the geometry approximation. We will give the details of this below in subsection 4.3.4.

To realise the necessary extension in the discrete setting, we shall make use of the ghost-penalty stabilisation, as this gives us control over the discrete solution on the part of the active mesh where we do not have an equation. In order for the discrete solution from the previous time-step to be well-defined on the domain at the next time-step, we need to extend the discrete domain Ω_h^n in every time-step by

$$\delta_h = c_{\delta_h} \boldsymbol{w}_\infty^{\boldsymbol{n}} \Delta t,$$

with $c_{\delta_h} > 1$ such that Ω_h^{n+1} is a subset of the extended discrete domain, but also sufficiently small so that $\mathcal{O}_{\delta_h}(\Omega_h^n) \subset \mathcal{O}_\delta(\Omega^n)$. We then collect all the elements with a contribution to this active domain in the *active mesh* at time-step n as

$$\mathscr{T}_{h,\delta_h}^n := \{T \in \widetilde{\mathscr{T}_h} \mid \exists \boldsymbol{x} \in T \text{ such that } \operatorname{dist}(\boldsymbol{x}, \Omega_h^n) \leq \delta_h\} \subset \widetilde{\mathscr{T}_h}.$$

The *active domain* at time n is then

$$\mathcal{O}_{\delta_h,\mathscr{T}}^n := \{\boldsymbol{x} \in T \mid T \in \mathscr{T}_{h,\delta_h}^n\}.$$

Similarly, we then define the set of elements which have some contribution to the physical domain at time n as the *cut mesh*, denoted by $\mathscr{T}_h^n := \mathscr{T}_{h,0}^n$, and the *cut domain* $\mathcal{O}_\mathscr{T}^n := \mathcal{O}_{0,\mathscr{T}}^n$ as the domain cut elements.

As we need to define the ghost-penalty operator on the δ_h-strip of elements, we require additional sets of facets in each time-step. To this end, we define the set of elements in the boundary strip as

$$\mathscr{T}_{h,\mathcal{S}^\pm}^n := \{T \in \widetilde{\mathscr{T}_h} \mid \exists \boldsymbol{x} \in T \text{ with } \operatorname{dist}(\boldsymbol{x}, \Gamma_h^n) \leq \delta_h\},$$

and the set of interior facets of this strip as

$$\mathscr{F}_{h,\delta_h}^n := \{F = \overline{T}_1 \cap \overline{T}_2 \mid T_1 \in \mathscr{T}_{h,\delta_h}^n, T_2 \in \mathscr{T}_{h,\mathcal{S}^\pm}^n \text{ with } T_1 \neq T_2 \text{ and } \operatorname{meas}_{d-1}(F) > 0\}.$$

Since the pressure does not need to be extended in the discrete formulation, we collect the set of boundary elements at time t^n as

$$\mathscr{T}_{h,\Gamma_h^n} := \{T \in \widetilde{\mathscr{T}_h} \mid T \cap \Gamma_h^n \neq \emptyset\},$$

and the set of interior facets of these elements as

$$\mathscr{F}_h^n := \{F = \overline{T}_1 \cap \overline{T}_2 \mid T_1 \in \mathscr{T}_h^n, T_2 \in \mathscr{T}_{h,\Gamma_h^n} \text{ with } T_1 \neq T_2 \text{ and } \operatorname{meas}_{d-1}(F) > 0\}.$$

A sketch of these elements and facets can be seen in Figure 4.1. As noted above and discussed in detail below, we do not use the parametric mapping in the analysis of the method. Consequently, we consider the standard family of Taylor-Hood elements for $k \geq 2$

$$\boldsymbol{V}_h^n := \{\boldsymbol{v}_h \in \boldsymbol{C}(\mathcal{O}_{\delta_h,\mathscr{T}}^n) \mid \boldsymbol{v}_h|_T \in [\mathbb{P}^k(T)]^d \text{ for all } T \in \mathscr{T}_{h,\delta_h}^n\}$$

and

$$Q_h^n := \{q_h \in C(\mathcal{O}_\mathscr{T}^n) \mid q_h|_T \in \mathbb{P}^k(T) \text{ for all } T \in \mathscr{T}_h^n\}.$$

Note that while the velocity space is defined on the entire active mesh, the pressure space is only defined on the cut mesh.
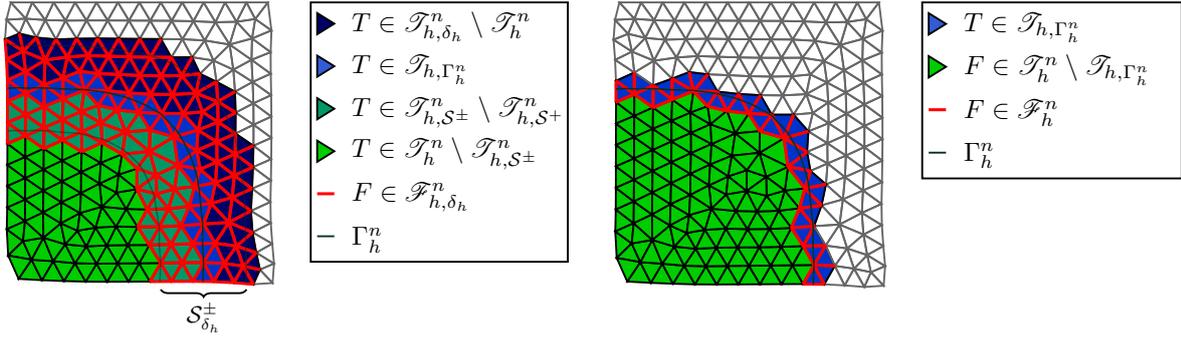
Figure 4.1: Left: Active elements and facets for the velocity. Right: Active elements and facets for the pressure.

Variational Formulation

With the newly introduced mesh notation, we can now formulate the fully discretised scheme for the time-dependent Stokes problem on moving domains.

Problem P7. Given an appropriate initial condition $\boldsymbol{u}_h^0 \in \boldsymbol{V}_h^0$, for $n = 1, \ldots, N$, find $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h^n \times Q_h^n$ such that

$$\int_{\Omega_h^n} \frac{\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + a_h^n(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + b_h^n(p_h^n, \boldsymbol{v}_h) + b(q_h, \boldsymbol{u}_h^n) + s_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) = f_h^n(\boldsymbol{v}_h) \quad (4.7)$$

for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^n \times Q_h^n$.

As in section 3.2, we consider a symmetric Nitsche formulation for the bilinear forms, now defined on the domains at time n. For the diffusion bilinear form, we then have

$$a_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) := \nu \int_{\Omega_h^n} \nabla \boldsymbol{u}_h : \nabla \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + N_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h)$$

$$N_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) := N_{h,c}^n(\boldsymbol{u}_h, \boldsymbol{v}_h) + N_{h,c}^n(\boldsymbol{v}_h, \boldsymbol{u}_h) + N_{h,s}^n(\boldsymbol{u}_h, \boldsymbol{v}_h), \quad \text{with}$$

$$N_{h,c}^n(\boldsymbol{u}_h, \boldsymbol{v}_h) := -\int_{\Gamma_h^n} \partial_{\boldsymbol{n}} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}s \quad \text{and} \quad N_{h,s}^n(\boldsymbol{u}_h, \boldsymbol{v}_h) := \frac{\sigma}{h} \int_{\Gamma_h^n} \boldsymbol{u}_h \cdot \boldsymbol{v}_h \, \mathrm{d}s,$$

where $\sigma > 0$ is the penalty parameter. For the pressure-coupling bilinear form, we have

$$b_h^n(q_h, \boldsymbol{v}_h) := \int_{\Omega_h^n} q_h \nabla \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + N_{h,c}^n(q_h, \boldsymbol{v}_h), \quad \text{where} \quad N_{h,c}^n(q_h, \boldsymbol{v}_h) := \int_{\Gamma_h^n} (q_h \boldsymbol{n}) \cdot \boldsymbol{v}_h \, \mathrm{d}s.$$

To realise the necessary ghost-penalty extension into the exterior of the domain, we define the stabilising bilinear form as

$$s_h^n((\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) := \gamma_{u,gp,1} \nu i_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) + \gamma_{u,gp,2} \frac{1}{\nu} i_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) + \gamma_{p,gp} \frac{1}{\nu} j_h^n(p_h, q_h),$$

with stabilisation parameters $\gamma_{u,gp,1}, \gamma_{u,gp,2}, \gamma_{p,gp} > 0$. We will discuss a suitable choice of these parameters in Remark 4.9 below. As before, the velocity and pressure ghost-penalties stabilise arbitrary cuts, while the pressure stabilisation is necessary for the unfitted inf-sup property. However, the velocity ghost-penalty operator now also defines the implicit extension into the δ_h-strip for the method to control the solution in the entire active domain. While we continue to use the direct ghost-penalties, the two forms are now defined as

$$i_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) := \sum_{F \in \mathscr{F}_{h,\delta_h}^n} \frac{1}{h^2} \int_{\omega_F} [\![\boldsymbol{u}_h]\!] \cdot [\![\boldsymbol{v}_h]\!] \, \mathrm{d}\boldsymbol{x} \quad \text{and} \quad j_h^n(p_h, q_h) := \sum_{F \in \mathscr{F}_h^n} \int_{\omega_F} [\![p_h]\!] [\![q_h]\!] \, \mathrm{d}\boldsymbol{x},$$

with the facet-patch jump operator $[\![\cdot]\!] = [\![\cdot]\!]_{\omega_F}$ defined in (3.2). Finally, the right-hand side term is

$$f_h^n(\boldsymbol{v}_h) := \int_{\Omega_h^n} \boldsymbol{f}^n \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x},$$

with the standard modification for non-homogeneous Dirichlet data $\boldsymbol{u}_D$ of

$$f_h^n(\boldsymbol{v}_h, q_h) := \int_{\Omega_h^n} \boldsymbol{f}^n \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + N_{h,c}^n(\boldsymbol{v}_h, \boldsymbol{u}_D) + N_{h,s}(\boldsymbol{u}_D, \boldsymbol{v}_h) + N_{h,c}(q_h, \boldsymbol{u}_D).$$

With an abuse on notation, we take $\boldsymbol{f}^n$ as a (smooth) extension of the forcing term, so that it is well-defined on Ω_h^n.

4.3 Analysis of the Method

We perform the numerical analysis of the fully discrete method (4.7) in this section. To this end, we introduce some additional notation and cover some of the general results needed for the numerical analysis. We then show that the problem is well-posed, prove a consistency estimate due to the time-discretisation and the geometry approximation and then prove an error estimate for the energy error.

4.3.1 Preliminaries

Further Notation

We define the set of *extension strip elements* as

$$\mathscr{T}_{h,\mathcal{S}^+}^n := \{T \in \widetilde{\mathscr{T}_h} \mid \exists \boldsymbol{x} \in \widetilde{\Omega} \setminus \Omega_h^n \text{ with } \mathrm{dist}(\boldsymbol{x}, \Gamma_h^n) \le \delta_h\}.$$

We also define the *sharp strips*

$$\mathcal{S}_{\delta_h}^{\pm}(\Omega_h^n) := \{\boldsymbol{x} \in \widetilde{\Omega} \mid \mathrm{dist}(\boldsymbol{x}, \Gamma_h^n) \le \delta_h\} \qquad \text{and} \qquad \mathcal{S}_{\delta_h}^{+}(\Omega_h^n) := \{\boldsymbol{x} \in \widetilde{\Omega} \setminus \Omega_h^n \mid \mathrm{dist}(\boldsymbol{x}, \Gamma_h^n) \le \delta_h\}.$$

With this, we then define the *discrete extended domain* $\mathcal{O}_{\delta_h}(\Omega_h^n) := \mathcal{S}^{\pm}(\Omega_h^n) \cup \Omega_h^n$. In the following analysis, we then require that δ is sufficiently large, such that

$$\mathcal{O}_{\delta_h,\mathscr{T}}^n \subset \mathcal{O}_{\delta}(\Omega^n) \qquad \text{and} \qquad \Omega_h^n \subset \mathcal{O}_{\delta}(\Omega(t)), \text{ for } t \in I^n = [t^{n-1}, t^n)$$

for $n = 1, \dots, N$.

In the analysis, we consider the following mesh-dependent norms. For the velocity, we take

$$|\!|\!|\boldsymbol{v}|\!|\!|_n^2 := \|\nabla \boldsymbol{v}\|_{\Omega_h^n}^2 + \|h^{-1/2}\boldsymbol{v}\|_{\Gamma_h^n}^2 + \|h^{1/2}\partial_{\boldsymbol{n}}\boldsymbol{v}\|_{\Gamma_h^n}^2$$

$$|\!|\!|\boldsymbol{v}|\!|\!|_{*,n}^2 := \|\nabla \boldsymbol{v}\|_{\mathcal{O}_{\delta_h,\mathscr{T}}^n}^2 + \|h^{-1/2}\boldsymbol{v}\|_{\Gamma_h^n}^2, \qquad \text{and} \qquad \|\boldsymbol{v}\|_{-1,n} := \sup_{\boldsymbol{w} \in \boldsymbol{V}_h^n} \frac{(\boldsymbol{v}, \boldsymbol{w})_{\Omega_h^n}}{|\!|\!|\boldsymbol{w}|\!|\!|_{*,n}}.$$

For the pressure, we introduce the norms

$$|\!|\!|q|\!|\!|_n^2 := \|q\|_{\Omega_h^n}^2 + \|h^{1/2}q\|_{\Gamma_h^n}^2, \qquad \text{and} \qquad |\!|\!|q|\!|\!|_{*,n} := \|q\|_{\mathcal{O}_{\mathscr{T}}^n},$$

while for the product space we take

$$|\!|\!|(\boldsymbol{v}, q)|\!|\!|_{*,n}^2 := |\!|\!|\boldsymbol{v}|\!|\!|_{*,n}^2 + |\!|\!|q|\!|\!|_{*,n}^2.$$

Note that the $|\!|\!|\cdot|\!|\!|_n$-norms are defined on the physical domain and add control on the normal-derivative of the velocity and the trace of the pressure at the boundary. These norms arise naturally to bound the

bilinear forms $a_h^n(\boldsymbol{u}, \boldsymbol{v})$ and $b_h^n(q, \boldsymbol{v})$. The second type of norms, the $\|\|\cdot\|\|_{*,n}$-norms, are defined on the entire active domain and therefore represent proper norms for discrete functions in our finite element spaces. As in (3.12), we find using the trace inequality (3.9c) for the velocity norm and the trace and inverse estimates (3.10), and (3.9a) for the pressure norm, that

$$\|\|\boldsymbol{v}_h\|\|_n \lesssim \|\|\boldsymbol{v}_h\|\|_{*,n} \qquad \text{and} \qquad \|\|q_h\|\|_n \lesssim \|\|q_h\|\|_{*,n} \tag{4.8}$$

for all $\boldsymbol{v}_h \in \boldsymbol{V}_h^n$ and $q_h \in Q_h^n$. Similar to (3.11), we also have a Poincaré type inequality

$$\|\boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^n} \leq c_{P,h} \|\|\boldsymbol{v}_h\|\|_{*,n} \tag{4.9}$$

for all $\boldsymbol{v}_h \in \boldsymbol{V}_h^n$.

Interpolation Properties

We recall the interpolation properties of the Scott-Zhang interpolant in Lemma 3.9. Furthermore, we have for the triple-norm the following interpolation result.

Lemma 4.4. *Let $\boldsymbol{v} \in \boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)$ and $q \in \mathcal{H}^k(\Omega^n)$, $m \geq 1$. Then the following estimates hold*

$$\|\|\mathcal{E}\boldsymbol{v} - \mathcal{I}_h^* \mathcal{E}\boldsymbol{v}\|\|_n \lesssim h^k \|\boldsymbol{v}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)} \tag{4.10a}$$

$$\|\|\mathcal{E}q - \mathcal{I}_h^* \mathcal{E}q\|\|_n \lesssim h^k \|q\|_{\mathcal{H}^k(\Omega^n)}. \tag{4.10b}$$

Proof. We refer to [WRL21, Lemma 5.2] for details of the proof. $\qquad\square$

Ghost-Penalty Properties

Assumption A2. We assume that for every strip element $T \in \mathscr{T}_{h,\mathcal{S}^+}^n$ there exists an uncut element $T' \in \mathscr{T}_h^n \setminus \mathscr{T}_{h,\mathcal{S}^+}^n$, which can be reached by a path which crosses a bounded number of facets $F \in \mathscr{F}_{h,\delta_h}^n$. We assume that the number of facets which have to be crossed to reach T' from T is bounded by a constant $L \lesssim (1 + \frac{\delta_h}{h})$ and that every uncut element $T' \in \mathscr{T}_h^n \setminus \mathscr{T}_{h,\mathcal{S}^+}^n$ is the end of at most M such paths, with M bounded independent of Δt and h. In other words, each uncut elements "supports" at most M strip elements.

See [LO19, Remark 5.4] for a justification as to why the above assumption is reasonable if the mesh resolves the domain boundary sufficiently well.

Lemma 4.5 (*Ghost-penalty mechanism*). *For all $\boldsymbol{v}_h \in \boldsymbol{V}_h^n$ and $q_h \in Q_h^n$ it holds that*

$$\|\nabla \boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^n}^2 \simeq \|\nabla \boldsymbol{v}_h\|_{\Omega_h^n}^2 + L \cdot i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h) \tag{4.11a}$$

$$\|\boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^n}^2 \simeq \|\boldsymbol{v}_h\|_{\Omega_h^n}^2 + h^2 L \cdot i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h) \tag{4.11b}$$

$$\|q_h\|_{\mathcal{O}_{\mathscr{T}}^n}^2 \simeq \|q_h\|_{\Omega_h^n}^2 + j_h^n(q_h, q_h). \tag{4.11c}$$

Proof. The equivalence (4.11c) is simply a reformulation of (3.4). The proof of (4.11a) and (4.11b) is a repetition of the arguments in the proof of Lemma 3.3, taking into account the number of elements that need to be crossed to reach an interior element, as described in Assumption A2. $\qquad\square$

4.3.2 Well-Posedness

Lemma 4.6 (*Continuity*). *For the diffusion bilinear form, we have for all $\boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\mathcal{O}_\delta(\Omega^n))$ that it holds that*

$$a^n(\boldsymbol{u}, \boldsymbol{v}) \lesssim \nu \|\|\boldsymbol{u}\|\|_n \|\|\boldsymbol{v}\|\|_n, \tag{4.12a}$$

and for all $\boldsymbol{u}_h, \boldsymbol{v}_h \in \boldsymbol{V}_h^n$ *it holds that*

$$a_h^n(\boldsymbol{u}_h, \boldsymbol{v}_h) + \nu L i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h) \lesssim \nu \|\|\boldsymbol{u}_h\|\|_{*,n} \|\|\boldsymbol{v}_h\|\|_{*,n}. \tag{4.12b}$$

Furthermore, for the velocity-pressure coupling bilinear form, we have for all $q \in \mathcal{L}^2(\mathcal{O}_\delta(\Omega^n))$ *and* $\boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\mathcal{O}_\delta(\Omega^n))$ *that*

$$b_h^n(q, \boldsymbol{v}) \lesssim \|\|q\|\|_n \|\|\boldsymbol{v}\|\|_n,$$

and for all $q_h \in Q_h^n$ *and* $\boldsymbol{v}_h \in \boldsymbol{V}_h^n$ *that*

$$b_h^n(q_h, \boldsymbol{v}_h) \lesssim \|\|q_h\|\|_{*,n} \|\|\boldsymbol{v}_h\|\|_{*,n}.$$

Proof. The proof is entirely analogue to that of Lemma 3.12 and taking Lemma 4.5 into account. □

Lemma 4.7 (*Coercivity*). *There exists a constant* $c_{L4.7} > 0$, *independent of* h *and the mesh-interface cut position, such that for sufficiently large* $\sigma > 0$ *there holds*

$$a_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h) + \nu L i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h) \geq \nu c_{L4.7} \|\|\boldsymbol{v}_h\|\|_{*,n}^2$$

for all $\boldsymbol{v}_h \in \boldsymbol{V}_h^n$.

Proof. For $\nu = 1$ and $L = 1$, the proof can be seen in [BH12, Lemma 6] or [BH14, Lemma 4.2]. The proof for general $\nu > 0$ and $L > 1$ is again a repetition of the identical arguments and taking the ghost-penalty strip-width into account. □

Corollary 4.8 (*Bad inf-sup*). *Let* $\mathcal{O}_{int}$ *denote the interior, uncut domain. Then for all* $q_h \in Q_h^n$ *with* $q_h|_{\mathcal{O}_{int}} \in \mathcal{L}_0^2(\mathcal{O}_{int})$, *it holds that*

$$\beta \|q_h\|_{\Omega_h^n} \leq \sup_{\boldsymbol{v}_h \in \boldsymbol{V}_h^n} \frac{b_h(q_h, \boldsymbol{v}_h)}{\|\|\boldsymbol{v}_h\|\|_{*,n}} + j_h^n(q_h, q_h)^{1/2}. \tag{4.13}$$

The constant $\beta > 0$ *is independent of* h *and* q_h.

Proof. For each $n = 1, \ldots, N$ this statement is given by Lemma 3.15, taking the parametric mapping as the identity. □

Remark 4.9 (Choice of ghost-penalty parameters). Lemma 4.6 and Lemma 4.7 show that the velocity ghost-penalty parameter should scale with the width of the extension strip L. This is necessary, in order for an exterior unphysical but active element to obtain the necessary support from an uncut interior element for which we have to cross at most L elements to reach it, c.f. Assumption A1. As the first part of the ghost-penalties term $s_h^n(\cdot, \cdot)$ is related to the stabilisation of the viscosity bilinear form $a_h^n(\cdot, \cdot)$, it only has a scaling with ν. We require the same mechanism also for the implicit extension of functions. As we will see in the analysis below, this requires the velocity ghost-penalties scaled with $1/\nu$.

The pressure ghost-penalty operator in Corollary 4.8 does not need to be scaled with L, as we require these ghost-penalties only to stabilise the velocity-pressure coupling at the interface and not for an extension of the pressure field into a δ-neighbourhood.

For simplicity of the analysis, we choose a common ghost-penalty stabilisation parameter γ_{gp}, and set $\gamma_{u,gp,1} = \gamma_{u,gp,2} = L\gamma_{gp}$ and $\gamma_{p,gp} = \gamma_{gp}$. ▲

We now collect all fully implicit non-ghost-penalty terms and all the explicit linear terms in the bilinear and linear forms

$$A_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) := \frac{1}{\Delta t}(\boldsymbol{u}_h^n, \boldsymbol{v}_h)_{\Omega_h^n} + a_h^n(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + b_h^n(p_h^n, \boldsymbol{v}_h) + b_h^n(q_h, \boldsymbol{u}_h^n)$$

$$F_h^n(\boldsymbol{v}_h) := \frac{1}{\Delta t}(\boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h)_{\Omega_h^n} + f_h^n(\boldsymbol{v}_h),$$

respectively. We can then rewrite problem (4.7) as: For $n = 1, \ldots, N$, find $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h^n \times Q_h^n$ such that

$$A_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) + s_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) = F_h^n(\boldsymbol{v}_h^n) \tag{4.14}$$

for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^n \times Q_h^n$.

Theorem 4.10 (*Well-posedness*). *Consider the norm* $\||(\boldsymbol{u}_h^n, p_h^n)\||_{\flat,n}^2 := \frac{1}{\Delta t}\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \||(\boldsymbol{u}_h^n, p_h^n)\||_{*,n}^2$. *Then there exists a constant* $c_{T4.10} > 0$ *such that for all* $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h^n \times Q_h^n$ *it holds that*

$$c_{T4.10}\||(\boldsymbol{u}_h^n, p_h^n)\||_{\flat,n} \leq \sup_{(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^n \times Q_h^n} \frac{A_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) + s_h^n((\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h))}{\||(\boldsymbol{v}_h, q_n)\||_{\flat,n}}.$$

The solution $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h^n \times Q_h^n$ *to (4.14) exists and is unique.*

Proof. The proof is a repetition of the arguments in the proofs of Lemma 3.16 and Corollary 3.17. See also the proof of [WRL21, Theorem 5.8] for a detailed proof. $\qquad\square$

4.3.3 Stability

Lemma 4.11 (*Stability of the velocity*). *For the velocity solution* $\boldsymbol{u}_h^n \in \boldsymbol{V}_h^n$, $n = 1, \ldots, N$ *of (4.7), we have the stability estimate*

$$\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \Delta t \sum_{i=1}^{n} \left[\frac{\nu c_{L4.11a}}{2} \||\boldsymbol{u}_h^i\||_{*,i}^2 + \frac{L}{\nu} i_h^n(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i) \right]$$

$$\leq \exp(c_{L4.11b}\nu^{-1}t^n) \left[\|\boldsymbol{u}_h^0\|_{\Omega_h^0}^2 + \frac{\nu \Delta t c_{L4.11a}}{2} \||\boldsymbol{u}_h^0\||_{*,0}^2 + \frac{L}{\nu} i_h^0(\boldsymbol{u}_h^0, \boldsymbol{u}_h^0) + \Delta t \sum_{i=1}^{n} \frac{c_{P,h}^2}{\nu c_{L4.11a}} \|\boldsymbol{f}_h^i\|_{\Omega_h^i}^2 \right].$$

Since this result is central to understand the scaling of the velocity ghost-penalty operator for the discrete extension, we give the proof as it is given in [WRL21].

Proof. We test the variational formulation (4.7) with $(\boldsymbol{v}_h, q_h) = 2\Delta t(\boldsymbol{u}_h^n, -p_h^n) \in \boldsymbol{V}_h^n \times Q_h^n$. Using the identity

$$2(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}, \boldsymbol{u}_h^n)_{\Omega_h^n} = \|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \|\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2 - \|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2 \tag{4.15}$$

then gives us the equation

$$\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \|\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2 + 2\Delta t \left[a_h^n(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n) + \nu \gamma_{gp} i_h^n(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n) + \frac{\gamma_{gp}}{\nu} i_h^n(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n) + \frac{\gamma_{gp}}{\nu} j_h^n(p_h^n, p_h^n) \right]$$
$$= 2\Delta t f_h^n(\boldsymbol{u}_h^n) + \|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2.$$

Using the coercivity result in Lemma 4.7, and the fact that $\|\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2 + \frac{2\Delta t \gamma_{gp}}{\nu} j_h^n(p_h^n, p_h^n) \geq 0$, we get

$$\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + 2\Delta t \nu c_{L4.7}\||\boldsymbol{u}_h^n\||_{*,n}^2 + \Delta t \frac{2}{\nu} L i_h^n(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n) \leq 2\Delta t f_h^n(\boldsymbol{u}_h^n) + \|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2. \tag{4.16}$$

To bound the forcing term, we use the Cauchy-Schwarz inequality, the discrete Poincaré-inequality (4.9) and a weighted Young's inequality. This gives the bound

$$f_h^n(\boldsymbol{u}_h^n) \leq \|\boldsymbol{f}^n\|_{\Omega_h^n} \|\boldsymbol{u}_h^n\|_{\Omega_h^n} \leq \|\boldsymbol{f}^n\|_{\Omega_h^n} c_{P,h} \||\boldsymbol{u}_h^n\||_{*,n} \leq \frac{c_{P,h}^2 \|\boldsymbol{f}^n\|_{\Omega_h^n}^2}{2\varepsilon} + \frac{\varepsilon \||\boldsymbol{u}_h^n\||_{*,n}^2}{2}.$$

With the choice $\varepsilon = \nu c_{L4.7}$, inserting this estimate into (4.16) then gives

$$\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \Delta t \nu c_{L4.7}\||\boldsymbol{u}_h^n\||_{*,n}^2 + \Delta t \frac{2}{\nu} L i_h^n(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n) \leq \frac{c_{P,h}^2 \Delta t}{\nu c_{L4.7}} \|\boldsymbol{f}^n\|_{\Omega_h^n}^2 + \|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2. \tag{4.17}$$

To obtain a bound on $\|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2$, we utilise the following result, c.f. [LO19, Lemma 5.7]:

$$\|\boldsymbol{u}_h\|_{\mathcal{O}_{\delta_h}(\Omega_h^n)}^2 \le (1 + c_1(\varepsilon)\Delta t)\|\boldsymbol{u}_h\|_{\Omega_h^n}^2 + c_2(\varepsilon)\nu\Delta t\|\nabla\boldsymbol{u}_h\|_{\Omega_h^n}^2 + c_3(\varepsilon, h)\Delta t L i_h^n(\boldsymbol{u}_h, \boldsymbol{u}_h) \tag{4.18}$$

with $c_1(\varepsilon) = c'c_{\delta_h}\boldsymbol{w}_\infty^n(1 + \varepsilon^{-1})$, $c_2(\varepsilon) = c'c_{\delta_h}\boldsymbol{w}_\infty^n\varepsilon/\nu$ and $c_3(\varepsilon) = c'c_{\delta_h}\boldsymbol{w}_\infty^n(\varepsilon + (1 + \varepsilon^{-1})h^2)$ and $c' > 0$ independent of h and Δt.

Choosing $\varepsilon = \nu c_{L4.7}/(2c'c_{\delta_h}\boldsymbol{w}^n)$, we then have $c_2 = c_{L4.7}/2$. We can then bound $c_1(\varepsilon) \le \overline{c}/\nu$, with $\overline{c}$ independent of Δt and h and $c_3 \le \nu c_{L4.7}/2 + h^2\overline{c}/\nu$. This yields

$$\|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^n}^2 \le \|\boldsymbol{u}_h^{n-1}\|_{\mathcal{O}_{\delta_h}(\Omega_h^{n-1})}^2$$
$$\le (1 + \frac{\overline{c}\Delta t}{\nu})\|\boldsymbol{u}_h^{n-1}\|_{\Omega_h^{n-1}}^2 + \frac{\nu\Delta t c_{L4.7}}{2}\|\|\boldsymbol{u}_h^{n-1}\|\|_{*,n-1}^2 + \frac{\overline{c}h^2}{\nu}\Delta t L i_h^{n-1}(\boldsymbol{u}_h^{n-1}, \boldsymbol{u}_h^{n-1}).$$

Inserting this into (4.17), taking $n \mapsto i$ and summing over $i = 1, \ldots, n$ for $n \le N$ then gives for sufficiently small h, such that $\overline{c}h^2 \le 1$, the bound

$$\|\boldsymbol{u}_h^n\|_{\Omega_h^n}^2 + \Delta t \sum_{i=1}^n \left[\frac{\nu c_{L4.7}}{2}\|\|\boldsymbol{u}_h^i\|\|_{*,i}^2 + \frac{L}{\nu}i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i)\right]$$
$$\le \|\boldsymbol{u}_h^0\|_{\Omega_h^0}^2 + \frac{\nu\Delta t c_{L4.7}}{2}\|\|\boldsymbol{u}_h^0\|\|_{*,0}^2 + \frac{L}{\nu}i_h^0(\boldsymbol{u}_h^0, \boldsymbol{u}_h^0) + \Delta t \sum_{i=1}^n \frac{c_{P,h}^2}{\nu c_{L4.7}}\|\boldsymbol{f}^i\|_{\Omega_h^i}^2 + \Delta t \sum_{i=0}^n \frac{\overline{c}}{\nu}\|\boldsymbol{u}_h^i\|_{\Omega_h^i}^2.$$

Applying a discrete Gronwall inequality, c.f. [Joh16, Lemma A.56], with $c_{L4.13a} = c_{L4.7}$ and $c_{L4.13b} = \overline{c}$ then gives the desired result. □

Remark 4.12. From the final two inequalities in the proof of Lemma 4.11, we see that if the estimate $\|\boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h,\mathscr{T}}^n} \le \|\boldsymbol{v}_h\|_{\Omega_h^n} + h^2 i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h)$ held true, i.e., if the ghost-penalty mechanism from below held true with constant $c = 1$, we could replace the $\frac{1}{\nu}i_h^n(\boldsymbol{u}_h^n, \boldsymbol{v}_h)$ ghost-penalties with $\frac{1}{\Delta t}h^2 i_h^n(\boldsymbol{u}_h^n, \boldsymbol{v}_h)$ ghost-penalties in the discrete method and obtain a stability result which was independent of the mesh-interface cut positions, i.e., replace the $\|\cdot\|_{\mathcal{F}_h^n}$ with a true discrete norm on the entire active mesh $\mathcal{O}_{\delta_h,\mathscr{T}}^n$. However, numerical experiments have also shown that this stabilisation does not result in a stable method. ▲

Lemma 4.13 (*Pressure stability*). *For the pressure solution $p_h^n \in Q_h^n$ of (4.7), we have that*

$$\|p_h^n\|_{\Omega_h^n} \le c_{L4.13}\left[\|\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})\|_{-1,n} + \|\|\boldsymbol{u}_h^n\|\|_{*,n} + \|\boldsymbol{f}^n\|_{\Omega_h^n} + j_h^n(p_h^n, p_h^n)^{1/2}\right]. \tag{4.19}$$

Proof. With the test-function $q_h = 0$, we have from (4.7) that

$$b_h^n(p_h^n, \boldsymbol{v}_h) = -\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h)_{\Omega_h^n} - (a_h^n + \nu L i_h^n)(\boldsymbol{u}_h^n, \boldsymbol{v}_h) - {}^1/\nu i_h^n(\boldsymbol{u}_h^n, \boldsymbol{v}_h) + f_h^n(\boldsymbol{v}_h)$$
$$\le \left[\|\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})\|_{-1,n} + c(\nu + {}^1/\nu)\|\|\boldsymbol{u}_h^n\|\|_{*,n} + c_{P,h}\|\boldsymbol{f}^n\|_{\Omega_h^n}\right]\|\|\boldsymbol{v}_h\|\|_{*,n}$$

Here we used the continuity of $(a_h^n + \nu L i_h^n)(\cdot, \cdot)$, the estimate $i_h^n(\boldsymbol{u}, \boldsymbol{v}) \le i_h^n(\boldsymbol{u}, \boldsymbol{u})^{1/2} i_h^n(\boldsymbol{v}, \boldsymbol{v})^{1/2}$ with (4.11a) and the Poincaré inequality. The result then follows from the inf-sup result in Corollary 4.8. □

Remark 4.14. The above stability estimate (4.19) is not optimal, since the right-hand side depends on Δt^{-1}. Therefore, it only gives a stability bound for $\Delta t^2 \sum_{i=1}^n \|p_h^i\|_{\Omega_h^i}^2$ rather than $\Delta t \sum_{i=1}^n \|p_h^i\|_{\Omega_h^i}^2$. To get such a bound, it is necessary to show a bound for $\|\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})\|_{-1,n}$, independent of negative powers of Δt. At the time of writing, we do not have such an estimate. However, we do expect that such a bound is possible, since $\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})$ is a first-order approximation of $\partial_t \boldsymbol{u}_h$. ▲

Remark 4.15 (Units of the stabilisation). Looking at the stabilisation scaling $\nu + \frac{1}{\nu}$, we note that the addition does not make sense when the units of ν are taken into account. Looking at the proof of Lemma 4.11, we see that the $\nu + \frac{1}{\nu}$ scaling results from the choice of ε in (4.18). Looking at (4.18) in more detail, we see that in order for the inequality to conserve dimensional units, $c_1(\varepsilon)$ must have units s^{-1}, $c_2(\varepsilon)$ must be dimensionless, and $c_3(\varepsilon, h)$ must have units $\mathrm{m}^2\,\mathrm{s}^{-1}$. By the definition of these three constants, it follows that ε must have units m. The choice $\varepsilon = \nu c_{L4.7}/(2c'c_{\delta_h}\boldsymbol{w}^{\boldsymbol{n}}_\infty)$ does indeed have units $\mathrm{m}^2\,\mathrm{s}^{-1}\,\mathrm{m}^{-1}\,\mathrm{s} = \mathrm{m}$. The correct units from the squared interface normal velocity $\boldsymbol{w}^{\boldsymbol{n}}_\infty$ are therefore lost in the constant $\bar{c}$ and in the estimate $\bar{c}h^2 < 1$.

An alternative, to keep the units of the extension stabilisation consistent, would be to estimate $\boldsymbol{w}^{\boldsymbol{n}}_\infty \leq \delta/\Delta t$ and then carry this through to the extension ghost-penalties. However, numerical experiments have shown that this does not help either the accuracy or the stability. Therefore, we have kept the scaling as in [WRL21] to keep the presentation readable. ▲

4.3.4 Error Analysis

Geometry Approximation

For the error analysis, we shall assume that we have a higher-order approximation of the level set geometry, i.e.,

$$\mathrm{dist}(\Omega^n, \Omega^n_h) \lesssim h^{q+1},$$

with the geometry approximation order $q \geq 1$. Furthermore, we assume that integrals over Ω^n_h and Γ^n_h can be computed sufficiently accurate. In comparison to the parametric higher-order geometry approximation described in subsection 3.1.3, the essential difference is that we assume to realise the higher geometry approximation *without deforming the mesh*. This is because for moving domain problems with the parametric mapping, the discrete function $\boldsymbol{u}^n_h$ and $\boldsymbol{u}^{n-1}_h$ are defined with respect to different mappings. Therefore, this method requires an efficient and accurate transfer operator in order to evaluate $\boldsymbol{u}^{n-1}_h$ correctly on the mesh which is deformed with respect to the geometry at time t^n. In the context of a scalar convection-diffusion problem on a moving domain, this method, including the transfer operator, has been analysed in [LL21].

We further assume the existence of a well-defined, continuous mapping $\Phi\colon \mathcal{O}_{\delta_h}(\Omega^n_h) \to \mathcal{O}_{\delta_h}(\Omega^n)$, which maps the approximated domain onto the exact domain, i.e., it holds that $\Omega^n = \Phi(\Omega^n_h)$, $\Gamma^n = \Phi(\Gamma^n_h)$ and $\mathcal{O}_{\delta_h}(\Omega^n) = \Phi(\mathcal{O}_{\delta_h}(\Omega^n_h))$, as well as

$$\|\Phi - \mathrm{id}\|_{\boldsymbol{\mathcal{L}}^\infty(\mathcal{O}_{\delta_h}(\Omega^n_h))} \lesssim h^{q+1}, \quad \|D\Phi - I\|_{\boldsymbol{\mathcal{L}}^\infty(\mathcal{O}_{\delta_h}(\Omega^n_h))} \lesssim h^q, \quad \|\det(D\Phi) - 1\|_{\boldsymbol{\mathcal{L}}^\infty(\mathcal{O}_{\delta_h}(\Omega^n_h))} \lesssim h^q. \quad (4.20)$$

For sufficiently small h, we take the mapping to be invertible. Such a mapping has been constructed, for example, in [GOR15, Section 7.1]. As in chapter 3, the mapping Φ is used to map from the discrete domain to the exact one. Let $\boldsymbol{v}_h \in \boldsymbol{V}^n_h$ and define $\boldsymbol{v}^\ell_h = \boldsymbol{v}_h \circ \Phi^{-1}$. From the third estimate in (4.20), we have $\det(D\Phi) \simeq 1$, hence we get using integration by substitution that

$$\|\boldsymbol{v}^\ell_h\|^2_{\mathcal{O}_{\delta_h}(\Omega^n)} = \sum_{i=1}^d \int_{\mathcal{O}_{\delta_h}(\Omega^n)} (\boldsymbol{v}^\ell_h)^2_i \,\mathrm{d}\hat{\boldsymbol{x}} = \sum_{i=1}^d \int_{\mathcal{O}_{\delta_h}(\Omega^n_h)} \det(D\Phi)(\boldsymbol{v}_h)^2_i \,\mathrm{d}\boldsymbol{x} \simeq \|\boldsymbol{v}_h\|^2_{\mathcal{O}_{\delta_h}(\Omega^n_h)}.$$

Using similar arguments, we also find

$$\|\boldsymbol{v}_h\|^2_{\Omega^n_h} \simeq \|\boldsymbol{v}^\ell_h\|^2_{\Omega^n} \qquad \text{and} \qquad \|\boldsymbol{v}_h\|^2_{\Gamma^n_h} \simeq \|\boldsymbol{v}^\ell_h\|^2_{\Gamma^n},$$

as well as

$$\|\nabla\boldsymbol{v}_h\|^2_{\mathcal{O}_{\delta_h}(\Omega^n_h)} \simeq \|\nabla\boldsymbol{v}^\ell_h\|^2_{\mathcal{O}_{\delta_h}(\Omega^n)} \qquad \text{and} \qquad \|\nabla\boldsymbol{v}_h\|^2_{\Omega^n_h} \simeq \|\nabla\boldsymbol{v}^\ell_h\|^2_{\Omega^n}.$$

For the extension, we also have the following result, c.f. [GOR15, Lemma 7.3].

Lemma 4.16.   *The estimates*

$$\|\mathcal{E}\boldsymbol{u} - \boldsymbol{u} \circ \Phi\|_{\Omega_h^n} \lesssim h^{q+1}\|\boldsymbol{u}\|_{\mathcal{H}^1(\Omega^n)}, \qquad\qquad \|\nabla(\mathcal{E}\boldsymbol{u}) - (\nabla\boldsymbol{u}) \circ \Phi\|_{\Omega_h^n} \lesssim h^{q+1}\|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega^n)},$$

$$\|\mathcal{E}\boldsymbol{u} - \boldsymbol{u} \circ \Phi\|_{\Gamma_h^n} \lesssim h^{q+1}\|\boldsymbol{u}\|_{\mathcal{H}^2(\Omega^n)}, \qquad\qquad \|\mathcal{E}\partial_{\boldsymbol{n}}\boldsymbol{u} - \partial_{\boldsymbol{n}}\boldsymbol{u} \circ \Phi\|_{\Gamma_h^n} \lesssim h^{q+1}\|\boldsymbol{u}\|_{\mathcal{H}^3(\Omega^n)},$$

hold for all $\boldsymbol{u} \in \mathcal{H}^3(\Omega^n)$, $n = 1, \dots, N$. Furthermore, it also holds that

$$\|\mathcal{E}\boldsymbol{u} - (\mathcal{E}\boldsymbol{u}) \circ \Phi\|_{\mathcal{O}_{\delta_h}(\Omega_h^n)} \lesssim h^{q+1}\|\boldsymbol{u}\|_{\mathcal{H}^1(\Omega^n)}. \tag{4.21}$$

Proof. For the first four estimates, we refer to [GOR15, Lemma 7.3] and c.f. Lemma 3.10. The proof of (4.21) follows the identical lines of the proof of the estimate over Ω_h^n but integration over $\mathcal{S}^\pm(\Omega_h^n)$ rather that $\mathcal{S}^\pm(\Omega_h^n) \setminus \mathcal{S}^+(\Omega_h^n)$. $\square$

Consistency

As in subsection 3.2.5, we cannot test the weak formulation of the smooth problem with discrete test-functions due to the mismatch of the exact and discrete domains. However, we can again use the mapping Φ to construct suitable test-functions $(\boldsymbol{v}_h^\ell, q_h^\ell) = (\boldsymbol{v}_h \circ \Phi^{-1}, q_h \circ \Phi^{-1})$. Using integration by parts, we then see that any smooth solution to (2.14) fulfills the variational form

$$\int_{\Omega^n} \partial_t \boldsymbol{u}(t_n) \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} + a_1^n(\boldsymbol{u}(t_n), \boldsymbol{v}_h^\ell) + b_1^n(p(t_n), \boldsymbol{v}_h^\ell) + b_1^n(q_h^\ell, \boldsymbol{u}(t_n)) = f^n(\boldsymbol{v}_h^\ell) \tag{4.22}$$

for all $(\boldsymbol{v}_h, q_h) \in \boldsymbol{V}_h^n \times Q_h^n$, with the the mapping Φ as described above, and the bilinear forms

$$a_1^n(\boldsymbol{u}, \boldsymbol{v}) := a^n(\boldsymbol{u}, \boldsymbol{v}) + \nu \int_{\Gamma^n} (-\partial_{\boldsymbol{n}}\boldsymbol{u}) \cdot \boldsymbol{v} \, \mathrm{d}s \qquad \text{and} \qquad b_1^n(p, \boldsymbol{v}) := b^n(p, \boldsymbol{v}) + \int_{\Gamma^n} p\boldsymbol{v} \cdot \boldsymbol{n} \, \mathrm{d}s,$$

as well as the forcing term $f^n(\boldsymbol{v}) := \int_{\Omega^n} \boldsymbol{f}(t^n) \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}$. For simplicity of notation, we shall identify the smooth extension $(\mathcal{E}\boldsymbol{u}, \mathcal{E}p)$ with $(\boldsymbol{u}, p)$, and denote as before

$$\mathbb{E}^n := \boldsymbol{u}^n - \boldsymbol{u}_h^n \qquad \text{and} \qquad \mathbb{D}^n := p^n - p_h^n.$$

Since $\Omega_h^n \subset \mathcal{O}_\delta(\Omega(t))$ for $t \in [t^{n-1}, t^n]$, we have that $\boldsymbol{u}(t^{n-1}) = \boldsymbol{u}^{n-1}$ is well-defined on Ω_h^n. Subtracting (4.7) from (4.22), adding and subtracting appropriate terms and rearranging, we obtain the error equation

$$\int_{\Omega_h^n} \frac{\mathbb{E}^n - \mathbb{E}^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + a_h^n(\mathbb{E}^n, \boldsymbol{v}_h) + b_h^n(\mathbb{D}^n, \boldsymbol{v}_h) + b_h^n(q_h, \mathbb{E}^n) + s_h^n((\mathbb{E}^n, \mathbb{D}^n), (\boldsymbol{v}_h, q_h))$$

$$= f^n(\boldsymbol{v}_h^\ell) - f_h^n(\boldsymbol{v}_h) + \int_{\Omega_h^n} \frac{\boldsymbol{u}^n - \boldsymbol{u}^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \int_{\Omega^n} \partial_t \boldsymbol{u}^n \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} + a_h^n(\boldsymbol{u}^n, \boldsymbol{v}_h) - a_1^n(\boldsymbol{u}^n, \boldsymbol{v}_h^\ell)$$

$$b_h^n(p^n, \boldsymbol{v}_h) - b_1^n(p^n, \boldsymbol{v}_h^\ell) + b_h^n(q_h, \boldsymbol{u}^n) - b_1^n(q_h^\ell, \boldsymbol{u}^n) + s_h^n((\boldsymbol{u}^n, p^n), (\boldsymbol{v}_h, q_h))$$

$$= \mathfrak{T}_1 + \mathfrak{T}_2 + \mathfrak{T}_3 + \mathfrak{T}_4 + \mathfrak{T}_5 + \mathfrak{T}_6$$

$$= \mathfrak{E}_c^n(\boldsymbol{v}_h, q_h). \tag{4.23}$$

These six terms correspond to the forcing, time-derivative, diffusion, pressure, divergence constraint and ghost-penalty contributions, respectively.

Lemma 4.17 (*Consistency estimate*).   *The consistency error of the Eulerian time-stepping scheme for the time-dependent Stokes problem on a moving domain has the bound*

$$|\mathfrak{E}_c^n((\boldsymbol{v}_h, q_h))| \lesssim \left(\Delta t + h^q + \frac{h^k L^{1/2}}{\nu}\right) R_{c,1}(\boldsymbol{u}, p, \boldsymbol{f})\|\|\boldsymbol{v}_h\|\|_{*,n} + (h^q + h^k) R_{c,2}(\boldsymbol{u}, p)\|\|q_h\|\|_{*,n}$$

with

$$R_{c,1}(\boldsymbol{u}, p, \boldsymbol{f}) = \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} + \sup_{t \in [0, t^{end}]} \Big(\|\boldsymbol{f}(t)\|_{\boldsymbol{\mathcal{H}}^1(\Omega(t))} + \|\boldsymbol{u}(t)\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|p(t)\|_{\mathcal{H}^k(\Omega(t))} \Big)$$

$$R_{c,2}(\boldsymbol{u}, p) = \sup_{t \in [0, t^{end}]} \Big(\|\boldsymbol{u}(t)\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|p(t)\|_{\mathcal{H}^k(\Omega(t))} \Big).$$

Proof. Most of the proof follows along the identical lines of Lemma 3.21 and Lemma 3.28, since the mapping Φ used here and the mapping Φ_h in chapter 3 fundamentally have the same properties, i.e., (4.20) and Lemma 4.16. Thus, the only term we need to deal with separately is the time-derivative term $\mathfrak{T}_2$. For the other terms, we summarise

$$|\mathfrak{T}_1| \lesssim h^k \|\boldsymbol{f}^n\|_{\boldsymbol{\mathcal{H}}^1(\Omega^n)} \|\boldsymbol{v}_h\|_{\Omega_h^n}, \qquad\qquad |\mathfrak{T}_3| \lesssim h^k \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^3(\Omega^n)} \|\|\boldsymbol{v}_h\|\|_{*,n},$$

$$|\mathfrak{T}_4| \lesssim h^k \|p^n\|_{\boldsymbol{\mathcal{H}}^2(\Omega^n)} \|\|\boldsymbol{v}_h\|\|_{*,n}, \qquad\qquad |\mathfrak{T}_5| \lesssim h^k \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^2(\Omega^n)} \|\|q_h\|\|_{*,n},$$

and

$$|\mathfrak{T}_6| \lesssim h^k (\nu + {}^1/\nu) L^{1/2} \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)} \|\|\boldsymbol{v}_h\|\|_{*,n} + h^k \|p^n\|_{\boldsymbol{\mathcal{H}}^k(\Omega^n)} \|\|q_h\|\|_{*,n},$$

where we have used that $\mathcal{O}^n_{\delta_h, \mathscr{T}} \subset \mathcal{O}_\delta(\Omega^n)$, and take the extension scaling into account.

For the time-derivative error, we proceed similarly as in the Oseen case but use different norms. We have

$$\mathfrak{T}_2 = \int_{\Omega_h^n} \int_{t^{n-1}}^{t^n} \frac{t - t^{n-1}}{\Delta t} \partial_t^2 \boldsymbol{u} \, \mathrm{d}t \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \int_{\Omega_h^n} \partial_t \boldsymbol{u}(t^n) \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \int_{\Omega^n} \partial_t \boldsymbol{u}(t^n) \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}}.$$

Then with $\Omega_h^n \subset \mathcal{O}_\delta(\Omega(t))$ for $t \in [t^{n-1}, t^n]$ and the stability of the extension (4.5c), it holds for the first component that

$$\left| \int_{\Omega_h^n} \int_{t^{n-1}}^{t^n} \frac{t - t^{n-1}}{\Delta t} \partial_t^2 \boldsymbol{u} \, \mathrm{d}t \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \right| \leq \frac{1}{2} \Delta t \|\partial_t^2 \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^\infty(\mathcal{O}_\delta(\mathcal{Q}))} \|\boldsymbol{v}_h\|_{\boldsymbol{\mathcal{L}}^1(\Omega_h^n)} \lesssim \Delta t \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \|\boldsymbol{v}_h\|_{\Omega_h^n}.$$

For the second part we have

$$\int_{\Omega_h^n} \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - \int_{\Omega^n} \partial_t \boldsymbol{u} \cdot \boldsymbol{v}_h^\ell \, \mathrm{d}\hat{\boldsymbol{x}} = \int_{\Omega_h^n} (\partial_t \boldsymbol{u} - (\partial_t \boldsymbol{u}) \circ \Phi_h) \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + \int_{\Omega_h^n} (\partial_t \boldsymbol{u}) \circ \Phi_h \cdot \boldsymbol{v}_h (1 - J) \, \mathrm{d}\boldsymbol{x}$$

$$\lesssim h^q \|\nabla \partial_t \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^\infty(\mathcal{O}_\delta(\Omega(t)))} \|\boldsymbol{v}_h\|_{\Omega_h^{ho}} + h^q \|\partial_t \boldsymbol{u}\|_{\Omega^n} \|\boldsymbol{v}_h\|_{\Omega_h^n}$$

$$\lesssim h^q \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \|\boldsymbol{v}_h\|_{\Omega_h^n}.$$

Combining these estimates and taking the supremum over $t \in [0, t^{end}]$. $\qquad\qquad\square$

Remark 4.18. The above consistency estimate is not balanced in the case of a piecewise linear level set approximation, i.e, $q = 1$, since we have $k \geq 2$. In [WRL21], we have shown a slightly different proof, in which the two balance. This was achieved by considering a different estimate for the difference between the mapped and extended functions on the boundary, which included additive interpolation terms, c.f. [WRL21, Lemma 5.13]. However, since we require the regularity $k + 1$ and k for the velocity and pressure for optimal order convergence, we only show this version of the result. $\qquad\blacktriangle$

Energy Error

For the error estimate, we again split the velocity and pressure errors into an interpolation and discretisation part

$$\mathbb{E}^n = (\boldsymbol{u}^n - \mathcal{I}_h^* \boldsymbol{u}^n) + (\mathcal{I}_h^* \boldsymbol{u}^n - \boldsymbol{u}_h) =: \boldsymbol{\eta}^n + \mathbf{e}_h^n \qquad\qquad (4.24\mathrm{a})$$

$$\mathbb{D}^n = (p^n - \mathcal{I}_h^* p^n) + (\mathcal{I}_h^* p^n - p_h^n) =: \zeta^n + \mathrm{d}_h^n, \qquad\qquad (4.24\mathrm{b})$$

c.f. (3.26). Note that we have implicitly included the extension operator in the smooth solution part so that all terms are well-defined on the entire active mesh. Inserting this splitting into the error equation (4.23) and rearranging the terms then yields

$$\int_{\Omega_h^n} \frac{\mathbf{e}_h^n - \mathbf{e}_h^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} + a_n^h(\mathbf{e}_h^n, \boldsymbol{v}_h) + b_h^n(\mathrm{d}_h^n, \boldsymbol{v}_h) + b_h^n(q_h, \mathbf{e}_h^n) + s_h^n((\mathbf{e}_h^n, \mathrm{d}_h^n), (\boldsymbol{v}_h, q_h))$$

$$= \mathfrak{E}_c^n(\boldsymbol{v}_h, q_h) + \mathfrak{E}_I^n(\boldsymbol{v}_h, q_h), \quad (4.25)$$

with

$$\mathfrak{E}_I^n(\boldsymbol{v}_h, q_h) = -\int_{\Omega_h^n} \frac{\boldsymbol{\eta}^n - \boldsymbol{\eta}^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} - a_n^h(\boldsymbol{\eta}^n, \boldsymbol{v}_h) - b_h^n(\zeta^n, \boldsymbol{v}_h) - b_h^n(q_h, \boldsymbol{\eta}^n) - s_h^n((\boldsymbol{\eta}^n, \zeta^n), (\boldsymbol{v}_h, q_h)).$$

For the interpolation error component we can then prove the following bound.

Lemma 4.19 (*Interpolation error estimate*).   *We assume for the smooth velocity and pressure solution the regularity* $\boldsymbol{u} \in \boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^{k+1}(\Omega(t)))$, $\boldsymbol{u}_t \in \boldsymbol{\mathcal{L}}^\infty(0, T; \boldsymbol{\mathcal{H}}^k(\Omega(t)))$ *and* $p \in \mathcal{L}^\infty(0, T; \mathcal{H}^k(\Omega(t)))$. *The interpolation error term can then be bounded by.*

$$|\mathfrak{E}_I^n(\boldsymbol{v}_h, q_h)| \lesssim \frac{h^k L^{1/2}}{\nu} R_{I,1}(\boldsymbol{u}, p) |||\boldsymbol{v}_h|||_{*,n} + h^k R_{I,2}(\boldsymbol{u}, p) |||q_h|||_{*,n},$$

with

$$R_{I,1}(\boldsymbol{u}, p) = \sup_{t \in [0, t^{end}]} \left(\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|\boldsymbol{u}_t\|_{\boldsymbol{\mathcal{H}}^k(\Omega(t))} + \|p\|_{\mathcal{H}^k(\Omega(t))} \right)$$

$$R_{I,2}(\boldsymbol{u}, p) = \sup_{t \in [0, t^{end}]} \left(\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|p\|_{\mathcal{H}^k(\Omega(t))} \right).$$

Proof. We split the interpolation error terms into five different parts $\mathfrak{E}_I^n(\boldsymbol{v}_h, q_h) = \mathfrak{T}_7 + \mathfrak{T}_8 + \mathfrak{T}_9 + \mathfrak{T}_{10} + \mathfrak{T}_{11}$. These are the time-derivative term, the diffusion bilinear form, the pressure coupling term, the divergence constraint and ghost-penalty operator, respectively. As in Lemma 4.17, we deal with each constituent term separately.

For the time-derivative contribution $\mathfrak{T}_7$, we have from [LO19, Lemma 5.12] that

$$|\mathfrak{T}_7| = \Big| \int_{\Omega_h^n} \frac{\boldsymbol{\eta}^n - \boldsymbol{\eta}^{n-1}}{\Delta t} \cdot \boldsymbol{v}_h \, \mathrm{d}\boldsymbol{x} \Big| \lesssim h^k \sup_{t \in [0, t^{end}]} \left(\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega(t))} + \|\boldsymbol{u}_t\|_{\boldsymbol{\mathcal{H}}^m(\Omega(t))} \right) \|\boldsymbol{v}_h\|_{\Omega_h^n}.$$

For the diffusion term $\mathfrak{T}_8$, we use the continuity result (4.12a) and Lemma 4.4 for the interpolation term, and (4.8) for the test function. This gives

$$|\mathfrak{T}_8| = |-a_h^n(\boldsymbol{\eta}^n, \boldsymbol{v}_h)| \lesssim |||\boldsymbol{\eta}^n|||_n |||\boldsymbol{v}_h|||_n \lesssim h^k \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)} |||\boldsymbol{v}_h|||_{*,n}.$$

Using the same technique, we can estimate the pressure and divergence bilinear forms as

$$|\mathfrak{T}_9| \ = |-b_h^n(\zeta^n, \boldsymbol{v}_h)| \lesssim h^k \|p^n\|_{\mathcal{H}^k(\Omega^n)} |||\boldsymbol{v}_h|||_{*,n}$$

$$|\mathfrak{T}_{10}| = |-b_h^n(q_h, \boldsymbol{\eta}^n)| \lesssim h^k \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)} |||q_h|||_{*,n}.$$

For the ghost-penalty term $\mathfrak{T}_{11}$, we use the Cauchy-Schwarz inequality and Lemma 3.4 taking the larger ghost-penalty strip into account, together with (4.5a); see also [LO19, Lemma 5.12]. We then have

$$|\mathfrak{T}_{11}| = |s_h^n((\boldsymbol{\eta}^n, \zeta^n), (\boldsymbol{v}_h, q_h))| \lesssim (\nu + 1/\nu) L^{1/2} i_h^n(\boldsymbol{\eta}^n, \boldsymbol{\eta}^n)^{1/2} L^{1/2} i_h^n(\boldsymbol{v}_h, \boldsymbol{v}_h)^{1/2} + j_h^n(\zeta^n, \zeta^n)^{1/2} j_h^n(q_h, q_h)^{1/2}$$

$$\lesssim (\nu + 1/\nu) h^k L^{1/2} \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^{k+1}(\mathcal{O}_{\delta_h, \mathscr{T}}^n)} \|\boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^n} + h^k \|p^n\|_{\mathcal{H}^k(\mathcal{O}_{\delta_h, \mathscr{T}}^n)} \|q_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^n}$$

$$\lesssim (\nu + 1/\nu) h^k L^{1/2} \|\boldsymbol{u}^n\|_{\boldsymbol{\mathcal{H}}^{k+1}(\Omega^n)} |||\boldsymbol{v}_h|||_{*,n} + h^k \|p^n\|_{\mathcal{H}^k(\Omega^n)} |||q_h|||_{*,n}.$$

Combining these estimates then proves the claim. $\square$

Theorem 4.20 (*Energy error estimate*). *For sufficiently small Δt and h, the velocity error can be bounded by*

$$\|\mathbb{E}^n\|_{\Omega_h^n}^2 + \sum_{i=1}^n \left\{ \|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\Omega_h^i}^2 + \Delta t \left[\nu c_{L4.7} \|\mathbb{E}^i\|_{*,i}^2 + \frac{L}{\nu} i_h^i(\mathbb{E}^i, \mathbb{E}^i) \right] \right\}$$

$$\leq \exp((c_{T4.20a}/\nu)t_n) \left[\Delta t \sum_{i=1}^n c_{T4.20b} \left[\Delta t^2 + h^{2q} + \frac{h^{2k}L}{\nu} + \frac{1}{\Delta t}(h^{2q} + \frac{h^{2k}}{\nu}) \right] R(\boldsymbol{u}, p, \boldsymbol{f}) \right],$$

with $R(\boldsymbol{u}, p, \boldsymbol{f}) = \sup_{t \in [0,t^{end}]}(\|\boldsymbol{u}\|_{\mathcal{H}^{k+1}(\Omega(t))}^2 + \|\boldsymbol{u}_t\|_{\mathcal{H}^k(\Omega(t))}^2 + \|p\|_{\mathcal{H}^k(\Omega(t))}^2) + \|\boldsymbol{u}\|_{\mathcal{W}^{2,\infty}(\mathcal{Q})}^2 + \|\boldsymbol{f}\|_{\mathcal{H}^1(\Omega^n)}^2$ and constants $c_{T4.20a}, c_{T4.20b}$ independent of Δt, n and h. For the pressure, we have the bound

$$\Delta t^2 \sum_{i=1}^n \|\mathbb{D}^i\|_{*,i}^2 \lesssim \Delta t \sum_{i=1}^{n-1} \frac{1}{\nu} \|\mathbb{E}^i\|_{\Omega_h^i}^2 + \Delta t \sum_{i=1}^n c \left[\Delta t^2 + h^{2q} + \frac{h^{2k}L}{\nu} + \frac{1}{\Delta t}(h^{2q} + \frac{h^{2k}}{\nu}) \right] R(\boldsymbol{u}, p, \boldsymbol{f}).$$

Proof. We prove the result for the discretisation error since the result then immediately follows by optimal interpolation properties. We start with the velocity estimate. Similar to the stability proof, for $n \mapsto i$, we test the error equation (4.25) with the test-function $(\boldsymbol{v}_h, q_h) = 2\Delta t(\mathbf{e}_h^i, -\mathrm{d}_h^i)$ and use the polarisation identity (4.15) to get

$$\|\mathbf{e}_h^i\|_{\Omega_h^i}^2 + \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2 + 2\Delta t(a_h^i + \nu L i_h^i)(\mathbf{e}_h^i, \mathbf{e}_h^i) + 2L\Delta t/\nu i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i) + 2\Delta t/\nu j_h^i(\mathrm{d}_h^i, \mathrm{d}_h^i)$$
$$= 2\Delta t(\mathfrak{C}_c^i + \mathfrak{E}_I^i)(\mathbf{e}_h^i, -\mathrm{d}_h^i) + \|\mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2.$$

Using the coercivity result Lemma 4.7 and (4.18), we get (with the appropriate choice of ε) that

$$\|\mathbf{e}_h^i\|_{\Omega_h^i}^2 + \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2 + 2\Delta t c_{L4.7} \|\mathbf{e}_h^i\|_{*,i}^2 + \Delta t \frac{2L}{\nu} i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i) + \Delta t \frac{2}{\nu} j_h^i(\mathrm{d}_h^i, \mathrm{d}_h^i)$$

$$\leq (1 + \frac{\overline{c}}{\nu}\Delta t)\|\mathbf{e}_h^{i-1}\|_{\Omega_h^{i-1}}^2 + \Delta t \frac{\nu c_{L4.7}}{2} \|\mathbf{e}_h^{i-1}\|_{*,i-1}^2 + \Delta t \frac{c'h^2}{\nu} L i_h^{i-1}(\mathbf{e}_h^{i-1}, \mathbf{e}_h^{i-1})$$
$$+ 2\Delta t(|\mathfrak{C}_c^i| + |\mathfrak{C}_I^i|)(\mathbf{e}_h^i, \mathrm{d}_h^i).$$

Applying the weighted Young inequality to Lemma 4.17 and Lemma 4.19 then gives

$$|\mathfrak{C}_c^i(\mathbf{e}_h^i, \mathrm{d}_h^i) + \mathfrak{E}_I^i(\mathbf{e}_h^i, \mathrm{d}_h^i)|$$
$$\leq \frac{1}{\varepsilon_1} c \left(\Delta t^2 + h^{2q} + \frac{h^{2k}L}{\nu^2} \right) R(\boldsymbol{u}, p, \boldsymbol{f}) + \varepsilon_1 \|\mathbf{e}_h^i\|_{*,i}^2 + \frac{1}{\varepsilon_2} c \left(h^{2q} + \frac{h^{2k}}{\nu^2} \right) R'(\boldsymbol{u}, p), + \varepsilon_2 \|\mathrm{d}_h^i\|_{*,i}^2 \quad (4.26)$$

with

$$R'(\boldsymbol{u}, p) = \sup_{t \in [0,t^{\mathrm{end}}]} \left(\|\boldsymbol{u}(t)\|_{\mathcal{H}^{k+1}(\Omega(t))}^2 + \|p(t)\|_{\mathcal{H}^k(\Omega(t))}^2 \right).$$

Now, we choose $\varepsilon_1 = \nu c_{L4.7}/4$ and $\varepsilon_2 = \Delta t \beta^2/4c_Y c_{P,h}^2$. With the constant $c_Y > 0$ to be specified later. Inserting these bounds on the consistency and interpolation estimates into the above inequality, summing over $i = 1, \ldots, n$ and using $\mathbf{e}_h^0 = \mathbf{0}$ gives

$$\|\mathbf{e}_h^n\|_{\Omega_h^n}^2 + \sum_{i=1}^n \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2 + \Delta t \sum_{i=1}^n \left[\nu c_{L4.7} \|\mathbf{e}_h^i\|_{*,i}^2 + \frac{L}{\nu} i_h^k(\mathbf{e}_h^i, \mathbf{e}_h^i) \right] + \Delta t \sum_{i=1}^n \frac{2}{\nu} j_h^k(\mathrm{d}_h^i, \mathrm{d}_h^i)$$

$$\leq \Delta t \sum_{i=1}^{n-1} \frac{\overline{c}}{\nu} \|\mathbf{e}_h^i\|_{\Omega_h^i}^2 + \Delta t^2 \sum_{i=1}^2 \frac{\beta^2}{2c_Y c_{P,h}^2} \|\mathrm{d}_h^i\|_{*,i}^2$$

$$+ \Delta t \sum_{i=1}^n c \left[\Delta t^2 + h^{2q} + \frac{h^{2k}L}{\nu} + \frac{1}{\Delta t}(h^{2q} + \frac{h^{2k}}{\nu}) \right] R(\boldsymbol{u}, p, \boldsymbol{f}), \quad (4.27)$$

under the assumption that h is sufficiently small such that $c'h^2 \leq 1$. To complete the velocity estimate, we therefore require the pressure estimate.

