Adaptive low and high-order hybridized methods for unsteady incompressible flow simulations

by

Alexandros Karkoulias

A dissertation submitted in fulfillment of the requirements
for the degree of
Doctor of Philosophy
in
Simulation in Engineering and Entrepreneurship Development

An Erasmus Mundus Joint Doctorate programme

Departament d'Enginyeria Civil i Ambiental
Universitat Politècnica de Catalunya, Spain

In joint supervision with
Dipartimento di Ingegneria Civile e Architettura
Università degli Studi di Pavia, Italy

Barcelona, February, 2020
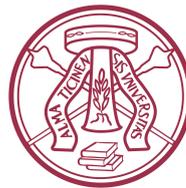
Acknowledgements

I would like to thank my advisors Prof. Antonio Huerta, Dr. Matteo Giacomini from the Mathematical and Computational Modeling group (also known as LaCàN) at the Universitàt Politècnica de Catalunya, as well as Prof. Ferdinando Auricchio from the Computational Mechanics & Advanced Materials Group (CompMech) at the Università degli studi di Pavia for their supervision, insightful feedback and valuable guidance through the thesis process. Their passion for research inspired me and helped me stay focused on the goal, even in times when this seemed blurry and far away. I was made to feel welcome in the research groups of both universities from day one, which facilitated my learning experience and encouraged me to contribute. Apart from obtaining professional knowledge, I also had the chance to grow and mature as an individual and I am grateful for the opportunity I was given.

I would also like to thank Dr. Ruben Sevilla from Zienkiewicz Centre for Computational Engineering at Swansea University for his commitment to the success of the group. Special acknowledgement to Dr. Jose Sarrate, Dr. Sergio Zlotnik and Dr. David Modesto who served as committee members in my thesis proposal defence and whose comments and contribution to the improvement of my thesis were highly valued. Many thanks to all faculty members, PhD students, technicians of LaCàN group at the Universitàt Politècnica de Catalunya and CompMech group at the Università degli studi di Pavia for their assistance. I would like to acknowledge and thank the International Centre for Numerical Methods in Engineering (CIMNE) for all the support from the beginning until the end of my PhD thesis.

I will always be grateful to Dr. Alex Viguerie, who I was lucky to meet at the Università degli studi di Pavia, and who actively believed in me and my abilities. I learned a lot from him and together we bridged the gap between mathematics and engineering. He definitely played an important role in my thesis and a few lines are not enough to express my gratitude towards him. I am glad to consider him as my friend. I am also happy I got the chance to spend many beautiful moments with my colleagues and friends David, Alberto, Olga, Arash, Pavlina, Simona, Karim, Hasini, Andrea, Ahmed and I thank them for being there both in and out of office.

At this point, I want to say a big thanks to my family, my mother Eftychia and my siblings Giota, Dimitra and Dionisis, for their continuous love and support through these years and for believing in me, even when I did not. I could not forget to mention my father who, even though he passed away a long time ago and sometimes seems like a distant memory, I always keep in my heart and I remember everything he taught me.

I would not have been able to make it this far without the invaluable support of my life coach, Giota, who has been by my side and taught me how to pursue my dreams. Last but not least, I want to thank Rizkallah, whose name in Arabic means "gift of God" and whose presence in my life has been a true gift to me. Thanks for all the support, I am blessed to have you in my life.

ii

Abstract

Simulations of incompressible flows are performed on a daily basis to solve problems of practical and industrial interest in several fields of engineering, including automotive, aeronautical, mechanical and biomedical applications.

Although finite volume (FV) methods are still the preferred choice by the industry due to their efficiency and robustness, sensitivity to mesh quality and limited accuracy represent two main bottlenecks of these approaches. This is especially critical in the context of transient phenomena, in which FV methods show excessive numerical diffusion. In this context, there has been a growing interest towards high-order discretization strategies in last decades.

In this PhD thesis, a high-order adaptive hybidizable discontinuous Galerkin (HDG) method is proposed for the approximation of steady and unsteady laminar incompressible Navier-Stokes equations. Voigt notation for symmetric second-order tensors is exploited to devise an HDG method for the Cauchy formulation of the momentum equation with optimal convergence properties, even when low-order polynomial degrees of approximation are considered. In addition, a postprocessing strategy accounting for rigid translational and rotational modes is proposed to construct an element-by-element superconvergent velocity field.

The discrepancy between the computed and postprocessed velocities is utilized to define a local error indicator to drive degree adaptivity procedures and accurately capture localized features of the flow. The resulting HDG solver is thus extended to the case of transient problems via high-order time integration schemes, namely the explicit singly diagonal implicit Runge-Kutta (ESDIRK) schemes. In this context, the embedded explicit step is exploited to define an inexpensive estimate of the temporal error to devise an efficient timestep control strategy.

Finally, in order to efficiently solve the global problem arising from the HDG discretization, a preconditioned iterative solver is proposed. This is critical in the context of high-order approximations in three-dimensional domains leading to large-scale problems, especially in transient simulations. A block diagonal preconditioner coupled with an inexpensive approximation of the Schur complement of the matrix is proposed to reduce the computational cost of the overall HDG solver.

Extensive numerical validation of two and three-dimensional steady and unsteady benchmark tests of viscous laminar incompressible flows is performed to validate the proposed methodology.

Keywords: hybridizable discontinuous Galerkin, incompressible Navier-Stokes equations, transient flows, high-order methods, preconditioning, Cauchy stress formulation, Voigt notation

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the last decades, Computational Fluid Dynamics (CFD) has become a necessary tool not only in the aerospace industry, but in automotive, chemical, civil, mechanical engineering industry, as well as in architecture and electronics. The majority of the commercial codes are based on Finite Volume (FV) and low-order Finite Element (FE) methods. Some well-known solvers that are used in commercial packages are the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE), Pressure-Implicit with Splitting of Operators (PISO) as well as density-based solvers, usually employed for the solution of compressible flows. However, when low-order methods are employed, many levels of mesh refinement are usually required for the solution to converge with an acceptable accuracy in the engineering context. This is due to the extra numerical diffusion introduced by low-order schemes with respect to high-order methods. Of course, this leads to increased computing time and required resources.

In an attempt to deliver more accurate results, the applicability of high-order methods has been studied during the last decades. High-order methods provide higher order of convergence and, consequently, additional accuracy even in presence of coarse meshes. More precisely, high-order methods successfully provided accurate solutions for complex flow problems with complex geometries, transient and vortex-dominated flows [1, 2, 3, 4]. Moreover, a misconception is that a high-order method is more expensive than a low-order one. When two methods are to be compared, then a comparison in terms of computational cost or resources should be done under the same level of accuracy. In that case, a low-order method is likely to be more expensive than a high-order one. Detailed comparisons between high and low-order methods are provided in [5, 2, 6].

Examples of successful application of high-order methods in engineering problems include the simulation of cavitation, fluid-structure interaction (FSI) and the design process of machine components. Simulating cavitation is necessary during the design phase of propellers and pumps [7, 8], otherwise, important mechanical wear and vibrations can be caused to the structure. Fluid-structure interaction is observed in blood flows inside arteries, flows around turbine blades and aircraft wings. The high-fidelity simulation of fluid-structure interaction is discussed in [9, 10]. High-order methods can

be a useful tool during the design process of machine components, such as blades, fans, compressors and turbochargers [11, 12].

(a) Propeller cavitation   (b) Aorta artery
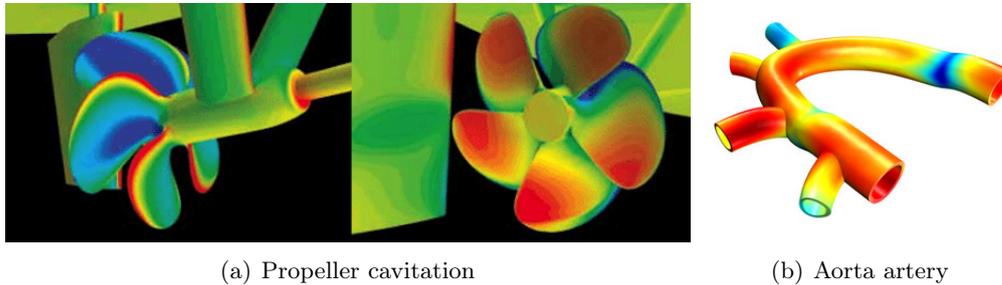
Figure 1.1: *Examples of simulations where high levels accuracy is significant.*
Images downloaded from a) https://gcaptain.com/propeller-cavitation-analysis/, b)
https://www.comsol.com/model/fluid-structure-interaction-in-a-network-of-blood-
vessels-660.

Among finite element methods, Continuous Galerkin (CG) has been widely used for the solution of partial differential equations that describe fluid flow problems [13, 14, 15]. CG provides a continuous, piecewise polynomial approximation of the solution and is known for its accuracy and its reliability with smooth solutions and on conforming meshes. However, degree adaptivity (p-adaptivity), which plays a crucial role in the approximation of flow problems, is not trivial in the CG context due to the difficulty to impose C^0 continuity. In the last decades, the interest in discontinuous Galerkin (DG) method has increased [16, 17]. DG enforces local conservation by using an element-by-element discontinuous polynomial approximation. Moreover, DG methods are especially suited to devise high-order approximations and to perform p-adaptivity. Nevertheless, DG duplicates the nodes on every face of the mesh and, hence, has increased number of globally coupled unknowns with respect to CG for the same mesh. This restricts its use in what concerns the solution of large problems, especially with high-order approximations.

To overcome this issue, two techniques named *static condensation* [18] and *hybridization* [19] have been explored in recent years. This lead to the development of the hybridizable discontinuous Galerkin (HDG) method, by defining the unknowns in each element as solution of a Dirichlet boundary value problem, while the communication between elements is performed by satisfying appropriate *transmission conditions*. This approach leads to *hybrid* discretization techniques, where the only coupled unknowns of the problem are located on the mesh faces. The number of degrees of freedom with respect to DG for the same mesh is reduced, see Figure 1.2, and there is a clear computational advantage with respect to DG [5]. The HDG method proposed by Cockburn and coworkers relies on a mixed hybrid formulation which is based on element-by-element discontinuous polynomial approximations [20]. HDG was derived for the second-order elliptic problem [21, 22] and the convection-diffusion equations

[23]. Afterwards, it was extended to the Stokes flow [24], the incompressible [25] and compressible Navier-Stokes equations [26].
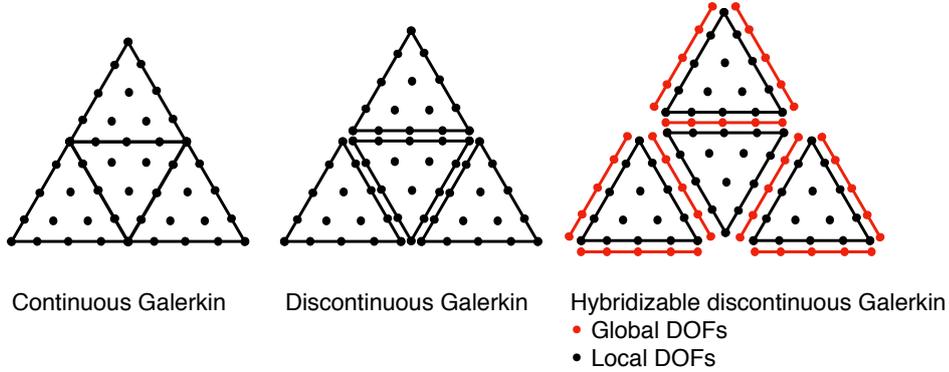
Figure 1.2: *Degrees of freedom in a mesh of four fourth-order triangles for continuous Galerkin, discontinuous Galerkin and hybridizable discontinuous Gelerkin.*

Besides the reduced number of global unknowns, in the case of incompressible flows, HDG allows equal order of interpolation for velocity, pressure and velocity gradient due to the definition of the numerical flux on the boundaries of the elements and the introduction of an appropriate stabilization parameter. In addition, using HDG in viscous dominated flows, the so-called *primal* variables, namely the velocity and pressure, and the *mixed* variable, that is the velocity gradient, converge optimally with order $k+1$ in the $\mathcal{L}_2$-norm when polynomials of degree k are used for all the variables [25]. Given the optimal convergence of the velocity gradient, it is possible to obtain a post-processed velocity field that superconverges with order $k+2$. This process is conducted inside each element and is computationally inexpensive. By exploiting the information of primal and post-processed variables, an error indicator is devised and p-adaptivity procedures are defined.

So far HDG has delivered accurate solutions in the academic setting, however, there is still a long way until the method can be used in an industrial setting. Challenges arise from the fact that, despite the reduced DOFs of HDG in comparison with DG, the linear systems that arise from HDG are still large, especially in three-dimensional flows and for high orders of approximation.This is especially critical in the case of transient problems. Under these circumstances, the goal of this PhD thesis is to provide a deeper understanding of the HDG method from a numerical point of view, showing its potential and its limitations in the context of practical engineering applications.

1.1   Objectives and thesis outline

This thesis proposes an adaptive HDG method, from low- to high-order, for the simulation of steady and unsteady incompressible flows. In order to achieve this aim, the following contributions have been realized:

1. **A hybridizable discontinuous Galerkin method for linear elasticity with optimal accuracy properties**. Chapter 2 presents an HDG method for the linear elasticity problem using the same degree of approximation for primal and mixed variables. Using Voigt notation, the symmetry of the strain gradient is imposed pointwise, without resorting to more complex mathematical frameworks [27, 28, 29]. The optimal convergence of the primal and mixed variables is verified in a variety of two-dimensional and three-dimensional cases even for low-order polynomials. Later, an original way to construct a super-convergent post-processed displacement field is proposed. This is found to lead to optimal convergence properties for a variety of approximation orders, even in low-order ones. The robustness of the method and its locking-free properties for nearly incompressible materials are verified.

2. **A hybridizable discontinuous Galerkin method for the Cauchy formulation of the Stokes equation with optimal convergence properties**. In Chapter 3 an HDG method for the Cauchy formulation of the Stokes equation with strong enforcement of the stress tensor is presented. It is worth noticing that this problem features an incompressibility constraint, thus pressure is also considered as an independent variable. In a similar fashion as in the linear elasticity problem, the same degree of polynomials for velocity, pressure and mixed variable is used. Optimal convergence and superconvergence properties are numerically verified for low-order and high-order approximations. The method is tested on various two-dimensional and three-dimensional flows.

3. **Solution of the steady laminar incompressible Navier-Stokes equations using a hybridizable discontinuous Galerkin method**. In Chapter 4 the HDG method for the laminar incompressible Navier-Stokes equations is presented, by introducing a Newton-Raphson iterative method to solve the nonlinear problem. Various benchmark problems involving steady flows of engineering interest are presented in Section 4.2. Owing to the excellent convergence rates of the post-processed velocity field, a spatial error estimator is constructed to perform p-adaptivity in Section 4.3.

4. **Solution of the transient laminar incompressible Navier-Stokes equations using a hybridizable discontinuous Galerkin method**. The method is extended to include the transient version of the laminar incompressible Navier-Stokes equations. Special attention is devoted to high-order time integration schemes. In Section 4.4 the employed temporal schemes are introduced and the results of HDG method for the simulation of transient flows are presented. In addition, in Section 4.5, transient simulations with degree adaptivity and timestep control are reported.

5. **Effective preconditioning for the solution of incompressible flows using a hybridizable discontinuous Galerkin method**. Direct solvers for the solution of the HDG global linear system scale poorly with problem size not only in three-dimensional flows, but also in two-dimensional flows when high orders of approximation are considered. The exploitation of HDG methods in practical settings therefore requires the use of iterative solvers and, consequently, effective preconditioning strategies. In Chapter 5, a preconditioning technique that is based on the use of a block-triangular preconditioner and the approximation of the Schur complement is presented. Owing to the proposed preconditioning strategy, memory limitations of direct solvers are overcome. The robustness of this approach is verified in various two- and three-dimensional flows and for different values of the Reynolds number. The resulting preconditioned HDG solver is applied to a real problem of medical interest.

1.2 State of the art

1.2.1 Accurate approximations of the linear elastic problem

The numerical approximation of the linear elasticity equation presents several difficulties as highlighted by the extensive literature available on the topic (cf. e.g. [30, 31]). In particular, locking phenomena in nearly incompressible and incompressible materials, construction of stable pairs of finite elements and strong enforcement of the symmetry of the stress tensor in mixed formulations, accurate computation of the stresses (classically recovered from the displacement field via numerical differentiation) and post-process procedures to improve the quality of the approximate displacement fields are some of the subjects that have attracted the attention of the scientific community over the last 40 years.

It is well-known that an accurate approximation of the linear elastic problem for nearly incompressible materials requires the discrete space in which the solution is sought to be rich enough to describe nontrivial divergence-free vector fields. Within this context, the primal formulation where the displacement field is the sole unknown fails to provide a locking-free approximation using low-order conforming Lagrangian finite element functions. In fact, Brenner and Sung [32] proposed a possible remedy by means of the nonconforming Crouzeix-Raviart element [33]. In order to circumvent this issue, two main approaches have been proposed in the literature. On the one hand, mixed formulations in which both the displacement field and the stress tensor act as unknowns of a saddle point problem [34]. On the other hand, discontinuous Galerkin (DG) discretizations in which the approximate displacement field is sought in a bigger space and the variational formulation of the problem is modified to account for the jumps of the discrete displacement field across the element interfaces [35, 36].

Starting from the seminal paper by Reissner [34], mixed variational formulations of the linear elasticity equation have known a great success in the scientific community. The solution of the resulting saddle point problem provides an approximation of both the displacement field and the stress tensor that is not retrieved as a post-processed quantity (with a consequent loss of precision) as in the primal formulation [37]. A major drawback of the mixed formulation lies in the difficulty of constructing a pair of finite element spaces that fulfil the requirements of Brezzi's theory [38] to guarantee the stability of the method. More precisely, concurrently imposing the balance of momentum by seeking a stress tensor in $H(\mathrm{div})$ (i.e. a square-integrable tensor with square-integrable row-wise divergence) and the balance of angular momentum by enforcing its symmetry proved to be an extremely difficult task [39]. Stemming from the pioneering work by Fraejis de Veubeke [40], a first approach discussed in the literature relies on maintaining the $H(\mathrm{div})$-conformity of the stress tensor while its symmetry is relaxed (cf. e.g. [41, 42, 43]). Among the most successful approaches, the so-called PEERS element by Arnold, Brezzi and Douglas Jr. [44] introduced a Lagrange multiplier as extra variable to account for the symmetry constraint, see also [45, 46, 47, 48, 49, 50]. In 2002, the first stable pair of finite element spaces for the discretization of the mixed formulation of the linear elasticity equation with $H(\mathrm{div})$-conforming strongly enforced symmetric stress tensor in two dimensions was proposed by Arnold and Winther [51]. The corresponding three dimensional case is discussed in [27, 28]. Nevertheless, the construction of these finite element pairs is based on nontrivial techniques of exterior calculus [29] and results in a large number of degrees of freedom per element (the lowest-order approximation of the stress tensor features 24 degrees of freedom on a triangle and 162 on a tetrahedron) making their application to complex problems unfeasible. More recently, an alternative mixed formulation featuring a tangential-continuous displacement field and a normal-normal continuous symmetric stress tensor has been proposed by Pechstein and Schöberl [52]. For a detailed discussion on mixed methods, the interested reader is referred to [53].

An alternative approach to the discretization of the linear elastic problem focuses on relaxing the $H(\mathrm{div})$-conformity of the stress tensor while strongly enforcing its pointwise symmetry. This results in nonconforming discretizations (cf. e.g. [54, 55, 56]). Moreover, owing to the fact that these methods use polynomial basis functions but no degrees of freedom is located in the vertices of the elements, Gopalakrishnan and Guzmán [57] show that the resulting nonconforming approximation may be efficiently implemented via hybridization. Nevertheless, the convergence rate of the stress tensor is sub-optimal when using this nonconforming discretization. Among nonconforming discretizations, the DG method has experienced a great success in recent years. The interest in DG methods for the linear elastic problem is motivated by their high-order convergence properties and their flexibility in performing local h- and p-adaptivity. Beside the aforementioned works by Hansbo and Larson [35, 36], see also [58, 59].

The ability of DG methods to efficiently construct a locking-free approximation in nearly incompressible materials has been recently analysed in [60] also for the case of heterogeneous media. In [61], a discontinuous Petrov-Galerkin (DPG) formulation is proposed to simultaneously approximate the displacement field and the symmetric stress tensor and is shown to be *hp*-optimal.

More recently, novel discretization techniques inspired by the previously discussed ones have been proposed. The local discontinuous Galerkin (LDG) method [62] is based on a mixed discontinuous Galerkin formulation and provides an exactly incompressible approximation of the displacement field, that is a displacement field which is normal-continuous across inter-element boundaries and pointwise incompressible inside each element. The method converges optimally for the displacement field whereas the strain tensor and the pressure results are sub-optimal by one order. Moreover, contrary to the framework discussed by Gopalakrishnan and Guzmán [57], the method cannot be hybridized, thus resulting in a considerable number of degrees of freedom for high-order approximations. Stemming from the work on LDG, Cockburn and co-workers have proposed the hybridizable discontinuous Galerkin (HDG) method whose analysis for the linear elasticity equation is available in [63, 64]. HDG is based on a mixed discontinuous Galerkin formulation with hybridization and provides an optimally convergent displacement field with order $k+1$ whereas the strain and stress tensor converge sub-optimally with order $k + 1/2$. The optimal approximation of the stress tensor may be retrieved by adding matrix bubble functions to the discrete space as discussed in [65]. The alternative HDG formulation by Qiu et al. [66] exploits polynomials of different degrees for the approximation of the displacement field (order $k+1$), its trace (order k) and the strain tensor (order k). By introducing a modified definition of the numerical trace, optimal convergence of order $k + 1$ for all the unknowns is retrieved.

In [67], Di Pietro and Ern discuss the hybrid high-order (HHO) method which features a nonconforming discretization based on a pure displacement formulation of the linear elastic problem. The method leads to a locking-free displacement field and a strongly symmetric strain tensor, both converging with optimal order. Similarly to the hybridization in HDG, the definition of the unknowns on the faces reduces the computational cost associated with the solution of the problem, making HHO suitable for high-order approximations. It is worth noting that in [68], Cockburn, Di Pietro and Ern re-interpreted the HHO method for scalar diffusion problems within the HDG framework originally discussed in the seminal contribution [21]. In a similar fashion, HHO for linear elastic problems [67] is a close relative of the HDG method by Cockburn and co-workers [63, 64] and, consequently, to the HDG formulation based on Voigt notation discussed in the present work. Nonetheless, as the authors remark, HHO differs from HDG in the choice of a novel set of local discretization spaces and in the definition of the numerical trace of the flux. On the contrary, the present work maintains the classical HDG mixed formulation with the corresponding numerical trace

of the flux proposed in [63, 64] and reduces the space of the mixed variable to vector-valued functions accounting solely for the non-redundant terms owing to Voigt notation.

Eventually, other high-order formulations based on a description of the displacement field by means of a hybrid variable and a strong enforcement of the symmetry of the stress tensor have been discussed in the literature [69, 70, 71]. For a comprehensive presentation of the dual hybrid approaches known as *equilibrium formulations*, the interested reader is referred to the monograph [72]. Inspired by the works of Arnold and Brezzi [73] and Stenberg [47] on mixed methods, Cockburn and co-workers have investigated several procedures to construct a super-convergent post-processed solution by exploiting both the optimally convergent primal and mixed variables. Nevertheless, as shown in [63], the case of linear elasticity experiences a sub-optimal convergence of the strain tensor and consequently a loss of super-convergence for the post-processed displacement field. A great effort within the HDG community is currently devoted to investigating techniques to remedy this issue and retrieve the super-convergence of the post-processed variable even for low-order approximations (i.e. $k = 1, 2$). More precisely, Cockburn and co-workers have recently introduced the concept of $\boldsymbol{M}$-decomposition [74, 75] to construct discrete spaces suitable to retrieve the aforementioned super-convergence property. This elegant theory guarantees that an HDG approximation for which the local space admits an $\boldsymbol{M}$-decomposition provides a locking-free approximate displacement field, an optimally convergent approximate stress tensor and a super-convergent post-processed displacement field obtained via an element-by-element procedure. Nevertheless, the construction of such spaces is non-trivial and their implementation in existing HDG library is not straightforward.

1.2.2 The incompressible limit: approximating the Stokes flow with HDG

The interest in discontinuous Galerkin (DG) methods [16, 76, 77, 17, 78, 79] has increased in the past years owing to their ability to construct high-order discretizations on unstructured meshes and to their flexibility in performing *p*-adaptivity. Among the different techniques proposed in the literature to approximate incompressible flow problems, the recent growing interest towards hybridizable discontinuous Galerkin (HDG) methods [25, 80] is due to multiple advantages these formulations have with respect to classical DG ones. Concerning Stokes flow, several HDG formulations have been proposed in the literature [24, 81, 82] and the interested reader is referred to [83] for an overview on the topic.

The use of hybridization was first introduced with the local discontinuous Galerkin (LDG) method to circumvent the construction of divergence-free approximations of the velocity field [84], see also [85, 86, 87, 88]. Moreover, owing to hybridization [89, 90], the globally coupled unknowns are defined on the boundary of the mesh elements and are connected solely to neighbouring elements. Thus, the size of the global problem is

greatly reduced. In addition, it is worth noting that HDG allows equal interpolation for velocity, pressure and strain rate tensor, owing to an appropriate definition of the numerical flux and to the introduction of a stabilization parameter. Thus, the limitations of using equal-order approximations for velocity and pressure in the incompressible limit, through the fulfillment of the Ladyzhenskaya-Babuška-Brezzi (LBB) condition, are circumvented by HDG. In particular, Cockburn and co-workers [91] proved solvability and stability under the aforementioned assumptions, without the need of an enriched space for the mixed variable, or a reduced space for the hybrid one. In [92, 93], under suitable assumptions on the regularity of the domain and the solution, optimal convergence rates of order $k + 1$ are obtained for all the variables using equal degree of approximation k, whereas classical DG display suboptimal convergence of order k for the pressure and the gradient of the velocity.

A key aspect of HDG is the ability to construct a postprocessed velocity field superconverging with order $k+2$ [94]. This is crucial when the superconvergent solution is sought to devise automatic procedures to perform p-adaptivity (cf. e.g. [95, 96]). Nevertheless, the classical HDG equal-order approximation for the Cauchy formulation is known to experience suboptimal convergence of the mixed variable and a loss of superconvergence of the postprocessed velocity field using low-order approximations [81].

Recently, in a series of publications [97, 98, 99, 100], Cockburn and co-workers devoted a great effort to develop a general framework, namely the $\boldsymbol{M}$-decomposition, to devise superconvergent HDG discretizations. This approach relies on enriching the local spaces for the approximation of the mixed variable by adding extra basis functions. The number of these additional basis functions is not significantly big and in most cases it depends on the type of element under analysis and not on the degree of approximation k. Despite only the size of the local problems increases and the additional computational effort is limited, it induces a more complex implementation compared to standard HDG methods. Alternative HDG formulations achieve convergence of order $k + 2$ for the velocity field when polynomials of degree k are chosen to approximate the hybrid variable [101, 102, 103]. These methods rely on utilizing smaller spaces for the mixed variable and larger ones for the velocity and exploiting a special stabilization function, the so-called *reduced stabilization*, to handle them. Closely related approaches, namely the hybrid high-order (HHO) [67] and the hybridized weak Galerkin (HWG) [104] methods can also achieve the same orders of convergence.

1.2.3 Adaptivity in transient simulations of the incompressible Navier-Stokes equations

Adaptive strategies play a crucial role in capturing localized features of complex flow phenomena. Depending on the characteristics of the flow, appropriate adaptivity procedures have been proposed in literature, for instance in the case of incompressible

flows [105], compressible flows [106, 107, 108], turbulent flows [109] and transient flows [110].

The most common refinement strategies in CFD are h-adaptivity [111, 112, 113], p-adaptivity [114, 95, 96] and r-adaptivity [115, 116]. Mesh or h-adaptivity consists in node insertion in specific areas of the mesh. Degree or p-adaptivity relies on changing the polynomial degree of approximation locally. It requires a flexible scheme that is able to handle non-uniform degrees in neighbouring elements. DG methods are suitable for degree adaptivity, since the degree can be non-uniform from element to element and the continuity is enforced weakly through numerical fluxes [108]. Relocation or r-adaptivity is based on that the nodes are allowed to move towards areas of the mesh where increased resolution is required, always maintaining their initial connectivity. Moreover, hp-adaptivity methods, with a change in order and size of the elements, have been used in literature [117, 118, 119, 120].

A key aspect to develop adaptive strategies is the definition of an appropriate error indicator. Exploiting the primal and postprocessed velocities, degree adaptive procedures have been developed in HDG for wave problems, Stokes equations and incompressible Navier-Stokes equations [114, 96, 95, 121].

As far as the temporal discretization is concerned, high-order time-marching methods have been used with HDG without affecting the spatial convergence properties of the method. In [25] a backward difference formula of order 3 is employed. An appropriate timestep size did not affect the spatial order of convergence of HDG, which was excellent in the case of gradient-pressure boundary conditions on the Neumann boundary, whereas in the case of stress boundary conditions, the post-processed velocity is found to converge with smaller order than expected. Later, a space-time HDG method for incompressible Navier-Stokes and Oseen equations on moving and deforming domains was introduced in [122]. Special space-time elements are employed and high order in time is achieved by increasing the order of polynomial approximation in these elements. In [103] the authors employ three different approaches for time integration, namely *additive decomposition* using implicit-explicit schemes, *operator-integration-factor splitting* and *modified fractional-step-θ-scheme*, while they used the concept of reduced stabilization for HDG. A projection method, namely the rotational incremental pressure correction scheme, was introduced in [123] along with implicit-explicit Runge-Kutta schemes for time integration. Theta-method for temporal discretization was employed in [124] with pointwise divergence-free and $H(\text{div})$-conforming velocity field and in [125] for the implementation of a particle-mesh method with HDG.

As far as degree adaptivity is concerned, a distinction is made between steady and transient flows. In steady simulations degree adaptivity is performed only once while in transient simulations it is performed repeatedly, thereby ensuring that the approximation is constantly adapted to the movement of the flow [126, 127]. In transient

simulations, the timestep size can be also controlled. Adaptivity in time can be performed via Richardson's extrapolation [128], computing every timestep twice, first with size Δt and then with size $\Delta t/2$. Although this approach is extremely simple, it is computationally unaffordable. An alternative approach relies on embedding techniques, in which an error estimator is devised starting from two solutions that differ by one order. Embedding techniques with Radau methods [129] and fully-implicit Runge-Kutta methods [130] have been developed. ESDIRK schemes include stages that allow for efficient embedded error estimation as discussed in [131, 132]. Timestep control has been also considered as a feedback control problem by various authors who have proposed a PID-controller type for the adjustment of timestep size [133, 131].

1.2.4 Preconditioning for Navier-Stokes equations

The bottleneck in the application of HDG methods to large-scale problems is the size of the global system to be solved. Although the use of effectively preconditioned iterative linear solvers with HDG is necessary, this is still an active field of investigation.

A famous approach on the application of iterative linear solvers to the HDG method involves the use of multigrid methods, an overview of which can be found in [134]. Geometric multigrid methods are based on iterative schemes on different mesh levels that start from a fine grid and construct coarse grids and are famous on structured meshes. In [135], a variable V-cycle algorithm was introduced for the solution of Laplace equation using HDG. On the other hand, especially when unstructured grids and high-order polynomials are considered, p-multigrid methods that iterate on solution approximations of different polynomial order are used, see [136] for the solution of Euler equations with DG. In [137] the convergence study of the multigrid method was performed and in [138] a performance comparison between continuous and discontinuous methods was conducted. A p–multigrid iterative method was used for transient compressible Navier–Stokes in [139]. It was shown that it led to fewer GMRES iterations when compared to other *single-grid* preconditioners, like ILU or linear Jacobi, though this did not have an impact on the CPU time when the HDG formulation was considered.

In [140] a diagonal Jacobi precoditioner was used along with the conjugate gradient method (PCG) for the solution of the elliptic and diffusion problems, demonstrating that the stabilization parameter τ, that is used to penalize the jump of the flux between elements, plays a role in the convergence of the PCG method. Block diagonal preconditioners were developed for a HDG method for the Stokes equations [141], where the iteration count was shown to be independent of the problem size. Other approaches for the solution of the linear systems that arise from the HDG method include the iterative solvers presented in [142, 143], Schwarz methods as shown in [144], and the hierarchical scale separation approach detailed in [145].

Chapter 2

Superconvergent HDG method for linear elasticity [1]

In this chapter, the hybridizable discontinuous Galerkin (HDG) method, that uses the same degree approximation for primal and mixed variables, for linear elastic problems is presented. An important feature is the strong imposition of the symmetry of the stress tensor, which remedies the loss of optimal convergence properties of the mixed variable in the HDG formulation. This is particularly important for low-order approximations (first and second order), since they tend to experience sub-optimal behaviour using the classical HDG formulation. Given the optimal convergence of the mixed variable for $k \geq 1$, a new approach to construct a superconvergent displacement field is proposed, even for low-order approximations. Since the displacement field is identified up to rigid motions (three in 2D and 6 in 3D), some additional constraints for the uniqueness of the displacement field are necessary.

In Section 2.1, the linear elastic problem is recalled and, using the Voigt notation for symmetric tensors, it is rewritten strongly enforcing the symmetry of the stress tensor. In Section 2.2, the HDG framework discussed in [63] is considered and discrete spaces featuring equal order interpolation for all the variables are employed. In Section 2.3, a novel way to achieve a superconvergence displacement field is presented.

Extensive numerical tests in both two and three dimensions are presented in Section 2.4 to validate the convergence rates of the primal, mixed and post-processed variables, using different types of elements commonly implemented in commercial finite element solvers. Special attention is given to the nearly incompressible limit case in which the novel formulation confirms to be locking-free and the optimal convergence rates are preserved. Section 2.5 summarises the discussed results whereas the implementation details are provided in Appendix A.

---

[1]This chapter is a modified version of the published article: R. Sevilla, M. Giacomini, A. Karkoulias, A. Huerta, A super-convergent hybridizable discontinuous Galerkin method for linear elasticity, International Journal for Numerical Methods in Engineering, 116 (2) (2018) 91-116 [146].

2.1   Problem statement

In this section, the governing equations that describe the mechanical behaviour of a deformable solid within the infinitesimal strain theory are introduced and the corresponding formulation using the Voigt notation for symmetric tensors is recalled. For a complete introduction to this subject, the interested reader is referred to [147, 148, 149].

2.1.1   Strong form of the linear elastic problem

Let $\Omega \subset \mathbb{R}^{n_{sd}}$ be an open bounded domain in n_{sd} spatial dimensions with boundary $\partial\Omega = \overline{\Gamma}_D \cup \overline{\Gamma}_N$, $\overline{\Gamma}_D \cap \overline{\Gamma}_N = \emptyset$ and Γ_D featuring positive $(n_{sd}-1)$-dimensional Hausdorff measure. The mechanical behaviour of a deformable solid Ω within the infinitesimal strain theory is the described by

$$
\begin{cases}
-\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{f} & \text{in } \Omega, \\
\boldsymbol{\sigma} = \boldsymbol{\sigma}^T & \text{in } \Omega, \\
\boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\
\boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t} & \text{on } \Gamma_N,
\end{cases}
\tag{2.1}
$$

where $\boldsymbol{u}$ is the displacement field and $\boldsymbol{\sigma}$ is the Cauchy stress tensor. The elastic structure Ω under analysis is thus subject to a volume force $\boldsymbol{f}$, a tension $\boldsymbol{t}$ on the surface Γ_N and an imposed displacement $\boldsymbol{u}_D$ on Γ_D.

Equation (2.1) is the strong form of the linear elastic problem and states two conservation laws, namely the balance of momentum and the balance of angular momentum. Remark that the latter implies the symmetry of the stress tensor, that is $\boldsymbol{\sigma}$ belongs to the space $\mathbb{S}^{n_{sd}}$ of $n_{sd} \times n_{sd}$ symmetric matrices. The full set of equations is closed by a material law that describes the relationship among the variables at play and depends on the type of solid under analysis. In particular, a linear elastic material is considered. Within this context, the so-called Hooke's law establishes a linear dependency between the stress tensor $\boldsymbol{\sigma}$ and the linearised strain tensor $\boldsymbol{\varepsilon}(\boldsymbol{u}) := \left(\boldsymbol{\nabla}\boldsymbol{u} + \boldsymbol{\nabla}\boldsymbol{u}^T\right)/2$ via the fourth-order tensor $A : \Omega \to \mathbb{S}^{n_{sd}}$ known as the elasticity tensor. In this work, only homogeneous isotropic materials are considered, whence the elasticity tensor A depends neither on the spatial coordinate $\boldsymbol{x}$ nor on the direction of the main strains. The mechanical properties of a linear elastic homogeneous isotropic material are determined by the pair (E, ν), respectively known as Young's modulus and Poisson's ratio (cf. e.g. [148]). Within the range of physically admissible values of these constants (i.e. $\nu \in (-1, 0.5)$), the relationship between the stress tensor and the linearised strain tensor reads

$$
\boldsymbol{\sigma} = A\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{E}{1+\nu}\boldsymbol{\varepsilon}(\boldsymbol{u}) + \frac{E\nu}{(1+\nu)(1-2\nu)} \operatorname{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}))\mathbf{I}_{n_{sd}},
\tag{2.2}
$$

where $\mathbf{I}_{n_{sd}}$ is the $n_{sd} \times n_{sd}$ identity matrix and $\mathrm{tr}(\cdot) := \cdot\,{:}\,\mathbf{I}_{n_{sd}}$ is the trace operator, being **:** the Frobenius product, also known as double contraction. For the purpose of the current work, only nonauxetic materials are considered, that is the Poisson's ratio ν is assumed to be nonnegative.

By plugging (2.2) into (2.1), the stress tensor may be expressed in terms of the displacement field and the pure displacement formulation of the linear elastic problem is retrieved:

$$\begin{cases} -\boldsymbol{\nabla} \cdot (A\boldsymbol{\varepsilon}(\boldsymbol{u})) = \boldsymbol{f} & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \boldsymbol{n} \cdot (A\boldsymbol{\varepsilon}(\boldsymbol{u})) = \boldsymbol{t} & \text{on } \Gamma_N. \end{cases} \tag{2.3}$$

Remark 1. The elasticity tensor exists and is invertible as long as $\nu < 0.5$. It is straightforward to observe that when $\nu \to 0.5$, the divergence of the displacement field in (2.2) has to vanish, that is, the material under analysis is incompressible.

This case cannot be properly handled by the pure displacement formulation since the elasticity tensor deteriorates and A fails to exist in the incompressible limit, thus preventing the stress tensor to be expressed in terms of the displacement field. A possible remedy is represented by mixed formulations in which both the displacement field and the stress tensor act as unknown of the problem. The associated first-order problem is thus obtained by considering the following system of equations:

$$\begin{cases} -\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{f} & \text{in } \Omega, \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}^T & \text{in } \Omega, \\ \boldsymbol{\sigma} = A\boldsymbol{\varepsilon}(\boldsymbol{u}) & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t} & \text{on } \Gamma_N. \end{cases} \tag{2.4}$$

2.1.2 Strong enforcement of the symmetry of the stress tensor

Consider the classical theory of linear elasticity [150]. Let $\boldsymbol{u} := \begin{bmatrix} u_i \end{bmatrix}^T \in \mathbb{R}^{n_{sd}}$, $i = 1,\dots,n_{sd}$ be the vector field describing the displacement. The strain tensor may be divided into its diagonal components (namely, the extensional strains ε_{ii}) and its off-diagonal terms γ_{ij} known as shear strains

$$\varepsilon_{ii} := \frac{\partial u_i}{\partial x_i}, \quad \gamma_{ij} := \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}, \quad \text{for } i,j = 1,\dots,n_{sd}. \tag{2.5}$$

Owing to its symmetry, only three components in 2D (two extensional and one shear strains) and six components in 3D (three extensional and three shear strains) need to be stored. More precisely, according to the so-called Voigt notation, the components

of the strain may be arranged as a column vector in $\mathbb{R}^{\mathtt{m_{sd}}}$ as follows:

$$\boldsymbol{\varepsilon}_{\mathrm{V}} := \begin{cases} \left[\varepsilon_{11},\ \varepsilon_{22},\ \gamma_{12}\right]^{T} & \text{in 2D,} \\ \left[\varepsilon_{11},\ \varepsilon_{22},\ \varepsilon_{33},\ \gamma_{12},\ \gamma_{13},\ \gamma_{23}\right]^{T} & \text{in 3D,} \end{cases} \tag{2.6}$$

where $\mathtt{m_{sd}} = \mathtt{n_{sd}}(\mathtt{n_{sd}} + 1)/2$. Here the components are ordered following the rationale proposed by Fish and Belytschko[150].