Rearranging the error equation (4.25) and using the test-function $q_h = 0$ gives

$$b_h^k(\mathrm{d}_h^i, \boldsymbol{v}_h) = -(1/\Delta t(\mathbf{e}_h^i - \mathbf{e}_h^{i-1}), \boldsymbol{v}_h)_{\Omega_h^i} - (a_h^k + \nu L i_h^k)(\mathbf{e}_h^i, \boldsymbol{v}_h) - \frac{L}{\nu} i_h^k(\mathbf{e}_h^i, \boldsymbol{v}_h) + (\mathfrak{E}_c^i + \mathfrak{E}_I^i)(\mathbf{e}_h^i, 0)$$

$$\leq \Big[\frac{c_{P,h}}{\Delta t} \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i} + \nu c_{L4.6} \|\|\mathbf{e}_h^i\|\|_{*,i} + \frac{L^{1/2}}{\nu} i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i)^{1/2}$$

$$+ \hat{c}\big(\Delta t + h^q + \frac{L^{1/2}h^m}{\nu}\big)(R_{c,1} + R_{I,1})(\boldsymbol{u}, p, \boldsymbol{f}) \Big] \|\|\boldsymbol{v}_h\|\|_{*,i},$$

where $\hat{c} = c_{L4.17} + c_{L4.19}$. Using the inf-sup result from Corollary 4.8 together with (4.11c), we then have

$$\beta \|\|\mathrm{d}_h^i\|\|_{*,i} \leq \frac{c_{P,h}}{\Delta t} \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i} + \nu c_{L4.6} \|\|\mathbf{e}_h^i\|\|_{*,i} + \frac{L^{1/2}}{\nu} i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i)^{1/2} + (1 + \beta) j_h^n(\mathrm{d}_h^i, \mathrm{d}_h^i)^{1/2}$$

$$+ \hat{c}\big(\Delta t + h^q + \frac{L^{1/2}h^k}{\nu}\big)(R_{c,1} + R_{I,1})(\boldsymbol{u}, p, \boldsymbol{f}).$$

Squaring this, using Young's inequality to remove the product terms multiplying with Δt^2 and summing over $i = 1, \dots, n$, we get

$$\Delta t^2 \sum_{i=1}^n \frac{\beta^2}{c_{P,h}^2 c_Y} \|\|\mathrm{d}_h^i\|\|_{*,i}^2 \leq \sum_{i=1}^n \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2 + \Delta t \sum_{i=1}^n \Big[\Delta t(\nu^2 c_{L4.6}^2) \|\|\mathbf{e}_h^i\|\|_{*,i}^2 + \frac{\Delta t L}{\nu^2} i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i) \Big]$$

$$+ \Delta t \sum_{i=1}^n \Delta t(1+\beta)^2 j_h^n(\mathrm{d}_h^i, \mathrm{d}_h^i) + \Delta t \sum_{i=1}^n \Delta t \hat{c}^2 \big(\Delta t^2 + h^{2q} + \frac{L h^{2k}}{\nu^2}\big) R(\boldsymbol{u}, p, \boldsymbol{f}), \quad (4.28)$$

where c_Y stems from the estimate $(\sum_{i=1}^n a_i)^2 \leq n \sum_{i=1}^n a_i^2$. We make the technical assumptions that $\Delta t \nu^2 c_{L4.6}^2 \leq \nu c_{L4.6}$ and $\Delta t(1 + \beta^2) \leq 2/\nu$. Note that since we are interested in the case of $\nu \ll 1$, these assumptions are not problematic and the inequalities are not sharp. For sufficiently small Δt, that is $\Delta t L/\nu^2 \leq 2L/\nu$, we can then bound the right-hand side of (4.28) with (4.27), which proves the error estimate.

The pressure discretisation error on the right-hand side of (4.27) can therefore be bounded by the other terms on the right-hand side of (4.27). This then gives us the estimate

$$\|\mathbf{e}_h^n\|_{\Omega_h^n}^2 + \sum_{i=1}^n \Big\{ \|\mathbf{e}_h^i - \mathbf{e}_h^{i-1}\|_{\Omega_h^i}^2 + \Delta t \big[\nu c_{L4.7} \|\|\mathbf{e}_h^i\|\|_{*,i}^2 + \frac{L}{\nu} i_h^i(\mathbf{e}_h^i, \mathbf{e}_h^i)\big] \Big\}$$

$$\leq \Delta t \sum_{i=1}^{n-1} \frac{2\bar{c}}{\nu} \|\mathbf{e}_h^i\|_{\Omega_h^k}^2 + \Delta t \sum_{i=1}^n 2c \Big[\Delta t^2 + h^{2q} + \frac{h^{2k}L}{\nu} + \frac{1}{\Delta t}(h^{2q} + \frac{h^{2k}}{\nu})\Big] R(\boldsymbol{u}, p, \boldsymbol{f}). \quad (4.29)$$

Applying Gronwall's Lemma then proves the result. $\qquad\square$

Remark 4.21. An alternative way to prove the velocity error estimate would be to follow the lines of the proof of Theorem 3.30 to deal with the discretisation error on the right-hand side of (4.26). That is, to use the inf-sup property of the Stokes part, which in turn then leaves the term $\frac{1}{\Delta t}\|\mathbb{E}^n - \mathbb{E}^{n-1}\|_{\Omega_h^n}$ on the right-hand side. As a result, we end up with the same sub-optimal scaling in Δt in our result. On the other hand, this illustrates that we do not need to expect that this estimate is sharp. $\qquad\blacktriangle$

Remark 4.22 (BDF2 time-discretisation). To extend the method to a BDF2 time-discretisation, we must ensure that $\Omega^n \subset \mathcal{O}_\delta(\Omega^{n-1}) \cap \mathcal{O}_\delta(\Omega^{n-2})$ and $\Omega_h^n \subset \mathcal{O}_{\delta_h}(\Omega_h^{n-1}) \cap \mathcal{O}_{\delta_h}(\Omega_h^{n-2})$. This can be achieved by increasing the strip-width to $\delta = 2c_\delta \boldsymbol{w}_\infty^n \Delta t$. Then one can use the BDF2 polarisation identity (3.42) and adapt the time-derivative estimates as described in Remark 3.32. Finally, the Gronwall arguments must be adapted to the corresponding version for higher-order BDF schemes. $\qquad\blacktriangle$
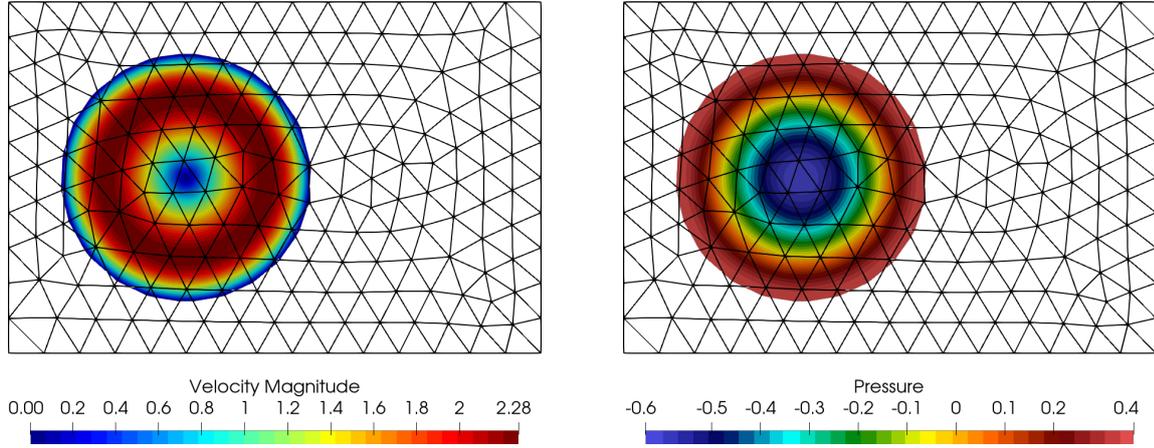
Figure 4.2: Initial condition for the numerical examples for the Eulerian time-stepping scheme for the transient Stokes problem on a moving domain. The background domain is $\widetilde{\Omega} = (-1, -2) \times (1, 1)$, $h_{\mathsf{max}} = 0.2$ and the solution is drawn with respect to the $\mathbb{P}^1$ level set function. Left: Velocity magnitude. Right: Pressure field.

Remark 4.23 (BDF2 initialisation). A fundamental question for a BDF2 time-discretisation is a sufficiently accurate initialisation $\boldsymbol{u}_h^1$. In the context of ODE's, a single BDF1 step gives a sufficiently accurate realisation of $\boldsymbol{u}_h^1$ [HNW93]. However, looking at Theorem 4.20, we see that after one BDF1 step, we only get a Δt^3 term in front of the $R(\boldsymbol{u}, p, \boldsymbol{f})$ term, which leads to a suboptimal $\ell^2(\mathcal{H}^1)$ error-estimate. Therefore, to initialise the BDF2 scheme with sufficient accuracy, we use multiple BDF1 steps with the smaller time-step $\tilde{\Delta} t \leq \Delta t^{4/3}$. We have found this to be necessary for our numerical experiments to obtain second-order convergence in the time-step. ▲

4.4 Numerical Examples

We consider several numerical tests to evaluate the Eulerian time-stepping method's performance in this chapter in practice. To compute errors and estimate the resulting order of convergence, we take a forcing term $\boldsymbol{f}^n$, such that the solution to the problem is known. This is based on the analytical solution we constructed in subsection 3.2.6 for the Stokes problem on a stationary domain.

We consider the moving circle $\Omega(t) = \{\boldsymbol{x} \in \mathbb{R}^2 \mid (\boldsymbol{x}_1 - t)^2 + \boldsymbol{x}_2^2 < {}^1\!/2\}$ and on this the velocity and pressure pair

$$\boldsymbol{u}(t) = \begin{pmatrix} 2\pi \boldsymbol{x}_2 \cos(\pi((\boldsymbol{x}_1 - t)^2 + \boldsymbol{x}_2^2)) \\ -2\pi \boldsymbol{x}_1 \cos(\pi((\boldsymbol{x}_1 - t)^2 + \boldsymbol{x}_2^2)) \end{pmatrix} \qquad \text{and} \qquad p(t) = \sin(\pi((\boldsymbol{x}_1 - t)^2 + \boldsymbol{x}_2^2)) - {}^2\!/\pi.$$

The velocity field $\boldsymbol{u}$ is then divergence-free and fulfils homogeneous Dirichlet boundary conditions on $\Gamma(t)$, while the pressure p is in $\mathcal{L}_0^2(\Omega(t))$. The solution at $t = 0$ is shown in Figure 4.2. The forcing term is then constructed as $\boldsymbol{f}(t) \coloneqq \partial_t \boldsymbol{u}(t) - \nu \Delta \boldsymbol{u}(t) + \nabla p(t)$.

Set-Up

For our computations, we consider the following general set-up. We take the time interval $[0, 1]$ and the background domain $\widetilde{\Omega} = (-1, -2) \times (1, 1)$, so that $\Omega(t) \subset \widetilde{\Omega}$ for all $t \in [0, 1]$. The maximal interface speed within the time interface is then $\boldsymbol{w}_\infty^{\boldsymbol{n}} = 1$. The strip-width parameter is chosen as $c_\delta = 1$, and unless otherwise stated, the Nitsche parameter is chosen as $\sigma = 40k^2$. As the ghost-penalty stabilisation plays a different role compared to the stationary domain case, we will determine a good choice for the ghost-penalty parameter below. Unless otherwise stated, we shall consider $k = 2$, i.e., the lowest order

$\nu \downarrow \backslash \gamma_{gp} \rightarrow$	0.1	1	10	100	1000
1	$4.02 \cdot 10^{-2}$	$4.15 \cdot 10^{-2}$	$4.45 \cdot 10^{-2}$	$6.29 \cdot 10^{-2}$	$1.64 \cdot 10^{-1}$
0.1	$1.81 \cdot 10^{-1}$	$1.88 \cdot 10^{-1}$	$2.32 \cdot 10^{-1}$	$4.51 \cdot 10^{-1}$	$1.17 \cdot 10^{0}$
0.01	$3.67 \cdot 10^{-1}$	$4.46 \cdot 10^{-1}$	$7.77 \cdot 10^{-1}$	$1.39 \cdot 10^{0}$	$1.58 \cdot 10^{0}$
0.001	$4.22 \cdot 10^{-1}$	$6.73 \cdot 10^{-1}$	$1.12 \cdot 10^{0}$	$1.26 \cdot 10^{0}$	$1.28 \cdot 10^{0}$
0.0001	$2.05 \cdot 10^{0}$	$3.60 \cdot 10^{0}$	$4.14 \cdot 10^{0}$	$4.32 \cdot 10^{0}$	$4.34 \cdot 10^{0}$

$\nu \downarrow \backslash \gamma_{gp} \rightarrow$	0.1	1	10	100	1000
1	$2.66 \cdot 10^{-2}$	$2.67 \cdot 10^{-2}$	$2.67 \cdot 10^{-2}$	$2.67 \cdot 10^{-2}$	$2.69 \cdot 10^{-2}$
0.1	$1.69 \cdot 10^{-1}$	$1.69 \cdot 10^{-1}$	$1.69 \cdot 10^{-1}$	$1.71 \cdot 10^{-1}$	$1.87 \cdot 10^{-1}$
0.01	$3.30 \cdot 10^{-1}$	$3.31 \cdot 10^{-1}$	$3.37 \cdot 10^{-1}$	$3.82 \cdot 10^{-1}$	$6.51 \cdot 10^{-1}$
0.001	$3.67 \cdot 10^{-1}$	$3.72 \cdot 10^{-1}$	$4.24 \cdot 10^{-1}$	$6.71 \cdot 10^{-1}$	$8.92 \cdot 10^{-1}$
0.0001	$3.49 \cdot 10^{-1}$	$3.47 \cdot 10^{-1}$	$4.36 \cdot 10^{-1}$	$5.60 \cdot 10^{-1}$	$5.65 \cdot 10^{-1}$

$\nu \downarrow \backslash \gamma_{gp} \rightarrow$	0.1	1	10	100	1000
1	$2.60 \cdot 10^{-2}$	$2.60 \cdot 10^{-2}$	$2.60 \cdot 10^{-2}$	$2.61 \cdot 10^{-2}$	$2.60 \cdot 10^{-2}$
0.1	$1.68 \cdot 10^{-1}$	$1.68 \cdot 10^{-1}$	$1.68 \cdot 10^{-1}$	$1.68 \cdot 10^{-1}$	$1.70 \cdot 10^{-1}$
0.01	$3.29 \cdot 10^{-1}$	$3.29 \cdot 10^{-1}$	$3.30 \cdot 10^{-1}$	$3.36 \cdot 10^{-1}$	$3.97 \cdot 10^{-1}$
0.001	$3.66 \cdot 10^{-1}$	$3.65 \cdot 10^{-1}$	$3.75 \cdot 10^{-1}$	$4.51 \cdot 10^{-1}$	$6.95 \cdot 10^{-1}$
0.0001	$3.63 \cdot 10^{-1}$	$3.66 \cdot 10^{-1}$	$3.90 \cdot 10^{-1}$	$4.80 \cdot 10^{-1}$	$5.40 \cdot 10^{-1}$

Table 4.1: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$ velocity error for the BDF1 method over a range of viscosities and ghost-penalty parameters with $\Delta t = 1/20$. **Top:** $h_{\max} = 0.1$ and $L = 1$. **Middle:** $h_{\max} = 0.025$ and $L = 2$. **Bottom:** $h_{\max} = 0.0125$ and $L = 4$.

Taylor-Hood element. However, since the geometry is approximated by a $\mathbb{P}^1$ level set function, i.e., $q = 1$, we cannot expect optimal spatial convergence for the velocity in the $\ell^2(\mathcal{L})^2$-norm.

To quantify the computational results, we will consider the following discrete space-time errors:

$$\|\boldsymbol{u}_h - \boldsymbol{u}\|_{\ell^2(\mathcal{L}^2)}^2 := \Delta t \sum_{k=1}^{n} \|\boldsymbol{u}_h - \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^2(\Omega_h^k)}^2 \qquad \|\boldsymbol{u}_h - \boldsymbol{u}\|_{\ell^2(\boldsymbol{\mathcal{H}}^1)}^2 := \Delta t \sum_{k=1}^{n} \|\nabla \boldsymbol{u}_h - \nabla \boldsymbol{u}\|_{\boldsymbol{\mathcal{L}}^2(\Omega_h^k)}^2$$

$$\|p_h - p\|_{\ell^2(\mathcal{L}^2)}^2 := \Delta t \sum_{k=1}^{n} \|p_h - p\|_{\mathcal{L}^2(\Omega_h^k)}^2.$$

4.4.1 Stability

As a first investigation, we look into the robustness of the method with respect to viscosity ν, the dependence on the ghost-penalty parameter γ_{gp} and the stability for larger extension strips. To this end, we consider the time-step $\Delta t = 1/20$. The viscosity is taken as $\nu \in \{1, 10^{-1}, \ldots, 10^{-4}\}$ and the ghost-penalty parameter is chosen as $\gamma_{gp} \in \{0.1, 1, \ldots, 10^3\}$. In obtain different strip widths, in the sense of the number of elements $L = \lceil \delta_h / h_{\max} \rceil$, we consider $h_{\max} = 0.1, 0.025, 0.025$ resulting in $L = 1, 2, 4$, respectively.

The resulting $\ell^2(\mathcal{L}^2)$-velocity errors can be seen in Table 4.1. We see that overall, the method is relatively robust with respect to over stabilisation. Furthermore, we see that the method is stable for the different strip-widths considered here. The smaller errors for the finer meshes can be attributed to a dominating spatial error. With respect to the viscosity, we note that by decreasing the viscosity by a factor of 10^4, the largest increase in the error is only a factor of 50 on the coarsest mesh and even smaller on the finer meshes. Finally, we note that the method is stable, even for viscosities $\nu \ll \Delta t$, contrary to our assumption in the proof of Theorem 4.20.

Remark 4.24. In our experience, we have found that if the pressure stabilisation is applied on the same set of facets as the velocity extension ghost-penalty operator and with the same parameter scaling $\gamma_{gp,p} = L\gamma_{gp}$, then the results are qualitatively the same, but with a larger error constant. ▲

4.4.2 Convergence

To investigate the asymptotic convergence properties of our method, we compute the problem over a series of uniform mesh refinements (bisections) of the mesh and the time-step. With the initial time-step $\Delta t_0 = 0.1$ and the initial mesh diameter $h_0 = 0.2$, the time-step and mesh size are then given by $\Delta t = \Delta t_0 \cdot 2^{-L_t}$ and $h_{\max} = h_0 \cdot 2^{-L_x}$, with the number of time-step and mesh refinements L_t, L_x, respectively. In order evaluate the convergence behaviour, we compute the *experimental order of convergence* as $eoc_{par} = (\log(err(L_{par} - 1)) - \log(err(L_{par})))/\log(2)$. For the spatial and temporal error order of convergence, we take the other parameter on the most refined level. Furthermore, since our analysis predicts a corruption of the spatial error by a factor Δt^{-1}, we also compute the diagonal order convergence eoc_{xt}, comparing the results after both a level of spatial and temporal refinement.

For the computations, we choose $\nu = 0.01$. Taking the previous results into account, we take $\gamma_{gp} = 1$. We then consider four levels of mesh refinement and a total of eight refinement levels for the time-step.

The results can be seen in Table 4.2. Here we observe the expected linear convergence in time for the velocity and pressure in all the considered norms. With respect to the spatial convergence (eoc_x), we see a drop in the convergence rate on the finest mesh, which we attribute to a dominating temporal error. However, the rates are also higher than expected, where the spatial error is dominant. We attribute this to an interplay between the geometry error and the $1/\nu$ scaled consistency error from the ghost-penalties. Nevertheless, we see at least second-order convergence under spatial refinement. While this is only optimal for the $\ell^2(\mathcal{H}^1)$-velocity error and the $\ell^2(\mathcal{L}^2)$-pressure error, this is the best we can expect for a piecewise linear geometry approximation.

To check whether the factor $(h^{2q} + h^{2m})/\Delta t$ is observable, we consider joint refinement of both time and space with $L_t = L_x + 4$ for which the theory predicts a loss half an order of convergence. However, the results in Table 4.2 show that $eox_{tx} \approx eox_x$. This suggests that this part of the analysis is indeed not sharp, as expected and discussed above.

For the $\ell^2(\mathcal{L}^2)$-pressure error, we observe that the experimental order of convergence in space is higher than expected. This suggests that the velocity error on the right-hand side of the pressure estimate is the dominating term here.

4.4.3 Error Development

As we have seen in the previous section that Theorem 4.20 is not sharp with respect to the powers of h and Δt, we investigate whether the exponential factor in the error estimate is too crude as well. Therefore, we consider the identical set-up as above for $L_t = L_x = 0, \ldots, 4$ and compute the space-time errors for the velocity and pressure over time.

The results for these computations can be seen in Figure 4.3. Here we see that after an initial fast increase in the errors, the errors do not accumulate significantly. This suggests that the exponential factor in our error estimate, due to the application of a Gronwall lemma, is not necessarily sharp. The slow growth in the error over time corresponds to a similar observation in [BFM19].

4.4.4 Extension to Higher-Order in Space and Time

BDF2 Time-Discretisation

As we have discussed the extension of our Eulerian time-stepping method to a BDF2 approximation of the time-derivative, we shall also investigate the convergence properties of this method. To this end, we

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$1.29 \cdot 10^0$	$7.41 \cdot 10^{-1}$	$6.60 \cdot 10^{-1}$	$6.49 \cdot 10^{-1}$	$6.54 \cdot 10^{-1}$	–
1	$1.02 \cdot 10^0$	$4.41 \cdot 10^{-1}$	$3.43 \cdot 10^{-1}$	$3.31 \cdot 10^{-1}$	$3.29 \cdot 10^{-1}$	0.99
2	$8.24 \cdot 10^{-1}$	$2.76 \cdot 10^{-1}$	$1.82 \cdot 10^{-1}$	$1.68 \cdot 10^{-1}$	$1.66 \cdot 10^{-1}$	0.99
3	$7.35 \cdot 10^{-1}$	$1.95 \cdot 10^{-1}$	$9.87 \cdot 10^{-2}$	$8.56 \cdot 10^{-2}$	$8.37 \cdot 10^{-2}$	0.99
4	$6.91 \cdot 10^{-1}$	$1.56 \cdot 10^{-1}$	$5.71 \cdot 10^{-2}$	$4.38 \cdot 10^{-2}$	$4.21 \cdot 10^{-2}$	0.99
5	$6.69 \cdot 10^{-1}$	$1.39 \cdot 10^{-1}$	$3.71 \cdot 10^{-2}$	$2.28 \cdot 10^{-2}$	$2.12 \cdot 10^{-2}$	0.99
6	$6.58 \cdot 10^{-1}$	$1.30 \cdot 10^{-1}$	$2.77 \cdot 10^{-2}$	$1.24 \cdot 10^{-2}$	$1.07 \cdot 10^{-2}$	0.98
7	$6.53 \cdot 10^{-1}$	$1.26 \cdot 10^{-1}$	$2.35 \cdot 10^{-2}$	$7.20 \cdot 10^{-3}$	$5.49 \cdot 10^{-3}$	0.97
8	$6.50 \cdot 10^{-1}$	$1.24 \cdot 10^{-1}$	$2.15 \cdot 10^{-2}$	$4.70 \cdot 10^{-3}$	$2.86 \cdot 10^{-3}$	0.94
eoc_x	–	2.39	2.53	2.19	0.72	
eoc_{xt}	–	2.31	2.32	1.95	1.33	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$8.92 \cdot 10^0$	$5.79 \cdot 10^0$	$5.42 \cdot 10^0$	$5.60 \cdot 10^0$	$6.28 \cdot 10^0$	–
1	$7.89 \cdot 10^0$	$3.97 \cdot 10^0$	$3.20 \cdot 10^0$	$3.24 \cdot 10^0$	$3.37 \cdot 10^0$	0.90
2	$6.90 \cdot 10^0$	$2.84 \cdot 10^0$	$1.89 \cdot 10^0$	$1.84 \cdot 10^0$	$1.89 \cdot 10^0$	0.83
3	$6.48 \cdot 10^0$	$2.32 \cdot 10^0$	$1.15 \cdot 10^0$	$1.01 \cdot 10^0$	$1.03 \cdot 10^0$	0.87
4	$6.27 \cdot 10^0$	$2.09 \cdot 10^0$	$7.59 \cdot 10^{-1}$	$5.52 \cdot 10^{-1}$	$5.50 \cdot 10^{-1}$	0.91
5	$6.17 \cdot 10^0$	$2.00 \cdot 10^0$	$5.92 \cdot 10^{-1}$	$3.03 \cdot 10^{-1}$	$2.87 \cdot 10^{-1}$	0.94
6	$6.12 \cdot 10^0$	$1.96 \cdot 10^0$	$5.28 \cdot 10^{-1}$	$1.79 \cdot 10^{-1}$	$1.48 \cdot 10^{-1}$	0.96
7	$6.10 \cdot 10^0$	$1.94 \cdot 10^0$	$5.04 \cdot 10^{-1}$	$1.24 \cdot 10^{-1}$	$7.58 \cdot 10^{-2}$	0.96
8	$6.08 \cdot 10^0$	$1.94 \cdot 10^0$	$4.94 \cdot 10^{-1}$	$1.02 \cdot 10^{-1}$	$3.99 \cdot 10^{-2}$	0.93
eoc_x	–	1.65	1.97	2.27	1.36	
eoc_{xt}	–	1.65	1.92	2.09	1.63	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$7.26 \cdot 10^{-1}$	$4.43 \cdot 10^{-1}$	$4.13 \cdot 10^{-1}$	$4.29 \cdot 10^{-1}$	$4.25 \cdot 10^{-1}$	–
1	$5.93 \cdot 10^{-1}$	$2.65 \cdot 10^{-1}$	$2.05 \cdot 10^{-1}$	$1.99 \cdot 10^{-1}$	$2.01 \cdot 10^{-1}$	1.08
2	$4.85 \cdot 10^{-1}$	$1.67 \cdot 10^{-1}$	$1.08 \cdot 10^{-1}$	$9.79 \cdot 10^{-2}$	$9.68 \cdot 10^{-2}$	1.06
3	$4.41 \cdot 10^{-1}$	$1.19 \cdot 10^{-1}$	$5.93 \cdot 10^{-2}$	$4.96 \cdot 10^{-2}$	$4.80 \cdot 10^{-2}$	1.01
4	$4.27 \cdot 10^{-1}$	$9.66 \cdot 10^{-2}$	$3.51 \cdot 10^{-2}$	$2.54 \cdot 10^{-2}$	$2.41 \cdot 10^{-2}$	1.00
5	$4.31 \cdot 10^{-1}$	$8.67 \cdot 10^{-2}$	$2.34 \cdot 10^{-2}$	$1.34 \cdot 10^{-2}$	$1.21 \cdot 10^{-2}$	0.99
6	$4.54 \cdot 10^{-1}$	$8.30 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$	$7.44 \cdot 10^{-3}$	$6.14 \cdot 10^{-3}$	0.98
7	$5.01 \cdot 10^{-1}$	$8.34 \cdot 10^{-2}$	$1.54 \cdot 10^{-2}$	$4.49 \cdot 10^{-3}$	$3.16 \cdot 10^{-3}$	0.96
8	$5.83 \cdot 10^{-1}$	$8.70 \cdot 10^{-2}$	$1.46 \cdot 10^{-2}$	$3.06 \cdot 10^{-3}$	$1.68 \cdot 10^{-3}$	0.92
eoc_x	–	2.75	2.57	2.25	0.87	
eoc_{xt}	–	2.30	2.28	1.99	1.42	

Table 4.2: Mesh and time-step convergence for the BDF1 method with $\nu = 10^{-2}$. Top: $\ell^2(\boldsymbol{\mathcal{L}}^2(\Omega_h(t)))$-velocity error. Middle: $\ell^2(\boldsymbol{\mathcal{H}}^1(\Omega_h(t)))$ velocity error. Bottom: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$-pressure error.
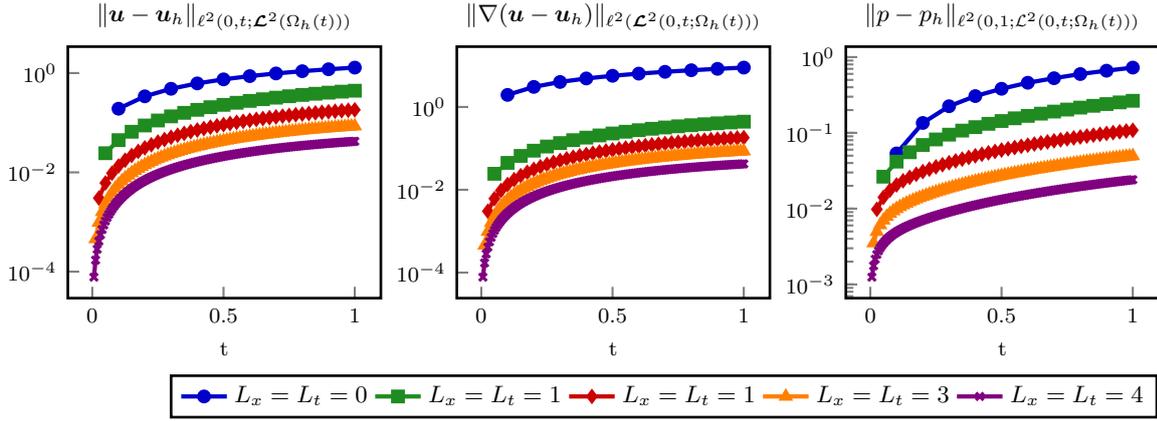
Figure 4.3: Space-time velocity and pressure errors over time for combined mesh and time-step refinement resulting from the unfitted Eulerian BDF1 time-stepping method with $\nu = 0.01$.

increase the strip-width to $\delta_h = 2c_\delta \boldsymbol{w}^{\boldsymbol{n}}_\infty \Delta t$. The remaining discretisation parameters are kept identical. However, we consider an additional spatial refinement level and one temporal refinement level less.

The resulting errors, together with the experimental orders of convergence, are presented in Table 4.3. With respect to the temporal convergence, we see that second-order convergence, i.e., $eoc_t \approx 2$. In fact, the spatial error starts to dominate on the two highest refinement levels. We also note that there appear to be some stability issues for very large time-steps on the finest meshes. In these computations, the strip width was $L = 16, 32$. However, these are outside of the viscosity/time-step range covered by our error analysis.

With respect to space, we see that the spatial order of convergence has increased in comparison to the BDF1 case. Indeed, we even see $eoc_x \approx 3$ for the $\ell^2(\boldsymbol{\mathcal{L}}^2)$-velocity error. This can be attributed to the fact that the temporal error is insignificant compared to the spatial error here, and that the geometry error contribution in the spatial error also does not dominate. Finally, we see that $eoc_{xt} \approx eoc_x$, i.e., a negative impact of temporal refinement on the spatial error was not observed.

Geometry Handling

In the previous computations in this section, we have used the standard CutFEM approach of a piecewise linear level set approximation to generate the necessary quadrature rules on cut elements. Unfortunately, this introduces a geometry approximation error of order $\mathcal{O}(h^2)$, and as we have seen in some of the above results, this leads to suboptimal spatial convergence rates. We now consider two approaches in order to try and recover optimal-order spatial error convergence.

Quadrature Through Subdivisions A straightforward and effective – however also very inefficient – method to essential "hide" the geometry approximation error and to reveal the underling approximation error is to approximate the boundary based on a piecewise linear level set after s subdivisions of cut elements. As a result, the geometry approximation error is then pushed down to $\mathcal{O}\left(\left(\frac{h}{2^s}\right)\right)$. The drawback of this approach is that for every decrease in h, we must increase s in order to balance the errors. As a consequence, the resulting quadrature is asymptotically infeasible.

In order to balance the geometry and approximation errors, we choose $s = \mathcal{O}(\log_2(1/h))$ such that we achieve a geometry error of order $\mathcal{O}(h^4)$. This is sufficient to realise optimal mesh convergence for both TH$_2$ and TH$_3$ elements. We emphasise that this is only to investigate the underlying spatial approximation

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	5	eoc_t
0	$1.39 \cdot 10^0$	$5.66 \cdot 10^{-1}$	$2.42 \cdot 10^{-1}$	$9.98 \cdot 10^{-2}$	$1.24 \cdot 10^{-1}$	$1.67 \cdot 10^0$	–
1	$1.10 \cdot 10^0$	$2.97 \cdot 10^{-1}$	$1.07 \cdot 10^{-1}$	$3.47 \cdot 10^{-2}$	$1.85 \cdot 10^{-2}$	$2.40 \cdot 10^{-1}$	2.80
2	$9.11 \cdot 10^{-1}$	$2.13 \cdot 10^{-1}$	$4.90 \cdot 10^{-2}$	$1.34 \cdot 10^{-2}$	$4.61 \cdot 10^{-3}$	$4.01 \cdot 10^{-3}$	5.90
3	$7.63 \cdot 10^{-1}$	$1.60 \cdot 10^{-1}$	$3.35 \cdot 10^{-2}$	$5.78 \cdot 10^{-3}$	$1.44 \cdot 10^{-3}$	$9.36 \cdot 10^{-4}$	2.10
4	$7.01 \cdot 10^{-1}$	$1.40 \cdot 10^{-1}$	$2.59 \cdot 10^{-2}$	$4.17 \cdot 10^{-3}$	$5.91 \cdot 10^{-4}$	$2.35 \cdot 10^{-4}$	1.99
5	$6.73 \cdot 10^{-1}$	$1.31 \cdot 10^{-1}$	$2.24 \cdot 10^{-2}$	$3.18 \cdot 10^{-3}$	$4.52 \cdot 10^{-4}$	$8.33 \cdot 10^{-5}$	1.50
6	$6.60 \cdot 10^{-1}$	$1.27 \cdot 10^{-1}$	$2.10 \cdot 10^{-2}$	$2.81 \cdot 10^{-3}$	$3.72 \cdot 10^{-4}$	$6.44 \cdot 10^{-5}$	0.37
7	$6.54 \cdot 10^{-1}$	$1.25 \cdot 10^{-1}$	$2.03 \cdot 10^{-2}$	$2.66 \cdot 10^{-3}$	$3.42 \cdot 10^{-4}$	$6.02 \cdot 10^{-5}$	0.10
eoc_x	–	2.39	2.61	2.94	2.96	2.51	
eoc_{xt}	–	2.51	2.63	3.02	3.10	2.63	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	5	eoc_t
0	$9.22 \cdot 10^0$	$5.00 \cdot 10^0$	$2.73 \cdot 10^0$	$1.26 \cdot 10^0$	$2.60 \cdot 10^0$	$4.30 \cdot 10^1$	–
1	$8.23 \cdot 10^0$	$3.35 \cdot 10^0$	$1.56 \cdot 10^0$	$6.01 \cdot 10^{-1}$	$2.59 \cdot 10^{-1}$	$1.05 \cdot 10^1$	2.03
2	$7.55 \cdot 10^0$	$2.74 \cdot 10^0$	$8.95 \cdot 10^{-1}$	$3.07 \cdot 10^{-1}$	$9.55 \cdot 10^{-2}$	$5.20 \cdot 10^{-2}$	7.66
3	$6.72 \cdot 10^0$	$2.26 \cdot 10^0$	$6.99 \cdot 10^{-1}$	$1.63 \cdot 10^{-1}$	$4.50 \cdot 10^{-2}$	$1.50 \cdot 10^{-2}$	1.79
4	$6.37 \cdot 10^0$	$2.08 \cdot 10^0$	$5.90 \cdot 10^{-1}$	$1.33 \cdot 10^{-1}$	$2.25 \cdot 10^{-2}$	$5.86 \cdot 10^{-3}$	1.36
5	$6.22 \cdot 10^0$	$2.00 \cdot 10^0$	$5.30 \cdot 10^{-1}$	$1.07 \cdot 10^{-1}$	$1.89 \cdot 10^{-2}$	$2.74 \cdot 10^{-3}$	1.10
6	$6.14 \cdot 10^0$	$1.96 \cdot 10^0$	$5.07 \cdot 10^{-1}$	$9.72 \cdot 10^{-2}$	$1.51 \cdot 10^{-2}$	$2.32 \cdot 10^{-3}$	0.24
7	$6.11 \cdot 10^0$	$1.95 \cdot 10^0$	$4.97 \cdot 10^{-1}$	$9.37 \cdot 10^{-2}$	$1.35 \cdot 10^{-2}$	$1.90 \cdot 10^{-3}$	0.29
eoc_x	–	1.65	1.97	2.41	2.79	2.83	
eoc_{xt}	–	1.74	1.94	2.46	2.83	2.99	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	5	eoc_t
0	$8.20 \cdot 10^{-1}$	$3.81 \cdot 10^{-1}$	$2.24 \cdot 10^{-1}$	$1.50 \cdot 10^{-1}$	$3.48 \cdot 10^{-1}$	$4.00 \cdot 10^0$	–
1	$6.60 \cdot 10^{-1}$	$1.57 \cdot 10^{-1}$	$6.77 \cdot 10^{-2}$	$4.48 \cdot 10^{-2}$	$4.06 \cdot 10^{-2}$	$9.57 \cdot 10^{-1}$	2.06
2	$5.60 \cdot 10^{-1}$	$1.22 \cdot 10^{-1}$	$2.35 \cdot 10^{-2}$	$1.08 \cdot 10^{-2}$	$1.15 \cdot 10^{-2}$	$1.05 \cdot 10^{-2}$	6.51
3	$4.76 \cdot 10^{-1}$	$9.67 \cdot 10^{-2}$	$1.87 \cdot 10^{-2}$	$2.54 \cdot 10^{-3}$	$2.65 \cdot 10^{-3}$	$2.83 \cdot 10^{-3}$	1.89
4	$4.59 \cdot 10^{-1}$	$8.73 \cdot 10^{-2}$	$1.56 \cdot 10^{-2}$	$2.29 \cdot 10^{-3}$	$5.09 \cdot 10^{-4}$	$6.81 \cdot 10^{-4}$	2.05
5	$4.79 \cdot 10^{-1}$	$8.44 \cdot 10^{-2}$	$1.42 \cdot 10^{-2}$	$2.01 \cdot 10^{-3}$	$2.59 \cdot 10^{-4}$	$1.46 \cdot 10^{-4}$	2.22
6	$5.31 \cdot 10^{-1}$	$8.54 \cdot 10^{-2}$	$1.39 \cdot 10^{-2}$	$1.87 \cdot 10^{-3}$	$2.54 \cdot 10^{-4}$	$4.29 \cdot 10^{-5}$	1.76
7	$6.28 \cdot 10^{-1}$	$8.99 \cdot 10^{-2}$	$1.41 \cdot 10^{-2}$	$1.82 \cdot 10^{-3}$	$2.46 \cdot 10^{-4}$	$4.10 \cdot 10^{-5}$	0.07
eoc_x	–	2.80	2.67	2.95	2.89	2.58	
eoc_{xt}	–	2.53	2.63	2.96	2.99	2.63	

Table 4.3: Mesh and time-step convergence for the BDF2 method with $\nu = 10^{-2}$. Top: $\ell^2(\boldsymbol{\mathcal{L}}^2(\Omega_h(t)))$-velocity error. Middle: $\ell^2(\boldsymbol{\mathcal{H}}^1(\Omega_h(t)))$ velocity error. Bottom: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$-pressure error.
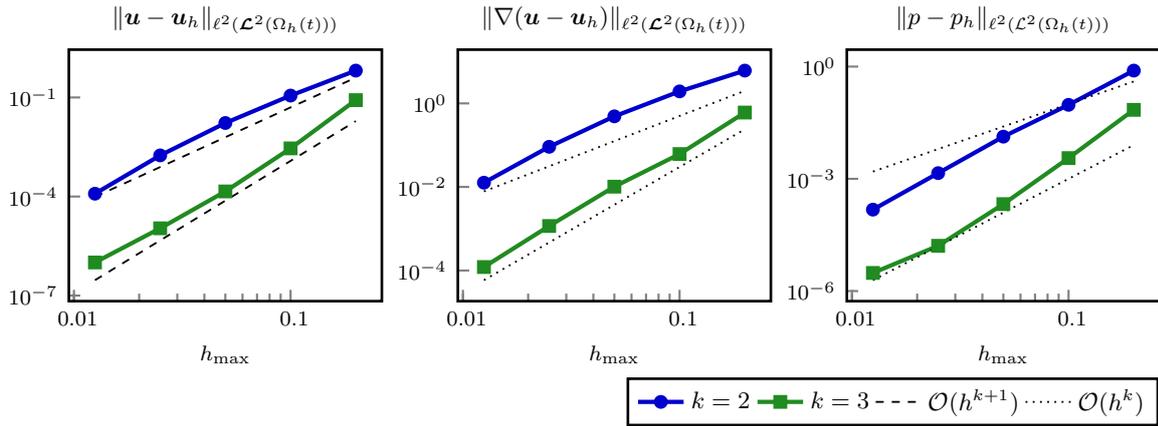
Figure 4.4: Mesh convergence for the unfitted Eulerian time-stepping method with $\texttt{TH}_k$ elements and using subdivisions on cut elements to hide the geometry approximation error.

error and cannot be seen as a general solution to the problem of approximating curved boundaries in unfitted finite element methods.

Using the set-up with the BDF2 time-discretisation as our starting point, we take the time-step $\Delta t = 0.1 \cdot 2^{-8}$ such that the spatial error is the dominant factor. We then consider $\texttt{TH}_2$ and $\texttt{TH}_3$ elements in conjunction with this subdivision strategy.

The resulting errors over a series of mesh refinements can be seen in Figure 4.4. We see here that we have recovered the optimal order spatial convergence for the velocity error in both the $\ell^2(\mathcal{L}^2)$- and $\ell^2(\mathcal{H}^1)$-norms. For the pressure, we see higher convergence rates than expected. As before, we attribute this to the velocity error dominating over the pressure error.

Parametric Mapping A better approach to deal with the geometry approximation, is to use the isoparametric approach we considered for fixed domain problems in chapter 3. As noted in subsection 4.3.4, the problem here is the analysis of an efficient and accurate transfer operator to evaluate function defined with respect to a differently deformed mesh. Since such an operator is implemented in $\texttt{ngsxfem}$, we shall consider this approach in practice.

To this end, we consider the same set-up as before, together with $\texttt{TH}_2^{\text{iso}}$ and $\texttt{TH}_3^{\text{iso}}$ elements. The results can be seen in Table 4.4 for $k = 2$ and Table 4.5 for $k = 3$. Comparing the results to those in Table 4.3 without the parametric mapping, we see that the spatial error has been reduced here and that the convergence rate with respect to the time-step has not been affected. Looking at the higher-order results in Table 4.5, we see that while the $\ell^2(\mathcal{L}^2)$-velocity error appears to be missing half an order, the $\ell^2(\mathcal{H}^1)$-velocity error converges with order 3 and the pressure error converges again more quickly than expected. This can again be explained by the observation that it is dominated by the velocity error. With respect to the time-step, we again see second-order convergence. However, there appears to be a stability issue for the largest time-step on the finest mesh. We do not see this as problematic, as this computation would always have a very unbalanced temporal and spatial error. We note that it was necessary to consider an additional level of temporal refinement to see the spatial error dominate.

4.4.5 Summary

Our numerical examples show that our method is stable over a wide range of viscosities and ghost-penalty parameters. Concerning the asymptotic convergence, we did not see any dependence on negative powers of Δt from Theorem 4.20 in practice, and the method converges optimally in time. This suggests an estimate

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$1.28 \cdot 10^0$	$5.66 \cdot 10^{-1}$	$2.40 \cdot 10^{-1}$	$9.94 \cdot 10^{-2}$	$1.42 \cdot 10^{-1}$	–
1	$8.70 \cdot 10^{-1}$	$2.58 \cdot 10^{-1}$	$1.05 \cdot 10^{-1}$	$3.38 \cdot 10^{-2}$	$1.84 \cdot 10^{-2}$	2.95
2	$6.46 \cdot 10^{-1}$	$1.63 \cdot 10^{-1}$	$3.83 \cdot 10^{-2}$	$1.25 \cdot 10^{-2}$	$4.46 \cdot 10^{-3}$	2.05
3	$5.01 \cdot 10^{-1}$	$1.04 \cdot 10^{-1}$	$2.44 \cdot 10^{-2}$	$4.06 \cdot 10^{-3}$	$1.26 \cdot 10^{-3}$	1.82
4	$4.40 \cdot 10^{-1}$	$8.24 \cdot 10^{-2}$	$1.59 \cdot 10^{-2}$	$2.63 \cdot 10^{-3}$	$3.80 \cdot 10^{-4}$	1.73
5	$4.10 \cdot 10^{-1}$	$7.23 \cdot 10^{-2}$	$1.19 \cdot 10^{-2}$	$1.61 \cdot 10^{-3}$	$2.19 \cdot 10^{-4}$	0.79
6	$3.94 \cdot 10^{-1}$	$6.72 \cdot 10^{-2}$	$1.02 \cdot 10^{-2}$	$1.17 \cdot 10^{-3}$	$1.26 \cdot 10^{-4}$	0.80
7	$3.85 \cdot 10^{-1}$	$6.45 \cdot 10^{-2}$	$9.37 \cdot 10^{-3}$	$9.93 \cdot 10^{-4}$	$8.67 \cdot 10^{-5}$	0.54
eoc_x	–	2.58	2.78	3.24	3.52	
eoc_{xt}	–	2.60	2.80	3.34	3.76	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$8.91 \cdot 10^0$	$5.11 \cdot 10^0$	$2.75 \cdot 10^0$	$1.26 \cdot 10^0$	$3.07 \cdot 10^0$	–
1	$7.23 \cdot 10^0$	$3.12 \cdot 10^0$	$1.60 \cdot 10^0$	$6.00 \cdot 10^{-1}$	$2.60 \cdot 10^{-1}$	3.56
2	$6.19 \cdot 10^0$	$2.36 \cdot 10^0$	$7.65 \cdot 10^{-1}$	$3.06 \cdot 10^{-1}$	$9.36 \cdot 10^{-2}$	1.48
3	$5.16 \cdot 10^0$	$1.72 \cdot 10^0$	$5.73 \cdot 10^{-1}$	$1.30 \cdot 10^{-1}$	$4.26 \cdot 10^{-2}$	1.14
4	$4.71 \cdot 10^0$	$1.46 \cdot 10^0$	$4.31 \cdot 10^{-1}$	$1.01 \cdot 10^{-1}$	$1.83 \cdot 10^{-2}$	1.22
5	$4.47 \cdot 10^0$	$1.32 \cdot 10^0$	$3.41 \cdot 10^{-1}$	$7.47 \cdot 10^{-2}$	$1.51 \cdot 10^{-2}$	0.28
6	$4.33 \cdot 10^0$	$1.24 \cdot 10^0$	$3.00 \cdot 10^{-1}$	$5.94 \cdot 10^{-2}$	$1.16 \cdot 10^{-2}$	0.38
7	$4.24 \cdot 10^0$	$1.20 \cdot 10^0$	$2.79 \cdot 10^{-1}$	$5.19 \cdot 10^{-2}$	$9.09 \cdot 10^{-3}$	0.35
eoc_x	–	1.82	2.11	2.43	2.51	
eoc_{xt}	–	1.82	2.10	2.52	2.71	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
0	$7.21 \cdot 10^{-1}$	$4.07 \cdot 10^{-1}$	$2.21 \cdot 10^{-1}$	$1.54 \cdot 10^{-1}$	$4.05 \cdot 10^{-1}$	–
1	$4.93 \cdot 10^{-1}$	$1.34 \cdot 10^{-1}$	$6.83 \cdot 10^{-2}$	$4.85 \cdot 10^{-2}$	$4.18 \cdot 10^{-2}$	3.28
2	$3.72 \cdot 10^{-1}$	$9.22 \cdot 10^{-2}$	$1.91 \cdot 10^{-2}$	$1.21 \cdot 10^{-2}$	$1.20 \cdot 10^{-2}$	1.80
3	$2.91 \cdot 10^{-1}$	$6.34 \cdot 10^{-2}$	$1.41 \cdot 10^{-2}$	$2.78 \cdot 10^{-3}$	$2.97 \cdot 10^{-3}$	2.01
4	$2.56 \cdot 10^{-1}$	$5.25 \cdot 10^{-2}$	$1.06 \cdot 10^{-2}$	$1.58 \cdot 10^{-3}$	$6.76 \cdot 10^{-4}$	2.14
5	$2.42 \cdot 10^{-1}$	$4.81 \cdot 10^{-2}$	$8.35 \cdot 10^{-3}$	$1.40 \cdot 10^{-3}$	$2.47 \cdot 10^{-4}$	1.45
6	$2.41 \cdot 10^{-1}$	$4.77 \cdot 10^{-2}$	$7.65 \cdot 10^{-3}$	$1.10 \cdot 10^{-3}$	$2.16 \cdot 10^{-4}$	0.20
7	$2.50 \cdot 10^{-1}$	$5.05 \cdot 10^{-2}$	$7.82 \cdot 10^{-3}$	$9.68 \cdot 10^{-4}$	$1.63 \cdot 10^{-4}$	0.40
eoc_x	–	2.31	2.69	3.01	2.57	
eoc_{xt}	–	2.47	2.65	2.92	2.75	

Table 4.4: Mesh and time-step convergence for the unfitted Eulerian BDF2 method and $\mathrm{TH}_2^{\mathsf{iso}}$ elements for the Stokes problem on a moving domain with $\nu = 10^{-2}$. Top: $\ell^2(\boldsymbol{\mathcal{L}}^2(\Omega_h(t)))$-velocity error. Middle: $\ell^2(\boldsymbol{\mathcal{H}}^1(\Omega_h(t)))$ velocity error. Bottom: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$-pressure error.

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
3	$3.09 \cdot 10^{-2}$	$3.19 \cdot 10^{-3}$	$1.08 \cdot 10^{-3}$	$9.40 \cdot 10^{-4}$	$1.01 \cdot 10^{2}$	–
4	$2.72 \cdot 10^{-2}$	$2.43 \cdot 10^{-3}$	$3.79 \cdot 10^{-4}$	$2.27 \cdot 10^{-4}$	$2.22 \cdot 10^{-4}$	18.80
5	$2.56 \cdot 10^{-2}$	$2.22 \cdot 10^{-3}$	$2.33 \cdot 10^{-4}$	$5.90 \cdot 10^{-5}$	$5.37 \cdot 10^{-5}$	2.05
6	$2.48 \cdot 10^{-2}$	$2.14 \cdot 10^{-3}$	$2.02 \cdot 10^{-4}$	$2.16 \cdot 10^{-5}$	$1.30 \cdot 10^{-5}$	2.05
7	$2.44 \cdot 10^{-2}$	$2.11 \cdot 10^{-3}$	$1.91 \cdot 10^{-4}$	$1.64 \cdot 10^{-5}$	$3.48 \cdot 10^{-6}$	1.90
8	$2.42 \cdot 10^{-2}$	$2.09 \cdot 10^{-3}$	$1.87 \cdot 10^{-4}$	$1.57 \cdot 10^{-5}$	$1.66 \cdot 10^{-6}$	1.07
9	$2.41 \cdot 10^{-2}$	$2.08 \cdot 10^{-3}$	$1.85 \cdot 10^{-4}$	$1.54 \cdot 10^{-5}$	$1.46 \cdot 10^{-6}$	0.18
eoc_x	–	3.53	3.50	3.58	3.40	
eoc_{xt}	–	3.58	3.49	3.61	3.42	
eoc_{xtt}	–	–	3.78	3.83	3.49	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
3	$5.71 \cdot 10^{-1}$	$9.26 \cdot 10^{-2}$	$1.97 \cdot 10^{-2}$	$1.13 \cdot 10^{-2}$	$1.50 \cdot 10^{4}$	–
4	$5.35 \cdot 10^{-1}$	$8.52 \cdot 10^{-2}$	$1.48 \cdot 10^{-2}$	$3.19 \cdot 10^{-3}$	$2.58 \cdot 10^{-3}$	22.47
5	$5.11 \cdot 10^{-1}$	$8.18 \cdot 10^{-2}$	$1.27 \cdot 10^{-2}$	$1.79 \cdot 10^{-3}$	$6.40 \cdot 10^{-4}$	2.01
6	$4.98 \cdot 10^{-1}$	$7.99 \cdot 10^{-2}$	$1.17 \cdot 10^{-2}$	$1.51 \cdot 10^{-3}$	$2.38 \cdot 10^{-4}$	1.43
7	$4.92 \cdot 10^{-1}$	$7.90 \cdot 10^{-2}$	$1.12 \cdot 10^{-2}$	$1.43 \cdot 10^{-3}$	$1.94 \cdot 10^{-4}$	0.30
8	$4.90 \cdot 10^{-1}$	$7.84 \cdot 10^{-2}$	$1.10 \cdot 10^{-2}$	$1.41 \cdot 10^{-3}$	$1.94 \cdot 10^{-4}$	0.00
9	$4.89 \cdot 10^{-1}$	$7.82 \cdot 10^{-2}$	$1.09 \cdot 10^{-2}$	$1.40 \cdot 10^{-3}$	$1.97 \cdot 10^{-4}$	-0.02
eoc_x	–	2.64	2.84	2.96	2.83	
eoc_{xt}	–	2.68	2.83	2.99	2.84	
eoc_{xtt}	–	–	2.87	3.15	2.87	

$L_t \downarrow \backslash L_x \rightarrow$	0	1	2	3	4	eoc_t
3	$2.68 \cdot 10^{-2}$	$4.33 \cdot 10^{-3}$	$2.81 \cdot 10^{-3}$	$2.63 \cdot 10^{-3}$	$1.09 \cdot 10^{3}$	–
4	$2.78 \cdot 10^{-2}$	$2.81 \cdot 10^{-3}$	$8.33 \cdot 10^{-4}$	$6.81 \cdot 10^{-4}$	$6.74 \cdot 10^{-4}$	20.62
5	$3.20 \cdot 10^{-2}$	$2.82 \cdot 10^{-3}$	$3.83 \cdot 10^{-4}$	$1.80 \cdot 10^{-4}$	$1.75 \cdot 10^{-4}$	1.95
6	$3.83 \cdot 10^{-2}$	$3.17 \cdot 10^{-3}$	$3.15 \cdot 10^{-4}$	$5.70 \cdot 10^{-5}$	$4.49 \cdot 10^{-5}$	1.96
7	$4.74 \cdot 10^{-2}$	$3.87 \cdot 10^{-3}$	$3.40 \cdot 10^{-4}$	$4.41 \cdot 10^{-5}$	$1.25 \cdot 10^{-5}$	1.85
8	$6.08 \cdot 10^{-2}$	$4.87 \cdot 10^{-3}$	$4.11 \cdot 10^{-4}$	$5.62 \cdot 10^{-5}$	$6.67 \cdot 10^{-6}$	0.91
9	$7.99 \cdot 10^{-2}$	$6.32 \cdot 10^{-3}$	$5.26 \cdot 10^{-4}$	$7.86 \cdot 10^{-5}$	$7.69 \cdot 10^{-6}$	-0.21
eoc_x	–	3.66	3.59	2.74	3.35	
eoc_{xt}	–	3.34	3.22	2.60	2.87	
eoc_{xtt}	–	–	3.50	3.12	2.52	

Table 4.5: Mesh and time-step convergence for the BDF2 time-stepping and $\mathrm{TH}_3^{\mathsf{iso}}$ elements for the Stokes problem on a moving domain with $\nu = 10^{-2}$. Top: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$-velocity error. Middle: $\ell^2(\mathcal{H}^1(\Omega_h(t)))$ velocity error. Bottom: $\ell^2(\mathcal{L}^2(\Omega_h(t)))$-pressure error.

of $\|\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})\|_{-1}$, independent of negative powers of Δt, holds. Furthermore, we did not observe the exponential growth of the error in time, which was predicted by our error estimate. Concerning the spatial approximation properties, we saw in the case of a low order geometry approximation that the geometry approximation error could be the dominating factor. Using a sub-division strategy to hide the geometry approximation error, we revealed the underlying discretisation error for TH$_2$ and TH$_3$ elements. Extending the approach to higher-order in both space and time, we observed that the extension to a BDF2 discretisation of the time-derivative is unproblematic, and second-order convergence in time holds. Furthermore, using isoparametric unfitted finite elements together with a transfer-operator to evaluate functions defined with respect to one deformation on a mesh deformed with a different deformation leads to higher-order convergence.

4.5 Application to Navier-Stokes on a Moving Domain and Comparison with Other Extension Techniques

Since our ultimate goal is to approximate fluid-rigid body problems, we consider applying the method developed and analysed for the time-dependent Stokes problem on moving domain, applied to the full nonlinear Navier-Stokes equations on a moving domain (2.12).

4.5.1 Related methods

We shall compare our method with the following two unfitted Eulerian methods.

Explicit Extension

In the thesis by SCHOTT [Sch17], an explicit extension technique for moving domain flow problems was described and used, which realises an extension to non-fluid elements also based on ghost-penalty stabilisation. The essential difference to our approach is that the extension is performed in a separate computational step from the fluid-flow solver and that the extension is only available on a *vertex patch* of cut elements. The procedure of this is as follows:

1. Solve the CutFEM flow problem on the current domain Ω_h^n using the standard, stationary domain approach discussed in chapter 3.

2. Mark the set of exterior elements, which share a vertex with a cut element

$$\mathscr{T}_{h,\text{ext}}^n \coloneqq \{T \in \widetilde{\mathscr{T}_h} \setminus \mathscr{T}_h^n \mid \exists T_1 \in \mathscr{T}_h^n \text{ with } \text{meas}_{d-2}(T \cap T_1) \neq 0\}.$$

3. Mark the set of interior extension facets

$$\mathscr{F}_{h,\text{ext}}^n = \{\overline{T_1} \cap \overline{T_2} \mid T_1 \neq T_2 \text{ with } T_1 \in \mathscr{T}_{h,\text{ext}}^n, T_2 \in \mathscr{T}_{h,\text{ext}}^n \cup \mathscr{T}_{h,\Gamma_h^n}\}.$$

4. Based on the facets $F \in \mathscr{F}_{h,\text{ext}}^n$, solve the $\mathcal{L}^2$-ghost-penalty problem $h^2 i_h(\boldsymbol{u}_h, \boldsymbol{v}_h) = 0$, on the degrees of freedom in elements $\mathscr{T}_{h,\text{ext}}^n$, which were not active in the fluid step 1, to extend the solution into the set of extension elements. If the Ω_h^{n+1} is a subset of the vertex-patch extended domain, continue to the next time-step.

See Figure 4.5 for an illustration of this element marking procedure. As the solution is only extended into a strip with a width of one element, this technique requires a CFL-type condition $\Delta t \lesssim h$ on the time-step. An advantage of the method is that the ghost-penalties in the fluid solver only act where they are necessary for the stability of the fluid system and do not interfere with the fluid solution. However, to the best of our knowledge, no numerical analysis is available for this approach.
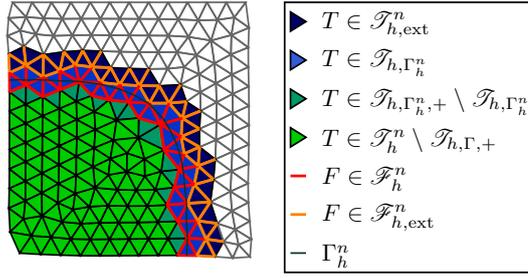
Figure 4.5: Element and facets markings for the explicit ghost-penalty extension technique from [Sch17].
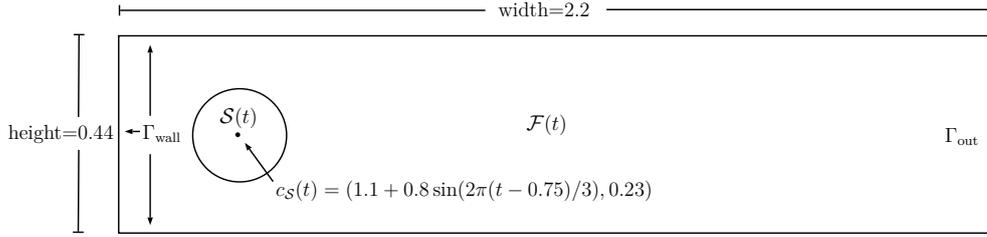
Figure 4.6: Computational domain for the flow driven by a moving cylinder problem at time $t = 0$.

Implicit Extension for Large Viscosities

The Eulerian time-stepping method for the time-dependent Stokes problem on moving domains, as covered in this chapter and in [WRL21], is an extension of the same technique for a time-dependent convection-diffusion problem on moving domains in [LO19]. While the authors did include a diffusion coefficient, the analysis in [LO19] assumes a viscosity $\nu \approx 1$. As a result, the ghost-penalty stabilisation in [LO19] realising the discrete extension only scales with ν, rather than $\nu + 1/\nu$ as above. As a result of our $1/\nu$ scaling, the ghost-penalties dominate in the entire extension strip, introducing a potentially very large consistency error. Therefore, we consider this approach, i.e., only scale the extension ghost-penalties with ν, applied to the Navier-Stokes equations on a moving domain, to investigate the effects this scaling has on the resulting flow and the stability of the method.