Remark 2. The linearised strain tensor $\boldsymbol{\varepsilon}(\boldsymbol{u}) \in \mathbb{S}^{\mathtt{n_{sd}}}$ differs from its Voigt counterpart $\boldsymbol{\varepsilon}_{\mathrm{V}} \in \mathbb{R}^{\mathtt{m_{sd}}}$ by a factor $1/2$ in the shear components, that is:

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) := \begin{cases} \begin{bmatrix} \varepsilon_{11} & \gamma_{12}/2 \\ \gamma_{12}/2 & \varepsilon_{22} \end{bmatrix} & \text{in 2D,} \\ \begin{bmatrix} \varepsilon_{11} & \gamma_{12}/2 & \gamma_{13}/2 \\ \gamma_{12}/2 & \varepsilon_{22} & \gamma_{23}/2 \\ \gamma_{13}/2 & \gamma_{23}/2 & \varepsilon_{33} \end{bmatrix} & \text{in 3D.} \end{cases} \tag{2.7}$$

Following the framework described by Fish and Belytschko in [150], the matrix $\boldsymbol{\nabla}_{\mathrm{S}} \in \mathbb{R}^{\mathtt{m_{sd}} \times \mathtt{n_{sd}}}$ accounting for the symmetric gradient operator is introduced:

$$\boldsymbol{\nabla}_{\mathrm{S}} := \begin{cases} \begin{bmatrix} \partial/\partial x_1 & 0 & \partial/\partial x_2 \\ 0 & \partial/\partial x_2 & \partial/\partial x_1 \end{bmatrix}^{T} & \text{in 2D,} \\ \begin{bmatrix} \partial/\partial x_1 & 0 & 0 & \partial/\partial x_2 & \partial/\partial x_3 & 0 \\ 0 & \partial/\partial x_2 & 0 & \partial/\partial x_1 & 0 & \partial/\partial x_3 \\ 0 & 0 & \partial/\partial x_3 & 0 & \partial/\partial x_1 & \partial/\partial x_2 \end{bmatrix}^{T} & \text{in 3D.} \end{cases} \tag{2.8}$$

Thus, the components of $\boldsymbol{\varepsilon}_{\mathrm{V}}$ may be expressed in terms of the displacements $\boldsymbol{u}$ by means of a single matrix equation $\boldsymbol{\varepsilon}_{\mathrm{V}} = \boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}$.

In a similar fashion, within the stress tensor $\boldsymbol{\sigma}$ two normal components σ_{ii} and one shear component τ_{ij} in 2D (respectively, three and three in 3D) may be identified owing to the symmetry arising form the balance of angular momentum (cf. equation (2.4)). Thus, according to Voigt notation, the stress tensor may be written as the following column vector in $\mathbb{R}^{\mathtt{m_{sd}}}$:

$$\boldsymbol{\sigma}_{\mathrm{V}} := \begin{cases} \left[\sigma_{11},\ \sigma_{22},\ \tau_{12}\right]^{T} & \text{in 2D,} \\ \left[\sigma_{11},\ \sigma_{22},\ \sigma_{33},\ \tau_{12},\ \tau_{13},\ \tau_{23}\right]^{T} & \text{in 3D.} \end{cases} \tag{2.9}$$

2.1.3 The linear elastic problem using Voigt notation

In this section, the previously introduced Voigt notation is exploited to rewrite the linear elastic problem (2.4) by strongly enforcing the symmetry of the stress tensor. The second equation in (2.4) is thus verified in a straightforward manner. The balance of

momentum may be rewritten as a matrix equation by exploiting the notation introduced in (2.8) for the symmetric gradient operator:

$$-\boldsymbol{\nabla}_{\mathrm{S}}^{T}\boldsymbol{\sigma}_{\mathrm{V}} = \boldsymbol{f}. \tag{2.10}$$

Moreover, the constitutive equation (2.2) may be expressed as $\boldsymbol{\sigma}_{\mathrm{V}} = \mathbf{D}\boldsymbol{\varepsilon}_{\mathrm{V}}$, where $\mathbf{D}$ is an $\mathtt{m}_{\mathrm{sd}} \times \mathtt{m}_{\mathrm{sd}}$ symmetric positive definite matrix describing the generalised Hooke's law.

Remark 3. In two dimensions, the structure of the matrix $\mathbf{D}$ depends on the assumption made to simplify the three dimensional model. On the one hand, according to the *plane strain* model, the body is thick with respect to the plane $x_1 x_2$ and consequently the extensional strain along x_3 and the shear strains $\gamma_{i3}, i = 1, 2$ vanish. On the other hand, the *plane stress* model is based on the assumption that the body is thin relative to the dimensions in the $x_1 x_2$ plane. Thus, no loads are applied along the x_3 direction and the component σ_{33} of the stress tensor is assumed to vanish.

$$\mathbf{D} := \begin{cases} \dfrac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & (1-2\nu)/2 \end{bmatrix} & \text{in 2D (plane strain)}, \\[3em] \dfrac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} & \text{in 2D (plane stress)}, \\[3em] \dfrac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & \\ \nu & 1-\nu & \nu & \mathbf{0}_{\mathrm{n_{sd}}} \\ \nu & \nu & 1-\nu & \\ & \mathbf{0}_{\mathrm{n_{sd}}} & & (1-2\nu)/2\mathbf{I}_{\mathrm{n_{sd}}} \end{bmatrix} & \text{in 3D.} \end{cases} \tag{2.11}$$

Following the same rationale discussed above, an $\mathtt{m}_{\mathrm{sd}} \times \mathtt{n}_{\mathrm{sd}}$ matrix accounting for the normal direction to the boundary is introduced:

$$\mathbf{N} := \begin{cases} \begin{bmatrix} n_1 & 0 & n_2 \\ 0 & n_2 & n_1 \end{bmatrix}^{T} & \text{in 2D}, \\[2em] \begin{bmatrix} n_1 & 0 & 0 & n_2 & n_3 & 0 \\ 0 & n_2 & 0 & n_1 & 0 & n_3 \\ 0 & 0 & n_3 & 0 & n_1 & n_2 \end{bmatrix}^{T} & \text{in 3D}, \end{cases} \tag{2.12}$$

and the matrix counterpart of the traction boundary conditions is imposed on Γ_N, that is $\mathbf{N}^T \boldsymbol{\sigma}_{\mathrm{V}} = \boldsymbol{t}$.

Hence, the linear elastic problem (2.4) using Voigt notation reads as follows:

$$\begin{cases} -\boldsymbol{\nabla}_S^T \boldsymbol{\sigma}_V = \boldsymbol{f} & \text{in } \Omega, \\ \boldsymbol{\sigma}_V = \mathbf{D}\boldsymbol{\varepsilon}_V & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \mathbf{N}^T \boldsymbol{\sigma}_V = \boldsymbol{t} & \text{on } \Gamma_N. \end{cases} \tag{2.13}$$

2.1.4 Generalised Gauss's and Stokes' theorems

In order to state the variational formulation of Equation (2.13), a counterpart of the classical Gauss's theorem using the Voigt matrices introduced in the previous sections is required. The following result holds:

Lemma 1 (Generalised Gauss's theorem). *Consider a vector $\boldsymbol{v} \in \mathbb{R}^{n_{sd}}$ and a symmetric tensor $\boldsymbol{\varsigma} \in \mathbb{S}^{n_{sd}}$ whose counterpart in Voigt notation is $\boldsymbol{\varsigma}_V$. It holds:*

$$\int_{\partial\Omega} \left(\mathbf{N}^T \boldsymbol{\varsigma}_V\right) \cdot \boldsymbol{v} \, d\Gamma = \int_\Omega \boldsymbol{\varsigma}_V \cdot (\boldsymbol{\nabla}_S \boldsymbol{v}) \, d\Omega + \int_\Omega \left(\boldsymbol{\nabla}_S^T \boldsymbol{\varsigma}_V\right) \cdot \boldsymbol{v} \, d\Omega. \tag{2.14}$$

Proof. Rewrite each term in (2.14) in terms of the operators associated with the matrices introduced by the Voigt notation:

$$\int_{\partial\Omega} \left(\mathbf{N}^T \boldsymbol{\varsigma}_V\right) \cdot \boldsymbol{v} \, d\Gamma = \int_{\partial\Omega} (\boldsymbol{n} \cdot \boldsymbol{\varsigma}) \cdot \boldsymbol{v} \, d\Gamma, \tag{2.15a}$$

$$\int_\Omega \boldsymbol{\varsigma}_V \cdot (\boldsymbol{\nabla}_S \boldsymbol{v}) \, d\Omega = \int_\Omega \boldsymbol{\varsigma} : \boldsymbol{\varepsilon}(\boldsymbol{v}) \, d\Omega, \tag{2.15b}$$

$$\int_\Omega \left(\boldsymbol{\nabla}_S^T \boldsymbol{\varsigma}_V\right) \cdot \boldsymbol{v} \, d\Omega = \int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{\varsigma}) \cdot \boldsymbol{v} \, d\Omega. \tag{2.15c}$$

By summing the right hand sides of (2.15), the classical statement of Gauss's theorem is retrieved and consequently (2.14) holds. □

The aforementioned result allows to derive the formulation of the HDG method which will be discussed in Section 2.2. Moreover, in Section 2.3, a novel post-process procedure of the HDG solution which relies on a condition on the curl operator will be introduced. In order to properly state the aforementioned results, first consider the infinitesimal rotation of a vector field using Voigt notation. Consider $\mathbf{R} \in \mathbb{R}^{n_{rr} \times n_{sd}}$, with n_{rr} the number of rigid body rotations in the space (one in 2D and three in 3D). Within this rationale, $\text{curl}(\boldsymbol{u}) := \boldsymbol{\nabla} \times \boldsymbol{u}$ may be written as the matrix equation $\text{curl}_V(\boldsymbol{u}) = \mathbf{R}\boldsymbol{u}$, where

$$\mathbf{R} := \begin{cases} \left[-\partial/\partial x_2, \ \partial/\partial x_1\right] & \text{in 2D}, \\ \begin{bmatrix} 0 & -\partial/\partial x_3 & \partial/\partial x_2 \\ \partial/\partial x_3 & 0 & -\partial/\partial x_1 \\ -\partial/\partial x_2 & \partial/\partial x_1 & 0 \end{bmatrix} & \text{in 3D}. \end{cases} \tag{2.16}$$

Remark 4. Recall that the curl of a vector field $\boldsymbol{v} \in \mathbb{R}^2$ exists solely as a scalar quantity, namely

$$\boldsymbol{\nabla} \times \boldsymbol{v} = \frac{\partial v_2}{\partial x_1} - \frac{\partial v_1}{\partial x_2}. \tag{2.17}$$

Nevertheless, by embedding $\boldsymbol{v}$ in $\mathbb{R}^3$ and setting its third component equal to zero, the curl may be interpreted as a vector pointing entirely in the direction x_3 with magnitude given by $\mathbf{R}\boldsymbol{v}$, that is, the value on the right hand side of Equation (2.17).

Moreover, consider the following matrix $\mathbf{T} \in \mathbb{R}^{n_{rr} \times n_{sd}}$ describing the tangent direction to the boundary of $\Omega \subset \mathbb{R}^{n_{sd}}$, that is a tangent line in 2D and a tangent surface in 3D:

$$\mathbf{T} := \begin{cases} \begin{bmatrix} -n_2, & n_1 \end{bmatrix} & \text{in 2D,} \\ \begin{bmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{bmatrix} & \text{in 3D.} \end{cases} \tag{2.18}$$

As previously done for the Gauss's theorem, a generalised Stokes' theorem using the Voigt matrices is stated:

Lemma 2 (Generalised Stokes' theorem). *Consider a vector $\boldsymbol{v} \in \mathbb{R}^{n_{sd}}$. It holds:*

$$\int_\Omega \mathbf{R}\boldsymbol{v} \; d\Omega = \int_{\partial\Omega} \mathbf{T}\boldsymbol{v} \; d\Gamma. \tag{2.19}$$

Proof. Following the same rationale used in Lemma 1, each term in (2.19) may be rewritten as follows:

$$\int_\Omega \mathbf{R}\boldsymbol{v} \; d\Omega = \int_\Omega \boldsymbol{\nabla} \times \boldsymbol{v} \; d\Omega, \tag{2.20a}$$

$$\int_{\partial\Omega} \mathbf{T}\boldsymbol{v} \; d\Gamma = \int_{\partial\Omega} \boldsymbol{v} \cdot \boldsymbol{t} \; d\Gamma, \tag{2.20b}$$

where $\boldsymbol{t}$ is the tangential direction to the boundary $\partial\Omega$. By plugging (2.20) into (2.19), the classical statement of Stokes' theorem is retrieved and consequently (2.19) holds.

$\square$

2.2 Hybridizable discontinuous Galerkin formulation

Consider a partition of the domain Ω in n_{el} disjoint subdomains Ω_e with boundaries $\partial\Omega_e$. The internal interface Γ is defined as

$$\Gamma := \left[\bigcup_{e=1}^{n_{el}} \partial\Omega_e \right] \setminus \partial\Omega. \tag{2.21}$$

The second-order elliptic problem of Equation (2.13) can be written in mixed form, in the so-called broken computational domain, as a system of first-order equations,

namely

$$
\begin{cases}
\boldsymbol{L} + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = \boldsymbol{0} & \text{in } \Omega_e, \text{ and for } e = 1,\dots,\mathtt{n}_{\mathtt{el}}, \\
\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2}\boldsymbol{L} = \boldsymbol{f} & \text{in } \Omega_e, \text{ and for } e = 1,\dots,\mathtt{n}_{\mathtt{el}}, \\
\boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\
\mathbf{N}^T \mathbf{D}^{1/2}\boldsymbol{L} = -\boldsymbol{t} & \text{on } \Gamma_N, \\
[\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \boldsymbol{0} & \text{on } \Gamma, \\
[\![\mathbf{N}^T \mathbf{D}^{1/2}\boldsymbol{L}]\!] = \boldsymbol{0} & \text{on } \Gamma,
\end{cases}
\tag{2.22}
$$

where $[\![\cdot]\!]$ denotes the *jump* operator, defined along each portion of the interface according to [87] as the sum of the values from the element on the right and left, say Ω_e and Ω_l:

$$
[\![\odot]\!] = \odot_e + \odot_l.
\tag{2.23}
$$

Therefore, the last two equations in (2.22) enforce the continuity of respectively the primal variable - i.e. the displacement field - and the normal trace of the stress across the interface Γ.

2.2.1   Strong form of the local and global problems

The HDG formulation solves the problem of Equation (2.22) in two stages [21, 91, 23, 151, 24, 25]. First a local pure Dirichlet problem is defined to compute $(\boldsymbol{L}_e, \boldsymbol{u}_e)$ element-by-element in terms of the unknown hybrid variable $\widehat{\boldsymbol{u}}$, namely

$$
\begin{cases}
\boldsymbol{L}_e + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = \boldsymbol{0} & \text{in } \Omega_e \\
\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2}\boldsymbol{L}_e = \boldsymbol{f} & \text{in } \Omega_e \\
\boldsymbol{u}_e = \boldsymbol{u}_D & \text{on } \partial\Omega_e \cap \Gamma_D, \\
\boldsymbol{u}_e = \widehat{\boldsymbol{u}} & \text{on } \partial\Omega_e \setminus \Gamma_D,
\end{cases}
\tag{2.24}
$$

for $e = 1,\dots,\mathtt{n}_{\mathtt{el}}$.

Second, the global problem is defined to determine the hybrid variable (i.e. the trace of the displacement field on the mesh skeleton $\Gamma \cup \Gamma_N$), namely

$$
\begin{cases}
[\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \boldsymbol{0} & \text{on } \Gamma, \\
[\![\mathbf{N}^T \mathbf{D}^{1/2}\boldsymbol{L}]\!] = \boldsymbol{0} & \text{on } \Gamma, \\
\mathbf{N}^T \mathbf{D}^{1/2}\boldsymbol{L} = -\boldsymbol{t} & \text{on } \Gamma_N.
\end{cases}
\tag{2.25}
$$

As usual in an HDG context, the first equation in (2.25) is automatically satisfied due to the unique definition of the hybrid variable $\widehat{\boldsymbol{u}}$ on each face and the Dirichlet boundary condition $\boldsymbol{u}_e = \widehat{\boldsymbol{u}}$ imposed in the local problems.

2.2.2 Weak form of the local and global problems

Following the notation in [152], the discrete functional spaces

$$\mathcal{V}^h(\Omega) := \left\{ v \in \mathcal{L}_2(\Omega) : v|_{\Omega_e} \in \mathcal{P}^k(\Omega_e) \ \forall \Omega_e \,, \ e = 1, \dots, \mathtt{n_{el}} \right\}, \tag{2.26a}$$

$$\hat{\mathcal{V}}^h(S) := \left\{ \hat{v} \in \mathcal{L}_2(S) : \hat{v}|_{\Gamma_i} \in \mathcal{P}^k(\Gamma_i) \ \forall \Gamma_i \subset S \subseteq \Gamma \cup \partial\Omega \right\}, \tag{2.26b}$$

are introduced, where $\mathcal{P}^k(\Omega_e)$ and $\mathcal{P}^k(\Gamma_i)$ are the spaces of polynomial functions of complete degree at most k in Ω_e and on Γ_i respectively. In addition, the classical internal products of vector functions in $\mathcal{L}_2(\Omega_e)$ and $\mathcal{L}_2(\Gamma_i)$

$$(\boldsymbol{p}, \boldsymbol{q})_{\Omega_e} := \int_{\Omega_e} \boldsymbol{p} \cdot \boldsymbol{q} \ d\Omega, \qquad \langle \hat{\boldsymbol{p}}, \hat{\boldsymbol{q}} \rangle_{\partial\Omega_e} := \sum_{\Gamma_i \subset \partial\Omega_e} \int_{\Gamma_i} \hat{\boldsymbol{p}} \cdot \hat{\boldsymbol{q}} \ d\Gamma \tag{2.27}$$

are considered.

For each element Ω_e, $e = 1, \dots, \mathtt{n_{el}}$, the discrete weak formulation of (2.24) reads as follows: given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}}$ that satisfies

$$-(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\!\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{2.28a}$$

$$-(\boldsymbol{\nabla}_{\!\mathrm{S}} \boldsymbol{w}, \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} + \langle \boldsymbol{w}, \mathbf{N}_e^T \widehat{\mathbf{D}^{1/2} \boldsymbol{L}_e^h} \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{f})_{\Omega_e}, \tag{2.28b}$$

for all $(\boldsymbol{v}, \boldsymbol{w}) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}}$.

Integrating by parts Equation (2.28b) and introducing the following definition of the trace of the numerical stress featuring a stabilisation parameter $\boldsymbol{\tau}_e$

$$\mathbf{N}_e^T \widehat{\mathbf{D}^{1/2} \boldsymbol{L}_e^h} := \begin{cases} \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h + \boldsymbol{\tau}_e (\boldsymbol{u}_e^h - \boldsymbol{u}_D) & \text{on } \partial\Omega_e \cap \Gamma_D, \\ \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h + \boldsymbol{\tau}_e (\boldsymbol{u}_e^h - \widehat{\boldsymbol{u}}^h) & \text{elsewhere}, \end{cases} \tag{2.29}$$

leads to the symmetric form of the discrete weak local problem: for $e = 1, \dots, \mathtt{n_{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}}$ that satisfies

$$-(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\!\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{2.30a}$$

$$(\boldsymbol{w}, \boldsymbol{\nabla}_{\!\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} + \langle \boldsymbol{w}, \boldsymbol{\tau}_e \boldsymbol{u}_e^h \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{f})_{\Omega_e} + \langle \boldsymbol{w}, \boldsymbol{\tau}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \boldsymbol{w}, \boldsymbol{\tau}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{2.30b}$$

for all $(\boldsymbol{v}, \boldsymbol{w}) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}}$.

Similarly, the discrete weak form of the global problem that accounts for the transmission conditions and the Neumann boundary condition is: find $\widehat{\boldsymbol{u}}^h \in [\hat{\mathcal{V}}^h(\Gamma \cup \Gamma_N)]^{\mathtt{n_{sd}}}$ such that

$$\sum_{e=1}^{\mathrm{n_{el}}}\left\{\langle\widehat{\boldsymbol{w}},\mathbf{N}_e^T\mathbf{D}^{1/2}\boldsymbol{L}_e^h\rangle_{\partial\Omega_e\setminus\Gamma_D}+\langle\widehat{\boldsymbol{w}},\boldsymbol{\tau}_e\,\boldsymbol{u}_e^h\rangle_{\partial\Omega_e\setminus\Gamma_D}-\langle\widehat{\boldsymbol{w}},\boldsymbol{\tau}_e\,\widehat{\boldsymbol{u}}^h\rangle_{\partial\Omega_e\setminus\Gamma_D}\right\}=$$
$$-\sum_{e=1}^{\mathrm{n_{el}}}\langle\widehat{\boldsymbol{w}},\boldsymbol{t}\rangle_{\partial\Omega_e\cap\Gamma_N},\tag{2.31}$$

for all $\widehat{\boldsymbol{w}}\in[\hat{\mathcal{V}}^h(\Gamma\cup\Gamma_N)]^{\mathrm{n_{sd}}}$.

2.2.3   Spatial discretization

The discretization of the weak form of the local problem given by Equation (2.30) using an isoparametric formulation for the primal and mixed variables leads to a linear system with the following structure

$$\begin{bmatrix}\mathbf{A}_{LL}&\mathbf{A}_{Lu}\\\mathbf{A}_{Lu}^T&\mathbf{A}_{uu}\end{bmatrix}_e\begin{Bmatrix}\mathbf{L}_e\\\mathbf{u}_e\end{Bmatrix}=\begin{Bmatrix}\mathbf{f}_L\\\mathbf{f}_u\end{Bmatrix}_e+\begin{bmatrix}\mathbf{A}_{L\hat{u}}\\\mathbf{A}_{u\hat{u}}\end{bmatrix}_e\hat{\mathbf{u}}_e,\tag{2.32}$$

for $e=1,\dots,\mathrm{n_{el}}$.

Similarly, using an isoparametric formulation for the hybrid variable produce the following system of equations

$$\sum_{e=1}^{\mathrm{n_{el}}}\left\{\begin{bmatrix}\mathbf{A}_{L\hat{u}}^T&\mathbf{A}_{u\hat{u}}^T\end{bmatrix}_e\begin{Bmatrix}\mathbf{L}_e\\\mathbf{u}_e\end{Bmatrix}+[\mathbf{A}_{\hat{u}\hat{u}}]_e\,\hat{\mathbf{u}}_e\right\}=\sum_{i=e}^{\mathrm{n_{el}}}[\mathbf{f}_{\hat{u}}]_e.\tag{2.33}$$

The expressions of the matrices and vectors appearing in Equation (2.32)-(2.33) are detailed in Appendix A.

After replacing the solution of the local problem of Equation (2.32) in Equation (2.33), the global problem becomes

$$\widehat{\mathbf{K}}\hat{\mathbf{u}}=\hat{\mathbf{f}},\tag{2.34}$$

with

$$\widehat{\mathbf{K}}=\mathbf{A}_{e=1}^{\mathrm{n_{el}}}\begin{bmatrix}\mathbf{A}_{L\hat{u}}^T&\mathbf{A}_{u\hat{u}}^T\end{bmatrix}_e\begin{bmatrix}\mathbf{A}_{LL}&\mathbf{A}_{Lu}\\\mathbf{A}_{Lu}^T&\mathbf{A}_{uu}\end{bmatrix}_e^{-1}\begin{bmatrix}\mathbf{A}_{L\hat{u}}\\\mathbf{A}_{u\hat{u}}\end{bmatrix}_e+[\mathbf{A}_{\hat{u}\hat{u}}]_e\tag{2.35a}$$

and

$$\hat{\mathbf{f}}=\mathbf{A}_{e=1}^{\mathrm{n_{el}}}[\mathbf{f}_{\hat{u}}]_e-\begin{bmatrix}\mathbf{A}_{L\hat{u}}^T&\mathbf{A}_{u\hat{u}}^T\end{bmatrix}_e\begin{bmatrix}\mathbf{A}_{LL}&\mathbf{A}_{Lu}\\\mathbf{A}_{Lu}^T&\mathbf{A}_{uu}\end{bmatrix}_e^{-1}\begin{Bmatrix}\mathbf{f}_L\\\mathbf{f}_u\end{Bmatrix}_e.\tag{2.35b}$$

2.2.4   A remark on the $\mathcal{L}_2$ convergence rates for the primal and mixed variables

Differently from the classical results for HDG [21, 91, 23, 151, 24, 25], the best convergence rates proved by Cockburn and co-workers for the linear elasticity equation [64]

only achieve a convergence of order k for the gradient of the displacement field. The convergence rate of both the strain and stress tensors achieves order $k+1/2$ but remains sub-optimal with respect to the one of the displacement field (order $k+1$). This issue vanishes when moving to high-order approximations in which the optimal convergence of the gradient of the displacement field is retrieved. Nevertheless, the aforementioned limitation represents a major drawback for the application of the classical HDG formulation using polynomials of degree less than 3.

The formulation based on Voigt notation discussed in this article outperforms the convergence rates proved in [64] by always achieving order $k+1$ for all the variables (cf. Section 2.4). In next Section, the optimal numerical convergence of the mixed variable is exploited to construct a post-processed displacement field which superconverges with order $k+2$.

2.3 Superconvergent post-process of the displacement field

As previously mentioned, a known feature of the HDG method is the possibility to exploit the accuracy granted by the convergence of order $k+1$ of the mixed variable (i.e. the stress tensor) to perform a local post-process of the primal variable and construct element-by-element a displacement field $\boldsymbol{u}^\star$ superconverging with order $k+2$. Nevertheless, for the linear elastic problem under analysis the classical approach in [63] shows some issues resulting in a loss of superconvergence of the post-processed solution for low-order approximations. Following [152], in this section a novel post-process procedure is discussed and the superconvergence of $\boldsymbol{u}^\star$ is retrieved.

Introduce the space $\mathcal{V}_\star^h(\Omega)$ of the polynomials of complete degree at most $k+1$ on each element Ω_e:

$$\mathcal{V}_\star^h(\Omega) := \left\{ v \in \mathcal{L}_2(\Omega) : v|_{\Omega_e} \in \mathcal{P}^{k+1}(\Omega_e) \; \forall \Omega_e \,, \; e = 1, \ldots, \mathtt{n}_{\mathtt{el}} \right\}. \tag{2.36}$$

For each element Ω_e, $e = 1, \ldots, \mathtt{n}_{\mathtt{el}}$, consider the definition of the mixed variable in (2.24):

$$\boldsymbol{L}_e + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = \mathbf{0}. \tag{2.37}$$

The post-processed solution $\boldsymbol{u}^\star$ is sought in the richer space $\left[\mathcal{V}_\star^h(\Omega)\right]^{\mathtt{n}_{\mathtt{sd}}}$ and fulfils the following element-by-element problem:

$$\begin{cases} \boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{u}_e^\star = -\boldsymbol{\nabla}_{\mathrm{S}}^T \boldsymbol{L}_e & \text{in } \Omega_e, \; e = 1, \ldots, \mathtt{n}_{\mathtt{el}}, \\ \mathbf{N}^T \mathbf{D}^{1/2} \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{u}_e^\star = -\mathbf{N}^T \boldsymbol{L}_e & \text{on } \partial \Omega_e. \end{cases} \tag{2.38}$$

Remark 5. The solution of Equation (2.38) is not uniquely identified in $\left[\mathcal{V}_\star^h(\Omega)\right]^{\mathtt{n}_{\mathtt{sd}}}$. More precisely, it is unique excluding *rigid motions*, that is up to a family of functions $\boldsymbol{v}^\star$ such that $\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{v}^\star = \mathbf{0}$. From a practical point of view, $\boldsymbol{u}^\star$ is identified up to

three (respectively, six) constants in two (respectively, three) dimensions. Each constant is associated with one rigid motion, namely two translations and one rotation (respectively, three and three) in 2D (respectively, 3D).

In order to remove the above mentioned underdetermination, three additional constraints in 2D and six in 3D are required. Moreover, these conditions have to converge with a *sufficiently high* degree of accuracy as will be detailed below (cf. Remarks 6 and 7). First, consider the classical superconvergent solvability constraint added in the HDG literature to close Equation (2.38):

$$\int_{\Omega_e} \boldsymbol{u}_e^{\star} \, d\Omega = \int_{\Omega_e} \boldsymbol{u}_e^{h} \, d\Omega. \tag{2.39}$$

Remark 6. For the post-processed variable $\boldsymbol{u}^{\star}$ to converge with order $k+2$ in the $\mathcal{L}_2(\Omega)$ norm, the mean value of $\boldsymbol{u}$ inside the element Ω_e in Equation (2.39) has to converge with order at least $k + 2$ as discussed by Cockburn and co-workers [91, 22, 153].

It is straightforward to observe that condition (2.39) removes the underdetermination related to the translational modes. Nonetheless, one additional constraint is required in 2D and three in 3D to remove the rotational modes. A first strategy proposed by Soon et al. [63] relies on decomposing the post-processed solution in two components, the first one arising from the projection of the HDG solution onto the space of rigid motion displacements and the second one from the solution of (2.38) in the space of polynomials with no rigid motion. This post-process technique is inspired by the work of Stenberg [47] on mixed finite elements and allows to retrieve the uniqueness of the post-processed solution but the superconvergence is lost for low-order approximations. Alternatively, in recent years [75], the extremely elegant, but rather complicated, framework of the $\boldsymbol{M}$-decomposition has been extensively studied to devise the proper discrete spaces to obtain optimal convergence and superconvergence of the post-processed solution.

In the following section, three different approaches to account for the rotational modes will be numerically compared. First, the previously mentioned idea by Soon et al. [63] is re-interpreted constraining Equations (2.38)-(2.39) by means of the following condition accounting for the rigid rotation of the displacement field with respect to the barycentre of the element:

$$\int_{\Omega_e} (\boldsymbol{x} - \boldsymbol{x}_b) \times \boldsymbol{u}_e^{\star} \, d\Omega = \int_{\Omega_e} (\boldsymbol{x} - \boldsymbol{x}_b) \times \boldsymbol{u}_e^{h} \, d\Omega, \tag{2.40}$$

where $\boldsymbol{x}$ is the position vector and $\boldsymbol{x}_b$ is the barycentre of the element Ω_e under analysis. Second, a novel constraint which has not been previously considered in the literature is proposed to substitute (2.40). More precisely, a constraint on the mean value of the curl inside the element Ω_e is introduced

$$\int_{\Omega_e} \boldsymbol{\nabla} \times \boldsymbol{u}_e^{\star} \, d\Omega = \int_{\Omega_e} \boldsymbol{\nabla} \times \boldsymbol{u}_e^{h} \, d\Omega. \tag{2.41}$$

By applying the Stokes' theorem to the right-hand side of (2.41), an alternative formulation which exploits the hybrid variable $\widehat{\boldsymbol{u}}$ is obtained

$$\int_{\Omega_e} \boldsymbol{\nabla} \times \boldsymbol{u}_e^\star \, d\Omega = \int_{\partial\Omega_e} \widehat{\boldsymbol{u}}^h \cdot \boldsymbol{t}_e \, d\Gamma, \qquad (2.42)$$

where $\boldsymbol{t}_e$ is the tangential direction to the boundary $\partial\Omega_e$.

Remark 7. Similarly to the observation in Remark 6, for the post-processed variable $\boldsymbol{u}^\star$ to converge with order $k+2$ in the $\mathcal{L}_2(\Omega)$ norm, the mean value of its curl on the left hand side of Equations (2.41) and (2.42) has to converge with order at least $k+1$. According to Equation (2.41), it follows that the mean value of the curl of the solution $\boldsymbol{u}$ inside the element Ω_e has to converge with order at least $k+1$, whereas owing to (2.42) the average of the tangential component of $\widehat{\boldsymbol{u}}^h$ along the boundary $\partial\Omega_e$ has to converge with order at least $k+3/2$.

By exploiting the Voigt notation introduced in Section 2.1.4, (2.41)-(2.42) may be written as

$$\int_{\Omega_e} \mathbf{R}\boldsymbol{u}_e^\star \, d\Omega = \begin{cases} \displaystyle\int_{\Omega_e} \mathbf{R}\boldsymbol{u}_e^h \, d\Omega & \text{according to (2.41)}, \\[2ex] \displaystyle\int_{\partial\Omega_e} \mathbf{T}\widehat{\boldsymbol{u}}^h \, d\Gamma & \text{according to (2.42)}. \end{cases} \qquad (2.43)$$

An alternative physical interpretation of conditions (2.41)-(2.42) is given in [154], exploiting the definition of the vorticity of a fluid as the curl of its velocity field. For additional details on the post-process procedures inspired by the velocity-pressure-vorticity formulation of the Stokes equation, the interested reader is referred to [81, 93].

2.4   Numerical examples

2.4.1   Optimal order of convergence

This section considers two examples, in two and three dimensions, with known analytical solution to test the optimal convergence properties of the error of the primal and mixed variables, $\boldsymbol{u}$ and $\boldsymbol{L}$ respectively, measured in the $\mathcal{L}_2(\Omega)$ norm and for different types of elements.

Two dimensional example

The first example considers a plane stress model for the problem of Equation (2.1) in the domain $\Omega = [0,1]^2$. The external load is selected so that the analytical solution is

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{c} \frac{1}{100}\, x_2 \sin(\pi x_1) \\[2ex] \frac{1}{100}\left(x_1^3 + \cos(\pi x_2)\right) \end{array} \right\}. \qquad (2.44)$$

(a) Quadrilateral mesh (b) Quadrilateral mesh (c) Triangular mesh 1 (d) Triangular mesh 2
1   2
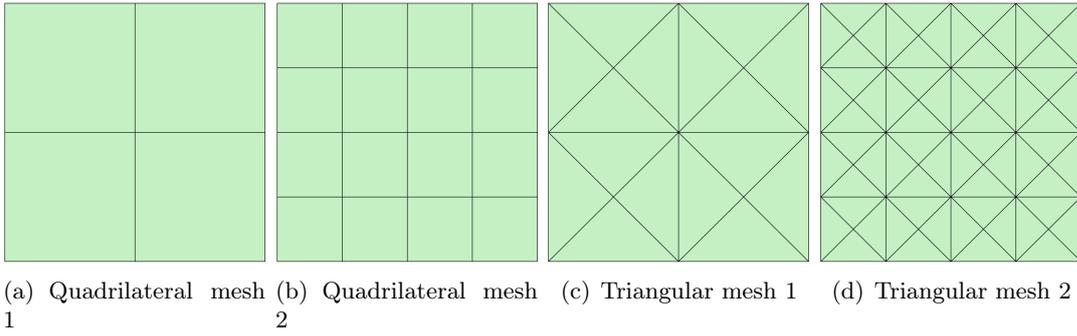
Figure 2.1: *Two dimensional meshes of $\Omega = [0,1]^2$ for the mesh convergence study.*

(a) u_1 (b) u_2 (c) σ_{VM}
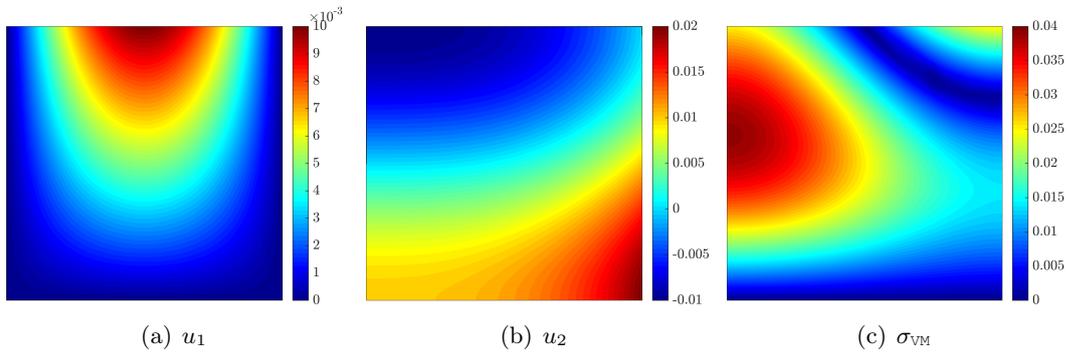
Figure 2.2: *Two dimensional problem: HDG approximation of the displacement field and the Von Mises stress using the third triangular mesh and $k = 2$.*

Neumann boundary conditions, corresponding to the analytical normal stress, are imposed on $\Gamma_N = \{(x_1, x_2) \in \Omega \mid x_2 = 0\}$ and Dirichlet boundary conditions, corresponding to the analytical solution, are imposed on $\Gamma_D = \partial\Omega \setminus \Gamma_N$. The Young's modulus is taken as $E = 1$ and the Poisson's ratio is $\nu = 0.25$.

Uniform meshes of quadrilateral and triangular elements are considered to perform an h-convergence study. The first two quadrilateral and triangular meshes are shown in Figure 2.1.

The displacement field and the Von Mises stress computed on the third triangular mesh and using a quadratic degree of approximation are depicted in Figure 2.2.

The convergence of the error of the primal and mixed variables $\boldsymbol{u}$ and $\boldsymbol{L}$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.3 for both quadrilateral and triangular elements and for a degree of approximation ranging from $k = 1$ up to $k = 3$. It can be observed that the optimal rate of convergence h^{k+1} is obtained for all the element types and degrees of approximation considered. It is worth noting that for the same characteristic element size, the triangular meshes have four times more internal faces than the quadrilateral mesh with the same element size. Therefore, despite the results in Figure 2.3 indicate that for the same element size triangular elements provide more accurate results, when a comparison in terms of the number of degrees of freedom is performed both elements

(a) Quadrilaterals   (b) Triangles
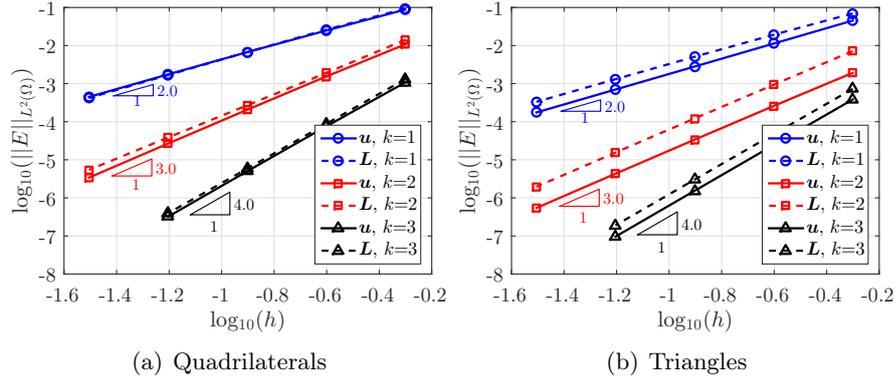
Figure 2.3: *Two dimensional problem: h-convergence of the error of the primal and mixed variables, $\boldsymbol{u}$ and $\boldsymbol{L}$ in the $\mathcal{L}_2(\Omega)$ norm for quadrilateral and triangular meshes with different orders of approximation.*

provide similar accuracy.

Three dimensional example

The next example considers the model problem of Equation (2.1) in the domain $\Omega = [0,1]^3$. The external load is selected so that the analytical solution is

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{l} \frac{1}{100}\left(x_1\sin(2\pi x_2) + x_2\cos(2\pi x_3)\right) \\ \frac{1}{100}\left(x_2\sin(2\pi x_3) + x_3\cos(2\pi x_1)\right) \\ \frac{1}{100}\left(x_3\sin(2\pi x_1) + x_1\cos(2\pi x_2)\right) \end{array} \right\}. \tag{2.45}$$

Neumann boundary conditions, corresponding to the analytical normal stress, are imposed on $\Gamma_N = \{(x_1, x_2, x_3) \in \Omega \mid x_3 = 0\}$ and Dirichlet boundary conditions, corresponding to the analytical solution, are imposed on $\Gamma_D = \partial\Omega \setminus \Gamma_N$. The Young's modulus is taken as $E = 1$ and the Poisson's ratio is $\nu = 0.25$.

Uniform meshes of hexahedral, tetrahedral, prismatic and pyramidal elements are considered to perform an h-convergence study. A cut through the meshes of the domain for the third level of refinement considered is represented in Figure 2.4 for all the element types.

The displacement field and the Von Mises stress computed on the fourth hexahedral mesh and using a cubic degree of approximation are depicted in Figure 2.5.

Analogously to the previous example, the convergence of the error of the primal and mixed variables $\boldsymbol{u}$ and $\boldsymbol{L}$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.6 for all the element types and for a degree of approximation ranging from $k = 1$ up to $k = 3$. It can be observed that a near optimal rate of convergence h^{k+1} is obtained for all the element types and degrees of approximation considered.