4.5.2 Numerical Examples

As our reference problem, we shall consider a moving cylinder inside a channel. This set-up is taken from [Sch17, Section 3.6.4.3]. The background domain is a channel $\widetilde{\Omega} = (0, 2.2) \times (0, 0.44)$. Within this channel, we have a rigid, circular obstacle $\mathcal{S}$, given by the time-dependent domain

$$\mathcal{S}(t) = \{\boldsymbol{x} \in \mathbb{R}^2 \mid (\boldsymbol{x}_1 - d(t))^2 + (\boldsymbol{x}_2 - 0.23)^2 < 0.1^2 \text{ with } d(t) = 1.1 + 0.8\sin(\pi^2/3(t - 0.75))\}.$$

The fluid domain is then $\mathcal{F}(t) = \widetilde{\Omega} \setminus \overline{\mathcal{S}(t)}$. We then consider the incompressible Navier-Stokes equations in the fluid domain, i.e., the reduced system (2.5a) – (2.5c). On the top, left and bottom boundaries of the channel Γ_{wall} we then consider no-slip boundary conditions, while at the right end of the channel Γ_{out} we consider the do-nothing boundary condition. At the fluid-solid interface $\mathcal{I}(t) = \partial\mathcal{S}(t)$ we impose a no-slip condition, i.e., continuity of the velocity. This is given by $U(t) = 0.8(2\pi/3)\cos((2\pi/3)(t - 0.75))$. A sketch of the spatial configuration at $t = 0$ can be seen in Figure 4.6.

Quantities of Interest

In order to quantify the quality of the resulting computations, we will look at the forces acting on the cylinder over time with $\boldsymbol{F}(t)$ as defined in (2.4). As quantities of interest, we consider $(\boldsymbol{F}_1^{\text{max}}, t_1^\circ)$ and

$(\boldsymbol{F}_2^{\max}, t_2^\circ)$, i.e., the maxima of the two components of the force acting on the cylinder and the times at which these maxima are realised. As an additional quantity, which is of more interest for fluid-structure interactions with an elastic solid, we consider the point evaluation of the normal stress $\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}$ at the top of the cylinder $\boldsymbol{x}(t) = (1.1 + 0.8\sin(\pi 2/3(t - 0.75)), 0.33)$. We again look for the maximum of the components of the normal point force and the time at which these maxima are realised. We denote these as $(\boldsymbol{f}_1^{\max}, t_1^*)$ and $(\boldsymbol{f}_2^{\max}, t_2^*)$, respectively.

Reference Values

To compute accurate reference values for these quantities of interest, we shall use a fitted ALE approach. This is feasible because the cylinder remains at a distance of $0.3 (= 1.5 \cdot diam)$ away from the left and right boundaries and the distance to the top and bottom boundaries remains constant. The ALE formulation of the incompressible Navier-Stokes equations is then given as

$$J_{\text{ALE}}\rho_{\mathcal{F}}\big(\partial_t \boldsymbol{u} + \boldsymbol{F}_{\text{ALE}}^{-1}(\boldsymbol{u} - \partial_t \hat{\boldsymbol{w}}) \cdot \nabla \boldsymbol{u}\big) - \text{div}\big(J\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{F}_{\text{ALE}}^{-T}\big) = J\rho_{\mathcal{F}}\boldsymbol{f} \qquad \text{in } \hat{\mathcal{F}}$$
$$- \text{div}\big(J\boldsymbol{F}_{\text{ALE}}^{-1}\boldsymbol{u}\big) = 0 \qquad \text{in } \hat{\mathcal{F}},$$

where $\hat{\mathcal{F}}$ is the reference fluid domain, $\boldsymbol{F}_{\text{ALE}} = \nabla T_{ALE}$ is the deformation gradient and $J_{ALE} = \det(\boldsymbol{F}_{\text{ALE}})$ $\hat{\boldsymbol{w}}$ is the mesh velocity. For our computations, we consider the reference domain $\hat{\mathcal{F}} := \mathcal{F}(0.75)$, i.e., we locate the cylinder in the centre of the domain. The ALE mapping for the translational motion is then defined as

$$T_{\text{ALE}}(t) = f_{\text{ALE}} \cdot \left(\text{id} + \begin{pmatrix} 0.8\sin(\pi 2/3(t - 0.75)) \\ 0 \end{pmatrix} \right)$$

$$f_{\text{ALE}} = \begin{cases} \boldsymbol{x}_1/0.9 & \text{for } \boldsymbol{x}_1 < 0.9 \\ 1 & \text{for } 0.9 \le \boldsymbol{x}_1 \le 1.1 \\ (2.2 - \boldsymbol{x}_1)/0.9 & \text{for } \boldsymbol{x}_1 > 1.3. \end{cases}$$

Consequently, the mesh velocity can therefore also be given analytically and the resulting deformation gradient is discontinuous. However, we overcome this problem by resolving the discontinuity lines at $\boldsymbol{x}_1 = 0.9, 1.3$ with the mesh. Because `NGSolve` uses mapped Gauß-Jacobi type quadrature rules with a Duffy-transformation, the deformation gradient remains well-defined in every quadrature point.

To compute the body forces, we test the variational formulation with a non-conforming test function $\boldsymbol{v}_h$, which takes the value 1 in the respective component on the cylinder boundary and is zero everywhere else. This is known as the Babuška-Miller trick [BM84], which is known to result in convergence of the forces at twice the order, c.f. [BR06; Wah+19b].

We start a mesh of the reference domain with $h_{\max} = 0.08$ in the bulk and $h = 0.008$ on the cylinder surface. To ensure that the values are computed accurately, we consider $L_x = 0, 1, 2, 3$ levels of uniform mesh refinements of this initial mesh. On these meshes, we consider TH$_5$ elements. We use the BDF2 scheme for the temporal discretisation and take the time-step $\Delta t = 0.005 \cdot 2^{-L_x}$. The resulting values for the quantities of interest can be seen in Table 4.6. While we only observe second-order convergence in the first three functional values, we still have at least three significant figures of accuracy in all quantities of interest. This rate indicates that the temporal error is the dominant factor here or that the exact solution lacks the regularity for higher-order convergence. Furthermore, the curves of the four functional values can be seen in Figure 4.7. Here we can see that it is challenging to identify the correct maximum of the body forces since there are two points in time where local maxima are realised with similar values.

Method Comparison

We now compare how the three Eulerian moving domain approaches using the moving cylinder example. In the following, we will denote the explicit extension technique as `EE`, the implicit extension approach

Discretisation		Results							
L_x	Δt	$\boldsymbol{F}_1^{\mathrm{max}}$	t_1°	$\boldsymbol{F}_2^{\mathrm{max}}$	t_2°	$\boldsymbol{f}_1^{\mathrm{max}}$	t_1^*	$\boldsymbol{f}_2^{\mathrm{max}}$	t_2^*
0	$1/200$	$0.973\,925 \cdot 10^0$	2.42	$1.032\,954 \cdot 10^0$	2.435	$0.297\,942 \cdot 10^0$	0.69	$0.648\,448 \cdot 10^0$	1.475
1	$1/400$	$0.974\,399 \cdot 10^0$	2.42	$1.038\,728 \cdot 10^0$	2.435	$0.298\,125 \cdot 10^0$	0.69	$0.642\,801 \cdot 10^0$	1.475
2	$1/800$	$0.973\,745 \cdot 10^0$	2.42	$1.041\,018 \cdot 10^0$	2.43375	$0.298\,138 \cdot 10^0$	0.69	$0.642\,894 \cdot 10^0$	1.47375
3	$1/1600$	$0.973\,482 \cdot 10^0$	2.42	$1.041\,359 \cdot 10^0$	2.43375	$0.298\,142 \cdot 10^0$	0.69	$0.642\,966 \cdot 10^0$	1.473125
Extrapolated(order)		$0.973\,306 \cdot 10^0 (1.32)$	–	$1.041\,418 \cdot 10^0 (2.75)$	–	$0.298\,143 \cdot 10^0 (1.84)$	–	$0.643\,202 \cdot 10^0 (0.38)$	–

Table 4.6: Results for the moving cylinder problem computed using a fitted ALE `TH`$_5$ discretisation.
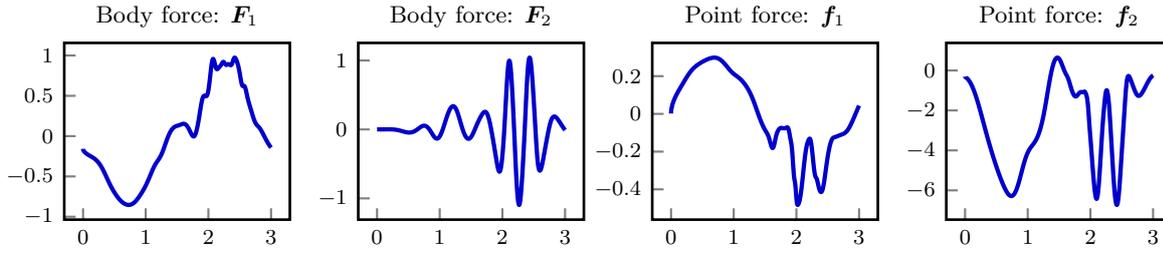
Figure 4.7: Reference curves for the moving cylinder problem. Computed using an ALE discretisation with `TH`$_5$ elements on a mesh with $h_{\mathrm{max}} = 0.01$ and $h = 0.001$ on the cylinder, BDF2 time-stepping with $\Delta t = 1/1600$ and an analytical formulation of the ALE mapping.

for large viscosities as `IE`$_{\mathrm{LO}}$ and our choice of extension scaling as `IE`$_{\mathrm{WRL}}$.

All computations are performed of a quasi-uniform mesh of the background domain. Specifically, no region of the mesh is refined towards the cylinder. As we have seen in subsection 3.4.3, this means that the force values are not very accurate. However, since the cylinder moves over time, we would have an over refined mesh in most of the bulk if we refined the mesh in the whole region where the cylinder will be over time, and the `EE` method would require even smaller time-steps.

Convergence Study We compute the moving cylinder problem using `TH`$_2$ elements and the mesh and time-step pars $(h_{\mathrm{max}}, \Delta t) = (0.02, 1/500), (0.01, 1/1000), (0.005, 1/2000)$. Note that we have chosen particularly small time-steps here in order for all three methods to work. For all the methods, the ghost-penalty parameter is $\gamma_{gp} = 0.1$.

The full results can be seen in Table 4.7, and a close up of the drag force around the two similar local maxima can be seen in Figure 4.8. Furthermore, the velocity and pressure solution for all three methods on the coarsest mesh, compared against an ALE reference solution, can be seen in Figure 4.9 and Figure 4.10, respectively.

In Table 4.7, we see that the results from the `EE` and `IE`$_{\mathrm{LO}}$ methods are very similar and that these methods manage to capture the correct peak of the drag and lift functionals of the two finer meshes. The point forces are less accurate but also converge towards the values determined with the high-order ALE computations. For the `IE`$_{\mathrm{WRL}}$ method, we see that while the drag and lift values also converge towards the correct values, we do not manage to capture the correct time of realisation. Furthermore, the realised values are not accurate.

Looking at the plots in Figure 4.8, we observe that the force evaluation is highly unstable in all three methods, but particularly in the `IE`$_{\mathrm{WRL}}$ method. Also, these instabilities remain large under mesh/time-step refinement for the `IE`$_{\mathrm{WRL}}$ computations. This explains why the maxima were realised at the wrong point in time, even on the finer meshes.

Examining the visualisation of the velocity magnitude in Figure 4.9, we see that both the `EE` and `IE`$_{\mathrm{LO}}$ methods manage to capture most artefacts of the flow well. However, we also see that the extension is very

Method	Discretisation		Results							
	$h_{\max}$	Δt	$\boldsymbol{F}_1^{\max}$	t_1°	$\boldsymbol{F}_2^{\max}$	t_2°	$\boldsymbol{f}_1^{\max}$	t_1^*	$\boldsymbol{f}_2^{\max}$	t_2^*
EE	0.02	$1/500$	$0.932\,332 \cdot 10^0$	2.25	$0.937\,533 \cdot 10^0$	2.122	$0.704\,371 \cdot 10^0$	0.724	$0.743\,263 \cdot 10^0$	1.474
EE	0.01	$1/1000$	$0.970\,163 \cdot 10^0$	2.425	$1.044\,377 \cdot 10^0$	2.115	$0.585\,481 \cdot 10^0$	0.77	$0.660\,947 \cdot 10^0$	1.463
EE	0.005	$1/2000$	$0.975\,244 \cdot 10^0$	2.4195	$1.039\,498 \cdot 10^0$	2.434	$0.388\,403 \cdot 10^0$	0.727	$0.642\,949 \cdot 10^0$	1.4705
IE_{LO}	0.02	$1/500$	$0.931\,615 \cdot 10^0$	2.25	$0.939\,532 \cdot 10^0$	2.122	$0.702\,661 \cdot 10^0$	0.724	$0.742\,830 \cdot 10^0$	1.474
IE_{LO}	0.01	$1/1000$	$0.970\,256 \cdot 10^0$	2.425	$1.045\,478 \cdot 10^0$	2.115	$0.584\,517 \cdot 10^0$	0.77	$0.661\,351 \cdot 10^0$	1.455
IE_{LO}	0.005	$1/2000$	$0.975\,705 \cdot 10^0$	2.42	$1.040\,348 \cdot 10^0$	2.434	$0.388\,481 \cdot 10^0$	0.727	$0.643\,203 \cdot 10^0$	1.4705
IE_{WRL}	0.02	$1/500$	$1.397\,117 \cdot 10^0$	2.186	$0.545\,835 \cdot 10^0$	2.206	$0.565\,051 \cdot 10^0$	0.724	$0.813\,698 \cdot 10^0$	1.476
IE_{WRL}	0.01	$1/1000$	$1.007\,292 \cdot 10^0$	2.116	$0.996\,508 \cdot 10^0$	2.137	$0.422\,517 \cdot 10^0$	0.77	$0.793\,305 \cdot 10^0$	1.487
IE_{WRL}	0.005	$1/2000$	$0.957\,557 \cdot 10^0$	2.08	$1.004\,391 \cdot 10^0$	2.1165	$0.354\,375 \cdot 10^0$	0.727	$0.645\,141 \cdot 10^0$	1.4685
	Reference		$0.973\,300 \cdot 10^0$	2.42	$1.041\,400 \cdot 10^0$	2.433	$0.298\,143 \cdot 10^0$	0.69	$0.643\,000 \cdot 10^0$	1.473

Table 4.7: Results from the three Eulerian methods for the moving cylinder problem.
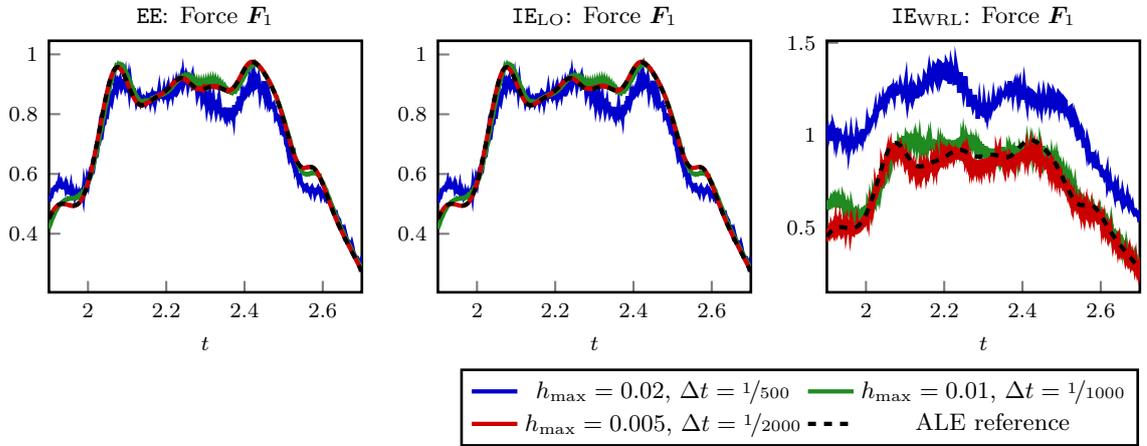
Figure 4.8: Close up of the drag functional resulting from the Eulerian time-stepping methods for the moving cylinder problem for $1.9 \le t \le 2.7$.

unstable. For the IE_{WRL} method, we have several observations. First, we note that the extension outside of the fluid domain is very stable. However, we also see that the extension has a large influence on the fluid in the region where we apply the extension ghost-penalties. We attribute this to the ghost-penalty stabilisation term, which dominates the system in this region due to the $1/\nu$ scaling. We view this as the influence of the consistency error of the ghost-penalty operator since the solution cannot be represented by a single polynomial in the extension region. This appears to have the effect that the fluid is obstructed, making the cylinder effectively larger, increasing the Reynolds number of the flow, explaining the more turbulent velocity field. If we further look at the pressure field in Figure 4.10, we again see that the EE and IE_{LO} results match the ALE solution well, while the IE_{WRL} pressure is very different. Note that both in Figure 4.9 and Figure 4.10, the colour range was based on the ALE reference solution.

Improving the Implicit Extension: Choice of Ghost-Penalty Parameter　The difference between the IE_{LO} and IE_{WRL} methods is the scaling of the extension ghost-penalties of ν and $\nu + \frac{1}{\nu}$, respectively. To investigate the effect of the very large and intrusive extension ghost-penalties, we consider the mesh with $h_{\max} = 0.01$ and time-step $\Delta t = 1/1000$, and decrease the ghost-penalty parameter.

The resulting quantities of interest can be seen in Table 4.8, and a close up of the drag functional around the two similar local maxima can be seen on the left of Figure 4.11. The values seen in Table 4.8 indicate that a smaller ghost-penalty term is indeed beneficial for this problem. However, with a very small stabilisation parameter, we also do not achieve good results. This could be due to the loss of control over
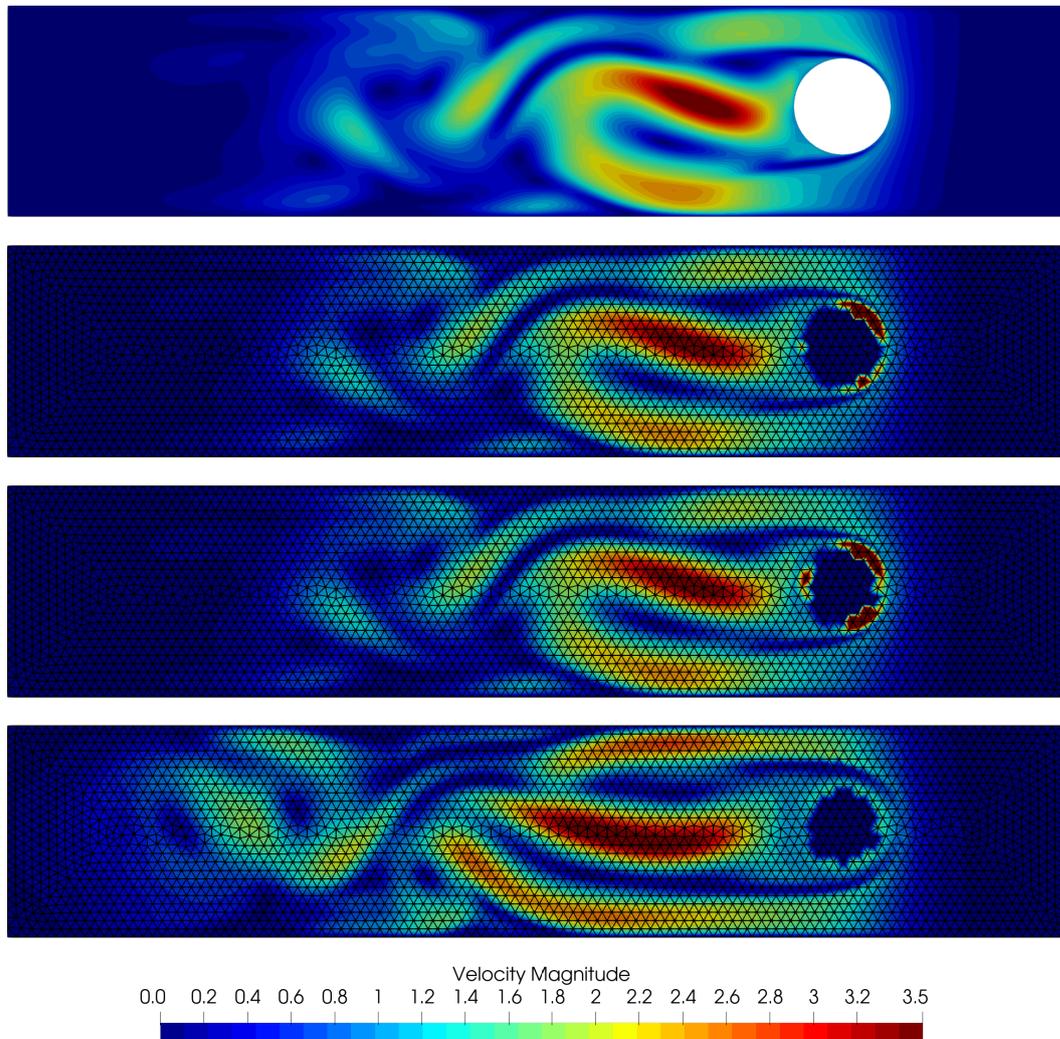
Figure 4.9: Velocity solution of the moving cylinder problem at $t = 1.18$. From top to bottom: ALE, EE, IE_{LO} and IE_{WRL}. ALE computed on a mesh with $h_{\text{max}} = 0.02$ and $\Delta t = 1/800$. Unfitted computations on a mesh with $h_{\text{max}} = 0.02$ and $\Delta t = 1/500$. (A video of the simulations is available at https://youtu.be/Wka_WyXyJmg.)
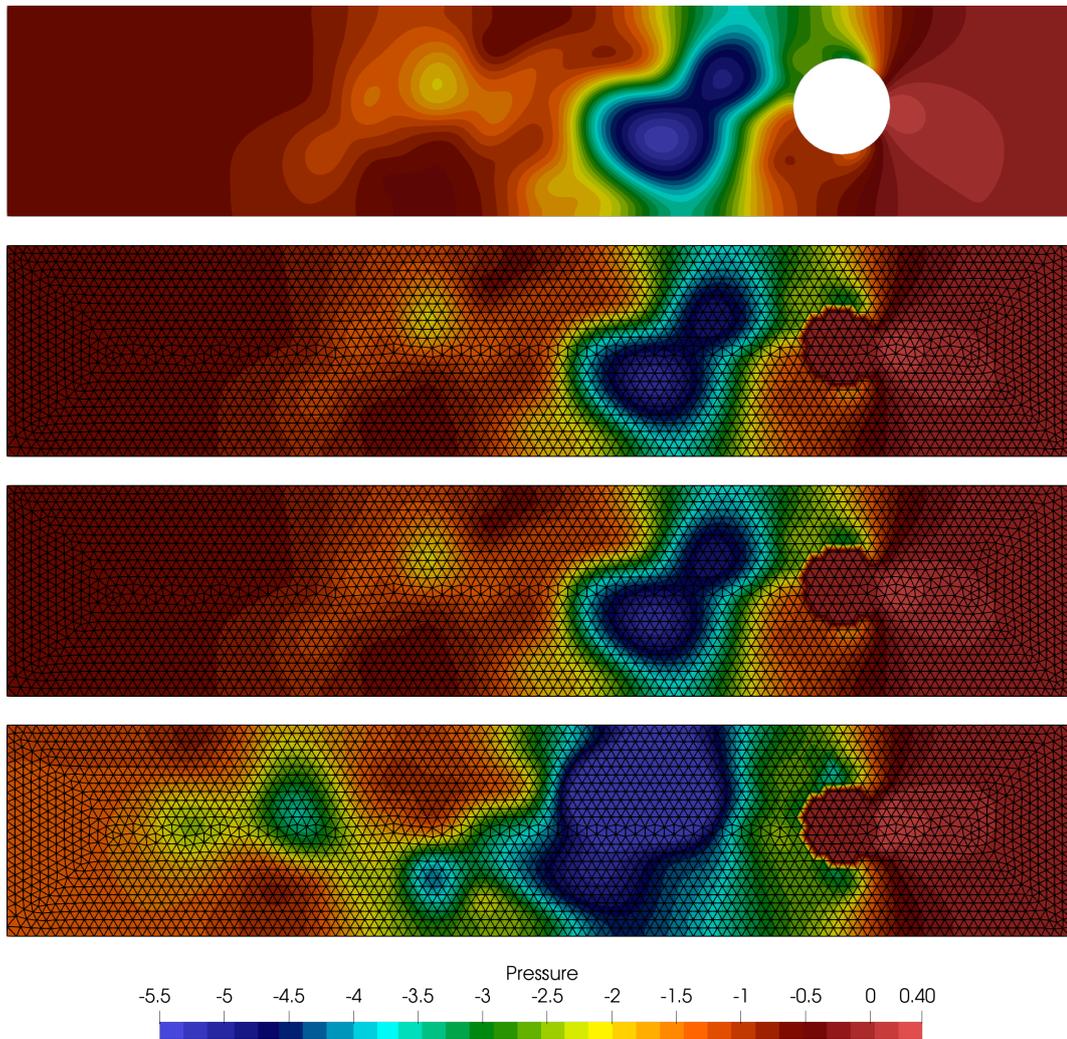
Figure 4.10: Pressure solution of the moving cylinder problem at $t = 1.18$. From top to bottom: ALE, EE, IE_{LO} and IE_{WRL}. ALE computed on a mesh with $h_{\text{max}} = 0.02$ and $\Delta t = 1/800$. Unfitted computations on a mesh with $h_{\text{max}} = 0.02$ and $\Delta t = 1/500$.

Discr.	Results							
γ_{gp}	$\boldsymbol{F}_1^{\max}$	t_1°	$\boldsymbol{F}_2^{\max}$	t_2°	$\boldsymbol{f}_1^{\max}$	t_1^*	$\boldsymbol{f}_2^{\max}$	t_2^*
0.1	1.007 292	2.116	0.996 508	2.137	0.422 517	0.77	0.793 305	1.487
0.01	0.958 161	2.253	1.059 681	2.134	0.443 084	0.77	0.819 559	1.46
0.001	0.929 864	2.079	1.064 706	2.129	0.486 860	0.77	0.765 608	1.461
0.0001	0.899 494	2.076	1.023 838	2.119	0.553 787	0.77	0.691 719	1.468
Reference	0.973 300	2.42	1.041 400	2.433	0.298 143	0.69	0.643 000	1.473

Table 4.8: Results for the $\mathtt{IE_{WRL}}$ method for the moving cylinder problem over a series of different ghost-penalty stabilisation parameters with $h_{\max} = 0.01$ and $\Delta t = {}^1/_{1000}$.

Discr.		Results							
$h_{\max}$	Δt	$\boldsymbol{F}_1^{\max}$	t_1°	$\boldsymbol{F}_2^{\max}$	t_2°	$\boldsymbol{f}_1^{\max}$	t_1^*	$\boldsymbol{f}_2^{\max}$	t_2^*
0.02	${}^1/_{500}$	1.180 75	2.158	0.706 26	2.49	0.546 45	0.724	1.125 70	1.498
0.01	${}^1/_{1000}$	1.053 76	2.103	1.185 67	2.14	0.473 37	0.77	0.968 98	1.499
0.005	${}^1/_{2000}$	0.961 62	2.079	1.059 11	2.442	0.370 53	0.727	0.804 64	1.4995
Reference		0.973 30	2.42	1.041 40	2.433	0.298 14	0.69	0.643 00	1.473

Table 4.9: Results for the $\mathtt{IE_{WRL}}$ method for the moving cylinder problem where the extension ghost-penalties are only applied in the area where they are strictly necessary.

the pressure on cut elements with a small stability parameter. Looking at the left plot in Figure 4.11, we see that we capture more of the dynamics. However, we also note that the instabilities of the drag evaluation increase. This makes sense, since on the one hand, a smaller ghost-penalty parameter means that the ghost-penalties are less obstructive to the flow, while on the other hand, we have less control over the solution on cut and unphysical extension elements.

Improving the Implicit Extension: Strictly Necessary Extension In the previous computations, the extension has been applied naively in the entire δ-extension strip. However, in this example, the motion of the solid and thereby the exact extension region is known a priori. We, therefore, refine the implementation to only apply the large extension $(\nu + {}^1/_\nu)$-scaled ghost-penalties in the portion of the extension strip, where the extension is strictly necessary. We then take the same series of meshes and time-steps as considered for the naive implementation above.

The results for the quantities of interest can be seen in Table 4.9, and a close up of the drag functional around the two largest local maxima can be seen on the right of Figure 4.11. Furthermore, we compare the velocity and pressure solution between the naive and the more refined $\mathtt{IE_{WRL}}$ implementations against the ALE reference solution in Figure 4.12 and Figure 4.13, respectively.

In Table 4.9, we see that the resulting values are much better than before. While still not as accurate as the other methods, $\boldsymbol{F}_{2,\max}$ is realised at the correct peak on the finest mesh. Looking at the solution from the coarsest discretisation in Figure 4.12 and Figure 4.13, we see that the solution is much closer to the ALE solution than before. However, we also observe in Figure 4.11 that the instabilities in the drag evaluation are still present, although not as large as before.

Improving the Implicit Extension: Functional Smoothing As we have observed in all the previous plots of the drag functional for all three extension methods, the functional values oscillate very strongly around a mean curve. Therefore, to establish whether more accurate values for the quantities of interest can be obtained from the $\mathtt{IE_{WRL}}$ method, we take the values obtained from the implementation where the extension is only applied in the necessary region, and approximate the data by a fourth-order spline.
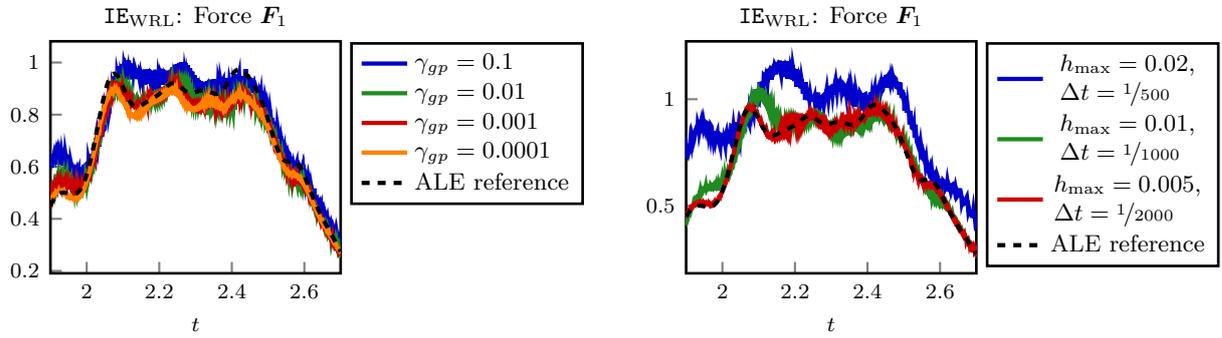
Figure 4.11: Drag force on the moving cylinder computed with the IE_{WRL} method. Left: Computed on a mesh with $h_{\max} = 0.01$ and time-step $\Delta t = \frac{1}{1000}$ for different ghost-penalty stabilisation parameters. Right: Extension scaled ghost-penalties only applied where the extension is strictly necessary.
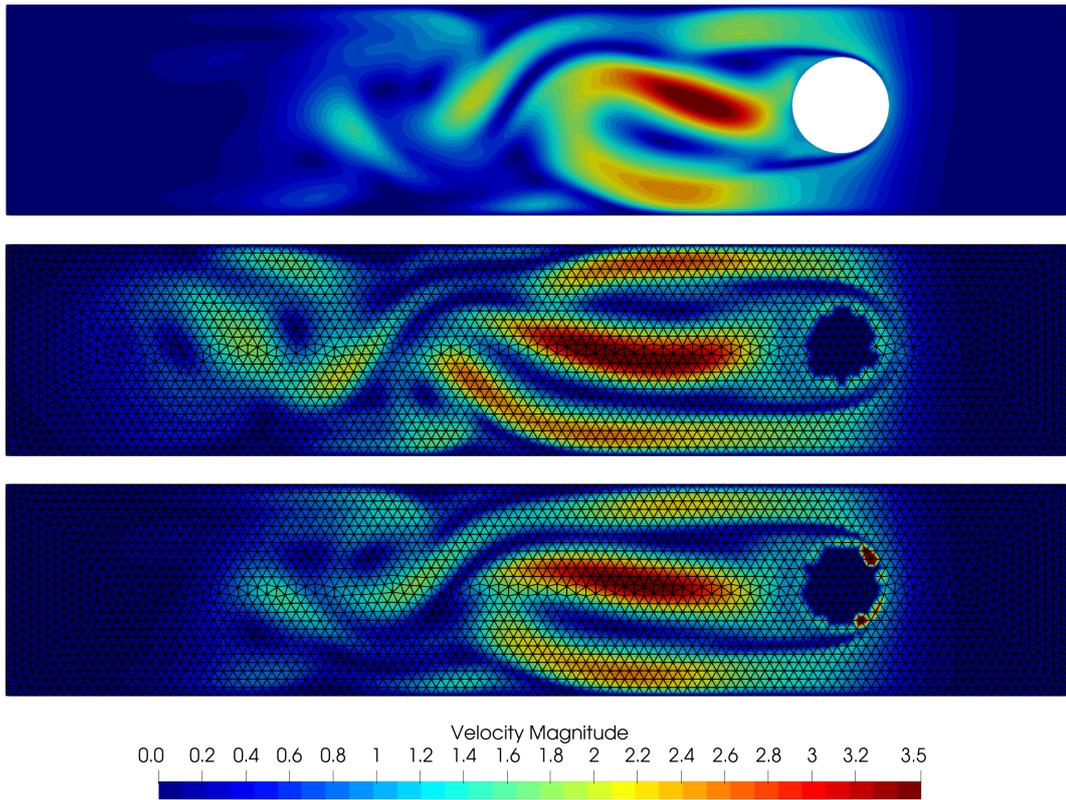
Figure 4.12: Velocity solution of the moving cylinder problem at t=1.18. Top: ALE reference, Middle: IE_{WRL}, Bottom: IE_{WRL} where the extension is only applied in the necessary region. Unfitted computations on a mesh with $h_{\max} = 0.02$ and $\Delta t = \frac{1}{500}$. (A video of these simulations is available at https://youtu.be/tunYqJ3LTb4.)
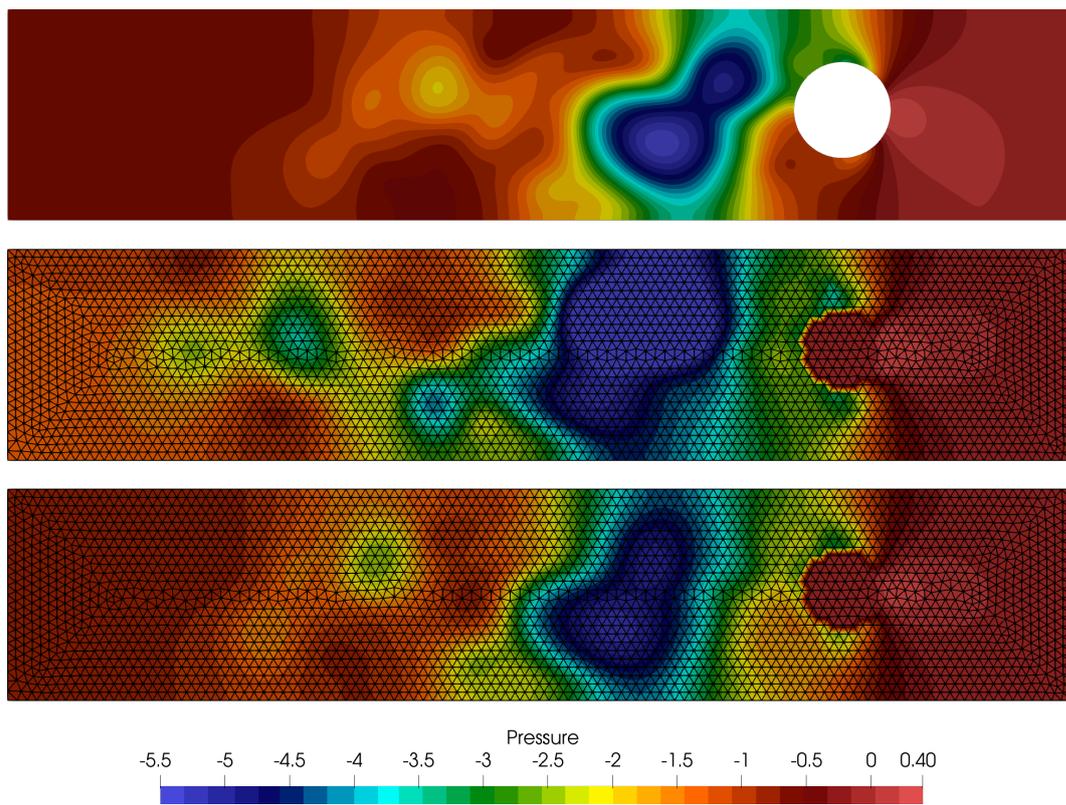
Figure 4.13: Pressure solution of the moving cylinder problem at $t = 1.18$. Top: ALE reference, Middle: $\mathrm{IE}_{\mathsf{WRL}}$, Bottom: $\mathrm{IE}_{\mathsf{WRL}}$ where the extension is only applied in the necessary region. Unfitted computations on a mesh with $h_{\mathsf{max}} = 0.02$ and $\Delta t = {}^{1}/{}_{500}$.

Discr.		Results							
$h_{\max}$	Δt	$\boldsymbol{F}_1^{\max}$	t_1°	$\boldsymbol{F}_2^{\max}$	t_2°	$\boldsymbol{f}_1^{\max}$	t_1^*	$\boldsymbol{f}_2^{\max}$	t_2^*
0.02	$^1/_{500}$	1.148 98	2.152 99	0.680 84	2.488 01	0.226 53	0.703 51	0.889 30	1.515 55
0.01	$^1/_{1000}$	1.025 66	2.101 66	1.174 62	2.139 24	0.261 63	0.713 67	0.694 91	1.441 27
0.005	$^1/_{2000}$	0.959 94	2.077 90	1.055 54	2.441 18	0.289 22	0.722 76	0.654 00	1.468 69

Table 4.10: Results for the IE$_{\text{WRL}}$ method for the moving cylinder problem after smoothing the data with a fourth-order spline. The extension scaled ghost-penalties are only applied in the area where they are strictly necessary.
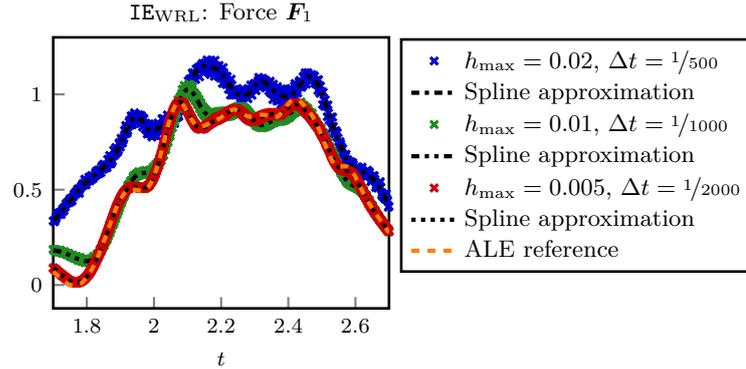
Figure 4.14: Drag force on the moving cylinder computed with the IE$_{\text{WRL}}$ method where the extension ghost-penalties only applied where the extension is strictly necessary and the spline approximation of the resulting data.

We construct the spline by placing a knot at every fifteenth time-step and determining the coefficients through a least-squares fit to the data.

The quantities of interest, as computed by the evaluation of the fourth-order spline, can be seen in Table 4.10. The original data points and the resulting splines for the drag functional around the two largest local maxima can be seen in Figure 4.14. Here we see that while the results are better than without the spline smoothing, the improvement is not particularly large. However, looking at Figure 4.14, we see that the curves have smoothed the data well, removing the unwanted variance while also capturing the functional's features.

Larger Extension Comparison In order to be able to compare the extension methods, the previous computations have been using very small time-steps in comparison to the mesh size. Therefore, we investigate the behaviour of the three methods for larger time-steps. To this end, we take the coarsest considered mesh with $h_{\max} = 0.02$ and take the time-step to $\Delta t = {}^1/_{125}, {}^1/_{250}, {}^1/_{500}$. For the IE$_{\text{WRL}}$ method, we take the implementation, which only applies the extension ghost-penalty stabilisation on elements where it is strictly necessary.

The resulting quantities of interest can be seen in Table 4.11. Here we see that the IE$_{\text{WRL}}$ extension method is the only robust method with respect to larger extension domains. On the coarsest mesh, the CFL condition for the EE method is violated, and IE$_{\text{LO}}$ is unstable. With the second time-step, both the EE and IE$_{\text{LO}}$ are unstable due to the weak extension. The results from the smallest time-steps are the same as above. These results are not surprising since the EE and IE$_{\text{LO}}$ methods do not give a very stable extension into the solid domain, as we have seen above.

As an addition, we consider the time-steps $\Delta t = {}^1/_{75}, {}^1/_{1000}$ as well as the time-step $\Delta t = {}^1/_{125}$ in combination with the smaller extension ghost-penalty parameter $\gamma_{u,gp,2} = 0.01$ on the mesh with $h_{\max} = 0.02$. Furthermore, we consider the finer mesh with $h_{\max} = 0.01$ and the large time-step $\Delta t = {}^1/_{125}$.

Method	Discr.		Results							
	Δt	K	$\boldsymbol{F}_1^{\max}$	t_1°	$\boldsymbol{F}_2^{\max}$	t_2°	$\boldsymbol{f}_1^{\max}$	t_1^*	$\boldsymbol{f}_2^{\max}$	t_2^*
EE	$^1/_{125}$	3				CFL condition violated				
EE	$^1/_{250}$	2				Method Unstable				
EE	$^1/_{500}$	1	0.932 332	2.25	0.937 533	2.122	0.704 371	0.724	0.743 263	1.474
IE_{LO}	$^1/_{125}$	3				Method unstable				
IE_{LO}	$^1/_{250}$	2				Method unstable				
IE_{LO}	$^1/_{500}$	1	0.975 244	2.4195	1.039 498	2.434	0.388 403	0.727	0.642 949	1.4705
IE_{WRL}	$^1/_{125}$	3	1.373 278	2.248	0.418 367	2.536	0.524 403	0.712	0.537 514	1.544
IE_{WRL}	$^1/_{250}$	2	1.247 883	2.184	0.548 222	2.492	0.526 115	0.724	0.947 352	1.496
IE_{WRL}	$^1/_{500}$	1	1.397 117	2.186	0.545 835	2.206	0.565 051	0.724	0.813 698	1.476
	Reference		0.973 300	2.42	1.041 400	2.433	0.298 143	0.69	0.643 000	1.473

Table 4.11: Comparison of the three Eulerian methods for the moving cylinder problem on a fixed mesh with $h_{max} = 0.02$ and a series of different time-steps, resulting in larger extension regions.
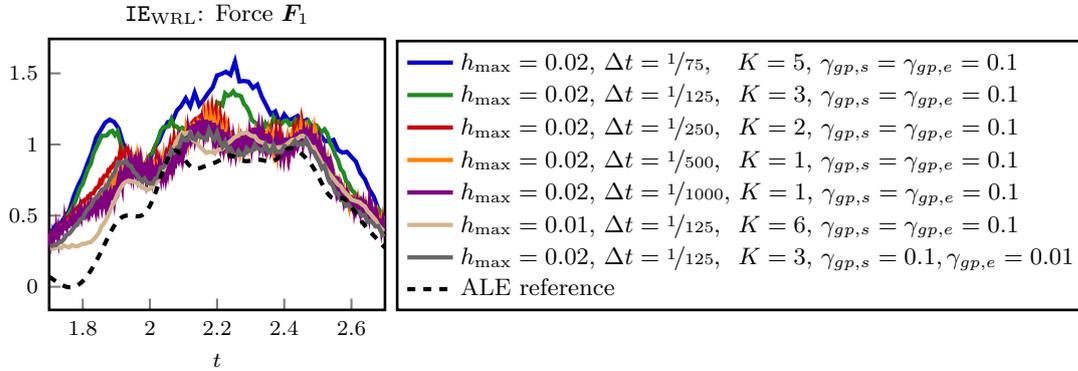
Figure 4.15: Drag force on the moving cylinder computed with the IE_{WRL} method. Computed on a mesh with $h_{\max} = 0.02$ and different time-steps, resulting in different extension region widths. Extension scaled ghost-penalties are only applied in the region where they are strictly necessary.

The resulting drag functional around the two largest maxima can be seen in Figure 4.15. Here we see that the computations with smaller time-steps are more accurate on the same mesh. However, the drag evaluation also becomes less stable with smaller time-steps. We also note that we have achieved a comparably accurate and stable computation on the medium mesh with $h_{max} = 0.01$ together with the relatively large time-step $\Delta t = ^1/_{125}$, resulting in a maximal extension domain of width $K = 6$ elements. For the computation with a smaller extension ghost-penalty parameter but the same stability parameter, we see that the drag is similarly accurate as the resulting value from the smallest time-step on that mesh while being about as stable as the results from the largest time-steps on this mesh.

Remark 4.25 (Choice of ghost-penalty operator). All presented computations have been implemented using the direct version of the ghost-penalty operator. A more popular choice is to penalise normal-derivative jumps across between elements. However, this choice has the disadvantage that normal-derivatives of higher-order are difficult to compute (accurately), making the normal-derivative version only feasible for lower-order methods. Tests with the IE_{WRL} method, using TH_2 elements and the normal-derivative jump ghost-penalty operator, resulted in functional values, which were significantly less stable than the results using the direct version on the same discretisation. ▲

Remark 4.26 (Evaluation of the drag/lift functional). The dominating ghost-penalties in the extension region of the IE_{WRL} method could be behind the observed instabilities in the drag/lift functional when evaluating the drag functional in its basic form, i.e., integrating directly over the relevant boundary. To

transfer the boundary integral on the unfitted boundary onto the other boundaries and a volume integral, we use integration by parts.

We split the boundary of the domain into fitted Dirichlet part Γ_D, a fitted Neumann part Γ_N and the unfitted interface at the cylinder $\Gamma_\circ$. As a result, we have $\partial\Omega = \Gamma_D \cup \Gamma_N \cup \Gamma_\circ$ and $\Gamma_D \cap \Gamma_N = \emptyset$, $\Gamma_D \cap \Gamma_\circ = \emptyset$, $\Gamma_N \cap \Gamma_\circ = \emptyset$. We take the momentum-balance equation of the Navier-Stokes equations in the strong form, multiply with a test function $\boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\Omega)$, integrate over Ω and apply integration by parts to the Stokes (stress) term, i.e.

$$(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v})_\Omega - (\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}, p), \boldsymbol{v})_\Omega = 0$$

$$\Rightarrow \qquad (\partial_t \boldsymbol{u} + u \cdot \nabla \boldsymbol{u}, \boldsymbol{v})_\Omega + (\boldsymbol{\sigma}(\boldsymbol{u}, p), \nabla \boldsymbol{v})_\Omega - (\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, \boldsymbol{v})_{\partial\Omega} = 0.$$

We can then split the boundary into the three different parts, and using that the normal stress vanishes on the Neumann boundary we have

$$(\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, \boldsymbol{v})_{\Gamma_\circ} = (\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v})_\Omega + (\boldsymbol{\sigma}(\boldsymbol{u}, p), \nabla \boldsymbol{v})_\Omega - (\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, \boldsymbol{v})_{\Gamma_D}.$$

Using appropriate test functions $\boldsymbol{v}$, we can therefore transfer integrals over the unfitted boundary into an integral over the volume plus an integral on the fitted boundary where there is no ghost-penalty stabilisation. The Stokes part vanishes for the drag and lift functionals since we can choose $\boldsymbol{v}$ to be constant.

Test have shown that while this approach is slightly more accurate than the single unfitted boundary integral evaluation, the observed instabilities do not vanish and are sometimes even more pronounced.   ▲

Example with Topology Change

As a further investigation into the capabilities of the IE_WRL method applied to the Navier-Stokes equations, we consider examples with topological changes in the geometry. While we do not have any theoretical justification to expect anything in this situation, we can be hopeful that the method yields reasonable results since we based the method on the work in [LO19], where the authors also presented a numerical example with topology changes for their method for convection-diffusion problems. Furthermore, if the method produces reasonable results, this would be advantageous compared with ALE methods since we know that the ALE approach breaks down if changes in topology occur.

We note that this is non-physical behaviour since the Navier-Stokes equations do not permit contact in two dimensions [Hil07]. However, since it is very difficult to resolve the problem sufficiently in practice, contact can easily occur numerically. It is therefore advantageous if the method can cope with this.

Specifically, we shall consider two situations. First, we take a situation where we have contact between two solids in the fluid. However, since we can treat the level sets of the solid regions separately, using the multiple level set functionality of ngsxfem [Leh+21], and only the smooth level sets touch, it is possible that the P^1 interpolated discrete level set functions never overlap or touch each other, even if each domain has a contribution on a single element. Secondly, we take a situation where there is a real topology change, in the sense that the two solid domains merge and separate again.

Set-Up   We extend the previous example of the *flow around a moving cylinder*. Taking the same background domain $\widetilde{\Omega}$, we consider two cylindrical obstacles $\mathcal{S}_1(t)$ and $\mathcal{S}_2(t)$, each with a diameter of 0.2 and the centres initially at $(0.3, 0.23)$ and $(1.9, 0.23)$, respectively. The fluid domain is then given by $\mathcal{F}(t) = \widetilde{\Omega} \setminus (\mathcal{S}_1(t) \cup \mathcal{S}_2(t))$. A sketch of this initial configuration can be seen in Figure 4.16.

As quantities of interest, we will look at the two components of the forces acting on each of the two cylinders. As we cannot compute an ALE reference solution for problems with topology changes, we also look at the fulfilment of the divergence constraint of the finite element solutions.

In the following, we will exclusively use the IE_WRL method, with the extension only applied in the strictly necessary regions.
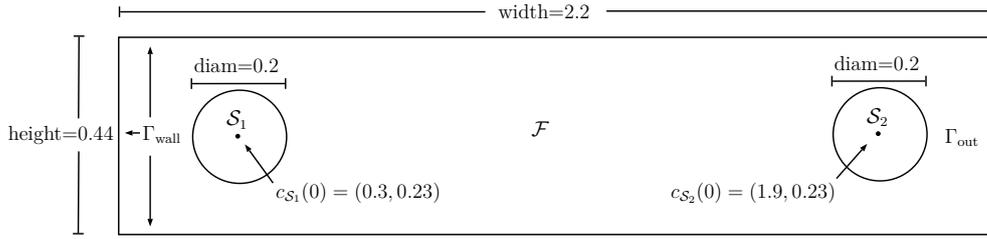
Figure 4.16: Initial configuration for the flow around two cylinders with contact problems.
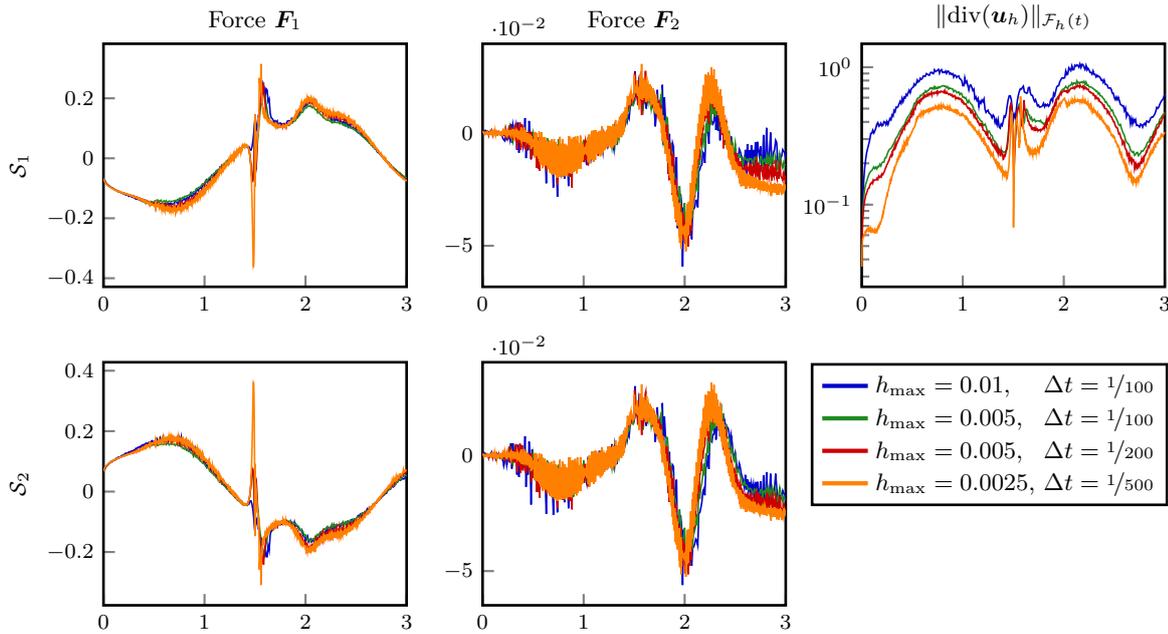
Figure 4.17: Drag and lift functionals on the two cylinders and the divergence error of the fluid velocity for two moving cylinders with contact at $t = 1.5$.

Contact We take the time interval $[0,3]$, and prescribe the centres of the two cylinder to be $c_{\mathcal{S}_1}(t) = (0.65 + 0.35\sin(2\pi(t - 0.75)/3), 0.23)$ and $c_{\mathcal{S}_2}(t) = (1.55 + 0.35\sin(2\pi(t + 0.75)/3), 0.23)$, respectively. As a result, contact occurs between the two cylinders at $t = 1.5$. We consider the mesh/time-step pairs $(0.01, 1/100)$, $(0.005, 1/100)$, $(0.005, 1/200)$ and $(0.0025, 1/500)$. The resulting quantities of interest can be seen in Figure 4.17.

We first note that the simulations worked without any problems. Looking at the drag and lift forces, we see the same instabilities as in the single moving cylinder example. However, the values appear reasonable. There is a spike in the drag at $t = 1.5$, i.e., when the contact occurs, and this spike becomes more pronounced on the finer meshes. We consider this to be reasonable since contact is unphysical for the Navier-Stokes equations in 2d and the forces in the horizontal axis act against the motion trying to create the contact. This shows that we require both very fine meshes and a small time-step to realise the limiting behaviour of a rebound without contact.

Looking at the divergence constraint, we see that it is not fulfilled well. However, we also see that it decreases with smaller meshes and time-step sizes and that it also depends on the speed of the cylinder at a given time.
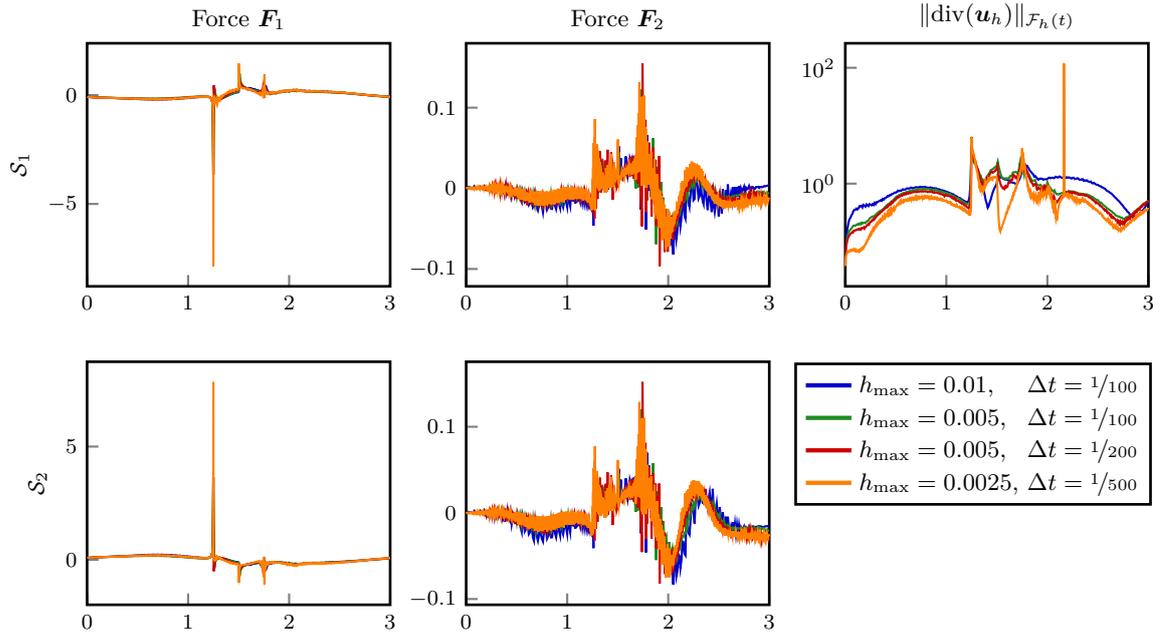
Figure 4.18: Drag and lift functionals on the two cylinders and the divergence error of the fluid velocity for two moving cylinders with overlap during $t \in [1.25061304, 1.74938696]$.

Overlap   We take the time interval $[0, 3]$, and prescribe the motion of the two cylinders such that the centres cylinders of the two cylinders come within a distance of 0.1 of each other at $t = 1.5$, i.e., two cylinders overlap each other for a period of time. The centres of the cylinders are then given by $c_{\mathcal{S}_1}(t) = (0.675 + 0.375 \sin(2\pi(t - 0.75)/3), 0.23)$ and $c_{\mathcal{S}_2}(t) = (1.525 + 0.375 \sin(2\pi(t + 0.75)/3), 0.23)$, respectively. The two cylinders therefore have some overlap in during the time interval $[1.25061304, 1.74938696]$. We note that the drag and lift functionals are no longer computed on the entire boundary $\partial \mathcal{S}_i$ but only on the interface section, $\mathcal{I}_i = \partial \mathcal{S}_i \cap (\widetilde{\Omega} \setminus \mathcal{S}_j)$, i.e., on the part of the boundary of $\mathcal{S}_i$, which is inside the complement of $\mathcal{S}_j$ for $i \neq j$. We consider the same mesh/time-step pairs as before. The resulting quantities of interest can be seen in Figure 4.18.

In general, the results are comparable to the results where contact but no overlap occurred. However, especially in the drag functionals $\boldsymbol{F}_1$, we see three big spikes. This occurs at the points in time where the two cylinders begin to overlap ($t \approx 1.25$), the point where the cylinders change direction ($t = 1.5$) and the time point where the two cylinders separate again. Furthermore, we note that the divergence constraint is almost completely lost at the point of topology change. Fortunately, this recovers again to the levels before the topology change, after the cylinders have separated again. This issue, visible in the results for the phase with overlap, is not surprising since there are issues with a mismatch of the boundary condition at the point where the two level sets cross.

4.5.3 Summary

To summarise the comparison of the different extension techniques, we have established that the IE$_{\mathrm{WRL}}$ method is the only approach that can handle problems where the extension is necessary in a strip that has a width of more than one element. On the other hand, we have seen that the large scaling of the extension ghost-penalty stabilisation terms leads to a deterioration of the velocity around the solid interface such that forces cannot be computed accurately. Therefore, the method must be implemented carefully so that the extension is only applied in the area where it is strictly necessary. Furthermore, scaling the extension and stability ghost-penalties with different parameters also seems to be a good approach to reducing

the ghost-penalty operator's influence. Using larger time-steps also appears to be helpful to avoid large instabilities in the evaluation of the force functional.

From the numerical experiments with topology changes, we can conclude the following. First, our implicit extension method is stable if contact occurs and even continues to work when we have an overlap of solid domains. Second, very highly resolved spatial meshes and time-steps are needed to realise the physical forces, which prevent contact in the context of the two-dimensional Navier-Stokes equations. Finally, we have observed that there can be issues in the fulfilment of the divergence constraint, especially during the unphysical merger of the two solid objects, but in general, this is recovered up to the pre-contact accuracy once the body motion is again physical.

Analysis of the Eulerian Time-Stepping for Coupled Fluid-Rigid Body Interactions

Structure of Chapter An Eulerian time-stepping scheme for the Stokes system coupled to rigid body motion (2.13), based on our work in chapter 4, is developed and analysed. The stability of the scheme is shown in the continuous, semi-discrete and fully discrete cases. The fully discrete error analysis is performed for the reduced case of the heat equation as the partial differential equation in the bulk domain.

Contents of Chapter

5.1 Method

We aim to apply the Eulerian time-stepping method, derived and analysed in chapter 4, not only in cases of prescribed motion of the solid, such as in section 4.5, but also in cases where the rigid body motion is driven by both the fluid forces and the acceleration due to gravity. Therefore, let us consider the time-dependent Stokes problem coupled to translational solid motion of a rigid particle (2.13).

To rewrite this into a weak formulation, let us consider the Sobolev and Lebesgue spaces

$$\boldsymbol{V}(t) \coloneqq \{\boldsymbol{v} \in \boldsymbol{\mathcal{H}}^1(\mathcal{F}(t)) \mid \boldsymbol{v}|_\Gamma = 0\}, \qquad Q(t) \coloneqq \mathcal{L}_0^2(\mathcal{F}(t)), \qquad \boldsymbol{N}(t) \coloneqq \boldsymbol{\mathcal{H}}^{-1/2}(\mathcal{I}(t)).$$

The weak formulation of (2.13) then reads as follows.

Problem P8. Find $(\boldsymbol{u}(t), p(t), U(t), \boldsymbol{\lambda}(t)) \in \boldsymbol{V}(t) \times Q(t) \times \mathbb{R}^d \times \boldsymbol{N}(t)$ such that

$$(\partial_t \boldsymbol{u}, \boldsymbol{v})_{\mathcal{F}(t)} + \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_{\mathcal{F}(t)} - (p, \nabla \cdot \boldsymbol{v})_{\mathcal{F}(t)} + (\boldsymbol{\lambda}, \boldsymbol{v})_{\mathcal{I}(t)} = 0 \tag{5.1a}$$

$$- (q, \nabla \cdot \boldsymbol{u})_{\mathcal{F}(t)} \qquad\qquad = 0 \tag{5.1b}$$

$$(\boldsymbol{\mu}, \boldsymbol{u})_{\mathcal{I}(t)} = (\boldsymbol{\mu}, U)_{\mathcal{I}(t)} \tag{5.1c}$$

$$(\frac{\mathrm{d}}{\mathrm{d}t} U, V)_2 = (\boldsymbol{F}_{\mathrm{ext}} + \boldsymbol{F}, V)_2 \tag{5.1d}$$

for all $(\boldsymbol{v}, q, V, \boldsymbol{\mu}) \in \boldsymbol{V}(t) \times Q(t) \times \mathbb{R}^d \times \boldsymbol{N}(t)$.

Here, $(\cdot, \cdot)_2$ denotes the scalar product in $\mathbb{R}^d$ and $\boldsymbol{F}_{\mathrm{ext}} \coloneqq \boldsymbol{F}_{\mathrm{buoancy}} + \boldsymbol{F}_{\mathrm{gravity}} = (\rho_{\mathcal{S}} - \rho_{\mathcal{F}}) \operatorname{vol}(\mathcal{S})\boldsymbol{g}$ is the sum of the external body forces acting on the solid due to gravity and buoyancy. Using integration

by parts in the momentum balance equation (5.1a), it is immediately apparent that the solution of the Lagrange multiplier is $\boldsymbol{\lambda} = -\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}$.

Remark 5.1 (Enforcement of boundary condition on solid).   Including the Dirichlet boundary condition at the interface to enforce continuity of the velocity in the space $\boldsymbol{V}(t)$ would be unnatural since the value which has to be enforced at $\mathcal{I}(t)$ is not known a priori. The boundary condition must therefore be enforced externally. In the previous chapters, we have worked with Nitsche's method to enforce Dirichlet boundary conditions on the unfitted interface. This has the advantage of not needing to discretise the space $\mathcal{H}^{-1/2}(\mathcal{I})$. Nevertheless, the Lagrange multiplier approach has also been applied in the context of CutFEM [BH10] and fictitious domain methods [GPP94b; Glo+00]. While we continue to implement Dirichlet conditions in practice using Nitsche's method, we consider the Lagrange multiplier formulation here since the Lagrange multiplier solution is the normal stress on the solid's boundary, which in turn makes the analysis easier to present and read.   ▲

We can equivalently reformulated the system (5.1) as

$$(\partial_t \boldsymbol{u}, \boldsymbol{v})_{\mathcal{F}(t)} + \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_{\mathcal{F}(t)} - (p, \nabla \cdot \boldsymbol{v})_{\mathcal{F}(t)} - (q, \nabla \cdot \boldsymbol{u})_{\mathcal{F}(t)}$$
$$+ (\boldsymbol{\lambda}, \boldsymbol{v})_{\mathcal{I}(t)} - (\boldsymbol{\mu}, \boldsymbol{u} - U)_{\mathcal{I}(t)} + (\partial_t U, V)_2 = (\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}, V)_2. \quad (5.2)$$

With this, we can then formulate our stability estimate for the fluid and solid velocity solutions.

Lemma 5.2 (*Stability of the continuous weak formulation*).   *For the fluid and solid velocity solution* $(\boldsymbol{u}, U)$ *of* (5.2), *we have that*

$$\|\boldsymbol{u}(t)\|_{\mathcal{F}(t)}^2 + \|U(t)\|_2^2 + \int_0^t \nu\|\nabla\boldsymbol{u}(s)\|_{\mathcal{F}(s)}^2 \, \mathrm{d}s \le \|\boldsymbol{u}(0)\|_0^2 + \|U(0)\|_2^2 + t\frac{c_{\mathcal{F}}^2 c_P^2}{\nu|\mathcal{I}|}\|\boldsymbol{F}_{ext}\|_2^2. \quad (5.3)$$

Proof. Testing (5.2) with $(\boldsymbol{v}, q, \boldsymbol{\mu}, V) = (\boldsymbol{u}, -p, \boldsymbol{\lambda}, U)$, we have

$$\underbrace{(\partial_t \boldsymbol{u}, \boldsymbol{u})_{\mathcal{F}(t)}}_{\mathfrak{T}_1} + \nu\|\nabla\boldsymbol{u}\|_{\mathcal{F}(t)}^2 \underbrace{-(\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, U)_{\mathcal{I}(t)}}_{\mathfrak{T}_2} + \underbrace{(\frac{\mathrm{d}}{\mathrm{d}t}U, U)_2}_{\mathfrak{T}_3} = \underbrace{(\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}, U)_2}_{\mathfrak{T}_4}. \quad (5.4)$$

Let us therefore reformulate and derive estimates for the individual terms.

<u>Fluid time derivative</u> $\mathfrak{T}_1 = (\partial_t \boldsymbol{u}, \boldsymbol{u})_{\mathcal{F}(t)}$. We recall Reynolds transport theorem for moving domains:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\omega(t)} \phi(\boldsymbol{x}, t) \, \mathrm{d}\boldsymbol{x} = \int_{\omega(t)} \partial_t \phi(\boldsymbol{x}, t) \, \mathrm{d}\boldsymbol{x} + \int_{\partial\omega(t)} \phi(\boldsymbol{x}, t)\boldsymbol{w} \cdot \boldsymbol{n} \, \mathrm{d}s, \quad (5.5)$$

where ϕ is some scalar quantity and $\boldsymbol{w}$ is the velocity of the moving interface $\partial\omega(t)$, c.f. for example [Joh16, Remark 2.3]. Setting $\phi = \boldsymbol{u} \cdot \boldsymbol{u}$ in (5.5), using the fact that U is both the velocity of the moving interface and the trace of $\boldsymbol{u}$ on this interface, we have

$$(\partial_t \boldsymbol{u}, \boldsymbol{u})_{\mathcal{F}(t)} = \frac{1}{2}\Big(\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{u}\|_{\mathcal{F}(t)}^2 - \int_{\mathcal{I}(t)} (\boldsymbol{u} \cdot \boldsymbol{u})U \cdot \boldsymbol{n} \, \mathrm{d}s\Big).$$

Now, since we have that $\boldsymbol{u}|_{\mathcal{I}} = U$, we may interchange U and $\boldsymbol{u}$ in the integral over the interface. Using the fact that U is constant in space, we then find using the divergence theorem that

$$-\int_{\mathcal{I}(t)} (\boldsymbol{u} \cdot \boldsymbol{u})U \cdot \boldsymbol{n} \, \mathrm{d}s = -\|U\|_2^2 \int_{\mathcal{I}(t)} \boldsymbol{u} \cdot \boldsymbol{n} \, \mathrm{d}s = -\|U\|_2^2 \int_{\mathcal{F}(t)} \nabla \cdot \boldsymbol{u} \, \mathrm{d}\boldsymbol{x} = 0.$$

As a result, we have

$$(\partial_t \boldsymbol{u}, \boldsymbol{u})_{\mathcal{F}(t)} = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{u}\|_{\mathcal{F}(t)}^2. \quad (5.6)$$

Stress Coupling $\mathfrak{T}_2 = -(\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, U)_{\mathcal{I}(t)}$. Using the fact that U is constant in space, we find that

$$-(\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}, U)_{\mathcal{I}(t)} = -\int_{\mathcal{I}(t)} \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} \,\mathrm{d}s \cdot U = (\boldsymbol{F}, U)_2.$$

This then cancels with the drag contribution on the right-hand side of (5.4).

Solid time derivative $\mathfrak{T}_3 = (\frac{\mathrm{d}}{\mathrm{d}t}U, U)_2$: Since U is independent of space, we immediately have

$$(\partial_t U, U)_2 = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\|U\|_2^2. \tag{5.7}$$

External solid forces $\mathfrak{T}_{4a} = (\boldsymbol{F}_{\text{ext}}, U)_2$. Under our assumption that $\boldsymbol{F}_{\text{ext}}$ is constant in space and using that $\boldsymbol{u} = U$ on $\mathcal{I}(t)$, we can rewrite the term as

$$(\boldsymbol{F}_{\text{ext}}, U)_2 = \frac{1}{|\mathcal{I}|}(\boldsymbol{F}_{\text{ext}}, \boldsymbol{u})_{\mathcal{I}(t)}.$$

Using the trace and Poincaré estimates, we then find

$$|(\boldsymbol{F}_{\text{ext}}, U)_2| \leq \frac{1}{\sqrt{|\mathcal{I}|}}\|\boldsymbol{F}_{\text{ext}}\|_2 c_{\mathcal{F}}\|\boldsymbol{u}\|_{\mathcal{H}^1(\mathcal{F}(t))} \leq \frac{1}{\sqrt{|\mathcal{I}|}}\|\boldsymbol{F}_{\text{ext}}\|_2 c_{\mathcal{F}} c_P\|\nabla\boldsymbol{u}\|_{\mathcal{F}(t)},$$

where with an abuse of notation $c_P = \max\{2, c_P\}$. Note that the Poincaré inequality is applicable, since $\boldsymbol{u}|_\Gamma = 0$, c.f. [Joh16, Remark A.37]. With a weighted Young's inequality, we then have

$$|(\boldsymbol{F}_{\text{ext}}, U)_2| \leq \frac{c_{\mathcal{F}}^2 c_P^2}{2\nu|\mathcal{I}|}\|\boldsymbol{F}_{\text{ext}}\|_2^2 + \frac{\nu}{2}\|\nabla\boldsymbol{u}\|_{\mathcal{F}(t)}^2. \tag{5.8}$$

Having derived estimates for the individual terms in (5.4), we insert (5.6), (5.7) and (5.8), and use the observation that $\mathfrak{T}_2$ cancels with $(\boldsymbol{F}, U)_2$. This then gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{u}\|_{\mathcal{F}(t)}^2 + \nu\|\nabla\boldsymbol{u}\|_{\mathcal{F}(t)}^2 + \frac{\mathrm{d}}{\mathrm{d}t}\|U\|_2^2 \leq \frac{c_{\mathcal{F}}^2 c_P^2}{\nu|\mathcal{I}|}\|\boldsymbol{F}_{\text{ext}}\|_2^2. \tag{5.9}$$

Integrating with respect to time then gives the result, having used that gravity is constant in time. $\qquad\square$

5.2 Discretisation and Stability Analysis

As the aim is to apply the Eulerian time-stepping method to the coupled problem (2.13), we proceed as before and begin with the temporal semi-discretisation.

5.2.1 Temporal Semi-Discretisation

Let us consider a finite time interval $[0, T]$, and a fixed time-step $\Delta t = T/N$ for some fix $N \in \mathbb{N}$. We then denote the time $t^n = n\Delta t$ and set $\mathcal{F}^n$ and $\mathcal{I}^n$ the fluid and interface domains at time t^n. Furthermore, we define the spaces $\boldsymbol{V}^n, Q^n$ and $\boldsymbol{N}^n$ to be the corresponding velocity, pressure and Lagrange multiplier space on $\mathcal{F}^n$ and $\mathcal{I}^n$, respectively. We note that the position of the solid (and therefore also the interface and fluid region) is determined through the discretised solid ODE. The ODE governing the solid motion, in turn, depends on the forces resulting from the discretised fluid problem. As a result, there is a mismatch between $\mathcal{F}^n$ and $\mathcal{F}(t^n)$. We will quantify this discrepancy in section 5.3 below. Furthermore, we define the diffusion and pressure-coupling bilinear forms as

$$a^n(\boldsymbol{u}, \boldsymbol{v}) := \nu\int_{\mathcal{F}^n} \nabla\boldsymbol{u} : \nabla\boldsymbol{v}\,\mathrm{d}\boldsymbol{x} \quad \text{and} \quad b^n(p, \boldsymbol{v}) := -\int_{\mathcal{F}^n} q\nabla\cdot\boldsymbol{v}\,\mathrm{d}\boldsymbol{x}.$$

We further take a smooth extension operator

$$\mathcal{E} \colon \mathcal{L}^2(\mathcal{F}(t)) \to \mathcal{L}^2(\mathcal{O}_\delta(\mathcal{F}(t)))$$

as in subsection 4.2.1 such that $\mathcal{F}^n \subset \mathcal{O}_\delta(\mathcal{F}^{n-1})$ for $n = 0, \ldots, N$. This is again achieved by setting $\delta = c_\delta \boldsymbol{w}_\infty^n \Delta t$ with $\boldsymbol{w}_\infty^n := \max_{t\in[0,T]} |U \cdot \boldsymbol{n}|$. The fact that this is bounded follows from the stability estimate below. Furthermore, bounds on the terminal settling speed are available in the literature, for example [Wei72] for slender bodies. We refer to Assumption A1 for the details of the necessary regularity assumption of this extension operator.

The temporally semi-discrete scheme then reads as follows: For $i = 1, \ldots, N$, find $(\boldsymbol{u}^i, p^i, U^i, \boldsymbol{\lambda}^i) \in \boldsymbol{V}^i \times Q^i \times \mathbb{R}^d \times \boldsymbol{N}^i$ such that

$$\frac{1}{\Delta t}(\boldsymbol{u}^i, \boldsymbol{v})_{\mathcal{F}^i} + a^i(\boldsymbol{u}^i, \boldsymbol{v}) + b^i(p^i, \boldsymbol{v}) + b^i(q, \boldsymbol{u}^i) + (\boldsymbol{\lambda}^i, \boldsymbol{v})_{\mathcal{I}^i} - (\boldsymbol{\mu}, \boldsymbol{u}^i - U^i)_{\mathcal{I}^i} + \frac{1}{\Delta t}(U^i, V)_2$$
$$= (\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}, V)_2 + \frac{1}{\Delta t}(\mathcal{E}\boldsymbol{u}^{i-1}, \boldsymbol{v})_{\mathcal{F}^i} + \frac{1}{\Delta t}(U^{i-1}, V)_2 \quad (5.10)$$

for all $(\boldsymbol{v}, q, V, \boldsymbol{\mu}) \in \boldsymbol{V}^i \times Q^i \times \mathbb{R}^d \times \boldsymbol{N}^i$. For this we then have the following stability result, analogous to Lemma 5.2.

Lemma 5.3 (*Stability of the semi-discrete velocity solution*). *Let $\{\boldsymbol{u}^i, U^i\}_{i=1}^N$ be the fluid and solid velocity solution to* (5.10) *with compatible initial data $(\boldsymbol{u}^0, U^0) \in \boldsymbol{V}^0 \times \mathbb{R}^d$. We then have for $n = 1, \ldots, N$ the stability estimate*

$$\|\boldsymbol{u}^n\|_{\mathcal{F}^n}^2 + \|U^n\|_2^2 + \Delta t \sum_{i=1}^n \frac{\nu}{2}\|\nabla \boldsymbol{u}^i\|_{\mathcal{F}^i}^2$$
$$\leq \exp\left(\frac{c_{L5.3a}}{\nu}t^n\right)\left[\|\boldsymbol{u}^0\|_{\Omega^0}^2 + \|U^0\|_2^2 + \frac{\nu \Delta t}{2}\|\nabla \boldsymbol{u}^0\|_{\Omega^0} + \frac{c_{L5.3b}\|\boldsymbol{F}_{ext}\|_2^2}{\nu}t^n,\right] \quad (5.11)$$

with constants $c_{L5.3a}, c_{L5.3b} > 0$ independent of the time-step and the number of steps n.

Proof. We test (5.10) with $(\boldsymbol{v}, p, V, \boldsymbol{\mu}) = 2\Delta t(\boldsymbol{u}^i, -p^i, U^i, \boldsymbol{\lambda}^i)$ to obtain

$$2(\boldsymbol{u}^i - \mathcal{E}\boldsymbol{u}^{i-1}, \boldsymbol{u}^i)_{\mathcal{F}^i} + 2(U^i - U^{i-1}, U^i)_2 + 2\Delta t\nu\|\nabla \boldsymbol{u}^i\|_{\mathcal{F}^i} + 2\Delta t(\boldsymbol{\lambda}^i, U^i)_{\mathcal{I}^i}$$
$$= 2\Delta t(\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}, U^i)_2. \quad (5.12)$$

For the two terms originating from the approximation of the time-derivative, we have the polarisation identity $2(\boldsymbol{u}^i - \mathcal{E}\boldsymbol{u}^{i-1}, \boldsymbol{u}^i)_{\mathcal{F}^i} = \|\boldsymbol{u}^i\|_{\mathcal{F}^i}^2 + \|\boldsymbol{u}^i - \mathcal{E}\boldsymbol{u}^{i-1}\|_{\mathcal{F}^i}^2 - \|\mathcal{E}\boldsymbol{u}^{i-1}\|_{\mathcal{F}^i}^2$. For the Lagrange multiplier and external forcing terms we have as in the proof of Lemma 5.2 above that

$$(\boldsymbol{\lambda}^i, U^i)_{\mathcal{I}^i} = (\boldsymbol{F}, U^i)_2 \qquad \text{and} \qquad (\boldsymbol{F}_{\text{ext}}, U^i)_2 \leq \frac{c_1}{2\nu}\|\boldsymbol{F}_{\text{ext}}\|_2^2 + \nu\|\nabla \boldsymbol{u}^i\|_{\mathcal{F}^i}^2$$

with $c_1 = \frac{c_{\mathcal{F}}^2 c_P^2}{|\mathcal{I}|}$. Using these equalities and estimates, we get from (5.12) that

$$\|\boldsymbol{u}^i\|_{\mathcal{F}^i}^2 + \|U^i\|_2^2 + \nu \Delta t\|\nabla \boldsymbol{u}^i\|_{\mathcal{F}^i}^2 \leq \|\mathcal{E}\boldsymbol{u}^{i-1}\|_{\mathcal{F}^i}^2 + \|U^{i-1}\|_2^2 + \frac{c_1 \Delta t}{\nu}\|F\|_2^2. \quad (5.13)$$

Now, for arbitrary $\varepsilon > 0$, we have

$$\|\mathcal{E}\boldsymbol{u}\|_{\mathcal{O}_\delta(\Omega)} \leq (1 + (1 + \varepsilon^{-1})\delta c')\|\boldsymbol{u}\|_\Omega^2 + \delta c''\varepsilon\|\nabla \boldsymbol{u}\|_\Omega^2,$$

c.f. [LO19, Lemma 3.5]. Then with the choice of δ and $\varepsilon = \frac{\nu}{2c''c_\delta \boldsymbol{w}_\infty^n}$, it follows

$$\|\mathcal{E}\boldsymbol{u}^{i-1}\|_{\mathcal{F}^i}^2 \leq \|\mathcal{E}\boldsymbol{u}^{i-1}\|_{\mathcal{O}_\delta(\mathcal{F}^{i-1})}^2 \leq \left(1 + \left(1 + \frac{2c''c_\delta \boldsymbol{w}_\infty^n}{\nu}\right)c'c_\delta \boldsymbol{w}_\infty^n \Delta t\right)\|\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \frac{\nu\Delta t}{2}\|\nabla\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2$$

$$\leq (1 + \frac{c_2\Delta t}{\nu})\|\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \frac{\nu\Delta t}{2}\|\nabla\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2.$$

Applying this to (5.13) then gives the estimate

$$\|\boldsymbol{u}^i\|_{\mathcal{F}^i}^2 + \|U^i\|_2^2 + \frac{\nu\Delta t}{2}\|\nabla\boldsymbol{u}^i\|_{\mathcal{F}^i}^2$$

$$\leq (1 + \frac{c_2\Delta t}{\nu})\|\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \frac{\nu\Delta t}{2}\|\nabla\boldsymbol{u}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \|U^{i-1}\|_2^2 + \frac{c_1\Delta t}{\nu}\|\boldsymbol{F}_{\text{ext}}\|_2^2.$$

Summing this over $i = 1, \ldots, n$ leads to

$$\|\boldsymbol{u}^n\|_{\mathcal{F}^n}^2 + \|U^n\|_2^2 + \frac{\nu\Delta t}{2}\sum_{i=1}^{n}\|\nabla\boldsymbol{u}^i\|_{\mathcal{F}^i}^2$$

$$\leq \|\boldsymbol{u}^0\|_{\mathcal{F}^0}^2 + \frac{\nu\Delta t}{2}\|\nabla\boldsymbol{u}^0\|_{\mathcal{F}^0}^2 + \|U^0\|_2^2 + \frac{c_2\Delta t}{\nu}\sum_{k=0}^{n-1}\|\boldsymbol{u}^i\|_{\mathcal{F}^i}^2 + \frac{c_1 t^n}{\nu}\|\boldsymbol{F}_{\text{ext}}\|_2^2. \quad (5.14)$$

Applying a discrete version of Gronwall's lemma, see [HR90, Lemma 5.1], with the choices $c_{L5.3a} = c_2$ and $c_{L5.3b} = c_1$ then proves the claim. $\qquad\square$

Remark 5.4. The form of the discrete version of Gronwall's lemma proven in [HR90, Lemma 5.1] is without the additional term $\|U^n\|_2^2$ on the left-hand side. However, the proof in [HR90] makes it immediately apparent that the form used above also holds. $\qquad\blacktriangle$

Remark 5.5. Compared to Lemma 5.2, we were only able to show stability for the temporally semi-discrete method for very short times due to Gronwall's lemma resulting in the exponential-in-time term. However, we note that this stability result is of the same form as Lemma 4.1 with a constant forcing term. This similarity is noteworthy since the interface motion in chapter 4 is presumed not to depend on the solution itself. $\qquad\blacktriangle$

Remark 5.6. We note that the interface position, an additional unknown in the system (5.10), as recovered from the initial position and the solid velocity, does not explicitly appear in the stability estimate. This is because the extension is made sufficiently large by choice of δ, which guarantees that the interface $\mathcal{I}^n$ is contained in $\mathcal{O}_\delta(\mathcal{F}^{n-1})$. Therefore, the exact position of the interface within $\mathcal{O}_\delta(\mathcal{F}^{n-1})$ does not play a role in the method's stability. $\qquad\blacktriangle$

5.2.2 Fully Discrete Method

We now consider a version of the fully discrete Eulerian time-stepping method from chapter 4 applied to the coupled fluid-solid problem (5.1). That is a CutFEM discretisation of the spatial integrals and an implicit extension of the velocity through extra ghost-penalty stabilisation in a δ-strip around the moving interface. As in subsection 4.2.2, let us consider continuous and inf-sup stable Taylor-Hood spaces on the active mesh for the velocity and pressure, and an appropriate discretisation of the Lagrange multiplier space. The fully discrete method then reads as follows: For $i = 1, \ldots, N$, find $(\boldsymbol{u}_h^i, p_h^i, U_h^i, \boldsymbol{\lambda}_h^i) \in \boldsymbol{V}_h^i \times Q_h^i \times \mathbb{R}^d \times \boldsymbol{N}_h^i$ such that

$$\frac{1}{\Delta t}(\boldsymbol{u}_h^i - \boldsymbol{u}_h^{i-1}, \boldsymbol{v}_h)_{\mathcal{F}_h^i} + a_h^i(\boldsymbol{u}_h^i, \boldsymbol{v}_h) + b_h^i(p_h^i, \boldsymbol{v}_h) + b_h^i(q_h, \boldsymbol{u}_h^i) + \left(\nu + \frac{1}{\nu}\right)i_h^i(\boldsymbol{u}_h^i, \boldsymbol{v}_h) - \frac{1}{\nu}j_h^i(p_h^i, q_h)$$

$$+ (\boldsymbol{\lambda}_h^i, \boldsymbol{v}_h)_{\mathcal{I}_h^i} - (\boldsymbol{\mu}_h, \boldsymbol{u}_h^i - U_h^i)_{\mathcal{I}_h^i} + \frac{1}{\Delta t}(U_h^i - U_h^{i-1}, V_h)_2 = (\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}_h, V_h)_2 \quad (5.15)$$

for all $(\boldsymbol{v}_h, q_h, V_h, \boldsymbol{\mu}_h) \in \boldsymbol{V}_h^i \times Q_h^i \times \mathbb{R}^d \times \boldsymbol{N}_h^i$. Here we denote $\mathcal{F}_h^n$ and $\mathcal{I}_h^n$ as the fluid and interface domains at time t^n resulting from the discretised fluid and solid time derivatives and forces resulting from the spatial discretisation used. Note that even though we consider exact geometry handling in the integrals, there is again a mismatch between $\mathcal{F}(t^n)$ and $\mathcal{F}_h^n$. Note that the subscript "h" indicates that the bilinear forms are defined with respect to the domains from the motion resulting from the discretised problem.

In the following, we shall consider the following norm, which is independent of the mesh-cut position

$$\||\boldsymbol{v}_h\||_{*,i} := \|\nabla \boldsymbol{v}_h\|_{\mathcal{O}_{\delta_h, \mathscr{T}}^i},$$

where $\mathcal{O}_{\delta_h, \mathscr{T}}^i$ is the entire active fluid mesh at time t^i, i.e., those elements which either contribute to the fluid domain or on which the extensions ghost-penalties act, c.f. subsection 4.3.1.

Lemma 5.7 (*Stability of the velocity from the fully discrete method*).   *Let $\{(\boldsymbol{u}_h^i, U_h^i)\}_{i=1}^N$ be the fluid and solid velocity solutions to* (5.15)*. Then for $n = 1, \ldots, N$, we have the stability estimate*

$$
\|\boldsymbol{u}_h^n\|_{\mathcal{F}_h^n}^2 + \|U_h^n\|_2^2 + \sum_{i=1}^n \left[\frac{c\nu\Delta t}{2} \||\boldsymbol{u}_h^i\||_{*,i}^2 + \frac{L}{\nu} i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i) \right]
$$
$$
\leq \exp((c_{L5.7a})t^n) \left[\|\boldsymbol{u}_h^0\|_{\mathcal{F}^0}^2 + \frac{\nu\Delta t}{2} \||\boldsymbol{u}_h^0\||_{*,0}^2 + \frac{L\Delta t}{\nu} i_h^0(\boldsymbol{u}_h^0, \boldsymbol{u}_h^0) + \|U_h^0\|_2^2 + t^n \frac{c_{L5.7b}}{|\mathcal{I}|\nu} \|\boldsymbol{F}_{ext}\|_2^2 \right].
$$

Proof. The proof in very similar to that of Lemma 5.3, with the main difference in the discrete way with which we deal with the $\boldsymbol{u}_h^{i-1}$ term. To this end we test (5.15) with $(\boldsymbol{v}_h, q_h, V_h, \boldsymbol{\mu}_h) = 2\Delta t(\boldsymbol{u}_h^i, -p_h^i, U_h^i, \boldsymbol{\lambda}_h^i)$. This gives with the BDF1 polarisation identity that

$$
\|\boldsymbol{u}_h^i\|_{\mathcal{F}_h^i}^2 + \|U_h^i\|_2^2 + 2\nu\Delta t(\|\nabla \boldsymbol{u}_h^i\|_{\mathcal{F}_h^i}^2 + i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i)) + \frac{2\Delta t L}{\nu} i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i) + j_h^i(p_h^i, p_h^i)
$$
$$
\leq 2\Delta t(F, U_h^i)_{\mathcal{I}_h^i} + \|\boldsymbol{u}_h^i\|_{\mathcal{F}_h^{i-1}}^2 + \|U_h^{i-1}\|_2^2. \quad (5.16)
$$

On the left-hand side of (5.16) we have with the ghost-penalty mechanism that

$$c_1 \||\boldsymbol{u}_h^i\||_{*,i}^2 \leq \|\nabla \boldsymbol{u}_h^i\|_{\mathcal{F}_h^i}^2 + i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i).$$

Furthermore, the pressure ghost-penalty term is positive and can therefore be bounded from below by 0. On the right-hand side, we proceed as above for (5.8) to get

$$2\Delta t(F, U_h^i)_{\mathcal{I}_h^i} \leq \frac{c_2\Delta t}{\nu|\mathcal{I}|} \|F\|_2^2 + \Delta t c_1 \nu \||\boldsymbol{u}_h^i\||_{*,i}.$$

For the $\|\boldsymbol{u}_h^i\|_{\mathcal{F}_h^{i-1}}^2$, we then apply (4.18) together with our choice of δ_h and the appropriate choice of $\varepsilon = \nu/(2c_1 c' c_{\delta_h} \boldsymbol{w}_\infty^{\boldsymbol{n}})$, which gives

$$\|\boldsymbol{u}_h^{i-1}\|_{\mathcal{F}_h^i}^2 \leq \|\boldsymbol{u}_h^{i-1}\|_{\mathcal{O}_{\delta_h}(\mathcal{F}_h^{i-1})}^2 \leq (1 + \frac{c_3\Delta t}{\nu})\|\boldsymbol{u}_h^{i-1}\|_{\mathcal{F}_h^{i-1}}^2 + \frac{c_1\nu}{2} \||\boldsymbol{u}_h^{i-1}\||_{*,i-1}^2 + c_3 h^2 \frac{\Delta t L}{\nu} i_h^{i-1}(\boldsymbol{u}_h^{i-1}, \boldsymbol{u}_h^{i-1}).$$

For h sufficiently small, we have that $c_3 h^2 \leq 1$. Inserting these four estimates into (5.16), we have

$$
\|\boldsymbol{u}_h^i\|_{\mathcal{F}_h^i}^2 + \|U_h^i\|_2^2 + \frac{\nu\Delta t c_1}{2} \||\boldsymbol{u}_h^i\||_{*,i}^2 + \frac{2\Delta t L}{\nu} i_h^i(\boldsymbol{u}_h^i, \boldsymbol{u}_h^i)
$$
$$
\leq \frac{c_2\Delta t}{\nu|\mathcal{I}|} \|F\|_2^2 + (1 + \frac{c_3\Delta t}{\nu})\|\boldsymbol{u}_h^{i-1}\|_{\mathcal{F}_h^{i-1}}^2 + \frac{c_1\nu}{2} \||\boldsymbol{u}_h^{i-1}\||_{*,i-1}^2 + \frac{\Delta t L}{\nu} i_h^{i-1}(\boldsymbol{u}_h^{i-1}, \boldsymbol{u}_h^{i-1}) + \|U_h^{i-1}\|_2^2.
$$

Summing up over $i = 1, \ldots, n$ and applying a discrete version of Gronwall's lemma then concludes the proof. $\qquad\square$

5.3 Semi-Discrete Error Analysis

In this section, we develop an error estimate for the temporally semi-discrete problem. This quantifies the error made by considering the motion from the coupled and discretised solid equation. To this end, we will simplify the equation under consideration to the heat equation. That is, we remove the pressure and the divergence constraint from the system, which can be seen as a reduction of the velocity space to divergence-free functions. This simplification is to make the (already complex structure) easier to present and understand. At the end of the section, we will remark on how the presented analysis can be transferred to the Stokes equation.

Consider the smooth weak formulation

$$(\partial_t \boldsymbol{u}, \boldsymbol{v})_{\mathcal{F}(t^i)} + (\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_{\mathcal{F}(t^i)} + (\boldsymbol{\lambda}, \boldsymbol{v})_{\mathcal{I}(t^i)} - (\boldsymbol{\mu}, \boldsymbol{u} - U)_{\mathcal{I}(t^i)} + (\partial_t U, V)_2 = (\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}, V)_2, \qquad (5.17)$$

with test-functions $(\boldsymbol{v}, V, \boldsymbol{\mu}) \in \boldsymbol{V}(t^i) \times \mathbb{R}^d \times \boldsymbol{N}(t^i)$. The temporally semi-discrete weak formulation is then given by

$$\frac{1}{\Delta t}(\boldsymbol{u}^i - \mathcal{E}\boldsymbol{u}^{i-1}, \boldsymbol{v}^i)_{\mathcal{F}^i} + (\nabla \boldsymbol{u}^i, \nabla \boldsymbol{v}^i)_{\mathcal{F}^i} + (\boldsymbol{\lambda}^i, \boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\mu}^i, \boldsymbol{u}^i - U^i)_{\mathcal{I}^i} + \frac{1}{\Delta t}(U^i - U^{i-1}, V^i)_2$$
$$= (\boldsymbol{F}_{\text{ext}} + \boldsymbol{F}^i, V^i)_2, \quad (5.18)$$

with test-functions $(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i) \in \boldsymbol{V}^i \times \mathbb{R}^d \times \boldsymbol{N}^i$ defined on the spatially smooth domains at time t^i, where the location of the interface is given by the temporally discrete solution. In the following, we shall drop the explicit notation of the extension operator by identifying $\boldsymbol{u}^{n-1}$ with $\mathcal{E}\boldsymbol{u}^{n-1}$ for better readability.

Now, let $C(t^i)$ and C^i denote the position of the solid at time t^i in the temporally smooth and temporally discrete cases, respectively. We then have for the temporally smooth case that

$$C(t^i) = C(0) + \int_0^{t^i} U(t)\,dt = C(t^{i-1}) + \int_{t^{i-1}}^{t^i} U(t)\,dt,$$

while for the temporally discrete case we have

$$C^i = C^{i-1} + \Delta t U^i.$$

The difference $C(t^i) - C^i$ therefore represents the miss-match in the domains resulting from the temporal semi discretisation (including the extension) of the smooth problem (5.17). We can therefore define a mapping from the temporally discrete domain to the exact domain at time t^i. We define this as $\Phi^i : \mathcal{F}^i \to \mathcal{F}(t^i)$ with

$$\Phi^i := \text{id} + (C(t^i) - C^i)\varphi^i, \qquad (5.19)$$

where $\varphi^i \in C^\infty(\mathcal{F}^i)$ such that $\varphi|_{\mathcal{I}^i} = 1$ and $\varphi^i|_{\mathcal{F}^i \setminus \mathcal{O}_\delta(\mathcal{I}^i)} = 0$. We then take this mapping to be invertible. In the following, this mapping takes a similar role as the geometry approximation mapping in chapter 4.

Lemma 5.8. *Let $\|\cdot\|_\infty$ be the $\mathcal{L}^\infty$ norm on $\mathcal{F}^i$. Then for the mapping $\Phi^i : \mathcal{F}^i \to \mathcal{F}(t^i)$ defined in (5.19), it holds that*

$$\|\text{id} - \Phi^i\|_\infty^2 \lesssim \Delta t^2 \sum_{j=0}^i \|\mathbb{U}^j\|_2^2 + \Delta t^3 t^i \|\partial_t U\|_\infty^2, \qquad (5.20)$$

where $\mathbb{U}^i := U(t^i) - U^i$.

Proof. Be definition of the mapping Φ^i and the fact that φ^i is smooth, we have that

$$\|\text{id} - \Phi^i\|_\infty^2 \lesssim \|C(t^i) - C^i\|_2^2 \|\varphi^i\|_\infty^2 \lesssim \|C(t^i) - C^i\|_2^2.$$

Then by the definition of $C(t^i)$ and C^i, we have

$$\|C(t^i) - C^i\|_2^2 \lesssim \|C(t^{i-1}) - C^{i-1}\|_2^2 + \|\int_{t^{i-1}}^{t^k} U(t) - U^i \, dt\|_2^2. \tag{5.21}$$

With respect to the final term, we have for $t \in [t^{i-1}, t^i]$ that $U(t) \leq U(t^i) - t\|\partial_t U\|_{\infty,[t^i-1,t^i]} \leq U(t^i) - \Delta t\|\partial_t U\|_{\infty,[t^i-1,t^i]}$. Therefore, we have

$$\left\| \int_{t^{i-1}}^{t^k} U(t) - U^i \, dt \right\|_2^2 \leq \left\| \int_{t^{i-1}}^{t^k} U(t^i) - \Delta t\|\partial_t U\|_{\infty,[t^i-1,t^i]} - U^i \, dt \right\|_2^2$$
$$\leq \Delta t^2 \|\mathbb{U}^i\|_2^2 + \Delta t^4 \|\partial_t U\|_{\infty,[t^i-1,t^i]}^2. \tag{5.22}$$

Applying (5.22) to (5.21) recursively, it follows

$$\|\mathrm{id} - \Phi^i\|_\infty^2 \lesssim \sum_{j=0}^{i} \left[\Delta t^2 \|\mathbb{U}^j\|_2^2 + \Delta t^4 \|\partial_t U\|_{\infty,[t^{j-1},t^j]}^2 \right] \lesssim \Delta t^2 \sum_{j=0}^{i} \|\mathbb{U}^j\|_2^2 + \Delta t^3 t^i \|\partial_t U\|_\infty^2.$$

$\square$

Lemma 5.9. *For the mapping Φ^i defined in* (5.19), *describing the mismatch between the exact and the discrete domain at time t^i, we have that*

$$\|I - D\Phi^i\|_\infty \simeq \|\mathrm{id} - \Phi^i\|_\infty, \qquad\qquad \|1 - \det(D\Phi^i)\|_\infty \simeq \|\mathrm{id} - \Phi^i\|_\infty,$$
$$\|\mathrm{id} - \Phi^i\|_{\infty,\mathcal{I}^i} \simeq \|\mathrm{id} - \Phi^i\|_\infty, \qquad\qquad \|1 - \det(D\Phi^i)\|_{\infty,\mathcal{I}^i} \simeq \|\mathrm{id} - \Phi^i\|_\infty.$$

Proof. This follows by $\varphi^i \in C^\infty(\mathcal{F}^i)$ and the fact that the remaining components of Φ^i are independent of space. $\square$

Lemma 5.10. *Let*

$$\mathfrak{M}(\mathbb{U}^i, \Delta t) = \Delta t^2 \sum_{j=0}^{i} \|\mathbb{U}^j\|_2^2 + \Delta t^3 t^i \|\partial_t U\|_\infty^2.$$

Then for $\boldsymbol{u} \in \boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))$, it holds that

$$\|\boldsymbol{u} \circ \Phi^i - \mathcal{E}\boldsymbol{u}\|_{\mathcal{F}^i}^2 \lesssim \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^1(\mathcal{F}(t^i))}^2 \mathfrak{M}(\mathbb{U}^i, \Delta t), \tag{5.23}$$
$$\|(\nabla\boldsymbol{u}) \circ \Phi^i - \nabla\mathcal{E}\boldsymbol{u}\|_{\mathcal{F}^i}^2 \lesssim \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))}^2 \mathfrak{M}(\mathbb{U}^i, \Delta t), \tag{5.24}$$
$$\|\boldsymbol{u} \circ \Phi^i - \mathcal{E}\boldsymbol{u}\|_{\mathcal{I}^i}^2 \lesssim \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))}^2 \mathfrak{M}(\mathbb{U}^i, \Delta t), \tag{5.25}$$
$$\|(\partial_{\boldsymbol{n}}\boldsymbol{u}) \circ \Phi^i - \partial_{\boldsymbol{n}}\mathcal{E}\boldsymbol{u}\|_{\mathcal{I}^i}^2 \lesssim \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))}^2 \mathfrak{M}(\mathbb{U}^i, \Delta t). \tag{5.26}$$

Proof. Φ^i maps the approximated interface location $\mathcal{I}^i$ to the exact interface location $\mathcal{I}(t^i)$, and we know that the distance between the two is given by $|C(t^i) - C^i|$ for which we have proven the estimate in the proof of Lemma 5.8. The proof of (5.23)–(5.26) is therefore completely analogous to that of [GOR15, Lemma 7.3]. $\square$

Let us now denote the fluid velocity, Lagrange multiplier and solid velocity errors by

$$\mathbb{E}^i := \boldsymbol{u}(t^i) - \boldsymbol{u}^i \qquad \mathbb{L}^i := \boldsymbol{\lambda}(t^i) - \boldsymbol{\lambda}^i \qquad \mathbb{U}^i := U(t^i) - U^i.$$

Now, if $(\boldsymbol{v}^i, \boldsymbol{V}^i, \boldsymbol{\mu}^i)$ are applicable test-functions for (5.18), then $\boldsymbol{v}_\ell^i = \boldsymbol{v}^i \circ (\Phi^i)^{-1}$, $V_\ell^i = V^i \circ (\Phi^i)^{-1} (= V^i)$ and $\boldsymbol{\mu}_\ell^i = \boldsymbol{\mu}^i \circ (\Phi^i)^{-1}$ are compatible test functions for the temporally smooth problem (5.17). Testing

(5.17) with $(\boldsymbol{v}_\ell^i, V_\ell^i, \boldsymbol{\mu}_\ell^i)$, subtracting (5.18) from the result of this, as well as adding and subtracting appropriate terms, we get the error equation

$$
\frac{1}{\Delta t}(\mathbb{E}^i - \mathbb{E}^{i-1}, \boldsymbol{v}^i)_{\mathcal{F}^i}^2 + (\nabla \mathbb{E}^i, \nabla \boldsymbol{v}^i)_{\mathcal{F}^i} + (\mathbb{L}^i, \boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\mu}^i, \mathbb{E}^i - \mathbb{U}^i)_{\mathcal{I}^i} + \frac{1}{\Delta t}(\mathbb{U}^i - \mathbb{U}^{i-1}, V^i)_2
$$
$$
= (\boldsymbol{F}(t^i) - \boldsymbol{F}^i, V^i)_2 + \frac{1}{\Delta t}(\boldsymbol{u}(t^i) - \boldsymbol{u}(t^{i-1}), \boldsymbol{v}^i)_{\mathcal{F}^i} - (\partial_t \boldsymbol{u}(t^i), \boldsymbol{v}_\ell^i)_{\mathcal{F}(t^i)}
$$
$$
+ (\nabla \boldsymbol{u}(t^i), \nabla \boldsymbol{v}^i)_{\mathcal{F}^i} - (\nabla \boldsymbol{u}(t^i), \nabla \boldsymbol{v}_\ell^i)_{\mathcal{F}(t^i)} + (\boldsymbol{\lambda}(t^i), \boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i), \boldsymbol{v}_\ell^i)_{\mathcal{I}(t^i)}
$$
$$
+ (\boldsymbol{\mu}_\ell^i, \boldsymbol{u}(t^i) - U(t^i))_{\mathcal{I}(t^i)} - (\boldsymbol{\mu}^i, \boldsymbol{u}(t^i) - U(t^i))_{\mathcal{I}^i} + \frac{1}{\Delta t}(U(t^i) - U(t^{i-1}), V^i)_2 - (\partial_t U(t^i), V_\ell^i)_2
$$
$$
= \mathfrak{T}_1 + \mathfrak{T}_2 + \mathfrak{T}_3 + \mathfrak{T}_4 + \mathfrak{T}_5 + \mathfrak{T}_6 =: \mathfrak{E}_c^i(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i). \quad (5.27)
$$

Therefore, the right-hand side represents the consistency error made by the Eulerian time-stepping scheme and the resulting error in the interface/solid position.

The goal is to test (5.27) with $(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i) = (\mathbb{E}^i, \mathbb{U}^i, \mathbb{L}^i)$ in order to get an error estimate along similar lines as the stability estimates before. As we have seen in section 5.1, a number of the Lagrange multiplier terms on the left-hand side cancel each other out when testing symmetrically. Therefore, we look at which terms remain in (5.27) after symmetric testing to not include unnecessary terms. We, therefore, look at the Lagrange multiplier terms on the left-hand side:

$$
(\mathbb{L}^i, \mathbb{E}^i)_{\mathcal{I}^i} - (\mathbb{L}^i, \mathbb{E}^i - \mathbb{U}^i)_{\mathcal{I}^i} = (\mathbb{L}^i, \mathbb{U}^i)_{\mathcal{I}^i} = (\boldsymbol{\lambda}(t^i), \mathbb{U}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}^i, \mathbb{U}^i)_{\mathcal{I}^i}.
$$

Furthermore, the drag terms on the right-hand side can be rewritten as

$$
(\boldsymbol{F} - \boldsymbol{F}^n, \mathbb{U}^i)_2 = (\partial_{\boldsymbol{n}} \boldsymbol{u}(t^i), \mathbb{U}^i)_{\mathcal{I}(t^i)} - (\partial_{\boldsymbol{n}} \boldsymbol{u}^i, \mathbb{U}^i)_{\mathcal{I}^i}.
$$

Now, the Lagrange multiplier solution is known to be $\boldsymbol{\lambda}(t^i) = \partial_{\boldsymbol{n}} \boldsymbol{u}(t^i)$ and $\boldsymbol{\lambda}^i = \partial_{\boldsymbol{n}} \boldsymbol{u}^i$. Combining the above two equalities in the tested error equation means that we are left with

$$
\mathfrak{T}_1' = (\boldsymbol{\lambda}(t^i), \mathbb{U}^i)_{\mathcal{I}(t^i)} - (\boldsymbol{\lambda}(t^i), \mathbb{U}^i)_{\mathcal{I}^i} \quad (5.28)
$$

on the right-hand side.

Now, we define the space-time domain resulting from the motion of the smooth problem as $\mathcal{Q} := \bigcup_{t \in [0,T]} \mathcal{F}(t) \times \{t\}$. We can then bound the error contributions in the following lemma.

Lemma 5.11. *For $(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i) \in \boldsymbol{V}^i \times \mathbb{R}^d \times \boldsymbol{N}^i$ we have that*

$$
|\mathfrak{T}_1'| \lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))} \|\mathbb{U}^i\|_{\mathcal{I}^i} \qquad |\mathfrak{T}_2| \lesssim (\Delta t + \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}) \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \|\nabla \boldsymbol{v}^i\|_{\mathcal{F}^i},
$$
$$
|\mathfrak{T}_3| \lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))} \|\nabla \boldsymbol{v}^i\|_{\mathcal{F}^i}, \qquad |\mathfrak{T}_4| \lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))} \|\nabla \boldsymbol{v}^i\|_{\mathcal{F}^i},
$$
$$
|\mathfrak{T}_5| \lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))} \|\boldsymbol{\mu}^i\|_{\mathcal{I}^i}, \qquad |\mathfrak{T}_6| \lesssim \Delta t \|\partial_t^2 U\|_{\infty, [0, t^i]} \|V^i\|_2,
$$

with $\mathfrak{T}_1'$ and $\mathfrak{T}_i$ $i = 2, \ldots, 6$, as defined in (5.28) and (5.27), respectively.

Proof. For the modified Lagrange multiplier term $\mathfrak{T}_1'$, we use that $\mathbb{U}^i \in \mathbb{R}^d$ such that $\mathbb{U}^i = \mathbb{U}^i \circ (\Phi^i)^{-1}$. Denoting $J := \det(D\Phi^i)$, we then get the estimate

$$
|\mathfrak{T}_1'| = |(\boldsymbol{\lambda}(t^i)) \circ \Phi^i, J\mathbb{U}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i), \mathbb{U}^i)_{\mathcal{I}^i}|
$$
$$
= |(\boldsymbol{\lambda}(t^i) \circ \Phi^i - \boldsymbol{\lambda}(t^i), J\mathbb{U}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i), (J-1)\mathbb{U}^i)_{\mathcal{I}^i}|
$$
$$
\lesssim \|\boldsymbol{\lambda}(t^i) \circ \Phi^i - \boldsymbol{\lambda}(t^i)\|_{\mathcal{I}(t^i)} \|\mathbb{U}^i\|_{\mathcal{I}^i} + \|1 - J\|_{\infty, \mathcal{I}^i} \|\partial_{\boldsymbol{n}} \boldsymbol{u}(t^i)\|_{\mathcal{I}(t^i)} \|\mathbb{U}^i\|_{\mathcal{I}^i}
$$
$$
\lesssim \left[\Delta t^2 \sum_{j=0}^i \|\mathbb{U}^j\|_2^2 + \Delta t^3 t^i \|\partial_t U\|_\infty^2\right]^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))} \|\mathbb{U}^i\|_{\mathcal{I}^i}, \quad (5.29)
$$

where the last estimate results from Lemma 5.9 and Lemma 5.10.

For the fluid time-derivative consistency error term, we have

$$
\begin{aligned}
|\mathfrak{T}_2| &= \Big| \frac{1}{\Delta t}(\boldsymbol{u}(t^i) - \boldsymbol{u}(t^{i-1}), \boldsymbol{v}^i)_{\mathcal{F}^i} - (\partial_t \boldsymbol{u}(t^i), \boldsymbol{v}^i_\ell)_{\mathcal{F}(t^i)} \Big| \\
&= \Big| - \int_{\mathcal{F}^i} \int_{t^{i-1}}^{t^i} \frac{t - t^n}{\Delta t} \partial_t^2 \boldsymbol{u}(t) \, \mathrm{d}t \cdot \boldsymbol{v}^i \, \mathrm{d}\boldsymbol{x} + (\partial_t \boldsymbol{u}(t^i), v^i)_{\mathcal{F}^i} - (\partial_t \boldsymbol{u}(t^i), v^i_\ell)_{\mathcal{F}(t^i)} \Big| \\
&\le \frac{1}{2}\Delta t \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \|\boldsymbol{v}^i\|_{\mathcal{F}^i} + |(\partial_t \boldsymbol{u}(t^i) - \partial_t \boldsymbol{u}(t^i) \circ \Phi^i, \boldsymbol{v}^i)_{\mathcal{F}^i}| \\
&\lesssim (\Delta t + \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}) \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})} \|\boldsymbol{v}^i\|_{\mathcal{F}^i}.
\end{aligned}
$$

See also [LO19, Lemma 5.11]. The claim then follows by an application of the Poincaré inequality.

For the diffusion consistency error, we have with Lemma 5.10, Lemma 5.9 and Lemma 5.8 that

$$
\begin{aligned}
|\mathfrak{T}_3| &= |(\nabla \boldsymbol{u}(t^i), \nabla \boldsymbol{v}^i)_{\mathcal{F}^i} - (\nabla \boldsymbol{u}(t^i), \nabla \boldsymbol{v}^i_\ell)_{\mathcal{F}(t^i)}| \\
&= |(\nabla \boldsymbol{u}(t^i), \nabla \boldsymbol{v}^i)_{\mathcal{F}^i} - (\nabla \boldsymbol{u}(t^i) \circ \Phi^i, J(D\Phi^i)^{-1})\nabla \boldsymbol{v}^i)_{\mathcal{F}^i}| \\
&= |(\nabla(\boldsymbol{u}(t^i) - \boldsymbol{u}(t^i) \circ \Phi^i), J(D\Phi^i)^{-1})\nabla \boldsymbol{v}^i)_{\mathcal{F}^i} + (\nabla \boldsymbol{u}(t^i), (I - J(D\Phi^i)^{-1})\nabla \boldsymbol{v}^i)_{\mathcal{F}^i}| \\
&\le \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))} \|\nabla \boldsymbol{v}^i\|_{\mathcal{F}^i}.
\end{aligned}
$$

For the Lagrange multiplier term, we similarly find by additionally using a trace and the Poincaré inequality that

$$
\begin{aligned}
|\mathfrak{T}_4| &= |(\boldsymbol{\lambda}(t^i), \boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i), \boldsymbol{v}^i_\ell)_{\mathcal{I}(t^i)}| = |(\boldsymbol{\lambda}(t^i), \boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i) \circ \Phi^i, J\boldsymbol{v}^i)_{\mathcal{I}^i}| \\
&= |(\boldsymbol{\lambda}(t^i) - \boldsymbol{\lambda}(t^i) \circ \Phi^i, J\boldsymbol{v}^i)_{\mathcal{I}^i} - (\boldsymbol{\lambda}(t^i), (1 - J)\boldsymbol{v}^i)_{\mathcal{I}^i}| \\
&\lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^3(\mathcal{F}(t^i))} \|\nabla \boldsymbol{v}^i\|_{\mathcal{F}^i}.
\end{aligned}
$$

Furthermore, using the fact that $U \in \mathbb{R}^d$, so that $U \circ \Phi^i = U$, we have for the boundary condition term that

$$
\begin{aligned}
|\mathfrak{T}_5| &= |(\boldsymbol{\mu}^i_\ell, \boldsymbol{u}(t^i) - U(t^i))_{\mathcal{I}(t^i)} - (\boldsymbol{\mu}^i, \boldsymbol{u}(t^i) - U(t^i))_{\mathcal{I}^i}| \\
&= |(J\boldsymbol{\mu}^i, \boldsymbol{u}(t^i) \circ \Phi^i - U(t^i) \circ \Phi^i)_{\mathcal{I}^i} - (\boldsymbol{\mu}^i, \boldsymbol{u}(t^i) - U(t^i))_{\mathcal{I}^i}| \\
&= |((J-1)\boldsymbol{\mu}^i, U(t^i))_{\mathcal{I}^i} + (J\boldsymbol{\mu}^i, \boldsymbol{u}(t^i) \circ \Phi^i - \boldsymbol{u}(t^i))_{\mathcal{I}^i} + ((J-1)\boldsymbol{\mu}^i, \boldsymbol{u}(t^i))_{\mathcal{I}^i}| \\
&\lesssim \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2} \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{H}}^2(\mathcal{F}(t^i))} \|\boldsymbol{\mu}^i\|_{\mathcal{I}^i}.
\end{aligned}
$$

The solid velocity consistency error is bounded similar to $\mathfrak{T}_2$. However, the situation is simpler here because $U \in \mathbb{R}^d$ does not depend on the domain consistency. Therefore,

$$
|\mathfrak{T}_6| = \Big| \frac{1}{\Delta t}(U(t^i) - U(t^{i-1}), V^i)_2 - (\partial_t U(t^i), V^i_\ell)_2 \Big| \lesssim \Delta t \|\partial_t^2 U\|_{\infty, [0, t^i]} \|V^i\|_2.
$$

$\square$

Theorem 5.12 (*Error estimate for the temporally semi-discrete problem*). *If for the solution of the smooth problem (2.13) the regularity $\boldsymbol{u} \in \boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})$ holds, then for Δt sufficiently small, we have the error estimate*

$$
\|\mathbb{E}^n\|_{\mathcal{F}^n}^2 + \Delta t \sum_{i=1}^n \frac{1}{4}\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}^2 + \|\mathbb{U}^n\|_2^2 \le \exp\Big(\Delta t \sum_{i=1}^n \frac{r(\boldsymbol{u})}{1 - \Delta t \, r(\boldsymbol{u})}\Big)\Big[c_{T5.12a}\Delta t \sum_{i=1}^n \Delta t^2 R(\boldsymbol{u}, U)\Big], \quad (5.30)
$$

where

$$r(\boldsymbol{u}) \coloneqq c_{T5.12b}\Big[t^n(3 + \Delta t)(1 + \|\boldsymbol{u}\|^2_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}) + 1\Big]$$

$$R(\boldsymbol{u}, U) \coloneqq \|\boldsymbol{u}\|^2_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})} + \|\boldsymbol{u}\|^4_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})} + \|\partial_t U\|^2_{\infty,[0,t^n]}$$

and $c_{T5.12a}, c_{T5.12b} > 0$ are constants independent of the time-step Δt and the number of time-steps n.

Proof. Testing (5.27) with $(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i) = 2\Delta t(\mathbb{E}^i, \mathbb{U}^i, \mathbb{L}^i)$, we get analogous to (5.16) that

$$
\begin{aligned}
\|\mathbb{E}^i\|^2_{\mathcal{F}^i} &+ \|\mathbb{E}^i - \mathbb{E}^{i-1}\|^2_{\mathcal{F}^i} + 2\Delta t\|\nabla\mathbb{E}^i\|^2_{\mathcal{F}^i} + \|\mathbb{U}^i\|^2_2 \\
&\leq \|\mathbb{E}^{i-1}\|^2_{\mathcal{F}^i} + \|\mathbb{U}^{i-1}\|^2_2 + 2\Delta t(|\mathfrak{T}_1'(\mathbb{U}^i)| + |\mathfrak{T}_2(\mathbb{E}^i)| + |\mathfrak{T}_3(\mathbb{U}^i)| \\
&\qquad\qquad\qquad\qquad + |\mathfrak{T}_4(\mathbb{E}^i)| + |\mathfrak{T}_5(\mathbb{L}^i)| + |\mathfrak{T}_6(\mathbb{U}^i)|) \\
&\leq (1 + c_1\Delta t)\|\mathbb{E}^{i-1}\|^2_{\mathcal{F}^{i-1}} + {}^{\Delta t}/_2\|\nabla\mathbb{E}^{i-1}\|_{\mathcal{F}^{i-1}} + \|\mathbb{U}^{i-1}\|^2_2 \\
&\quad + 2c_2\Delta t\Big[\mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\big\{\|\mathbb{U}^i\|_{\mathcal{I}^i} + \|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} + \|\mathbb{L}^i\|_{\mathcal{I}^i}\big\} \\
&\qquad\qquad + \|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})}\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} + \|\partial_t U\|_{\infty,[0,t^i]}\|\mathbb{U}^i\|_2\Big],
\end{aligned}
\tag{5.31}
$$

where the second inequality is due to (5.29) and Lemma 5.11.

As the aim is to sum this over $i = 1, \ldots, n$, we need to deal with the error terms at time t^i on the right-hand side.

<u>Motion error with solid velocity error:</u> With the weighted Young's inequality and the fact that $\mathbb{U}^i \in \mathbb{R}^d$, we have that

$$2c_2\Delta t\mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\|\mathbb{U}^i\|_{\mathcal{I}^i} \leq \mathfrak{M}(\mathbb{U}^i, \Delta t) + c\Delta t^2|\mathcal{I}^i|\|\boldsymbol{u}(t^i)\|^2_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\|\mathbb{U}^i\|^2_2.$$

<u>Motion error with the gradient of the fluid velocity error:</u> Again, with a weighted Youngs inequality, we have that

$$2c_2\Delta t\mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\|\nabla\mathbb{E}^i\|_{\mathcal{I}^i} \leq \mathfrak{M}(\mathbb{U}^i, \Delta t) + c\Delta t^2\|\boldsymbol{u}(t^i)\|^2_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\|\nabla\mathbb{E}^i\|^2_{\mathcal{F}^i}. \tag{5.32}$$

<u>Motion error with the Lagrange multiplier error:</u> For the Lagrange multiplier term, we note that we have an inf-sup condition:

There exists a constant $\beta > 0$ such that for all $\boldsymbol{\mu} \in \boldsymbol{N}$ it holds that

$$\beta\|\boldsymbol{\mu}\|_{\boldsymbol{N}} \leq \sup_{\boldsymbol{v}\in\boldsymbol{V}}\frac{(\boldsymbol{\mu}, \boldsymbol{v})_{\mathcal{I}}}{\|\nabla\boldsymbol{v}\|_{\mathcal{F}}}. \tag{5.33}$$

The bilinear form $(\cdot, \cdot)_{\mathcal{I}}$ is bounded on $\boldsymbol{N} \times \boldsymbol{V}$ due to the Cauchy-Schwarz, trace and Poincaré estimates. The inf-sup condition (5.33) is therefore an immediate consequence of the unique solvability of the Poisson problem with Lagrange multipliers to enforce boundary condition [Bab73a] and the more general theory for saddle-point problems [BBF13].

Testing (5.27) with $(\boldsymbol{v}^i, V^i, \boldsymbol{\mu}^i) = (\boldsymbol{v}^i, 0, 0)$, we have that

$$
\begin{aligned}
(\mathbb{L}^i, \boldsymbol{v}^i)_{\mathcal{I}^i} &= -\frac{1}{\Delta t}(\mathbb{E}^i - \mathbb{E}^{i-1}, \boldsymbol{v}^i)^2_{\mathcal{F}^i} - (\nabla\mathbb{E}^i, \nabla\boldsymbol{v}^i)_{\mathcal{F}^i} - \mathfrak{T}_2 - \mathfrak{T}_4 - \mathfrak{T}_4 \\
&\leq \Big[\frac{c_P}{\Delta t}\|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\mathcal{F}^i} + \|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} + c(\Delta t + \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})})\Big]\|\nabla\boldsymbol{v}\|_{\mathcal{F}^i}.
\end{aligned}
$$

As a result of the inf-sup condition (5.33), we then have

$$\|\mathbb{L}^i\|_{\mathcal{I}^i} \leq \frac{1}{\beta}\Big[\frac{c_P}{\Delta t}\|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\mathcal{F}^i} + \|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} + c(\Delta t + \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})})\Big]. \tag{5.34}$$

With this, the Lagrange multiplier error contribution in (5.31) then becomes

$$\frac{2c_2}{\beta}\Delta t \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\Big[\frac{c_P}{\Delta t}\|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\mathcal{F}^i} + \|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} + c(\Delta t + \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})})\Big].$$

Now we deal with the additive components of this term separately:

- With the weighted Young's inequality we get for the first term that

$$\frac{2c_2}{\beta}\Delta t \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\frac{c_P}{\Delta t}\|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\mathcal{F}^i} \leq c\mathfrak{M}(\mathbb{U}^i, \Delta t)\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}^2 + \|\mathbb{E}^i - \mathbb{E}^{i-1}\|_{\mathcal{F}^i}^2.$$

- The second term, containing the fluid velocity gradient term, is

$$\frac{2c_2}{\beta}\Delta t \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}$$

for which we already have a bound with (5.32), such that this term can be absorbed there.

- The next term is estimated as

$$\frac{2c_2}{\beta}\Delta t^2 \mathfrak{M}(\mathbb{U}^i, \Delta t)^{1/2}\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}^2 \leq \Delta t \mathfrak{M}(\mathbb{U}^i, \Delta t) + c\Delta t^3\|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}^4.$$

- The final term is then left as

$$c\Delta t \mathfrak{M}(\mathbb{U}^i, \Delta t)\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}^2.$$

Uncoupled gradient of the velocity error: This term appears in the same form in the uncoupled setting. With a weighted Young's inequality we have

$$2c_2\Delta t^2\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})}\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i} \leq c\Delta t^3\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})}^2 + \Delta t\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}^2,$$

such that the $\Delta t\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}^2$-term can be absorbed in the left-hand side of (5.31).

Uncoupled solid velocity error: We proceed again with the weighted Young's inequality to get the estimate

$$2c_2\Delta t\Big[\Delta t\|\partial_t U\|_{\infty,[0,t^i]}\|\mathbb{U}^i\|_2\Big] \leq \Delta t\Big[\Delta t^2 c\|\partial_t U\|_{\infty,[0,t^i]}^2 + \|\mathbb{U}^i\|_2^2\Big].$$

To ease the notation, we shall denote $R_1(\boldsymbol{u}) \coloneqq \|\boldsymbol{u}(t^i)\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}^2$ and $R_2(U) \coloneqq \|\partial_t U\|_{\infty,[0,t^n]}^2$. Then combining the above estimates for the right side terms of (5.31) gives

$$\|\mathbb{E}^i\|_{\mathcal{F}^i}^2 + \Delta t\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}^2 + \|\mathbb{U}^i\|_2^2 \leq (1 + c_1\Delta t)\|\mathbb{E}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \frac{\Delta t}{2}\|\nabla\mathbb{E}^{i-1}\|_{\mathcal{F}^{i-1}}^2 + \|\mathbb{U}^{i-1}\|_2^2$$
$$+ c\Delta t^2 R_1(\boldsymbol{u})\|\mathbb{U}^i\|_2^2 + 2c\Delta t^2 R_1(\boldsymbol{u})\|\nabla\mathbb{E}^i\|_2^2 + \Delta t\|\mathbb{U}^i\|_2^2$$
$$+ c(3 + R_1(\boldsymbol{u}) + \Delta t + \Delta t R_1(\boldsymbol{u}))\mathfrak{M}(\mathbb{U}^i, \Delta t)$$
$$+ c\Delta t^3 R_1(\boldsymbol{u})^2 + c\Delta t^3 R_1(\boldsymbol{u}) + c\Delta t^3 R_2(U).$$

Summing this over $i = 1, \ldots, n$ then gives

$$\|\mathbb{E}^n\|_{\mathcal{F}^n}^2 + \Delta t\sum_{i=1}^n\Big[\frac{1}{2} - c\Delta t R_1(\boldsymbol{u})\Big]\|\nabla\mathbb{E}^i\|_{\mathcal{F}^i}^2 + \|\mathbb{U}^n\|_2^2$$

$$\leq \Delta t\sum_{i=1}^{n-1} c_1\|\mathbb{E}^i\|_{\mathcal{F}^k}^2 + \Delta t\sum_{i=1}^n\Big[c\Delta t R_1(\boldsymbol{u}) + 1\Big]\|\mathbb{U}^i\|_2^2$$

$$+ c(3 + R_1(\boldsymbol{u}) + \Delta t + \Delta t R_1(\boldsymbol{u}))\sum_{i=1}^n\mathfrak{M}(\mathbb{U}^i, \Delta t)$$

$$+ \Delta t\sum_{i=1}^n\Big[c\Delta t^2 R_1(\boldsymbol{u})^2 + c\Delta t^2 R_1(\boldsymbol{u}) + c\Delta t^2 R_2(U)\Big]. \quad (5.35)$$

Before we can apply Gronwall's lemma, we need to deal with the term $\sum_{i=1}^{n} \mathfrak{M}(\mathbb{U}^i, \Delta t)$. For this we estimate

$$\sum_{i=1}^{n} \mathfrak{M}(\mathbb{U}^i, \Delta t) = \sum_{i=1}^{n} \left[\Delta t^3 t^i \|\partial_t U\|_{\infty,[0,t^i]}^2 + \Delta t^2 \sum_{i=1}^{k} \|\mathbb{U}^i\|_2^2 \right] \leq \Delta t^3 \sum_{i=1}^{n} t^i R_2(U) + \Delta t \, t^n \sum_{i=1}^{n} \|\mathbb{U}^i\|_2^2.$$

Inserting this into (5.35) then gives

$$\|\mathbb{E}^n\|_{\mathcal{F}^n}^2 + \Delta t \sum_{i=1}^{n} \left[\frac{1}{2} - c\Delta t R_1(\boldsymbol{u}) \right] \|\nabla \mathbb{E}^i\|_{\mathcal{F}^i}^2 + \|\mathbb{U}^n\|_2^2$$

$$\leq \Delta t \sum_{i=1}^{n-1} c_1 \|\mathbb{E}^i\|_{\mathcal{F}^k}^2 + \Delta t \sum_{i=1}^{n} \left[c\Delta t R_1(\boldsymbol{u}) + ct^n(3 + R_1(\boldsymbol{u}) + \Delta t + \Delta t R_1(\boldsymbol{u})) + 1 \right] \|\mathbb{U}^i\|_2^2$$

$$+ \Delta t \sum_{i=1}^{n} \left[c\Delta t^2 R_1(\boldsymbol{u})^2 + c\Delta t^2 R_1(\boldsymbol{u}) + c\Delta t^2 (1 + t^i) R_2(U) \right].$$

For Δt sufficiently small we have that $\frac{1}{4} < \frac{1}{2} - c\Delta t R_1(\boldsymbol{u})$ so that we we can apply a discrete version of Gronwall's lemma to conclude the proof. $\square$

Remark 5.13 (Necessary regularity for the error estimate). Compared to the case in chapter 4, where the domain motion is given, the error estimate derived here requires a higher regularity of the solution of $\boldsymbol{u} \in \boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})$ resulting from the domain error in the estimate of the Lagrange multiplier consistency error, compared to $\boldsymbol{u} \in \boldsymbol{\mathcal{W}}^{2,\infty}(\mathcal{Q})$ needed in [LO19; WRL21] and section 4.3. We note here that this higher regularity condition was avoided in [WRL21] by a discrete "detour", thereby requiring less regularity but adding additional error terms. ▲

Remark 5.14 (Time-step restriction). The stability estimates in Lemma 5.3 and Lemma 5.7 in chapter 4 and the literature [LO19; BFM19; WRL21] used the form of Gronwall's lemma, in which the time-step did not affect the exponential factor. However, the more general form of Gronwall's lemma had to be applied here due to the error resulting from the coupling between the solution and the domain motion. As a result, the above error estimate only holds under a time-step restriction dependent on $\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{3,\infty}(\mathcal{Q})}$. We note, however, that such a restriction can also be seen in the literature. For example, a time-step restriction depending on $\|\boldsymbol{u}\|_{\boldsymbol{\mathcal{W}}^{2,\infty}(\Omega)}$ is required in [SR20] for the error estimate of the solution resulting from the Crank-Nicolson scheme for the Navier-Stokes equations. ▲

Remark 5.15 (Error estimate for the fully discrete method applied to the Stokes problem). Looking at the proofs for the error estimates of the unfitted Eulerian time-stepping method in section 4.3 and the literature [LO19; BFM19; WRL21], we find that most of the work towards a full error estimate for the discretised scheme applied to the Stokes equations in this coupled setting, has been done by quantifying the domain consistency error and then carrying this through to the temporal estimate in Theorem 5.12.