(a) Hexahedral mesh 3   (b) Tetrahedral mesh 3   (c) Prismatic mesh 3   (d) Pyramidal mesh 3
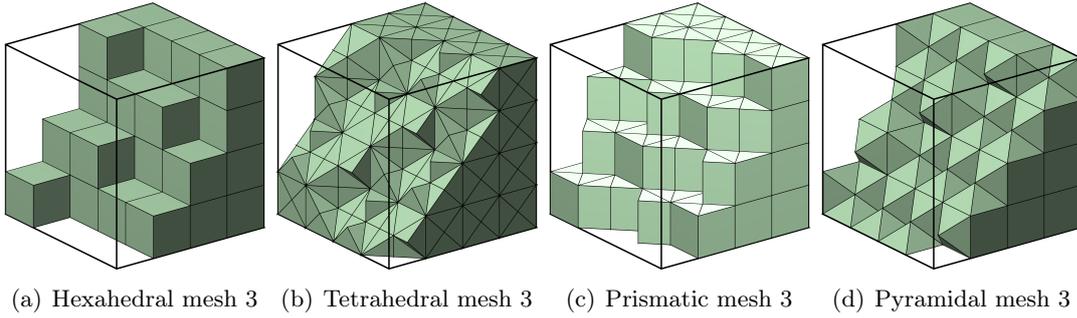
Figure 2.4: *Three dimensional meshes of $\Omega = [0,1]^3$ for the mesh convergence study.*

(a) u_1 (b) u_2 (c) u_3 (d) σ_{VM}
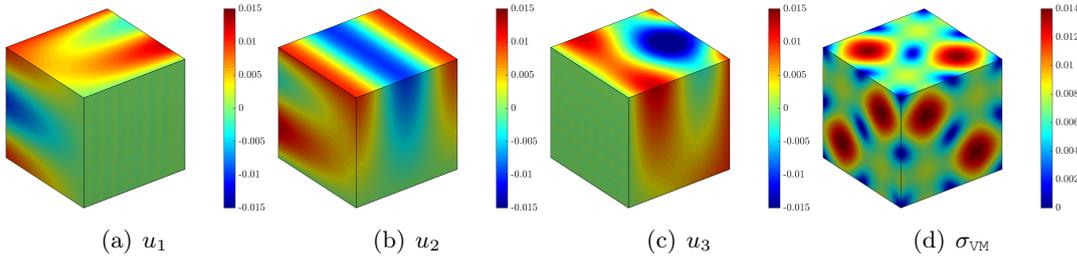
Figure 2.5: *Three dimensional problem: HDG approximation of the displacement field and the Von Mises stress using the fourth hexahedral mesh and $k = 3$.*

2.4.2   Superconvergence of the displacement field

In this section, the three post-process procedures described in Section 2.3 are tested using numerical examples. It is worth recalling that the three post-process options differ in the condition used to remove the indeterminacy related to the rigid rotational modes.

Two dimensional example

The different post-process techniques are applied to the two dimensional example of Section 2.4.1.

The first post-process considers the condition of Equation (2.40). The convergence of the error of the post-processed variable $\boldsymbol{u}^\star$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.7 for both quadrilateral and triangular elements and for a degree of approximation ranging from $k = 1$ up to $k = 3$. The results indicate that, as other HDG methods for linear elasticity [63], superconvergence of the post-processed solution is obtained for $k \geq 2$. When a linear approximation is used, quadrilateral elements show almost optimal convergence but for triangular elements a sub-optimal rate of 2.4 is observed.

Comparing the errors of the post-processed solution to the errors of the HDG solution in Figure 2.3, it is apparent that, despite no superconvergent results are provided by the first post-process technique, the post-processed solution is substantially more

(a) Hexahedrons

(b) Tetrahedrons

(c) Prisms
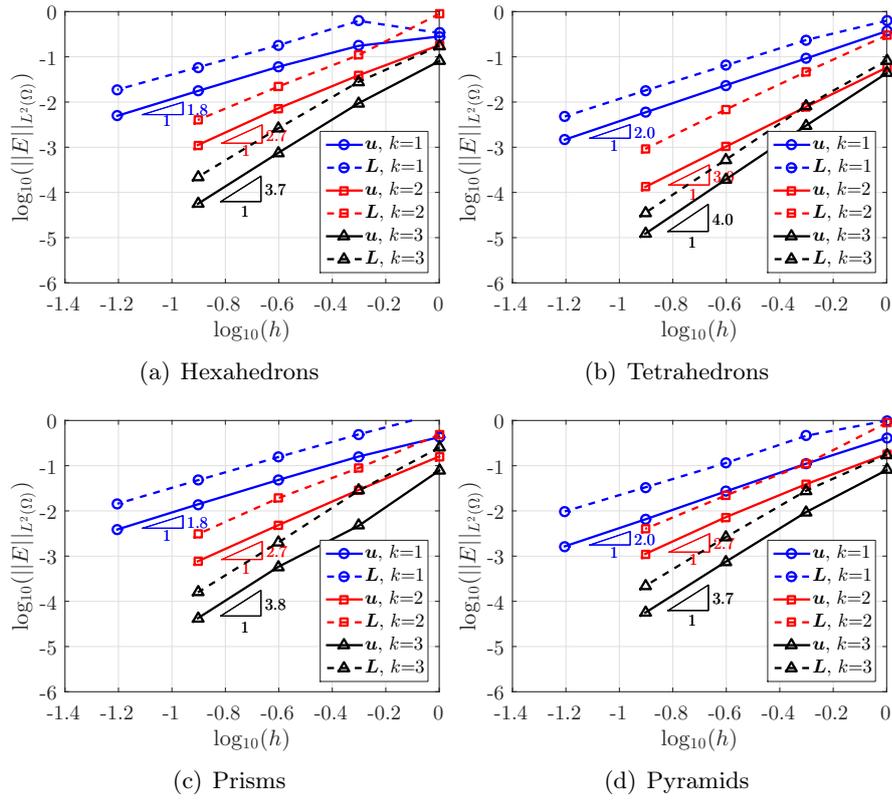
(d) Pyramids

Figure 2.6: *Three dimensional problem: h-convergence of the error of the primal and mixed variables, $\boldsymbol{u}$ and $\boldsymbol{L}$ in the $\mathcal{L}_2(\Omega)$ norm for hexahedral, tetrahedral, prismatic and pyramidal meshes with different orders of approximation.*

(a) Quadrilaterals

(b) Triangles
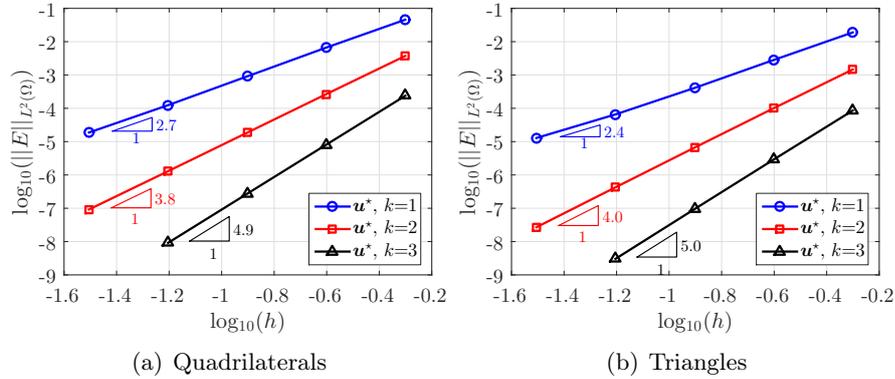
Figure 2.7: *Two dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for quadrilateral and triangular meshes with different orders of approximation using the post-process technique of Equation (2.40).*

accurate than the HDG solution for both quadrilateral and triangular elements. It is worth noting that this post-process was utilised in a different HDG formulation of the linear elastic problem for linear triangles [63] and sub-optimal convergence was also observed.

Next, the post-process that considers the condition of Equation (2.41) is tested.

(a) Quadrilaterals (b) Triangles
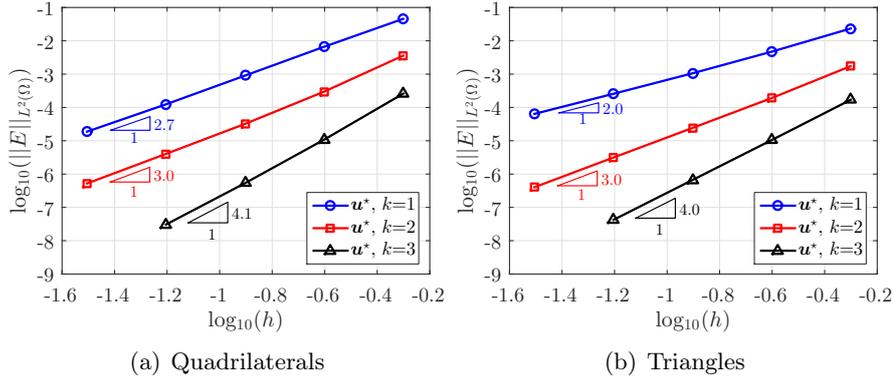
Figure 2.8: *Two dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for quadrilateral and triangular meshes with different orders of approximation using the post-process technique of Equation* (2.41).

(a) Quadrilaterals (b) Triangles
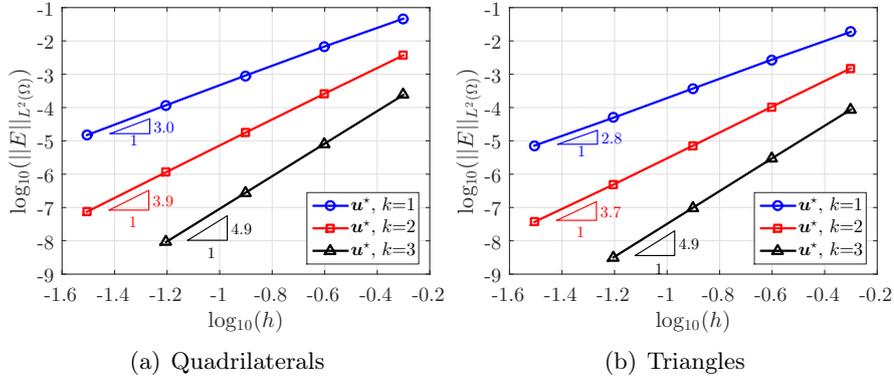
Figure 2.9: *Two dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for quadrilateral and triangular meshes with different orders of approximation using the post-process technique of Equation* (2.42).

Figure 2.8 shows the convergence study for the error of the post-processed variable $\boldsymbol{u}^\star$ measured in the $\mathcal{L}_2(\Omega)$ norm. The results for quadrilateral elements are almost identical to the results obtained with the first technique, whereas, for triangles, a sub-optimal order $k+1$ is observed for all degrees of approximation.

Comparing the errors of the post-processed solution with triangles to the errors of the HDG solution in Figure 2.3, it is apparent that little gain in accuracy is obtained with the post-processed solution. This is crucial when the superconvergent solution is sought to devise automatic degree adaptive processes [95, 155] and suggests that the post-process provided by the second option cannot be used to produce an accurate error estimator with triangles.

The last post-process technique proposed in this chapter is considered, consisting of imposing the condition of Equation (2.42). The convergence of the error of the post-processed variable $\boldsymbol{u}^\star$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.9. The results reveal that almost the optimal rate of convergence is attained for both quadrilateral and triangular elements and for
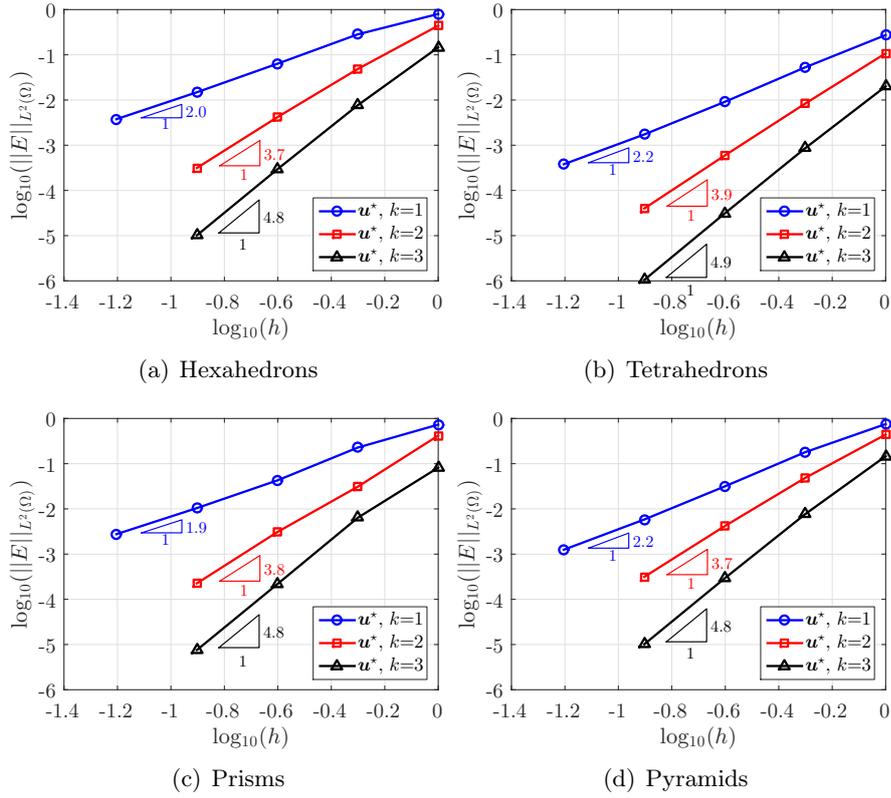
Figure 2.10: *Three dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for hexahedral, tetrahedral, prismatic and pyramidal meshes with different orders of approximation using the post-process technique of Equation* (2.40).

all degrees of approximation. This indicates that the average of the hybrid variable on the boundary leads to superconvergent results. It is worth noting that the error of the post-processed solution obtained with the third post-process technique, proposed here, is not only showing the optimal rate but it also provides an extra gain in accuracy when compared to the first post-process technique, previously used in an HDG context.

Three dimensional example

The different post-process techniques are considered in the three dimensional example of Section 2.4.1.

The convergence of the error of the post-processed variable $\boldsymbol{u}^{\star}$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.10 when using the post-process technique of Equation (2.40). The results reveal that superconvergent results are obtained with $k \geq 2$ whereas with linear elements sub-optimal convergence is attained. It is worth noting that in the two dimensional example almost superconvergent results where obtained with quadrilateral elements whereas in three dimensions sub-optimal convergence of order $k+1$ is observed for all the different element types considered.

(a) Hexahedrons   (b) Tetrahedrons
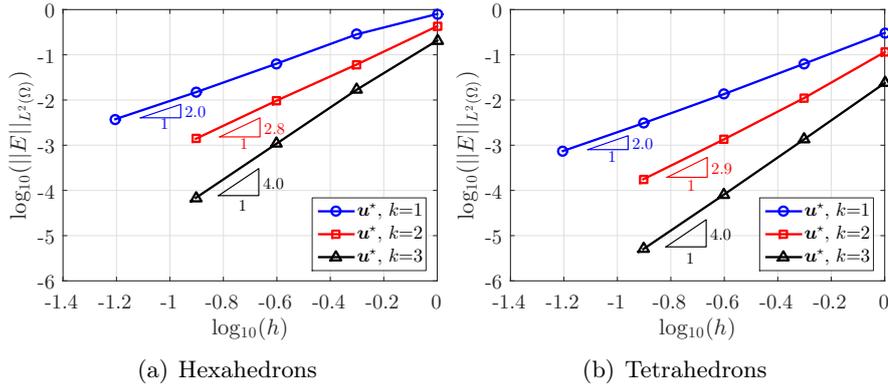
Figure 2.11: *Three dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for hexahedral and tetrahedral meshes with different orders of approximation using the post-process technique of Equation* (2.41)*.*

Next, the post-process that considers the condition of Equation (2.41) is tested. Figure 2.11 shows the convergence study for the error of the post-processed variable $\boldsymbol{u}^\star$ measured in the $\mathcal{L}_2(\Omega)$ norm. Only the results for hexahedral and tetrahedral elements are reported in Figure 2.11 because, analogously to the two dimensional example, this post-process leads to a sub-optimal rate $k + 1$ for all the different elements types and degrees of approximation.

Finally, the last post-process technique proposed in this section is considered. The convergence of the error of the post-processed variable $\boldsymbol{u}^\star$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is represented in Figure 2.12. The results show that almost the optimal rate of convergence is attained for all the element types and for all degrees of approximation considered. The numerical experiments performed in two and three dimensions confirm that the post-process technique proposed in this chapter for the first time lead to optimal superconvergent results of the primal variable.

As in the two dimensional example, the error of the post-processed solution obtained with the third technique, proposed here, is not only showing the optimal rate but it also provides an extra gain in accuracy when compared to the first post-process technique, previously used in an HDG context. When compared to the error of the HDG solution, represented in Figure 2.6, the post-process proposed here provides a solution that is almost one order of magnitude more accurate than the HDG solution, even for linear approximation of the solution.

2.4.3   Influence of the stabilisation parameter

The stabilisation tensor $\boldsymbol{\tau}$ is known to have an important effect on the stability, accuracy and convergence properties of the resulting HDG method [21, 91, 63]. This section presents a numerical study to assess the influence of the stabilisation parameter on the
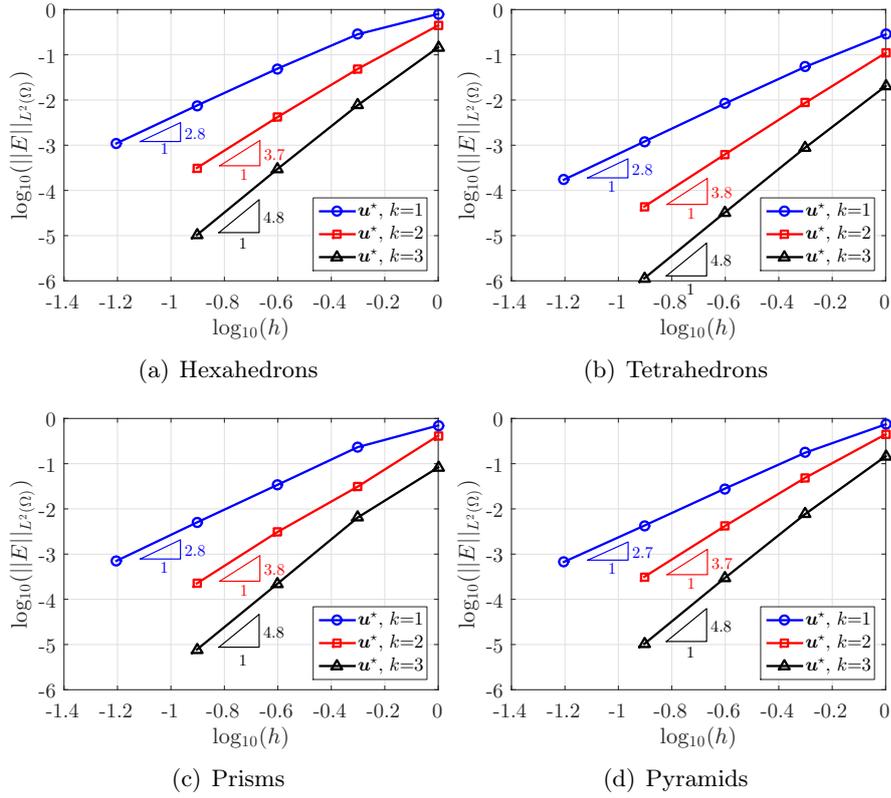
Figure 2.12: *Three dimensional problem: h-convergence of the error of the post-processed solution in the $\mathcal{L}_2(\Omega)$ norm for hexahedral, tetrahedral, prismatic and pyramidal meshes meshes with different orders of approximation using the post-process technique of Equation* (2.42).

accuracy of the results. For simplicity, it is assumed that $\boldsymbol{\tau} = \tau \mathbf{I}_{\mathrm{n_{sd}}}$ and the influence of the scalar stabilisation parameter τ is investigated.

Two dimensional example

Figure 2.13 shows the evolution of the error of the primal, mixed and post-processed variables, $\boldsymbol{u}$, $\boldsymbol{L}$ and $\boldsymbol{u}^\star$ respectively, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilisation parameter τ for the two dimensional example studied in Section 2.4.1. The numerical experiment is performed using linear and quadratic approximations and for quadrilateral and triangular meshes and the value of τ varies from 0.1 to 1,000.

The results reveal that there is a value of τ for which the error of the primal solution is minimum. For both quadrilateral and triangular meshes with linear and quadratic approximation, this value is near $\tau = 10$. However, it is worth noting that for $\tau = 10$ the post-process of the displacement field offers little or no extra gain in accuracy. When the error of the mixed variable is of interest, the minimum error is achieved for a different value of the stabilisation parameter, near $\tau = 3$. It is worth noting that the value of $\tau = 3$ also provides the best accurate results for the post-processed variable. As a result, the value of $\tau = 3$ is considered the optimum value in this experiment as it

(a) Quadrilaterals   (b) Triangles
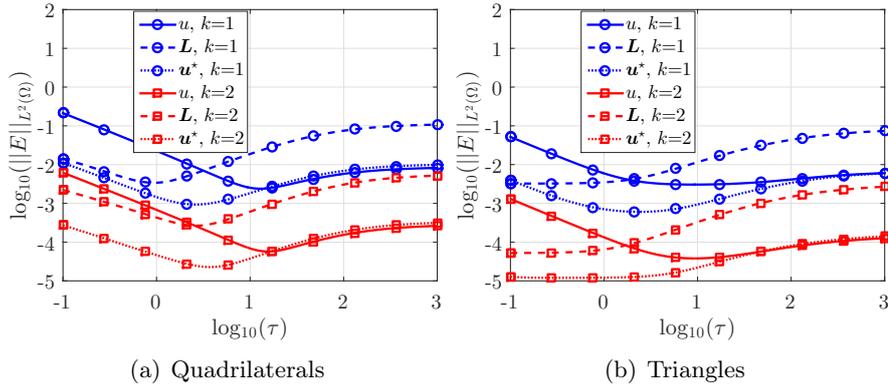
Figure 2.13: *Two dimensional problem: error of the primal, mixed and post-processed variables, u, L and $u^\star$ respectively, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilisation parameter τ.*

provides the most accurate solution for both the displacement (i.e. the post-processed variable) and the stress (i.e. the mixed variable). It is also interesting to observe that for $\tau = 3$ the accuracy on the primal and mixed variables is almost identical.

Three dimensional example

A similar study is performed next for the three dimensional example of Section 2.4.1. Figure 2.14 shows the evolution of the error of the primal, mixed and post-processed variables, u, L and $u^\star$ respectively, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilisation parameter τ. As in the two dimensional example a value of the stabilisation parameter near $\tau = 10$ provides the minimum error for the primal variable but with no extra gain in accuracy when the post-process is performed. Also, as in the two dimensional case, the value of τ that provides the most accurate results for both the mixed and the post-processed variables is near $\tau = 3$. It is worth noting that for hexahedral, prismatic and pyramidal elements, and contrary to the results obtained in the two dimensional problem, the value that provides the most accurate results for the primal and mixed variable is almost identical.

The conclusions that are extracted from this study are similar to the ones obtained in the two dimensional example and show that the optimal value of the stabilisation parameter is not dependent upon the degree of approximation, the type of element or the dimensionality of the problem.

2.4.4   Locking-free behaviour in the incompressible limit

The last example considers a problem with a nearly incompressible material (i.e. $\nu \sim 0.5$) that is commonly used in the literature [63]. The problem, defined in $\Omega = [0,1]^2$,

(a) Hexahedrons

(b) Tetrahedrons

(c) Prisms

(d) Pyramids
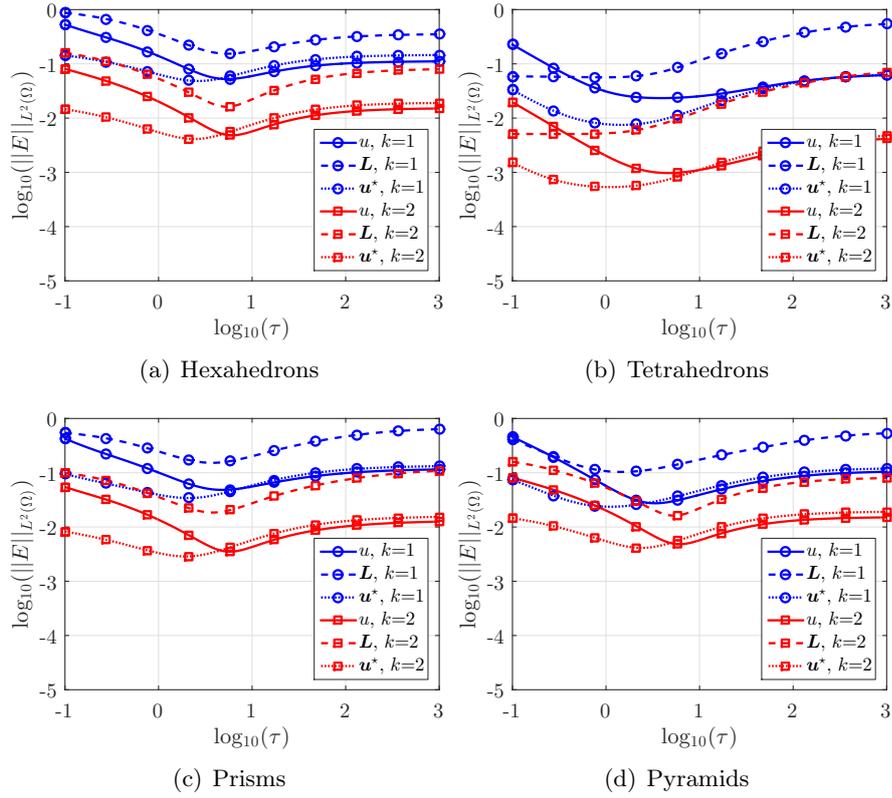
Figure 2.14: *Three dimensional problem: error of the primal, mixed and post-processed variables in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilisation parameter.*

has analytical solution given by

$$\boldsymbol{u}(\boldsymbol{x}) = \begin{Bmatrix} -x_1^2 x_2 (x_1 - 1)^2 (x_2 - 1)(2x_2 - 1) \\ x_2^2 x_1 (x_2 - 1)^2 (x_1 - 1)(2x_1 - 1) \end{Bmatrix}. \tag{2.46}$$

The external load and boundary conditions are derived from the exact solution. The Young's modulus is taken as $E = 3$ and the Poisson's ratio is varied from $\nu = 0.49$ up to $\nu = 0.49999$.

Only triangular meshes with the arrangement represented in Figure 2.15 are shown as this particular arrangement is known to exhibit a volumetric locking effect when considered with a traditional continuous Galerkin finite element formulation. Note that other the arrangements depicted in Figure 2.1 also produce optimal rates of convergence.

The displacement field and the Von Mises stress computed on the fourth triangular mesh and using a cubic degree of approximation are depicted in Figure 2.16.

The mesh convergence results for the primal and mixed variables, $\boldsymbol{u}$ and $\boldsymbol{L}$, are represented in Figure 2.17 for the triangular meshes shown in Figure 2.15, for different orders of approximation and for increasing value of the Poisson's ratio. The results show that the proposed HDG formulation is volumetric locking-free. In addition, it

(a) Mesh 1   (b) Mesh 2   (c) Mesh 3
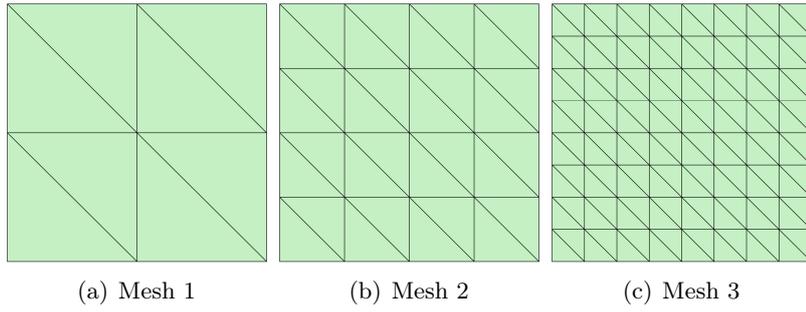
Figure 2.15: *Three two dimensional triangular meshes of $\Omega = [0,1]^2$ for the mesh convergence study with a nearly incompressible material.*

(a) u_1 (b) u_2 (c) σ_{VM}
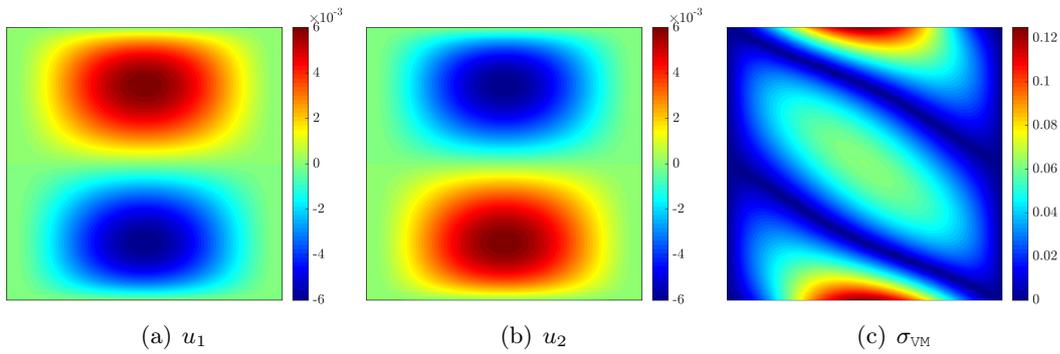
Figure 2.16: *HDG approximation of the displacement field and the Von Mises stress using the fourth triangular mesh and $k = 3$ for a material with $\nu = 0.49999$.*

is worth noting that the accuracy of the displacement and the stress is almost independent on the Poisson's ratio. This behaviour has also been observed when using a different HDG formulation [63]. However, contrary to the results reported in [63], the proposed formulation shows the optimal rate of convergence, whereas the formulation in [63] exhibits a slight degradation of the rate of convergence for nearly incompressible materials. This degradation of the rate of convergence in the HDG formulation of [63] is sizeable when the error on the mixed variable is considered, even for high-order approximations.

Next, the mesh convergence study is performed for the post-processed variable using the technique proposed in this chapter that resulted in optimal convergence in the numerical example of Section 2.4.2. The mesh convergence results for the post-processed displacement field $\boldsymbol{u}^\star$, are represented in Figure 2.18 for different orders of approximation and for increasing value of the Poisson's ratio. The results show again that the accuracy is independent on the Poisson's ratio. More important, the mesh convergence study demonstrates that the proposed formulation together with the proposed post-process is able to provide superconvergent solutions for all degrees of approximation, even for linear triangular elements in the particular arrangement that causes volumetric locking in a continuous Galerkin formulation.
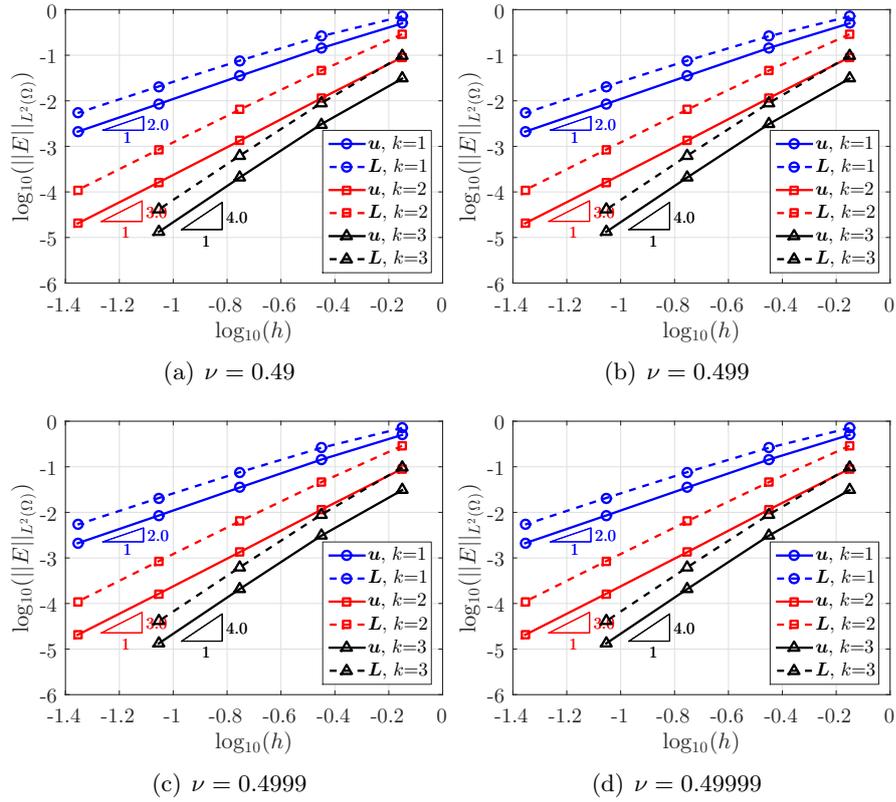
Figure 2.17: *h-convergence of the error of the primal and mixed variables, **u** and **L**, in the $\mathcal{L}_2(\Omega)$ norm for different orders of approximation and for an increasing value of the Poisson's ratio.*

2.5   Conclusion

In this chapter, a novel HDG formulation for the linear elastic problem strongly enforcing the symmetry of the stress tensor is proposed. Owing to the Voigt notation, the second-order tensors appearing in the linear elasticity equation are expressed as vectors featuring the diagonal and half of the off-diagonal terms. Thus, the resulting method does not introduce any extra cost to guarantee the symmetry of the stress tensor and, in fact, it is more computationally efficient than other HDG formulations due to the reduced number of degrees of freedom of the mixed variable.

For $k \geq 1$, as all existing HDG formulations for linear elasticity, the resulting method provides optimal convergence rate of order $k + 1$ for the displacement field. The optimal order $k+1$ is also obtained for the stress tensor which usually experiences sub-optimal behaviour using low-order approximations in the original HDG formulation by Cockburn and co-workers. Furthermore, contrary to other proposed variants of HDG for linear elasticity, the optimality is achieved using equal-order approximation spaces for the primal and mixed variables and no special enrichment is required.

The optimally convergent stress tensor is thus utilized to locally construct a post-processed displacement field. The element-by-element procedure uses the equilibrated
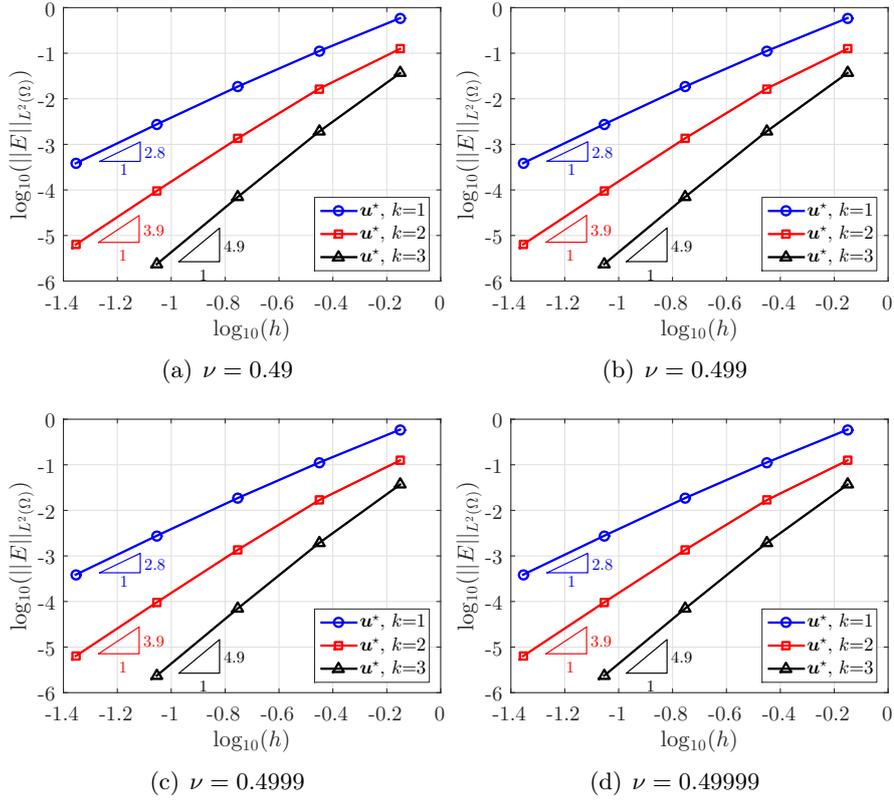
Figure 2.18: *h-convergence of the error of the post-processed variable, $\boldsymbol{u}^\star$, in the $\mathcal{L}_2(\Omega)$ norm for different orders of approximation and for an increasing value of the Poisson's ratio.*

stresses as boundary conditions of the local problems and exploits the optimal convergence of the trace of the displacements to remove the underdetermination associated with the rigid rotational modes. The post-processed displacement field belongs to the richer space of polynomials of degree at most $k+1$ in each element and superconverges with order $k+2$. Therefore, the current formulation provides a workaround to avoid the construction of discrete spaces fulfilling the $\boldsymbol{M}$-decomposition property to guarantee the superconvergence of the post-processed solution.

An extensive set of numerical simulations has been presented to verify the optimal approximation properties of the method in 2D and 3D, to study the influence of the HDG stabilisation parameter and to show the robustness of the formulation using meshes featuring different element shapes, commonly implemented in commercial softwares. Special attention has been dedicated to the analysis of the limit case of nearly incompressible materials: numerical evidence shows that the method is locking-free and the optimal convergence and superconvergence rates of the primal, mixed and post-processed variables are preserved.

Given the demonstrated optimal convergence and superconvergence rates near the incompressibility limit, as a next step the approach, which was introduced in this chapter for the linear elasticity problem, will be applied to the Stokes flow. The Stokes flow

describes the motion of incompressible, highly viscous fluids and the Stokes problem has a similar structure to the almost incompressible linear elasticity problem. Therefore, in Chapter 3 the hybridizable discontinuous Galerkin method with the strong enforcement of the stress tensor for the Stokes flow will be presented.

Chapter 3

Superconvergent HDG method for Stokes flow [1]

The Stokes flow describes the movement of an incompressible fluid with extremely high viscosity, a flow with a very small velocity, usually referred to as creeping flow, as well as a flow at small length scales, for example some μm. Under these circumstances, inertial forces are extremely smaller than viscous forces inside the fluid, therefore, viscosity has a predominant role and the convection term can be neglected. Studying the Stokes flow is broadly used as a preliminary step before studying the Navier-Stokes equations, that do account for convection. In addition, as far as the proposed formulation of the HDG method in this thesis is concerned, applying it to the Stokes flow is reasonable, especially after demonstrating the excellent convergence properties in nearly incompressible elastic materials. In this chapter, a superconvergent HDG method for the solution of the Stokes problem is presented. A similar approach to the aforementioned linear elasticity problem will be followed.

This approach considers the same degree of polynomials for all variables, either primal or mixed. The symmetry of the stress tensor is enforced explicitly. Then, with the use of the Voigt notation, the governing equations of the problem are rewritten considering a symmetric stress tensor. As previously mentioned for the linear elasticity problem, this approach remedies the suboptimal convergence behaviour of the mixed variable for low-order approximations. The optimal convergence rates of the mixed variable are obtained, thereby enabling the construction of an element-by-element post-process procedure that leads to a superconvergent velocity field, even for low-order approximations. Moreover, no additional enrichment of the discrete spaces is required and a gain in computational efficiency follows from reducing the quantity of stored information and the size of the local problems. Finally, using this formulation it is possible to impose directly physical tractions on the boundaries, without necessitating the imposition of pseudo-tractions. This is the case for formulations that are based on a different set of variables and do not consider the symmetry of the stress tensor.

---

[1]This section is a modified version of the published article: M. Giacomini, A. Karkoulias, R. Sevilla, A. Huerta, A superconvergent HDG for Stokes flow with strongly enforced symmetry of the stress tensor, Journal of Scientific Computing 77 (3) (2018) 1679-1702 [154].

This chapter is organized as follows. In Section 3.1, first, the equations governing the Stokes flow are recalled. Then, according to the rationale introduced in Chapter 2 for the linear elasticity equation, the symmetry of the stress tensor is strongly enforced by means of a well-known technique in the computational mechanics community, namely the Voigt notation for symmetric tensors. The corresponding Cauchy formulation of the Stokes equation with strongly enforced symmetry of the stress tensor is derived. In Section 3.2, the HDG discretization for the Stokes flow is introduced and a local post-process procedure providing a superconvergent velocity field is discussed. The proposed HDG formulation is validated in Section 3.3. Extensive analysis of the optimal convergence and superconvergence rates of the primal, mixed and post-processed variables, for two and three dimensional problems is provided by means of numerical simulations. Special emphasis is placed on the influence of the stabilization parameter and on the robustness of the method using meshes of different element types. Eventually, the capability of the method to accurately compute quantities of interest depending on the solution of the Stokes equation (e.g. the drag force) is discussed and Section 3.4 summarizes the results of this chapter.

3.1 Stokes flow with strongly enforced symmetry of the stress tensor

In this section, the framework to handle symmetric tensors by means of Voigt notation and the governing equations of a Stokes flow are introduced. First, the Cauchy formulation of the Stokes equation is recalled.

3.1.1 Cauchy formulation of the Stokes flow

Consider an open bounded domain $\Omega \subset \mathbb{R}^{n_{sd}}$ with boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$, $\Gamma_D \cap \Gamma_N = \emptyset$ and n_{sd} being the number of spatial dimensions. The strong form of the problem under analysis reads as follows:

$$\begin{cases} -\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{s} & \text{in } \Omega, \\ \boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 & \text{in } \Omega, \\ \boldsymbol{\sigma} = -p\mathbf{I}_{n_{sd}} + 2\nu\boldsymbol{\varepsilon}(\boldsymbol{u}) & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t} & \text{on } \Gamma_N, \end{cases} \tag{3.1}$$

where the pair $(\boldsymbol{u}, p)$ represents the velocity and pressure fields and $\boldsymbol{\sigma}$ is the Cauchy stress tensor. The terms $\boldsymbol{s}$, $\boldsymbol{u}_D$ and $\boldsymbol{t}$ respectively are the volumetric source term, the Dirichlet boundary datum to impose the value of the velocity on Γ_D and the traction applied on the Neumann boundary Γ_N. The third equation, known as Stokes law, provides the relationship between the stress tensor and the velocity and pressure

variables, through the viscosity coefficient $\nu > 0$, the $\mathrm{n_{sd}} \times \mathrm{n_{sd}}$ identity matrix $\mathbf{I}_{\mathrm{n_{sd}}}$ and the strain rate tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$, $\boldsymbol{\varepsilon}() := \frac{1}{2}\left(\boldsymbol{\nabla} + \boldsymbol{\nabla}^T\right)$ being the symmetric part of the gradient.

It is well-known that the Cauchy and the velocity-pressure formulations of the Stokes equation are equivalent from the variational point of view. Nevertheless, a major difference arises when considering the imposition of Neumann boundary conditions. On the one hand, natural boundary conditions for the Cauchy formulation enforce the value of the normal stress which represents a physical traction. On the other hand, the velocity-pressure formulation only accounts for the gradient of the velocity field instead of its symmetric part, leading to the imposition of the so-called *pseudo-tractions*. Hence, the physical interpretation is lost [156]. Within this context, an artificial handling of Neumann boundary conditions is required to impose physically meaningful tractions. This represents a drawback when dealing with real-life and industrial applications in which the enforcement of physically relevant quantities is a major constraint to perform reliable numerical simulations and compare them with experimental data.

3.1.2 Voigt notation for symmetric tensors

From the Stokes law in Equation (3.1), it is straightforward to observe that the Cauchy stress tensor is symmetric. It is worth noting that this property expresses a conservation law, namely the balance of angular momentum. As remarked in [39], the strong enforcement of this conservation law is not trivial and has lead to the development of the elegant, but rather complicated, framework of finite element exterior calculus to construct strongly symmetric approximations of second-order tensors [29]. In this section, the Voigt notation, that is widely used in the solid mechanics community and was previously presented for the linear elasticity problem, is exploited to simply enforce the symmetry of the stress tensor and, consequently, to fulfill the conservation of angular momentum pointwise. The Voigt notation relies on the idea of storing a second-order tensor in a vectorial format by appropriately rearranging its diagonal and off-diagonal components. Consequently, the application of differential operators (e.g. symmetric gradient, divergence and curl) and the geometrical projections (e.g. in the normal and tangential directions to a surface) may be expressed as matrix equations. For this purpose, the rationale for the construction of differential operator and geometrical quantities using Voigt notation is recalled.

Consider the previously defined strain rate tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$. Owing to its symmetry, only $\mathrm{m_{sd}} = \mathrm{n_{sd}}(\mathrm{n_{sd}} + 1)/2$ components (i.e. three in 2D and six in 3D) have to be stored. According to the arrangement presented in Chapter 2, the following column vector in $\mathbb{R}^{\mathrm{m_{sd}}}$ is obtained:

$$\boldsymbol{\varepsilon}_{\mathrm{V}} := \begin{cases} \left[e_{11},\ e_{22},\ e_{12}\right]^T & \text{in 2D,} \\ \left[e_{11},\ e_{22},\ e_{33},\ e_{12},\ e_{13},\ e_{23}\right]^T & \text{in 3D.} \end{cases} \tag{3.2}$$

The components of the strain rate in Equation (3.2) read as

$$e_{ij} := \frac{\partial u_i}{\partial x_j} + (1 - \delta_{ij})\frac{\partial u_j}{\partial x_i}, \quad \text{for } i, j = 1, \ldots, \mathrm{n_{sd}}, \tag{3.3}$$

where δ_{ij} is the classical Kronecker delta. In order to retrieve the aforementioned strain rate tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$, the off-diagonal terms e_{ij}, $i \neq j$ have to be multiplied by a factor $1/2$, namely

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) := \begin{cases} \begin{bmatrix} e_{11} & e_{12}/2 \\ e_{12}/2 & e_{22} \end{bmatrix} & \text{in 2D,} \\ \begin{bmatrix} e_{11} & e_{12}/2 & e_{13}/2 \\ e_{12}/2 & e_{22} & e_{23}/2 \\ e_{13}/2 & e_{23}/2 & e_{33} \end{bmatrix} & \text{in 3D.} \end{cases} \tag{3.4}$$

Similarly, the symmetry of the stress tensor $\boldsymbol{\sigma}$ is exploited to store only $\mathrm{m_{sd}}$ components in the column vector

$$\boldsymbol{\sigma}_{\mathrm{V}} := \begin{cases} \begin{bmatrix} \sigma_{11}, & \sigma_{22}, & \sigma_{12} \end{bmatrix}^T & \text{in 2D,} \\ \begin{bmatrix} \sigma_{11}, & \sigma_{22}, & \sigma_{33}, & \sigma_{12}, & \sigma_{13}, & \sigma_{23} \end{bmatrix}^T & \text{in 3D.} \end{cases} \tag{3.5}$$

Differential operators using Voigt notation

Following [150], the strain rate tensor can be written as $\boldsymbol{\varepsilon}_{\mathrm{V}} = \boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}$ by introducing the $\mathrm{m_{sd}} \times \mathrm{n_{sd}}$ matrix $\boldsymbol{\nabla}_{\mathrm{S}}$ of Equation (2.8). In a similar fashion, the vorticity $\boldsymbol{\omega} := \boldsymbol{\nabla} \times \boldsymbol{u}$ may be expressed in terms of Voigt notation as $\boldsymbol{\omega} = \mathbf{R}\boldsymbol{u}$ through the $\mathrm{n_{rr}} \times \mathrm{n_{sd}}$ matrix $\mathbf{R}$ of Equation (2.16).

3.1.3   Cauchy formulation of the Stokes flow using Voigt notation

Owing to the notation introduced in this section, the Stokes constitutive law may be expressed as $\boldsymbol{\sigma}_{\mathrm{V}} = -\mathbf{E}p + \mathbf{D}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}$, where the vector $\mathbf{E} \in \mathbb{R}^{\mathrm{m_{sd}}}$ and the matrix $\mathbf{D} \in \mathbb{R}^{\mathrm{m_{sd}} \times \mathrm{m_{sd}}}$ read as

$$\mathbf{E} := \begin{cases} \begin{bmatrix} 1, & 1, & 0 \end{bmatrix}^T & \text{in 2D,} \\ \begin{bmatrix} 1, & 1, & 1, & 0, & 0, & 0 \end{bmatrix}^T & \text{in 3D.} \end{cases} \qquad \mathbf{D} := \begin{cases} \begin{bmatrix} 2\nu \mathbf{I}_{\mathrm{n_{sd}}} & \mathbf{0}_{\mathrm{n_{sd}} \times 1} \\ \mathbf{0}^T_{\mathrm{n_{sd}} \times 1} & \nu \end{bmatrix} & \text{in 2D,} \\ \begin{bmatrix} 2\nu \mathbf{I}_{\mathrm{n_{sd}}} & \mathbf{0}_{\mathrm{n_{sd}}} \\ \mathbf{0}_{\mathrm{n_{sd}}} & \nu \mathbf{I}_{\mathrm{n_{sd}}} \end{bmatrix} & \text{in 3D.} \end{cases} \tag{3.6}$$

Moreover, the Neumann boundary condition applied on Γ_N can be written as $\mathbf{N}^T\boldsymbol{\sigma}_{\mathrm{V}} = \boldsymbol{t}$ by introducing the $\mathrm{m_{sd}} \times \mathrm{n_{sd}}$ matrix $\mathbf{N}$ of Equation (2.12), accounting for the normal direction to the boundary.

Similarly, the projection of a vector along the tangential direction $\boldsymbol{\tau}$, namely a tangent line in 2D and a tangent surface in 3D, reads as $\boldsymbol{u} \cdot \boldsymbol{\tau} = \mathbf{T}\boldsymbol{u}$, where $\mathbf{T}$ is defined in Equation (2.18).

In order to rewrite Equation (3.1) using Voigt notation, the divergence of a symmetric tensor is expressed in terms of the transpose of the matrix $\boldsymbol{\nabla}_{\mathrm{S}}$ accounting for the symmetric part of the gradient [150]. In a similar fashion, recall that $\boldsymbol{\nabla} \cdot \boldsymbol{u} = \mathrm{tr}(\boldsymbol{\nabla}\boldsymbol{u})$ and observe that the trace operator may be expressed via the vector $\mathbf{E}$ introduced in Equation (3.6). Combining the matrix forms of the symmetric gradient, the Stokes law and the normal direction presented above, the following formulation of the Stokes equation using Voigt notation is obtained:

$$
\begin{cases}
-\boldsymbol{\nabla}_{\mathrm{S}}^{T}\boldsymbol{\sigma}_{\mathrm{V}} = \boldsymbol{s} & \text{in } \Omega, \\
\mathbf{E}^{T}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = 0 & \text{in } \Omega, \\
\boldsymbol{\sigma}_{\mathrm{V}} = -\mathbf{E}p + \mathbf{D}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} & \text{in } \Omega, \\
\boldsymbol{u} = \boldsymbol{u}_{D} & \text{on } \Gamma_{D}, \\
\mathbf{N}^{T}\boldsymbol{\sigma}_{\mathrm{V}} = \boldsymbol{t} & \text{on } \Gamma_{N}.
\end{cases}
\tag{3.7}
$$

3.2 A hybridizable discontinuous Galerkin method

As presented in Section 2.2, a similar approach is followed and the second-order problem in Equation (3.7) may be written as a system of first-order equations as follows:

$$
\begin{cases}
\boldsymbol{L} + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = \boldsymbol{0} & \text{in } \Omega_{e}, \text{ and for } e = 1, \dots, \mathtt{n_{el}}, \\
\boldsymbol{\nabla}_{\mathrm{S}}^{T}\left(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p\right) = \boldsymbol{s} & \text{in } \Omega_{e}, \text{ and for } e = 1, \dots, \mathtt{n_{el}}, \\
\mathbf{E}^{T}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = 0 & \text{in } \Omega_{e}, \text{ and for } e = 1, \dots, \mathtt{n_{el}}, \\
\boldsymbol{u} = \boldsymbol{u}_{D} & \text{on } \Gamma_{D}, \\
\mathbf{N}^{T}(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p) = -\boldsymbol{t} & \text{on } \Gamma_{N}, \\
[\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \boldsymbol{0} & \text{on } \Gamma, \\
[\![\mathbf{N}^{T}(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p)]\!] = \boldsymbol{0} & \text{on } \Gamma,
\end{cases}
\tag{3.8}
$$

where the last two equations are the *transmission conditions* enforcing the continuity of respectively the velocity and the flux across the interface Γ.

Remark 8. In the case of purely Dirichlet boundary conditions (i.e. $\Gamma_{N} = \emptyset$), an additional constraint is required to avoid the indeterminacy of the pressure. A common choice relies on imposing zero mean value of the pressure on the boundary (cf. e.g. [92, 157, 24]):

$$
\frac{1}{|\partial\Omega|}\langle p, 1\rangle_{\partial\Omega} = 0.
\tag{3.9}
$$

3.2.1   Strong form of the local and global problems

In a series of papers by Cockburn and co-workers [92, 24, 81, 93], the hybridizable discontinuous Galerkin formulation for Stokes flow has been theoretically and numerically analyzed. Starting from the mixed formulation on the broken computational domain in Equation (3.8), HDG features two stages.

First, a set of $\mathtt{n}_{\mathtt{el}}$ local problems are defined element-by-element to compute $(\boldsymbol{L}_e, \boldsymbol{u}_e, p_e)$ for $e = 1, \ldots, \mathtt{n}_{\mathtt{el}}$:

$$\begin{cases} \boldsymbol{L}_e + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = \mathbf{0} & \text{in } \Omega_e \\ \boldsymbol{\nabla}_{\mathrm{S}}^T\mathbf{D}^{1/2}\boldsymbol{L}_e + \boldsymbol{\nabla}_{\mathrm{S}}^T\mathbf{E}\,p_e = \boldsymbol{s} & \text{in } \Omega_e \\ \mathbf{E}^T\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = 0 & \text{in } \Omega_e \\ \boldsymbol{u}_e = \boldsymbol{u}_D & \text{on } \partial\Omega_e \cap \Gamma_D, \\ \boldsymbol{u}_e = \widehat{\boldsymbol{u}} & \text{on } \partial\Omega_e \setminus \Gamma_D, \end{cases} \tag{3.10}$$

where $\widehat{\boldsymbol{u}}$ is an independent variable representing the trace of the velocity on the mesh skeleton $\Gamma \cup \Gamma_N$. Remark that Equation (3.10) is a purely Dirichlet boundary value problem. As previously observed, an additional constraint has to be added to remove the indeterminacy of the pressure, namely

$$\frac{1}{|\partial\Omega_e|}\langle p_e, 1\rangle_{\partial\Omega_e} = \rho_e, \tag{3.11}$$

where ρ_e denotes the mean pressure on the boundary of the element Ω_e. Hence, for $e = 1, \ldots, \mathtt{n}_{\mathtt{el}}$ the local problem in Equation (3.10) provides $(\boldsymbol{L}_e, \boldsymbol{u}_e, p_e)$ in terms of the global unknowns $\widehat{\boldsymbol{u}}$ and ρ.

The trace of the velocity $\widehat{\boldsymbol{u}}$ and the mean pressure ρ on the element boundaries are determined by solving the global problem accounting for the transmission conditions and the Neumann boundary condition:

$$\begin{cases} [\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \mathbf{0} & \text{on } \Gamma, \\ [\![\mathbf{N}^T(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}\,p)]\!] = \mathbf{0} & \text{on } \Gamma, \\ \mathbf{N}^T(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}\,p) = -\boldsymbol{t} & \text{on } \Gamma_N. \end{cases} \tag{3.12}$$

The first equation is automatically satisfied due to the Dirichlet boundary condition $\boldsymbol{u}_e = \widehat{\boldsymbol{u}}$ imposed in the local problems and the unique definition of the hybrid variable $\widehat{\boldsymbol{u}}$ on each face of the mesh skeleton. Moreover, the divergence-free condition in the local problem induces the following compatibility condition for each element Ω_e, $e = 1, \ldots, \mathtt{n}_{\mathtt{el}}$

$$\langle\widehat{\boldsymbol{u}} \cdot \boldsymbol{n}_e, 1\rangle_{\partial\Omega_e\setminus\Gamma_D} + \langle\boldsymbol{u}_D \cdot \boldsymbol{n}_e, 1\rangle_{\partial\Omega_e\cap\Gamma_D} = 0. \tag{3.13}$$

Consider the Voigt counterpart $\mathbf{E}^T \boldsymbol{\nabla}_{\text{S}} \boldsymbol{u}_e = 0$ of the aforementioned constraint (cf. Equation (3.10)). The resulting compatibility condition reads as

$$\langle \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}, 1 \rangle_{\partial\Omega_e \backslash \Gamma_D} + \langle \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D, 1 \rangle_{\partial\Omega_e \cap \Gamma_D} = 0 \quad \text{for } e = 1, \dots, \mathtt{n}_{\mathtt{el}} \tag{3.14}$$

and it is utilized to close the global problem.

3.2.2 Weak form of the local and global problems

Consider the discrete functional spaces in Equation (2.26). The discrete weak formulation of the local problems in Equation (3.10) is as follows: for $e = 1, \dots, \mathtt{n}_{\mathtt{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}^h$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h, p_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m}_{\mathtt{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n}_{\mathtt{sd}}} \times \mathcal{V}^h(\Omega_e)$ such that

$$\begin{aligned} -(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\text{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} \\ = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D}, \end{aligned} \tag{3.15a}$$

$$\begin{aligned} -(\boldsymbol{\nabla}_{\text{S}} \boldsymbol{w}, \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} - (\mathbf{E}^T \boldsymbol{\nabla}_{\text{S}} \boldsymbol{w}, p_e^h)_{\Omega_e} \\ + \langle \boldsymbol{w}, \mathbf{N}_e^T (\widehat{\mathbf{D}^{1/2} \boldsymbol{L}_e^h + \mathbf{E} \, p_e^h}) \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{s})_{\Omega_e}, \end{aligned} \tag{3.15b}$$

$$(\boldsymbol{\nabla}_{\text{S}}^T \mathbf{E} \, q, \boldsymbol{u}_e^h)_{\Omega_e} = \langle q, \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle q, \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D}, \tag{3.15c}$$

$$\frac{1}{|\partial\Omega_e|} \langle p_e^h, 1 \rangle_{\partial\Omega_e} = \rho_e^h, \tag{3.15d}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, q) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m}_{\mathtt{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n}_{\mathtt{sd}}} \times \mathcal{V}^h(\Omega_e)$. Following the denifition of the trace of numerical stress in Equation (2.29) for the linear elasticity, the trace of the numerical normal flux in Equation (3.15b) is defined as follows

$$\mathbf{N}_e^T (\widehat{\mathbf{D}^{1/2} \boldsymbol{L}_e^h + \mathbf{E} \, p_e^h}) := \begin{cases} \mathbf{N}_e^T (\mathbf{D}^{1/2} \boldsymbol{L}_e^h + \mathbf{E} \, p_e^h) + \boldsymbol{\tau}(\boldsymbol{u}_e^h - \boldsymbol{u}_D) & \text{on } \partial\Omega_e \cap \Gamma_D, \\ \mathbf{N}_e^T (\mathbf{D}^{1/2} \boldsymbol{L}_e^h + \mathbf{E} \, p_e^h) + \boldsymbol{\tau}(\boldsymbol{u}_e^h - \widehat{\boldsymbol{u}}^h) & \text{elsewhere,} \end{cases} \tag{3.16}$$

where the stabilization parameter $\boldsymbol{\tau}$ plays a crucial role in the stability, accuracy and convergence properties of the resulting HDG method [21, 23, 151]. By plugging Equation (3.16) into Equation (3.15b) and integrating by parts, the symmetric form of the discrete weak local problem is obtained: for $e = 1, \dots, \mathtt{n}_{\mathtt{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}^h$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h, p_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m}_{\mathtt{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n}_{\mathtt{sd}}} \times \mathcal{V}^h(\Omega_e)$ that satisfy

$$\begin{aligned} -(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\text{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} \\ = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D}, \end{aligned} \tag{3.17a}$$

$$\begin{aligned} (\boldsymbol{w}, \boldsymbol{\nabla}_{\text{S}}^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} + \langle \boldsymbol{w}, \boldsymbol{\tau} \boldsymbol{u}_e^h \rangle_{\partial\Omega_e} + (\boldsymbol{w}, \boldsymbol{\nabla}_{\text{S}}^T \mathbf{E} \, p_e^h)_{\Omega_e} \\ = (\boldsymbol{w}, \boldsymbol{s})_{\Omega_e} + \langle \boldsymbol{w}, \boldsymbol{\tau} \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle \boldsymbol{w}, \boldsymbol{\tau} \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D}, \end{aligned} \tag{3.17b}$$

$$(\boldsymbol{\nabla}_{\text{S}}^T \mathbf{E} \, q, \boldsymbol{u}_e^h)_{\Omega_e} = \langle q, \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle q, \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D}, \tag{3.17c}$$

$$\frac{1}{|\partial\Omega_e|} \langle p_e^h, 1 \rangle_{\partial\Omega_e} = \rho_e^h, \tag{3.17d}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, q) \in [\mathcal{V}^h(\Omega_e)]^{\mathrm{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathrm{n_{sd}}} \times \mathcal{V}^h(\Omega_e)$.

Remark 9. From a practical point of view, the constraint on the mean value of the pressure on the boundary of the element introduced in Equation (3.17d) is handled by means of a Lagrange multiplier. Thus, the matrix associated with the resulting local problem has a saddle point structure [155].

Remark 10. Following the notation used in [152, 155, 146], an isoparametric discretization using equal interpolation for the primal and mixed variables is considered. The linear system associated with the discretization of the HDG local problem of Equation (3.17) has the following structure:

$$
\begin{bmatrix}
\mathbf{A}_{LL} & \mathbf{A}_{Lu} & \mathbf{0} & \mathbf{0} \\
\mathbf{A}_{Lu}^T & \mathbf{A}_{uu} & \mathbf{A}_{up} & \mathbf{0} \\
\mathbf{0} & \mathbf{A}_{up}^T & \mathbf{0} & \boldsymbol{a}_{\rho p}^T \\
\mathbf{0} & \mathbf{0} & \boldsymbol{a}_{\rho p} & 0
\end{bmatrix}_e
\begin{Bmatrix}
\mathbf{L}_e \\
\mathbf{u}_e \\
\mathbf{p}_e \\
\zeta
\end{Bmatrix}
=
\begin{Bmatrix}
\mathbf{f}_L \\
\mathbf{f}_u \\
\mathbf{f}_p \\
0
\end{Bmatrix}_e
+
\begin{bmatrix}
\mathbf{A}_{L\hat{u}} \\
\mathbf{A}_{u\hat{u}} \\
\mathbf{A}_{p\hat{u}} \\
\mathbf{0}
\end{bmatrix}_e
\hat{\mathbf{u}}_e
+
\begin{Bmatrix}
\mathbf{0} \\
\mathbf{0} \\
\mathbf{0} \\
1
\end{Bmatrix} \rho_e,
\qquad (3.18)
$$

for $e = 1, \ldots, \mathrm{n_{el}}$ and ζ being the Lagrange multiplier associated with the constraint on the mean value of the pressure on the boundary of the element introduced in Equation (3.17d).

It is straightforward to observe that the formulation of Equation (3.18) is general and also holds for the classical hybridizable discontinuous Galerkin formulation presented in [81]. Nonetheless, owing to Voigt notation, solely the non-redundant components of the second-order strain rate tensor are stored in $\mathbf{L}_e$, that is $\mathrm{m_{sd}}$ components instead of $\mathrm{n_{sd}^2}$. Thus, the size of the block matrices $\mathbf{A}_{LL}$, $\mathbf{A}_{Lu}$ and $\mathbf{A}_{L\hat{u}}$, of the block vector $\mathbf{f}_L$ and of the first zero block vector of the last term of the right-hand side of Equation (3.18) changes when considering the classical HDG formulation or the one based on Voigt notation. In particular, being $\mathrm{n_{en}}$ the number of nodes per element, in the former case, the block $\mathbf{A}_{LL}$ is a $\mathrm{n_{sd}^2 n_{en}} \times \mathrm{n_{sd}^2 n_{en}}$ matrix, whereas in the latter it reduces to a $\mathrm{m_{sd} n_{en}} \times \mathrm{m_{sd} n_{en}}$ one. Thus, the $\mathrm{n_{en}(n_{sd}^2 + n_{sd} + 1) + 1} \times \mathrm{n_{en}(n_{sd}^2 + n_{sd} + 1) + 1}$ linear system of Equation (3.18) arising from the classical HDG local problem reduces to a $\mathrm{n_{en}(m_{sd} + n_{sd} + 1) + 1} \times \mathrm{n_{en}(m_{sd} + n_{sd} + 1) + 1}$ using Voigt notation. In two dimensions using triangular mesh elements and polynomials of degree 3, each elemental local problem features 71 degrees of freedom using the classical formulation versus 61 when exploiting Voigt notation. The computational saving greatly increases with high-order approximations, e.g. for $k = 6$ the size of each local problem is reduced from 197 to 169 equations. In a similar fashion, in a three-dimensional domain discretized using tetrahedral mesh elements, each elemental local problem reduces from 261 to 201 degrees of freedom for $k = 3$ and from 1093 to 841 using polynomials of degree 6 depending on the selected HDG formulation. Despite these numbers clearly highlight an important reduction of the dimension of the system in Equation (3.18), it is worth reminding that these problems are solved element-by-element and may be easily tackled in parallel,

whereas the most expensive step is represented by the solution of the global problem discussed below.

For the global problem, the discrete weak formulation equivalent to (3.12) is: find $\widehat{\boldsymbol{u}}^h \in [\hat{\mathcal{V}}^h(\Gamma \cup \Gamma_N)]^{\mathtt{n}_{\mathtt{sd}}}$ and $\rho^h \in \mathbb{R}^{\mathtt{n}_{\mathtt{el}}}$ such that

$$\sum_{e=1}^{\mathtt{n}_{\mathtt{el}}} \Big\{ \langle \widehat{\boldsymbol{w}}, \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D} + \langle \widehat{\boldsymbol{w}}, \mathbf{E}^T \mathbf{N}_e p_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D} + \langle \widehat{\boldsymbol{w}}, \boldsymbol{\tau}\, \boldsymbol{u}_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D}$$

$$- \langle \widehat{\boldsymbol{w}}, \boldsymbol{\tau}\, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D} \Big\} = -\sum_{e=1}^{\mathtt{n}_{\mathtt{el}}} \langle \widehat{\boldsymbol{w}}, \boldsymbol{t} \rangle_{\partial\Omega_e \cap \Gamma_N}, \quad (3.19\text{a})$$

$$\langle \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}, 1 \rangle_{\partial\Omega_e \backslash \Gamma_D} = -\langle \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D, 1 \rangle_{\partial\Omega_e \cap \Gamma_D} = 0 \quad \text{for } e = 1, \ldots, \mathtt{n}_{\mathtt{el}}, \quad (3.19\text{b})$$

for all $\widehat{\boldsymbol{w}} \in [\hat{\mathcal{V}}^h(\Gamma \cup \Gamma_N)]^{\mathtt{n}_{\mathtt{sd}}}$.

Using the discretization described in Remark 10, the linear system associated with the discretization of the HDG global problem of Equation (3.19) has the following structure:

$$\sum_{e=1}^{\mathtt{n}_{\mathtt{el}}} \Big\{ \begin{bmatrix} \mathbf{A}_{L\hat{u}}^T & \mathbf{A}_{u\hat{u}}^T & \mathbf{A}_{p\hat{u}}^T \end{bmatrix}_e \begin{Bmatrix} \mathbf{L}_e \\ \mathbf{u}_e \\ \mathbf{p}_e \end{Bmatrix} + [\mathbf{A}_{\hat{u}\hat{u}}]_e\, \hat{\mathbf{u}}_e \Big\} = \sum_{i=e}^{\mathtt{n}_{\mathtt{el}}} [\mathbf{f}_{\hat{u}}]_e. \quad (3.20\text{a})$$

$$[\mathbf{A}_{\rho\hat{u}}]_e \hat{\mathbf{u}}_e = [\mathbf{f}_\rho]_e. \quad (3.20\text{b})$$

The expressions of the matrices and vectors in Equations (3.18) and (3.20) are given in Appendix A.

3.2.3 Local post-process of the velocity field

As usual in HDG, an element-by-element post-process procedure is considered to construct an improved approximation of the velocity field. Modifying the Brezzi-Douglas-Marini (BDM) projection operator, see [53], in [93, 81] a technique to retrieve an $H(\text{div})$-conforming and exactly divergence-free velocity field was discussed. In this section, the requirement of $H(\text{div})$-conformity is relaxed and a simpler approach inspired by the work of Stenberg [158] and exploited in [24, 152, 155] is considered. Nevertheless, it is known [81] that using the Cauchy formulation of the Stokes equation, a loss of superconvergence is experienced by low-order approximations. It is worth recalling that in order to construct a superconvergent post-processed velocity field, two ingredients are required:

(i) a mixed variable $\boldsymbol{L}$ optimally convergent with order $k+1$;

(ii) a post-processing procedure able to resolve the underdetermination of the rigid body motions.

The adoption of Voigt notation allows the strong imposition of the symmetry of the stress tensor and, consequently, the pointwise fulfilment of the conservation of angular

momentum which is only weakly satisfied by classical HDG formulations as the one discussed in [81]. This allows to retrieve the optimal convergence of the strain gradient, even for low-order approximations, and requirement (i) is thus fulfilled.

In this section, a novel strategy to handle rigid body motions and fulfill requirement (ii) is presented. Basic idea relies on introducing a constraint in the post-processing equation without modifying the discrete spaces in which the variables are sought. Thus, Voigt notation allows to circumvent the complex mathematical framework of $\boldsymbol{M}$-decomposition discussed in [97, 98, 99, 100] to devise superconvergent HDG approximations with strongly and weakly symmetric stress tensors. The resulting local post-process problem exploits the optimal convergence rate of order $k+1$ of the mixed variable and additional conditions to take care of translational and rotational rigid body motions to construct a velocity field $\boldsymbol{u}^\star$ superconverging with order $k+2$.

For each element $\Omega_e, \ e = 1, \ldots, \mathtt{n}_{\mathtt{el}}$, the post-processed velocity $\boldsymbol{u}^\star$ is the solution of the problem

$$\begin{cases} \boldsymbol{\nabla}_{\mathrm{S}}^{T}\mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e^\star = -\boldsymbol{\nabla}_{\mathrm{S}}^{T}\boldsymbol{L}_e^h & \text{in } \Omega_e, \\ \mathbf{N}_e^{T}\mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e^\star = -\mathbf{N}_e^{T}\boldsymbol{L}_e^h & \text{on } \partial\Omega_e, \end{cases} \tag{3.21}$$

in the space $\left[\mathcal{V}_\star^h(\Omega)\right]^{\mathtt{n}_{\mathtt{sd}}}$, as it is described in (2.36). The element-by-element problem in Equation (3.21) is obtained by the definition of the mixed variable in Equation (3.10) and exploits the naturally equilibrated fluxes as condition on the boundary of the element.

The solution of Equation (3.21) is determined up to rigid motions, namely $\mathtt{n}_{\mathtt{sd}}$ translations and $\mathtt{n}_{\mathtt{rr}}$ rotations, being $\mathtt{n}_{\mathtt{sd}} = 2$ and $\mathtt{n}_{\mathtt{rr}} = 1$ in 2D and $\mathtt{n}_{\mathtt{sd}} = \mathtt{n}_{\mathtt{rr}} = 3$ in 3D. As discussed in Section 2.3, a set of $\mathtt{n}_{\mathtt{sd}} + \mathtt{n}_{\mathtt{rr}}$ constraints is introduced to retrieve the uniqueness of the solution. On the one hand, the indeterminacy due to the $\mathtt{n}_{\mathtt{sd}}$ rigid translational modes is resolved introducing the following constraint on the mean value of the velocity:

$$(\boldsymbol{u}_e^\star, 1)_{\Omega_e} = (\boldsymbol{u}_e^h, 1)_{\Omega_e}. \tag{3.22}$$

Remark 11. According to Equation (3.22), the mean value of the velocity in each element has to converge with order at least $k+2$ to guarantee that $\boldsymbol{u}^\star$ converges with order $k+2$ [91, 22, 153].

On the other hand, the $\mathtt{n}_{\mathtt{rr}}$ rigid rotational modes are taken care of by means of a condition on the curl of the velocity, namely

$$(\boldsymbol{\nabla} \times \boldsymbol{u}_e^\star, 1)_{\Omega_e} = \langle \widehat{\boldsymbol{u}}^h \cdot \boldsymbol{\tau}_e, 1 \rangle_{\partial\Omega_e}, \tag{3.23}$$

where the right-hand side of Equation (3.23) follows from the application of Stokes' theorem, being $\boldsymbol{u}_e^h = \widehat{\boldsymbol{u}}^h$ on $\partial\Omega_e$ and $\boldsymbol{\tau}_e$ the tangential direction to the boundary $\partial\Omega_e$.

Remark 12. For the post-processed velocity $\boldsymbol{u}^\star$ to superconverge with order $k+2$, the mean value of its curl inside each element (i.e., the left-hand side of Equation (3.23)) has to converge with order at least $k+1$. Consequently, the mean value of the tangential component of the hybrid variable $\widehat{\boldsymbol{u}}$ along the boundary of each element has to converge with order at least $k + 3/2$.

It is worth noting that other conditions may be considered to resolve the indeterminacy of the problem in Equation (3.21). Nevertheless, in order for the post-processed velocity to be superconvergent, the quantities appearing on the left-hand sides of these constraints have to converge with order $m \geq k + 2$. If this is not the case, despite the resulting system admits a unique solution, the superconvergence property is lost, as it is demonstrated in Section 2.4.2. For the strategy discussed in the present work, extensive numerical experiments have shown that the right-hand sides of both (3.22) and (3.23) converge with order $m > k + 2$. A rigorous proof of this result is currently under investigation.

Remark 13. Recall that the curl of the velocity represents the vorticity of the fluid. Within this context, the left hand side of Equation (3.23) may be physically interpreted as the mean value of the vorticity inside the element Ω_e. Similarly, the right-hand side represents the circulation of the flow around the boundary $\partial\Omega_e$.

Eventually, by exploiting the Voigt notation, Equation (3.23) is equivalent to

$$(\boldsymbol{\nabla}_{\mathbb{W}}\boldsymbol{u}_e^\star, 1)_{\Omega_e} = \langle \mathbf{T}\widehat{\boldsymbol{u}}, 1 \rangle_{\partial\Omega_e}. \tag{3.24}$$

3.3 Numerical studies

In this section, several examples with known analytical solution are considered, in two and three dimensions, to verify the optimal convergence and superconvergence properties of the error of the primal, mixed and post-processed variables, measured in the $\mathcal{L}_2(\Omega)$ norm and for different element types. As for all finite element methods (cf. e.g. [159]), the accuracy and convergence properties of the discussed HDG strategy depend both on the degree of the chosen polynomial approximations and on the regularity of the analytical solution of the problem. In the following numerical studies, classical assumptions on the regularity of the domain and the solution of the problem are considered [159] in order to highlight that the method is able to provide optimally-convergent high-order approximations without any restrictions, as shown in the analysis by Cockburn and co-workers [93]. If the regularity assumptions in [93] are not fulfilled, the experimental convergence rates will be bounded by the limited regularity of the analytical problem as observed in classical finite element as well [160]. First, a numerical study of the influence of the stabilization parameter $\boldsymbol{\tau}$ on the accuracy of the proposed HDG method is performed.

(a) Quadrilateral mesh   (b) Triangular mesh #1   (c) Triangular mesh #2
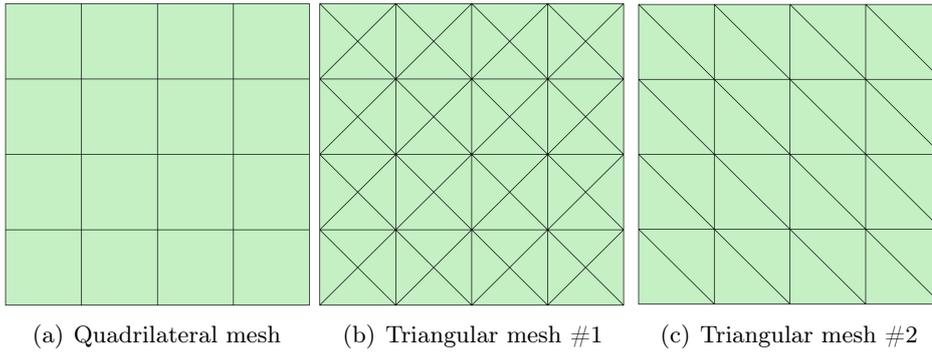
Figure 3.1: *Second level of refinement for three types of two dimensional meshes of $\Omega = [0,1]^2$ utilized for the mesh convergence study.*

3.3.1   Influence of the stabilization parameter

As previously stated and extensively studied in a series of publications by Cockburn and co-workers (cf. e.g. [21, 23, 151]), the HDG stabilization parameter has an important effect on the convergence properties of the method. For the sake of simplicity, a stabilization tensor of the form $\boldsymbol{\tau} = \tau \mathbf{I}_{\mathrm{n_{sd}}}$, equal on all the faces of the internal skeleton $\Gamma \cup \Gamma_N$ is considered. In what follows, a numerical study of the role of the scalar parameter τ is presented.

Two dimensional example

The first example considers the well-known problem of the Wang flow in the domain $\Omega = [0,1]^2$. The source term $\boldsymbol{s}$ is selected so that the analytical velocity field has the following expression

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{c} 2ax_2 - b\lambda \cos(\lambda x_1)\exp(-\lambda x_2) \\[2mm] b\lambda \sin(\lambda x_1)\exp(-\lambda x_2) \end{array} \right\}, \tag{3.25}$$

whereas the pressure is uniformly zero in the domain. The values $a = b = \lambda = 1$ are set for the constants and the kinematic viscosity ν is taken equal to 1. Neumann boundary conditions, corresponding to the analytical normal flux, are imposed on $\Gamma_N = \{(x_1, x_2) \in \Omega \mid x_2 = 0\}$ and the analytical velocity field is enforced on $\Gamma_D = \partial\Omega \setminus \Gamma_N$ via Dirichlet boundary conditions.

Uniform meshes of quadrilateral and triangular elements are considered. The second level of refinement of the meshes is shown in Figure 3.1. It is worth noting that the triangular mesh #1 has considerably more degrees of freedom than the triangular mesh #2 for a similar characteristic size.

The components of the velocity field computed on the fourth level of refinement of the triangular mesh #2 and using a quadratic degree of approximation are depicted in Figure 3.2.

(a) u_1 (b) u_2
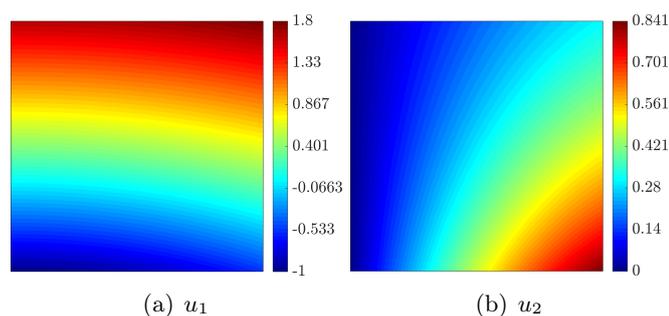
Figure 3.2: *Two dimensional problem: HDG approximation of the velocity field using the fourth refinement of the triangular mesh #2 and $k = 2$.*

Figure 3.3 shows the evolution of the error of the primal, mixed and post-processed variables, $\boldsymbol{u}$, p, $\boldsymbol{L}$ and $\boldsymbol{u}^\star$, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilization parameter τ. The numerical study is performed on the fourth level of mesh refinement, using polynomial approximations of complete degree 1 and 2 and values of τ spanning from 0.1 to 10,000. It is straightforward to observe that for all the meshes under analysis, there exists a value of τ minimizing the $\mathcal{L}_2(\Omega)$ norm of the error of the velocity. Nevertheless, to guarantee the accuracy of the approximation, the $\mathcal{H}^1(\Omega)$ norm of the error should be accounted for and consequently both $\boldsymbol{u}$ and $\boldsymbol{L}$ are considered in the choice of the optimal value of τ. Within this context and in order for the post-processed velocity field $\boldsymbol{u}^\star$ to provide a gain in accuracy with respect to $\boldsymbol{u}$, the value $\tau = 4$ is chosen for quadrilateral meshes and triangular meshes of the first type. For triangular meshes of the second type, the minimum of the error in the primal variable is achieved for values of τ substantially larger than 10. Despite the approximation of the mixed variable deteriorates when the stabilization parameter increases, this effect is limited for values of $\tau < 50$. The value of $\tau = 40$ is thus considered as it provides a good compromise for the quality of the approximation of the primal, mixed and post-processed variables.

Remark 14. Consider the family of meshes in Figure 3.1. The triangular mesh #1 features one node located in the barycenter of each underlying quadrilateral. The resulting mesh provides significantly more information than the triangular mesh #2 of the corresponding refinement level. Thus, owing to the aforementioned extra node and to the tensorial nature of the basis functions defined on the quadrilateral meshes, the behavior of the triangular meshes #1 is expected to be more similar to the quadrilateral ones than to the triangular meshes #2, as observed in the previous numerical simulations in Figure 3.3.