For an error estimate for the fully discretised method, we would get an additional discretisation factor h^{2k} in the domain error in $\mathfrak{M}$ in Lemma 5.11. This would then lead to additional additive terms in the consistency estimate, which can be dealt with as above. This, together with standard interpolation estimates, then makes the derivation of the full error estimate a technical exercise in combining the elements used for the proof of Theorem 5.12 and [WRL21, Theorem 5.16]. ▲

CHAPTER 6

A Fluid-Structure Interaction Problem with Contact

Structure of Chapter The full fluid-rigid body system (2.5) is considered in the context of a fluid-structure interaction problem with solid contact, for which we have experimental data to validate our results against. The rotational symmetry of the set-up of this experiment in cylindrical coordinates allows the consideration of the underlying partial differential equation in a spatially reduced formulation. This allows the simulation of the three-dimensional problem using a two-dimensional discretisation. We present the known derivation of this reduced formulation and illustrate the accuracy and efficiency of this approach numerically. Using a simple contact avoidance scheme from the literature, we then simulate the full fluid-structure interaction scheme using our Eulerian time-stepping scheme developed in chapter 4. This follows our work in [Wah+21]. Details of the discretisation for the coupled problem, without prescribed geometry motion, are given, and the contact model's impact is studied numerically.

Contents of Chapter

6.1 Spatial Reduction of Rotationally Symmetric PDEs

Three-dimensional finite element discretisations are computationally more expensive than the corresponding two-dimensional discretisations due to the larger number of couplings between elements, resulting in denser systems. Under the assumptions of symmetry of the computational domain, data and the solution, the question arises whether a lower-dimensional computation suffices to solve the higher-dimensional problem and how to formulate such a reduced discretisation. In the following, we consider domains that are symmetric with respect to rotations in cylindrical coordinates. In order to compute flow problems posed in such geometries using a spatially reduced formulation, we derive the corresponding reduced weak formulations of the terms that constitute the weak formulation of the partial differential equations governing the fluid motion.

6.1.1 Derivation

Notation In the following, we shall use x, y, z to denote the three Cartesian coordinate directions. Images of functions and differential operators in Cartesian coordinates will be indicated by a " $\widehat{}$ ", i.e., $\widehat{u}$ or $\widehat{\nabla}$. Furthermore, we shall use lower case Roman indices to indicate an index from the Cartesian coordinate set. The three cylindrical coordinates (radius, angle and height) are denoted as r, ϕ, z while functions and differential operators are denoted without additional notation, i.e., simply u or ∇. Indices running over the cylindrical coordinate directions will be denoted by using lower case Greek letters.

The domain in which the PDE problem is posed is denoted as Ω in cylindrical coordinates. For the solution of a given PDE to be rotationally symmetric, we need to assume that the domain is also symmetric with respect to ϕ. To reduce the dimension of the problem, we will rotate the domain into the $r^+ - z$-plane. The two-dimensional domain is then denoted as Ω^{2d}. We further split the boundary of Ω^{2d} into the rotational axis part $\Gamma_{\text{rot}} = \{ \boldsymbol{x} \in \partial\Omega^{2d} \mid r = 0 \}$ and the remainder $\Gamma_{\text{out}} = \partial\Omega^{2d} \setminus \Gamma_{\text{rot}}$.

Coordinate Transformation In order to transform coordinates and derivatives, we consider the following mapping from cylinder to Cartesian coordinates:

$$T_t : (r, \phi, z) \mapsto (r\cos(\phi), r\sin(\phi), z) = (x, y, z).$$

The Jacobi matrix and its inverse of this mapping are given by

$$\boldsymbol{J} = \begin{pmatrix} \cos(\phi) & -r\sin(\phi) & 0 \\ \sin(\phi) & r\cos(\phi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{J}^{-1} = \begin{pmatrix} \cos(\phi) & \sin(\phi) & 0 \\ -\sin(\phi)/r & \cos(\phi)/r & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Assuming that the solution to a given partial differential equation is symmetric with respect to rotation, we can reduce the spatial description from a three-dimensional problem into a two-dimensional one by rotating the image of the solution into the $(x - z)$-plane. This transformation is defined by

$$T(\phi) = \begin{pmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{pmatrix} = (T_r, T_\phi, T_z).$$

For a velocity field $\widehat{\boldsymbol{u}} \in \mathbb{R}^3$, we then have

$$\widehat{\boldsymbol{u}}(x, y, z) = T(\phi)\boldsymbol{u}(r, \phi, z).$$

We note that it holds that $\partial_\phi T_r = T_\phi$ and $\partial_\phi T_\phi = -T_r$. In addition, we have that the columns of $T(\phi)$ are orthogonal, i.e., $T_\alpha \cdot T_\beta = \delta_{\alpha\beta}$ where $\delta_{\alpha\beta}$ denotes the Kronecker delta.

As we will need the inverse Jacobean often, we shall denote this as $F = \boldsymbol{J}^{-1}$. For this, we then have the identities

$$FF^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/r^2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad T^T F^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/r & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{6.1}$$

Transformation of Weak Forms

We transform the weak forms constituting the Navier-Stokes equations from the three-dimensional description in Cartesian coordinates to the rotationally symmetric and two-dimensional form.

Vector Laplace   We begin by transforming the image and the coordinates from Cartesian to cylindrical coordinates. For brevity and simplicity of notation, we denote the domain of integration in the r- and z-components in cylinder coordinates as R and Z, respectively:

$$(\widehat{\nabla}\widehat{\boldsymbol{u}}, \widehat{\nabla}\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = \sum_i \int_\Omega F^T\nabla(T\boldsymbol{u})_i \cdot F^T\nabla(T\boldsymbol{v}_i)\,\mathrm{d}\boldsymbol{x} = \sum_i \int_0^{2\pi}\int_{R,Z} FF^T\nabla(T\boldsymbol{u})_i \cdot \nabla(T\boldsymbol{v})_i r\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi$$

$$= \sum_i \int_0^{2\pi}\int_{R,Z} r\partial_r(T\boldsymbol{u})_i\partial_r(T\boldsymbol{v})_i + \frac{1}{r}\partial_\phi(T\boldsymbol{u})_i\partial_\phi(T\boldsymbol{v})_i + r\partial_z(T\boldsymbol{u})_i\partial_z(T\boldsymbol{v})_i\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi. \quad (6.2)$$

Now, since the basis vectors are dependent on ϕ, we do not have that $\partial_\phi(T\boldsymbol{u}) = 0$. Therefore, we need to transform the basis as well. For the first term in (6.2), we have that

$$\partial_r(T\boldsymbol{u})_i = \partial_r(\widehat{e}_i, T\boldsymbol{u}) = \partial_r(\widehat{e}_i, \sum_\alpha T_\alpha \boldsymbol{u}_\alpha) = (\widehat{e}_i, \sum_\alpha(\partial_r T_\alpha)\boldsymbol{u}_\alpha + T_\alpha(\partial_r\boldsymbol{u}_\alpha)) = (\widehat{e}_i, \sum_\alpha T_\alpha(\partial_r\boldsymbol{u}_\alpha)), \quad (6.3)$$

where the final equality followed from $\partial_r T_\alpha = 0$. For the remaining two terms, we find

$$\partial_z(T\boldsymbol{u})_i = (\widehat{e}_i, \sum_\alpha T_\alpha(\partial_z\boldsymbol{u}_\alpha)), \quad \partial_\phi(T\boldsymbol{u})_i = (\widehat{e}_i, \sum_\alpha(\partial_\phi T_\alpha)\boldsymbol{u}_\alpha + T_\alpha(\partial_\phi\boldsymbol{u}_\alpha)) = (\widehat{e}_i, T_\phi\boldsymbol{u}_r - T_r\boldsymbol{u}_\phi), \quad (6.4)$$

where the last equality follows from $\partial_\phi T_z = 0$ and $\partial_\phi\boldsymbol{u} = 0$. Now, using the orthogonality of the columns of T, we find using the above three representations of the derivatives of the transformed images that

$$\sum_i \partial_r(T\boldsymbol{u})_i\partial_r(T\boldsymbol{v})_i = (\sum_\alpha T_\alpha(\partial_r\boldsymbol{u}_\alpha), \sum_\beta T_\beta(\partial_r\boldsymbol{v}_\beta)) = \sum_{\alpha,\beta} T_\alpha \cdot T_\beta\partial_r\boldsymbol{u}_\alpha\partial_r\boldsymbol{v}_\beta = \sum_\alpha \partial_r\boldsymbol{u}_\alpha\partial_r\boldsymbol{v}_\alpha,$$

and similarly

$$\sum_i \partial_z(T\boldsymbol{u})_i\partial_z(T\boldsymbol{v})_i = \sum_\alpha \partial_z\boldsymbol{u}_\alpha\partial_z\boldsymbol{v}_\alpha$$

$$\sum_i \partial_\phi(T\boldsymbol{u})_i\partial_\phi(T\boldsymbol{v})_i = (T_\phi\boldsymbol{u}_r - T_r\boldsymbol{u}_\phi, T_\phi\boldsymbol{v}_r - T_r\boldsymbol{v}_\phi) = \boldsymbol{u}_r\boldsymbol{v}_r + \boldsymbol{u}_\phi\boldsymbol{v}_\phi.$$

The weak form of the vector Laplace (6.2) can therefore be written as

$$(\widehat{\nabla}\widehat{\boldsymbol{u}}, \widehat{\nabla}\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = 2\pi\int_{R,Z} r\Big[\sum_\alpha \partial_r\boldsymbol{u}_\alpha\partial_r\boldsymbol{v}_\alpha + \partial_z\boldsymbol{u}_\alpha\partial_z\boldsymbol{v}_\alpha\Big] + \frac{1}{r}(\boldsymbol{u}_r\boldsymbol{v}_r + \boldsymbol{u}_\phi\boldsymbol{v}_\phi)\,\mathrm{d}r\,\mathrm{d}z.$$

Unfortunately, the expression on the right-hand side of this is still three-dimensional. As a result, we need an additional equation for $\boldsymbol{u}_\phi$. In order to rotate $\widehat{\boldsymbol{u}}$ into the $(x-z)$-plane, we have assumed that $\boldsymbol{u}_\phi$ is constant. The only physical choice in the context of $\boldsymbol{u}$ being a fluid velocity is $\boldsymbol{u}_\phi = 0$. Any non-zero value would correspond to a rotation of the frame of reference. This would then cause Coriolis and centrifugal forces after the transformation back to Cartesian coordinates. However, there is no reason for these to appear. We can therefore write the vector Laplace equation into the two-dimensional expression

$$(\widehat{\nabla}\widehat{\boldsymbol{u}}, \widehat{\nabla}\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = 2\pi\Big[(r\nabla\boldsymbol{u}^{2d}, \nabla\boldsymbol{v}^{2d})_{\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r^{2d}, \boldsymbol{v}_r^{2d})_{\Omega^{2d}}\Big]. \quad (6.5)$$

Finally, we need a boundary condition for $\boldsymbol{u}_r$ on the axis of rotational symmetry at $r = 0$ since the original problem does not give this. Since we have that $\boldsymbol{u}$ is a flow velocity, and therefore divergence-free, it holds

$$0 = \nabla\cdot\boldsymbol{u} = \frac{1}{r}\partial_r(r\boldsymbol{u}_r) + \frac{1}{r}\partial_\phi\boldsymbol{u}_\phi + \partial_z\boldsymbol{u}_z = \frac{1}{r}\boldsymbol{u}_r + \partial_r\boldsymbol{u}_r + \partial_z\boldsymbol{u}_z$$

for a rotationally symmetric $\boldsymbol{u}$. For $r \to 0$ it must therefore hold that $\boldsymbol{u}_r = 0$ on the symmetry axis.

Remark 6.1 (Quadrature).   The second term on the right-hand side of (6.5) has a singularity on the rotational axis $r = 0$. As a result, a finite element discretisation for this equation must be based on quadrature rules that do not contain any points on the boundary. As noted in subsection 4.5.2, NGSolve uses mapped Gauß-Jacobi quadrature rules on simplices for higher-order ($k > 1$) elements such that shape functions do not have to be evaluated on the element boundary.   ▲

Transposed Gradient   For fluid-structure interaction problems, we require the transformed transposed gradient for the Cauchy stress tensor. We look at the components of the two matrices in $((\widehat{\nabla}\widehat{\boldsymbol{u}})^T, \widehat{\nabla}\widehat{\boldsymbol{v}})_{\widehat{\Omega}}$. For the test function, we have with the product rule that

$$(\widehat{\nabla}\widehat{\boldsymbol{v}})_{ij} = (F^T\nabla(T\boldsymbol{v}))_{ij} = \sum_{\alpha,\beta} f^t_{i\alpha}\partial_\alpha(t_{j\alpha}\boldsymbol{v}_\alpha) = \sum_{\alpha,\beta} f_{i\alpha}(\partial_\alpha\boldsymbol{v}_\beta)t^t_{\beta j} + f^t_{i\phi}t_{j\phi}\boldsymbol{v}_r = (F^T(\nabla\boldsymbol{v})T^T)_{ij} + f^t_{i\phi}t_{j\phi}\boldsymbol{v}_r,$$

where the penultimate equality follows from $\boldsymbol{v}_\phi = 0$ and $\partial_\phi T_r = T_\phi$. Similarly, we have for the trial function

$$(\widehat{\nabla}\widehat{\boldsymbol{u}})^T_{ij} = (T(\nabla\boldsymbol{u})^T F)_{ij} + t_{i\phi}f_{\phi j}\boldsymbol{u}_r.$$

Writing this out in full and using that $\partial_\phi\boldsymbol{v} = 0$ then gives

$$\widehat{\nabla}\widehat{\boldsymbol{v}} = \begin{pmatrix} \cos^2(\phi)\partial_r\boldsymbol{v}_r + \sin^2(\phi)\boldsymbol{v}_r/r & \cos(\phi)\sin(\phi)\partial_r\boldsymbol{v}_r - \cos(\phi)\sin(\phi)\boldsymbol{v}_r/r & \cos(\phi)\partial_r\boldsymbol{v}_z \\ \cos(\phi)\sin(\phi)\partial_r\boldsymbol{v}_r - \cos(\phi)\sin(\phi)\boldsymbol{v}_r/r & \sin^2(\phi)\partial_r\boldsymbol{v}_r + \cos^2(\phi)\boldsymbol{v}_r/r & \sin(\phi)\partial_r\boldsymbol{v}_z \\ \cos(\phi)\partial_r\boldsymbol{v}_z & \sin(\phi)\partial_r\boldsymbol{v}_z & \partial_z\boldsymbol{v}_z \end{pmatrix}.$$

This then leads to

$$(\widehat{\nabla}\widehat{\boldsymbol{u}})^T : \widehat{\nabla}\widehat{\boldsymbol{v}} = \partial_r\boldsymbol{u}_r\partial_r\boldsymbol{v}_r + \partial_r\boldsymbol{u}_z\partial_z\boldsymbol{v}_r + \partial_z\boldsymbol{u}_r\partial_r\boldsymbol{v}_z + \partial_z\boldsymbol{u}_z\partial_z\boldsymbol{v}_z + \frac{1}{r^2}\boldsymbol{u}_r\boldsymbol{v}_r$$

$$= (\nabla^{2d}\boldsymbol{u}^{2d})^T : \nabla^{2d}\boldsymbol{v}^{2d} + \frac{1}{r^2}\boldsymbol{u}_r\boldsymbol{v}_r.$$

Integrating with respect to $\widehat{\Omega}$, using the above expression, changing the integration to cylinder coordinated and integrating with respect to ϕ then gives the two-dimensional weak expression

$$((\widehat{\nabla}\widehat{\boldsymbol{u}})^T, \widehat{\nabla}\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = 2\pi\left[(r(\nabla\boldsymbol{u}^{2d})^T, \nabla\boldsymbol{v}^{2d})_{\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r^{2d}, \boldsymbol{v}_r^{2d})_{\Omega^{2d}}\right]. \tag{6.6}$$

Pressure Coupling   We proceed as before by transforming the image and coordinates into cylinder coordinates. In the following, lower case letters will denote the matrix entries for the coordinate and image transformation matrices. We compute using the product rule that

$$(\widehat{p}, \widehat{\nabla}\cdot\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = \sum_i \int_{\widehat{\Omega}} \widehat{p}\,\widehat{\partial}_i(T\boldsymbol{v})_i\,\mathrm{d}\widehat{\boldsymbol{x}} = \sum_{i,\alpha} \int_0^{2\pi}\int_{R,Z} rpf^T_{i\alpha}\partial_\alpha(T\boldsymbol{v})_i\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi$$

$$= \sum_{i,\alpha,\beta} \int_0^{2\pi}\int_{R,Z} rpf^T_{i\alpha}\partial_\alpha(t_{i,\beta}\boldsymbol{v}_\beta)\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi = \sum_{i,\alpha,\beta} \int_0^{2\pi}\int_{R,Z} rpf^T_{i\alpha}((\partial_\alpha t_{i,\beta})\boldsymbol{v}_\beta + t_{i\beta}\partial_\alpha\boldsymbol{v}_\beta)\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi$$

$$= \sum_{i,\alpha,\beta} \int_0^{2\pi}\int_{R,Z} rpf_{\phi i}(\partial_\phi t_{i\beta})\boldsymbol{v}_\beta + rp(t^T_{\beta i}f^T_{ir}\partial_r\boldsymbol{v}_\beta + t^T_{\beta i}f^T_{iz}\partial_z\boldsymbol{v}_\beta)\,\mathrm{d}r\,\mathrm{d}z\,\mathrm{d}\phi, \tag{6.7}$$

where the last equality follows from T only depending on ϕ and $\boldsymbol{v}$ being independent of ϕ. For the first part of (6.7), we have

$$\sum_{i,\beta} f_{\phi i}(\partial_\phi t_{i\beta})\boldsymbol{v}_\beta = \begin{pmatrix} \frac{-\sin(\phi)}{r} & \frac{\cos(\phi)}{r} & 0 \end{pmatrix}\begin{pmatrix} -\sin(\phi) & -\cos(\phi) & 0 \\ \cos(\phi) & -\sin(\phi) & 0 \\ 0 & 0 & 0 \end{pmatrix}\boldsymbol{v} = \frac{\boldsymbol{v}_r}{r}.$$

For the second part of (6.7), we use (6.1) such that

$$\sum_{i,\beta} t^T_{\beta i}f^T_{ir}\partial_r\boldsymbol{v}_\beta + t^T_{\beta i}f^T_{iz}\partial_z\boldsymbol{v}_\beta = \partial_r\boldsymbol{v}_r + \partial_z\boldsymbol{v}_z.$$

In total, we therefore have

$$(\widehat{p}, \widehat{\nabla}\cdot\widehat{\boldsymbol{v}})_{\widehat{\Omega}} = 2\pi(p, \boldsymbol{v}_r^{2d} + r\nabla\cdot\boldsymbol{v}^{2d})_{\Omega^{2d}}.$$

Time-Derivative   As this does not contain spatial derivatives, we only need to transform the image. Using the orthogonality of the transformation matrix T and the fact that we assume $\boldsymbol{u}_\phi = 0$, we get

$$(\partial_t \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{v}})_{\widehat{\Omega}} = \sum_i \int_0^{2\pi} \int_{R,Z} \partial_t (T\boldsymbol{u})_i (T\boldsymbol{v})_i r \, \mathrm{d}r \, \mathrm{d}z \, \mathrm{d}\phi$$

$$= \sum_{\alpha,\beta} \int_0^{2\pi} \int_{R,Z} T_\alpha \cdot T_\beta \partial_t \boldsymbol{u}_\alpha \boldsymbol{v}_\beta r \, \mathrm{d}r \, \mathrm{d}z \, \mathrm{d}\phi = 2\pi (r \partial_t \boldsymbol{u}^{2d}, \boldsymbol{v}^{2d})_{\Omega^{2d}}. \tag{6.8}$$

Transport   For the convention term, we compute using (6.1)

$$(\widehat{\boldsymbol{u}} \cdot \widehat{\nabla} \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{w}})_{\widehat{\Omega}} = \sum_i \int_0^{2\pi} \int_{R,Z} \left(\boldsymbol{u} \cdot (T^T F^T \nabla(T\boldsymbol{v})) \right)_i (T\boldsymbol{w})_i r \, \mathrm{d}r \, \mathrm{d}z \, \mathrm{d}\phi$$

$$= \sum_i \int_0^{2\pi} \int_{R,Z} \left(r \boldsymbol{u}_r \partial_r (T\boldsymbol{v}) + \boldsymbol{u}_\phi \partial_\phi (T\boldsymbol{v}) + r \boldsymbol{u}_z \partial_z (T\boldsymbol{v}) \right)_i (T\boldsymbol{w})_i \, \mathrm{d}r \, \mathrm{d}z \, \mathrm{d}\phi.$$

The middle term vanishes by assumption. For the first term, we make use of (6.3) such that

$$\sum_i r \boldsymbol{u}_r \partial_r (T\boldsymbol{v})_i (T\boldsymbol{w})_i = \sum_{\alpha,\beta} T_\alpha \cdot T_\beta r \boldsymbol{u}_r (\partial_r \boldsymbol{v}_\alpha) \boldsymbol{w}_\beta = \sum_\alpha r \boldsymbol{u}_r (\partial_r \boldsymbol{v}_\alpha) \boldsymbol{w}_\alpha.$$

We get the analogous result for the third term by using (6.4). In sum, this gives

$$(\widehat{\boldsymbol{u}} \cdot \widehat{\nabla} \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{w}})_{\widehat{\Omega}} = 2\pi (r \boldsymbol{u}^{2d} \cdot \nabla \boldsymbol{v}^{2d}, \boldsymbol{w}^{2d})_{\Omega^{2d}}. \tag{6.9}$$

Forcing   To transform the forcing term $\widehat{\boldsymbol{f}}$, we simply need to transform the image. Using the fact that the columns of T are orthogonal, we have

$$(\widehat{\boldsymbol{f}}, \widehat{\boldsymbol{v}})_{\widehat{\Omega}} = \int_0^{2\pi} \int_{R,Z} T^T T \boldsymbol{f} \cdot \boldsymbol{v} r \, \mathrm{d}r \, \mathrm{d}z \, \mathrm{d}\phi = 2\pi \int_{R,Z} \boldsymbol{f} \cdot \boldsymbol{v} r \, \mathrm{d}r \, \mathrm{d}z.$$

Since we have assumed that $\boldsymbol{u}_\phi = 0$, we also have $\boldsymbol{f}_\phi = 0$. The above expression is therefore a two-dimensional one.

Boundary Forces   In order to compute forces acting on an object within our flow domain, we also have to transform the stress $\boldsymbol{\sigma}(\boldsymbol{u}, p)$ into two dimensions.

Transforming the image and gradient into cylindrical coordinates and using the orthogonality of the transformation T gives

$$((\nu \widehat{\nabla} \widehat{\boldsymbol{u}} - I\widehat{p})\widehat{\boldsymbol{n}}, \widehat{\boldsymbol{v}})_{\widehat{\Gamma}} = (T^T F^T \nabla(T\boldsymbol{u}) T\boldsymbol{n} - T^T T p\boldsymbol{n}, \boldsymbol{v})_\Gamma = (T^T F^T \nabla(T\boldsymbol{u}) T\boldsymbol{n} - Ip\boldsymbol{n}, \boldsymbol{v})_\Gamma.$$

Using (6.3) and (6.4) together with the orthogonality of the columns of T then gives

$$\sum_i \partial_r (T\boldsymbol{u})_i \cdot (T\boldsymbol{n})_i = \sum_\alpha \partial_r \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha \qquad \text{and} \qquad \sum_i \partial_z (T\boldsymbol{u})_i \cdot (T\boldsymbol{n})_i = \sum_\alpha \partial_z \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha,$$

while (6.4) and $\boldsymbol{u}_\phi = 0$ give

$$\sum_i \partial_\phi (T\boldsymbol{u})_i \cdot (T\boldsymbol{n})_i = \sum_\beta T_\phi \cdot T_\beta \boldsymbol{u}_r \boldsymbol{n}_\beta - T_r \cdot T_\beta \boldsymbol{u}_\phi \boldsymbol{n}_\beta = \boldsymbol{u}_r \boldsymbol{n}_\phi.$$

Combining these results with (6.1), we have

$$T^T F^T \nabla(T\boldsymbol{u})T\boldsymbol{n} \cdot \boldsymbol{v} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/r & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \sum_\alpha \partial_r \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha \\ \boldsymbol{u}_r \boldsymbol{n}_\phi \\ \sum_\alpha \partial_z \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_r \\ \boldsymbol{v}_\phi \\ \boldsymbol{v}_z \end{pmatrix} = \sum_\alpha \left[\partial_r \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha \boldsymbol{v}_r + \partial_z \boldsymbol{u}_\alpha \boldsymbol{n}_\alpha \boldsymbol{v}_z \right] + \frac{1}{r} \boldsymbol{u}_r \boldsymbol{n}_\phi.$$

Integrating with respect to ϕ and using the fact that $\int_0^{2\pi} 1/r \boldsymbol{u}_r \boldsymbol{n}_\phi \, \mathrm{d}\phi = 0$ then gives

$$((\widehat{\nabla}\widehat{\boldsymbol{u}} - I\widehat{p})\widehat{\boldsymbol{n}}, \widehat{\boldsymbol{v}})_{\widehat{\Gamma}} = 2\pi (r(\nu\nabla^{2d}\boldsymbol{u}^{2d} - Ip)\boldsymbol{n}^{2d}, \boldsymbol{v}^{2d})_{\Gamma^{2d}}.$$

Note: The surface element in cylindrical coordinates is $\mathrm{d}s = r\boldsymbol{e}_r \, \mathrm{d}\phi \, \mathrm{d}z + \boldsymbol{e}_\phi \, \mathrm{d}r \, \mathrm{d}z + r\boldsymbol{e}_z \, \mathrm{d}r \, \mathrm{d}\phi.$

CutFEM Nitsche Discretisation

In order to use the rotationally symmetric formulation to compute three-dimensional problems with a two-dimensional discretisation, we need to derive the appropriate Nitsche formulation to implement boundary conditions on an unfitted interface and appropriate ghost-penalty operators.

Enforcing Dirichlet boundary conditions   To derive the rotationally symmetric Nitsche formulation, we follow the heuristic derivation of the symmetric interior penalty method in [PE12]. Integration by parts of (6.5) gives

$$(r\nabla^{2d}\boldsymbol{u}^{2d}, \nabla^{2d}\boldsymbol{v}^{2d})_{\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r^{2d}, \boldsymbol{v}_r^{2d})_{\Omega^{2d}}$$

$$= -(\nabla^{2d} \cdot (r\nabla^{2d}\boldsymbol{u}^{2d}), \nabla^{2d}\boldsymbol{v}^{2d})_{\Omega^{2d}} + (r(\nabla^{2d}\boldsymbol{u}^{2d})\boldsymbol{n}, \boldsymbol{v}^{2d})_{\partial\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r^{2d}, \boldsymbol{v}_r^{2d})_{\Omega^{2d}}$$

$$= -(\partial_r \boldsymbol{u}^{2d} + r\Delta^{2d}\boldsymbol{u}^{2d}, \boldsymbol{v}^{2d})_{\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r^{2d}, \boldsymbol{v}_r^{2d})_{\Omega^{2d}} + (r(\nabla^{2d}\boldsymbol{u}^{2d})\boldsymbol{n}, \boldsymbol{v}^{2d})_{\partial\Omega^{2d}}.$$

The two volume terms correspond exactly to $\frac{1}{2\pi}(-\Delta\boldsymbol{u}, \boldsymbol{v})_{\Omega^{3d}}$ in cylindrical coordinates, under our assumption that $\partial_\phi \boldsymbol{u}_\alpha = \boldsymbol{u}_\varphi = 0$ and taking the volume measure in cylindrical coordinates into consideration. The boundary term $-(r(\nabla^{2d}\boldsymbol{u}^{2d})\boldsymbol{n}, \boldsymbol{v}^{2d})_{\partial\Omega^{2d}}$ is therefore needed in our weak formulation for consistency. To make the formulation symmetric, we add the term $-(\boldsymbol{u}^{2d}, r(\nabla^{2d}\boldsymbol{v}^{2d})\boldsymbol{n})_{\partial\Omega^{2d}}$, which is consistent due to the homogeneous Dirichlet boundary condition for $\boldsymbol{u}^{2d}$. To enforce the boundary condition, we add the consistent penalty term $\frac{\sigma}{h}(r\boldsymbol{u}^{2d}, \boldsymbol{v}^{2d})_{\partial\Omega^{2d}}$ with the penalty parameter $\sigma > 0$ and mesh size h. This term then also gives coercivity of the combined bilinear form, under the assumption that σ is sufficiently large in the appropriate natural norm. In total, the consistent and symmetric Nitsche formulation for the vector valued diffusion operator reads

$$a^{2d}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq (r\nabla^{2d}\boldsymbol{u}, \nabla^{2d}\boldsymbol{v})_{\Omega^{2d}} + (\frac{1}{r}\boldsymbol{u}_r, \boldsymbol{v}_r)_{\Omega^{2d}}$$

$$- (r(\nabla^{2d}\boldsymbol{u})\boldsymbol{n}, \boldsymbol{v})_{\Gamma^{2d}} - (r\boldsymbol{u}, (\nabla^{2d}\boldsymbol{v})\boldsymbol{n})_{\Gamma^{2d}} + \sigma\frac{k^2}{h}(r\boldsymbol{u}, \boldsymbol{v})_{\Gamma^{2d}}, \qquad (6.10)$$

where Γ^{2d} is the part of the two-dimensional boundary on which we enforce Dirichlet values. Similarly integration by parts gives a consistent two-dimensional formulation for the pressure coupling as

$$b^{2d}(\boldsymbol{v}, q) \coloneqq -(q, \boldsymbol{v}_r + r\operatorname{div}(\boldsymbol{v}))_{\Omega^{2d}} + (rq, \boldsymbol{v}\boldsymbol{n})_{\Gamma^{2d}}. \qquad (6.11)$$

Finally, if we consider inhomogeneous boundary data $\boldsymbol{u}_D$, then the right-hand side is adapted accordingly with the symmetry and penalty terms for the velocity and pressure

$$f^{2d}(\boldsymbol{v}) \coloneqq (r\boldsymbol{f}, \boldsymbol{v})_{\Omega^{2d}} - (r\boldsymbol{u}_D, (\nabla^{2d}\boldsymbol{v})\boldsymbol{n})_{\Gamma^{2d}} + \sigma\frac{k^2}{h}(r\boldsymbol{u}_D, \boldsymbol{v})_{\Gamma^{2d}} + (rq, \boldsymbol{u}_D\boldsymbol{n})_{\Gamma^{2d}}. \qquad (6.12)$$

Ghost-Penalty Stabilisation   As we have seen in chapter 3, ghost-penalty stabilisation is intended to provide stability estimates for CutFEM discretisations in natural norms, defined on the entire active mesh. To this end, let $\widetilde{\Omega}^{2d}$ be the background domain to the rotationally reduced domain Ω^{2d}. In the spirit of our previous notation $\mathscr{T}_h^{2d}$ then denotes a shape-regular and quasi-uniform mesh of $\widetilde{\Omega}^{2d}$ and $\mathcal{O}_{\mathscr{T}}^{2d}$ denotes the domain of the active mesh. In the unfitted, rotationally reduced setting, we then consider the following natural norms for the rotationally reduced Stokes problem

$$\||\boldsymbol{v}\||_\sharp^2 := \|r^{1/2}\nabla\boldsymbol{v}\|_{\mathcal{O}_{\mathscr{T}}^{2d}}^2 + \|r^{-1/2}\boldsymbol{v}_r\|_{\mathcal{O}_{\mathscr{T}}^{2d}}^2 + \|h^{-1/2}r^{1/2}\boldsymbol{v}\|_{\Gamma^{2d}}^2 \qquad \text{and} \qquad \||q\||_\sharp^2 := \|r^{1/2}q\|_{\mathcal{O}_{\mathscr{T}}^{2d}}^2,$$

c.f. subsection 3.2.2.

Looking at the reduced formulation of the diffusion operator (6.5), it is clear that the ghost-penalty operator must stabilise $r^{1/2}$-scaled gradients and the $r^{1/2}$-scaled r-component in the $\mathcal{L}^2$-sense. The direct version of the velocity ghost-penalty operator in the rotationally reduced formulation is therefore given by

$$i_h^{2d}(\boldsymbol{u},\boldsymbol{v}) := \gamma_{gp,\boldsymbol{u}} \sum_{F\in\mathscr{F}_h} \int_{\omega_F} \frac{r}{h^2}[\![\boldsymbol{u}]\!]\cdot[\![\boldsymbol{v}]\!] + \frac{1}{r}[\![\boldsymbol{u}_r]\!][\![\boldsymbol{v}_r]\!]\,\mathrm{d}\boldsymbol{x}. \tag{6.13}$$

Similarly for the pressure, we have

$$j_h^{2d}(p,q) := \gamma_{gp,p} \sum_{F\in\mathscr{F}_h} \int_{\omega_F} r[\![p]\!][\![q]\!]\,\mathrm{d}\boldsymbol{x}. \tag{6.14}$$

With the all the terms needed transformed into the rotationally reduced formulation, we can now formulate the reduced problems.

Problem P9.   The rotationally reduced CutFEM discretisation of the Stokes problem reads as follows. Find $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_h \times Q_h$ such that

$$\nu a_h^{2d}(\boldsymbol{u}_h,\boldsymbol{v}_h) + \nu i_h^{2d}(\boldsymbol{u}_h,\boldsymbol{v}_h) + b_h^{2d}(\boldsymbol{v}_h,p_h) + b_h^{2d}(\boldsymbol{u}_h,q_h) - \frac{1}{\nu}j_h^{2d}(p_h,q_h) = f_h^{2d}(\boldsymbol{v}_h,q_h)$$

for all $(\boldsymbol{v}_h,q_h) \in \boldsymbol{V}_h \times Q_h$, with the rotationally reduced diffusion, pressure coupling, forcing and stabilising forms given in (6.10), (6.11), (6.12), (6.13) and (6.14), respectively, and the boundary conditions $\boldsymbol{u}_r = 0$ and $\partial_r\boldsymbol{u}_z = 0$ on the symmetry axis $r = 0$.

Due to the scaling of the ghost-penalty operators, it is a repetition of the arguments presented in chapter 3 to show stability and unique solvability of Problem P9 using the natural norms $\||\cdot\||_\sharp$. Error estimates then follow immediately in these norms since they only include an additional scaling in r, which is independent of the mesh size and the polynomial order. Furthermore, we note that by the construction of the weak forms, it is clear that these natural norms are equivalent to the three-dimensional standard norms in the case of a rotationally symmetric flow solution.

The rotationally reduced CutFEM discretisation of the Navier-Stokes problem using the BDF1 time-discretisation reads as follows:

Problem P10.   With compatible initial data $\boldsymbol{u}_h^0$, for $n = 1, \ldots, N$, find $(\boldsymbol{u}_h^n, p_h^n) \in \boldsymbol{V}_h \times Q_h$ such that

$$\frac{1}{\Delta t}m_h^{2d}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1},\boldsymbol{v}_h) + \nu a_h^{2d}(\boldsymbol{u}_h^n,\boldsymbol{v}_h) + \nu i_h^{2d}(\boldsymbol{u}_h^n,\boldsymbol{v}_h) + c_h(\boldsymbol{u}_h^n,\boldsymbol{u}_h^n,\boldsymbol{v}_h)$$

$$+ b_h^{2d}(\boldsymbol{v}_h,p_h) + b_h^{2d}(\boldsymbol{u}_h,q_h) - \frac{1}{\nu}j_h^{2d}(p_h,q_h) = f_h^{2d}(\boldsymbol{v}_h,q_h)$$

for all $(\boldsymbol{v}_h,q_h) \in \boldsymbol{V}_h \times Q_h$, together with the boundary conditions $\boldsymbol{u}_r = 0$ and $\partial_r\boldsymbol{u}_z = 0$ on the symmetry axis $r = 0$ and the rotationally reduced diffusion, pressure coupling, forcing and stabilising forms given in (6.10), (6.11), (6.12), (6.13) and (6.14), respectively, as well as the mass term

$$m_h^{2d}(\boldsymbol{u}_h,\boldsymbol{v}_h) = (r\boldsymbol{u}_h,\boldsymbol{v}_h)_{\Omega_h}$$

and transport term

$$c_h^{2d}(\boldsymbol{u}_h,\boldsymbol{v}_h,\boldsymbol{w}_h) = (r\boldsymbol{u}_h\cdot\nabla\boldsymbol{v}_h,\boldsymbol{w}_h)_{\Omega_h}.$$
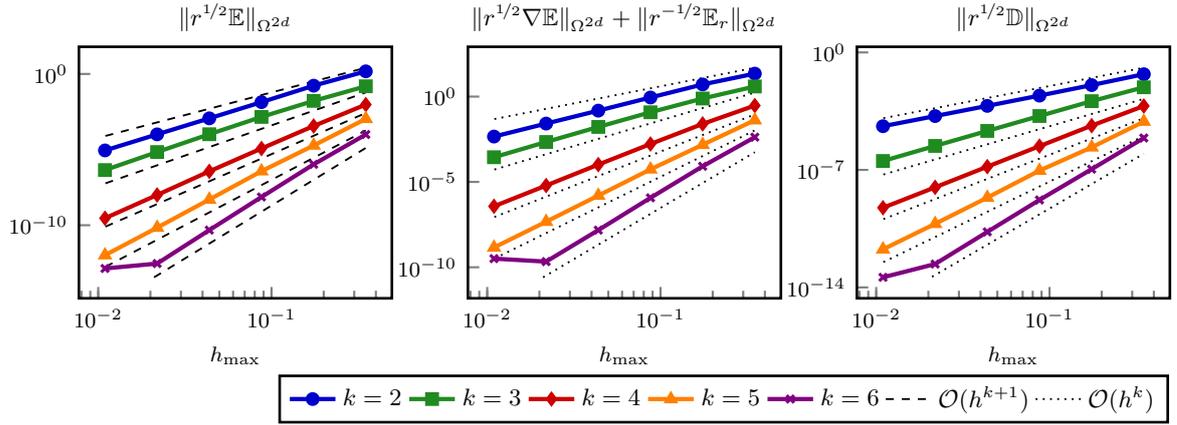
Figure 6.1: Mesh convergence for the rotationally reduced Stokes problem using unfitted Taylor-Hood elements in two-dimensional norms equivalent to the three-dimensional norms.

6.1.2 Numerical Examples

We illustrate the accuracy and efficiency of the rotationally reduced formulation with two numerical examples.

Example 1: Stokes

To illustrate the convergence behaviour of the rotationally reduced formulation, we consider a CutFEM discretisation of a simple cylinder with forcing data such that the exact solution is known. Consider the cylinder $Z = \{(r\sin(\phi), r\cos(\phi), z) \in \mathbb{R}^3 \mid r \in [0,1), \phi \in [0, 2\pi), z \in (0,1)\}$. The rotationally reduced domain is then given by $\Omega^{2d} = (0,1)^2$. In Z, consider the rotationally symmetric, divergence-free velocity field and the pressure with mean zero

$$\boldsymbol{u} = \begin{pmatrix} \pi r^2 (1 - r^2) \cos(\pi z) \\ 0 \\ (5r^3 - 3r) \sin(\pi z) \end{pmatrix} \qquad \text{and} \qquad p = r^2(1 - r^2)\sin(\pi z) - \frac{4}{15\pi}.$$

We then take the forcing term as $\boldsymbol{f} = -\nu\Delta\boldsymbol{u} + \nabla p$ and choose the viscosity as $\nu = 10^{-2}$.

For simplicity, we take the background domain as $\widetilde{\Omega} = (0,1)\times(0,1.1)$ and the level set function as $\phi = y - 1$. The ghost-penalty parameters are chosen as $\gamma_{gp,\boldsymbol{u}} = \gamma_{gp,p} = 0.01$ and the Nitsche penalty parameter as $\sigma = 40k^2$. We compute the problem on a series of uniformly refined meshes using Taylor-Hood elements of order $2 \leq k \leq 6$ and measure the error in the scaled two-dimensional norms, corresponding to the three-dimensional norm under the assumption of rotational symmetry.

The resulting errors can be seen in Figure 6.1. We can clearly see optimal-order convergence for the velocity and pressure error in the considered norms until machine precision is reached, thereby validating our transformed unfitted formulation.

Example 2: Navier-Stokes

To illustrate the validity of the transformation for a more involved problem, we consider the stationary Navier-Stokes flow in a cylinder around a spherical obstacle. To this end, let us consider the cylinder $Z = \{(r\sin(\phi), r\cos(\phi), z) \in \mathbb{R}^3 \mid r \in [0,1), \phi \in [0,2\pi), z \in (0,3)\}$ and the sphere $S = \{\boldsymbol{x} \in \mathbb{R}^3 \mid \boldsymbol{x}_1^2 + \boldsymbol{x}_2^2 + (\boldsymbol{x}_3 - 1.5)^2 < 0.5^2\}$. The fluid domain is then given by $\Omega^{3d} = Z \setminus \overline{S}$. The rotationally reduced domain is then given as $\Omega^{2d} = (0,1) \times (0,3) \setminus \{\boldsymbol{x} \in \mathbb{R}^2_+ \mid \boldsymbol{x}_1^2 + (\boldsymbol{x}_2 - 1.5)^2 \leq 0.5^2\}$. A sketch of these two domains can be seen in Figure 6.2.
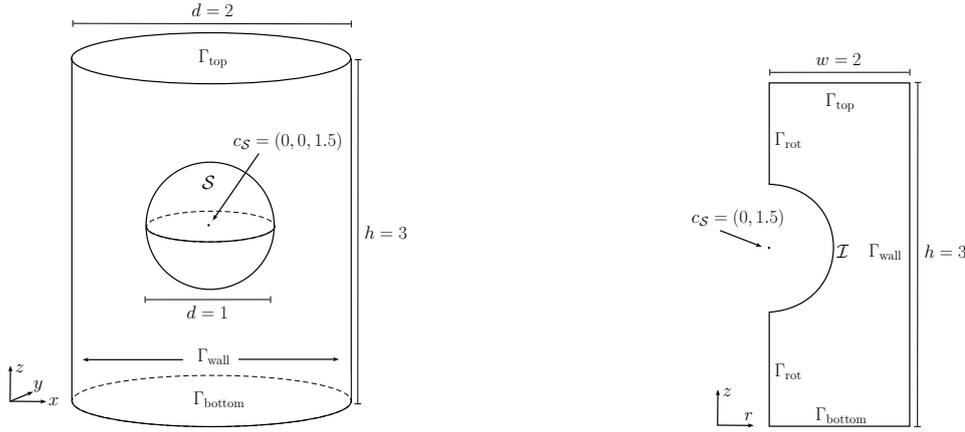
Figure 6.2: Left: Three-dimensional domain for the flow around a sphere in a cylinder. Right: Rotationally reduced two-dimensional domain.

At the top boundary of the cylinder Γ_{top} at $z = 3$, we impose a parabolic inflow profile given as $\boldsymbol{u}_{\text{in}} = (0, 0, u_{\text{max}}(r^2 - 1))^T$. Taking the maximal inflow speed as $u_{\text{max}} = 0.2$, we get the mean inflow speed $u_m = 0.2^\pi/2 \approx 0.31$. With the viscosity $\nu = 10^{-3}$, the fluid density $\rho_{\mathcal{F}} = 1$ and the diameter of the obstacle as the reference length $L = 1$, the resulting flow has the Reynolds number $Re = \frac{u_m L}{\nu} \approx 31$. This is sufficiently small for the flow to be stationary. At the outer walls of the cylinder and on the surface of the obstacle $\Gamma_{\text{wall}} \cup \mathcal{I}$, we impose a no-slip condition $\boldsymbol{u} = 0$ and at the bottom boundary of the cylinder Γ_{bottom} at $z = 0$, we set the "do-nothing" condition.

As the quantity of interest, we consider the dimensionless drag coefficient $C_{\boldsymbol{F}_3} = \frac{2\boldsymbol{F}_3}{u_m^2 \rho_{\mathcal{F}} L}$ with $\boldsymbol{F} = \int_{\partial S} \widehat{\boldsymbol{\sigma}}(\widehat{\boldsymbol{u}}, \widehat{p}) \, d\widehat{s}$.

Reference: Fitted Three-Dimensional Computation To establish a reference for our computations, we compute the problem in full three dimensions using a high-order, point-wise divergence-free, hybrid discontinuous Galerkin (HDG) method based on [LS16]. This uses $\boldsymbol{\mathcal{H}}(\text{div})$-conforming finite elements of order k for the velocity, discontinuous pressure elements of order $k - 1$ and is inf-sup stable. The code is adapted from [Wah+19a].

We use a tetrahedral mesh of the three-dimensional domain with local meshing parameter $h = h_{\text{max}}/6$ on the sphere. The forces on the sphere are evaluated using the boundary-integral formulation. To evaluate the accuracy of the force evaluation, we also look at the first two components, which should be zero by the symmetry of the set-up.

The computational results can be seen in Table 6.1, and a visualisation of the velocity can be seen on the left of Figure 6.3. Looking at the two horizontal forces, we see that they indeed converge towards zero for h and p refinement. In fact, we see that going for higher-order elements leads to in significantly more efficient results compared to mesh refinement. From these results, we conclude that we can have $-C_{\boldsymbol{F}_3} = 4.38983$ as a reference for the drag coefficient.

Rotationally Symmetric CutFEM Computation For the rotationally reduced CutFEM computations, we take a uniform mesh of the background domain and consider the isoparametric Taylor-Hood discretisation from chapter 3. In line with our previous numerical experiments, we take the ghost-penalty parameter as $\gamma_{gp,\boldsymbol{u}} = \gamma_{gp,p} = 0.01$ and the Nitsche penalty parameter as $\sigma = 40k^2$. The results are presented in Table 6.2, and a sample velocity solution is shown on the right of Figure 6.3.

Here we see again that higher-order elements appear to be very beneficial with respect to accuracy

Discretisation				Results			
$h_{\max}$	k	$\texttt{dof(gdof)} \times 10^3$	$\texttt{nze} \times 10^6$	$C_{\boldsymbol{F}_1}$	$C_{\boldsymbol{F}_2}$	$-C_{\boldsymbol{F}_3}$	Time(s)
0.5	1	961.6(113.0)	6.467	$-1.44 \cdot 10^{-2}$	$-2.68 \cdot 10^{-3}$	4.47336	9
0.5	2	1931.2(271.3)	25.866	$-5.54 \cdot 10^{-3}$	$-2.34 \cdot 10^{-3}$	4.43372	32
0.5	3	3221.0(527.6)	71.851	$-9.64 \cdot 10^{-4}$	$-3.81 \cdot 10^{-4}$	4.39744	94
0.5	4	4831.0(904.5)	161.664	$-9.35 \cdot 10^{-5}$	$-2.62 \cdot 10^{-6}$	4.39802	246
0.5	5	6761.2(1424.8)	316.861	$-1.01 \cdot 10^{-5}$	$-5.09 \cdot 10^{-6}$	4.39829	718
0.5	6	9011.7(2110.9)	563.309	$-1.98 \cdot 10^{-7}$	$1.85 \cdot 10^{-7}$	4.39833	1600
0.25	1	5210.7(593.3)	34.315	$-1.34 \cdot 10^{-3}$	$-6.05 \cdot 10^{-3}$	4.38948	138
0.25	2	10 411.5(1427.3)	137.258	$4.18 \cdot 10^{-4}$	$-1.58 \cdot 10^{-4}$	4.40718	454
0.25	3	17 315.8(2780.0)	381.272	$5.23 \cdot 10^{-5}$	$-3.04 \cdot 10^{-5}$	4.39804	2459
0.25	4	26 013.7(4771.8)	857.863	$1.96 \cdot 10^{-6}$	$-2.54 \cdot 10^{-6}$	4.39830	8581
0.5	2	1913.2(271.3)	25.866	$-5.54 \cdot 10^{-3}$	$-2.34 \cdot 10^{-3}$	4.43372	32
0.25	2	10 411.5(1427.3)	137.258	$4.18 \cdot 10^{-4}$	$-1.58 \cdot 10^{-4}$	4.40718	454
0.125	2	42 811.7(5858.1)	563.621	$5.72 \cdot 10^{-5}$	$2.41 \cdot 10^{-5}$	4.39951	6135

Table 6.1: Computational results for the three-dimensional HDG discretisation for the flow around a sphere in a cylinder. `ndof`: Degrees of freedom of the finite element space, `gdof`: Global degrees of freedom (after static condensation), `nze`: Non-zero entries of the linearised system.

for this problem. The drag value converges towards the value established above, and in fact, we can conclude that $-C_{\boldsymbol{F}_3} = 4.398326$. Furthermore, we note that to realise this value to this accuracy, the two-dimensional computation were faster by a factor 100 compared to the three-dimensional computations (2d: $h_{\max} = 0.04, k = 4$, 3d: $h_{\max} = 0.5, k = 6$). We note that this gain in efficiency is not surprising since the matrices resulting from a three-dimensional finite element discretisation are less sparse compared to those resulting from a two-dimensional discretisation. This can be seen by the number of non-zero entries in the stiffness matrix per condensed degree of freedom in Table 6.1 and Table 6.2. Finally, comparing the two velocity solutions from the three and two-dimensional computations in Figure 6.3, we see that the two-dimensional solution does indeed match the three-dimensional solution in the centre plane of the cylinder.

6.1.3 Summary

The above numerical results show that the transformed weak formulation of the Navier-Stokes equations and the corresponding Nitsche and ghost-penalty terms result in a method that retains the optimal order convergence of the underlying discretisation. Comparing the results obtained with a three-dimensional discretisation using a point-wise divergence-free HDG method with the results from a Taylor-Hood, isoparametric CutFEM method, we saw that the drag coefficient from the rotationally reduced formulation converged to the same value as the three-dimensional reference. Furthermore, we saw that the reduced formulation was significantly more efficient in obtaining the drag coefficient accurately, even without any mesh refinement towards the obstacle.

6.2 Experiment

For our work in [Wah+21], we considered specific set-ups of experiments conducted by Hagemeier, Thévenin and Richter [HTR21] and the resulting data set [Hag20] to establish a numerical experiment backed up by physical data for a fluid-structure interaction problem with solid-solid contact. This consists of a single elastic spherical particle falling freely in a viscous fluid and rebounding off the bottom wall of the fluid domain, where the resulting Reynolds numbers are in the transitional regime between creeping and turbulent flows. The multiphase and fluid-structure interaction (FSI) problem with solid contact

Discretisation				Results	
$h_{\max}$	k	$\texttt{dof(gdof)} \times 10^3$	$\texttt{nze} \times 10^6$	$-C_{F_3}$	Time(s)
0.24	2	0.6(0.6)	0.019	4.16118412719	0.483
0.12	2	2.1(2.0)	0.065	4.34105445496	1.780
0.06	2	8.1(7.8)	0.242	4.38343130848	6.898
0.03	2	31.1(30.5)	0.927	4.39459237132	7.868
0.015	2	122.7(121.6)	3.650	4.39717451783	14.135
0.0075	2	486.0(483.8)	14.436	4.39804879025	47.259
0.24	3	1.4(1.1)	0.062	4.32877441109	0.927
0.12	3	5.0(4.1)	0.203	4.38876196773	3.485
0.06	3	19.4(15.8)	0.740	4.39730912657	7.236
0.03	3	75.3(61.4)	2.804	4.39822190768	9.826
0.015	3	298.7(243.9)	10.980	4.39830416341	23.693
0.0075	3	1185.3(968.8)	43.288	4.39832420403	84.184
0.24	4	2.6(1.8)	0.141	4.38079362737	2.127
0.12	4	9.2(6.3)	0.440	4.39689061845	4.953
0.06	4	35.9(24.0)	1.543	4.39825020273	7.971
0.03	4	139.8(92.8)	5.742	4.39832605973	16.746
0.015	4	555.5(367.2)	22.252	4.39832632417	49.775

Table 6.2: Computational results for the rotationally reduced formulation using the isoparametric Taylor-Hood CutFEM discretisation for the flow around a sphere in a cylinder. $\texttt{ndof}$: Degrees of freedom of the finite element space, $\texttt{gdof}$: Global degrees of freedom (after static condensation), $\texttt{nze}$: Non-zero entries of the linearised system.
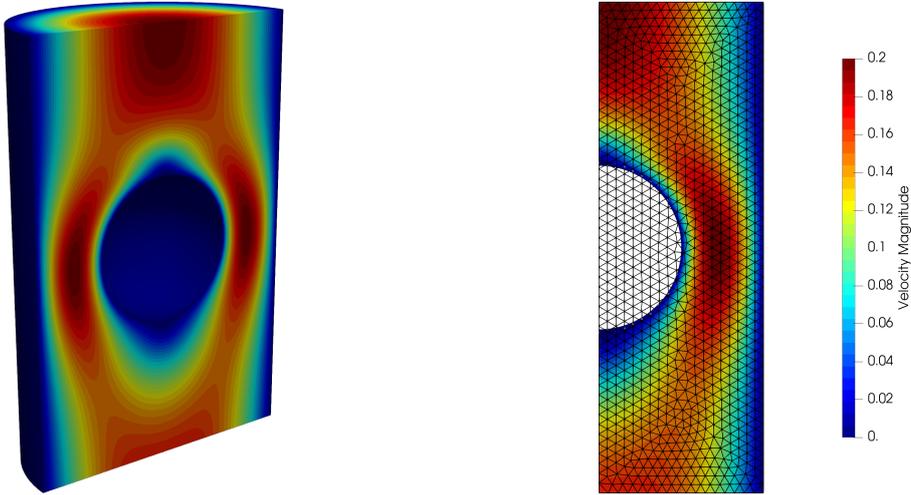
Figure 6.3: Velocity solution for the flow around a sphere in a cylinder. Left: Three-dimensional solution computed on a tetrahedral grid with $h_{\max} = 0.5$ using a point-wise divergence-free HDG discretisation of order $k = 5$. Right: two-dimensional computation with the rotationally reduced formulation using unfitted $\texttt{TH}_4^{\text{iso}}$ elements on a uniform mesh with $h_{\max} = 0.06$.

posed by the settling in the fluid and rebounding off a wall is challenging from both an analytical and numerical perspective.

From the theoretical side, the topic of contact in the context of fluid-structure interactions remains an active area of research. For example, the correct model for the transition to contact with a solid wall is not yet fully understood. If, as in our case, a rigid solid is assumed, most flow models lead to results contradicting real-world observations. For example, if a creeping flow is assumed such that the linear Stokes equations are applicable, contact can only occur under singular forces, c.f. [Bre61]. When the non-linear incompressible Navier-Stokes equations with no-slip boundary conditions are taken for the fluid model, then contact cannot occur, and it is impossible to release contact [Fei03a]. This problem can be overcome if the boundary condition is modified to a free-slip condition [GHC15], the rough nature of the surface is taken into account [GH14] or the fluid is taken to be compressible [Fei03b]. If the solid model is changed to take the body's elasticity into account, then it is currently assumed, that even with perfectly smooth boundaries and incompressibility, rebounding without contact can occur due to the storage of energy in the elastic solid [DSH86; GH16; HT09]. In the setting of the Stokes equations, this has been refined recently [Gra+20], where it has been shown that internal storage of energy is not sufficient, but that additionally, a change in the "flatness" is necessary to achieve physically meaningful rebound without topological contact.

From the numerical side, fluid-structure interaction problems with solid contact also pose significant challenges. The discretisation of the FSI system results in a free boundary value problem with a moving interface. As remarked upon in section 5.1, the most established method for this is the ALE approach. However, the ALE method breaks down when the deformation becomes too large or when contact occurs [FRW16]. To deal with large deformations, overlapping mesh techniques have been developed [JLL15]. Here the background fluid domain and the region around the structure are meshed separately to resolve the fluid-solid interface. The two meshes are then coupled using unfitted approaches. This approach then allows a hybrid approach, where the solid and the near fluid are treated using the ALE framework while the remaining fluid is treated in Eulerian coordinates [SAW19]. To overcome both the problems of large deformations and contact, fully Eulerian approaches have lately become the focus of research. In the case of rigid bodies, a number of different approaches have been considered. For example, based on fictitious domain methods using Lagrange multipliers [Glo+99], XFEM type approaches [CFL13; CF15] and CutFEM approaches using Nitsche's method, such as in chapter 4 and [BFM19; AB21]. A considerable problem for these methods is to achieve a realistic rebound effect since an artificial contact/lubrication force is added to the equation governing the motion of the solid to prevent overlap of the solid regions [Glo+99]. Nevertheless, as we have seen in section 4.5, topological changes appear to be unproblematic for the CutFEM type approaches. In the case of the full fluid elastic structure problem, several different approaches have been considered. For example, immersed approaches mesh the fluid and solid domains separately, treat the equations in their natural Eulerian and Lagrangian formulations, respectively, and then couple the two together, either using Nitsche's method [BF14; HH03] or using Lagrange multipliers [Baa01; LCB06; GW08]. Another new approach to deal both with large deformations and contact is to formulate both the fluid and elastic-solid problems in the Eulerian framework [Dun06; CMM08; Ric13; Fre16; HP17].

All the above methods have been applied to different problem set-ups, and a priori error analysis is available for most approaches. Furthermore, the most well established FSI benchmark [TH06] avoids solid contact, and a number of numerical FSI methods can now specifically deal with solid contact [Age+19; ASW20; BFF20; ZAV21; Bur+21], it is interesting to consider an FSI set-up resulting in solid contact, which is backed up by experimental data. Thereby we are able to validate results rather than interpreting the rebound dynamics qualitatively.

In the remainder of this section, we will give a brief overview of the experiment and describe the resulting set-up extracted from the data for numerical experiments.
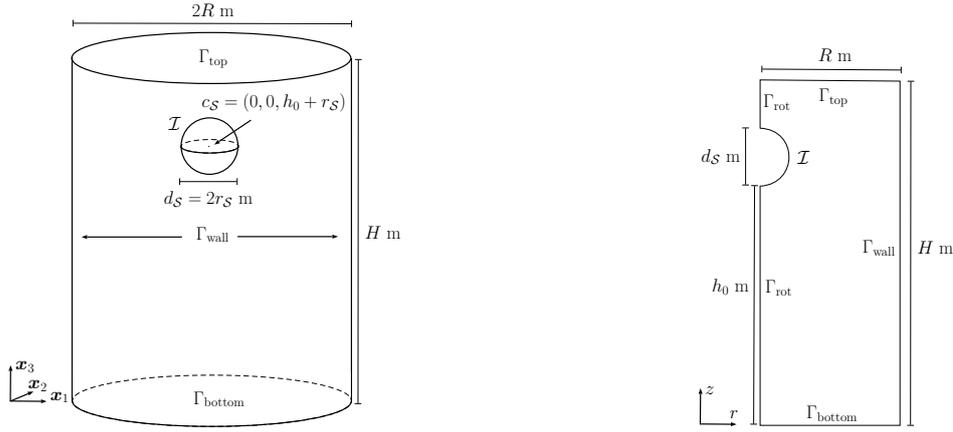
Figure 6.4: Spatial configuration for the simulation of falling ball in a cylindrical container. Left: Three-dimensional domain. Right: Rotationally reduced domain.

Physical Experiment

The experiments in [HTR21] capture the vertical settling of spherical particles of different sizes and densities inside a cylindrical tank filled with a 1:1 water-glycerine mixture. The walls of the cylindrical tank consist of acrylic glass, and the tank sits inside a rectangular container filled with a refractive index matching liquid to compensate for optical distortions caused by the curved walls of the cylinder. This cylinder rests on a steel anvil for an impact object. The filling level then allows for a vertical settling distance of between 140mm–160mm, depending on the particle's size. At the beginning of the experiment, the particle is submerged and held in place by vacuum tweezers. The experiment is then started by switching off the vacuum pump to release the particle. The particle is then tracked in the settling and rebound from the steel anvil with a high-speed CMOS-camera to acquire shadow images at 1000fps and a scale factor of 8.89 pixel/mm. A post-processing algorithm then established the in-plane trajectories and velocities of each particle, and the data is available via Mendeley Data [Hag20].

6.2.1 Problem Description

Geometry Description

Based on the physical experiments, we consider the following domain description for our numerical experiments. We consider the cylindrical domain $\Omega = \{\boldsymbol{x} \in \mathbb{R}^3 \mid \boldsymbol{x}_1^2 + \boldsymbol{x}_2^2 < R^2, 0 < \boldsymbol{x}_3 < H\}$ for a given radius R and a height H. In the initial state at $t = 0$, the solid domain is given by $\mathcal{S}(0) = \{\boldsymbol{x} \in \mathbb{R}^3 \mid \boldsymbol{x}_1^2 + \boldsymbol{x}_2^2 + (\boldsymbol{x}_3 - (h_0 + r_\mathcal{S}))^2 < r_\mathcal{S}^2\}$ for a given radius $r_\mathcal{S}$ of the particle and an initial height h_0 of the bottom of the particle.

We denote the top cylinder boundary at $\boldsymbol{x}_3 = H$ as Γ_{top}, the bottom boundary at $\boldsymbol{x}_3 = 0$ as Γ_{bottom} and the side of the cylinder at $\boldsymbol{x}_1^2 + \boldsymbol{x}_2^2 = R^2$ as Γ_{wall}. A sketch of the domain with the boundary nomenclature can be seen on the left of Figure 6.4.

Boundary Conditions

At the bottom and wall boundaries, we impose a no-slip boundary condition $\boldsymbol{u} = 0$, while at the top boundary, we set a slip boundary condition $\boldsymbol{u}_3 = \boldsymbol{u}_z = 0$ to approximate the free surface of the fluid in the experiments. At the interface $\mathcal{I} = \partial\mathcal{S}$, we impose a no-slip condition, i.e., continuity of the velocity $\boldsymbol{u} = U$. For the rotationally reduced case, appropriate boundary conditions on the rotational axis $r = 0$ are given by $\boldsymbol{u}_r = 0$ and $\partial_r \boldsymbol{u}_z = 0$.

Experiment	Geometry				Boundary Conditions		
	R (m)	H (m)	$r_{\mathcal{S}}$ (m)	h_0 (m)	$\Gamma_{\text{wall}} \cup \Gamma_{\text{bottom}}$	Γ_{top}	$\mathcal{I}$
PTFE6	0.055	0.2	0.003	0.1616616	$\boldsymbol{u} = 0$	$\boldsymbol{u}_d = 0$	$\boldsymbol{u} = U_{\mathcal{S}}$
Rubber22			0.011	0.1461203			

Table 6.3: Spatial parameters for the falling ball in a cylindrical container.

Experiment	Material parameters			
	g (m s^{-2})	$\mu_{\mathcal{F}}$ (kg m^{-1} s^{-1})	$\rho_{\mathcal{F}}$ (kg m^{-3})	$\rho_{\mathcal{S}}$ (kg m^{-3})
PTFE6	−9.807	0.008	1141	2122
Rubber22				1361

Table 6.4: Material parameters for the falling balls in a cylindrical container under consideration.

Rotationally Reduced Domain

The three-dimensional domain is rotationally symmetric in cylindrical coordinates. The experimental data shows a rotational component to the solid motion as well as a small deflection from the centreline. However, since we intend to consider material parameters for the flow to be in the intermediate Reynolds number regime [HTR21], we assume that the solution is described sufficiently well by a rotationally symmetric solution.

Considering the three-dimensional domain in cylindrical coordinates (r, ϕ, z), we rotate the domain into the (r^+, z)-plane to reduce the computational problem into a two-dimensional one, as discussed in detail in section 6.1.

The reduced background domain is then given by $\Omega^{2d} = \{(r, z)^T \in \mathbb{R}^2 \mid 0 < r < R, 0 < z < H\}$, and the reduced initial solid domain is given by $\mathcal{S}^{2d} = \{(r, z)^T \in \mathbb{R}_+^2 \mid r^2 + (z - (h_0 + r_{\mathcal{S}}))^2 < r_{\mathcal{S}}^2\}$. The upper, lower and right boundaries of the fluid domain are then referred to as Γ_{top}, Γ_{bottom} and Γ_{wall}, respectively. A sketch of this domain can be seen on the right of Figure 6.4.

Problem Parameters

We consider two different cases from [HTR21].

Spatial Parameters   The radius of the cylindrical fluid tank is $R = 0.055$m, and the height is $H = 0.2$m. We consider the cases of a Polytetrafluorethylene/Teflon (PFTE) ball with diameter $d_{\mathcal{S}} = 2r_S O = 6$mm and of a rubber ball of diameter $d_{\mathcal{S}} = 22$mm. We refer to these cases as *PTFE6* and *Rubber22*, respectively. The initial heights of these balls are given by $h_0 = 0.1616616$m and $h_0 = 0.1461203$m, respectively. A summary of these parameters is given in Table 6.3.

Material Parameters   For our fluid-rigid body model, we require the fluid's density and viscosity as well as the solid's density. These parameters, as provided in [HTR21, Table 1], are summarised in Table 6.4.

Quantities of Interest

To compare the numerical results with the experimental data, we consider the following quantities of interest:
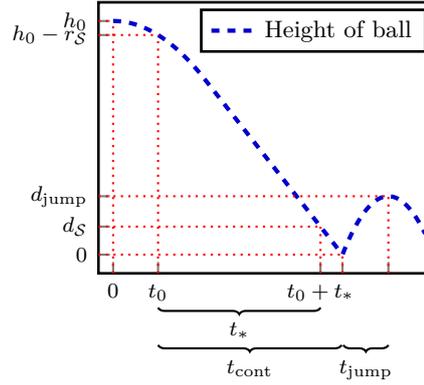
Figure 6.5: Illustration of the quantities of interest for the falling ball in a cylindrical container experiment.

t_* Let $t_0 = t|_{\boldsymbol{c}_{\mathcal{S}} = h_0}$ be the time at which the centre of mass is at h_0, i.e., when the ball has traveled a vertical distance of $r_{\mathcal{S}}$. The reference time t_* is then defined as the time between t_0 and the time at which $\mathrm{dist}(\mathcal{I}, \Gamma_{\mathrm{bottom}}) = d_{\mathcal{S}}$, i.e., for PTFE6 $t_* = t|_{\boldsymbol{c}_{\mathcal{S}} = (0,0,0.009)} - t_0$ and for Rubber22 $t_* = t|_{\boldsymbol{c}_{\mathcal{S}} = (0,0,0.033)} - t_0$.

v_* The velocity of the ball in the z-direction at $t = t_* + t_0$.

f_* The vertical component of the force $\boldsymbol{F}$ acting on the ball at time $t_* + t_0$.

t_{cont} The time of the first solid contact relative to t_0, i.e., $t_{\mathrm{cont}} = t|_{\boldsymbol{c}_{\mathcal{S}} = (0,0,d_{\mathcal{S}}/2)} - t_0$.

t_{jump} The time between contact and the the time at which the second change in direction is realised, i.e., the amount of time the balls travels upwards after the first contact.

d_{jump} The maximum of $\mathrm{dist}(\mathcal{I}, \Gamma_{\mathrm{bottom}})$ after contact, i.e., the size of the bounce.

An illustration of how these quantities are defined can be seen in Figure 6.5.

Remark 6.2 (Design of reference quantities). We observed in the data from the physical experiments that the particles did not immediately start to fall when they are released. While the particles were fully submerged at the beginning of the experiment, the particles were nevertheless close to the free liquid surface and in contact with the vacuum pump. The observation that the particles do not immediately start to settle can therefore be attributed to the fact that particles experience an increased drag force when moving away or towards a free fluid surface or solid wall.

Due to the resulting slow motion at the beginning of the experiment, it is difficult to determine the moment of release from the data set. The time t_0 was therefore chosen as the moment in time, where the particle has already travelled a distance of one radius, which is easily determined both in the experimental and numerical data. The remaining quantities are then defined relative to t_0.   ▲

Remark 6.3. To extract the reference quantities from the experimental data, we interpolated the data using a spline of order 3. The quantities of interest were then computed by evaluating the resulting spline.   ▲

Remark 6.4. The experimental study [HTR21] was conducted, so that the horizontal displacement of the particles was minimal. Looking at the data from the experiments, we see that the maximal horizontal displacement is less than $2.00 \cdot 10^0$ mm and $0.75 \cdot 10^0$ mm in the PTFE6 and Rubber22 cases, respectively. This compares with the mean over time of the maximal deviation in the centre's location between experiment repetitions of $0.19 \cdot 10^0$ mm and $0.14 \cdot 10^0$ mm for the PTFE6 and Rubber22 cases, respectively.

Since the experiment was only able to capture the projection of the horizontal displacement onto the x-z-plane, it is not possible to detect the true horizontal motion. As a result, we have therefore ignored

the horizontal motion in the computation of the reference values. However, as the overall horizontal deflection is small, the results are reasonable for the purposes here. ▲

6.2.2 Mathematical model

By considering the rotationally reduced formulation, we assume that the ball neither rotates nor moves horizontally. Therefore, we consider a reduced form of the fluid-rigid body interaction system (2.5). In the fluid domain, we continue to consider the incompressible Navier-Stokes equations (2.5a)–(2.5b) together with the continuity of the velocity at the fluid-solid interface (2.5c). However, the assumption on the solid velocity reduces the equation of motion for the solid's to $U(t) = \boldsymbol{v}(t)$. The first two components of the solid's velocity are then constant zero and the third component is governed by the scalar ordinary differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}U_3 = \frac{\rho_{\mathcal{S}} - \rho_{\mathcal{F}}}{\rho_{\mathcal{S}}}g + \frac{\boldsymbol{F}_3}{\mathrm{vol}(\mathcal{S})\rho_{\mathcal{S}}}, \tag{6.15}$$

c.f (2.3a). The position of the solid $\boldsymbol{c}_{\mathcal{S}}$ is then recovered via the relationship $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{c}_{\mathcal{S}} = U$.

6.3 Discretisation

The following section discusses the details of our unfitted solver for the coupled fluid-rigid body interaction problem.

The discretisation is based on the rotationally symmetric formulation of the three-dimensional problem and the isoparametric mapping approach from chapter 3 together with the Eulerian time-stepping from chapter 4 for the temporal discretisation of the coupled moving domain problem.

6.3.1 Partitioned Approach for Fluid-Rigid Body Interactions

To solve the coupled fluid-solid system (2.5), we consider a partitioned approach with relaxation. The relaxation update with parameter $\omega \in (0, 1)$ is necessary for the stability of the scheme. The resulting procedure for a single time-step of the fluid-solid problem can be seen in Algorithm 6.1.

Data: Fluid state $(\boldsymbol{u}^n, p^n)$ and solid state $(\boldsymbol{v}^n, \boldsymbol{\omega}^n)$ at time $t = t^n$.

1 Set $(\boldsymbol{u}_0^{n+1}, p_0^{n+1}) = (\boldsymbol{u}^n, p^n)$
2 **for** $i = 0, \ldots, \textit{max\_it}$ **do**
3   Use $(\boldsymbol{u}_i^{n+1}, p_i^{n+1})$ to compute the forces acting on the solid $(\boldsymbol{F}_i^{n+1}, \boldsymbol{T}_i^{n+1})$.
4   Solve for solid states $(\tilde{\boldsymbol{v}}_{i+1}^{n+1}, \tilde{\boldsymbol{\omega}}_{i+1}^{n+1})$ using an ODE solver with right-hand sides $(\boldsymbol{F}_i^{n+1}, \boldsymbol{T}_i^{n+1})$.
5   Do relaxation step $\boldsymbol{v}_{i+1}^{n+1} = \omega\tilde{\boldsymbol{v}}_{i+1}^{n+1} + (1 - \omega)\boldsymbol{v}_i^{n+1}$ and $\boldsymbol{\omega}_{i+1}^{n+1} = \omega\tilde{\boldsymbol{\omega}}_{i+1}^{n+1} + (1 - \omega)\boldsymbol{\omega}_i^{n+1}$.
6   Update solid position via the ODE relation with right-hand side $\boldsymbol{v}_{i+1}^{n+1}, \boldsymbol{\omega}_{i+1}^{n+1}$.
7   Solve fluid equations for $(\boldsymbol{u}_{i+1}^{n+1}, p_{i+1}^{n+1})$ according to the solid states $\boldsymbol{v}_{i+1}^{n+1}$ and $\boldsymbol{\omega}_{i+1}^{n+1}$.
8   **if** $\|\boldsymbol{v}_{i+1}^{n+1} - \boldsymbol{v}_i^{n+1}\| < \textit{tol}$ *and* $\|\boldsymbol{\omega}_{i+1}^{n+1} - \boldsymbol{\omega}_i^{n+1}\| < \textit{tol}$ **then**
9   **break**

Result: Updated states $(\boldsymbol{u}_i^{n+1}, p_i^{n+1})$ and $(\boldsymbol{v}_i^{n+1}, \boldsymbol{\omega}_i^{n+1})$ solving the fluid-solid system implicitly up to the given tolerance $\textit{tol} > 0$.

Algorithm 6.1: Partitioned iteration scheme to solve the fluid-solid system.

Our unfitted Eulerian time-stepping method from chapter 4 works by including extension ghost-penalties such that all elements in the active fluid domain $\mathcal{O}_{\mathcal{F}}^n$ have a sufficient history for the right-hand side of the finite difference approximation $\partial_t \boldsymbol{u}_h(t^n) \approx \frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})$ to be well-defined. In chapter 4, this was

realised by applying the extension ghost-penalties in a δ_h-strip $\mathcal{S}^{\pm}$ around the interface, with the choice $\delta_h = c_{\delta_h} \boldsymbol{w}_{\infty}^{\boldsymbol{n}} \Delta t$ guaranteeing that the strip is sufficiently large based on the a priori knowledge of $\boldsymbol{w}_{\infty}^{\boldsymbol{n}}$. This was possible because the domain motion was a given quantity. In the coupled fluid-solid problem under consideration here, the motion of the interface is part of the solution. As we have also seen in section 4.5, simply applying the extension in a very large δ-strip, based on some educated guess of the terminal settling velocity is not a good option since this deteriorates the solution significantly. In order to realise a stable extension method, while also only applying the extension where it is strictly necessary, we compute a new δ_h in each time-step based on $\boldsymbol{w}_n^n = \max\{\boldsymbol{v}^{n-1}\boldsymbol{n}, (2\boldsymbol{v}^{n-1} - \boldsymbol{v}^{n-2})\boldsymbol{n}\}$, i.e., the maximum of a first- and second-order extrapolation of the solid's velocity as a good guess for the necessary extension for the next time-step.

Aitken Relaxation

The idea of Aitken's Δ^2 method is to use the values from two previous iteration steps to predict an optimal relaxation parameter. For simplicity of notation, we focus on the velocity component $\boldsymbol{v}_i^{n+1}$ to describe the method. The subscript i here denotes the step of the sub-iteration in a time-step. Since the procedure is the same for every time-step, we drop the explicit superscript $n+1$.

Given two solution states $(\tilde{\boldsymbol{v}}_i, \boldsymbol{v}_{i-1})$ and $(\tilde{\boldsymbol{v}}_{i+1}, \boldsymbol{v}_i)$, the next state used for the boundary condition is

$$\boldsymbol{v}_{i+1} = \frac{\boldsymbol{v}_{i-1}\tilde{\boldsymbol{v}}_{i+1} - \tilde{\boldsymbol{v}}_i\boldsymbol{v}_i}{\boldsymbol{v}_{i-1} - \tilde{\boldsymbol{v}}_i - \boldsymbol{v}_i + \tilde{\boldsymbol{v}}_{i+1}}. \tag{6.16}$$

Now, allowing for a variable relaxation parameter, we can rewrite the relaxation step in line 5 of Algorithm 6.1 as

$$\begin{aligned} \boldsymbol{v}_{i+1} &= \omega_{i+1}\tilde{\boldsymbol{v}}_{i+1} + (1 - \omega_{i+1})\boldsymbol{v}_i \\ &= \boldsymbol{v}_i + \omega_{i+1}(\tilde{\boldsymbol{v}}_{i+1} - \boldsymbol{v}_i). \end{aligned} \tag{6.17}$$

We denote $\tilde{\boldsymbol{v}}_{i+1} - \boldsymbol{v}_i =: r_{i+1}$ as the residual in the $(i+1)^{\text{th}}$ sub-iteration step. Setting (6.16) equal to (6.17), we find that the relaxation parameter is

$$\omega_{i+1} = \frac{\boldsymbol{v}_{i-1} - \boldsymbol{v}_i}{(\boldsymbol{v}_{i-1} - \tilde{\boldsymbol{v}}_i) - (\boldsymbol{v}_i - \tilde{\boldsymbol{v}}_{i+1})}.$$

From (6.17), for $i \mapsto i-1$, we have $\boldsymbol{v}_{i-1} - \boldsymbol{v}_i = \omega_i(\boldsymbol{v}_{i-1} - \tilde{\boldsymbol{v}}_i)$. The formula for the $(i+1)^{\text{th}}$ relaxation parameter then has the iterative representation

$$\begin{aligned} \omega_{i+1} &= \omega_i \frac{\boldsymbol{v}_{i-1} - \tilde{\boldsymbol{v}}_i}{(\boldsymbol{v}_{i-1} - \tilde{\boldsymbol{v}}_i) - (\boldsymbol{v}_i - \tilde{\boldsymbol{v}}_{i+1})} \\ &= -\omega_i \frac{r_i}{r_{i+1} - r_i}. \end{aligned}$$

We note that the division in the above expression only makes sense for scalar quantities. For vector valued quantities, it is suggested in [KW08] to follow [IT69], and to use the vector inverse. This then gives the relaxation parameter

$$\omega_{i+1} = -\omega_i \frac{\boldsymbol{r}_i^T(\boldsymbol{r}_{i+1} - \boldsymbol{r}_i)}{|\boldsymbol{r}_{i+1} - \boldsymbol{r}_i|^2}.$$

Since the computation of the relaxation parameter is not possible in the first iteration, the relaxation ω_1 must be chosen differently. We suggest to use $\min\{\omega_0, \omega_{max}^n\}$, where ω_0 is some manually specified relaxation parameter and ω_{max}^n is the final relaxation parameter from the last time-step. The minimum is taken to ensure that the first relaxation is not too large, potentially leading to the instabilities that we aim to avoid by using a relaxation scheme.

6.3.2 Contact Algorithm

One of the biggest challenges in simulating the fluid-structure interaction problem described in section 6.2 is the occurrence of contact when the particles rebound off the bottom of the fluid tank. As we have seen in subsection 4.5.2, our unfitted Eulerian discretisation can deal with contact/topology changes. However, as we have described above, there are already issues due to solid-solid contact in fluid-structure interaction problems on the modelling side. On the practical side, we cannot realise the forces necessary to prevent contact with the mesh size and time-steps used in practice. Therefore, we require some additional methodology to ensure that a rebound is achieved.

Inspired by [ST08; Fre16; Fre17; FR17a], we consider a basic contact avoidance algorithm. The idea of this algorithm is to introduce an artificial force acting on the rigid body in the vicinity of the contact wall, which increases the closer the ball comes to the wall and acts in the direction away from the wall such that contact does not occur. We define this force as

$$f_c(\mathcal{S}) = \begin{cases} 0 & \text{if } \text{dist}(\mathcal{I}, \Gamma_c) \geq dist_0 \\ \gamma_c \frac{dist_0 - \text{dist}(\mathcal{I}, \Gamma_c)}{\text{dist}(\mathcal{I}, \Gamma_c)} & \text{if } \text{dist}(\mathcal{I}, \Gamma_c) < dist_0, \end{cases} \tag{6.18}$$

where $dist_0$ and γ_c are free parameters which remain to be chosen, Γ_c is the contact wall and $\text{dist}(\mathcal{I}, \Gamma_c)$ is the minimal distance between the solid boundary and the contact wall. Therefore, the penalty force is 0 when the solid is at a considerable distance from the contact wall and goes to $+\infty$ when the distance between the wall and the solid interface goes to zero. We then add this artificial force to the right-hand side of the ODE (2.3a). For the specific application of vertical motion with a rebound off the bottom of the fluid tank, we take $\Gamma_c = \Gamma_{\text{bottom}}$. As a result, we add this penalty force to the vertical component of (2.3a). This change is then carried through to (6.15), so that the right-hand side becomes $\frac{\rho_{\mathcal{S}} - \rho_{\mathcal{F}}}{\rho_{\mathcal{S}}} g + \frac{\boldsymbol{F}_z + f_c}{\text{vol}(\mathcal{S})\rho_{\mathcal{S}}}$.

Remark 6.5. In [Fre16; Fre17], the artificial force was defined to act point-wise on the elastic body. As a result, the force could be interpreted as an additional pressure acting below $dist_0$. In our case, the force acts on the ODE governing the motion of the rigid solid. Therefore, we can interpret the contact force as additional buoyancy. As we have seen in Remark 2.1, this inclusion of buoyancy in the solid ODE is a consequence of not including gravity on the right-hand side of the fluid equation. If gravity were to be included in the fluid equation, this would only affect the pressure. Therefore, the extra buoyancy here can also be seen as an additional pressure acting on the solid. ▲

6.4 Numerical Computations

We use the temporal discretisation methodology as described in section 6.3 with our Eulerian time-stepping scheme using a BDF2 discretisation of the temporal derivatives to simulate the PTFE6 and Rubber22 scenarios described in section 6.2. For the spatial discretisation, we use the rotationally reduced formulation of the Navier-Stokes equations derived in section 6.1 using unfitted finite elements.

The rotationally reduced background domain $\Omega^{2d} = (0, 0.055) \times (0, 0.2)$ is meshed with a global mesh parameter $h_{\max}$ and the left part with $0 \leq r \leq 2/3 \cdot d_{\mathcal{S}}$ has a local mesh parameter of $h = h_{\max}/7$ and $h = h_{\max}/4$ for the PTFE6 and Rubber22 cases, respectively. This is to resolve the interface of the falling ball more accurately and the local mesh size is chosen in order to construct meshes with a similar number of active elements for the two scenarios.

On these meshes, we consider $\texttt{TH}_2^{\text{iso}}$ elements and the Nitsche parameter is chosen as $\sigma = 100k^2$. As we have seen in section 4.5, it is beneficial to choose a smaller ghost-penalty parameter for the $1/\nu$-scaled ghost-penalties. We therefore set the stabilising ghost-penalty parameters as $\gamma_{u,gp,1} = \gamma_{p,gp} =: \gamma_{gp,s} = 0.1$ and the extension ghost-penalty parameter as $\gamma_{u,gp,2} =: \gamma_{pg,e} = 0.01$, respectively. The extension-strip parameter is $c_\delta = 4$ to compensate for any underestimation of $\boldsymbol{w}_\infty^{\boldsymbol{n}}$. We iterate between the fluid and solid systems until the velocity update is below the tolerance of $\texttt{tol} < 10^{-8}$.

$h_{\max}$	Δt	t_*			v_*			f_*		
		value	|err|	eoc	value	|err|	eoc	value	|err|	eoc
0.008	$1/2000$	0.579181	$4.017 \cdot 10^{-2}$		-0.289922	$2.407 \cdot 10^{-2}$		$1.116\,00 \cdot 10^{-3}$	$4.420 \cdot 10^{-6}$	
0.004	$1/2000$	0.536698	$2.312 \cdot 10^{-3}$	4.12	-0.316896	$2.900 \cdot 10^{-3}$	3.05	$1.087\,00 \cdot 10^{-3}$	$3.327 \cdot 10^{-5}$	-2.91
0.002	$1/2000$	0.530537	$8.473 \cdot 10^{-3}$	-1.87	-0.321826	$7.830 \cdot 10^{-3}$	-1.43	$1.082\,00 \cdot 10^{-3}$	$3.836 \cdot 10^{-5}$	-0.21
0.002	$1/1000$	0.522575	$1.644 \cdot 10^{-2}$		-0.328963	$1.497 \cdot 10^{-2}$		$1.081\,00 \cdot 10^{-3}$	$3.914 \cdot 10^{-5}$	
0.002	$1/2000$	0.530537	$8.473 \cdot 10^{-3}$	0.96	-0.321826	$7.830 \cdot 10^{-3}$	0.93	$1.082\,00 \cdot 10^{-3}$	$3.836 \cdot 10^{-5}$	0.03
0.002	$1/4000$	0.537140	$1.870 \cdot 10^{-3}$	2.18	-0.314027	$3.098 \cdot 10^{-5}$	7.98	$1.092\,00 \cdot 10^{-3}$	$2.845 \cdot 10^{-5}$	0.43
0.008	$1/500$	0.559084	$2.007 \cdot 10^{-2}$		-0.298432	$1.556 \cdot 10^{-2}$		$1.098\,00 \cdot 10^{-3}$	$2.207 \cdot 10^{-5}$	
0.004	$1/1000$	0.524553	$1.446 \cdot 10^{-2}$	0.47	-0.328431	$1.444 \cdot 10^{-2}$	0.11	$1.085\,00 \cdot 10^{-3}$	$3.559 \cdot 10^{-5}$	-0.69
0.002	$1/2000$	0.530537	$8.473 \cdot 10^{-3}$	0.77	-0.321826	$7.830 \cdot 10^{-3}$	0.88	$1.082\,00 \cdot 10^{-3}$	$3.836 \cdot 10^{-5}$	-0.11
Reference		0.53901			-0.313996			$1.120\,21 \cdot 10^{-3}$		

Table 6.5: Results for the pre-contact quantities of interest for the PTFE6 scenario. Reference values are taken from the extrapolated ALE results in [Wah+21].

6.4.1 Pre-Contact

We consider the pre-contact dynamics of the system. This is the part of the experiment considered in our theory in chapter 5.

PTFE6

We compute the PTEF6 set-up as described above and consider mesh-refinement with a constant time-step, time-step refinement on a constant mesh and combined mesh and time-step refinement. The full results for the pre-contact quantities of interest can be seen in Table 6.5, and the resulting height of the ball under mesh refinement can be seen below in Figure 6.11. The reference values in Table 6.5 are based on the extrapolated results of the ALE simulation of the fluid-solid system in [Wah+21].

In general, we see that our method approximates the dynamics of the system well. Looking at the convergence behaviour of the quantities of interest in Table 6.5, we see that it appears that the temporal discretisation error is the dominating factor in the presented simulations. While we even see some diverging behaviour in the drag reference quantity, we see convergence both for the time at which the ball reaches the distance of $d_{\mathcal{S}}$ to the bottom of the tank and the solid velocity at this moment in time. We attribute this to the instabilities we have previously observed in the drag functional in section 4.5. Nevertheless, in Figure 6.11, we see that the ball's position converges towards the experimental data.

Remark 6.6. A video of the solution on the mesh with $h_{\max} = 0.002$ is available at `https://youtu.be/TuoIP94g1Hs` or by scanning the QR code. ▲

Rubber22

We compute the Rubber22 set-up as described above and consider mesh-refinement, time-step refinement and combined mesh/time-step refinement. The resulting values for the quantities of interest can be seen in Table 6.6, and the height resulting from the computations under mesh-refinement can be seen in Figure 6.12. Furthermore, a visualisation of the velocity and pressure solutions can be seen in Figure 6.6.

For the reference quantity t_*, we see convergence under time and diagonal refinement. This again indicates that while the scheme is of higher order, the temporal discretisation is the dominating error component. While we see some convergence in the drag f_*, we cannot see convergence in v_*. Since we have the convergence in t_*, we attribute the lack of convergence to the inaccuracies and instabilities in the force evaluation, as f_* and v_* represent quantities that are by definition more sensitive to local in

$h_{\max}$	Δt	t_*			v_*			f_*		
		value	|err|	eoc	value	|err|	eoc	value	|err|	eoc
0.008	$1/2000$	0.453455	$1.877 \cdot 10^{-3}$		-0.282641	$2.098 \cdot 10^{-2}$		0.009 64	$1.670 \cdot 10^{-3}$	
0.004	$1/2000$	0.454081	$1.251 \cdot 10^{-3}$	0.59	-0.281923	$2.170 \cdot 10^{-2}$	-0.05	0.010 26	$1.048 \cdot 10^{-3}$	0.67
0.002	$1/2000$	0.453789	$1.543 \cdot 10^{-3}$	-0.30	-0.282187	$2.143 \cdot 10^{-2}$	0.02	0.010 41	$9.035 \cdot 10^{-4}$	0.21
0.002	$1/1000$	0.450554	$4.778 \cdot 10^{-3}$		-0.284958	$1.866 \cdot 10^{-2}$		0.010 25	$1.065 \cdot 10^{-3}$	
0.002	$1/2000$	0.453789	$1.543 \cdot 10^{-3}$	1.63	-0.282187	$2.143 \cdot 10^{-2}$	-0.20	0.010 41	$9.035 \cdot 10^{-4}$	0.24
0.002	$1/4000$	0.455396	$6.416 \cdot 10^{-5}$	4.59	-0.280821	$2.280 \cdot 10^{-2}$	-0.09	0.010 51	$8.020 \cdot 10^{-4}$	0.17
0.008	$1/500$	0.447853	$7.479 \cdot 10^{-3}$		-0.287129	$1.649 \cdot 10^{-2}$		0.010 47	$8.429 \cdot 10^{-4}$	
0.004	$1/1000$	0.449218	$6.114 \cdot 10^{-3}$	0.29	-0.285817	$1.780 \cdot 10^{-2}$	-0.11	0.009 97	$1.343 \cdot 10^{-3}$	-0.67
0.002	$1/2000$	0.453789	$1.543 \cdot 10^{-3}$	1.99	-0.282187	$2.143 \cdot 10^{-2}$	-0.27	0.010 41	$9.035 \cdot 10^{-4}$	0.57
Reference		0.455332			-0.30362			0.011 31		

Table 6.6: Results for the pre-contact quantities of interest for the Rubber22 scenario. Reference values are taken from the extrapolated ALE results in [Wah+21].

time inaccuracies compared to t_*. In total, we can conclude that the fluid-solid model can capture the pre-contact dynamics well, even for the more elastic rubber solid material.

6.4.2 Contact and Rebound

We investigate the behaviour of our numerical method, with focus on the contact algorithm and the resulting rebound.

PTFE6

To investigate the effect and usability of our simple contact avoidance model, we consider a number of simulations together with different parameters in the contact model. To this end, we consider $\mathrm{TH}_2^{\mathrm{iso}}$ elements on a mesh with $h_{\max} = 0.004$ and $h = h_{\max}/7$ in the central region where the ball is situated, together with the time-step $\Delta t = 1/2000$.

We consider $dist_0 \in \{1 \cdot 10^{-4}, 2 \cdot 10^{-4}, 4 \cdot 10^{-4}\}$ and a range of values for γ_c such that the forces acting on the ball are broadly similar. The resulting height of the bottom of the ball for each of these computations can be seen in Figure 6.7. For the smallest value $dist_0 = 10^{-4}$, we see that a rebound only occurs for a very small range of γ_c. In fact, the algorithm failed for $\gamma_c = 0.21$. Note that here $dist_0 \approx 0.2h$ for the local mesh-size at the contact boundary. Increasing the value of $dist_0$ to $2 \cdot 10^{-4}$, we see that the scheme is successful for a much wider range of γ_c. However, we also note that the resulting rebound is highly dependent on the choice of γ_c, with the rebound being about three times as large for $\gamma_c = 0.76$ compared with the other considered values of γ_c. Finally, for $dist_0 = 4 \cdot 10^{-4}$, we see that the resulting rebound is less dependent on the choice of γ_c and that the resulting rebound is larger than the experimental value of $2.21 \cdot 10^{-3}$ m for all considered choices here.

To study the effects of the contact model more closely, we look at the height of the ball for the choice $dist_0 = 2.00 \cdot 10^{-4}$ at the time-steps where the contact model is active in Figure 6.8. We have marked the time-steps where the contact model is active, i.e., $\mathrm{dist}(\mathcal{I}, \Gamma_{\mathrm{bottom}}) < dist_0$, using square mark and the remaining time-steps are identified with circular marks. In Figure 6.8, we see that for smaller γ_c, the ball is allowed to drop lower, and the contact model is active for a larger number of time-steps. Looking further at the forces acting on the rigid body during this time in Figure 6.9, we see that relative to the drag force where we have no contact model, the contact force is very large. However, the drag experienced by the ball during these time-steps is also very large, so that the total force acting on the ball is about half the value of the contact force. The very large drag experienced can be explained by the observation that the sudden upward movement of the ball, due to the repelling contact force, created a very high negative
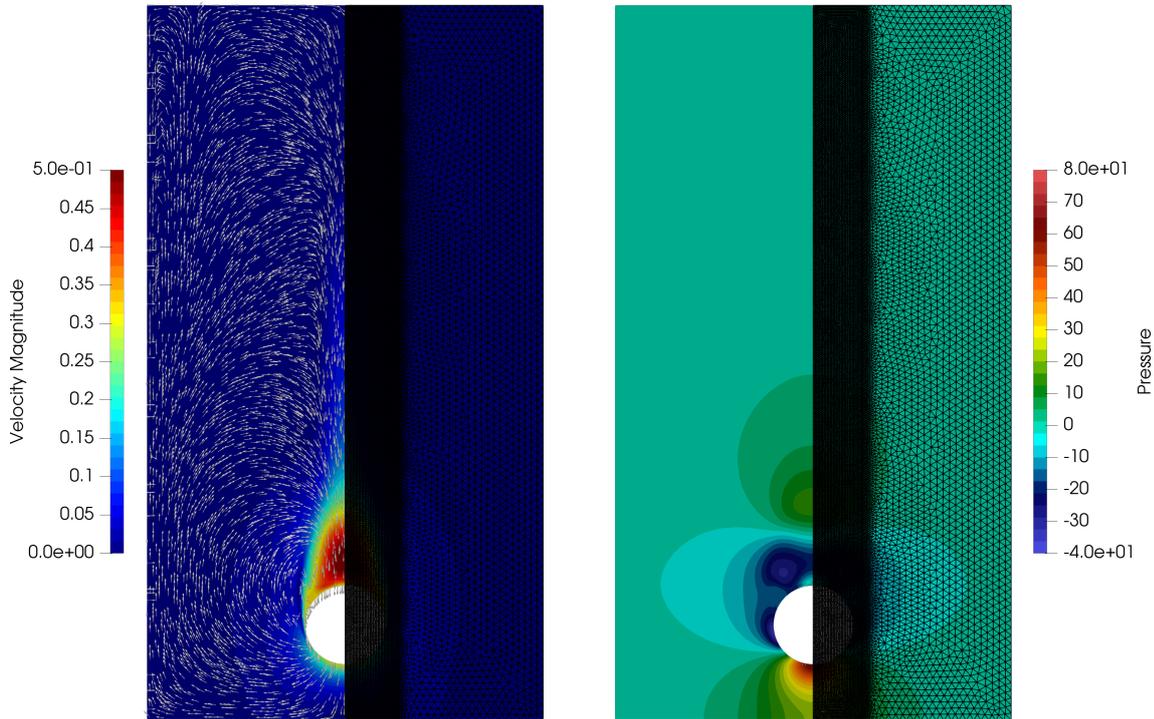
Figure 6.6: Velocity and pressure solution for the Rubber22 set-up at $t = 0.625$. Computed using TH_2^{iso} elements on a mesh with $h_{\text{max}} = 0.002$ with $\Delta t = \text{}^1/_{2000}$. (A video of the simulation is available at https://youtu.be/fls9V5z212Q.)
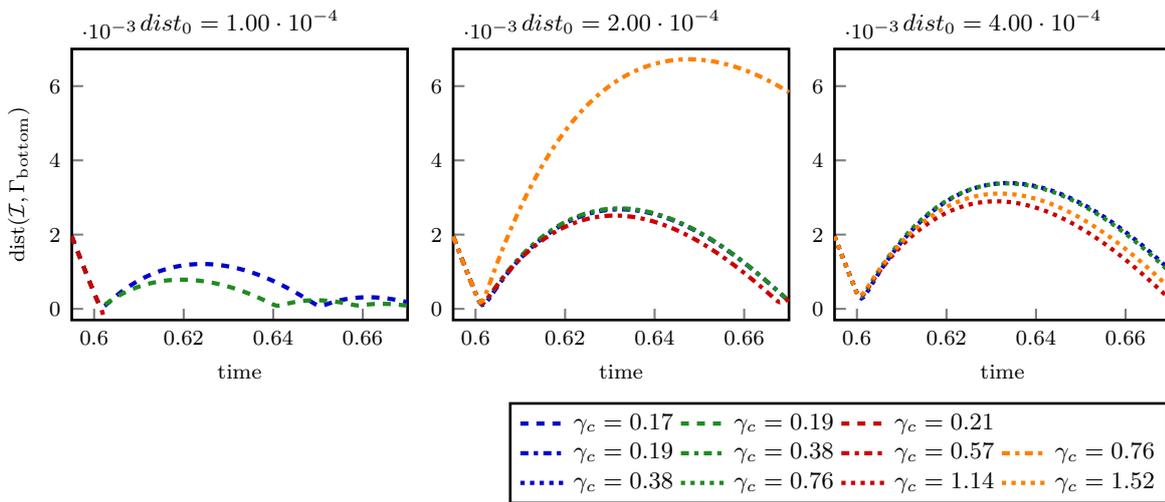
Figure 6.7: Rebound of the PFTE6 ball depending on the free parameters of the contact model. Computed on a mesh with $h_{\text{max}} = 0.004$ and the time-step $\Delta t = \text{}^1/_{2000}$.
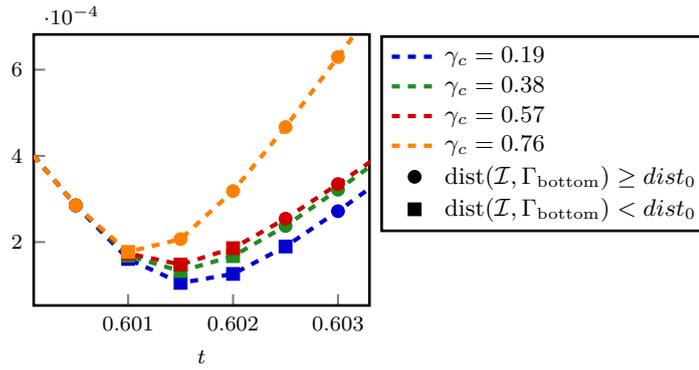
Figure 6.8: Close up of the PFTE6 ball's height while the contact model is active for $dist_0 = 2 \cdot 10^{-4}$. Computed on a mesh with $h_{\max} = 0.004$, and the time-step $\Delta t = 1/2000$.
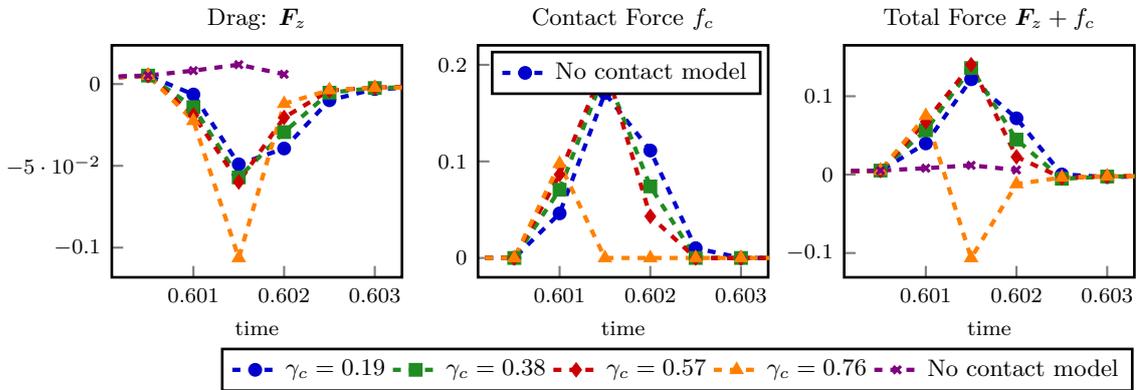
Figure 6.9: The forces acting on the rigid PTFE6 ball during the time the contact model is active for $dist_0 = 2 \cdot 10^{-4}$. Computed on a mesh with $h_{\max} = 0.004$ and the time-step $\Delta t = 1/2000$.

pressure between the ball and the bottom wall. This effectively sucks the ball back down, as shown in the negative drag force realised here. This is further illustrated when looking at the finite element solution during these time-steps in Figure 6.10.

Finally, we consider the effects of the contact model on the rebound under mesh refinement. We choose the parameters $\gamma_c = 0.38$ and $dist_0 = 2 \cdot 10^{-4}$ since these parameters lead to the most realistic rebound above. A plot of the resulting height can be seen in Figure 6.11, while the contact and rebound quantities of interest can be seen in Table 6.7.

Looking at the results in Figure 6.11, we see that the rebound is similar to the experimental data on the two finest meshes. Furthermore, we see some convergence of the quantities of interest towards the experimental reference values in Table 6.7.

$h_{\max}$	Δt	t_{cont}	t_{jump}	d_{jump}
0.008	$1/2000$	0.600328	0.017827	$1.339\,74 \cdot 10^{-3}$
0.004	$1/2000$	0.556037	0.030627	$2.706\,81 \cdot 10^{-3}$
0.002	$1/2000$	0.549543	0.030749	$2.692\,50 \cdot 10^{-3}$
Experiment		0.534503	0.02792	$2.211\,70 \cdot 10^{-3}$

Table 6.7: Contact and rebound results results for the PTFE6 scenario. The experimental values are from [Hag20].
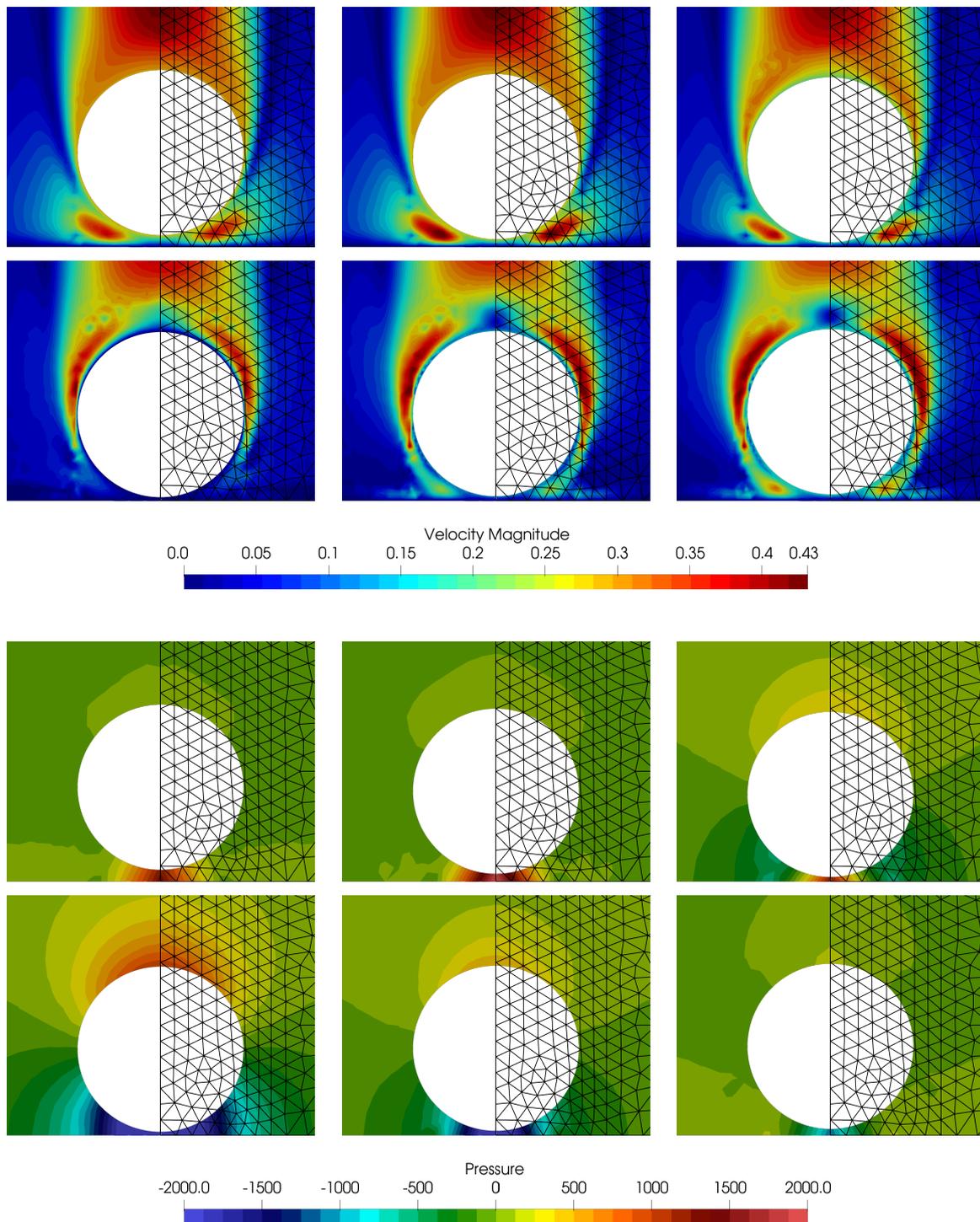
Figure 6.10: Close up of the PTFE6 velocity and pressure solution at the contact boundary for $0.6 \leq t \leq 0.6025$. The artificial contact force acts at $t = 0.601, 0.6015, 0.602$.
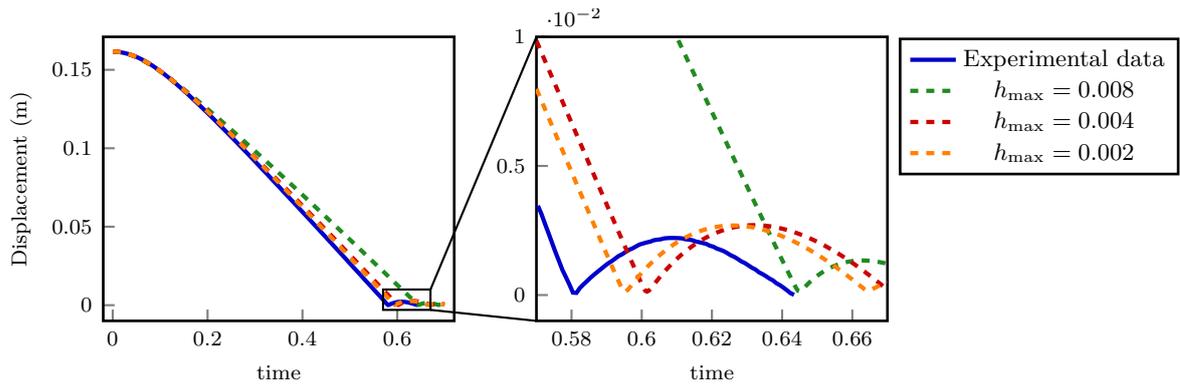
Figure 6.11: Vertical height of the PTFE6 ball over time for a series of meshes using the time-step $\Delta t = 1/2000$ and the contact parameters $dist_0 = 2 \cdot 10^{-4}$ and $\gamma_c = 0.38$, together with the experimental data.
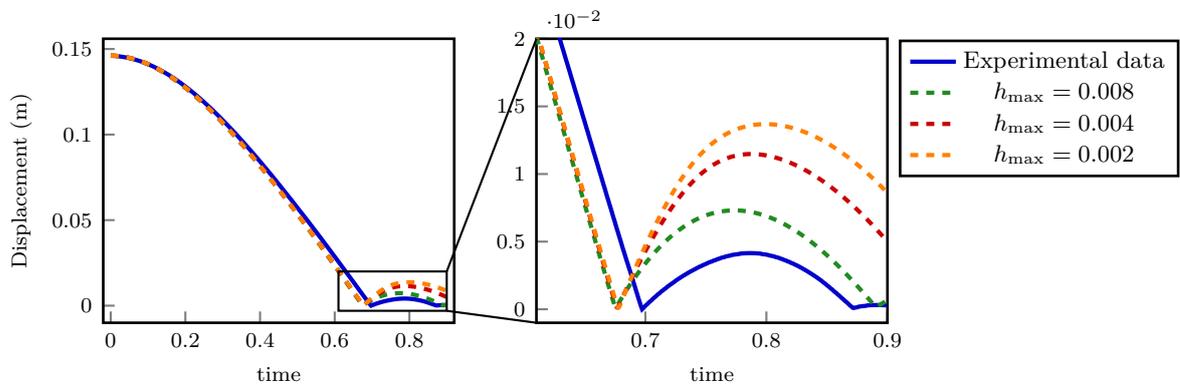
Figure 6.12: Vertical height of the Rubber22 ball over time for a series of meshes using the time-step $\Delta t = 1/2000$ and the contact parameters $dist_0 = 2 \cdot 10^{-4}$ and $\gamma_c = 0.38$, together with the experimental data.

Rubber22

Having considered "tuned" parameters for the PTFE6 case, we investigate how these parameters perform when applied to the Rubber22 case. Since the rubber ball has a mass ~ 31.6 times larger than the PTFE ball, we increase the contact force parameter to $\gamma_c = 12$. As a result, the acceleration acting on the solid body, depending on $\mathrm{dist}(\mathcal{I}, \Gamma_{\mathrm{bottom}})$, is the same.

The resulting height of the ball can be seen in Figure 6.12. Here we see that the resulting rebound is significantly larger compared to the experimental data. In fact, the rebound is between two to three times larger than in the experiment. This illustrates that while the contact algorithm can lead to reasonable rebound results, the accuracy is highly dependent on the "correct" choice of the model parameters, which are unknown a priori.

6.4.3 Summary

From the above numerical computation of the considered fluid-rigid body scenarios, we are able to conclude the following for our Eulerian time-stepping method, applied to a coupled fluid-rigid body system. First, we note that we observed higher-order convergence in the time at which the solid reached a given position and the velocity of the body at that time. Here we saw that the temporal discretisation error was the dominating error component.

With respect to the simple contact avoidance algorithm, we conclude that while it is possible to tune

the parameters to realise a realistic rebound, this task is not easy. The model is especially sensitive with respect to the parameter $dist_0$. We have also seen that, while the parameters tuned to the PTFE6 case were able to cause a rebound for the Rubber22 scenario, the resulting jump was off by up to a factor of three. Therefore, using a full FSI model that takes the modelling of the elastic body into account, also considered in our work [Wah+21], is the most accurate approach to deal with the rebound.

Finally, we note that for the above scenarios, the rotationally reduced formulation could capture the three-dimensional dynamics of the experimental data well, with significant gains in efficiency, as illustrated in section 6.1.

Non-Smooth Rigid Bodies

Structure of Chapter   The fluid-rigid body system (2.5) is considered in the context of non-smooth rigid-body particles. As a prototype for a particle with corners, we consider triangular bodies. The Eulerian time-stepping scheme is considered both for prescribed and free rotational motion of large triangular bodies on resolved meshes. In order to work with the time-stepping scheme on problems with small particles on under-resolved meshes, we develop an artificial deep neural network approach following our work in [WR21]. This neural network is trained to predict the forces acting on small triangular particles accurately based on volumetric fluid information in under-resolved cases where the directly evaluated forces do not lead to accurate solid motion.

Contents of Chapter

7.1 Rotating Triangular Bodies in a Channel Flow

In the previous chapter, we have considered smooth geometries defined using a single level set function. We now move away from this setting into a more complex area by considering non-smooth geometries, defined with multiple level set functions.

As a prototype of a non-smooth rigid particle in two dimensions, we consider triangular rigid bodies. This is a challenging scenario since triangular bodies involve re-entrant corners in the fluid domain, whereby the approximation property of the finite element solution suffers due to a lack of regularity of the solution, posing a challenge for the coupled simulation. Therefore, the setting of triangular particles contains all the challenges posed by considering non-smooth domains with re-entrant corners while keeping the parametrisation of the level set geometry as simple as possible. To this end, we initially consider a single, large triangular particle in two settings. First, we prescribe the motion of the triangular body, and second, we consider free rotational around the body's centre of mass. These are already settings where standard ALE methods will fail due to mesh entanglement such that frequent re-meshing and projection onto the new finite element spaces are needed.
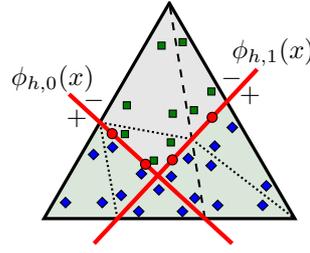
Figure 7.1: Quadrature points for an integration rule of order 2 on an element cut by two level sets.
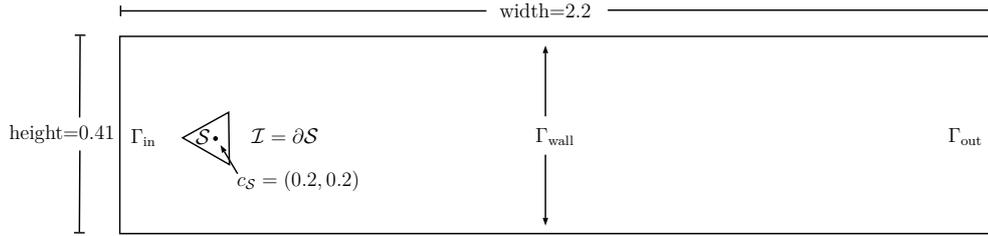
Figure 7.2: Domain sketch and initial configuration for the flow around a rotating triangular obstacle.

7.1.1 Complex Geometries Defined by Multiple Level Set Functions

Let $\Omega_0 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \phi_0(\boldsymbol{x}) < 0\}$ and $\Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \phi_1(\boldsymbol{x}) < 0\}$ be two simple domains described by two level set functions ϕ_0 and ϕ_1, respectively. We can then construct more complex geometries using standard Boolean set operations, which in turn can be translated into a manipulation of the level sets. For example

$$\Omega_0 \cup \Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \min\big(\phi_0(\boldsymbol{x}), \phi_1(\boldsymbol{x})\big) = \phi(\boldsymbol{x}) < 0\}$$
$$\Omega_0 \cap \Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \max\big(\phi_0(\boldsymbol{x}), \phi_1(\boldsymbol{x})\big) = \phi(\boldsymbol{x}) < 0\}$$
$$\Omega_0 \setminus \Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \max\big(\phi_0(\boldsymbol{x}), -\phi_1(\boldsymbol{x})\big) = \phi(\boldsymbol{x}) < 0\}$$
$$\Omega_0 \triangle \Omega_1 = \{\boldsymbol{x} \in \mathbb{R}^d \mid \min\big(\max\big(\phi_0(\boldsymbol{x}), -\phi_1(\boldsymbol{x})\big), \max\big(\phi_1(\boldsymbol{x}), -\phi_0(\boldsymbol{x})\big)\big) = \phi(\boldsymbol{x}) < 0\},$$

c.f. [Bur+14, section 4].

Compared to the previous cases with a single level set function, constructing the cut quadrature on elements cut by both level sets needs more care. In CutFEM, the quadrature rules on cut elements are usually constructed with respect to a piecewise linear level set function, c.f. section 3.1. If one takes the single level set ϕ constructed using ϕ_0 and ϕ_1, then on elements where both $\phi_0 = 0$ and $\phi_1 = 0$, a piecewise linear approximation ϕ_h would not take the sharp corner on the interior of the element into account. To integrate accurately on domains described by multiple level set functions, `ngsxfem` uses piecewise linear approximations of the individual level sets to then constructing the quadrature rules based on the resulting straight cuts; see Figure 7.1 and compare with Figure 3.1.

7.1.2 Example 1: Prescribed Motion

Based on our work in [Wah+19b], we consider a rotating rigid body in a channel flow. We consider the domain $\Omega = (0, 2.2) \times (0, 0.41)$ and the solid is given by an equilateral triangle of side-length $L = 0.1$ with the centre of mass at $c_{\mathcal{S}} = (0.2, 0.2)$. The initial rotation of the triangle can be seen in Figure 7.2.

At the top and bottom boundary Γ_{wall}, we impose a no-slip Dirichlet condition $\boldsymbol{u} = 0$. On the outflow boundary Γ_{out}, we take the "do-nothing" outflow condition, and at the inflow boundary Γ_{in}, we impose a parabolic inflow profile (3.43). Since the motion of the triangle is restricted to rotation arounds its centre

---

Data: Triangle $\mathtt{t}_{in}$ with vertices abc and angle α with which the triangle is rotated for the next time-step.

Result: Elements on which to apply ghost-penalties needed for the extension.

```
/* We denote the sides of the triangle abc opposite each vertex as ABC. Each side
   is assumed to be described by a level set function.                           */
```

1 Make a copy $\mathtt{t}_r$ of the input triangle $\mathtt{t}_{in}$ and rotate $\mathtt{t}_r$ by angle α.

2 for *vertex* $v \in abc$ *of* $\mathtt{t}_{in}$ **do**

3   Compute the closest intersection points v_1, v_2 between $\mathtt{t}_{in}$ and $\mathtt{t}_r$ on the sides connected to vertex v.

4   Compute area a_v of the triangle $v_1 v_2 v$, i.e., the area inside $\mathtt{t}_{in}$ and outside $\mathtt{t}_r$ at vertex v.

5   Compute h_v such that $h_v \cdot (|v_1 v| + |v_2 v|) = a_v$.

6 for *side* $S \in ABC$ *of* $\mathtt{t}_{in}$ **do**

7   Shift the level set describing S by $h_S = \max_{v \in abc \setminus s}\{h_v\}$ perpendicular to S outwards.

8 Compute triangle $\mathtt{t}_{ext}$ described by the shifted level sets of $\mathtt{t}_{in}$.

9 Compute a bounding circle $\mathtt{c}_b$ of $\mathtt{t}_{in}$.

10 Select all elements that have some overlap with $(\mathtt{t}_{ext} \cup \mathtt{c}_b) \setminus \mathtt{t}_r$.

Algorithm 7.1: Procedure to select extension elements of a rotating triangle.

of mass, the boundary condition on the fluid-solid interface $\mathcal{I}$ is given by $\boldsymbol{u} = \omega \cdot \begin{pmatrix} -(\boldsymbol{x}_2 - c_{\mathcal{S},2}) \\ \boldsymbol{x}_1 - c_{\mathcal{S},1} \end{pmatrix}$ for a given scalar angular velocity ω.

The fluid parameters are $\nu = 0.001$ and $\rho_{\mathcal{F}} = 1$. We then consider the maximal inflow speed $U = 1.5$, resulting in a mean inflow speed of $u_m = 1$. The angular velocity of the rotating triangle is prescribed as 1. As the reference speed, we take the mean inflow relative to the maximal horizontal component of the rigid body's velocity. This is given by $U = 1 + 1 \cdot \sqrt[2]{3} \cdot \sqrt{3}/2 \cdot 0.1 \approx 1.057$. As a result the resulting flow has a Reynolds number of $Re \approx 106$. We then look at the dimensionless drag, lift and torque coefficients as defined in (3.44).

Element Marking Strategy   To apply our unfitted Eulerian finite element scheme from chapter 4 to the above problem, we need to define the extension region $\mathcal{S}^+$ to realise the necessary extension. However, as we have seen in subsection 4.5.2, this needs to be done carefully so that the extension ghost-penalties do not disturb the solution, where the extension is not needed.

To select the extension and support elements as sparingly as possible, we use our knowledge of the angle of rotation $\alpha = \omega \Delta t$ in a given time-step Δt. By determining the area outside the current domain but inside the domain after a rotation of the triangle by α, we can select the elements that contribute to these sections as the extension elements. We then select approximately the same number of elements inside the physical domain as support elements. The details of this selection procedure can be found in Algorithm 7.1.

In Figure 7.3, we illustrate the area used to select elements on which we apply the extension ghost-penalties, i.e., lines $3 - 5$ in Algorithm 7.1. The striped area at each vertex is the area a_v (lines $3 - 4$). At each of these vertices, h_v is computed such that the combined hatched area of the two rectangles at each vertex matches the area of the striped triangle at that vertex (line 5). Then at each side, we take the maximal distance h_v from each of the two vertices and shift the side's level sets by this distance away from the centre (line 8). These shifted level sets then describe the triangle $\mathtt{t}_{ext}$ (line 10).
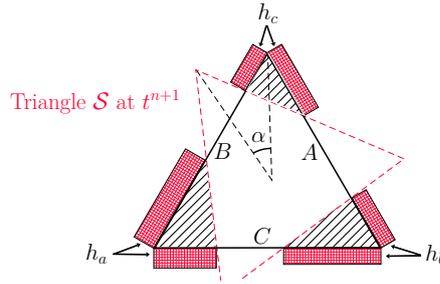
Figure 7.3: Illustration to Algorithm 7.1 for marking extension elements.

Set-Up   We consider a mesh constructed with a local mesh parameter $h_{\max}$ in the right three-quarters of the channel and $h = h_{\max}/5$ on the left quarter for our numerical computations. On the resulting meshes, we consider standard `TH`$_2$ elements. We represent the triangle using three separate level set functions, each describing one side of the triangle. We note that since each of the triangle sides is already a linear function, the resulting cut cell integration is exact for the geometry.

For our unfitted Eulerian time-stepping scheme, we take the Nitsche parameter $\sigma = 40k^2$ and the extension and stability ghost-penalty parameters are given by $\gamma_{gp,e} = 0.01$ and $\gamma_{gp,s} = 0.1$, respectively.

As initial "warm-up" data for the time-dependent problem, we take a stationary Stokes solution on the initial domain with the corresponding boundary conditions.

Results

BDF1   We use the BDF1 discretisation of the time-derivative and consider the problem over the interval $[0, 5]$. In Figure 7.4, we see the resulting drag lift and torque coefficients over the time interval $[4, 5]$, i.e., one full rotation of the equilateral triangle, for a series of joint mesh and time-step refinements. Additionally, we have plotted the splines approximating the curves to try and remove the observed instabilities; see also subsection 4.5.2 where we first applied this approach. The spline is established by placing a knot at every fifteen time-steps and then solving the least-squares problem to minimise the difference between the spline and the data using the `python` library `scipy.interpolate`. Looking at these results, we see the periodic nature of the flow, given by the constant rotation of the obstacle and the symmetry of the triangular obstacle. This can also be observed in the snapshot of the solution in Figure 7.5. Here we also note that compared to the *flow around a rotating cylinder* benchmark at $Re = 100$, see for example [Joh16, Fig. D.9], there is almost no wake behind the obstacle here, before the vortices form.

In the force coefficients in Figure 7.4, we see that the coarsest computation is under-resolved. Especially the curve of the lift and torque coefficients have a different form compared to curves from the other two computations. Furthermore, while the raw data from each of the three plotted periods is different, the smaller features in the spline curves appear in every period. This suggests that these smaller features are features of the solution.

To investigate whether the lack of accuracy on the previously coarsest computation is due to the mesh or time-step, we consider the mesh with $h_{\max} = 0.08$ over the same series of time-steps and the fixed time-step $\Delta t = 1/250$ over the same series of meshes. The resulting force coefficients over the period $[4, 5]$ can be seen in Figure 7.6 and Figure 7.7, respectively. Comparing these results, we see that while the smaller time-step do realise smaller features in the spline approximation, the mesh size seems to be the dominating factor in determining the overall accuracy of the resulting force coefficients.