Three dimensional example

The second example, inspired by [161], is an analytical solution of the problem in Equation (3.1) set in the domain $\Omega = [0, 1]^3$. The source term is selected so that the

(a) Quadrilateral mesh

(b) Triangular mesh #1

(c) Triangular mesh #2
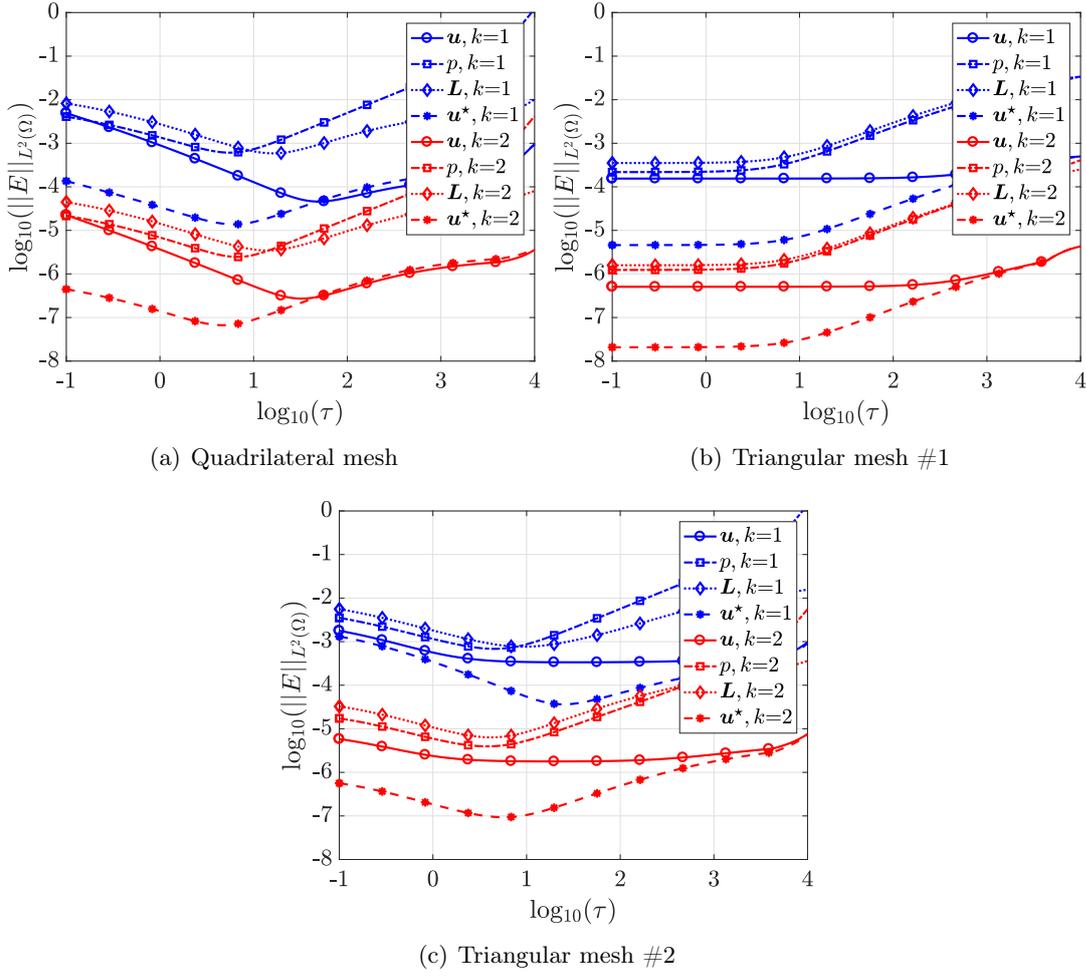
Figure 3.3: *Two dimensional problem: error of the primal, mixed and post-processed variables, $\boldsymbol{u}$, p, $\boldsymbol{L}$ and $\boldsymbol{u}^\star$, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilization parameter and for the fourth level of mesh refinement.*

analytical velocity is

$$
\boldsymbol{u}(\boldsymbol{x}) = \begin{Bmatrix} b\exp\{a(x_1-x_3)+b(x_2-x_3)\} - a\exp\{a(x_3-x_2)+b(x_1-x_2)\} \\ b\exp\{a(x_2-x_1)+b(x_3-x_1)\} - a\exp\{a(x_1-x_3)+b(x_2-x_3)\} \\ b\exp\{a(x_3-x_2)+b(x_1-x_2)\} - a\exp\{a(x_2-x_1)+b(x_3-x_1)\} \end{Bmatrix} \tag{3.26}
$$

and the corresponding pressure field is

$$
p(\boldsymbol{x}) = x_1(1-x_1). \tag{3.27}
$$

The values $a = 1$ and $b = 0.5$ are considered and the kinematic viscosity ν is taken equal to 1. Neumann boundary conditions, corresponding to the analytical flux, are imposed on $\Gamma_N = \{(x_1, x_2, x_3) \in \Omega \mid x_3 = 0\}$ and the analytical velocity field is enforced on $\Gamma_D = \partial\Omega \setminus \Gamma_N$ via Dirichlet boundary conditions.

Figure 3.4 shows a cut through the third level of refinement of the uniform meshes

(a) Hexahedral mesh   (b) Tetrahedral mesh   (c) Prismatic mesh   (d) Pyramidal mesh
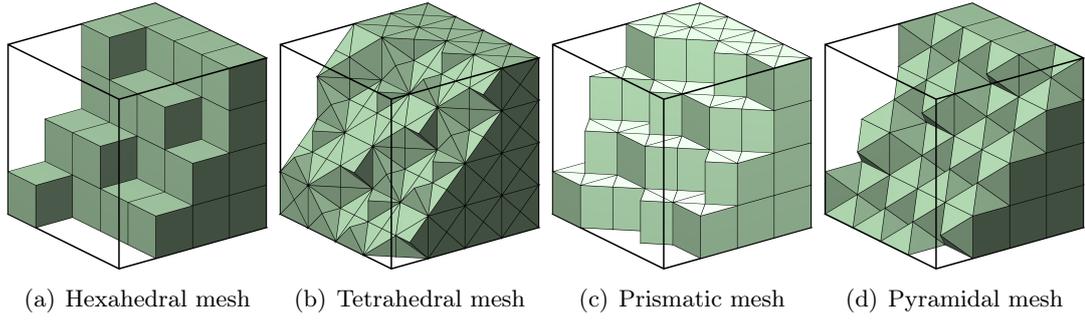
Figure 3.4: *Third level of refinement for four types of three dimensional meshes of $\Omega = [0,1]^3$ utilized for the mesh convergence study.*

(a) u_1 (b) u_2 (c) u_3 (d) p
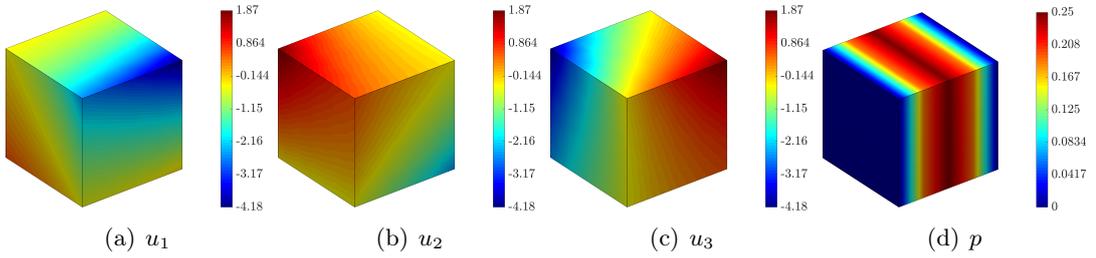
Figure 3.5: *Three dimensional problem: HDG approximation of the velocity and pressure fields using the third refinement of the hexahedral mesh and $k = 3$.*

of hexahedral, tetrahedral, prismatic and pyramidal elements considered in this study.

The velocity and pressure fields computed on the third level of refinement of the hexahedral mesh and using a cubic degree of approximation are depicted in Figure 3.5.

The evolution of the error of the primal, mixed and post-processed variables, $\boldsymbol{u}$, p, $\boldsymbol{L}$ and $\boldsymbol{u}^\star$, in the $\mathcal{L}_2(\Omega)$ norm as a function of the stabilization parameter τ is presented in Figure 3.6. As highlighted by the theory [93] and confirmed by the analysis of the two dimensional case, a value of the stabilization parameter of order one (i.e. $\tau \in [1, 10)$) guarantees stability and convergence of the HDG method. More precisely, a value near $\tau = 10$ provides the minimum error for the primal variable but limited or no extra gain in accuracy is obtained through the post-process of the velocity field. Thus, a value of $\tau = 4$ is selected for the following simulations.

The discussed numerical results show that the HDG discretization is robust to the choice of the stabilization parameter. Moreover, the optimal value of τ is not dependent upon the degree of approximation or the dimensionality of the problem. Considering the different types of elements under analysis, the triangular meshes #2 require a slightly larger value of the stabilization parameter to enter the asymptotic regime and show the optimal convergence and superconvergence properties expected from the theory.

(a) Hexahedral mesh   (b) Tetrahedral mesh

(c) Prismatic mesh   (d) Pyramidal mesh
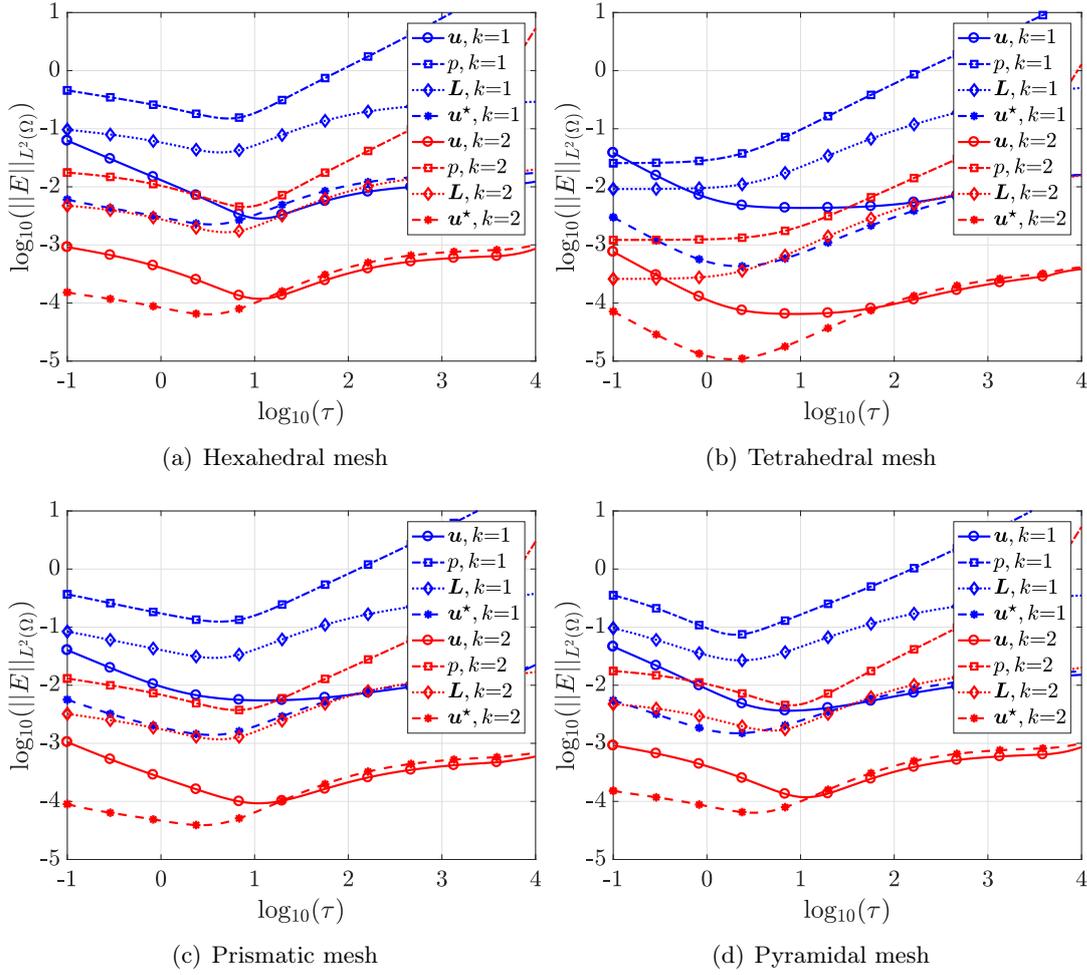
Figure 3.6: *Three dimensional problem: error of the primal, mixed and post-processed variables,* $\boldsymbol{u}$, p, $\boldsymbol{L}$ *and* $\boldsymbol{u}^\star$, *in the* $\mathcal{L}_2(\Omega)$ *norm as a function of the stabilization parameter and for the third level of mesh refinement.*

3.3.2 Optimal convergence and superconvergence of the primal, mixed and postprocessed variables

Consider the optimal values of τ identified in the previous section. The optimal convergence properties of the velocity $\boldsymbol{u}$, the pressure p and the mixed variable $\boldsymbol{L}$ representing the strain rate tensor, are tested for different element types using the $\mathcal{L}_2(\Omega)$ norm. Moreover, the superconvergence of the post-processed velocity field $\boldsymbol{u}^\star$ is also analyzed.

Two dimensional example

In Figure 3.7, the first column presents the convergence of the error of the primal and mixed variables p and $\boldsymbol{L}$, measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h for both quadrilateral and triangular elements and for a degree of approximation ranging from $k = 1$ up to $k = 3$. In a similar fashion, the second column
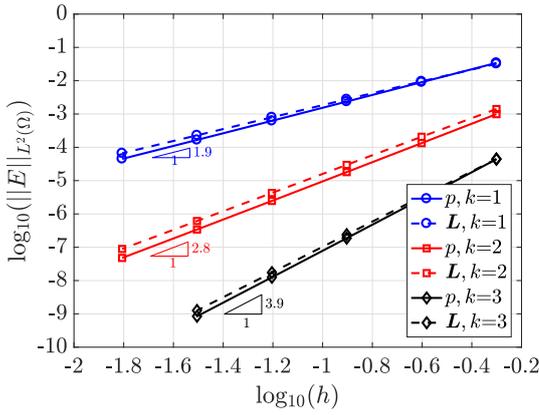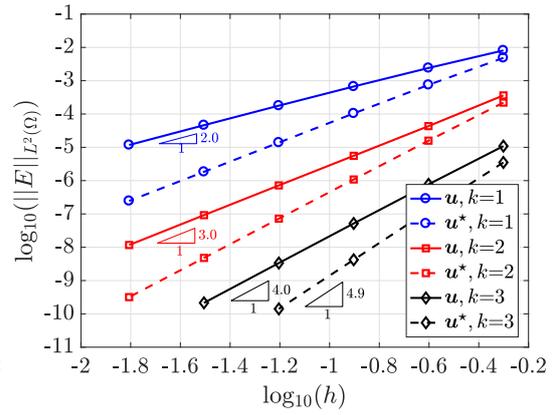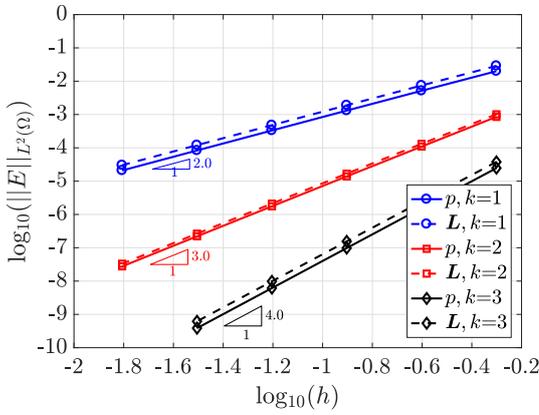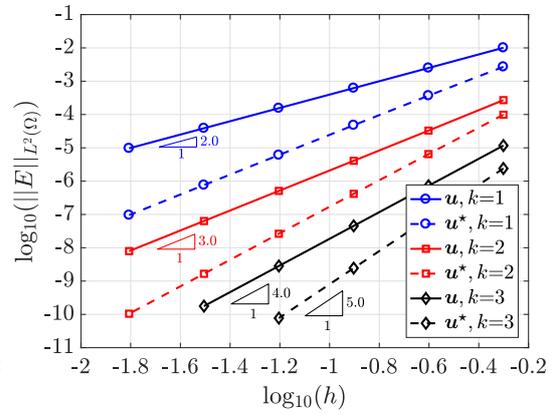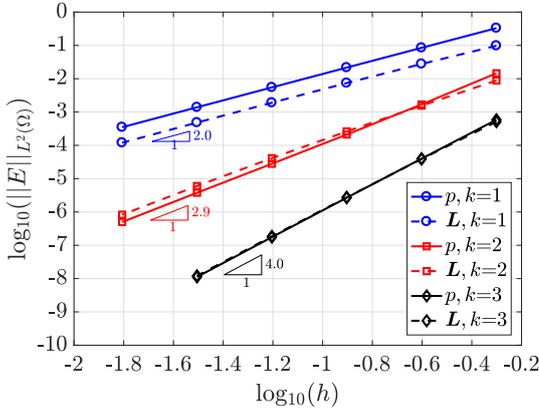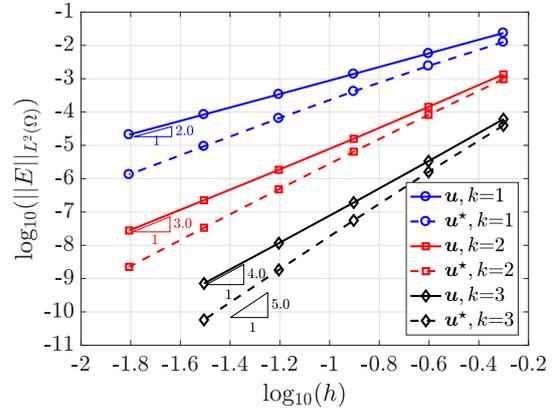
Figure 3.7: *Two dimensional problem: h-convergence of the error of the primal, mixed and post-processed variables, p and $\boldsymbol{L}$ (on the left), $\boldsymbol{u}$ and $\boldsymbol{u}^\star$ (on the right), in the $\mathcal{L}_2(\Omega)$ norm for quadrilateral and triangular meshes with different degrees of approximation.*

provides the corresponding convergence history for the primal and the post-processed velocities $\boldsymbol{u}$ and $\boldsymbol{u}^\star$.

It can be observed that almost the optimal or the optimal rate of convergence h^{k+1}

is obtained for $\boldsymbol{u}$, p and $\boldsymbol{L}$, for all the element types and degrees of approximation considered. As previously mentioned, the triangular mesh #1 has considerably more degrees of freedom than the other meshes: in particular, for the same characteristic element size, the triangular meshes #1 have approximately 2.5 times (respectively, 5 times) more internal faces than the triangular mesh #2 (respectively, the quadrilateral mesh). Thus, despite the results in Figure 3.7 indicate that the triangular mesh #1 provides more accuracy than the other meshes, a comparison in terms of the global number of degrees of freedom confirms that similar results are obtained using meshes of different element types. Concerning the post-processed variable, the rate of convergence h^{k+2} is achieved and the superconvergence property is verified. This confirms that the average of the hybrid variable $\widehat{\boldsymbol{u}}$ on the boundary leads to a superconvergent approximation, as observed in Chapter 2 for the linear elastic problem. Beside the improved convergence rate, the discussed post-process procedure is responsible for a gain in accuracy of $\boldsymbol{u}^{\star}$ with respect to the original approximation $\boldsymbol{u}$ of the velocity field. Hence, the information encapsulated in the primal and post-processed variables may be exploited to construct an error indicator and devise an automatic degree adaptivity strategy as discussed in [95, 155].

Three dimensional example

Similarly to the previous example, the convergence of the error of p and $\boldsymbol{L}$ (Fig. 3.8) and $\boldsymbol{u}$ and $\boldsymbol{u}^{\star}$ (Fig. 3.9), measured in the $\mathcal{L}_2(\Omega)$ norm, as a function of the characteristic element size h is presented for hexahedral, tetrahedral, prismatic and pyramidal elements and for a degree of approximation ranging from $k = 1$ up to $k = 3$.

As for the two dimensional case, almost the optimal or the optimal rate of convergence h^{k+1} is obtained for $\boldsymbol{u}$, p and $\boldsymbol{L}$ in 3D, for all the element types and degrees of approximation considered (cf. Fig. 3.8-3.9). In Figure 3.9, the post-processed variable is shown to superconverge with a rate of convergence h^{k+2}. Beside the improved convergence rate, the discussed post-process procedure is responsible of a gain in accuracy of $\boldsymbol{u}^{\star}$ with respect to the original approximation $\boldsymbol{u}$ of the velocity field.

The presented numerical experiments in two and three dimensions confirm that exploiting Voigt notation the HDG approximation of the Stokes equation achieves optimal convergence rate h^{k+1} for both the primal variables $\boldsymbol{u}$ and p and the mixed one $\boldsymbol{L}$. In particular, contrary to what observed in [81], the convergence of the mixed variable does not deteriorate when considering the Cauchy formulation of the Stokes flow. As discussed in Chapter 2 for the linear elastic problem, the post-process technique exploiting the curl of $\boldsymbol{u}$ allows to construct an approximation of the primal vector field superconverging with order $k + 2$. Moreover, the post-process strategy provides an extra gain in accuracy with respect to the original approximation of the velocity field. As highlighted in Figure 3.9, a solution that is almost one order of magnitude more precise than the HDG solution is obtained, even for linear approximations.

(a) Hexahedral meshes: $p, \boldsymbol{L}$

(b) Tetrahedral meshes: $p, \boldsymbol{L}$

(c) Prismatic meshes: $p, \boldsymbol{L}$

(d) Pyramidal meshes: $p, \boldsymbol{L}$
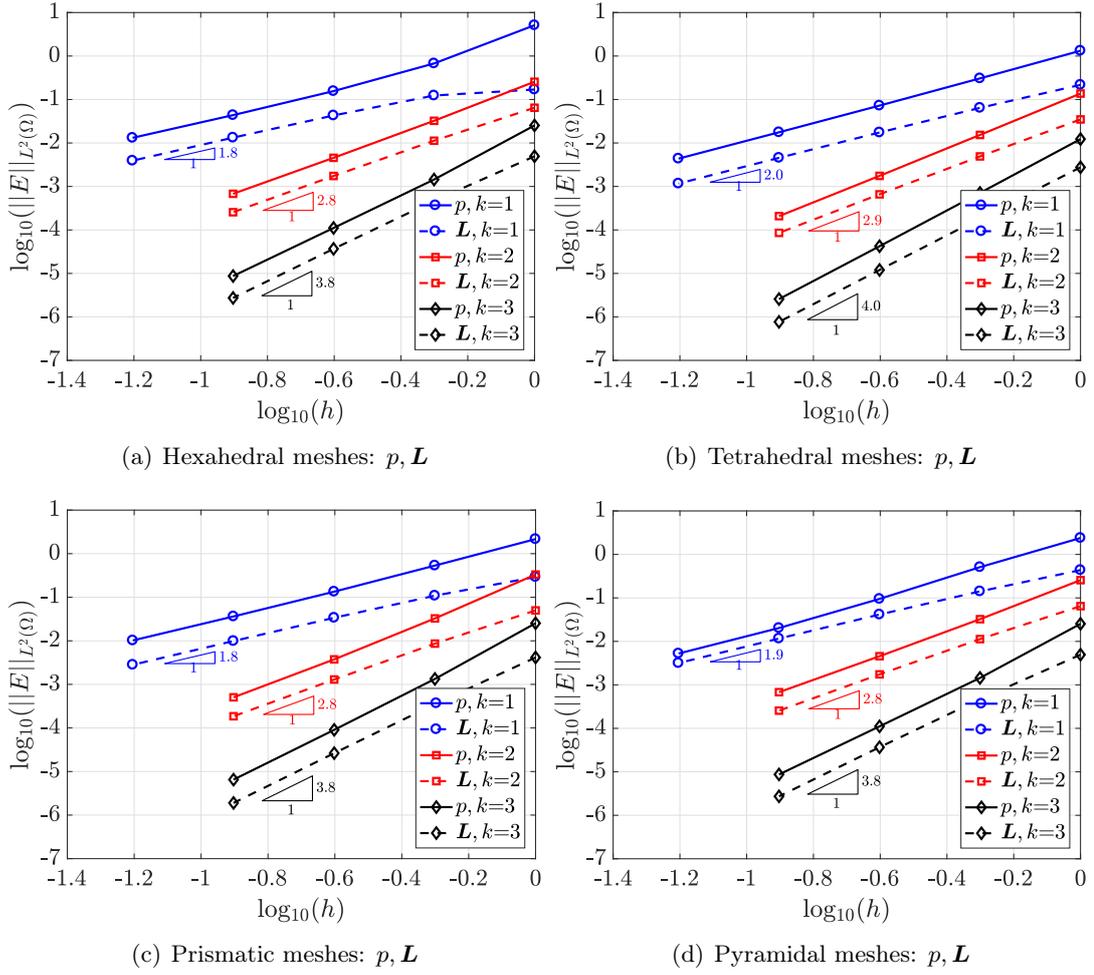
Figure 3.8: *Three dimensional problem:* h-*convergence of the error of the primal and mixed variables,* p *and* $\boldsymbol{L}$, *in the* $\mathcal{L}_2(\Omega)$ *norm for hexahedral, tetrahedral, prismatic and pyramidal meshes with different degrees of approximation.*

3.3.3   Numerical evaluation of quantities of interest: drag force on a sphere

The last example considers the classical test case of the viscous flow around a sphere. The objective of this test is to show the capability of the described HDG method to provide an approximation of the pressure and the viscous forces sufficiently accurate to evaluate a quantity of interest with the precision required by industrial standards. Consider the domain $\Omega = ([-7, 15] \times [-5, 5] \times [-5, 5]) \setminus \mathcal{B}_{1,\boldsymbol{0}}$, $\mathcal{B}_{1,\boldsymbol{0}}$ being a ball of unit radius centered at the origin. To reduce the computational effort, the symmetry of Ω is exploited and solely one fourth of the domain is taken into account to perform the numerical experiments. Different tetrahedral meshes of the domain are considered, ranging from 3,107 to 204,099 elements. High-order computations employ isoparametric curved meshes. The extension to high-order is performed using the solid mechanics analogy described in [162, 163]. Figure 3.10 (a)-(b) shows the magnitude of the velocity
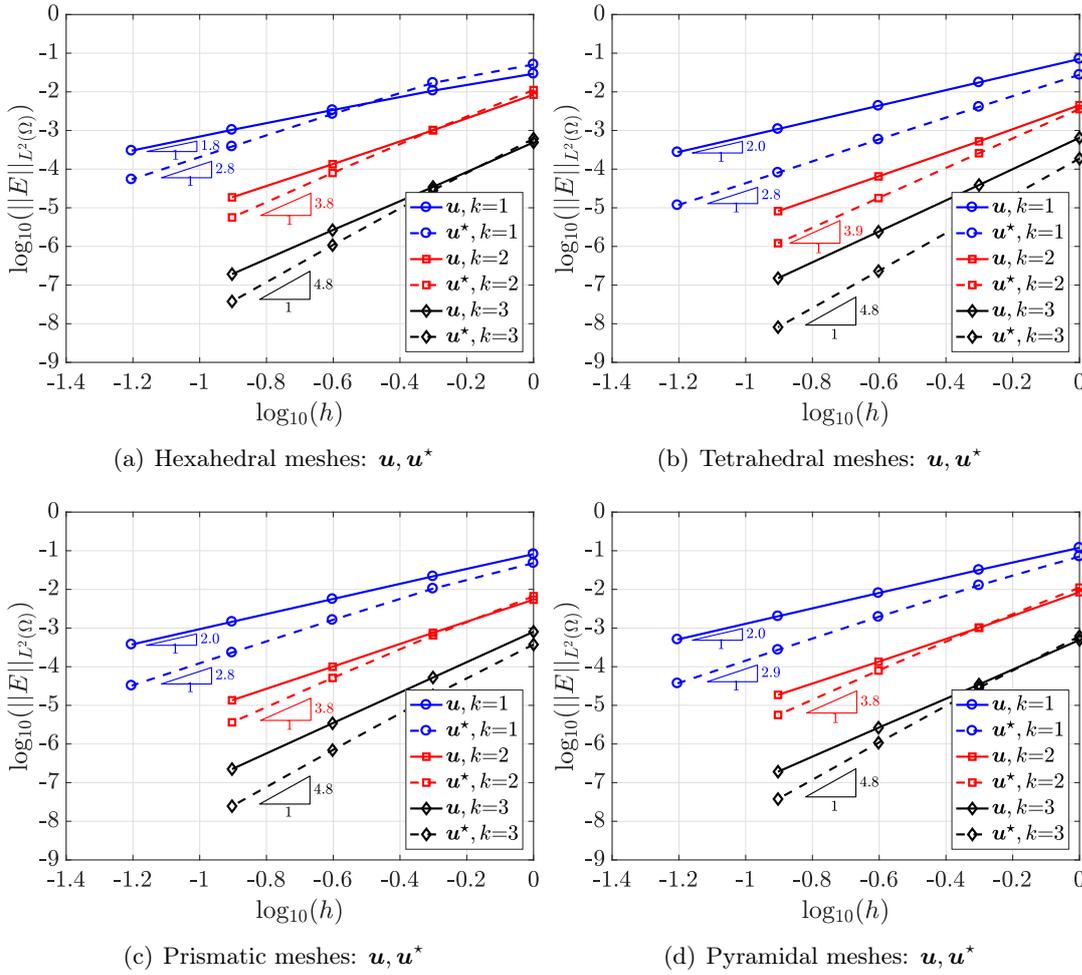
Figure 3.9: *Three dimensional problem: h-convergence of the error of the primal and post-processed variables, $\boldsymbol{u}$ and $\boldsymbol{u}^\star$, in the $\mathcal{L}_2(\Omega)$ norm for hexahedral, tetrahedral, prismatic and pyramidal meshes with different degrees of approximation.*

with streamlines of the flow and the pressure field computed on the third level of refinement of the mesh, featuring 43,682 tetrahedrons, and using a quadratic degree of approximation.

The results in Figure 3.10 (c) show the convergence of the drag force as the number of degrees of freedom is increased, i.e. for different levels of mesh refinement and for a degree of approximation ranging from $k = 1$ up to $k = 3$. The numerically computed drag is compared with the analytical value from the literature [164]. In Table 3.1, a quantitative analysis of the relative error in the computation of the drag force is reported for all the mesh refinements and degrees of approximation considered. Using linear elements, almost 4 millions degrees of freedom are required by the method to compute the drag coefficient with a relative error of 2%. The same level of accuracy is achieved by quadratic and cubic elements using the coarsest mesh under analysis and less than 200,000 degrees of freedom. More precisely, moving to high-order approximations, errors lower than 0.5% are obtained using few hundreds thousands degrees of

(a) Magnitude of the velocity with streamlines
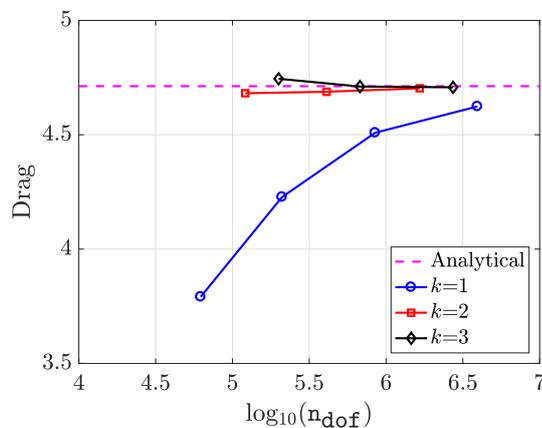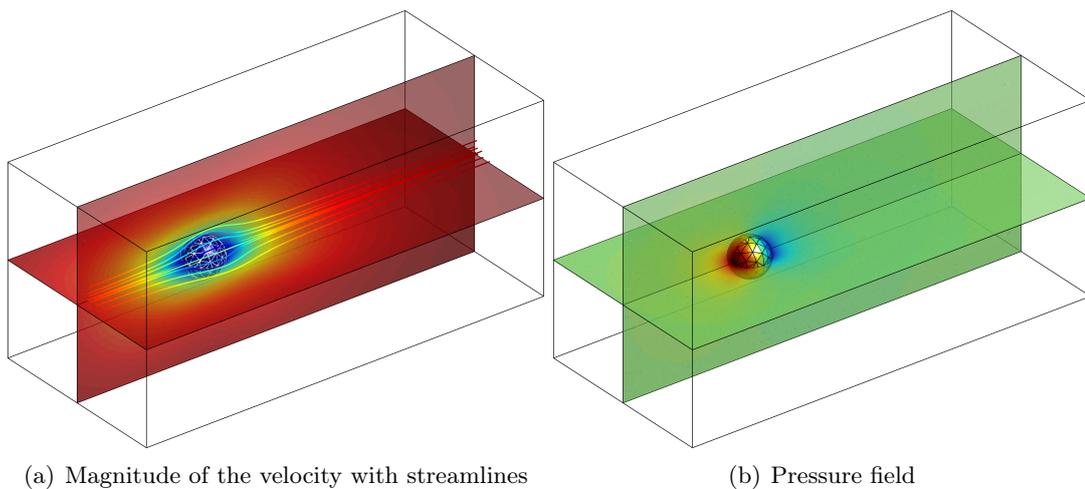
(b) Pressure field

(c) Drag force

Figure 3.10: *Flow past a sphere: HDG approximation of (a) the velocity field with streamlines of the flow and (b) the pressure field using the third level of refinement of a tetrahedral mesh and $k = 2$. (c) Convergence of the drag as a function of the number of degrees of freedom.*

freedom. The observed additional accuracy results from the concurrent use of high-order polynomial functions for the discretization of the unknown variables and high-order approximations of the geometry via meshes featuring curved elements. Thus, the superiority of high-order methods with respect to low-order ones discussed in the literature (cf. e.g. [165]) is confirmed.

3.4 Conclusion

In this chapter, a hybridizable discontinuous Galerkin method using Voigt notation, first introduced in [146] and presented in Chapter 2 of this thesis, for the Cauchy formulation of the Stokes equation is presented. Owing to Voigt notation, the symmetry of the stress tensor is strongly enforced by storing in a vector format only half of the off-diagonal terms. This allows to fulfill pointwise the conservation of angular momentum which is satisfied only in a weak sense by classical HDG formulations.

k	Mesh	Elements	$\mathtt{n_{dof}}$	Drag error
	1	3,107	62,147	$1.95 \cdot 10^{-1}$
	2	10,680	210,453	$1.03 \cdot 10^{-1}$
1	3	43,682	849,452	$4.32 \cdot 10^{-2}$
	4	204,099	3,934,212	$1.88 \cdot 10^{-2}$
	1	3,107	121,187	$6.52 \cdot 10^{-3}$
2	2	10,680	410,226	$5.18 \cdot 10^{-3}$
	3	43,682	1,655,222	$1.96 \cdot 10^{-3}$
	1	3,107	199,907	$6.88 \cdot 10^{-3}$
3	2	10,680	676,590	$4.25 \cdot 10^{-4}$
	3	43,682	2,729,582	$1.02 \cdot 10^{-3}$

Table 3.1: *Flow past a sphere: relative error in the computation of the drag force for different levels of mesh refinement and with different degrees of approximation.*

Moreover, physically meaningful tractions may be naturally imposed on the Neumann boundary. Contrary to the existing superconvergent HDG formulations involving the symmetric part of the gradient, the proposed method does not enrich the discrete spaces of approximation and it reduces the number of degrees of freedom of the mixed variable. Hence, the resulting local problems are smaller and computationally more efficient.

The optimal convergence order $k + 1$ is achieved for all the unknowns, as proved for the classical HDG equal-order approximation of the velocity-pressure formulation and for the more involved discretization of the Cauchy formulation based on the $\boldsymbol{M}$-decomposition. The novelty and main advantage of the present approach relies on being able to exploit the same degree of approximation for both primal and mixed variables, in presence of the symmetric part of the gradient. In addition, a velocity field superconverging with order $k + 2$ is obtained via a local post-process procedure, without modifying the discrete spaces in which the variable are sought. In particular, the optimal convergence of the mixed and hybrid variables is exploited to devise the superconvergent velocity and additional constraints are added to the post-process problem to resolve the underdetermination associated with rigid body motions.

Numerical studies show the optimal convergence and superconvergence properties of the method in 2D and 3D using meshes of different element types and the robustness of the approach with respect to the choice of the HDG stabilization parameter. Eventually, the drag force on a sphere is evaluated using different degrees of approximations to show the capability of the method to compute industrially relevant quantities of interest with an acceptable precision.

Given the demonstrated superconvergent properties of the proposed HDG method in the Stokes problem, the method will be extended to the case of Navier-Stokes equations. In this way, a wider spectrum of incompressible flows in terms of Reynolds number can be studied.

Chapter 4

HDG method for incompressible Navier-Stokes equations

When inertial forces grow and become comparable with viscous forces, they may not be neglected and the Navier-Stokes equations are used for describing the motion of a viscous fluid. The difference with the previously presented Stokes flow is the presence of a nonlinear convective term. The formulation of the HDG method that was proposed in the previous chapter for the Stokes flow is extended to the laminar Navier-Stokes equations. The approach will be based on the same key elements, namely the same degree of approximation polynomials for all primal, mixed and hybrid variables, and the strong enforcement of the symmetry of the stress tensor via the use of Voigt notation.

After demonstrating the excellent convergence properties of all variables and the superconvergence of the velocity field, the improved velocity field is used in order to construct an element-by-element estimator of the spatial error and, thus, perform degree adaptivity. In addition, many practical applications are time-dependent, therefore the transient version of Navier-Stokes equations is also studied. The high-order in space HDG method can be combined with classical time-integration schemes for the simulation of transient flows in a fairly straightforward manner. These schemes have to be high-order accurate in time, in order for the overall accuracy to be maintained, hence special attention to the choice of the time-integration scheme must be paid. Certain temporal schemes include inexpensive error indicators, which enable the design of an adaptive stepsize procedure, thereby controlling the temporal error during the simulation. In this chapter, this possibility is investigated in the context of HDG.

This chapter has the following structure. In Section 4.1, the proposed formulation of the HDG method is presented for the steady incompressible laminar Navier-Stokes equations. Extensive numerical experiments are conducted and the results are shown in Section 4.2. Later, the superconvergence of the velocity field is utilized to perform degree-adaptive simulations, see Section 4.3. Time-integration schemes are presented in Section 4.4 using the proposed formulation of the HDG method. Numerical tests are conducted in Section 4.4.2 whereby the high temporal accuracy of the proposed method for transient simulations is demonstrated. The timestep control and the degree

adaptivity in transient simulations are introduced in Section 4.5. The results of this chapter are summarized in Section 4.6.

4.1 The HDG method for steady incompressible Navier-Stokes equations

In this section, the HDG framework and the symmetric tensors using Voigt notation for the Navier-Stokes equations is presented. Based on the framework presented in Chapter 3 for the Stokes flow, the same approach is followed here. First, the Cauchy formulation of the Navier-Stokes equations is recalled and the equations are rewritten using the Voigt notation. Afterwards, the strong and weak form of both local and local problems is briefly presented.

4.1.1 Cauchy formulation of the Navier-Stokes equations using Voigt notation

We consider a domain $\Omega \subset \mathbb{R}^{n_{sd}}$ with boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$, $\Gamma_D \cap \Gamma_N = \emptyset$ and n_{sd} the number of spatial dimensions. The strong form of the incompressible Navier–Stokes equations reads as follows:

$$
\begin{cases}
\boldsymbol{\nabla} \cdot (\boldsymbol{u} \otimes \boldsymbol{u}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{s} & \text{in } \Omega, \\
\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 & \text{in } \Omega, \\
\boldsymbol{\sigma} = -p\mathbf{I}_{n_{sd}} + 2\nu\boldsymbol{\varepsilon}(\boldsymbol{u}) & \text{in } \Omega, \\
\boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\
\boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t} & \text{on } \Gamma_N,
\end{cases}
\tag{4.1}
$$

where $(\boldsymbol{u}, p, \boldsymbol{\sigma})$ are, respectively, the velocity vector, pressure scalar and Cauchy second-order stress tensor. The kinematic viscosity is described by the coefficient ν. The terms $\boldsymbol{s}, \boldsymbol{u_D}$ and $\boldsymbol{t}$ are the volumetric source term, the Dirichlet and the traction boundary conditions, defined respectively in the domain Ω, on the boundary Γ_D and boundary Γ_N. The normal vector to the surface is $\boldsymbol{n}$. The strain rate tensor $\boldsymbol{\varepsilon}(\boldsymbol{u})$ is the symmetric part of the gradient of velocity, as it is defined in Equation (3.4) and its components in Equation (3.3).