BDF2   Unlike in the case of a flow around a cylinder benchmark from [ST96], the BDF1/implicit Euler discretisation is sufficient here to realise the vortex shedding. Nevertheless, we also consider the BDF2
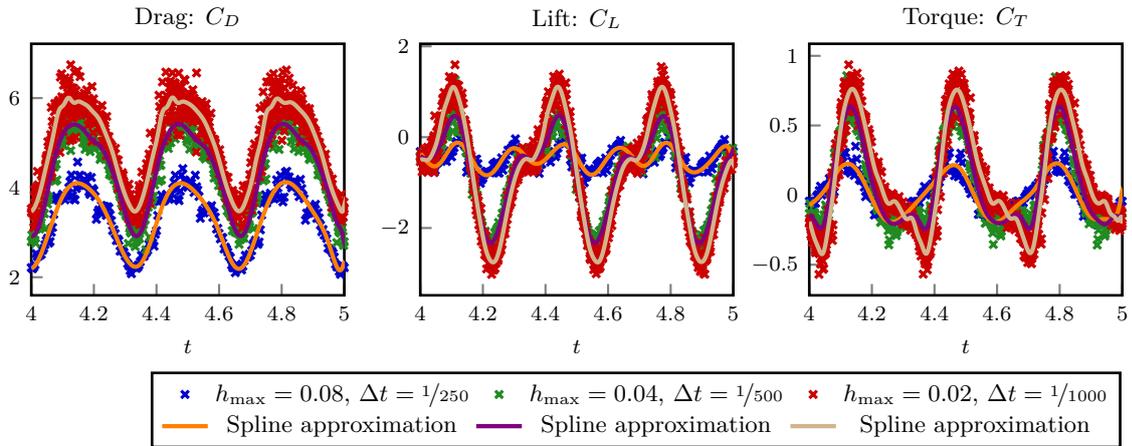
Figure 7.4: Force functionals on an equilateral triangle, rotating with constant speed. The spline approximations are defined with a knot at every 15th time-step. Computed using TH₂ elements and the Eulerian BDF1 time-discretisation.
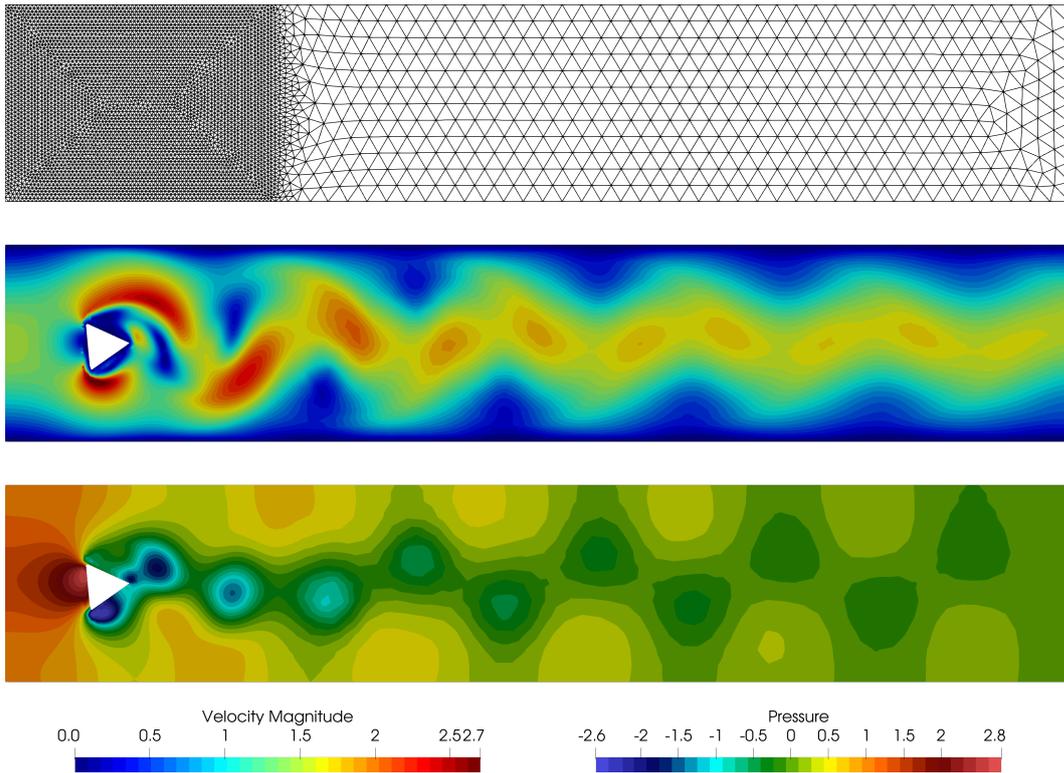
Figure 7.5: Unfitted mesh, velocity- and pressure-solution of an equilateral triangular obstacle rotating with constant speed in a channel flow. Computed on a mesh with $h_{max} = 0.04$ using the unfitted Eulerian BDF1 time-stepping method with the time-step $\Delta t = 1/500$. (A video of the velocity solution is available at https://youtu.be/Pg1aENR7XzY.)
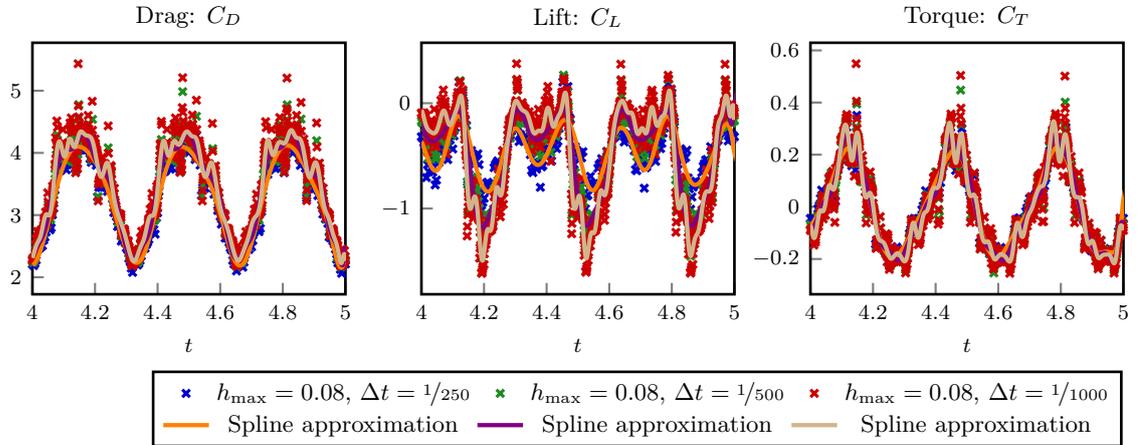
Figure 7.6: Force functionals on an equilateral triangle, rotating with constant speed. The spline approximations are defined with a knot at every 15th time-step. Computed using TH$_2$ elements on a fixed mesh with and the Eulerian BDF1 time-discretisation.
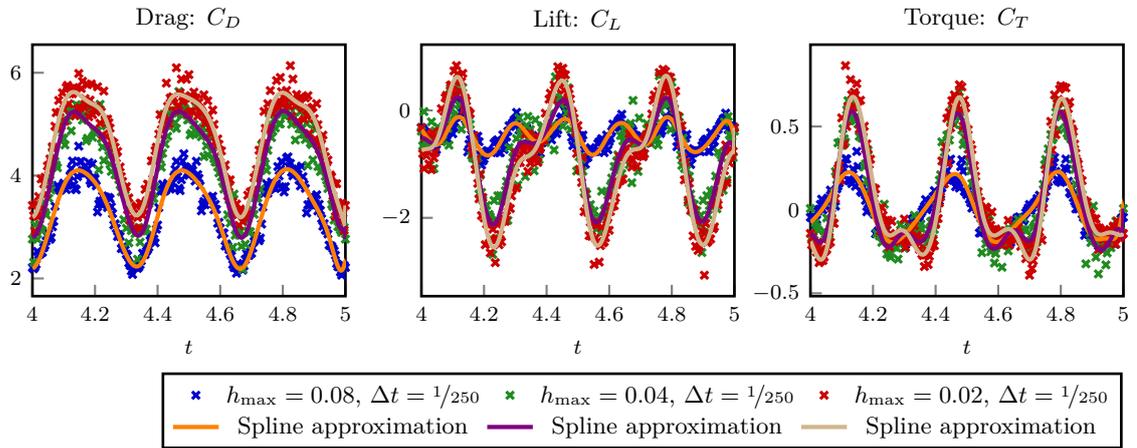
Figure 7.7: Force functionals on an equilateral triangle, rotating with constant speed. The spline approximations are defined with a knot at every 15th time-step. Computed using TH$_2$ elements on a series of meshes and the Eulerian BDF1 time-discretisation with a fixed time-step.
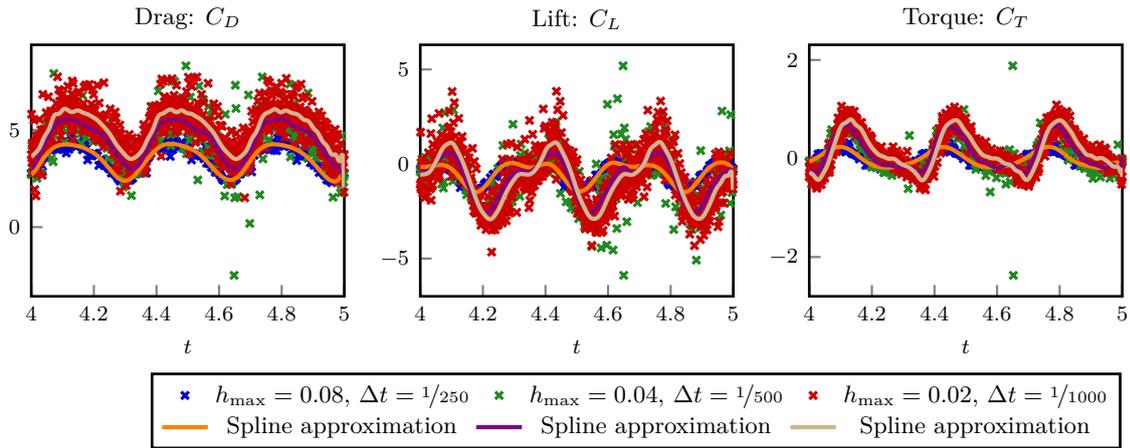
Figure 7.8: Force functionals on an equilateral triangle, rotating with constant speed. The spline approximations are defined with a knot at every 15th time-step. Computed using TH$_2$ elements and the Eulerian BDF2 time-discretisation.
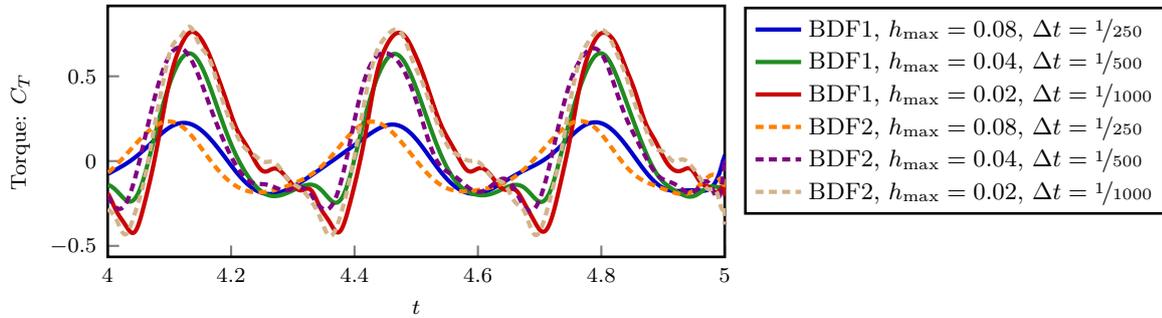
Figure 7.9: Spline approximation of the torque functional resulting from the unfitted Eulerian BDF1 and BDF2 time-stepping schemes.

time-discretisation to study the flow around a rotating, equilateral triangular obstacle.

We consider the same combined mesh and time-step refinement study as above. The results can be seen in Figure 7.8. Compared to the BDF1 case, we see larger instabilities in the data, which attribute to the larger necessary extension region. However, the resulting splines appear to be very similar to the BDF1 case. In Figure 7.9, we can see the splines for the torque coefficient resulting from both the BDF1 and BDF2 discretisations. Here we see that the main difference between the BDF1 and BDF2 results is a small phase shift. This again shows that the main challenge here is the spatial resolution.

7.1.3 Example 2: Free Rotation

As a second example, we consider a triangular rigid body that is free to rotate around its centre of mass. Therefore, the angular velocity of rigid body is the solution of

$$I_{\mathcal{S}} \partial_t \omega = \boldsymbol{T},$$

where $I_{\mathcal{S}} = \rho_{\mathcal{S}} \int_{\mathcal{S}} \|\boldsymbol{r}\|_2^2 \, \mathrm{d}\boldsymbol{x}$ is the solid's moment of inertia, and $\boldsymbol{T}$ is the scalar torque exerted by the fluid onto the solid, c.f. subsection 2.1.2. For the solid $\mathcal{S}$, we consider an isosceles triangle with side-length 0.16 and 0.08 with the centre of mass as $c_{\mathcal{S}} = (0.2, 0.205)$ and density $\rho_{\mathcal{S}} = 100$. The background domain and fluid set-up are identical to subsection 7.1.2. A sketch of this configuration can be seen in Figure 7.10.
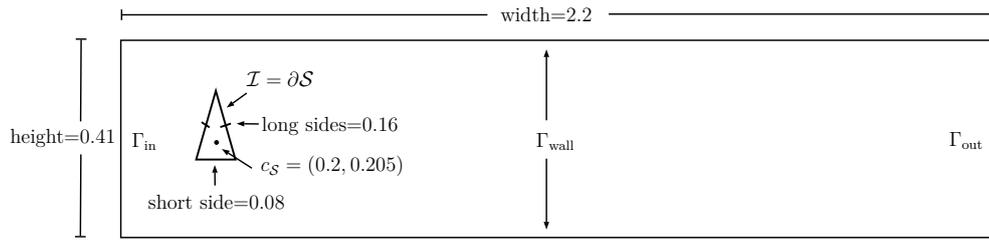
Figure 7.10: Initial configuration for a freely rotating isosceles triangle shaped rigid body in a channel.

Taking the blocked vertical section of the channel as the reference length, we then have that the resulting flow has the Reynolds number $Re = 80$–160. Consequently, we expect vortex shedding and a periodic solution state.

To solve the resulting coupled fluid-solid problem, we use the partitioned solver described in section 6.3 with a BDF1 discretisation of the time-derivatives. As we have seen in the above example, this is sufficient to realise the expected vortex shedding. The discretisation parameters are the same as in the previous example in subsection 7.1.2, while we go into the details of the mesh construction below.

Results

Mesh Construction   As we have seen in the previous case of an equilateral triangle, the spatial resolution is more important for the accuracy of the resulting flow than the temporal resolution. Note that we made a similar observation in the case of the moving cylinder in subsection 4.5.2 above. Furthermore, since the fluid-solid interface is not smooth, we cannot expect high-order regularity of the solution and, therefore, only low order convergence locally. As a result, we need to consider a finer mesh in the region of the moving interface compared to the bulk of the domain. Consequently, we investigate how we can construct an appropriate mesh for this coupled fluid-solid problem.

To this end, we consider the following meshing strategy for the background domain. In the right three-quarters of the domain, we prescribe the meshing parameter as $h_{\max}$ and in the front quarter as $h_{\max}/\delta_{\mathrm{front}}$. Additionally, we iteratively bisect any elements potentially containing the solid n_{ref} times. As a result, the mesh has three local mesh sizes $h_{\max}$, $h_f = h_{\max} \cdot \delta_{\mathrm{front}}$ and $h_{tr} = h_f \cdot 2^{-n_{\mathrm{ref}}}$. The time-step is chosen as $\Delta t = 1/250$.

In order to make the comparison between the different meshes fair, we choose the parameters such that the resulting active part of the mesh has approximately the same number of elements. as a reference we the solution from a mesh with $h_{\max} = 0.02$, $\delta_{\mathrm{front}} = 0.25$ and $n_{\mathrm{ref}} = 2$. Therefore, the resulting mesh is more resolved in every region of the mesh compared with the meshes we will be considering. As a measure of the accuracy of the solution compared with our reference, we integrate the absolute difference of the total rotation of the triangular rigid body over the time interval $[0, 6]$.

The results for a range of different meshes can be seen in Table 7.1. Looking at the integrated error in the rotation over time, we see that the most important factor is h_{tr}. Since the mesh with with $\delta_{\mathrm{front}} = 1$ and $n_{\mathrm{ref}} = 3$ gave the best results, we will use this mesh construction for the remaining computations.

Convergence   We consider the problem with mesh and time-step refinement. Due to our observation that larger time-steps are sufficient, we consider larger time-steps compared with the computations in subsection 7.1.2. The dimensionless force coefficients, the angular velocity and the total rotation relative to the initial state can be seen in Figure 7.11. Furthermore, a sample solution of the resulting fluid system in the periodic state can be seen in Figure 7.12.

In Figure 7.11, we first see that the mesh with $h_{\max} = 0.08$ is again too coarse, although the mesh around the triangle here is comparable with the resolution around the equilateral triangle with $h_{\max} = 0.04$. For

Parameters			Mesh				Result
$h_{\max}$	δ_{front}	n_{ref}	$h_{\max}$	h_f	h_{tr}	# Els.	Err. Rotation
0.08	1	4	0.08	0.08	0.005	5470	$5.60 \cdot 10^{-1}$
0.08	0.5	3	0.08	0.04	0.005	5390	$6.05 \cdot 10^{-1}$
0.08	0.25	2	0.08	0.02	0.005	5910	$3.72 \cdot 10^{-1}$
0.044	1	3	0.044	0.044	0.0055	5450	$3.36 \cdot 10^{-1}$
0.044	0.5	2	0.044	0.022	0.0055	5490	$7.28 \cdot 10^{-1}$
0.052	0.25	1	0.052	0.013	0.0065	5820	$7.40 \cdot 10^{-1}$
0.0265	1	2	0.0265	0.0265	0.006625	5650	$6.29 \cdot 10^{-1}$
0.0295	0.5	1	0.0295	0.01475	0.007375	5930	$7.41 \cdot 10^{-1}$

Table 7.1: Rotation error for different meshes with approximately 5000 elements.
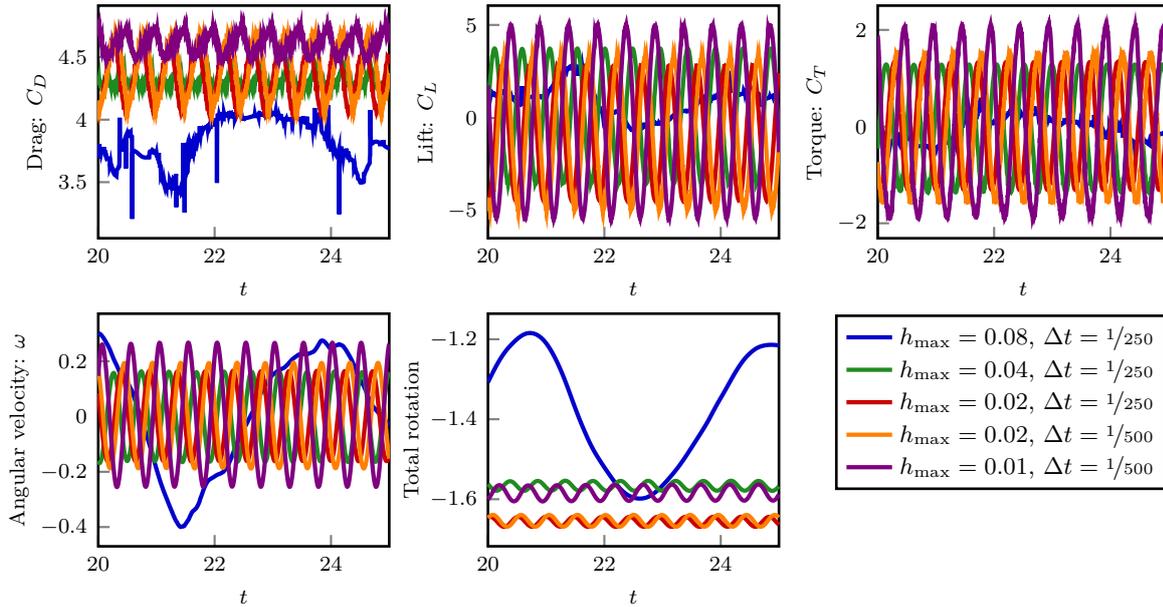
Figure 7.11: Force coefficients angular velocity and total rotation of a freely rotating isosceles triangle in a channel flow in the periodic state. Computed using the Eulerian time-stepping with a BDF1 time-discretisation and three levels of mesh refinements in a bounding circle of the triangle.

the remaining computations, we see that the periodic state has been reached at $t = 20$, and the frequency of the functional values appears to be the same. However, the mean of the total rotation in the periodic state is not entirely clear. As a result, we also observe some difference in the amplitude and mean of the drag and lift coefficients.

7.1.4 Summary

We draw the following conclusions from the above computations. First, we note that non-smooth domains are in general unproblematic for the Eulerian time-stepping scheme and that even the BDF1 method allows for the appearance of vortex shedding. This is in contrast to the BDF1 method applied to flow problems on smooth and fixed domains such as the Schäfer-Turek 2d-3 problem, where it is known that the BDF1 method is too diffusive and therefore leads to very inaccurate results [JMR06].

Concerning the force coefficient functionals, we saw the same instabilities as for smooth geometries for a prescribed motion of the solid domain, c.f. subsection 4.5.2. Approximating this data by a spline allowed
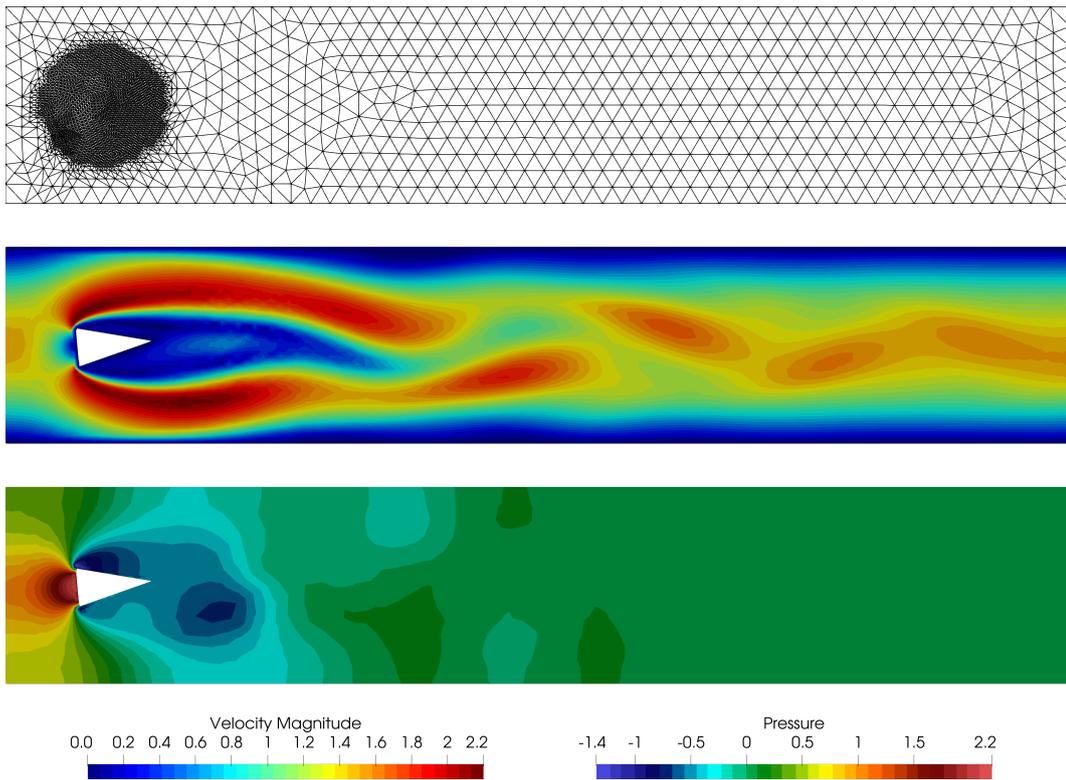
Figure 7.12: Unfitted mesh, velocity- and pressure-solution at $t = 20$ of the freely rotating isosceles triangle shaped rigid body in a channel flow. Computed on a mesh with $h_{\max} = 0.04$ and three mesh refinements in a bounding circle of the rigid body using the unfitted Eulerian BDF1 time-stepping with $\Delta t = 1/250$. (A video of the velocity solution is available at https://youtu.be/ST8H1EYMF7w.)
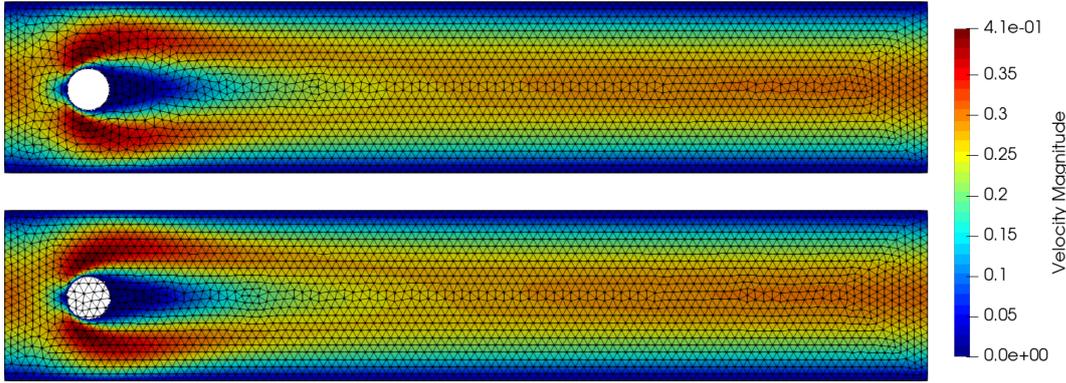
Figure 7.13: Velocity solution to the ST-2D1 problem with $h_{max} = 0.02$. Top: Fitted FEM. Bottom: CutFEM.

us to realise a periodic curve, as expected for these functionals. With our partitioned fluid-solid solver, these instabilities were significantly smaller for the case of the solid motion being driven by the fluid forces.

Finally, we observed that the most significant discretisation parameter for accurate results is the local mesh parameter in the area where the moving solid body is located. Even when using a BDF1 time-discretisation, relatively large time-steps were sufficient, and the use of the BDF2 method did not improve the results. In fact, the necessity of a larger extension region for the BDF2 method generally leads to larger instabilities in the functional values.

7.2 Learning Forces Using Deep Neural Networks for Under-Resolved Computations

In subsection 7.1.3, we have seen that very fine meshes are necessary in the vicinity of a triangular rigid body to obtain accurate motion of the particle. This was possible there since we considered a single very large rigid body, and the motion was restricted to a small region of the fluid domain. However, in the following, we want to consider small particles moving freely in the fluid domain. As a result, a fully resolved mesh is unfeasible. Consequently, we need to consider alternative methods to obtain accurate values for the forces acting on a rigid particle in an under-resolved simulation. To this end, we will train an artificial neural network to predict the forces acting on a particle, which takes fluid features as input, which are available accurately in an under-resolved setting. This section follows our work in [WR21].

Example 7.1 (Motivation of approach). To illustrate that we can compute volumetric flow features accurately, while the boundary forces are not computed accurately in an under-resolved CutFEM simulation, we consider the *flow around a cylinder* benchmark [ST96] and take the stationary "2D-1" test case. We compute this once using a fitted approach together with the Babuška-Miller trick [BM84] to evaluate the drag and lift functional, and once using the isoparametric CutFEM approach with the direct evaluation of the boundary integral to realise the force values. For both the fitted and unfitted simulations, we use TH$_3$ elements. To make the comparison as fair as possible, we consider unstructured meshes with the same mesh size in each computation.

In Figure 7.13, we can see the velocity solution to these computations together with the computational meshes. Note that it is nearly impossible to distinguish the two solutions visually. In Table 7.2, we see the benchmark quantities resulting from the two computations. Here we immediately see that while the values from the fitted computations are reasonably for such coarse mesh, the values resulting from the CutFEM computations even result in the wrong sign for the lift coefficient.

Since the solutions in the bulk of the domains look indistinguishable, it is natural to ask whether there

Method	Benchmark quantities					
	C_D	err.	C_L	err.	Δp	err.
Fitted TH$_3$	5.579521	0.00025%	0.0105942	0.232%	0.117162	0.30%
CutFEM TH$_3$	5.571512	0.14379%	-0.0063662	160.0%	0.119766	1.91%
Ref.	5.579535		0.0106189		0.117520	

Table 7.2: Benchmark quantities for the ST-2D1 problem computed using fitted FEM and CutFEM on meshes with $h_{max} = 0.02$. Reference values obtained from `www.featflow.de` (visited on 15.02.2021).

Method	Volumetric Quantities			
	Avg. $\boldsymbol{u}_1$	err.	Avg. $\boldsymbol{u}_2$	err.
Fitted TH$_3$	$4.293\,27 \cdot 10^{-3}$	0.0013%	$1.492\,00 \cdot 10^{-5}$	0.031%
CutFEM TH$_3$	$4.287\,96 \cdot 10^{-3}$	1.2506%	$1.493\,17 \cdot 10^{-5}$	0.109%
Ref.	$4.293\,33 \cdot 10^{-3}$		$1.491\,55 \cdot 10^{-5}$	

Table 7.3: Benchmark quantities for the ST-2D1 problem computed using fitted FEM and CutFEM on meshes with $h_{max} = 0.02$. Reference values computed using fitted TH$_6$ elements on a mesh with $h_{max} = 0.005$.

are other quantities based on the solution near the obstacle, which we can compute accurately in both settings. To this end, we compute the average velocity in a strip of width 0.05 around the obstacle. The results from this are presented in Table 7.3. Here we see that while the fitted FEM solution resulted in values closer to the reference values compared to CutFEM, the difference between the accuracy of the fitted and unfitted computations are significantly smaller. Furthermore, we see that we keep multiple significant figures of accuracy in the functional value, even at values of order 10^{-5}.

We conclude from these simulations that while it is difficult to obtain accurate forces from the boundary integral formulation in an under-resolved CutFEM computation, other solution features can be obtained much more accurately, even on such coarse meshes. Therefore, if we can construct a mapping from flow features near the obstacle to the forces acting on the obstacle, we should be able to get more accurate force values in the under resolved CutFEM setting.   ▲

Material parameters

We choose our fluid and solid material parameters to have a setting that can be given some physical meaning while remaining in the regime of small to moderate Reynolds numbers.

We choose the fluid and solid parameters to approximate coarse sand in a glycerol/water mixture. We take a mixture of 1 part water to 4 parts glycerol at a temperature of 21°C. The resulting relevant material parameters are summarised in Table 7.4. The fluid parameters are obtained through an online calculator tool[1] and the density of the solid is taken as the density of quartz[2]. The ISO standard 14688-1:2017[3] defines coarse sand to have a particle size of $0.63 - 2.0$ mm. As a prototype for a sand particle, we shall consider an equilateral triangle to keep the parametrisation simple. We then take the side length of our triangles to be $d_{\mathcal{S}} = 2.00 \cdot 10^{-3}$ m.

A Priori Computations   In order to establish the regime of the resulting flow when the triangular particle is left in free fall, we consider a single triangular particle in free-fall with the material parameters as summarised in Table 7.4. We restrict the motion of the particle to free vertical motion and use an ALE

---

[1] `www.met.reading.ac.uk/ sws04cdw/viscosity_calc.html`, visited on 25.09.2020
[2] `www.matweb.com/search/datasheet _print.aspx?matguid=8715a9d3d1a149babe853b465c79f73e`, visited on 25.09.2020
[3] `www.sis.se/api/document/preview/80000191`, visited on 25.09.2020

Parameter	$\mu_{\mathcal{F}}$	$\rho_{\mathcal{F}}$	$\nu_{\mathcal{F}}$	$\rho_{\mathcal{S}}$
Value	$8.5679 \cdot 10^{-2}\,\mathrm{N\,s\,m^{-2}}$	$1.2167 \cdot 10^{3}\,\mathrm{kg\,m^{-3}}$	$7.0419 \cdot 10^{-5}\,\mathrm{m^{2}\,s^{-1}}$	$2.65 \cdot 10^{3}\,\mathrm{kg\,m^{-3}}$

Table 7.4: Fluid and solid material properties of a 1:4 water-glycerine mixture and quartz

discretisation, similar to the discretisation in section 4.5, using the partitioned fluid-solid approach to solve the coupled fluid-solid system accurately.

We choose the domain $\Omega = (0, 0.1) \times (0.0, 2)$ over the time interval $[0, 10]$ with the initial centroid of the triangular particle at $c_{\mathcal{S}} = (0.05, 0.15)$. Additionally, we consider a range of different angles of attack.

For the ALE computation, we use a mesh with $h_{\max} = 1.00 \cdot 10^{-3}$ in the bulk and $h = 2.50 \cdot 10^{-5}$ on the solid boundary together with inf-sup stable $\mathbb{P}^5/\mathbb{P}^3_{dc}$ elements. The time derivatives in the ALE formulation are then discretised using the BDF1 scheme and the time-step $\Delta t = 1/500$.

We observed in these computations that the maximal terminal velocity of the triangle was $v = 0.047\mathrm{m\,s^{-1}}$. Taking the side-length of the triangular particle as the reference length then leads to a Reynolds number of $Re = \frac{vL}{\nu_{\mathcal{F}}} \approx 1.33$. This is sufficiently large to justify considering the full Navier-Stokes equations, rather than the creeping flow Stokes equations, and small enough to ensure that the local flow around the triangle is laminar.

7.2.1 Neural Network

The idea to predict the forces acting on a small, moving particle originated from MINAKOWSKA, RICHTER and SAGER [MRS21]. Due to the similar aim of our neural network, we shall take [MRS21] as our starting point for the design of our network. We note that while in [MRS21] it was the aim to use a neural network to predict the forces based on the speed and shape of very small particles in a linear Stokes flow and without back-coupling of the particles onto the fluid, we want to predict the forces based on the fluid solution near our particle, which couples back to the fluid governed by the non-linear Navier-Stokes equations.

Network Design

Architecture   We shall consider a fully connected feed-forward network with at least three hidden layers and the ReLU activation function, i.e., $f(x) = \max\{0, x\}$. An example of such a network can be seen in Figure 7.14. We shall refer to networks by the number of neurones. For example, a network with three hidden layers consisting of l, m and n neurones is denoted by $l/m/n$. The number of neurones per layer and the number of layers we shall need for our network will be determined experimentally by inspecting the results achieved during training.

Input   A good choice for the network is vital since this must contain sufficient information for the network to learn the input to output map. To capture the force acting on the triangular rigid body, it makes sense that the features have to be in some sense local to the rigid body. Point evaluations of the velocity and pressure near the rigid body would be one choice. Unfortunately, while we found that this does indeed capture the necessary information needed by a neural network, the unresolved nature of the CutFEM discretisation means that we do not have a chance of obtaining sufficiently accurate values to feed to the network at run-time.

We have seen in Example 7.1 that an integral of the velocity components can be obtained accurately in a coarse CutFEM computation. For a rigid body $\mathcal{S}$, we define $\mathcal{O}(\mathcal{S}) := \{\boldsymbol{x} \in \mathcal{F} \mid \|\boldsymbol{x} - c_{\mathcal{S}}\|_2^2 < (d_{\mathcal{O}}/2)^2\}$ to be a circular fluid domain around the solid, centred at the solid's centre of mass with radius $d_{\mathcal{S}}$. As
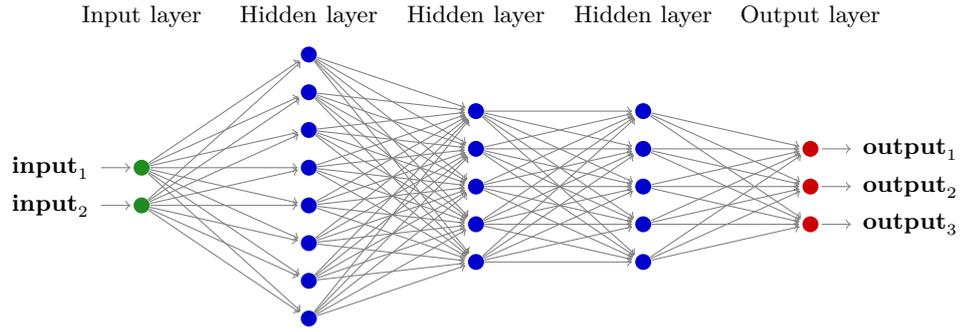
Figure 7.14: Illustration of a fully connected feed-forward networks with two inputs, three hidden layers consisting of 8, 5, and 5 neurones, respectively, and a three outputs.

the network input, we then take the mean fluid velocity, relative to the solid's velocity in $\mathcal{O}(\mathcal{S})$, i.e,

$$\mathbf{rel\ vel} \coloneqq \int_{\mathcal{O}(\mathcal{S})} \boldsymbol{u} - U \, \mathrm{d}\boldsymbol{x}.$$

We have found $d_{\mathcal{O}} = 4d_{\mathcal{S}}$ to be an appropriate choice. For the input, we then de-dimensionalise the data and take $\mathbf{input} = \mathbf{rel\ vel}/v$, where v is the characteristic velocity, which we take as the terminal settling velocity $v = 0.047\mathrm{m\,s}^{-1}$.

As the forces only depend on the angle of attack between the mean (integrated) flow relative to the triangle's velocity and orientation, we shall consider the input in a reference configuration. For this, we choose the bottom side of the triangle to be parallel to the x-axis. As a result, the network learns the forces resulting from any angle of attack in this reference orientation. To obtain the physical forces, we rotate the input into the reference configuration and rotate the drag/lift predictions back into the physical orientation. Since the torque in two spatial dimensions is a scalar quantity, it is invariant with respect to rotation.

Output We train the network to learn the dimensionless drag, lift and torque coefficients C_D, C_L and C_T, respectively, to keep the network general. See (3.44) for the definition of these coefficients. The reference speed is taken as the terminal velocity established below as $u_m = v = 0.047\mathrm{m\,s}^{-1}$ and the reference length $L = d_{\mathcal{S}} = 0.002\mathrm{m}$.

Training Data

In order to train a neural network of which we can expect sufficient accuracy for our application, we need to generate an appropriate training data set. To this end, we generate training data based on idealised configurations which are very close to the final application, computed using highly resolved discretisations.

Set-Up We consider two idealised situations for the training data. The first setting consists of purely translational motion, thereby allowing a simple and accurate ALE discretisation. To generate learning data for translational motion, we take the domain $(0, 0.5)^2$ with the equilateral triangular obstacle located at $(0.25, 0.25)$ in the reference configuration. This rigid body then moved from $(0.1, 0.25)$ to $(0.4, 0.25)$ and back again over a time interval $[0, t_{end}]$. To get a wide range of relative velocities between the triangle and the mean flow around the triangle, we accelerate the particle at different rates, i.e., consider different values for t_{end}. The physical location of the body is then given by

$$c_{\mathcal{S}}(t) = (0.25 - 0.15 \cos\left(\frac{\pi t}{2t_{end}}\right), 0.25).$$

Case(Method)	$h_{\max}$	h_{loc}	Δt	$C_{D,\max}$	$C_{L,\min}$	$C_{T,\max}$
Translational (ALE)	0.02	$2 \cdot 10^{-4}$	$^1/_{250}$	$2.6094 \cdot 10^2$	$-5.9548 \cdot 10^1$	$2.9674 \cdot 10^1$
	0.01	$1 \cdot 10^{-4}$	$^1/_{500}$	$2.6105 \cdot 10^2$	$-5.9673 \cdot 10^1$	$2.9699 \cdot 10^1$
	0.005	$5 \cdot 10^{-5}$	$^1/_{1000}$	$2.6111 \cdot 10^2$	$-5.9751 \cdot 10^1$	$2.9694 \cdot 10^1$
Rotational: (CutFEM)	0.0084	–	$^1/_{250}$	$1.6025 \cdot 10^{-2}$	$-1.3880 \cdot 10^{-2}$	$3.3697 \cdot 10^{-1}$
	0.0042	–	$^1/_{500}$	$1.3443 \cdot 10^{-2}$	$-1.5876 \cdot 10^{-2}$	$3.5969 \cdot 10^{-1}$
	0.0021	–	$^1/_{1000}$	$1.0735 \cdot 10^{-2}$	$-8.9824 \cdot 10^{-3}$	$3.5636 \cdot 10^{-1}$

Table 7.5: Validation results of the discretisation to generate the training data set.

for $t \in [0, t_{end}]$. To implement this motion, we use a prescribed ALE mapping, as above. To generate the data with different angles of attack, we rotate the rigid body by an angle α around its centre of mass and rotate the resulting relative velocities and forces back into the reference configuration. As a result, we can reuse the same ALE mapping to simulate different directions of relative motion of the solid body.

Since the above ALE computations only include translational motion, this is equivalent to only considering a parallel flow around a fixed obstacle. For the network to be universally usable, we also need to include rotational flow data. This corresponds to rotating the triangle. To this end, we consider a triangular rigid body located at the centre of the domain $\Omega = (0, 0.1)^2$. This is then rotated at different speeds, clockwise and anti-clockwise, such that the total rotation at time t with respect to the initial configuration is given by

$$\alpha(t) = \sin(2\pi \cdot t/t_{end}).$$

This set-up is not suitable for a simple ALE discretisation as relatively small rotations will lead to mesh-entanglement. Consequently, we use a highly resolved moving domain CutFEM simulation to generate data with rotational input.

Validation To ensure that the generated learning data is computed sufficiently accurate, we consider the above cases over a series of meshes and time-steps. For this test case, we take $t_{end} = 2.0$ for both set-ups and $\alpha = 0$ for the translational set-up.

For the ALE discretisation, we take TH$_4$ elements on a mesh with diameter $h_{\max}$ in the bulk and a local mesh parameter h_{loc} on the boundary of the rigid body. In time we discretise using the BDF1 scheme with the time-step Δt.

For the CutFEM discretisation, we take TH$_2$ elements on a mesh with global meshing parameter $h_{\max}$, three levels of mesh bisections in the domain where we compute the average relative velocity and an additional five levels of mesh bisections in the bounding circle of the rotating triangle.

The results for the convergence study can be seen in Table 7.5. For the ALE computations, we see that the discretisation is accurate and that the second mesh already provides $2 - 3$ significant figures of accuracy in the target data. Since the neural network prediction will introduce an additional approximation error, we consider this to be sufficiently accurate for the training data. For the rotational CutFEM data, we see that the forces are (in absolute value) significantly smaller than the translational data. However, we also see that the second finest discretisation should be accurate enough for the training of the network.

Data Generation To generate the data set, we consider $t_{end} \in \{2, 2.5, 3, 4, 6, 8, 1\}$, and rotate the triangle with angle $\alpha \in \{ \frac{2i\pi}{3 \cdot 40} \mid i = 0, \dots, 39 \}$. Since the triangle is equilateral, the remaining angles of attack $\alpha \in [2\pi/3, 2\pi)$ can be obtained by post-processing the data appropriately. Based on the above validation computations, we consider the mesh with $h_{\max} = 0.01$, and take the time-step $\Delta t = ^1/_{500}$. The resulting data set then contains $2.13 \cdot 10^6$ input/output pairs.

Architecture	Unknowns	Target	Training data		Validation data	
			$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$	$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$
30/20/10	953	C_D	$1.66 \cdot 10^0$	$1.73 \cdot 10^1$	$1.87 \cdot 10^0$	$1.19 \cdot 10^1$
		C_L	$1.64 \cdot 10^0$	$1.88 \cdot 10^1$	$1.77 \cdot 10^0$	$1.46 \cdot 10^1$
		C_T	$5.87 \cdot 10^{-1}$	$5.01 \cdot 10^0$	$6.57 \cdot 10^{-1}$	$4.29 \cdot 10^0$
50/20/20	1653	C_D	$1.05 \cdot 10^0$	$8.62 \cdot 10^0$	$1.19 \cdot 10^0$	$7.88 \cdot 10^0$
		C_L	$1.05 \cdot 10^0$	$8.92 \cdot 10^0$	$1.17 \cdot 10^0$	$8.12 \cdot 10^0$
		C_T	$5.56 \cdot 10^{-1}$	$4.16 \cdot 10^0$	$6.25 \cdot 10^{-1}$	$3.75 \cdot 10^0$
50/20/20/10	1833	C_D	$1.19 \cdot 10^0$	$1.06 \cdot 10^1$	$1.32 \cdot 10^0$	$7.92 \cdot 10^0$
		C_L	$1.16 \cdot 10^0$	$8.62 \cdot 10^0$	$1.32 \cdot 10^0$	$7.88 \cdot 10^0$
		C_T	$5.61 \cdot 10^{-1}$	$4.50 \cdot 10^0$	$6.32 \cdot 10^{-1}$	$3.76 \cdot 10^0$
100/50/20/20	6853	C_D	$9.58 \cdot 10^{-1}$	$8.75 \cdot 10^0$	$1.08 \cdot 10^0$	$8.02 \cdot 10^0$
		C_L	$9.68 \cdot 10^{-1}$	$8.60 \cdot 10^0$	$1.09 \cdot 10^0$	$7.88 \cdot 10^0$
		C_T	$5.51 \cdot 10^{-1}$	$4.09 \cdot 10^0$	$6.20 \cdot 10^{-1}$	$3.76 \cdot 10^0$
100/50/50/20	8983	C_D	$9.73 \cdot 10^{-1}$	$8.55 \cdot 10^0$	$1.10 \cdot 10^0$	$7.80 \cdot 10^0$
		C_L	$9.58 \cdot 10^{-1}$	$8.77 \cdot 10^0$	$1.07 \cdot 10^0$	$8.01 \cdot 10^0$
		C_T	$5.51 \cdot 10^{-1}$	$3.96 \cdot 10^0$	$6.19 \cdot 10^{-1}$	$3.58 \cdot 10^0$

Table 7.6: Prediction errors in a weighted ℓ^2 norm and the maximum norm on the training and validation data sets after 20000 epochs of training for a number of different network sizes. Each network predicts all three functional values simultaneously.

For the rotational data we choose $t_{end} \in \{0.5, 1, 2, 3, 5, 6\}$. Based on the above validation computations, we take the mesh with $h_{\max} = 0.0042$, and the time-step is chosen as $\Delta t = 1/500$. As a result, we then obtain an additional $8.79 \cdot 10^3$ input/output pairs.

Training

We implement the neural networks described above using `PyTorch` [Pas+19]. We use the mean squared error as the loss function and take the Adam algorithm [KB14; LH17] as the optimiser with a step size of 10^{-4}. The networks are trained for a total of 20000 epochs. For each network to predict all three values simultaneously, we scale both the input and output data to be in the interval $[-1, 1]$. In practice, we scale the predictions back appropriately so that the appropriate coefficients are obtained.

To make sure that we do not over fit the network to the training data set, we also generated a validation data set in the same fashion as the translational part of the training data set but with different angles of attack and values for t_{end}. This then consists of $4.32 \cdot 10^5$ data points. We evaluate the network on the validation data set during training to ensure that the training error does not decrease while the validation error increases.

The errors of the predictions made by the networks on the training and validation data sets can be seen in Table 7.6 for the case where a single network was used to predict drag lift and torque, while Table 7.7 shows the error for separate networks for each force. To make the results on data sets of different sizes comparable, we use the norm $\|\text{err}\|^2_{mean} := \frac{1}{N} \sum_{i=1}^{N} (\text{prediction}_i - \text{value}_i)^2$ and $\|\text{err}\|_\infty := \max_{i=1,\dots,N} (|\text{prediction}_i - \text{value}_i|)$.

To find an appropriate network size, which is large enough to capture all the information contained in the data set while also being small enough for fast evaluations in the final solver, we consider a number of different networks. The chosen number of layers and neurones per layer can be seen in the first column of Table 7.6.

The results in Table 7.6 show that the results are broadly similar for all six considered network architectures. Looking at the three-layer networks, we observe that the prediction error resulting from the

Target	Training data		Validation data	
	$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$	$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$
C_D	$9.59 \cdot 10^{-1}$	$8.83 \cdot 10^0$	$1.08 \cdot 10^0$	$8.10 \cdot 10^0$
C_L	$9.96 \cdot 10^{-1}$	$8.67 \cdot 10^0$	$1.12 \cdot 10^0$	$7.87 \cdot 10^0$
C_T	$5.54 \cdot 10^{-1}$	$4.14 \cdot 10^0$	$6.23 \cdot 10^{-1}$	$3.63 \cdot 10^0$

Table 7.7: Prediction errors in a weighted ℓ^2 norm and the maximum norm on the training and validation data sets after 20000 epochs of training. Separate networks of the architecture 50/20/20 with 1611 unknowns are used to predict the three functional values.
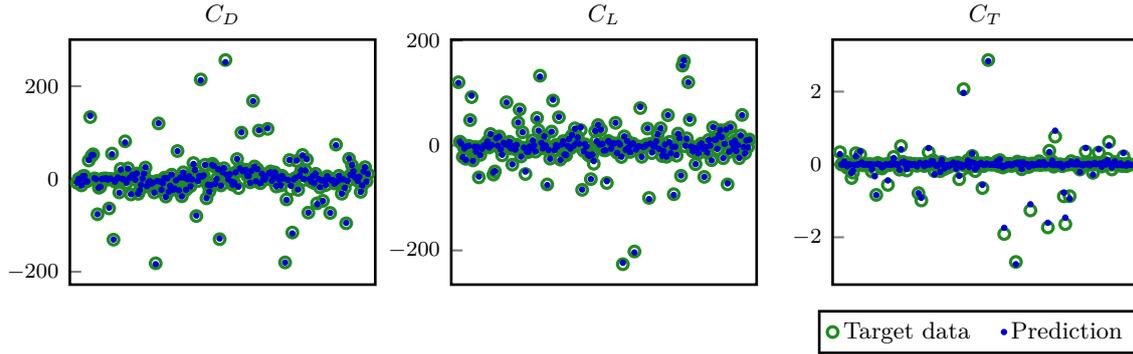
Figure 7.15: Target data and network prediction for 300 random points in the training data set. Network architecture: 50/20/20, 1653 unknowns.

30/20/10 network are almost two times larger than that of the 50/20/20 network. Looking at the four-layer networks, we see that the errors are generally the same as for the 50/20/20 network, with some small deviations in both directions.

In order to check whether we can gain more accuracy by considering separate networks for the three functionals, we train three separate 50/20/20 networks. The results thereof can be seen in Table 7.7, where we observe that the prediction errors are about the same as those realised by the single network with the same architecture in Table 7.6.

Both in Table 7.6 and Table 7.7, we see that the errors are generally similar and very large. To give these errors more meaning, we plot the predictions and the target values for a random selection of point from the validation data set in Figure 7.15. Here we see that the predicted values generally match the target values well. This indicates that the size of the errors in Table 7.6 and Table 7.7 are due to the size of the target values. The errors in Table 7.6 indicate that overall the force dynamics are captured with about 1 – 2 significant figures of accuracy.

As a result of the above considerations, we choose the single 50/20/20 network to predict the drag, lift and torque from the average velocity around the rigid particle in our application. Considering this network as a function $\mathbb{R}^2 \to \mathbb{R}^3$, we can then plot the individual components as a function of the input in a three-dimensional plot. This can be seen in Figure 7.16. This illustrates that while the drag and lift coefficients are represented by a relatively simple function, the torque coefficient functional is significantly more involved.

Remark 7.2. The input for the network requires two integrals over a relatively small area and a rotation of these two values into the reference configuration. Furthermore, because the network is relatively small, the additional computation effort introduced to the solver by predicting the forces, rather than evaluating them directly, is negligible compared to the effort required to solve the non-linear system resulting from the FEM discretisation of the Navier-Stokes equations. ▲
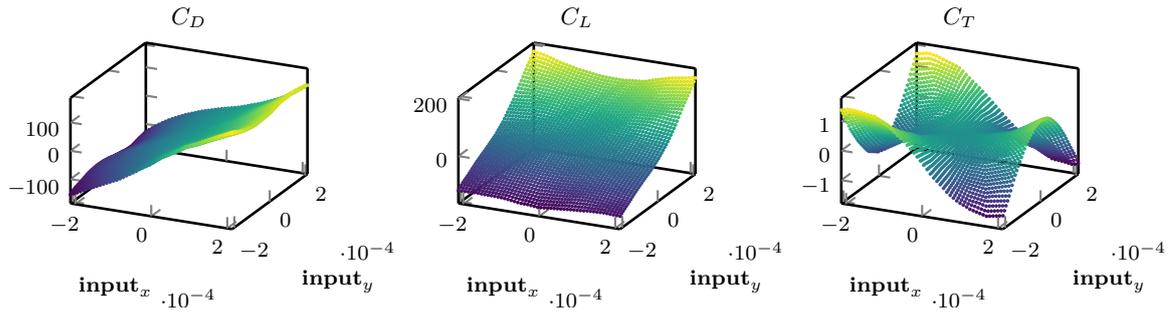
Figure 7.16: Network predictions from the 50/20/20 network as functions of input variables.

Remark 7.3 (Training times).   The training of the above networks was performed on a Tesla V100 PCIe 16GB graphics card with `PyTorch` using `CUDA` version 10.1. Due to the small size and simple structure of the networks, the training times for $2 \cdot 10^4$ epochs ranged between 277 seconds for the 30/20/10 network and 936 seconds for the 100/50/20/20 network.   ▲

Validation

During training, we have compared the predictions against the ground truth in the sense of the ALE training data. However, the network aims to predict the forces acting on a particle in a CutFEM simulation and be more accurate than evaluating the boundary integral.

To validate that the neural network predictions are more accurate than the direct evaluation of the forces from the boundary integral, we run a moving domain CutFEM simulation of the set-up, with which we generated the training data for $t_{end} = 3$ and $\alpha \in \{0, \pi/6, \pi/3, \pi/2, 2\pi/3\}$. Here we take a background mesh with $h = 10^{-3}$ in the area where we compute the average of the velocity around the triangle and $h_{max} = 0.04$ in the remaining part of the domain. The mesh in the averaging area is therefore a factor of two smaller than the size of the rigid body. On this mesh, we work with both unfitted TH$_2$ and TH$_3$ elements. The errors of the force prediction and evaluation are then evaluated by comparing the values against the direct evaluation of a highly resolved ALE computation of the identical set-up. The spatial discretisation is identical to the training data generation above. In both cases, the time-step $\Delta t = 1/300$ was chosen.

The resulting prediction errors of the forces in the CutFEM simulation can be seen in Table 7.8. Here we see that the mean and maximal predictions are at least one order of magnitude smaller than the direct evaluation. In Figure 7.17, we plot the resulting forces for a single run of the above computations ($\alpha = 0$). This shows that while there is a visible difference between the prediction and the "ground truth", the predictions approximate the real forces significantly more accurately than the direct computation via the boundary-integral evaluation. We also note that there was no significant improvement in the direct evaluation when using higher-order elements. We conclude that while the predictions are not perfect, we have improved on the direct computation of the forces on an under-resolved computational mesh. However, we also note that we cannot expect any asymptotic mesh convergence here, as the prediction error will begin to dominate once the interface is sufficiently resolved in every time-step.

7.2.2 Numerical Examples

We now consider several numerical examples which use the neural network constructed in subsection 7.2.1 in the context of the fluid-rigid body system (2.5) with small triangular rigid bodies and the system discretised with an under resolved moving-domain CutFEM discretisation.

Force comp.	Discr.	C_D		C_L		C_T	
		$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$	$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$	$\|\cdot\|_{mean}$	$\|\cdot\|_\infty$
Evaluation	TH$_2$	$7.86 \cdot 10^0$	$8.41 \cdot 10^1$	$1.33 \cdot 10^0$	$3.32 \cdot 10^1$	$1.44 \cdot 10^2$	$2.29 \cdot 10^3$
Prediction	TH$_2$	$1.54 \cdot 10^0$	$2.32 \cdot 10^1$	$2.26 \cdot 10^{-1}$	$4.81 \cdot 10^0$	$1.48 \cdot 10^{-1}$	$2.92 \cdot 10^0$
Evaluation	TH$_3$	$6.59 \cdot 10^0$	$7.17 \cdot 10^1$	$1.47 \cdot 10^0$	$4.15 \cdot 10^1$	$2.41 \cdot 10^2$	$3.92 \cdot 10^3$
Prediction	TH$_3$	$1.52 \cdot 10^0$	$2.28 \cdot 10^1$	$2.37 \cdot 10^{-1}$	$5.05 \cdot 10^0$	$1.48 \cdot 10^{-1}$	$2.90 \cdot 10^0$

Table 7.8: Absolute errors in the forces resulting from the direct evaluation of the boundary integrals and the prediction made by the neural network in a CutFEM computation.
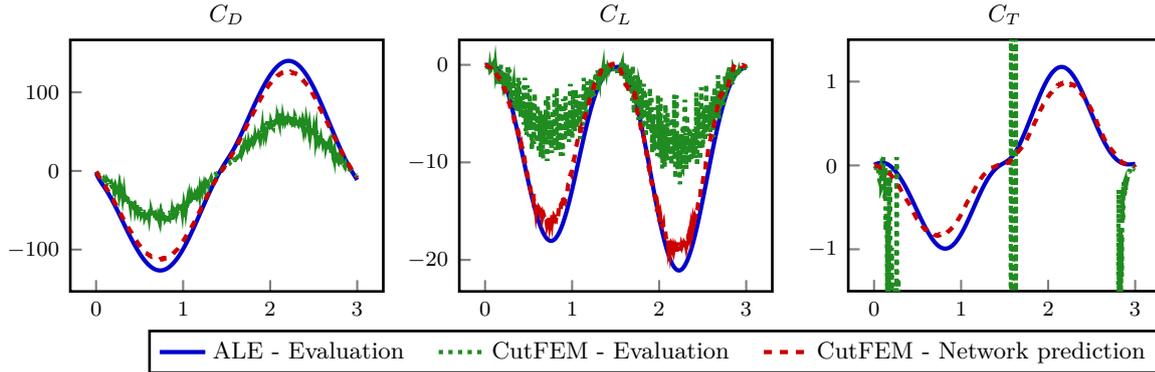
Figure 7.17: Prediction and evaluation in a CutFEM simulation with TH$_2$ elements compared against the values evaluated in a fitted ALE computation.

Restricted Motion with ALE Comparison

In subsection 7.2.1, we validated that the neural network is able to predict the forces acting on a particle with prescribed motion more accurately than the direct evaluation thereof. To validate the method in our target setting of free-fall, we shall consider the settling of a single particle restricted to vertical motion. We consider this simplification to compare the results against a resolved ALE simulation.

To this end, we take the domain $\Omega = (0, 0.1)^2$, assert no-slip boundary conditions at the left and right walls $\Gamma_{\text{left}} \cup \Gamma_{\text{right}}$, and the "do-nothing" at the top and bottom boundaries $\Gamma_{\text{top}} \cup \Gamma_{\text{bottom}}$. This approximates the section of a vertical pipe. The rigid body is an equilateral triangle with side-length $d_\mathcal{S} = 2 \cdot 10^{-3}$, and the fluid and solid material parameters are as in Table 7.4. At $t = 0$, the centre of mass of the rigid body is at $(0.05, 0.08)$, and we rotate the body by an angle α with respect to reference configuration, in which the bottom of the triangle is parallel to the x-axis. We shall consider $\alpha = 0, \pi/12, \pi/6$. An illustration of this configuration can be seen in the left of Figure 7.18.

For the ALE reference computation, we consider TH$_5$ elements on a mesh with $h_{\max} = 5 \cdot 10^{-3}$, $h = 1.25 \cdot 10^{-3}$ in a horizontal strip of height $8 \cdot 10^{-3}$ around the rigid body and $h = 4 \cdot 10^{-4}$ on the interface of the rigid body. Based on our validation experiments above, this discretisation is sufficiently accurate to serve as a reference here.

We run one simulation with the solid motion based on the forces boundary integral evaluation and one with the forces predicted by our neural network to validate the neural network approach. For both CutFEM computations, we consider a mesh with $h_{\max} = 10^{-3}$ on which we consider TH$_2$ elements. For the unfitted discretisation used, we take the Nitsche parameter to enforce the Dirichlet boundary conditions on the level set interfaces as $\sigma = 100k^2$ while the stability and extension ghost-penalty parameters are $\gamma_{gp,s} = 0.1$ and $\gamma_{gp,e} = 0.001$, respectively.

We take the time-step $\Delta t = 1/250$ and compute until $t = 1.0$. For the partitioned fluid-solid solver, we take the initial relaxation parameter as $\omega = 0.5$, allow a maximum of 10 sub-iterations per time-step
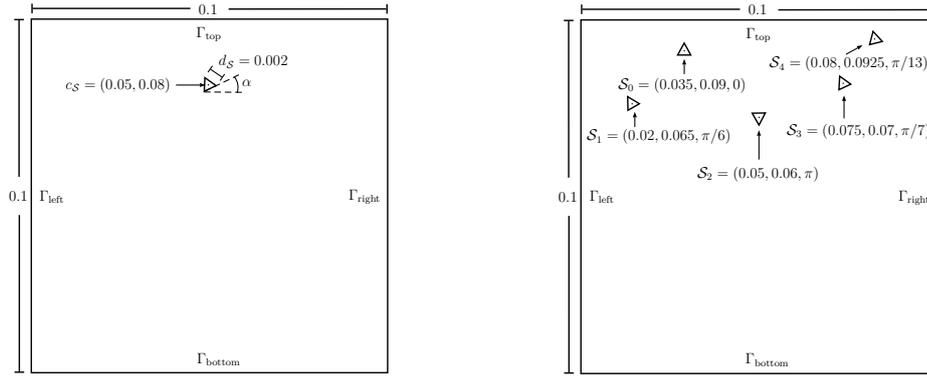
Figure 7.18: Domain sketches for the initial configurations of the computational examples. Left: A single particle. Right: Five particles.

and a tolerance of 1% in the relative update. The forces acting from the fluid onto the rigid body are evaluated using the single 50/20/20 network for drag, lift and torque.

In Figure 7.19, we can see the vertical force, the vertical speed and the vertical position of the centre of mass of the rigid body over time for each of the three triangle orientations. Here we see that for all three orientations, the drag obtained by the network prediction is significantly more accurate than the direct evaluation of the boundary integral. Looking at the body's vertical speed and position, we see a visible difference between the ALE reference solution and the movement resulting from the predicted forces. The uniformly faster speed in the CutFEM prediction computations is because, at low speeds of the particle, the network underestimated the drag, as can be seen in the left column of Figure 7.19. However, we also see that the predictions are significantly more accurate than the result from the direct evaluation of the forces. This clearly shows that the neural network approach is able to realise accurate results on very coarse and under-resolved meshes, where the unfitted approach with the standard evaluation of the resulting forces does not lead to reasonable motion.

Free Motion of a Single Particle

We now consider the full fluid-rigid body system, including translational movement in all directions as well as rotational motion. As a result, we do not have an ALE reference to compare the results with, so that we only compare qualitatively against the results from the previous example with restricted motion. The initial configuration is therefore chosen as in the previous example with restricted motion. The initial rotation is chosen as $\alpha = \pi/3$.

We again take the background mesh with $h_{\max} = 10^{-3}$ and consider a second mesh with $h_{\max} = 7.5 \cdot 10^{-4}$ to investigate the mesh dependence. On each mesh, we take TH$_2$ elements. The remaining discretisation parameters are chosen as in the unfitted simulation in the previous example.

The fluid solution at time $t = 1.0$ on the coarser of the two meshes is visualised in Figure 7.20. In Figure 7.21, we see the resulting velocity components that make up the movement of the rigid body for the two considered meshes.

Looking at velocity components in Figure 7.21, we immediately see that there is very little difference between the results from the two considered meshes. While this does not indicate the method's accuracy, this does show that the method is stable. As is to be expected, the vertical velocity component dominates the translational motion. Furthermore, we see that while a terminal velocity is not fully reached, that the acceleration between $t = 0.5$ and $t = 1.5$ is small, and the velocity during this time is very similar and only about 10% faster compared to the a priori computations to establish the terminal settling velocity above and in our validation example with restricted motion above.
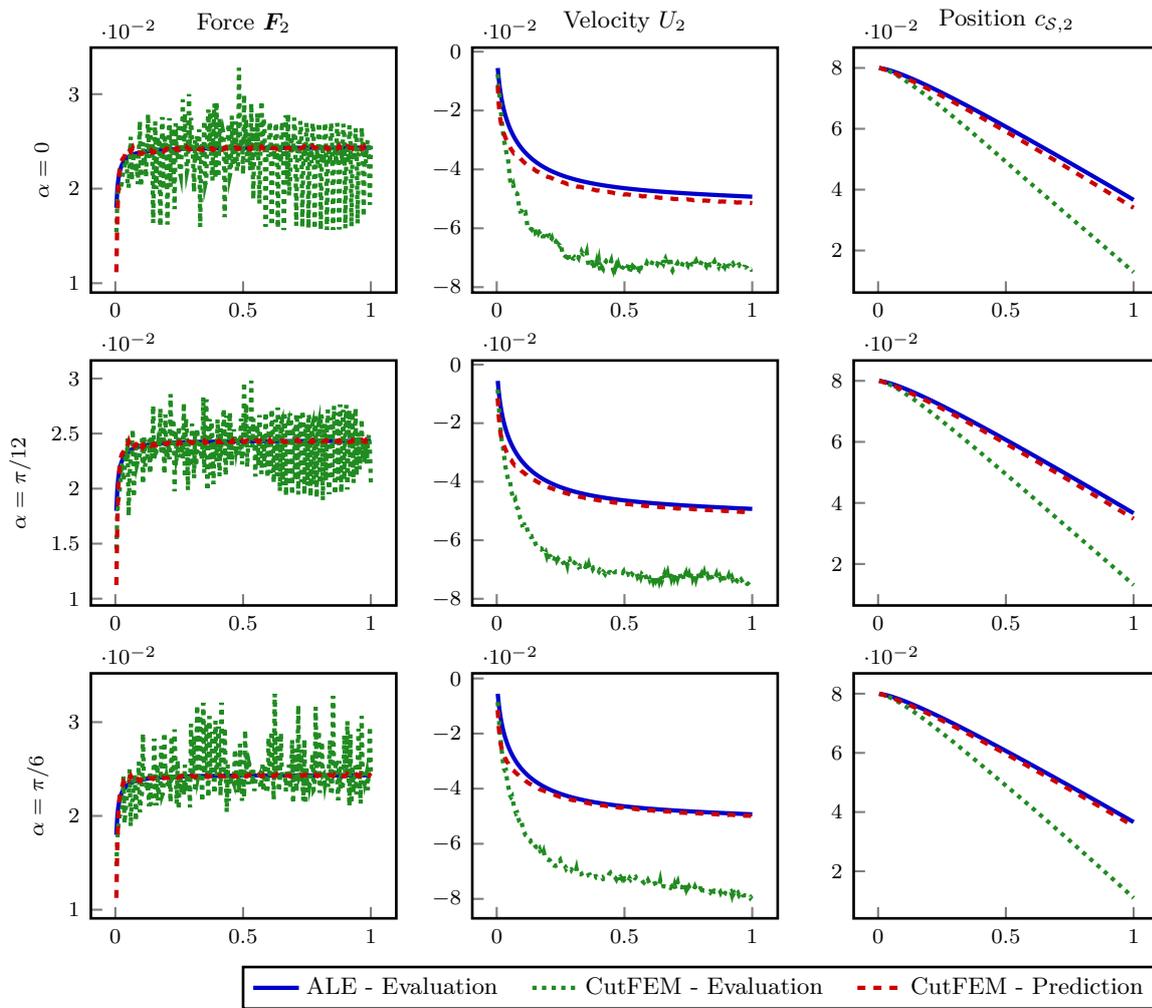
Figure 7.19: Vertical force, velocity and position of a single falling triangular rigid body restricted to vertical motion. Comparison between an ALE reference computation, the force computation in an under-resolved CutFEM computation and the network prediction in an under-resolved CutFEM computation.
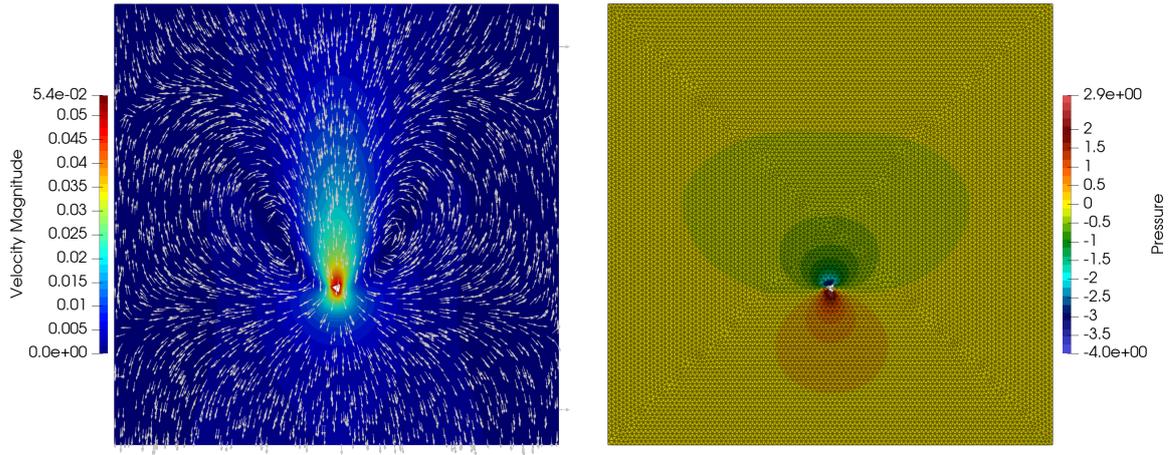
Figure 7.20: Velocity field (left) and pressure with mesh (right) at $t = 1.0$ resulting from a single triangular rigid body in free fall on a mesh with $h_{\max} = 10^{-3}$ with the forces governing the solid motion obtained from a neural network. (A video of the simulation is available at `https://youtu.be/BiWKHN4fzt4`.)
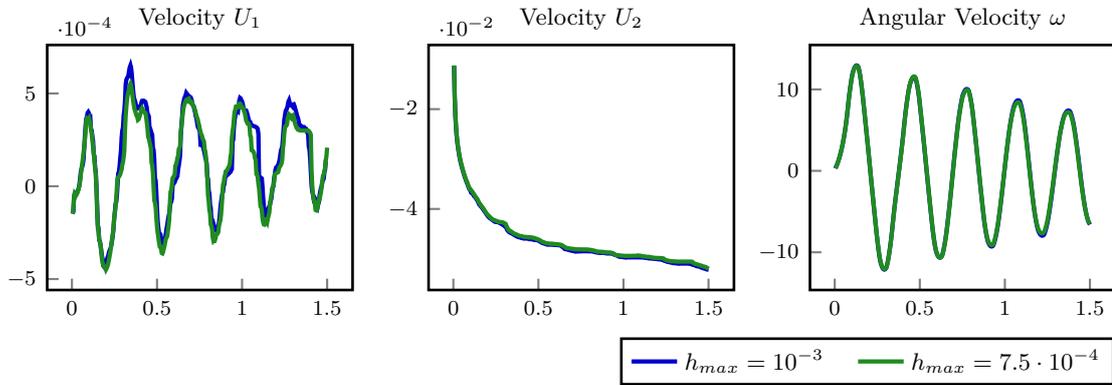
Figure 7.21: Translational and rotational velocity components of a single triangular rigid body in free fall on an under-resolved mesh with the forces from the predictions by a deep neural network.

The angular velocity in Figure 7.21 appears very large at first sight. Factoring in that the side-length of the body is $2 \cdot 10^{-3}$, we find that the resulting maximal velocity at the corners of the triangular rigid bodies is of order 10^{-2}. So, in general, the velocity of the triangle resulting from rotation is smaller than the vertical velocity component.

Overall, this example shows that our method leads to plausible results on highly under-resolved meshes, on which the standard CutFEM approach cannot realise accurate forces and thereby cannot realise accurate solid motion. Finally, we emphasise that a comparison to a fitted ALE simulation is out of scope for this example. An ALE discretisation of this problem would have to include re-meshing procedures and projections of the solution onto the resulting new meshes to avoid mesh-entanglement resulting from the rotation of the particles.

Free Motion of a Multiple Particles

As a final and more advanced example, consider the same basic fluid set-up as before, but now take five triangular rigid bodies denoted by $\mathcal{S}_i$, $1 = 0, \ldots, 4$. Each particle's initial position and orientation is described by the centre of mass and the angle of rotation with respect to the reference configuration. In the reference configuration, the bottom side of the triangle is parallel to the x-axis. We then denote the initial
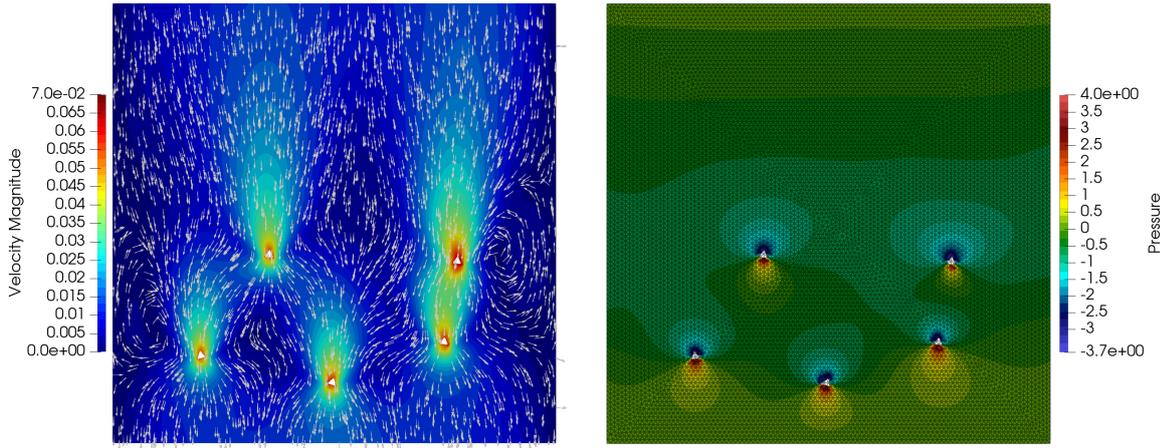
Figure 7.22: Velocity field (left) and pressure with mesh (right) at $t = 1.0$ resulting from five triangular rigid bodies in free fall on a mesh with $h_{max} = 10^{-3}$ with the forces governing the solid motion obtained from a neural network. (A video of the simulation is available at https://youtu.be/y-BTRvX9IIU.)

centre of mass and rotation by (c_x, c_y, α). We consider the initial states $(0.035, 0.09, 0)$, $(0.02, 0.065, \pi/6)$, $(0.05, 0.06, \pi)$, $(0.075, 0.07, \pi/7)$ and $(0.08, 0.0925, \pi/13)$ for $\mathcal{S}_0, \ldots, \mathcal{S}_4$, respectively. A sketch of this configuration can be seen in the right of Figure 7.18. The discretisation parameters of the moving domain CutFEM method remain as in the previous example.

The results from these computations are shown in Figure 7.22 and Figure 7.23. The fluid solution at $t = 1.0$ is shown in Figure 7.22, while Figure 7.23 shows the translational and rotational velocity components of the individual particles.

In Figure 7.23, we again see that there is no significant dependence on the mesh and that, in general, the translational velocity of all particles are larger than in the single-particle case above. Nevertheless, the order of magnitude is the same as before. These slightly faster translational speeds of the particles make sense, as each particle sets the fluid in motion, which then, in turn, helps to transport the other particles. Furthermore, we see both in Figure 7.23 and Figure 7.22 that $\mathcal{S}_4$ has the largest (vertical) velocity. This observation also makes sense, as this is directly in the wake of $\mathcal{S}_3$. In addition, we note that the angular velocity of $\mathcal{S}_0$ and $\mathcal{S}_4$, which are above the other particles, is smaller than that of the other particles. We attribute this to the fact that these two particles are in the parallel flow wake of the other three particles, resulting in a smaller torque acting on these two particles.

7.2.3 Summary

We conclude the following from the above hybrid finite element/neural network approach for small triangular rigid particles settling in an incompressible fluid. First, we saw that relatively small deep neural networks can capture the information contained in the average velocity components around the particle to predict the forces acting on the particle. Applying this network to an under-resolved CutFEM simulation of the idealised training data scenarios, the network could predict the forces significantly more accurately than the direct evaluation of the boundary integral defining these forces. The same could be observed in the case of a particle in free-fall, restricted to vertical motion. Applying the network to completely free settling of the triangular particles in an under-resolved CutFEM setting, we found that the results were stable with respect to the mesh size, and the settling behaviour was reasonable. Since we did not have a reference solution to compare the behaviours for the free particles with, it would remain very interesting to compare these results with an appropriate ALE simulation with re-meshing to include the rotational effects. Finally, we note that it remains open to see if other flow features can feed into a neural network to obtain more accurate predictions and extend the approach to higher Reynolds-number flows.
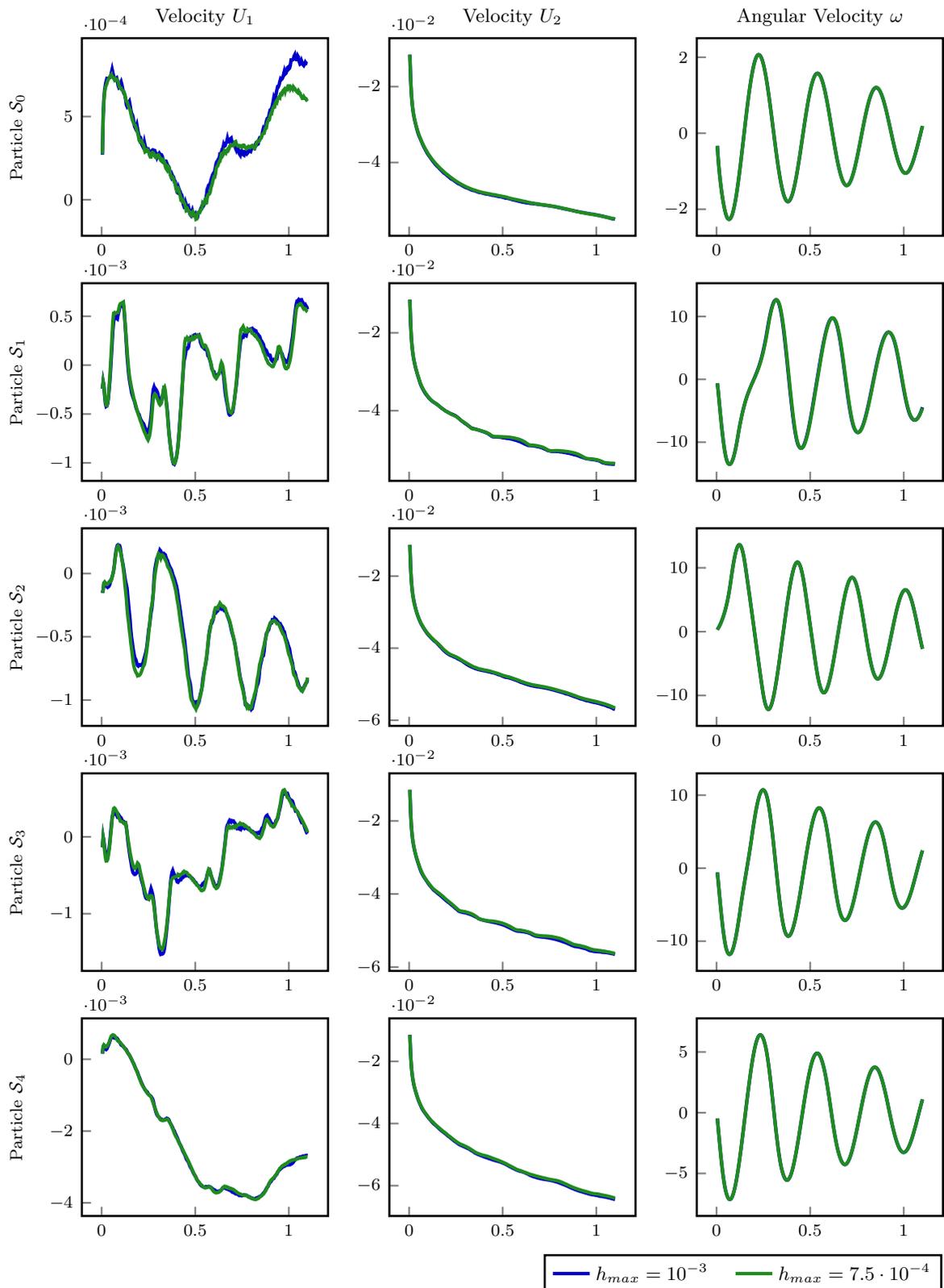
Figure 7.23: Translational and rotational velocity of five triangular rigid bodies in free fall in an under-resolved CutFEM simulation where the forces governing the solid motion are from the evaluation of a deep neural network.

Conclusions and Outlook

8.1 Conclusions

We have presented a new numerical method for solving fluid-rigid body interaction problems. This is based on a CutFEM discretisation of the fluid domain and an implicit extension, via ghost-penalty stabilisation, to enable an Eulerian time-stepping scheme for the fluid problem. We now summarise the most important achievements in this thesis.

Isoparametric CutFEM for Flow Problems

We extended the isoparametric unfitted finite element approach [Leh16; Leh17] to the Stokes equations using Taylor-Hood elements in section 3.2 and proved an optimal order error estimate in the energy norm. This estimate included the geometry approximation error. Optimal order convergence of the velocity in the $\mathcal{L}^2$-norm was also shown in practice. In section 3.3, we extended the method to the time-dependent Oseen equations. However, the resulting error estimate was not optimal in time due to the velocity-pressure coupling. In practice, we nevertheless observed optimal order convergence in both the mesh and time-step sizes. Finally, we solved a benchmark problem with this method, consisting of a freely rotating sphere in a channel flow, with similar accuracy as the available reference values. To achieve this, high-order elements and significant local mesh refinement near the fluid-solid interface was necessary, see subsection 3.4.3.

Eulerian Time-Stepping for Moving Domain Flow Problems

To work towards fluid-rigid body interaction problems, we extended the analysis of an Eulerian time-stepping scheme [LO19; BFM19] to the time-dependent Stokes problem on moving domains using Taylor-Hood elements in chapter 4. In the stability analysis, we were careful in the stabilisation scaling with respect to viscosity, and we included the geometry approximation error in our error analysis. In the error analysis, we saw that geometrical consistency error, introduced by the discrete approximation of the level set domain, leads to additional coupling between the velocity and pressure error which is non-standard. Similar to the case of CutFEM for the Oseen equations on a stationary domain, additional coupling appeared because the approximation of the time-derivative is not weakly divergence-free with respect to the pressure space at time t^n, which resulted in a suboptimal error estimate of $\|\frac{1}{\Delta t}(\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1})\|_{-1}$. While our error estimate was not optimal in the time-step, the method converged with the expected rates. Our numerical examples also illustrated the method's stability with respect to a wide range of viscosities and sizes of the extension strip, see subsection 4.4.1. Furthermore, we illustrated the potential of extending the method to higher-order both in space and time in subsection 4.4.4. When applying this method to the Navier-Stokes equations on moving domain in section 4.5, we observed that the inverse-viscosity scaling of the ghost-penalty stabilisation term realising the discrete extension, smears out the solution at the moving interface. This scaling has a significant negative impact on both the flow and the forces acting on the moving interface. Nevertheless, this scaling of the extension ghost-penalties was necessary to have a stable method for large time-steps where the interface crosses multiple elements. Finally, we illustrated that the method remains stable in the case of topology changes.

Eulerian Time-Stepping for Coupled Fluid-Rigid Body Problems

Working further towards the fully coupled fluid-solid interaction problem, we considered a version of
the Eulerian time-stepping scheme, using Lagrange multipliers to implement the boundary condition,
and applied this to the problem of the time-dependent Stokes equations coupled to translational rigid
body motion. With respect to stability, we saw in section 5.2 that the method has the same stability
bound as in the case of prescribed motion. We then developed an error estimate for this approach in
the temporally semi-discrete case for the heat equation in the bulk in section 5.3. Here we observed that
higher regularity of the solution and a more restrictive time-step restriction is necessary for the coupled
problem, compared to the case of prescribed motion. Applying the Nitsche version of the method to a
fluid-structure interaction problem in section 6.4, we found the method to be convergent in the pre-contact
dynamics. Here we also observed that with the model reduction of the solid from an elastic to a rigid
body, we were still able to simulate the pre-contact dynamics with good agreement with the experimental
data on which the set-up is based. With respect to the contact dynamics in the FSI problem, we saw
that while it is possible to realise a realistic rebounding effect with the very simple contact algorithm
considered here, this is very heavily parameter and material dependent, and there is no a priori method
to determine appropriate model parameters.

Non-Smooth Rigid-Body Particles

As a prototype for rigid body particles with non-smooth geometries, we considered triangular particles,
described using multiple level set functions, in chapter 7. Our method was able to simulate cases where
standard ALE approaches will fail. As before, local mesh refinement near the moving interface was
vital for accurate results. To work with small particles that are both too small to work on resolved
meshes and too large to assume no back coupling to the fluid, we developed a hybrid finite element/deep
neural network approach in section 7.2. We trained an artificial deep neural network based on data
from resolved simulations of idealised situations. The resulting neural network was then able to predict
the forces acting on triangle-shaped particles based on information available in highly under-resolved
situations. The neural network predictions were significantly more accurate than the direct evaluation of
the forces in cases where the fluid-solid interface was under-resolved. As a result, the particle motion was
comparable to that obtained from resolved meshes.

8.2 Outlook and Future Work

We discuss a number of open problems relating to the discretisation presented in this thesis, which we
consider interesting for future research.

Isoparametric CutFEM for Flow Problems

Concerning the isoparametric CutFEM method, it remains to prove optimal error estimates for the
time-dependent Stokes and Oseen problems. This is because in the unfitted case, the geometry error
and pressure stabilisation make it difficult to decouple the velocity and pressure, which in turn leads to
suboptimal results in the approximation error of the time-derivative, c.f. Remark 3.31. Furthermore, it is
known that pressure-robust and exactly divergence-free methods are of great benefit for high Reynolds-
number flow problems. However, CutFEM is in its current form cannot be pressure-robust due to the
necessary pressure stabilisation. Therefore, we consider it an interesting and relevant topic of future
research to develop unfitted finite element methods with improved mass conservation or which are even
fully pressure-robust.

Eulerian Time-Stepping for Moving Domain Flow Problems

Since we found that the Eulerian time-stepping scheme converges optimally with respect to the time-step, it remains an open topic for future research to prove optimal order and sharp error estimates for the Eulerian time-stepping scheme on moving domains. Since the extension of the approach to higher-order in both space and time works in practice, it remains to put this on a solid theoretical fundament, as done in the case for a scalar convection-diffusion problem on moving domains in [LL21]. Furthermore, we have seen in subsection 4.5.2, the conservation of mass is particularly poor here. Improving the conservation of mass is, therefore, also relevant for the moving domain approach.

As we have seen, the Eulerian time-stepping scheme derived for the time-dependent Stokes equations on moving domains requires $1/\nu$-scaled ghost-penalties for the discrete extension. However, this does not take into account the direction of motion of the domain, or if a transport term is included, the direction into which the extension must act. For the Navier-Stokes case, it would therefore be interesting to investigate other approaches which do not disturb the solution at the interface. For example, using unfitted space-time appears to be a promising approach [AB21].

Coupled Fluid-Rigid Body Problems

While our analysis of the Eulerian time-stepping scheme for the coupled problem of a body in free-fall in a fluid relied on a Lagrange multiplier formulation in the discretisation, our applications used Nitsche's method to impose the boundary condition on the moving interface. Therefore, it remains an open problem to extend the analysis to the Nitsche approach. Furthermore, the error analysis was only performed for the temporally semi-discrete case, so that is remains to go through the analysis in the fully discrete case.

As we have seen in section 6.4, the results from our rotationally symmetric formulation and rigid-body model for the solid are in good agreement with the data from the physical experiments. However, it remains to investigate the role of imperfections in the mass distribution within the solid and the roughness of the solid's surface and whether these factors can explain the deviation between the experimental and numerical results.

Several questions warrant future research for our hybrid finite element/neural network approach to simulate the interaction between rigid-body particles and the fluid on under-resolved meshes. For example, it would be very interesting to compare the results realised with our approach with those from an ALE simulation with re-meshing. Furthermore, generalising the approach to other particle shapes and sizes remains open. In this context, it would be interesting to see if a different choice of input data could generalise the approach to a single network to predict accurate forces for multiple material parameters. For example, alternative input data could include additional information such as the acceleration of the particle or the fluid could be taken into account. Also, the approach should be generalised and extended to include the interaction of closely neighbouring particles and particle-wall interaction. Finally, we note that all our unfitted simulations were performed in two spatial dimensions so that further effort should be put into extending the approach to full three-dimensional computations. Furthermore, as we only used direct solvers to solve the linear systems in our simulations, this includes work into efficient CutFEM preconditioners; see for example [Lud20] for work in this direction.

Bibliography

[Age+19]   C. Ager, B. Schott, A.-T. Vuong, A. Popp and W. A. Wall. 'A consistent approach for fluid-structure-contact interaction based on a porous flow model for rough surface contact'. In: *Internat. J. Numer. Methods Engrg.* 119.13 (June 2019), pp. 1345–1378. DOI: 10.1002/nme.6094.

[ASW20]   C. Ager, A. Seitz and W. A. Wall. 'A consistent and versatile computational approach for general fluid-structure-contact interaction problems'. In: *Internat. J. Numer. Methods Engrg.* (1st Oct. 2020). DOI: 10.1002/nme.6556.

[AB21]   M. Anselmann and M. Bause. *CutFEM and ghost stabilization techniques for higher order space-time discretizations of the Navier-Stokes equations*. 30th Mar. 2021. arXiv: 2103.16249 [math.NA].

[Ape99]   T. Apel. *Anisotropic finite elements: Local estimates and applications*. Advances in Numerical Mathematics. Stuttgart: Teubner, 1999. ISBN: 3-519-02744-5.

[ARW95]   U. M. Ascher, S. J. Ruuth and B. T. R. Wetton. 'Implicit-explicit methods for time-dependent partial differential equations'. In: *SIAM J. Numer. Anal.* 32.3 (June 1995), pp. 797–823. DOI: 10.1137/0732037.

[Baa01]   F. P. T. Baaijens. 'A fictitious domain/mortar element method for fluid-structure interaction'. In: *Internat. J. Numer. Methods Fluids* 35.7 (2001), pp. 743–761. DOI: 10.1002/1097-0363(20010415)35:7<743::aid-fld109>3.0.co;2-a.

[Bab73a]   I. Babuška. 'The finite element method with Lagrangian multipliers'. In: *Numer. Math.* 20.3 (June 1973), pp. 179–192. DOI: 10.1007/bf01436561.

[Bab73b]   I. Babuška. 'The finite element method with penalty'. In: *Math. Comp.* 27.122 (May 1973), pp. 221–221. DOI: 10.1090/s0025-5718-1973-0351118-5.

[BM84]   I. Babuška and A. Miller. 'The post-processing approach in the finite element method-part 1: Calculation of displacements, stresses and other higher derivatives of the displacements'. In: *Internat. J. Numer. Methods Engrg.* 20.6 (June 1984), pp. 1085–1109. DOI: 10.1002/nme.1620200610.

[BE86]   J. W. Barrett and C. M. Elliott. 'Finite element approximation of the Dirichlet problem using the boundary penalty method'. In: *Numer. Math.* 49.4 (July 1986), pp. 343–366. DOI: 10.1007/bf01389536.

[BH07]   Y. Bazilevs and T. J. R. Hughes. 'Weak imposition of Dirichlet boundary conditions in fluid mechanics'. In: *Comput. & Fluids* 36.1 (Jan. 2007), pp. 12–26. DOI: 10.1016/j.compfluid.2005.07.012.

[Beh01]   M. Behr. 'Stabilized space-time finite element formulations for free-surface flows'. In: *Comm. Numer. Methods Eng.* 17.11 (Oct. 2001), pp. 813–819. DOI: 10.1002/cnm.451.

[Beh08]   M. Behr. 'Simplex space-time meshes in finite element simulations'. In: *Internat. J. Numer. Methods Fluids* 57.9 (July 2008), pp. 1421–1434. DOI: 10.1002/fld.1796.

[Boc77]   D. N. Bock. 'On the Navier-Stokes equations in noncylindrical domains'. In: *J. Differ. Equ.* 25.2 (Aug. 1977), pp. 151–162. DOI: 10.1016/0022-0396(77)90197-8.

[BBF13]   D. Boffi, F. Brezzi and M. Fortin. *Mixed finite element methods and applications*. Berlin, Heidelberg: Springer, 2013. DOI: 10.1007/978-3-642-36519-5.

[BF13]   F. Boyer and P. Fabrie. *Mathematical tools for the study of the incompressible Navier-Stokes equations and related models*. New York: Springer, 2013. DOI: 10.1007/978-1-4614-5975-0.

[Bra+07]   M. Braack, E. Burman, V. John and G. Lube. 'Stabilized finite element methods for the generalized Oseen problem'. In: *Comput. Methods Appl. Mech. Engrg.* 196.4-6 (Jan. 2007), pp. 853–866. DOI: 10.1016/j.cma.2006.07.011.

[BR06]   M. Braack and T. Richter. 'Solutions of 3D Navier-Stokes benchmark problems with adaptive finite elements'. In: *Comput. & Fluids* 35.4 (May 2006), pp. 372–392. DOI: 10.1016/j.compfluid.2005.02.001.

[Bre61]   H. Brenner. 'The slow motion of a sphere through a viscous fluid towards a plane surface'. In: *Chem. Eng. Sci.* 16.3-4 (Dec. 1961), pp. 242–251. DOI: 10.1016/0009-2509(61)80035-3.

[BF91]   F. Brezzi and R. S. Falk. 'Stability of higher-order Hood-Taylor methods'. In: *SIAM J. Numer. Anal.* 28.3 (June 1991), pp. 581–590. DOI: 10.1137/0728032.

[Bur10]   E. Burman. 'Ghost penalty'. In: *C.R. Math.* 348.21-22 (Nov. 2010), pp. 1217–1220. DOI: 10.1016/j.crma.2010.10.006.

[Bur+14]   E. Burman, S. Claus, P. Hansbo, M. G. Larson and A. Massing. 'CutFEM: Discretizing geometry and partial differential equations'. In: *Internat. J. Numer. Methods Engrg.* 104.7 (Dec. 2014), pp. 472–501. DOI: 10.1002/nme.4823.

[BF14]   E. Burman and M. A. Fernández. 'An unfitted Nitsche method for incompressible fluid-structure interaction using overlapping meshes'. In: *Comput. Methods Appl. Mech. Engrg.* 279 (Sept. 2014), pp. 497–514. DOI: 10.1016/j.cma.2014.07.007.

[BFF20]   E. Burman, M. A. Fernández and S. Frei. 'A Nitsche-based formulation for fluid-structure interactions with contact'. In: *ESAIM Math. Model. Numer. Anal.* 54.2 (Feb. 2020), pp. 531–564. DOI: 10.1051/m2an/2019072.

[Bur+21] E. Burman, M. A. Fernández, S. Frei and F. M. Gerosa. *A mechanically consistent model for fluid-structure interactions with contact including seepage.* 20th Mar. 2021. arXiv: 2103.11243 [math.NA].

[BFM19] E. Burman, S. Frei and A. Massing. *Eulerian time-stepping schemes for the non-stationary Stokes equations on time-dependent domains.* 7th Oct. 2019. arXiv: 1910.03054v1 [math.NA].

[BH10] E. Burman and P. Hansbo. 'Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method'. In: *Comput. Methods Appl. Mech. Engrg.* 199.41-44 (Oct. 2010), pp. 2680–2686. DOI: 10.1016/j.cma.2010.05.011.

[BH12] E. Burman and P. Hansbo. 'Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method'. In: *Appl. Numer. Math.* 62.4 (Apr. 2012), pp. 328–341. DOI: 10.1016/j.apnum.2011.01.008.

[BH14] E. Burman and P. Hansbo. 'Fictitious domain methods using cut elements: III. A stabilized Nitsche method for Stokes' problem'. In: *ESAIM Math. Model. Numer. Anal.* 48.3 (Apr. 2014), pp. 859–874. DOI: 10.1051/m2an/2013123.

[BHL21] E. Burman, P. Hansbo and M. G. Larson. *CutFEM Based on Extended Finite Element Spaces.* 25th Jan. 2021. arXiv: 2101.10052 [math.NA].

[CMM08] G.-H. Cottet, E. Maitre and T. Milcent. 'Eulerian formulation and level set models for incompressible fluid-structure interaction'. In: *ESAIM Math. Model. Numer. Anal.* 42.3 (Apr. 2008), pp. 471–492. DOI: 10.1051/m2an:2008013.

[CF15] S. Court and M. Fournié. 'A fictitious domain finite element method for simulations of fluid-structure interactions: The Navier-Stokes equations coupled with a moving solid'. In: *J. Fluids Struct.* 55 (May 2015), pp. 398–408. DOI: 10.1016/j.jfluidstructs.2015.03.013.

[CFL13] S. Court, M. Fournié and A. Lozinski. 'A fictitious domain approach for the Stokes problem based on the extended finite element method'. In: *Internat. J. Numer. Methods Fluids* 74.2 (Aug. 2013), pp. 73–99. DOI: 10.1002/fld.3839.

[DSH86] R. H. Davis, J.-M. Serayssol and E. J. Hinch. 'The elastohydrodynamic collision of two spheres'. In: *J. Fluid Mech.* 163 (Feb. 1986), pp. 479–497. DOI: 10.1017/s0022112086002392.

[Deu21] P. Deuring. L^q-*weak solutions to the time-dependent Oseen system: decay estimates.* 12th Jan. 2021. hal: hal-02465651.

[DGH82] J. Donea, S. Giuliani and J. P. Halleux. 'An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions'. In: *Comput. Methods Appl. Mech. Engrg.* 33.1-3 (Sept. 1982), pp. 689–723. DOI: 10.1016/0045-7825(82)90128-1.

[Don+04] J. Donea, A. Huerta, J.-Ph. Ponthot and A. Rodríguez-Ferran. 'Arbitrary Lagrangian–Eulerian Methods. Fundamentals'. In: *Encyclopedia of Computational Mechanics.* Ed. by E. Stein, R. de Borst and T. J. R. Hughes. Vol. 1. John Wiley & Sons, Ltd, 15th Nov. 2004. Chap. 14. DOI: 10.1002/0470091355.ecm009.

[Dun06] T. Dunne. 'An Eulerian approach to fluid-structure interaction and goal-oriented mesh adaptation'. In: *Internat. J. Numer. Methods Fluids* 51.9-10 (2006), pp. 1017–1039. DOI: 10.1002/fld.1205.

[DL20] M. Duprez and A. Lozinski. 'ϕ-FEM: A Finite Element Method on Domains Defined by Level-Sets'. In: *SIAM J. Numer. Anal.* 58.2 (Jan. 2020), pp. 1008–1028. DOI: 10.1137/19m1248947.

[EG04] A. Ern and J.-L. Guermond. *Theory and practice of finite elements.* New York: Springer, 2004. DOI: 10.1007/978-1-4757-4355-5.

[EG21a] A. Ern and J.-L. Guermond. *Finite Elements I. Approximation and Interpolation.* Vol. 72. TAM. Cham: Springer, 2021. DOI: 10.1007/978-3-030-56341-7.

[EG21b] A. Ern and J.-L. Guermond. *Finite Elements II. Galerkin Approximation, Elliptic and Mixed PDEs.* Vol. 73. TAM. Cham: Springer, 2021. DOI: 10.1007/978-3-030-56923-5.

[EG21c] A. Ern and J.-L. Guermond. *Finite Elements III. First-Order and Time-Dependent PDEs.* Vol. 74. TAM. Cham: Springe, 2021. DOI: 10.1007/978-3-030-57348-5.

[Eva98] L. C. Evans. *Partial Differential Equations.* Vol. 19. Graduate Studies in Mathematics. Providence, RI: American Mathematical Society, 1998.

[Fei03a] E. Feireisl. 'On the motion of rigid bodies in a viscous incompressible fluid'. In: *Nonlinear Evolution Equations and Related Topics.* Basel: Birkhäuser, 2003, pp. 419–441. DOI: 10.1007/978-3-0348-7924-8_23.

[Fei03b] E. Feireisl. 'On the Motion of Rigid Bodies in a Viscous Compressible Fluid'. In: *Arch. Ration. Mech. Anal.* 167.4 (May 2003), pp. 281–308. DOI: 10.1007/s00205-002-0242-5.

[Fre16] S. Frei. 'Eulerian finite element methods for interface problems and fluid-structure interactions'. PhD thesis. Ruprecht-Karls-Universität Heidelberg, 23rd Aug. 2016. DOI: 10.11588/heidok.00021590.

[Fre17] S. Frei. 'Fluid-structure Interaction with Contact'. In: T. Richter. *Fluid-structure interactions.* Lecture Notes in Computational Science and Engineering. Cham: Springer, 2017. Chap. 12, pp. 399–416. DOI: 10.1007/978-3-319-63970-3_12.

[FR17a] S. Frei and T. Richter. 'An accurate Eulerian approach for fluid-structure interactions'. In: *Fluid-Structure Interaction.* Ed. by S. Frei, B. Holm, T. Richter, T. Wick and H.Yang. Berlin, Boston: De Gruyter, Nov. 2017. DOI: 10.1515/9783110494259-003.

[FR17b] S. Frei and T. Richter. 'A second order time-stepping scheme for parabolic interface problems with moving interfaces'. In: *ESAIM Math. Model. Numer. Anal.* 51.4 (July 2017), pp. 1539–1560. DOI: 10.1051/m2an/2016072.

[FRW16] S. Frei, T. Richter and T. Wick. 'Long-term simulation of large deformation, mechano-chemical fluid-structure interactions in ALE and fully Eulerian coordinates'. In: *J. Comput. Phys.* 321 (Sept. 2016), pp. 874–891. DOI: 10.1016/j.jcp.2016.06.015.

[FB10]   T.-P. FRIES and T. BELYTSCHKO. 'The extended/generalized finite element method: An overview of the method and its applications'. In: *Internat. J. Numer. Methods Engrg.* 84.3 (Aug. 2010), pp. 253–304. DOI: 10.1002/nme.2914.

[Gal00]   G. P. GALDI. 'An Introduction to the Navier-Stokes Initial-Boundary Value Problem'. In: *Fundamental Directions in Mathematical Fluid Mechanics*. Ed. by G. P. GALDI, J. G. HEYWOOD and R. RANNACHER. Basel: Birkhäuser, 2000, pp. 1–70. DOI: 10.1007/978-3-0348-8424-2_1.

[Gal02]   G. P. GALDI. 'On the Motion of a Rigid Body in a Viscous Liquid: A Mathematical Analysis with Applications'. In: *Handbook of Mathematical Fluid Dynamics*. Amsterdam: Elsevier, 2002, pp. 653–791. DOI: 10.1016/s1874-5792(02)80014-3.

[Gal11]   G. P. GALDI. *An Introduction to the Mathematical Theory of the Navier-Stokes Equations. Steady-State Problems*. New York: Springer, 2011. DOI: 10.1007/978-0-387-09620-9.

[GS06]   G. P. GALDI and A. L. SILVESTRE. 'Existence of time-periodic solutions to the Navier-Stokes equations around a moving body'. In: *Pacific J. Math.* 223.2 (Feb. 2006), pp. 251–267. DOI: 10.2140/pjm.2006.223.251.

[GGH12]   M. GEISSERT, K. GÖTZE and M. HIEBER. 'L^p-theory for strong solutions to fluid-rigid body interaction in Newtonian and generalized Newtonian fluids'. In: *Trans. Amer. Math. Soc.* 365.3 (Aug. 2012), pp. 1393–1439. DOI: 10.1090/s0002-9947-2012-05652-2.

[GH14]   D. GÉRARD-VARET and M. HILLAIRET. 'Existence of Weak Solutions Up to Collision for Viscous Fluid-Solid Systems with Slip'. In: *Commun. Pure Appl. Math.* 67.12 (May 2014), pp. 2022–2076. DOI: 10.1002/cpa.21523.

[GHC15]   D. GÉRARD-VARET, M. HILLAIRET and C.WANG. 'The influence of boundary conditions on the contact problem in a 3D Navier-Stokes flow'. In: *J. Math. Pures Appl. (9)* 103.1 (Jan. 2015), pp. 1–38. DOI: 10.1016/j.matpur.2014.03.005.

[GW08]   A. GERSTENBERGER and W. A. WALL. 'An eXtended Finite Element Method/Lagrange multiplier based approach for fluid-structure interaction'. In: *Comput. Methods Appl. Mech. Engrg.* 197.19-20 (Mar. 2008), pp. 1699–1714. DOI: 10.1016/j.cma.2007.07.002.

[Glo+99]   R. GLOWINSKI, T.-W. PAN, T. I. HESLA and D. D. JOSEPH. 'A distributed Lagrange multiplier/fictitious domain method for particulate flows'. In: *Int. J. Multiph. Flow* 25.5 (Aug. 1999), pp. 755–794. DOI: 10.1016/s0301-9322(98)00048-2.

[Glo+00]   R. GLOWINSKI, T.-W. PAN, T. I. HESLA, D. D. JOSEPH and J. PERIAUX. 'A distributed Lagrange multiplier/fictitious domain method for the simulation of flow around moving rigid bodies: application to particulate flow'. In: *Comput. Methods Appl. Mech. Engrg.* 184.2-4 (Apr. 2000), pp. 241–267. DOI: 10.1016/s0045-7825(99)00230-3.

[GPP94a]   R. GLOWINSKI, T.-W. PAN and J. PERIAUX. 'A fictitious domain method for Dirichlet problem and applications'. In: *Comput. Methods Appl. Mech. Engrg.* 111.3-4 (Jan. 1994), pp. 283–303. DOI: 10.1016/0045-7825(94)90135-x.

[GPP94b]   R. GLOWINSKI, T.-W. PAN and J. PERIAUX. 'A fictitious domain method for external incompressible viscous flow modeled by Navier-Stokes equations'. In: *Comput. Methods Appl. Mech. Engrg.* 112.1-4 (Feb. 1994), pp. 133–148. DOI: 10.1016/0045-7825(94)90022-1.

[GH16]   C. GRANDMONT and M. HILLAIRET. 'Existence of Global Strong Solutions to a Beam-Fluid Interaction System'. In: *Arch. Ration. Mech. Anal.* 220.3 (Jan. 2016), pp. 1283–1333. DOI: 10.1007/s00205-015-0954-y.

[Gra+20]   G. GRAVINA, S. SCHWARZACHER, O. SOUČEK and K. TŮMA. *Contactless rebound of elastic bodies in a viscous incompressible fluid*. 3rd Nov. 2020. arXiv: 2011.01932 [math.AP].

[GOR14]   S. GROSS, M. A. OLSHANSKII and A. REUSKEN. *A trace finite element method for a class of coupled bulk-interface transport problems*. 2014. arXiv: 1406.7694v2.

[GOR15]   S. GROSS, M. A. OLSHANSKII and A. REUSKEN. 'A trace finite element method for a class of coupled bulk-interface transport problems'. In: *ESAIM Math. Model. Numer. Anal.* 49.5 (Aug. 2015), pp. 1303–1330. DOI: 10.1051/m2an/2015013.

[GM19]   C. GÜRKAN and A. MASSING. 'A stabilized cut discontinuous Galerkin framework for elliptic boundary value and interface problems'. In: *Comput. Methods Appl. Mech. Engrg.* 348 (May 2019), pp. 466–499. DOI: 10.1016/j.cma.2018.12.041.

[GSM20]   C. GÜRKAN, S. STICKO and A. MASSING. 'Stabilized Cut Discontinuous Galerkin Methods for Advection-Reaction Problems'. In: *SIAM J. Sci. Comput.* 42.5 (9th Sept. 2020), A2620–A2654. DOI: 10.1137/18m1206461.

[Guy65]   R. J. GUYAN. 'Reduction of stiffness and mass matrices'. In: *AIAA J.* 3.2 (Feb. 1965), pp. 380–380. DOI: 10.2514/3.2874.

[GO17]   J. GUZMÁN and M. A. OLSHANSKII. 'Inf-sup stability of geometrically unfitted Stokes finite elements'. In: *Math. Comp.* 87.313 (Dec. 2017), pp. 2091–2112. DOI: 10.1090/mcom/3288.

[Hag20]   T. HAGEMEIER. *Particle Settling-Transitional Regime*. Mendeley Data. Version V1. 1st Sept. 2020. DOI: 10.17632/mf27c92nc3.1.

[HTR21]   T. HAGEMEIER, D. THÉVENIN and T. RICHTER. 'Settling of spherical particles in the transitional regime'. In: *Int. J. Multiph. Flow* 138, 103589 (Feb. 2021). DOI: 10.1016/j.ijmultiphaseflow.2021.103589.

[HNW93]   E. HAIRER, S. P. NØRSETT and G. WANNER. *Solving Ordinary Differential Equations I. Nonstiff Problems*. Berlin, Heidelberg: Springer, 1993. DOI: 10.1007/978-3-540-78862-1.

[HH02]   A. HANSBO and P. HANSBO. 'An unfitted finite element method, based on Nitsche's method, for elliptic interface problems'. In: *Comput. Methods Appl. Mech. Engrg.* 191.47-48 (Nov. 2002), pp. 5537–5552. DOI: 10.1016/s0045-7825(02)00524-8.

[HH03]   P. HANSBO and J. HERMANSSON. 'Nitsche's method for coupling non-matching meshes in fluid-structure vibration problems'. In: *Comput. Mech.* 32.1-2 (Sept. 2003), pp. 134–139. DOI: 10.1007/s00466-003-0467-7.

[HLZ16] P. Hansbo, M. G. Larson and S. Zahedi. 'A cut finite element method for coupled bulk-surface problems on time-dependent domains'. In: *Comput. Methods Appl. Mech. Engrg.* 307 (Aug. 2016), pp. 96–116. doi: 10.1016/j.cma.2016.04.012.

[HB81] J. Happel and H. Brenner. *Low Reynolds number hydrodynamics*. Dordrecht: Springer, 1981. doi: 10.1007/978-94-009-8352-6.

[HP17] F. Hecht and O. Pironneau. 'An energy stable monolithic Eulerian fluid-structure finite element method'. In: *Internat. J. Numer. Methods Fluids* 85.7 (May 2017), pp. 430–446. doi: 10.1002/fld.4388.

[Hei20] F. Heimann. 'On Discontinuous- and Continuous-In-Time Unfitted Space-Time Methods for PDEs on Moving Domains'. MA thesis. Georg-August-Universität Göttingen, Oct. 2020. url: http://cpde.math.uni-goettingen.de/data/Hei20_Ma.pdf.

[HR90] J. G. Heywood and R. Rannacher. 'Finite-element approximation of the nonstationary Navier-Stokes problem. Part IV: Error analysis for second-order time discretization'. In: *SIAM J. Numer. Anal.* 27.2 (Apr. 1990), pp. 353–384. doi: 10.1137/0727022.

[Hil07] M. Hillairet. 'Lack of Collision Between Solid Bodies in a 2D Incompressible Viscous Flow'. In: *Comm. Partial Differential Equations* 32.9 (Sept. 2007), pp. 1345–1371. doi: 10.1080/03605300601088740.

[HT09] M. Hillairet and T. Takahashi. 'Collisions in Three-Dimensional Fluid Structure Interaction Problems'. In: *SIAM J. Math. Anal.* 40.6 (Jan. 2009), pp. 2451–2477. doi: 10.1137/080716074.

[HLZ81] T. J. R. Hughes, W. Kam Liu and T. K. Zimmermann. 'Lagrangian-Eulerian finite element formulation for incompressible viscous flows'. In: *Comput. Methods Appl. Mech. Engrg.* 29.3 (Dec. 1981), pp. 329–349. doi: 10.1016/0045-7825(81)90049-9.

[Iro65] B. Irons. 'Structural eigenvalue problems - elimination of unwanted variables'. In: *AIAA J.* 3.5 (May 1965), pp. 961–962. doi: 10.2514/3.3027.

[IT69] B. M. Irons and R. C. Tuck. 'A version of the Aitken accelerator for computer iteration'. In: *Internat. J. Numer. Methods Engrg.* 1.3 (July 1969), pp. 275–277. doi: 10.1002/nme.1620010306.

[JLL15] A. Johansson, M. G. Larson and A. Logg. 'High order cut finite element methods for the Stokes problem'. In: *Adv. Model. Simul. Eng. Sci.* 2.1 (Sept. 2015). doi: 10.1186/s40323-015-0043-7.

[Joh16] V. John. *Finite element methods for incompressible flow problems*. Cham: Springer, 2016. doi: 10.1007/978-3-319-45750-5.

[Joh+17] V. John, A. Linke, C. Merdon, M. Neilan and L. G. Rebholz. 'On the divergence constraint in mixed finite element methods for incompressible flows'. In: *SIAM Rev.* 59.3 (Jan. 2017), pp. 492–544. doi: 10.1137/15m1047696.

[JMR06] V. John, G. Matthies and J. Rang. 'A comparison of time-discretization/linearization approaches for the incompressible Navier-Stokes equations'. In: *Comput. Methods Appl. Mech. Engrg.* 195.44-47 (Sept. 2006), pp. 5995–6010. doi: 10.1016/j.cma.2005.10.007.

[KB14] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 22nd Dec. 2014. arXiv: 1412.6980 [cs.LG].

[KVV06] C. M. Klaij, J. J. W. van der Vegt and H. van der Ven. 'Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations'. In: *J. Comput. Phys.* 217.2 (Sept. 2006), pp. 589–611. doi: 10.1016/j.jcp.2006.01.018.

[KW08] U. Küttler and W. A. Wall. 'Fixed-point fluid-structure interaction solvers with dynamic relaxation'. In: *Comput. Mech.* 43.1 (Feb. 2008), pp. 61–72. doi: 10.1007/s00466-008-0255-5.

[LCB06] A. Legay, J. Chessa and T. Belytschko. 'An Eulerian-Lagrangian method for fluid-structure interaction based on level sets'. In: *Comput. Methods Appl. Mech. Engrg.* 195.17-18 (Mar. 2006), pp. 2070–2087. doi: 10.1016/j.cma.2005.02.025.

[Leh15] C. Lehrenfeld. 'The Nitsche XFEM-DG space-time method and its implementation in three space dimensions'. In: *SIAM J. Sci. Comput.* 37.1 (Jan. 2015), A245–A270. doi: 10.1137/130943534.

[Leh16] C. Lehrenfeld. 'High order unfitted finite element methods on level set domains using isoparametric mappings'. In: *Comput. Methods Appl. Mech. Engrg.* 300 (Mar. 2016), pp. 716–733. doi: 10.1016/j.cma.2015.12.005.

[Leh17] C. Lehrenfeld. 'A higher order isoparametric fictitious domain method for level set domains'. In: *Geometrically Unfitted Finite Element Methods and Applications - Proceedings of the UCL Workshop 2016*. Ed. by S. Bordas, E. Burman, M. Larson and M. A. Olshanskii. Vol. 121. Lecture Notes in Computational Science and Engineering. Cham: Springer, 2017, pp. 65–92. doi: 10.1007/978-3-319-71431-8_3.

[Leh+21] C. Lehrenfeld, F. Heimann, J. Preuss and H. von Wahl. 'ngsxfem: Add-on to NGSolve for geometrically unfitted finite element discretizations'. In: *J. Open Source Softw.* 6.64 (10th Aug. 2021), p. 3237. doi: 10.21105/joss.03237. url: https://github.com/ngsxfem/ngsxfem.

[LO19] C. Lehrenfeld and M. A. Olshanskii. 'An Eulerian finite element method for PDEs in time-dependent domains'. In: *ESAIM Math. Model. Numer. Anal.* 53.2 (Mar. 2019), pp. 585–614. doi: 10.1051/m2an/2018068.

[LR13] C. Lehrenfeld and A. Reusken. 'Analysis of a Nitsche XFEM-DG Discretization for a Class of Two-Phase Mass Transport Problems'. In: *SIAM J. Numer. Anal.* 51.2 (Jan. 2013), pp. 958–983. doi: 10.1137/120875260.

[LR17] C. Lehrenfeld and A. Reusken. 'Analysis of a high-order unfitted finite element method for elliptic interface problems'. In: *IMA J. Numer. Anal.* 38.3 (Aug. 2017), pp. 1351–1387. doi: 10.1093/imanum/drx041.

[LS16] C. Lehrenfeld and J. Schöberl. 'High order exactly divergence-free Hybrid Discontinuous Galerkin Methods for unsteady incompressible flows'. In: *Comput. Methods Appl. Mech. Eng.* 307 (Aug. 2016), pp. 339–361. doi: 10.1016/j.cma.2016.04.025.

[LH17] I. LOSHCHILOV and F. HUTTER. *Decoupled Weight Decay Regularization*. 14th Nov. 2017. arXiv: 1711.05101 [cs.LG].

[LL21] Y. LOU and C. LEHRENFELD. *Isoparametric unfitted BDF – Finite element method for PDEs on evolving domains*. 19th May 2021. arXiv: 2105.09162 [math.NA].

[Lud20] T. LUDESCHER. 'Multilevel preconditioning of stabilized unfitted finite element discretizations'. PhD thesis. RWTH Aachen, 27th May 2020. DOI: 10.18154/RWTH-2020-07305.

[Mas+14] A. MASSING, M. G. LARSON, A. LOGG and M. E. ROGNES. 'A stabilized Nitsche fictitious domain method for the Stokes problem'. In: *J. Sci. Comput.* 61.3 (Mar. 2014), pp. 604–628. DOI: 10.1007/s10915-014-9838-9.

[MSW18] A. MASSING, B. SCHOTT and W. A. WALL. 'A stabilized Nitsche cut finite element method for the Oseen problem'. In: *Comput. Methods Appl. Mech. Engrg.* 328 (Jan. 2018), pp. 262–300. DOI: 10.1016/j.cma.2017.09.003.

[MRS21] M. MINAKOWSKA, T. RICHTER and S. SAGER. 'A finite element / neural network framework for modeling suspensions of non-spherical particles.' In: *Vietnam J. Math.* 49 (19th Mar. 2021), pp. 207–235. DOI: 10.1007/s10013-021-00477-9.

[MDB99] N. MOËS, J. DOLBOW and T. BELYTSCHKO. 'A finite element method for crack growth without remeshing'. In: *Internat. J. Numer. Methods Engrg.* 46.1 (Sept. 1999), pp. 131–150. DOI: 10.1002/(sici)1097-0207(19990910)46:1<131::aid-nme726>3.0.co;2-j.

[Neu13] M. NEUMÜLLER. 'Space-time methods: fast solvers and applications'. PhD thesis. TU Graz, 3rd June 2013. URL: https://graz.pure.elsevier.com/en/publications/space-time-methods-fast-solvers-and-applications.

[Nit71] J. NITSCHE. 'Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind'. In: *Abh. Math. Semin. Univ. Hambg.* 36.1 (July 1971), pp. 9–15. DOI: 10.1007/bf02995904.

[OQS21] M. A. OLSHANSKII, A. QUAINI and Q. SUN. *An unfitted finite element method for two-phase Stokes problems with slip between phases*. 24th Jan. 2021. arXiv: 2101.09627 [math.NA].

[OR14] M. A. OLSHANSKII and A. REUSKEN. 'Error Analysis of a Space-Time Finite Element Method for Solving PDEs on Evolving Surfaces'. In: *SIAM J. Numer. Anal.* 52.4 (Jan. 2014), pp. 2092–2120. DOI: 10.1137/130936877.

[ORG09] M. A. OLSHANSKII, A. REUSKEN and J. GRANDE. 'A Finite Element Method for Elliptic Equations on Surfaces'. In: *SIAM J. Numer. Anal.* 47.5 (Jan. 2009), pp. 3339–3358. DOI: 10.1137/080717602.

[ORX14] M. A. OLSHANSKII, A. REUSKEN and X. XU. 'An Eulerian Space-Time Finite Element Method for Diffusion Problems on Evolving Surfaces'. In: *SIAM J. Numer. Anal.* 52.3 (Jan. 2014), pp. 1354–1377. DOI: 10.1137/130918149.

[OX17] M. A. OLSHANSKII and X. XU. 'A Trace Finite Element Method for PDEs on Evolving Surfaces'. In: *SIAM J. Sci. Comput.* 39.4 (Jan. 2017), A1301–A1319. DOI: 10.1137/16m1099388.

[PDR07] J. PARVIZIAN, A. DÜSTER and E. RANK. 'Finite cell method'. In: *Computational Mechanics* 41.1 (Apr. 2007), pp. 121–133. DOI: 10.1007/s00466-007-0173-y.

[Pas+19] A. PASZKE et al. 'PyTorch: An Imperative Style, High-Performance Deep Learning Library'. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. WALLACH et al. Vol. 2. Curran Associates, Inc., 2019, pp. 8026–8037. URL: http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf (visited on 08/10/2020).

[PM89] C. S. PESKIN and D. M. MCQUEEN. 'A three-dimensional computational method for blood flow in the heart I. Immersed elastic fibers in a viscous incompressible fluid'. In: *J. Comput. Phys.* 81.2 (Apr. 1989), pp. 372–405. DOI: 10.1016/0021-9991(89)90213-1.

[PE12] D. A. DI PIETRO and A. ERN. *Mathematical aspects of discontinuous Galerkin methods*. Berlin, Heidelberg: Springer, 2012. DOI: 10.1007/978-3-642-22980-0.

[Pre18] J. PREUSS. 'Higher order unfitted isoparametric space-time FEM on moving domains'. MA thesis. Georg-August-Universität Göttingen, 2018. URL: http://cpde.math.uni-goettingen.de/data/Pre18_Ma.pdf.

[Qua17] A. QUARTERONI. *Numerical Models for Differential Problems*. Cham: Springer, 2017. DOI: 10.1007/978-3-319-49316-9.

[Ric13] T. RICHTER. 'A Fully Eulerian formulation for fluid-structure-interaction problems'. In: *J. Comput. Phys.* 233 (Jan. 2013), pp. 227–240. DOI: 10.1016/j.jcp.2012.08.047.

[Ric17] T. RICHTER. *Fluid-structure Interactions. Models, Analysis and Finite Elements*. Vol. 118. Lecture Notes in Computational Science and Engineering. Cham: Springer, 2017. DOI: 10.1007/978-3-319-63970-3.

[Sal88] R. SALVI. 'On the Navier-Stokes equations in non-cylindrical domains: On the existence and regularity'. In: *Math. Z.* 199.2 (June 1988), pp. 153–170. DOI: 10.1007/BF01159649.

[ST08] S. SATHE and T. E. TEZDUYAR. 'Modeling of fluid-structure interactions with the space-time finite elements: contact problems'. In: *Comput. Mech.* 43.1 (May 2008), pp. 51–60. DOI: 10.1007/s00466-008-0299-6.

[ST96] M. SCHÄFER and S. TUREK. 'Benchmark computations of laminar flow around a cylinder'. In: *Notes on Numerical Fluid Mechanics (NNFM)*. Wiesbaden: Vieweg+Teubner Verlag, 1996, pp. 547–566. DOI: 10.1007/978-3-322-89849-4_39.

[Sch97] J. SCHÖBERL. 'NETGEN an advancing front 2D/3D-mesh generator based on abstract rules'. In: *Comput. Vis. Sci.* 1.1 (July 1997), pp. 41–52. DOI: 10.1007/s007910050004.

[Sch14] J. SCHÖBERL. *C++11 implementation of finite elements in NGSolve*. ASC Report No. 30/2014. Tech. rep. Institute for Analysis and Scientific Computing, TU Wien, 26th Sept. 2014. URL: http://www.asc.tuwien.ac.at/~schoeberl/wiki/publications/ngs-cpp11.pdf (visited on 02/10/2020).

[Sch17]   B. Schott. 'Stabilized cut finite element methods for complex interface coupled flow problems'. PhD thesis. Technischen Universität München, 2017. URL: http://mediatum.ub.tum.de?id=1304754.

[SAW19]   B. Schott, C. Ager and W. A. Wall. 'A monolithic approach to fluid-structure interaction based on a hybrid Eulerian-ALE fluid domain decomposition involving cut elements'. In: *Internat. J. Numer. Methods Engrg.* (Mar. 2019). DOI: 10.1002/nme.6047.

[SW14]   B. Schott and W. A. Wall. 'A new face-oriented stabilized XFEM approach for 2D and 3D incompressible Navier-Stokes equations'. In: *Comput. Methods Appl. Mech. Engrg.* 276 (July 2014), pp. 233–265. DOI: 10.1016/j.cma.2014.02.014.

[SZ90]   L. R. Scott and S. Zhang. 'Finite element interpolation of nonsmooth functions satisfying boundary conditions'. In: *Math. Comp.* 54.190 (May 1990), pp. 483–483. DOI: 10.1090/s0025-5718-1990-1011446-7.

[SR20]   F. Sonner and T. Richter. 'Second Order Pressure Estimates for the Crank–Nicolson Discretization of the Incompressible Navier–Stokes Equations'. In: *SIAM J. Numer. Anal.* 58.1 (Jan. 2020), pp. 375–409. DOI: 10.1137/18m1234813.

[Ste70]   E. M. Stein. *Singular Integrals and Differentiability Properties of Functions.* Vol. 30. Princeton Mathematical Series. Princeton, NJ: Princeton University Press, 1970. 304 pp. ISBN: 0-691-08079-8.

[TH73]   C. Taylor and P. Hood. 'A numerical solution of the Navier-Stokes equations using the finite element technique'. In: *Comput. & Fluids* 1.1 (Jan. 1973), pp. 73–100. DOI: 10.1016/0045-7930(73)90027-3.

[TH06]   S. Turek and J. Hron. 'Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow'. In: *Lecture Notes in Computational Science and Engineering.* Berlin, Heidelberg: Springer, 2006, pp. 371–385. DOI: 10.1007/3-540-34596-5_15.

[VR18]   I. Voulis and A. Reusken. 'A time dependent Stokes interface problem: well-posedness and space-time finite element discretization'. In: *ESAIM Math. Model. Numer. Anal.* 52.6 (Nov. 2018), pp. 2187–2213. DOI: 10.1051/m2an/2018053.

[Wah18]   H. von Wahl. 'Implicit-explicit time splitting schemes for incompressible Navier-Stokes flows'. MA thesis. Georg-August-Universität Göttingen, 2018. URL: https://num.math.uni-goettingen.de/picap/pdf/E908.pdf.

[WR21]   H. von Wahl and T. Richter. 'Using a deep neural network to predict the motion of under-resolved triangular rigid bodies in an incompressible flow'. In: *Internat. J. Numer. Methods Fluids* (12th Aug. 2021). DOI: 10.1002/fld.5037.

[Wah+20]   H. von Wahl, T. Richter, S. Frei and T. Hagemeier. *Falling balls in a viscous fluid with contact: Comparing numerical simulations with experimental data.* Zenodo repository, Nov. 2020. DOI: 10.5281/zenodo.3989604.

[Wah+21]   H. von Wahl, T. Richter, S. Frei and T. Hagemeier. 'Falling balls in a viscous fluid with contact: Comparing numerical simulations with experimental data'. In: *Phys. Fluids* 33.3, 033304 (2nd Mar. 2021). DOI: 10.1063/5.0037971.

[WRL20]   H. von Wahl, T. Richter and C. Lehrenfeld. *An unfitted Eulerian finite element method for the time-dependent Stokes problem on moving domains.* Zenodo repository, Feb. 2020. DOI: 10.5281/zenodo.3647571.

[WRL21]   H. von Wahl, T. Richter and C. Lehrenfeld. 'An unfitted Eulerian finite element method for the time-dependent Stokes problem on moving domains'. In: *IMA J. Numer. Anal.* (5th July 2021). DOI: 10.1093/imanum/drab044.

[Wah+19a] H. von Wahl, T. Richter, C. Lehrenfeld, J. Heiland and P. Minakowski. *Numerical benchmarking of fluid-rigid body interactions.* Zenodo repository, June 2019. DOI: 10.5281/zenodo.3253455.

[Wah+19b] H. von Wahl, T. Richter, C. Lehrenfeld, J. Heiland and P. Minakowski. 'Numerical benchmarking of fluid-rigid body interactions'. In: *Comput. & Fluids* 193, 104290 (30th Oct. 2019). DOI: 10.1016/j.compfluid.2019.104290.

[WH03]   T. Warburton and J. S. Hesthaven. 'On the constants in hp-finite element trace inverse inequalities'. In: *Comput. Methods Appl. Mech. Eng.* 192.25 (June 2003), pp. 2765–2773. DOI: 10.1016/s0045-7825(03)00294-9.

[Wei72]   H. Weinberger. 'Variational properties of steady fall in Stokes flow'. In: *J. Fluid Mech.* 52.2 (28th Mar. 1972), pp. 321–344. DOI: 10.1017/S0022112072001442.

[Zah18]   S. Zahedi. 'A Space-Time Cut Finite Element Method with Quadrature in Time'. In: *Geometrically Unfitted Finite Element Methods and Applications - Proceedings of the UCL Workshop 2016.* Ed. by S. Bordas, E. Burman, M. Larson and M. A. Olshanskii. Vol. 121. Lecture Notes in Computational Science and Engineering. Cham: Springer, 14th Mar. 2018, pp. 281–306. DOI: 10.1007/978-3-319-71431-8_9.

[ZAV21]   S. Zonca, P. F. Antonietti and C. Vergara. 'A Polygonal Discontinuous Galerkin Formulation for Contact Mechanics in Fluid-Structure Interaction Problems'. In: *Commun. Comput. Phys.* 30.1 (July 2021), pp. 1–33. DOI: 10.4208/cicp.OA-2020-0079.