Using the Voigt notation for symmetric tensors, as it was introduced in the previous chapters for linear elastic materials and the Stokes flow, the system of equations (4.1)

is written

$$\begin{cases} \boldsymbol{\nabla}_{\mathrm{S}}^T \left(\boldsymbol{u} \otimes \boldsymbol{u}\right)_V - \boldsymbol{\nabla}_{\mathrm{S}}^T \boldsymbol{\sigma}_V = \boldsymbol{s} & \text{in } \Omega, \\ \mathbf{E}^T \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{u} = 0 & \text{in } \Omega, \\ \boldsymbol{\sigma}_V = -\mathbf{E}p + \mathbf{D}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \mathbf{N}^T \boldsymbol{\sigma}_V = \boldsymbol{t} & \text{on } \Gamma_N, \end{cases} \tag{4.2}$$

where the matrix $\boldsymbol{\nabla}_{\mathrm{S}}$, accounting for the symmetric gradient operator, is defined in Equation (2.8). The matrices $\mathbf{E}$, $\mathbf{D}$ and $\mathbf{N}$, that represent the trace operator, the diffusion part of the Stokes law and the normal vector respectively, are given by Equations (3.6) and (2.12). The stress tensor written in vector form $\boldsymbol{\sigma}_V$ has the same structure as in Equation (3.5). The outer product $(\boldsymbol{u} \otimes \boldsymbol{u})$ is also symmetric and, using the Voigt notation, can be written in the following form

$$(\boldsymbol{u} \otimes \boldsymbol{u})_V := \begin{cases} \begin{bmatrix} u_1 u_1 & u_2 u_2 & u_1 u_2 \end{bmatrix}^T & \text{in 2D}, \\ \begin{bmatrix} u_1 u_1 & u_2 u_2 & u_3 u_3 & u_1 u_2 & u_1 u_3 & u_2 u_3 \end{bmatrix}^T & \text{in 3D}. \end{cases} \tag{4.3}$$

4.1.2 Hybridizable discontinuous Galerkin method

The system of Equations (4.2) may be written as a system of first-order equations by introducing the so-called mixed variable $\boldsymbol{L}$ as follows

$$\begin{cases} \boldsymbol{L} + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = \mathbf{0} & \text{in } \Omega_e, \text{ for } e = 1, \ldots, \mathtt{n}_{\mathtt{el}}, \\ \boldsymbol{\nabla}_{\mathrm{S}}^T \left(\boldsymbol{u} \otimes \boldsymbol{u}\right)_V + \boldsymbol{\nabla}_{\mathrm{S}}^T (\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p) = \boldsymbol{s} & \text{in } \Omega_e, \text{ for } e = 1, \ldots, \mathtt{n}_{\mathtt{el}}, \\ \mathbf{E}^T \boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u} = 0 & \text{in } \Omega_e, \text{ for } e = 1, \ldots, \mathtt{n}_{\mathtt{el}}, \\ \boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D, \\ \mathbf{N}^T (\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p) = -\boldsymbol{t} & \text{on } \Gamma_N, \\ [\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \mathbf{0} & \text{on } \Gamma, \\ [\![\mathbf{N}^T (\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p + (\boldsymbol{u} \otimes \boldsymbol{u})_V)]\!] = \mathbf{0} & \text{on } \Gamma. \end{cases} \tag{4.4}$$

The variable $\boldsymbol{L}$ represents the symmetric part of the scaled strain-rate tensor and the last two equations are the transmission conditions that enforce the continuity of the normal component of velocity and flux across the interface between the elements Γ (see also Equation (2.21) for the definition of Γ).

Strong form of the local and global problems

The HDG formulation leads to the solution of the problem in two stages. In the first stage, a local problem is defined inside each element, giving a total number of $\mathtt{n}_{\mathtt{el}}$ problems. The local variables $(\boldsymbol{L}_e, \boldsymbol{u}_e, p_e)$ are computed by solving the following

system of equations

$$\begin{cases} \boldsymbol{L}_e + \mathbf{D}^{1/2}\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = \mathbf{0} & \text{in } \Omega_e, \\ \boldsymbol{\nabla}_{\mathrm{S}}^T\left(\boldsymbol{u}_e \otimes \boldsymbol{u}_e\right)_V + \boldsymbol{\nabla}_{\mathrm{S}}^T(\mathbf{D}^{1/2}\boldsymbol{L}_e + \mathbf{E}p_e) = \boldsymbol{s_e} & \text{in } \Omega_e, \\ \mathbf{E}^T\boldsymbol{\nabla}_{\mathrm{S}}\boldsymbol{u}_e = 0 & \text{in } \Omega_e, \\ \boldsymbol{u}_e = \boldsymbol{u}_D & \text{on } \partial\Omega_e \cap \Gamma_D, \\ \boldsymbol{u}_e = \widehat{\boldsymbol{u}} & \text{on } \partial\Omega_e \setminus \Gamma_D, \end{cases} \tag{4.5}$$

where $\widehat{\boldsymbol{u}}$ is a variable representing the trace of the velocity on the mesh skeleton and allows the local problems to be solved independently. Note that the problem of Equation (4.5) is a Dirichlet problem, hence an additional constraint is necessary to make the system solvable. The mean pressure constraint on the element's boundary ρ_e is used for this reason, as described in Equation (3.11). By solving the local problem it is possible to express the variables $(\boldsymbol{L}_e, \boldsymbol{u}_e, p_e)$ in terms of the variables $\widehat{\boldsymbol{u}}, \rho$.

In the second stage, $\widehat{\boldsymbol{u}}$ and ρ are computed by solving a global problem on the elements' interface and the Neumann boundary, by satisfying the transmission conditions and the Neumann boundary condition as follows

$$\begin{cases} [\![\boldsymbol{u} \otimes \boldsymbol{n}]\!] = \mathbf{0} & \text{on } \Gamma, \\ [\![\mathbf{N}^T(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p + (\boldsymbol{u} \otimes \boldsymbol{u})_V)]\!] = \mathbf{0} & \text{on } \Gamma, \\ \mathbf{N}^T(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p) = -\boldsymbol{t} & \text{on } \Gamma_N. \end{cases} \tag{4.6}$$

The first equation is automatically satisfied due to the Dirichlet boundary condition $\boldsymbol{u}_e = \widehat{\boldsymbol{u}}$ imposed in (4.5) and the uniqueness of the hybrid variable $\widehat{\boldsymbol{u}}$ on the faces of Γ. The term $[\![\mathbf{N}^T(\boldsymbol{u} \otimes \boldsymbol{u})_V]\!]$ is equal to 0 because the involved flux $\mathbf{N}^T(\widehat{\boldsymbol{u}} \otimes \widehat{\boldsymbol{u}})_V$ is 0 on Γ, due to the uniqueness of $\widehat{\boldsymbol{u}}$ on Γ. Therefore, the second equation reduces to $[\![\mathbf{N}^T(\mathbf{D}^{1/2}\boldsymbol{L} + \mathbf{E}p)]\!] = \mathbf{0}$. Finally, due to the introduction of the variable ρ for the imposition of the compatibility constraint (3.11), the size of the global problem is increased and, therefore, an extra equation has to be employed for the solution of the global problem. The divergence-free condition in each element is typically employed, refer to Equation (3.14).

Weak form of the local and global problems

Consider the discrete functional spaces in Equation (2.26). The discrete weak formulation of the local problems in Equation (4.5) reads: for $e = 1, \dots, \mathrm{n_{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}^h$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h, p_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathrm{msd}} \times [\mathcal{V}^h(\Omega_e)]^{\mathrm{nsd}} \times \mathcal{V}^h(\Omega_e)$ such that

$$-(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D}$$
$$+ \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{4.7a}$$

$$-(\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}, \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} - (\mathbf{E}^T \boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}, p_e^h)_{\Omega_e} - (\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}, (\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V)_{\Omega_e}$$
$$+ \langle \boldsymbol{w}, \mathbf{N}_e^T \overline{(\mathbf{D}^{1/2} \boldsymbol{L}_e^h + \mathbf{E} p_e^h)} \rangle_{\partial\Omega_e} + \langle \boldsymbol{w}, \mathbf{N}_e^T \overline{(\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V} \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{b})_{\Omega_e}, \tag{4.7b}$$

$$(\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{E} q, \boldsymbol{u}_e^h)_{\Omega_e} = \langle q, \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle q, \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{4.7c}$$

$$\frac{1}{|\partial\Omega_e|} \langle p_e^h, 1 \rangle_{\partial\Omega_e} = \rho_e^h, \tag{4.7d}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, q) \in [\mathcal{V}^h(\Omega_e)]^{\mathrm{msd}} \times [\mathcal{V}^h(\Omega_e)]^{\mathrm{nsd}} \times \mathcal{V}^h(\Omega_e)$. The trace of the numerical normal flux in Equation (4.7b) is split into two terms; one due to diffusion and the other due to convection. The diffusive part is already defined in Equation (3.16) for the Stokes problem and the convective part is defined as follows

$$\mathbf{N}_e^T \overline{(\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V} := \begin{cases} \mathbf{N}_e^T (\boldsymbol{u}_D^h \otimes \boldsymbol{u}_D^h)_V + \tau^\alpha (\boldsymbol{u}_e^h - \boldsymbol{u}_D) & \text{on } \partial\Omega_e \cap \Gamma_D, \\ \mathbf{N}_e^T (\widehat{\boldsymbol{u}}_e^h \otimes \widehat{\boldsymbol{u}}_e^h)_V + \tau^\alpha (\boldsymbol{u}_e^h - \widehat{\boldsymbol{u}}^h) & \text{elsewhere.} \end{cases} \tag{4.8}$$

where the stabilization parameters due to convection τ^α as well as due to diffusion τ^d are crucial for the stability, accuracy and convergence of the HDG method. In literature a variety of expressions of the parameter have been proposed, that may depend on the local velocity magnitude and be different for each face of the mesh [24, 25]. In this study for simplicity the parameters τ^d and τ^α will be considered constant in space, that is τ^d and τ^α. In the case of Navier-Stokes equations we use a similar definition to the one presented in [25]:

$$\tau^d = \hat{\tau} \frac{\nu}{\bar{L}}, \qquad \tau^\alpha = \hat{\tau} \bar{U}, \qquad \tau = \tau^d + \tau^\alpha \tag{4.9}$$

where $\bar{L}, \bar{U}$ are the characteristic length and velocity of the problem. If we plug Equation (4.8) into Equation (4.7) and integrate the diffusion term by parts, we obtain the final form of the weak problem: for $e = 1, \ldots, \mathrm{n_{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}^h$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h, p_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathrm{msd}} \times [\mathcal{V}^h(\Omega_e)]^{\mathrm{nsd}} \times \mathcal{V}^h(\Omega_e)$ that satisfy

$$-(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D}$$
$$+ \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{4.10a}$$

$$(\boldsymbol{w}, \boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{w}, \boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{E} p_e^h)_{\Omega_e} - (\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}, (\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V)_{\Omega_e}$$
$$+ \langle \boldsymbol{w}, \tau \boldsymbol{u}_e^h \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{b})_{\Omega_e} + \langle \boldsymbol{w}, (\tau \widehat{\boldsymbol{u}}^h - \mathbf{N}_e^T (\widehat{\boldsymbol{u}} \otimes \widehat{\boldsymbol{u}})_V) \rangle_{\partial\Omega_e \setminus \Gamma_D} \tag{4.10b}$$
$$+ \langle \boldsymbol{w}, (\tau \boldsymbol{u}_D - \mathbf{N}_e^T (\boldsymbol{u}_D \otimes \boldsymbol{u}_D)_V) \rangle_{\partial\Omega_e \cap \Gamma_D},$$

$$(\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{E} q, \boldsymbol{u}_e^h)_{\Omega_e} = \langle q, \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle q, \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{4.10c}$$

$$\frac{1}{|\partial\Omega_e|} \langle p_e^h, 1 \rangle_{\partial\Omega_e} = \rho_e^h, \tag{4.10d}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, q) \in [\mathcal{V}^h(\Omega_e)]^{\mathrm{m}_{\mathrm{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathrm{n}_{\mathrm{sd}}} \times \mathcal{V}^h(\Omega_e)$.

Remark 15. The set of local problems described in Equation(4.27) are non-linear due to the non-linear terms $(\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V$ and $(\widehat{\boldsymbol{u}} \otimes \widehat{\boldsymbol{u}})_V$. The Newton-Raphson method is used to linearize the system. The system after the linearization has the following form

$$\begin{bmatrix} \mathbf{A}_{LL} & \mathbf{A}_{Lu} & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{Lu}^T & \mathbf{A}_{uu}^r & \mathbf{A}_{up} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{up}^T & \mathbf{0} & \boldsymbol{a}_{\rho p}^T \\ \mathbf{0} & \mathbf{0} & \boldsymbol{a}_{\rho p} & \mathbf{0} \end{bmatrix}_e \begin{Bmatrix} \mathbf{L}^{r+1} \\ \mathbf{u}^{r+1} \\ \mathbf{p}^{r+1} \\ \zeta^{r+1} \end{Bmatrix}_e = \begin{Bmatrix} \mathbf{f}_L \\ \mathbf{f}_u^r \\ \mathbf{f}_p \\ 0 \end{Bmatrix}_e + \begin{bmatrix} \mathbf{A}_{L\hat{u}} \\ \mathbf{A}_{u\hat{u}}^r \\ \mathbf{A}_{p\hat{u}} \\ 0 \end{bmatrix}_e \hat{\mathbf{u}}_e^{r+1} + \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix}_e \rho_e^{r+1}, \quad (4.11)$$

where r, $r+1$ denote the non-linear iteration count. The linearization consists in evaluating the non-linear submatrices, e.g. $\mathbf{A}_{uu}$, using the velocity of iteration r, therefore the superscript r is used, e.g. $\mathbf{A}_{uu}^r$. Further details on the linearization procedure can be found in Appendix A.3.

The matrix form of the local problem in Equation (4.11) may be also written as

$$\left[\boldsymbol{A}_e^r\right] \begin{Bmatrix} \mathbf{L}^{r+1} \\ \mathbf{u}^{r+1} \\ \mathbf{p}^{r+1} \\ \zeta^{r+1} \end{Bmatrix}_e = \left[\boldsymbol{f}_e^r\right] + \left[\hat{\boldsymbol{A}}_e^r\right] \hat{\mathbf{u}}_e^{r+1} + \left[\boldsymbol{r}_e\right] \rho_e^{r+1}. \quad (4.12)$$

The above system is solvable, since the matrix $\left[\boldsymbol{A}_e^r\right]$ is invertible, therefore the local variables $\begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} & \zeta \end{bmatrix}^T$ can be expressed as a function of the global variables $\begin{bmatrix} \hat{\mathbf{u}} & \rho \end{bmatrix}^T$, as follows

$$\begin{Bmatrix} \mathbf{L}^{r+1} \\ \mathbf{u}^{r+1} \\ \mathbf{p}^{r+1} \\ \zeta^{r+1} \end{Bmatrix}_e = \left[\boldsymbol{z}_e^{f,r}\right] + \left[\mathbf{Z}_e^{\hat{u},r}\right] \widehat{\boldsymbol{u}}_e^{r+1} + \left[\mathbf{Z}_e^{\rho,r}\right] \rho_e^{r+1} \quad (4.13)$$

where

$$\boldsymbol{z}_e^{f,r} = \left[\boldsymbol{A}_e^r\right]^{-1} \boldsymbol{f}_e^r, \qquad \mathbf{Z}_e^{\hat{u},r} = \left[\boldsymbol{A}_e^r\right]^{-1} \hat{\boldsymbol{A}}_e^r, \qquad \mathbf{Z}_e^{\rho,r} = \left[\boldsymbol{A}_e^r\right]^{-1} \boldsymbol{r}_e \quad (4.14)$$

The final form of the discrete weak global problem in Equation (4.6) is: for $e = 1, \ldots, \mathrm{n}_{\mathrm{el}}$, given $\boldsymbol{t}$ on Γ_N and $\boldsymbol{L}_e, \boldsymbol{u}_e, p_e$ on $\Gamma \cup \Gamma_N$, find $(\widehat{\boldsymbol{u}}^h, \rho^h) \in [\hat{\mathcal{V}}^h(\Gamma \cup \Gamma_N)]^{\mathrm{n}_{\mathrm{sd}}} \times \mathbb{R}^{\mathrm{n}_{\mathrm{el}}}$ such that

$$\sum_{e=1}^{\mathrm{n}_{\mathrm{el}}} \Big\{ \langle \widehat{\boldsymbol{w}}, \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D} + \langle \widehat{\boldsymbol{w}}, \mathbf{N}_e^T \mathbf{E} p_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D} + \langle \widehat{\boldsymbol{w}}, \tau \boldsymbol{u}_e^h \rangle_{\partial\Omega_e \backslash \Gamma_D}$$

$$\quad (4.15a)$$

$$- \langle \widehat{\boldsymbol{w}}, \tau \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \backslash \Gamma_D} \Big\} = - \sum_{e=1}^{\mathrm{n}_{\mathrm{el}}} \langle \widehat{\boldsymbol{w}}, \boldsymbol{t} \rangle_{\partial\Omega_e \cap \Gamma_N},$$

$$\langle \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}, 1 \rangle_{\partial\Omega_e \backslash \Gamma_D} = - \langle \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D, 1 \rangle_{\partial\Omega_e \cap \Gamma_D} \quad \text{for } e = 1, \ldots, \mathrm{n}_{\mathrm{el}}, \quad (4.15b)$$

for all $\widehat{\boldsymbol{w}} \in [\hat{\mathcal{V}}^h(\Gamma \cup \Gamma_N)]^{\mathrm{n}_{\mathrm{sd}}}$.

The linear system associated with the discretization of the HDG global problem of Equation (4.15) has the following structure:

$$\sum_{e=1}^{n_{el}} \left\{ \begin{bmatrix} \mathbf{A}_{L\hat{u}}^T & \mathbf{A}_{u\hat{u}}^T & \mathbf{A}_{p\hat{u}}^T \end{bmatrix}_e \begin{Bmatrix} \mathbf{L}^{r+1} \\ \mathbf{u}^{r+1} \\ \mathbf{p}^{r+1} \end{Bmatrix}_e + [\mathbf{A}_{\hat{u}\hat{u}}]_e\, \hat{\mathbf{u}}_e^{r+1} \right\} = \sum_{i=e}^{n_{el}} [\mathbf{f}_{\hat{u}}]_e. \qquad (4.16a)$$

$$[\mathbf{A}_{\rho\hat{u}}]_e \hat{\mathbf{u}}_e^{r+1} = [\mathbf{f}_\rho]_e. \qquad (4.16b)$$

Algorithm 1 Solution process of the non-linear problem for the steady incompressible Navier-Stokes equations using the proposed HDG method

---

1: Assume an initial guess of the local elemental variables $\begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} \end{bmatrix}_e^0$ and the global variables $\begin{bmatrix} \hat{\mathbf{u}} & \rho \end{bmatrix}_e^0$
2: Non-linear iteration counter r=0
3: **while** Non-linear solver not converged **do**
4:      r=r+1
5:      For each element e:
        Express the local variables $\begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} \end{bmatrix}_e^{r+1}$ in Ω_e as a function of the global variables $\begin{bmatrix} \hat{\mathbf{u}} & \rho \end{bmatrix}_e^{r+1}$, by inverting the left-hand side matrix in (4.11)
6:      Assembly the global problem:
        Substitute the $\begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} \end{bmatrix}_e^{r+1}$ in (4.16)
7:      Solve the global problem:
        Compute $\begin{bmatrix} \hat{\mathbf{u}} & \rho \end{bmatrix}$, by solving the linear system in (4.16)
8:      Retrieve the local variables from (4.13)
9:      Check if the convergence criterion is met
10: **end while**

Local post-process of the velocity field

The local post-process of the velocity field is a well-known feature of the HDG method, that gives the opportunity to construct a more accurate velocity field. In order for this to happen, the mixed variable $\boldsymbol{L}$ should converge optimally, with order $k+1$, when k is the order of the approximating polynomials, as well as the rigid body motions should be determined.

With the help of strongly-enforced symmetric tensors, it is demonstrated in Chapters 2 and 3 that the mixed variable $\boldsymbol{L}$ converges optimally for $k \geq 1$. Moreover, making use of appropriate conditions it is shown that it is possible to resolve the underdeterminacy of the translational and rotacional body motions of the velocity field and, thus, construct a superconvergent velocity field for both low-order and high-order approximations.

In the case of Navier-Stokes equations, the same approach is followed and the optimal convergent behaviour of the proposed formulation of the HDG method is demonstrated through numerical studies.

4.2   Numerical studies for steady incompressible Navier-Stokes

Numerical studies are conducted in order to test the HDG method on steady Navier-Stokes equations. These studies include an analytical two-dimensional flow field, through which it is possible to evaluate the discretization error and assess the order of convergence, a steady laminar two-dimensional flow around a circle, which is a common benchmark problem for incompressible Navier-Stokes equations and where quantities of interest of the flow are calculated, as well as the steady incompressible flow inside a microvalve with no moving parts.

4.2.1   Kovasznay flow

The objective of the first test is to evaluate the convergence rates of all the variables using the approach of the HDG method that is discussed in this thesis. For this reason, an analytical flow field is considered, in this case the Kovasznay flow [166], which approximates the flow behind equally-spaced parallel rods. The analytical expression of the velocity and pressure is

$$\boldsymbol{u}(\boldsymbol{x}) = \begin{Bmatrix} 1 - \exp(\lambda x_1)\cos(2\pi x_2) \\ \frac{\lambda}{2\pi}\exp(\lambda x_1)\sin(2\pi x_2) \end{Bmatrix}, \tag{4.17}$$

$$p(\boldsymbol{x}) = -\frac{\exp(2\lambda x_1)}{2} + \frac{1 + \exp(2\lambda) - \frac{1}{\lambda}\big(1 - \exp(2\lambda)\big)}{8} \tag{4.18}$$

where $\lambda = \frac{Re}{2} - \sqrt{\frac{Re^2}{4} + 4\pi^2}$.

A square unit $\Omega = [0,1]^2$ is typically chosen to be the flow domain for this problem. Both Dirichlet and Neumann boundary conditions are prescribed on the domain boundaries. In fact, the analytical traction $\boldsymbol{t}$, which is evaluated based on the analytical expression of the velocity and pressure field, is set to the bottom boundary $\Gamma_N = \{(x_1, x_2) \in \Omega \mid x_2 = 0\}$ and the analytical velocity field $\boldsymbol{u}_D$ is set to the rest of the boundaries. The source term $\boldsymbol{s}$ is also chosen according to the analytical velocity and pressure. The Reynolds number is set to $Re = 20$ and the dimensionless stabilization parameter is chosen $\hat{\tau} = 2$. In this experiment, uniform triangular meshes are considered, as the ones shown in Figure 3.1(b). The velocity components' fields as well as the pressure field, as they are computed using the HDG method, are shown in Figure 4.1.

Furthermore, the relative error and the rate of convergence of the all the variables in the $\mathcal{L}_2$-norm are presented in Table 4.1. The degree of approximation varies from $k = 1$ up to $k = 4$ and the element size from $h = 1/2$ up to $h = 1/64$ for $k = 1, 2$ and $h = 1/32$ for $k = 3, 4$. It is observed that the velocity, pressure and the scaled strain

(a) u_1 (b) u_2 (c) p
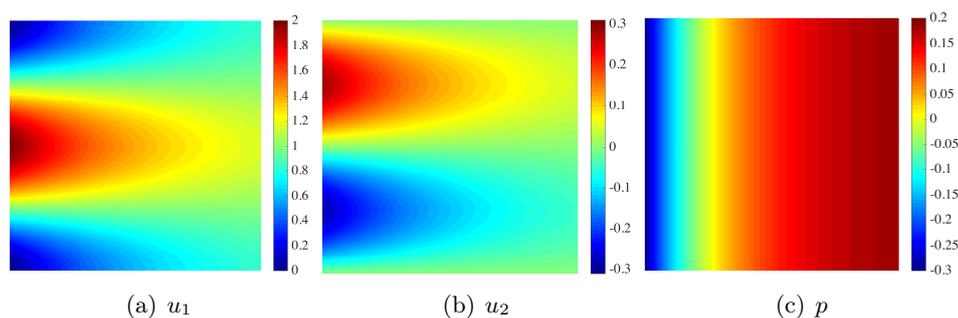
Figure 4.1: *Kovasznay flow: HDG approximation of the velocity and pressure field using the fourth refinement of the triangular mesh #1 and $k = 3$.*

rate tensor (mixed variable $\boldsymbol{L}$) converge optimally with rate $k+1$ for $k = 1, 2, 3, 4$. The post-processed velocity shows a convergence rate of $k+2$ and lower levels of numerical error in comparison with the original velocity field.

k	h	$\frac{\|\mathbf{u}-\mathbf{u}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{u}\|_{\mathcal{L}_2(\Omega)}}$	Rate	$\frac{\|\mathbf{p}-\mathbf{p}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{p}\|_{\mathcal{L}_2(\Omega)}}$	Rate	$\frac{\|\mathbf{L}-\mathbf{L}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{L}\|_{\mathcal{L}_2(\Omega)}}$	Rate	$\frac{\|\mathbf{u}^\star-\mathbf{u}_h^\star\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{u}^\star\|_{\mathcal{L}_2(\Omega)}}$	Rate
	1/2	3.32e-1	-	6.84e-1	-	6.22e-1	-	3.09e-1	-
	1/4	6.47e-2	2.36	1.51e-1	2.18	2.61e-1	1.25	6.29e-2	2.30
1	1/8	1.38e-2	2.23	3.79e-2	2.00	8.46e-2	1.63	1.00e-2	2.65
	1/16	3.02e-3	2.19	9.93e-3	1.93	2.56e-2	1.72	1.45e-3	2.79
	1/32	7.14e-4	2.08	2.60e-3	1.93	7.26e-3	1.82	1.99e-4	2.86
	1/64	1.75e-4	2.03	6.74e-4	1.95	1.95e-3	1.90	2.66e-5	2.90
	1/2	5.49e-2	-	1.49e-1	-	2.64e-1	-	8.94e-2	-
	1/4	3.76e-3	3.87	1.86e-2	3.00	3.87e-2	2.77	6.41e-3	3.80
2	1/8	3.18e-4	3.56	2.45e-3	2.93	6.17e-3	2.65	5.05e-4	3.67
	1/16	2.99e-5	3.41	3.25e-4	2.92	9.09e-4	2.76	3.70e-5	3.77
	1/32	3.12e-6	3.26	4.25e-5	2.94	1.25e-4	2.86	2.53e-6	3.87
	1/64	3.64e-7	3.10	5.45e-6	2.96	1.66e-5	2.92	1.64e-7	3.95
	1/2	1.14e-2	-	1.89e-2	-	3.47e-2	-	8.45e-3	-
	1/4	5.24e-4	4.44	1.55e-3	3.60	3.99e-3	3.12	4.63e-4	4.19
3	1/8	2.93e-5	4.16	9.97e-5	3.96	2.98e-4	3.74	1.74e-5	4.73
	1/16	1.74e-6	4.08	6.56e-6	3.93	2.09e-5	3.83	6.16e-7	4.82
	1/32	1.07e-7	4.03	4.24e-7	3.95	1.40e-6	3.90	2.07e-8	4.90
	1/2	1.26e-3	-	5.0e-3	-	1.03e-2	-	1.74e-3	-
	1/4	2.40e-5	5.71	1.22e-4	5.36	2.92e-4	5.13	2.52e-5	6.11
4	1/8	4.55e-7	5.72	3.91e-6	4.96	1.09e-5	4.75	4.67e-7	5.75
	1/16	1.07e-8	5.41	1.24e-7	4.98	3.78e-7	4.84	8.11e-9	5.85
	1/32	2.93e-10	5.19	3.93e-9	4.98	1.26e-8	4.91	1.34e-10	5.92

Table 4.1: *Kovasznay flow: history of convergence of the HDG method for $Re = 20$ and $\hat{\tau} = 2$.*

4.2.2   Two-dimensional steady laminar flow around circle

The steady flow around a circle in two dimensions is widely used as a benchmark problem for the incompressible Navier-Stokes equations. The problem has a simple geometry, as depicted in Figure 4.2; a rectangular duct $[0, 2.2] \times [0, 0.41]$ minus a circle

of diameter $D = 0.1$, whose center is placed at $(0.2, 0.2)$. All dimensions are expressed in meters. The geometry and the flow parameters were given in [167], where an overview of various approaches and techniques for the simulation of the laminar incompressible flow around a cylinder in 2D and 3D is presented.

This benchmark test corresponds the case problem named "2D-1" in the afore-mentioned reference. The same flow parameters are prescribed: kinematic viscosity $\nu = 0.001\,\mathrm{m^2/s}$, fluid density $\bar{\rho} = 1.0\,\mathrm{kg/m^3}$, mean velocity at inlet $\bar{U} = 0.2\,\mathrm{m/s}$, which is also the problem's characteristic velocity, therefore the Reynolds number is $Re = \bar{U}D/\nu = 20$. The inflow velocity field is parabolic and has the following analytical expression

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{c} 4U_m \frac{x_2(H-x_2)}{H^2} \\ 0 \end{array} \right\}, \tag{4.19}$$

where H is the duct width $H = 0.41\,\mathrm{m}$ and U_m is the maximum velocity at inlet, at the center of the duct inlet $(x_1 = 0, x_2 = H/2)$. Due to the parabolic velocity profile, the maximum velocity at inlet is related to the mean velocity as $U_m = 3\,\bar{U}/2$. No-slip boundary condition is set on the surface of the circle as well as the upper and lower boundary $(x_2 = 0$ and $x_2 = H)$. At the outlet, a homogeneous Neumann boundary condition is prescribed.
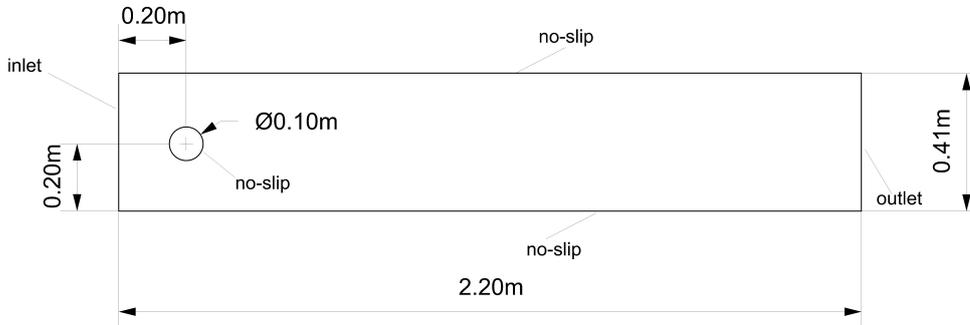
Figure 4.2: *Two-dimensional steady laminar flow around circle: problem geometry and boundary conditions.*

The performance of the numerical methods in this test is evaluated through the calculation of some quantities of interest related to the flow around circle in 2D. These are the lift coefficient C_l and the drag coefficient C_d on the surface of the circle, the pressure drop ΔP between the leading and the trailing point of the circle and, lastly, the length of recirculation L_a.

Following, the definition of these quantities will be given. The lift and drag coefficients are

$$C_l = \frac{2F_\alpha}{\bar{\rho}\bar{U}^2 D}, \qquad C_d = \frac{2F_w}{\bar{\rho}\bar{U}^2 D}, \tag{4.20}$$

(a) $k = 1$ (b) $k = 4$
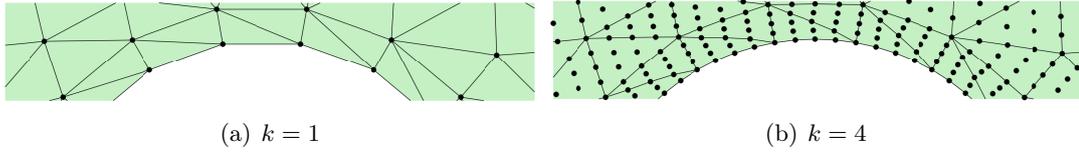
Figure 4.3: *Two-dimensional steady laminar flow around circle: mesh detail near the upper part of the circle, linear mesh versus quartic mesh.*

where F_α and F_w are respectively the lift and drag force that act on the surface S of the circle and $\bar\rho$ the density of the fluid. These forces are calculated with the use of the symmetric stress tensor $\boldsymbol{\sigma}_V$ in the following way

$$F_\alpha = - \int_S \bar\rho\, \mathbf{N}^T \boldsymbol{\sigma}_V \cdot [0, 1]^T \, dS, \qquad F_w = - \int_S \bar\rho\, \mathbf{N}^T \boldsymbol{\sigma}_V \cdot [1, 0]^T \, dS. \qquad (4.21)$$

The minus sign is added in order to account for the force on the surface of the circle, as a reaction to the force on the fluid, while the vectors $[0, 1]^T$ and $[1, 0]^T$ are used with a view to obtaining the x_1 and x_2 components of the force, respectively. It is reminded that the momentum equation was initially divided by the constant density $\bar\rho$, therefore at this point the density has to multiply the symmetric stress tensor in order to obtain force units. As far as the use of symmetric stress tensors is concerned, another advantage is apparent in this test. It is a fact that the lift and drag forces can be evaluated in a straightforward manner, based on the symmetric stress tensor. No complex calculations are needed in order to derive the symmetric part of the gradient of velocity and the symmetric stress tensor $\boldsymbol{\sigma}_V$ already contains all the necessary information for the calculation of the forces on the circle. It is reminded that $\boldsymbol{\sigma}_V$ is defined based on the mixed variable $\boldsymbol{L}$ and the pressure p. Furthermore, the pressure drop and the length of recirculation are

$$\Delta P = \bar\rho \left(p(x_{1,a}, x_{2,a}) - p(x_{1,e}, x_{2,e}) \right) \qquad L_\alpha = x_{1,r} - x_{1,e}, \qquad (4.22)$$

where $(x_{1,a}, x_{2,a}) = (0.15, 0.2)$ and $(x_{1,e}, x_{2,e}) = (0.25, 0.2)$ are the leading edge and the trailing edge, respectively. Once again it should be noted that the pressure variable p, which is calculated from the momentum equation, is already divided by density, hence it must be multiplied by the density in order to obtain pressure units. The x_1-coordinate of the end of the recirculation zone is $x_{1,r}$ and all the coordinates are expressed in meters.

For this experiment, triangular meshes consisting of isoparametric elements of uniform degrees $k = 1$ up to $k = 4$ are considered. If $k \geq 2$, the elements have curved edges wherever it is necessary, for example the elements on the surface of the circle, see also Figure 4.3. Different refinement levels are tested, from coarse meshes, with as few as 607 triangles, to fine ones, with 45,066 triangles.

The velocity and pressure contours are depicted in Figure 4.4 using the fourth refinement level and fourth degree of approximation. Moreover, in Figure 4.5 the

(a) Velocity magnitude contour   (b) Pressure contour
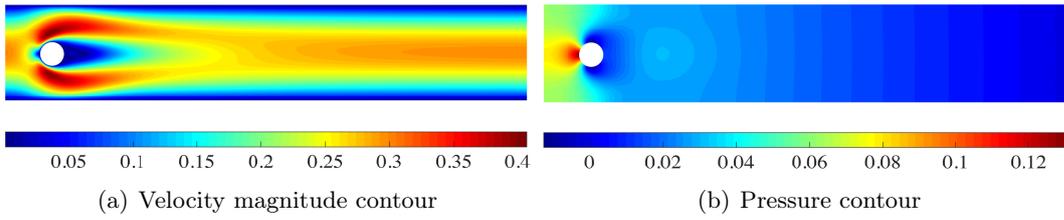
Figure 4.4: *Two-dimensional steady laminar flow around circle: velocity magnitude and pressure contours using the fourth refinement level and fourth degree of approximation, $Re = 20$, $\hat{\tau} = 1$, velocity is expressed in* m/s *and pressure in* Pa.

(a) $k = 1$ (b) $k = 4$
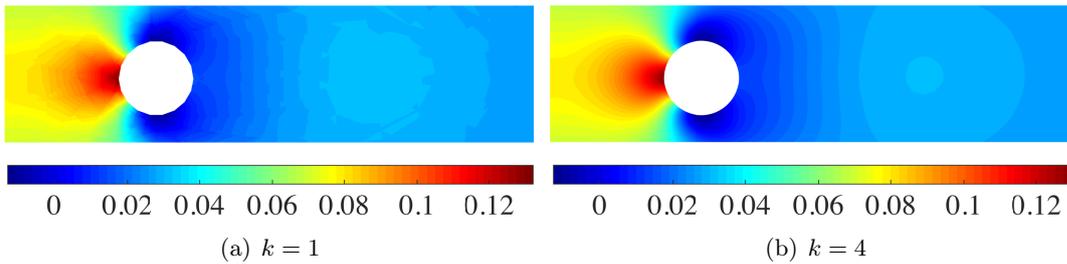
Figure 4.5: *Two-dimensional steady laminar flow around circle: pressure contour in the vicinity of the circle, solution using linear and quartic approximation on the same triangular grid, $Re = 20$, $\hat{\tau} = 1$ and pressure is expressed in* Pa.

pressure contour in a zone near the circle for degree $k = 1$ and $k = 4$ is shown. Although the same number of triangular elements for both approximations is used, the superiority of the quartic approximation in the solution accuracy is obvious.

Table 4.2 summarizes the calculated quantities of interest for the flow around circle in 2D for various degrees of approximation and mesh refinement levels. No exact values for these quantities exist, so the two bounds that were reported in [167] and are based on the results of the research groups that participated in that experiment are given as a reference. One can observe the superiority of high degrees of approximation when it comes to calculating important flow quantities. Cubic and quartic approximations are more accurate with the same and often fewer degrees of freedom than quadratic and, especially, linear approximations.

4.2.3   Flow inside a passive check valve with no moving parts

Passive check valves allow the flow in one direction (forward flow) and prevent the flow in the opposite direction (reverse flow). They are part of many hydro-mechanical systems used in industrial applications, such as pumps and compressors. They have a moving part, usually a door or disc, that opens when the fluid flows in one direction and closes when the flow decreases or is reversed. The most common types of check valves are the ball check valve, the swing check valve and the dual plate check valve. Depending on the application and its technical requirements (e.g. required pressure drop, flow

k	n_{dof}	C_l	C_d	$L_\alpha(m)$	$\Delta P(Pa)$
1	4,119	0.03215	5.4122	0.0822	0.10938
	6,550	0.00368	5.5450	0.0747	0.11795
	20,338	0.00953	5.5583	0.0738	0.11873
	42,499	0.00717	5.5634	0.0855	0.11860
	72,644	0.00823	5.5668	0.0827	0.11835
	152,846	0.01024	5.5655	0.0835	0.11816
	314,982	0.01066	5.5688	0.0850	0.11814
	598,920	0.01061	5.5769	0.0845	0.11761
2	5,875	0.00589	5.5470	0.0817	0.11494
	9,346	0.01027	5.5820	0.0830	0.11744
	29,040	0.01058	5.5797	0.0835	0.11742
	60,695	0.01060	5.5796	0.0842	0.11743
	103,756	0.01060	5.5795	0.0840	0.11745
	218,324	0.01062	5.5795	0.0846	0.11747
	449,940	0.01061	5.5795	0.0847	0.11749
3	7,631	0.01034	5.5764	0.0817	0.11604
	12,142	0.01040	5.5790	0.0830	0.11744
	37,742	0.01063	5.5794	0.0845	0.11746
	78,891	0.01062	5.5795	0.0842	0.11749
	134,868	0.01062	5.5795	0.0840	0.11751
	283,802	0.01062	5.5795	0.0844	0.11751
4	9,387	0.01035	5.5781	0.0824	0.11743
	14,938	0.01060	5.5795	0.0832	0.11752
	48,268	0.01062	5.5795	0.0849	0.11754
	97,087	0.01062	5.5795	0.0842	0.11753
	165,980	0.01062	5.5795	0.0843	0.11752
	lower bound [167]	0.0104	5.5700	0.0842	0.1172
	upper bound [167]	0.0110	5.5900	0.0852	0.1176

Table 4.2: *Two-dimensional steady laminar flow around circle: quantities of interest for various degrees of approximation and degrees of freedom, $Re = 20$ and $\hat{\tau} = 1$.*

rate, operating pressure, temperature, vertical or horizontal direction, transportation length), a different type of valve is used, each one offering certain benefits.

Those valves have also other mechanical components, such as pins, disc arms, springs and balls. All of them suffer from mechanical wear that will eventually lead to failure. Alternatively, a type of check valves with no-moving parts mainly for microfluidic applications has been developed. Micro-electro-mechanical systems (MEMS) often contain microvalves and micropumps which drive small quantities of fluid. Various microfluidics applications are listed by Gravesen and co-workers [168], as well as Shoji and Esashi [169], including flow sensors, separation capillaries and microflow control systems among others. A comprehensive overview on the use of microvalves is presented in [170].

For the purpose of this numerical study, a passive microvalve introduced by N.

(a) Valve geometry

(b) Forward and reverse flow and points A, B where the pressure is measured
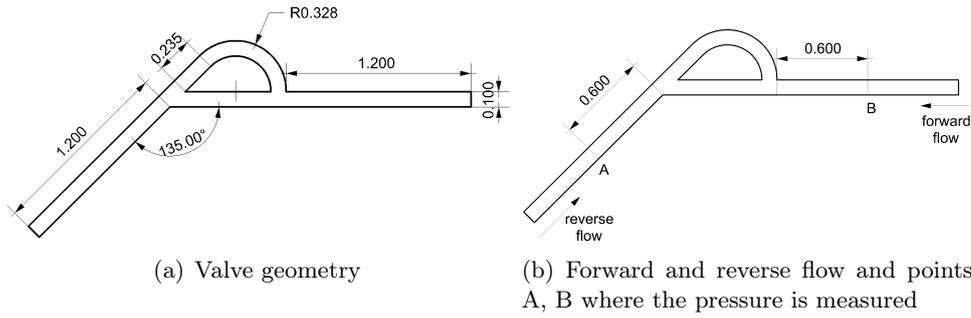
Figure 4.6: *Tesla valve T45-R configuration, dimensions in millimeters.*

Tesla [171] is studied. More precisely, for this numerical experiment the configuration employed in [172] and further studied in [173, 174] will be used. The geometry of the microvalve is shown in Figure 4.6. Based on the geometry, it is intuitively understood that the forward flow is facilitated while the reverse flow is obstructed. From a scientific point of view, when the fluid is flowing in the reverse direction inside this microvalve, then there is high pressure drop due to the flow splitting in the 135°angle (bifurcation), sudden expansion at the end of the upper arc where the upper branch of the flow joins the lower branch, as well as jetting effects appearing in recirculation areas.

The efficiency of this type of microvalves is measured with diodicity, which is the ratio of the pressure drop between inlet and outlet in the reverse direction over the corresponding pressure drop in the forward direction

$$Di = \frac{\Delta P_r}{\Delta P_f}, \tag{4.23}$$

where the subscripts r and f denote respectively the reverse and the forward flow.

The pressure difference is measured between two points, one close to the inlet and one close to the outlet of the valve. These points are A and B, as shown in Figure 4.6(b). Depending on the flow direction, points A and B can be either close to the inlet or close the outlet. The inlet and outlet are extended by 0.6 mm each, so that the inlet flow is fully developed in both forward and reverse directions.

In the current experiment, the working fluid is chosen as in [173], that is, water with dynamic viscosity $\mu = 0.46$ mPa · s and density $\bar{\rho} = 1$ g/cm^3. The Reynolds number varies from 50 to 300, which is the same interval used in [174]. In the same reference it is highlighted that, for these numbers of Reynolds number the flow throughout the valve is steady and laminar, and may be simulated as such. The Reynolds number is controlled through the mean inlet velocity. More specifically, a parabolic velocity profile is prescribed at the inlet, similar to the one described by Equation (4.19). On the walls, the chosen condition involves zero velocity, while at the outlet a zero-traction condition is set.

Velocity magnitude contours for both forward and reverse flow are shown in Figure 4.7. The contours for two Reynolds numbers are depicted, namely 50 and 300. The

(a) $Re = 50$, forward flow

(b) $Re = 50$, reverse flow

(c) $Re = 300$, forward flow

(d) $Re = 300$, reverse flow
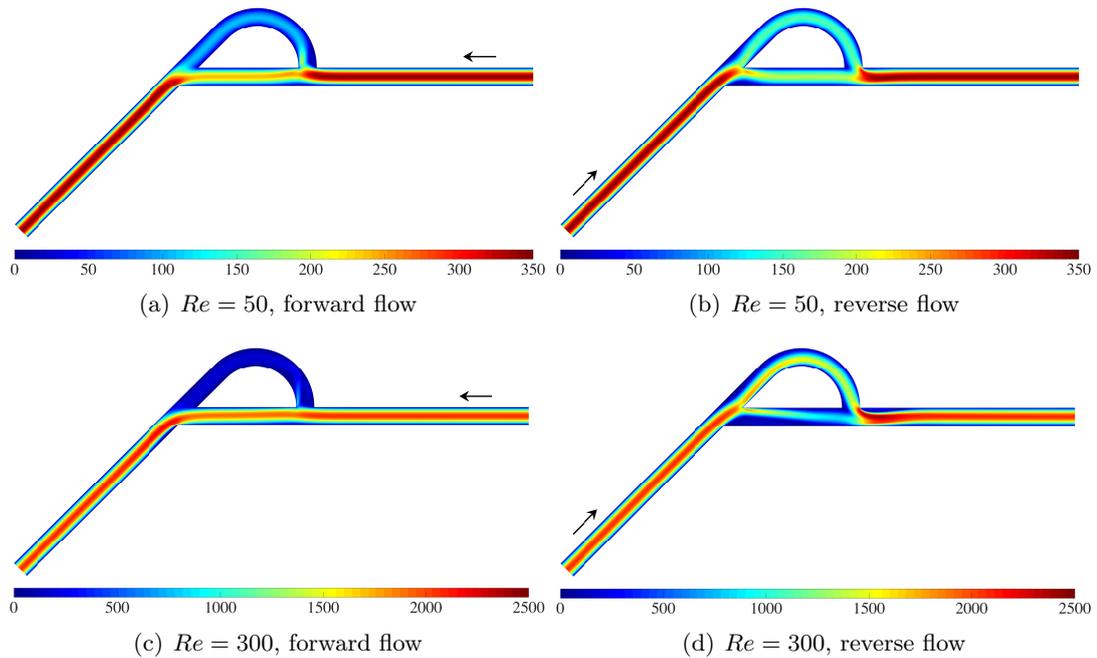
Figure 4.7: *Flow inside a passive check valve: velocity magnitude contours for various Reynolds numbers, forward flow (left) and reverse flow (right), arrows indicate the flow direction, velocity is expressed in* mm/s.

employed mesh contains 5,141 isoparametric triangular elements and the polynomial degree of approximation is three. These elements are capable of approximating precisely the curved branch of the valve. In the forward flow, the velocity is almost zero in the arc of the valve, especially for high Reynolds numbers. The flow bypasses that arc, while in the reverse flow the bifurcation of the flow is visible. Moreover, it is primarily high Reynolds numbers that cause large recirculation zones, especially in reverse flow, see Figure 4.7(d). Higher velocity levels are observed in the reverse flow and are found around the recirculation zones.

Pressure contours are plotted for the same cases in Figure 4.8. The major pressure losses during the reverse flow are due to recirculation effects at the end of the upper arc. As for the forward flow, the highest pressure drop is observed around the 135°corner, which disrupts the flow. Based on the contour plots, it can be deduced that for both flow directions the higher the Reynolds number, the higher the pressure drop.

The streamlines of the flow in the forward direction and Reynolds number 100 and 300 are depicted in the top row of Figure 4.9. In both cases, the recirculation area at the beginning of the the arc, where there is flow separation, is shown. For higher Reynolds number the recirculation region is bigger and, therefore, the flow in the upper branch is obstructed. In the bottom row of the same figure, the vector fields near these regions are plotted.

In the reverse flow, the streamlines for the same Reynolds numbers are shown in Figure 4.10. It is noted that higher Reynolds number lead to bigger recirculation

(a) $Re = 50$, forward flow

(b) $Re = 50$, reverse flow

(c) $Re = 300$, forward flow

(d) $Re = 300$, reverse flow
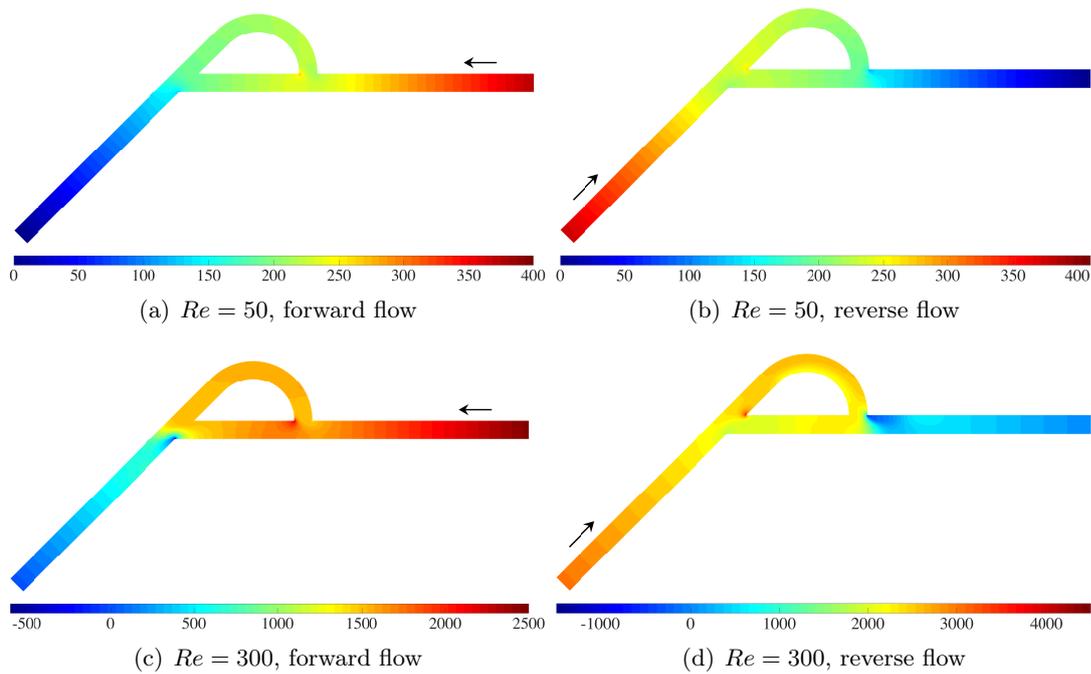
Figure 4.8: *Flow inside a passive check valve: pressure contours for various Reynolds numbers, forward flow (left) and reverse flow (right), arrows indicate the flow direction, pressure is expressed in* Pa.

(a) $Re = 100$, streamlines

(b) $Re = 300$, streamlines

(c) $Re = 100$, detail A

(d) $Re = 300$, detail B
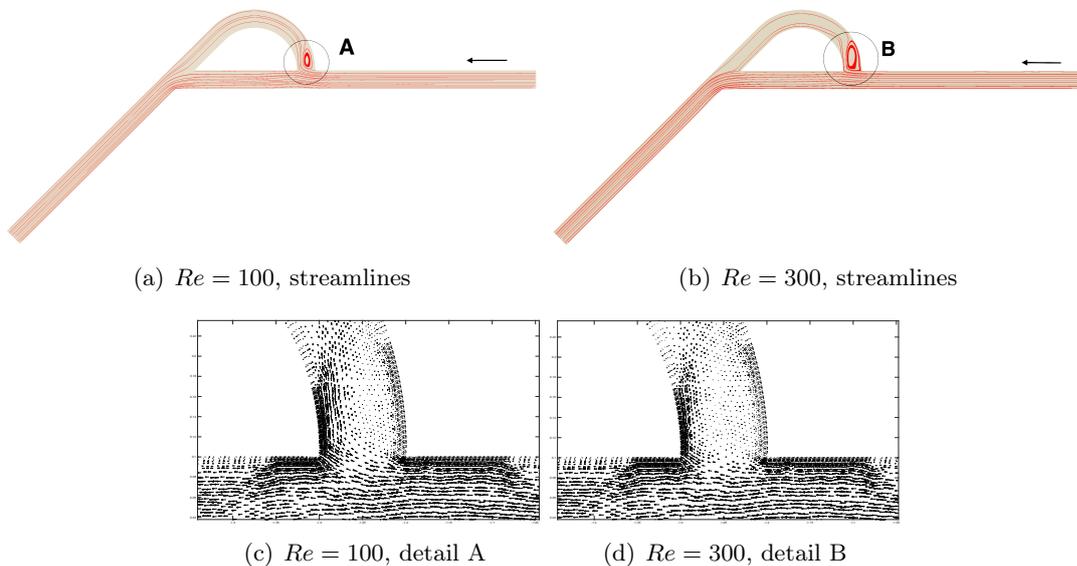
Figure 4.9: *Flow inside a passive check valve: streamlines (up) and vector fields (down) for various Reynolds number, fluid flows in the forward direction.*

area in the 135°angle where the bifurcation takes place. Moreover, for Reynolds 300 a second recirculation area is observed at the end of the arc where the two flows reunite, which leads to jetting effects, increased velocity and, consequently, bigger pressure drop around that region. However, this area disappears for Reynolds 100.

(a) $Re = 100$, streamlines

(b) $Re = 300$, streamlines

(c) $Re = 100$, detail A

(d) $Re = 300$, detail B

(e) $Re = 300$, detail C
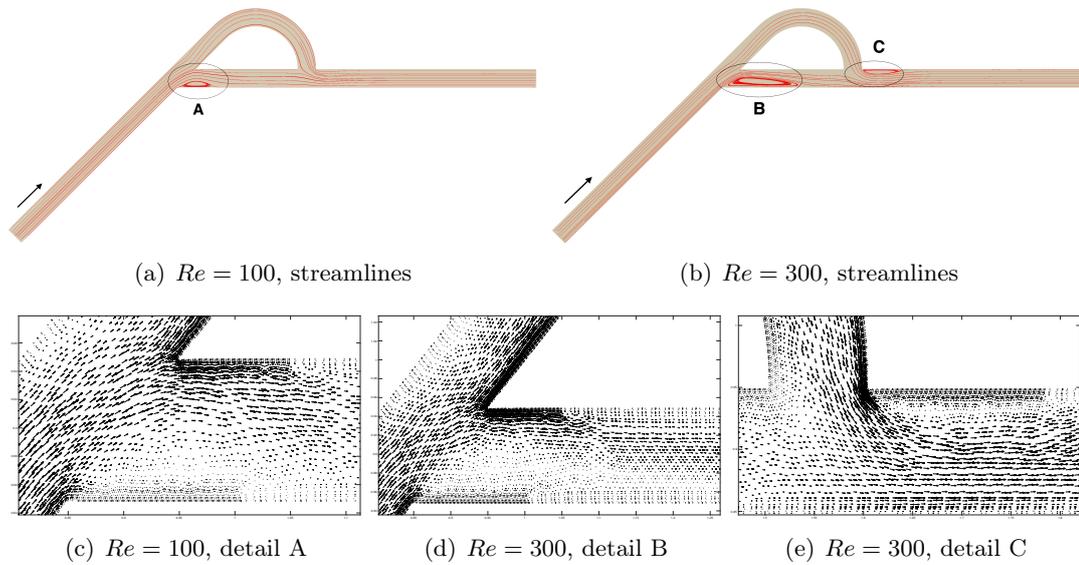
Figure 4.10: *Flow inside a passive check valve: streamlines (up) and vector fields (down) for various Reynolds number, fluid flows in the reverse direction.*
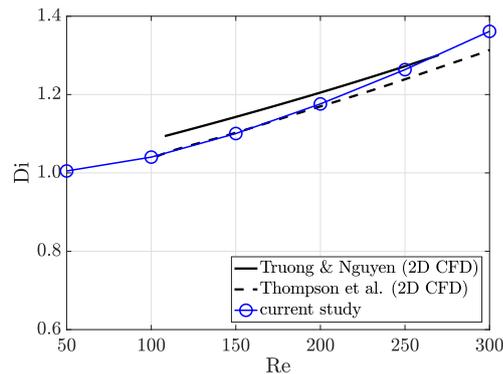
Figure 4.11: *Flow inside a passive check valve: computed diodicity of the Tesla T45-R valve and comparison with other studies.*

As stated before, the effectiveness of this kind of valves is measured though Diodicity. The resulting diodicity is reported in Figure 4.11. The employed mesh consists of 37,865 isoparametric triangular elements of second order and the approximation polynomials are of second order, giving a total 375,762 degrees of freedom to solve. The solution that is obtained using the HDG method is compared with the solutions reported in [173, 174] for the two-dimensional simulation of the flow. The computed diodicity is found to be in agreement with the two solutions reported in literature.

4.3   Degree adaptivity using HDG method for steady Navier-Stokes equations

Making use of the previous superconvergent solution $\mathbf{u}^\star$ it is possible to construct an a posteriori and inexpensive spatial error indicator. The main idea lies in the fact that the post-processed velocity field is of improved accuracy with respect to the original velocity field. For instance, using a polynomial approximation of order k, the obtained velocity and post-processed velocity converge with order $k+1$ and $k+2$ in the $\mathcal{L}_2$ error norm respectively. In this regard, a comparison between these two fields may serve as a local spatial error indicator. In this section, an indicator of the local error in the $\mathcal{L}_2$ norm inside each element Ω_e may be defined as

$$E_e = \left(\frac{1}{A_e} \int_{\Omega_e} (\mathbf{u} - \mathbf{u}^\star)^2 \, d\Omega \right)^{1/2}, \tag{4.24}$$

where A_e is the elemental area. Dividing with the elemental area creates an error indicator that is independent of the varying size of the elements throughout the mesh. This is particularly useful in non-uniform meshes [112]. Similar estimators are defined in [95, 155].

Given a desired accuracy ϵ, for instance 1%, the degree adaptivity process intends to reduce the local spatial error in each element below the desired accuracy, so that

$$E_e < \epsilon \bar{U}, \forall \, \Omega_e \subset \Omega. \tag{4.25}$$

The characteristic velocity of the problem $\bar{U}$ appears in the above equation to ensure consistency of units. It should be noted that the error indicator E_e of Equation (4.24) is expressed in units of velocity. Therefore, $\bar{U}$ is multiplied with the desired accuracy, so that both sides of the inequality (4.25) are expressed in velocity units. The mean velocity at inlet might be chosen as characteristic velocity. For instance, for the flow around a circle in 2D, which was presented in Section 4.2.2, the characteristic velocity was the mean velocity at inlet $\bar{U}$.

The inequality in (4.25) is satisfied by increasing the degree of each element according to the level of the estimated error. In the elements where the error is already below the desired one, the order of approximation might be decreased. This could be particularly useful, especially in regions where a high order of accuracy is not necessary, e.g. farfield flow. In addition, it is guaranteed that the elements that either describe the curved parts of the geometry or have curved faces will not reduce to first-order straight elements.

According to [108] and [95] the increment, also decrement, of the order of an element is described by the following expression

$$\Delta k_e = \left\lceil \frac{\log \left((\epsilon \bar{U})/E_e \right)}{\log (h_e)} \right\rceil, \text{for } e = 1, ..., n_{el}, \tag{4.26}$$

where $\lceil \cdot \rceil$ represents the ceiling function. From a practical point of view, the use of this function guarantees, for instance, that the order will be increased by 1 in those elements where $\epsilon \bar{U} < E_e < 10\epsilon \bar{U}$ and will be decreased by 1 in the elements where $0.01\epsilon \bar{U} < E_e < 0.1\epsilon \bar{U}$. The elements that show an error $0.1\epsilon \bar{U} < E_e < \epsilon \bar{U}$ will not see their degree decreasing. In this way, the error is safely maintained below the desired accuracy.

A different logarithmic base can be used in Equation (4.26), in order to make the change in degree stiffer or easier. For instance, if the value of 100 is chosen as base of the logarithm, instead of 10, then the criterion becomes stiffer. This yields a smoother solution but might also slow down the convergence to the final solution. In this study, the logarithmic base is set to 10.

This process is iterative and stops when all elements satisfy the inequality of Equation (4.25) or when a maximum number of iterations is reached. At each iteration, after the new degree map has been computed, the already computed solution in the mesh of order k_e^{old} is projected to the new mesh of order k_e^{new}. The purpose of this is to use a good approximation as initial guess for the solution of the Navier-Stokes equations and decrease the number of required non-linear (Newton-Raphson) iterations to reach convergence. Algorithm 2 describes the steps of the p-adaptivity process for the steady incompressible Navier-Stokes equations.

Algorithm 2 Degree-adaptivity process for the steady incompressible Navier-Stokes equations using the proposed HDG method

---

 1: Load mesh
 2: Read initial degree map k_e
 3: Read desired accuracy ϵ
 4: Read curved elements
 5: **while** Maximum number of adaptivity iterations i_{max} is not reached AND convergence is not achieved **do**
 6:   $i = i + 1$
 7:   Solve the problem and calculate $\mathbf{L}, \mathbf{u}, \mathbf{p}, \rho, \hat{\mathbf{u}}$
 8:   Calculate the post-processed velocity field $\mathbf{u}^{\star}$
 9:   Estimate the local error E_e using Equation (4.24)
10:   **if** $max(E_e) < \epsilon \bar{U}$ AND $i > 1$ **then**
11:    Convergence achieved
12:   **else**
13:    Calculate the new degree map k_e^{new} using Equation (4.26)
14:    **if** curved elements **then**
15:     minimum degree in curved elements $k = 2$
16:    **end if**
17:    Build new mesh based on k_e^{new}
18:    Project solution $\mathbf{L}, \mathbf{u}, \mathbf{p}, \rho, \hat{\mathbf{u}}$ on the new mesh
19:   **end if**
20: **end while**

4.3.1   Numerical studies

Numerical tests are conducted in order to assess the effectiveness of the proposed p-adaptivity method. The test cases presented in Section 4.2 are considered at this point. First, the Kovasznay flow, being an analytical flow problem, serves as a means to test the ability of the error indicator to identify the regions with high spatial error as well as quantify the error accurately. Next, the method is tested on the steady laminar flow around a circle. This problem involves curved geometry and the calculation of some quantities of interest. Last, the method is applied on a more complicated steady flow, the flow inside a T45-R Tesla valve. The p-adaptivity method is tested whether it can refine appropriately and deliver a reliable approximation, capable of assessing important flow quantities.

Kovasznay flow

The velocity and the pressure field of the Kovasznay flow have an analytical expression, given in Equations (4.17) and (4.18). In this experiment, quadrilateral elements are used, the Reynolds number is $Re = 20$ and the non-dimensional stabilization parameter is chosen $\hat{\tau} = 0.2$, which is a value for which the method converges optimally. Starting from a second-order mesh and quadratic approximation for all the primal and mixed variables and following the procedure described in Algorithm 2, the objective is to find the k_e, so that the estimated error in each element is below the prescribed accuracy.

In Figure 4.12 the degree map, as well as the map of the exact and the estimated error are shown. The process is concluded in two iterations. In the first iteration, the initial mesh is used and the maximum estimated error is approximately 3×10^{-4}, which is bigger than the desired accuracy $\epsilon = 5 \times 10^{-8}$. Then, the degree is refined locally according to the level of the error. The degree is increased to 6 in the elements where the error is very high and 5 in the rest of the elements, as seen in Figure 4.12(d). Next, the estimated error in each element is found to be less than the desired accuracy and thus the process stops. In both iterations, the estimated error is in accordance with the exact error. Both error maps have the same trend and the same levels, even though the proposed indicator slightly overestimates the error.

Two-dimensional laminar flow around circle

The p-adaptivity process will be applied to the steady laminar flow around a circle that is presented in Section 4.2.2. The same geometry, flow properties and boundary conditions are used. The effectiveness of the proposed process for degree adaptation will be evaluated by calculating the quantities of interest of the flow. These are the lift coefficient and the drag coefficient on the surface of the circle, as well as the pressure drop between the leading and the trailing point of the circle.

(a) Iteration 1: degree map   (b) Exact error   (c) Estimated error

(d) Iteration 2: degree map   (e) Exact error   (f) Estimated error
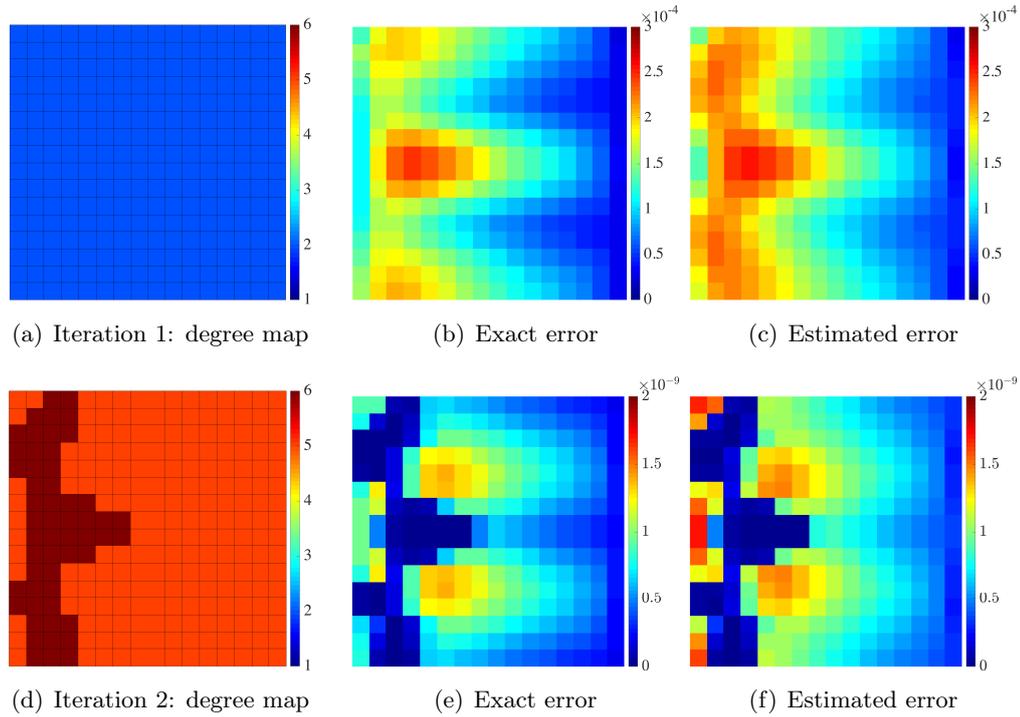
Figure 4.12: *Kovasznay flow: degree adaptivity process using a uniform mesh of 256 quadrilateral elements and initial degree 2, desired accuracy $\epsilon = 5 \times 10^{-8}$, the process stops after 2 iterations.*
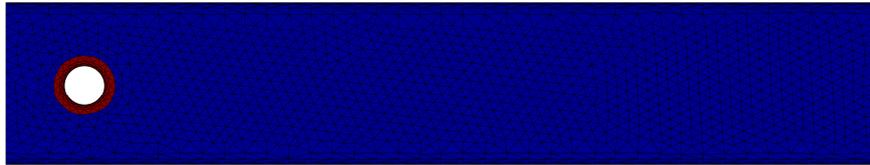
Figure 4.13: *Two-dimensional steady laminar flow around circle: curved elements (red) and straight elements (blue) in a mesh consisting of 2,934 triangular elements.*

Starting from a mesh of a uniform degree $k = 2$ and 2,934 triangular elements, the first step is to label the curved elements. It is reminded that the purpose of this is to ensure that during the adaptivity process these elements will not downgrade to first-order straight elements. In Figure 4.13 the curved elements are plotted in red color, while the straight in blue color. The elements that lie directly on the circle surface and the elements in their vicinity are curved and the rest are straight.

As it was done previously for the Kovasznay flow, the p-adaptivity process receives a desired accuracy ϵ and tries to refine the degree of each element as a means to reach this accuracy. Unlike the mesh used for Kovasznay problem, the computational mesh used to simulate the laminar flow around circle is not uniform. Therefore, the normalization with the elemental area in Equation (4.24) has an effect in this problem. Three different levels of ϵ are used in order to obtain three distinct solutions, each one being of a specific accuracy. In Figure 4.14, each one of the three plots corresponds to a value of ϵ. It is clear that a stricter criterion in general leads to higher polynomial

(a) $\epsilon = 5 \times 10^{-3}$

(b) $\epsilon = 5 \times 10^{-4}$

(c) $\epsilon = 5 \times 10^{-5}$
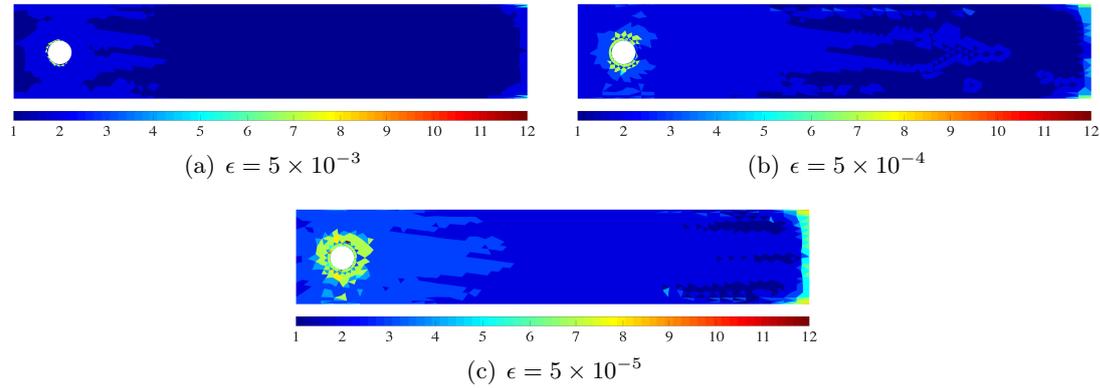
Figure 4.14: *Two-dimensional steady laminar flow around circle: degree map at the end of the p-adaptivity process and for various values of desired accuracy ϵ, $Re = 20$.*

degrees, especially in the region close to the circle and the flow outlet.

Afterwards, the quantities of interest using the mesh of Figure 4.13 and various values of uniform degree of approximation throughout the mesh ($k = 1, ..., 4$) as well as the degree maps of Figure 4.14 are shown in Figure 4.15. They are compared to the value of an overkilled solution, which is the solution on a fine mesh of 45,066 elements and degree $k = 3$, being equivalent to 584,898 degrees of freedom. The effectiveness of the adaptivity process to deliver meshes that can calculate accurately the quantities of interest for fewer degrees of freedom than uniform-degree meshes is apparent. The relative drag coefficient error using the mesh delivered by $\epsilon = 5 \times 10^{-4}$ is 1.44×10^{-5}, lower than the corresponding error of the $k = 3$ uniform mesh, the error of which is 3.34×10^{-5}. It has to be noted that the first mesh uses 24.5% fewer degrees of freedom than the second one. As long as the lift coefficient is concerned, the adapted mesh delivered by $\epsilon = 5 \times 10^{-5}$ calculates the coefficient with 9.64×10^{-5} relative error, less than finest uniform-degree mesh with $k = 4$, whose relative error is 3.94×10^{-4}. It should also be mentioned that the adapted mesh uses 25.2% fewer degrees of freedom than the mesh of uniform $k = 4$. The pressure drop is calculated with a relative error of 1.91×10^{-4} when the resulting adapted mesh of $\epsilon = 5 \times 10^{-5}$ is employed. The $k = 4$ uniform mesh shows approximately the same error 1.51×10^{-4}, yet a bit better than the adapted mesh.

Flow inside a passive check valve with no moving parts

The simulation of the flow inside the Tesla T45-R valve is used to assess the degree-adaptive method. The flow was presented in detail in Section 4.2.3. For the p-adaptivity study, the initial mesh is a relatively coarse mesh of 2,777 triangular elements, evenly distributed throughtout the mesh, as seen in Figure 4.16 and a uniform degree $k = 2$ across all its elements. The reason behind choosing a coarse mesh is to test if by refining only the degree of the elements, the method is capable of capturing important flow details, such as boundary layers, vortices, recirculation zones.

(a) Drag coefficient

(b) Lift coefficient

(c) Pressure drop
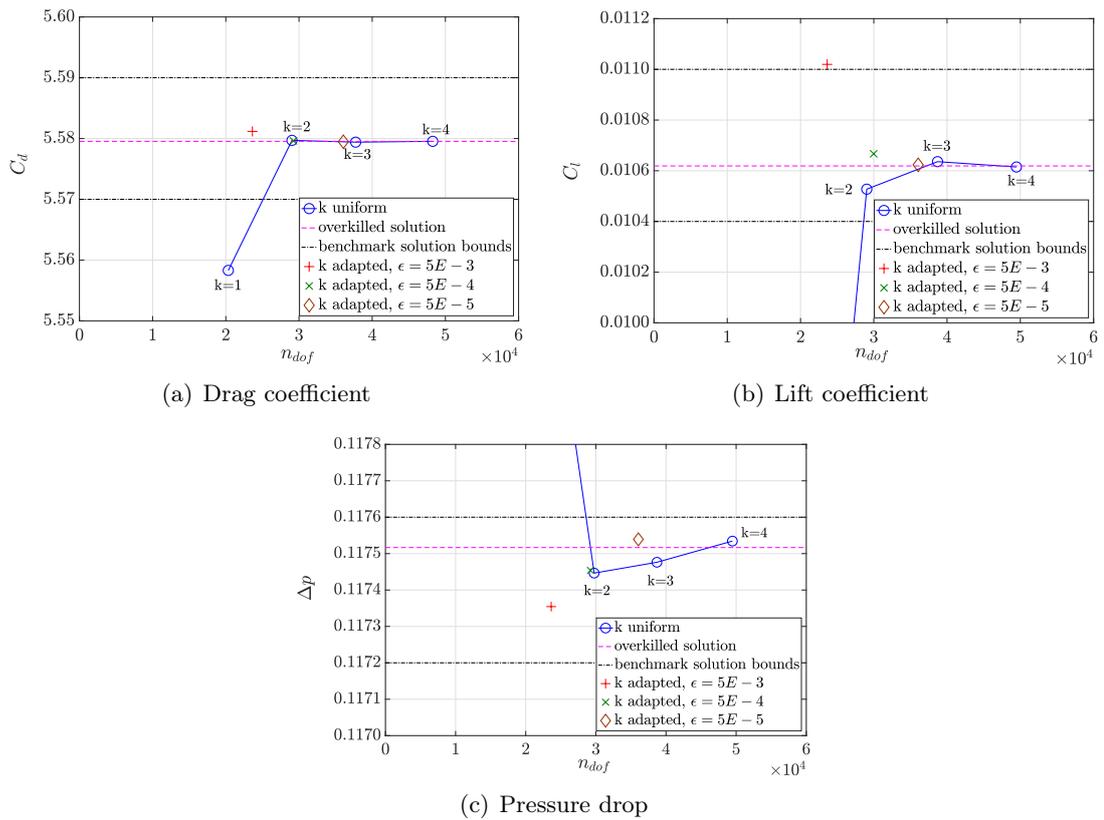
Figure 4.15: *Two-dimensional steady laminar flow around circle: quantities of interest calculated with uniform degree approximations and the degree distributions that arise from the proposed p-adaptivity process, $Re = 20$.*
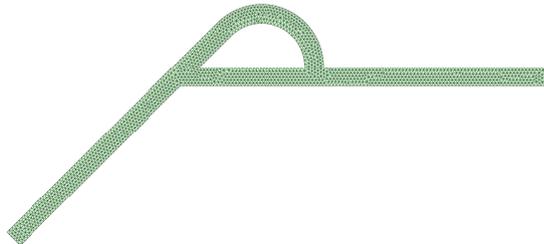
Figure 4.16: *Steady laminar flow inside a T45-R Tesla valve: grid used for the p-adaptivity procedure, consisting of 2,777 triangular elements.*

Starting with this mesh, the adaptive process is performed for two different levels of desired accuracy, the results of which are shown in Table 4.3 for $Re = 300$. Initially, the curved regions of the mesh are identified and then the iterative adaptive process starts. The process is concluded after 3 iterations when the desired accuracy is set to $\epsilon = 5 \times 10^{-2}$. For a stricter criterion $\epsilon = 5 \times 10^{-3}$ and $\epsilon = 5 \times 10^{-4}$ the intermediate solution oscillates around the final solution, therefore the procedure is stopped after 6 iterations.

The quantity of interest for this problem, the diodicity Di, is evaluated in every case and it is compared to a reference solution. The reference solutions is computed using

Mesh degree k	Number of DOFs	Diodicity	Relative error
1	18,545	1.32366	2.77×10^{-2}
2	26,429	1.36924	5.81×10^{-3}
3	34,313	1.36534	2.95×10^{-3}
adaptive, $\epsilon = 5 \times 10^{-2}$	19,665	1.34012	1.56×10^{-2}
adaptive, $\epsilon = 5 \times 10^{-3}$	21,785	1.35074	7.78×10^{-3}
adaptive, $\epsilon = 5 \times 10^{-4}$	27,253	1.36044	6.54×10^{-4}
reference solution	375,762	1.36133	—

Table 4.3: *Steady laminar flow inside a T45-R Tesla valve: calculated diodicity of the valve in meshes with uniform degree $k = 1, 2, 3$ and adaptive degree according to various accuracy criteria ϵ, $Re = 300$.*

a fine mesh and a uniform degree $k = 2$, equal to a total number of 375,762 degrees of freedom. According to Table 4.3, the adaptive procedure with a suitable criterion can generate a mesh and deliver a solution which is more accurate than the solution of a uniform-degree mesh of $k = 3$ and, on top of that, has 20.5% fewer degrees of freedom compared to the latter.

In Figure 4.17 the final distribution of the elemental degree for $Re = 300$ is plotted. Both forward and reverse flows for two levels of desired accuracy are shown. When $\epsilon = 5 \times 10^{-4}$, high-order elements $(k > 2)$ are placed at the corners where the flow experiences sudden expansion or bifurcation, as well as in the recirculation zones. On the contrary, when a looser accuracy is chosen, for instance $\epsilon = 5 \times 10^{-2}$, then only first and second-order elements are placed at these points of interest, leading to less precise solutions. In both cases it is clear that order $k = 2$ has been preserved in the curved elements of the mesh.

In Figure 4.18 the degree maps after the end of the adaptivity process for Re 50 and 300 are depicted. The Reynolds number has an effect on the resulting degree map in that it affects not only the overall velocity inside the valve, but also the recirculation zones around the corners and the bifurcation of the flow. Therefore, around the corners, where the sudden expansion of the flow takes place and recirculation zones are formed, higher-order elements are placed when the Reynolds number is increased. In this manner, larger velocity and pressure gradients can be accurately approximated.

4.4   The HDG method for transient incompressible Navier-Stokes equations

In this section, time-dependent flows using the HDG method with strongly enforced symmetry of the stress tensor are studied. As in previous sections of this chapter, the governing equations are the Navier-Stokes equations, however, in this case accounting for time-dependent terms. First, the weak forms of the local and global problems, that arise using the proposed formulation of the method, are presented. A list of

(a) Forward flow, $\epsilon = 5 \times 10^{-2}$

(b) Reverse flow, $\epsilon = 5 \times 10^{-2}$

(c) Forward flow, $\epsilon = 5 \times 10^{-4}$
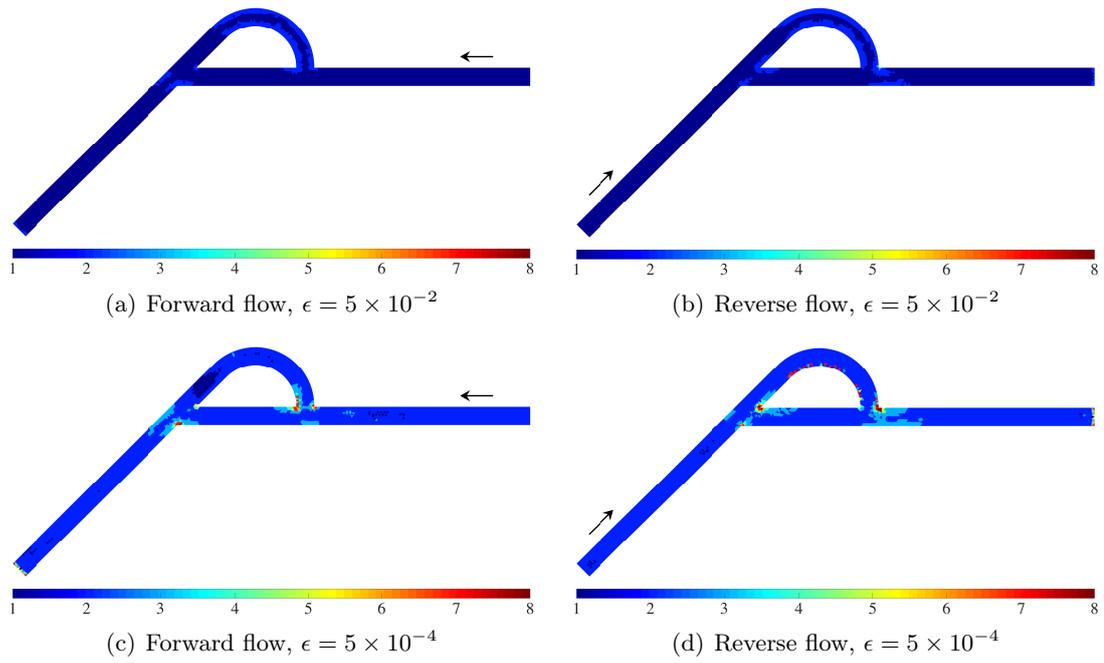
(d) Reverse flow, $\epsilon = 5 \times 10^{-4}$

Figure 4.17: *Steady laminar flow inside a T45-R Tesla valve: effect of desired accuracy ϵ on the degree map for $Re = 150$, forward flow (left) and reverse flow (right), arrows indicate the flow direction.*

(a) Forward flow, $Re = 50$

(b) Reverse flow, $Re = 50$

(c) Forward flow, $Re = 300$

(d) Reverse flow, $Re = 300$
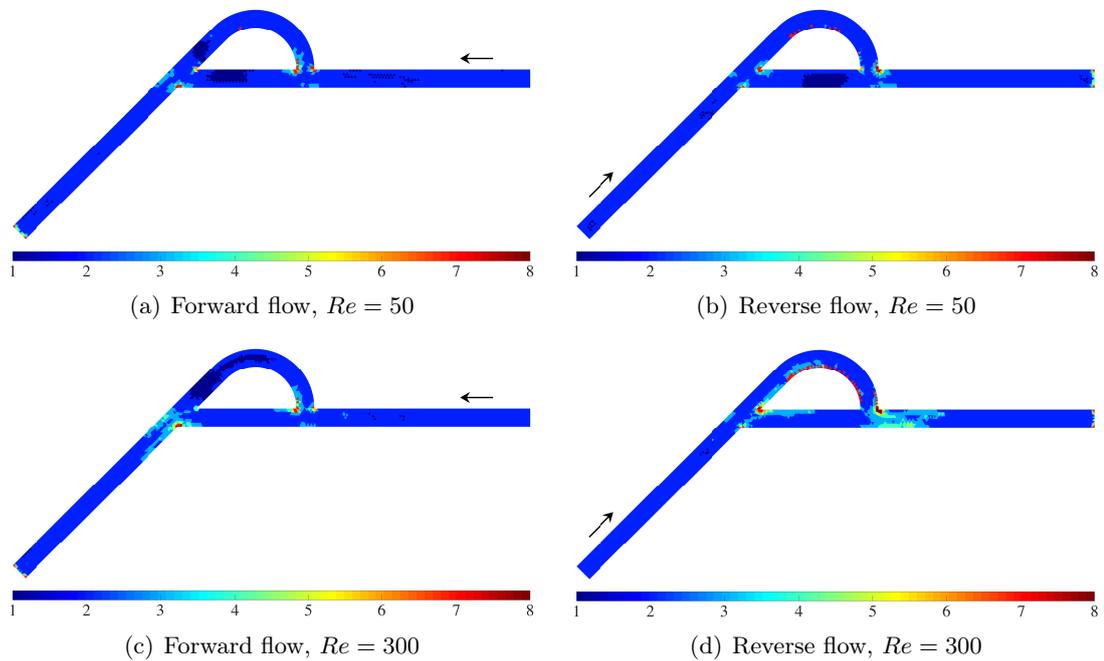
Figure 4.18: *Steady laminar flow inside a T45-R Tesla valve: effect of Reynolds number on the degree map for $\epsilon = 5 \times 10^{-4}$, forward flow (left) and reverse flow (right), arrows indicate the flow direction.*

well-known time-integration schemes and how these are applied in the context of HDG follows. Last, numerical studies are conducted in order to assess the performance of the

various schemes and the ability of the proposed HDG formulation to simulate transient flows.

Weak form of local and global problems

Similar to Section 4.1.2 and after the addition of the temporal term, the discrete weak form of the transient local problems is:

For $e = 1, \ldots, \mathtt{n_{el}}$, given $\boldsymbol{u}_D$ on Γ_D and $\widehat{\boldsymbol{u}}^h$ on $\Gamma \cup \Gamma_N$, find $(\boldsymbol{L}_e^h, \boldsymbol{u}_e^h, p_e^h) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}} \times \mathcal{V}^h(\Omega_e)$ that satisfy

$$-(\boldsymbol{v}, \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_e^h)_{\Omega_e} = \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} \tag{4.27a}$$
$$+ \langle \mathbf{N}_e^T \mathbf{D}^{1/2} \boldsymbol{v}, \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D},$$

$$\left(\boldsymbol{w}, \frac{\partial \boldsymbol{u}_e^h}{\partial t}\right)_{\Omega_e} + (\boldsymbol{w}, \boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{D}^{1/2} \boldsymbol{L}_e^h)_{\Omega_e} + (\boldsymbol{w}, \boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{E} p_e^h)_{\Omega_e} - (\boldsymbol{\nabla}_{\mathrm{S}} \boldsymbol{w}, (\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V)_{\Omega_e}$$
$$+ \langle \boldsymbol{w}, \tau \boldsymbol{u}_e^h \rangle_{\partial\Omega_e} = (\boldsymbol{w}, \boldsymbol{b})_{\Omega_e} + \langle \boldsymbol{w}, (\tau \widehat{\boldsymbol{u}}^h - \mathbf{N}_e^T (\widehat{\boldsymbol{u}} \otimes \widehat{\boldsymbol{u}})_V) \rangle_{\partial\Omega_e \setminus \Gamma_D} \tag{4.27b}$$
$$+ \langle \boldsymbol{w}, (\tau \boldsymbol{u}_D - \mathbf{N}_e^T (\boldsymbol{u}_D \otimes \boldsymbol{u}_D)_V) \rangle_{\partial\Omega_e \cap \Gamma_D},$$

$$(\boldsymbol{\nabla}_{\mathrm{S}}^T \mathbf{E} q, \boldsymbol{u}_e^h)_{\Omega_e} = \langle q, \mathbf{E}^T \mathbf{N}_e \boldsymbol{u}_D \rangle_{\partial\Omega_e \cap \Gamma_D} + \langle q, \mathbf{E}^T \mathbf{N}_e \widehat{\boldsymbol{u}}^h \rangle_{\partial\Omega_e \setminus \Gamma_D}, \tag{4.27c}$$

$$\frac{1}{|\partial\Omega_e|} \langle p_e^h, 1 \rangle_{\partial\Omega_e} = \rho_e^h, \tag{4.27d}$$

for all $(\boldsymbol{v}, \boldsymbol{w}, q) \in [\mathcal{V}^h(\Omega_e)]^{\mathtt{m_{sd}}} \times [\mathcal{V}^h(\Omega_e)]^{\mathtt{n_{sd}}} \times \mathcal{V}^h(\Omega_e)$.

The temporal variable t (not to be confused with the traction vector $\boldsymbol{t}$) counts from 0 till a final time t^{end}. The final form of the discrete weak global problem coincides with that of the steady incompressible Navier-Stokes, refer to Equation (4.15).

4.4.1   Time integration schemes

Given the weak forms of the local and global problems previously described, the "Method of Lines" is used for the temporal discretization of the system. The following system of differential algebraic equations (DAE) is obtained, namely

$$\mathbf{M}\dot{\mathbf{U}} + \mathbf{K}(\mathbf{U})\mathbf{U} = \mathbf{f}(t, \mathbf{U}) \tag{4.28}$$

where $\mathbf{U} = \begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} & \zeta & \widehat{\mathbf{u}} & \rho \end{bmatrix}^T$ and

$$\mathbf{M} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{uu} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \mathbf{A}_{LL} & \mathbf{A}_{Lu} & \mathbf{0} & \mathbf{0} & \mathbf{A}_{L\widehat{u}} & \mathbf{0} \\ \mathbf{A}_{uL} & \mathbf{A}_{uu} & \mathbf{A}_{up} & \mathbf{0} & \mathbf{A}_{\widehat{u}\widehat{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{pu} & \mathbf{0} & \mathbf{A}_{\rho p}^T & \mathbf{A}_{p\widehat{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{\rho p} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{A}_{\widehat{u}L} & \mathbf{A}_{\widehat{u}u} & \mathbf{A}_{\widehat{u}p} & \mathbf{0} & \mathbf{A}_{\widehat{u}\widehat{u}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_{\rho\widehat{u}} & \mathbf{0} \end{bmatrix}. \tag{4.29}$$

The aforementioned components of the matrices arise from the discretization of the weak form of the Equations (4.27) and (4.15). More details on their definition are given in Appendix A.3. Following, a number of time integration schemes for the temporal discretization of Equation (4.28) will be presented. These include single-step methods, multiple-step methods and multi-stage methods of Runge-Kutta type. It should be mentioned that in this thesis only implicit schemes are studied.

Single-step methods

A popular family of methods for parabolic problems are the θ-methods. Applying to Equation (4.28) the following is obtained

$$\mathbf{M}\big(t^{n+\theta}, \mathbf{U}^{n+\theta}\big)\dot{\mathbf{U}} + \mathbf{K}\big(t^{n+\theta}, \mathbf{U}^{n+\theta}\big)\mathbf{U}^{t^{n+\theta}} = \boldsymbol{f}\big(t^{n+\theta}, \mathbf{U}^{n+\theta}\big), \tag{4.30}$$

where θ is a number in the interval $(0, 1)$ and

$$
\begin{aligned}
t^{n+\theta} &= (1 - \theta)\, t^n + \theta\, t^{n+1} \\
\mathbf{U}^{n+\theta} &= (1 - \theta)\, \mathbf{U}^n + \theta\, \mathbf{U}^{n+1}.
\end{aligned}
\tag{4.31}
$$

Therefore, the system of equation discretized with θ-method is written as:

$$
\begin{aligned}
\mathbf{M}\frac{\mathbf{U}^{n+1}}{\Delta t} &+ \theta\,\mathbf{K}\big((1-\theta)\,\mathbf{U}^n + \theta\,\mathbf{U}^{n+1}\big)\mathbf{U}^{n+1} = \mathbf{M}\frac{\mathbf{U}^n}{\Delta t} \\
&- (1-\theta)\,\mathbf{K}\big((1-\theta)\,\mathbf{U}^n + \theta\,\mathbf{U}^{n+1}\big)\mathbf{U}^n + (1-\theta)\,\boldsymbol{f}\big(t^n, \mathbf{U}^n\big) + \theta\,\boldsymbol{f}\big(t^{n+1}, \mathbf{U}^{n+1}\big).
\end{aligned}
\tag{4.32}
$$

When $\theta = 1$, then θ-method describes the backward Euler method (BE), while $\theta = 0.5$, then it describes the Crank-Nicolson method (CN). The backward Euler is first-order accurate in time and the Crank-Nicolson is second-order accurate. The order of temporal convergence is an important property of time integration schemes.

Another important property is *A-stability*. It is reminded a scheme is A-stable when its region of stability lies entirely on the left half of the complex plane. Explicit methods cannot be A-stable, unless the employed timestep complies with a stability condition, the Courant–Friedrichs–Lewy (CFL) condition. On the other hand, most implicit methods are unconditionally A-stable, meaning that they have A-stability regardless the chosen timestep. Both backward Euler and Crank-Nicolson are unconditionally A-stable.

Nevertheless, A-stability alone is not sufficient, yet time-integration schemes should also have *stiff decay*. To define the concept of stiff decay we consider the generalized test problem, see also [175]

$$y' = \lambda\big(y - g(t)\big), \qquad 0 < t < b, \tag{4.33}$$

where $g(t)$ is a bounded but arbitrary function. As the real part of λ, $Re(\lambda) \to -\infty$, the exact solution of Equation (4.33) satisfies

$$y(t) \rightarrow g(t), \qquad 0 < t < b. \tag{4.34}$$

A scheme has then stiff decay when for a fixed $0 < t_n < b$

$$|y_n - g(t_n)| \rightarrow 0 \qquad as \qquad \Delta t_n Re(\lambda) \rightarrow -\infty. \tag{4.35}$$

The backward Euler method has stiff decay, whereas the Crank-Nicolson method does not. This makes the latter exhibit an oscillatory behaviour when large timesteps are employed.

Second-order backward differentiation formula

By using the already computed velocity field of the two previous time levels, it is possible to obtain a second-order time integration scheme, namely the second-order backward differentiation formula (BDF2). Applying this method to Equation (4.28) yields

$$\mathbf{M}(t^{n+1}, \mathbf{U}^{n+1}) \frac{3\mathbf{U}^{n+1} - 4\mathbf{U}^n + \mathbf{U}^{n-1}}{2\Delta t} + \mathbf{K}(t^{n+1}, \mathbf{U}^{n+1})\mathbf{U}^{t^{n+1}} = \boldsymbol{f}(t^{n+1}, \mathbf{U}^{n+1}) \Leftrightarrow$$
$$\mathbf{M}\frac{3\mathbf{U}^{n+1}}{2\Delta t} + \mathbf{K}(t^{n+1}, \mathbf{U}^{n+1})\mathbf{U}^{t^{n+1}} = \mathbf{M}\frac{4\mathbf{U}^n - \mathbf{U}^{n-1}}{2\Delta t} + \boldsymbol{f}(t^{n+1}, \mathbf{U}^{n+1})$$
.
$$\tag{4.36}$$

This method is not self-starting, so the first time level t^1 is calculated with another method, for example the backward Euler method. Following, the second time level t^2 can be calculated using the solution at t^1 and the initial solution at t^0.

The second-order backward differentiation formula is unconditionally A-stable and has stiff decay. It should be also mentioned that higher-order backward differentiation formulas exist, however unlike BDF2, they are not unconditionally stable and therefore will not be studied.

Runge-Kutta methods

The Runge–Kutta (RK) methods are a family of multi-stage methods that can be explicit, implicit or semi–implicit. The Runge-Kutta methods are described by a table of coefficients, usually referred to as Butcher-tableau, which has the following structure for a method of q stages

c_1	α_{11}	α_{12}	...	α_{1q}
c_2	α_{21}	α_{22}	...	α_{2q}
...	....	...	...	...
c_q	α_{q1}	α_{q2}	...	α_{qq}
	b_1	b_2	...	b_q

Table 4.4: *Butcher tableau of a Runge-Kutta method.*

The method is explicit when the coefficient matrix α_{ij} is a lower-triangular matrix. On the other hand, it is fully implicit when the same matrix is full. As far as A-stability is concerned, explicit Runge-Kutta are not unconditionally A-stable, while implicit Runge-Kutta are. More specifically, the stability function of a Runge-Kutta method is

$$R(z) = \frac{\det(\mathbf{I} - z\mathbf{A} + z\mathbf{1}_q \boldsymbol{b}^T)}{\det(\mathbf{I} - z\mathbf{A})}, \tag{4.37}$$

where $\mathbf{A}$ is a matrix that contains the elements α_{ij}, $\mathbf{1}_q$ a vector of ones of size q and $\boldsymbol{b}$ a vector that contains the elements b_i.

Furthermore, an A-stable Runge-Kutta method is also L-stable when

$$\lim_{Re(z)\to-\infty} R(z) = 0. \tag{4.38}$$

L-stability is a desirable property of Runge-Kutta methods.

Applying a Runge-Kutta method to Equation (4.28), the following multi-stage method is obtained. For $i = 1, ..., q$ the solution in the intermediate stage i is calculated by solving

$$\mathbf{M}\frac{\mathbf{U}^{n+b_i} - \mathbf{U}^n}{\Delta t} + \sum_{j=1}^{q} \alpha_{ij}\Big(\mathbf{K}\big(t^n + c_j\Delta t, \mathbf{U}^{n+b_j}\big)\mathbf{U}^{n+b_j} - \boldsymbol{f}\big(t^n + c_j\Delta t, \mathbf{U}^{n+b_j}\big)\Big) = \mathbf{0},$$
$$\tag{4.39}$$

while the advance to the next time level $n+1$ is performed by solving

$$\mathbf{M}\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} + \sum_{i=1}^{q} b_i\Big(\mathbf{K}\big(t^n + c_i\Delta t, \mathbf{U}^{n+b_i}\big)\mathbf{U}^{n+b_i} - \boldsymbol{f}\big(t^n + c_i\Delta t, \mathbf{U}^{n+b_i}\big)\Big) = \mathbf{0}. \tag{4.40}$$

It should be noted that the implicit method is computationally expensive, since at every stage i the solutions of all the stages $i = 1, ..., q$ are involved.

Singly-diagonally implicit Runge-Kutta methods

In order to circumvent the increased complexity of fully implicit Runge-Kutta methods, diagonally-implicit Runge-Kutta (DIRK) methods are usually used in practice. Referring to Table 4.4, it is possible to obtain a diagonally-implicit method when the elements a_{ij} are zero for $i > j$. Moreover, if the same diagonal element α_{11} is used, then the method is called singly-diagonally-implicit Runge-Kutta (SDIRK) method. The Butcher tableau of an SDIRK method has the form given in Table 4.5.

c_1	α_{11}			
c_2	α_{21}	α_{11}		
...	....	...	...	
c_q	α_{q1}	α_{q2}	...	α_{11}
	b_1	b_2	...	b_q

Table 4.5: *Butcher tableau of a singly-diagonally-implicit Runge-Kutta method of q stages.*

The Equation (4.39), for the solution at each stage i, can be rewritten as

$$\mathbf{M}\frac{\mathbf{U}^{n+b_i} - \mathbf{U}^n}{\alpha_{11}\Delta t} + \mathbf{K}\big(t^n + c_i\Delta t, \mathbf{U}^{n+b_i}\big)\mathbf{U}^{n+b_i} = \boldsymbol{f}\big(t^n + c_i\Delta t, \mathbf{U}^{n+b_i}\big) - \boldsymbol{g}^i, \quad (4.41)$$

where

$$\boldsymbol{g}^i = \frac{1}{\alpha_{11}}\sum_{j=1}^{i-1}\alpha_{ij}\Big(\mathbf{K}\big(t^n + c_j\Delta t, \mathbf{U}^{n+b_j}\big)\mathbf{U}^{n+b_j} - \boldsymbol{f}\big(t^n + c_j\Delta t, \mathbf{U}^{n+b_j}\big)\Big). \quad (4.42)$$

It is therefore clear that every stage of SDIRK is equivalent to backward Euler and their computational cost is similar. Typically, however, the computational cost of solving the non-linear problem with SDIRK is lower than solving it with backward Euler due to the better initial approximation that leads to fewer non-linear iterations.

The stability function of an SDIRK method has the expression

$$R(z) = \frac{\mathbf{I} - z\mathbf{A} + z\mathbf{1}_q\boldsymbol{b}^T}{(1 - z\alpha_{11})^q}, \quad (4.43)$$

Moreover, if the last row of $\mathbf{A}$ coincides with $\boldsymbol{b}$, the SDIRK method is L-stable.

In [176] a second-order SDIRK method of two stages (SDIRK2) is presented, the Butcher tableau of which is

$$
\begin{array}{c|cc}
\gamma & \gamma & \\
1 & 1-\gamma & \gamma \\
\hline
 & 1-\gamma & \gamma
\end{array}
$$

Table 4.6: *Butcher tableau of a second-order singly-diagonally-implicit Runge-Kutta method of two stages (SDIRK2).*

The method is L-stable, stiffly-accurate and the second-order conditions yield values of $\gamma = (2 \pm \sqrt{2})/2$. However, the value $\gamma = (2 - \sqrt{2})/2$ is better because it is less than 1.

In addition, an L-stable, stiffly-accurate SDIRK method of third-order with three stages (SDIRK3) has the following table of coefficients

$$
\begin{array}{c|ccc}
\gamma & \gamma & & \\
c_2 & c_2 - \gamma & \gamma & \\
1 & b_1 & b_2 & \gamma \\
\hline
 & b_1 & b_2 & \gamma
\end{array}
$$

Table 4.7: *Butcher tableau of a third-order singly-diagonally-implicit Runge-Kutta method of three stages (SDIRK3).*

In order for the method to be L-stable, the value of γ has to be the solution of the root of the polynomial $\gamma^3 - 3\gamma^2 + \frac{3}{2}\gamma - \frac{1}{6} = 0$ in the domain $\left[\frac{1}{6}, \frac{1}{2}\right]$, that is $\gamma = 0.4358665215084590$. The other coefficients are defined after γ as

$$c_2 = \frac{2 - 9\gamma + 6\gamma^2}{3\alpha}, \qquad b_1 = \frac{4\gamma - 1}{4\beta}, \qquad b_2 = \frac{-3\gamma^2}{4\beta}, \tag{4.44}$$

where $\alpha = 1 - 4\gamma + 2\gamma^2$ and $\beta = -1 + 6\gamma - 9\gamma^2 + 3\gamma^3$.

Singly-diagonally implicit Runge-Kutta methods with an explicit first stage

It is possible for an SDIRK method to have an explicit first stage, thus increasing the stage order from 1, that holds for SDIRK methods, to 2. These methods are called singly-diagonally implicit Runge-Kutta methods with an explicit first stage (ESDIRK). Such time-marching methods are presented in [177]. One of them is an L-stable third-order ESDIRK method of four stages (referred to ESDIRK3/2 by Kværnø), the coefficients of which are given in Table 4.8.

0	0	0	0	0
2γ	γ	γ	0	0
1	α_{31}	α_{32}	γ	0
1	α_{41}	α_{42}	α_{43}	γ

Table 4.8: *Butcher tableau of a third-order singly-diagonally-implicit Runge-Kutta method of four stages with explicit first stage (ESDIRK3/2).*

The parameter γ is chosen based on stability properties and its value is $\gamma = 0.435866521508$, whereas the rest of coefficients are equal to

$$\begin{aligned}
\alpha_{31} &= \frac{-4\gamma^2 + 6\gamma - 1}{4\gamma}, \qquad \alpha_{32} = \frac{-2\gamma + 1}{4\gamma}, \\
\alpha_{41} &= \frac{6\gamma - 1}{12\gamma}, \qquad \alpha_{42} = \frac{-1}{(24\gamma - 12)\gamma}, \qquad \alpha_{43} = \frac{-6\gamma^2 + 6\gamma - 1}{6\gamma - 3}.
\end{aligned} \tag{4.45}$$

It should be mentioned that although the method has initially four stages, in reality only three of them are computationally expensive due to the fact that the first stage is explicit and, therefore, is computationally inexpensive.

Observing the third and the fourth rows of Table 4.8, it is seen that the solution for both stages is calculated at time $n + 1$. The last two stages can be used in order to construct a temporal error estimator, based on which it is possible to control the size of the timestep during the simulation. Such methods are also called embedded methods. This idea will be discussed in detail in Section 4.5.2.

An L-stable fourth-order ESDIRK method of six stages (referred to as ESDIRK46) is presented in [176], the Butcher tableau of which is Table 4.9, and where the last row refers to the optional stage that is used in timestep size adaptivity. The parameter γ is chosen based on stability properties and its value is $\gamma = 0.25$, whereas the rest of coefficients are equal to

$$c_2 = \frac{1}{2}, \quad c_3 = \frac{2 - \sqrt{2}}{4}, \quad c_4 = \frac{5}{8}, \quad c_5 = \frac{26}{25},$$

$$\alpha_{21} = 0.25, \quad \alpha_{31} = \frac{1 - \sqrt{2}}{8}, \quad \alpha_{32} = \frac{1 - \sqrt{2}}{8}, \quad \alpha_{41} = \frac{5 - 7\sqrt{2}}{64}, \quad \alpha_{42} = \frac{5 - 7\sqrt{2}}{64},$$

$$\alpha_{43} = \frac{7 + 7\sqrt{2}}{32}, \quad \alpha_{51} = \frac{-13796 - 54539\sqrt{2}}{125000}, \quad \alpha_{52} = \frac{-13796 - 54539\sqrt{2}}{125000},$$

$$\alpha_{53} = \frac{506605 + 132109\sqrt{2}}{437500}, \quad \alpha_{54} = \frac{166(-97 + 376\sqrt{2})}{109375}, \quad b_1 = \frac{1181 - 987\sqrt{2}}{13782},$$

$$b_2 = \frac{1181 - 987\sqrt{2}}{13782}, \quad b_3 = \frac{47(-267 + 1783\sqrt{2})}{273343}, \quad b_4 = \frac{-16(-22922 + 3525\sqrt{(2)})}{571953},$$

$$b_5 = \frac{-15625(97 + 376\sqrt{2})}{90749876}, \quad \hat{b}_1 = \frac{-480923228411}{4982971448372}, \quad \hat{b}_2 = \frac{-480923228411}{4982971448372},$$

$$\hat{b}_3 = \frac{6709447293961}{12833189095359}, \quad \hat{b}_4 = \frac{3513175791894}{6748737351361}, \quad \hat{b}_5 = \frac{-498863281070}{6042575550617},$$

$$\hat{b}_6 = \frac{2077005547802}{8945017530137}.$$

$$(4.46)$$

0	0	0	0	0	0	0
c_2	α_{21}	γ	0	0	0	0
c_3	α_{31}	α_{32}	γ	0	0	0
c_4	α_{41}	α_{42}	α_{43}	γ	0	0
c_5	α_{51}	α_{52}	α_{53}	α_{54}	γ	0
1	b_1	b_2	b_3	b_4	b_5	γ
$\hat{b}_i$	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$	$\hat{b}_4$	$\hat{b}_5$	$\hat{b}_6$

Table 4.9: *Butcher tableau of a fourth-order singly-diagonally-implicit Runge-Kutta method of six stages with explicit first stage (ESDIRK46).*

4.4.2   Numerical studies

At this point, various tests using the aforementioned time-integration methods in the HDG context will be conducted. First, the performance of the methods will be assessed by evaluating their order of convergence and by comparing the level of temporal error for the same amount of computational resources. Both tests are performed considering an analytical flow field, therefore it is possible to estimate the temporal error. Later, the various temporal methods will be used for the simulation of the transient two-dimensional flow around a circle, a well-known benchmark problem for transient laminar incompressible Navier-Stokes equations.

Performance of time-integration schemes

At this point, the performance of the time-integration schemes presented in Section 4.4.1 will be assessed by studying the following:

(a) u_1 (b) u_2 (c) p
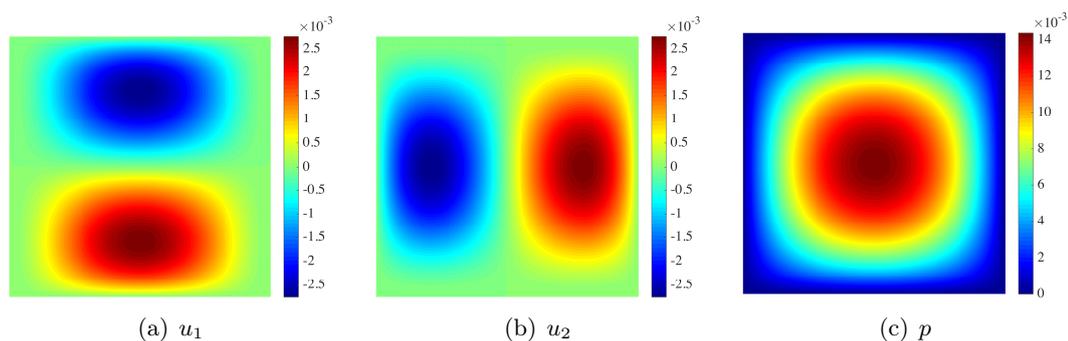
Figure 4.19: *Transient analytical flow: HDG approximation of the velocity and pressure field using the fourth refinement of the quadrilateral mesh at time $t = 1$, BDF2 is used for time integration, domain $\Omega = [0, 1]^2$ and $\hat{\tau} = 0.3$.*

1. the order of temporal convergence of the various schemes;

2. the level of temporal error of the various schemes for the same computational effort.

For both studies, an analytical flow is used with the following transient velocity and pressure profiles:

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{c} x_1^2(1 - x_1)^2(2x_2 - 6x_2^2 + 4x_2^3)\sin^2(t) \\ -x_2^2(1 - x_2)^2(2x_1 - 6x_1^2 + 4x_1^3)\sin^2(t) \end{array} \right\}, \tag{4.47}$$

$$p(\boldsymbol{x}) = x_1(1 - x_1)x_2(1 - x_2)\sin^2(t). \tag{4.48}$$

In Figure 4.19 the velocity components and the pressure are plotted at time $t = 1$. The chosen spatial domain is $\Omega = [0, 1]^2$ and the time marching is performed using the BDF2 method and timestep size $\Delta t = 0.03125$.

Order of temporal convergence

A convergence study is performed in order to assess the accuracy and convergence rate of the time-integration schemes when these are used with the proposed HDG method. For this study, the source term and the boundary conditions will be chosen in such way so that the flow is described by the aforementioned analytical solution, while the chosen time interval is $[0, 3]$. The simulation is performed with different timestep sizes and the $\mathcal{L}_2$-norm of the error of the variables $\boldsymbol{u}, p, \boldsymbol{L}$ is calculated at the final time $t^{end} = 3$.

At this point it has to be highlighted that the computed error contains two components; a spatial and a temporal one, and in order to ensure that the measured error represents the temporal part, the spatial part should be tiny. This is achieved by using a fine computational mesh, both in mesh size h and mesh degree p. In this study so far, it has been demonstrated that, using the proposed formulation of the HDG method, the

$\mathcal{L}_2$-norm of the spatial error of velocity, pressure and mixed variable have the following general expression

$$||E_{\mathcal{L}_2(\Omega)}|| = ch^{k+1}, \qquad k \geq 1$$

where c is a constant, h is the mesh size and k the degree of approximation. For example, the spatial error arising form a fourth-order mesh of uniform size $1/32$ would be approximately $2.9e-8$, without accounting for the constant c. Such levels of spatial error are found to be low and do not affect the temporal convergence study.

In Figure 4.20, the order of convergence of the various temporal methods for the analytical problem described in Equations (4.47) and (4.48) is displayed. The $\mathcal{L}_2$-norm of the error of velocity, pressure and mixed variable (scaled strain rate tensor) versus the timestep size is plotted. Each method shows a constant behaviour for all variables. The backward Euler method is the only first-order method and shows the slowest convergence among all the considered methods. Following, the second-order methods, Crank-Nicolson, second-order backward differentiation method (BDF2) and second-order singly-diagonally RK method (SDIRK2), display a second-order accuracy experimentally. A better accuracy is achieved using the singly-diagonally RK method of order three (SDIRK3) as well as the third-order singly-diagonally RK method with explicit first step (ESDIRK3/2). The highest order is the four and is attained when the fourth-order singly-diagonally RK method with explicit first step (ESDIRK46) is used.

Level of temporal error for a given computational effort

The second experiment involves the calculation of the temporal error and the comparison between the various schemes. A simple comparison using the same timestep size for all methods would be unfair, since some of them are multi-stage and, thus, advancing one timestep takes significantly more time than with single-step methods. In this regard, the number of evaluations of the Jacobian matrix in each timestep should be taken into account. For example, the SDIRK2 method requires the evaluation and solution of the non-linear problem twice in every timestep, while the BDF2 method requires the same only once. Therefore and in order for the comparison to be fair, the chosen timestep size with SDIRK2 should be twice as large as the one of BDF2.

In Table 4.10 the calculated error level of the primal and mixed variables for the simulation of the flow described by Equations (4.47) and (4.48) at the final time $t^{end} = 3$ is shown. In the third column, the equivalent chosen timestep size for each method is shown. The timestep size varies from 0.05 for single-step methods and BDF2 to 0.25 for the ESDIRK46 of six stages (five actual stages, the first stage is explicit).

Despite the large timestep, the ESDIRK methods appear to be the most accurate, in particular ESDIRK46 shows the lowest error, followed by ESDIRK3/2. Crank-Nicolson also seems competitive, however, as mentioned in Section 4.4.1, it can deliver

(a) $\boldsymbol{u}$

(b) p

(c) $\boldsymbol{L}$
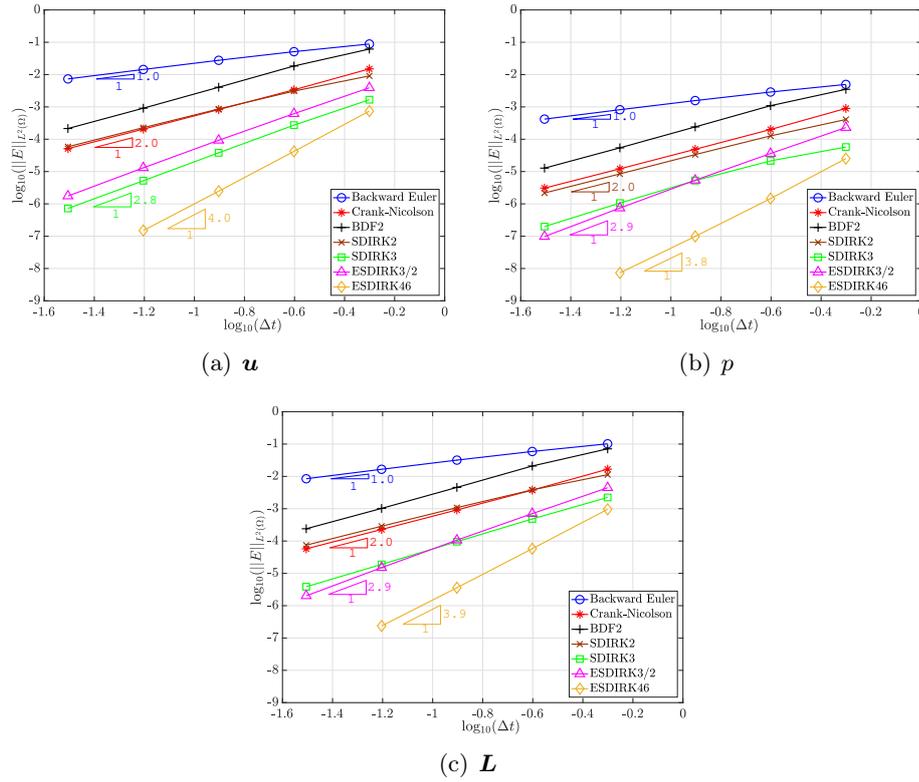
Figure 4.20: *Transient analytical flow: study of temporal convergence for various time-marching methods, initial time $t^0 = 0$ and final time $t^{end} = 1$, fourth-order quadrilateral mesh is used and $\hat{\tau} = 0.3$.*

an oscillatory solution when a large timestep is used, thus it is usually not preferred in practice.

Two-dimensional transient flow around circle

The two-dimensional transient flow around a circle is a well-known benchmark test that is widely used for testing a transient incompressible Navier-Stokes code. It is known that for Reynolds number higher than approximately 40, the flow stops being steady and symmetric and time-dependent phenomena appear. The flow is separated and swirling vortices are formed downstream of the circle, which is known as Von Kármán street vortex, named after the physicist Theodor von Kármán.

In this section, the test case "2D-2" presented in [167] will be studied. The geometry of the problem is the same as the one used for the steady flow around circle, refer to Figure 4.2. The inflow condition is also the same, refer to Equation (4.19). The difference lies in the mean velocity at inlet $\bar{U}$, which in the transient case is $\bar{U} = 1.0 \, \text{m/s}$, yielding a Reynolds number 100. It is reminded that the maximum velocity at inlet, at the center of the duct inlet, is related to the mean velocity according to the formula $U_m = 3\bar{U}/2$.

Method	Number of Jacobian evaluations	Δt	$\frac{\|\mathbf{u}-\mathbf{u}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{u}\|_{\mathcal{L}_2(\Omega)}}$	$\frac{\|\mathbf{p}-\mathbf{p}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{p}\|_{\mathcal{L}_2(\Omega)}}$	$\frac{\|\mathbf{L}-\mathbf{L}_h\|_{\mathcal{L}_2(\Omega)}}{\|\mathbf{L}\|_{\mathcal{L}_2(\Omega)}}$
Backward Euler	1	0.05	5.16e-1	2.93e-2	5.98e-1
Crank-Nicolson	1	0.05	7.87e-3	4.63e-4	8.80e-3
BDF2	1	0.05	3.40e-2	2.00e-3	3.81e-2
SDIRK2	2	0.1	2.41e-2	8.90e-4	3.07e-2
SDIRK3	3	0.15	1.20e-2	5.71e-4	1.77e-2
ESDIRK3/2	3	0.15	8.40e-3	4.80e-4	9.68e-3
ESDIRK46	5	0.25	2.34e-3	8.67e-5	3.29e-3

Table 4.10: *Transient analytical flow: level of temporal error of various time-marching methods for a given computational effort, initial time $t^0 = 0$ and final time $t^{end} = 3$, fourth-order quadrilateral mesh is used and $\hat{\tau} = 0.3$.*

(a) Velocity magnitude contour   (b) Pressure contour
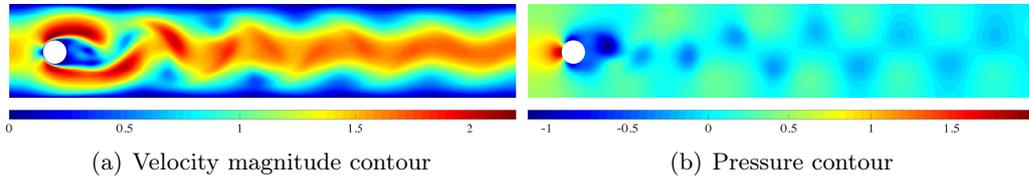
Figure 4.21: *Two-dimensional transient laminar flow around circle: velocity magnitude and pressure contours at $t = 6\,\mathrm{s}$ when vortex shedding is fully developed, mesh consists of 2,435 triangular elements of degree $k = 2$, employed time-marching scheme ESDIRK3/2, $Re = 100$, $\hat{\tau} = 1$, velocity is expressed in $\mathrm{m/s}$ and pressure in Pa.*

The unsteady nature of the problem leads to an oscillating lift, drag coefficient on the circle as well as oscillating pressure drop between the leading and the trailing edge of the circle. This oscillation is of periodic nature and the reported results include quantities of interest, such as the maximum lift coefficient, maximum drag coefficient, pressure drop when lift is minimum and Strouhal number. This last number is a dimensionless number that describes the vortex shedding and is given as $St = f\,D/\bar{U}$, where D is the circle diameter and f the vortex shedding frequency.

Similar to the steady flow around circle, there is no exact solution for the quantities of interest, but a set of solutions given by different research groups. These are summarized in [167] and the bounds of them are used for comparison with the present study.

The simulation is initiated using the solution of the steady case for $Re = 20$ as initial condition. Then, the flow past the circle separates and vortices are formed. These vortices have a periodic pattern and eventually exit the domain, as seen in Figure 4.21. The lift and drag coefficients on the circle surface oscillate until they reach their maximum amplitude, as seen in Figure 4.22. Finally, the simulation is stopped after 8.1 seconds, after the vortex shedding is fully developed.

For the calculation of the flow quantities, in [167] it is stated that the flow has to be initiated using as initial condition any solution where C_l is maximum. Then, the

(a) Lift coefficient

(b) Drag coefficient
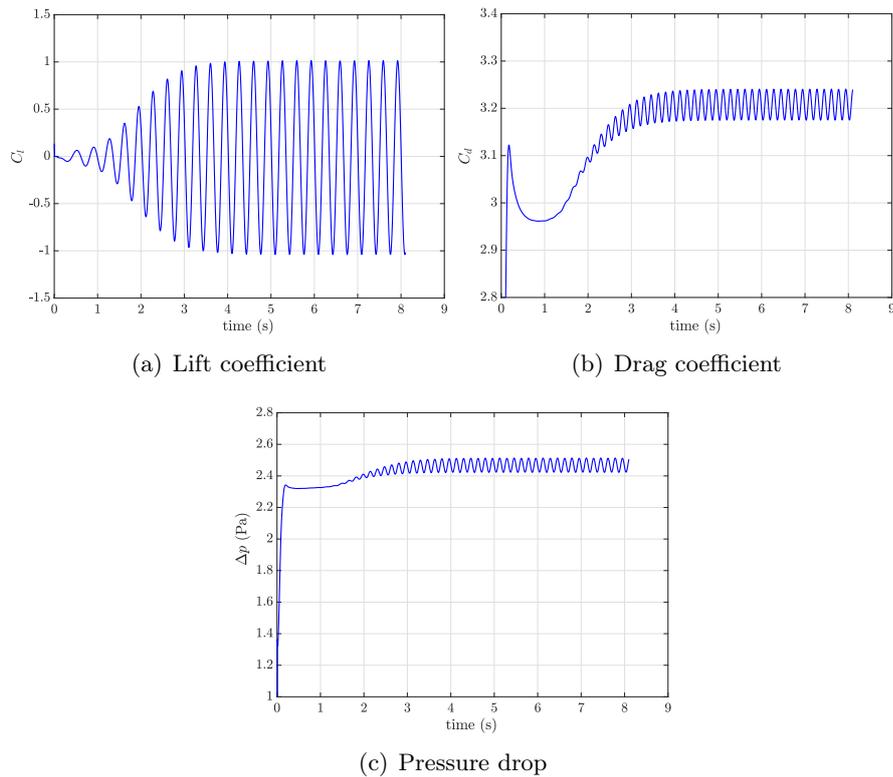
(c) Pressure drop

Figure 4.22: *Two-dimensional transient laminar flow around circle: evolution of lift, drag and pressure drop versus time, time-marching scheme is ESDIRK3/2, degree $k = 3$, timestep size $\Delta t = 0.005$ s, $Re = 100$ and $\hat{\tau} = 1$.*

simulation is run for one period and the maximum lift and drag coefficients are obtained, as well as the pressure drop at half of the period (when C_l is minimum). In Table 4.11 the resulting quantities of interest are summarized for degrees of approximation $k = 1, 2, 3, 6$. The employed time integration method is ESDIRK46 and the timestep size is chosen to be $\Delta t = 0.01$ s for all degrees k. Moreover, for $k = 3$ two more timestep sizes are considered, a coarser one and a finer one, with the aim of demonstrating the importance of the chosen Δt. As already mentioned, the initial condition must be a solution when C_l is maximum, therefore, the first solution after 6 s whose C_l is maximum is used as initial condition. The limit of 6 s is chosen bearing in mind that the flow at this time is fully developed.

It is clear that both the polynomial degree k and the timestep size Δt have a positive effect on the calculation of the flow quantities. The influence of k can be also seen in Figure 4.23 where the different evolutions of lift and drag coefficients versus time are plotted for $k = 1, 2, 3$. The impact of Δt on lift and drag is also clear in Figure 4.24. Not only the values of the coefficients change, but also the frequency of the vortex shedding, which is better approximated with a smaller timestep.

k	n_{dof}	Δt(s)	$C_{l,max}$	$C_{d,max}$	St	ΔP(Pa)
1	17,459	0.01	0.9247	3.1925	0.2941	2.4782
2	24,971	0.01	0.9934	3.2350	0.2941	2.4871
3	32,483	0.02	0.9634	3.2230	0.2941	2.4688
	32,483	0.01	0.9986	3.2245	0.3030	2.4796
	32,483	0.005	1.0060	3.2250	0.2985	2.4821
6	55,019	0.01	1.0080	3.2260	0.3030	2.4820
	lower bound [167]		0.9900	3.2200	0.2950	2.4600
	upper bound [167]		1.0100	3.2400	0.3050	2.5000

Table 4.11: *Two-dimensional transient laminar flow around circle: quantities of interest for various degrees of approximation k, degrees of freedom and timestep size, employed time-marching scheme is ESDIRK46, mesh consists of 2,435 triangular elements, $Re = 100$, $\hat{\tau} = 1$, simulation is initiated with initial condition being the first instant where C_l is maximum for $t > 6$ s.*

(a) Lift coefficient

(b) Drag coefficient
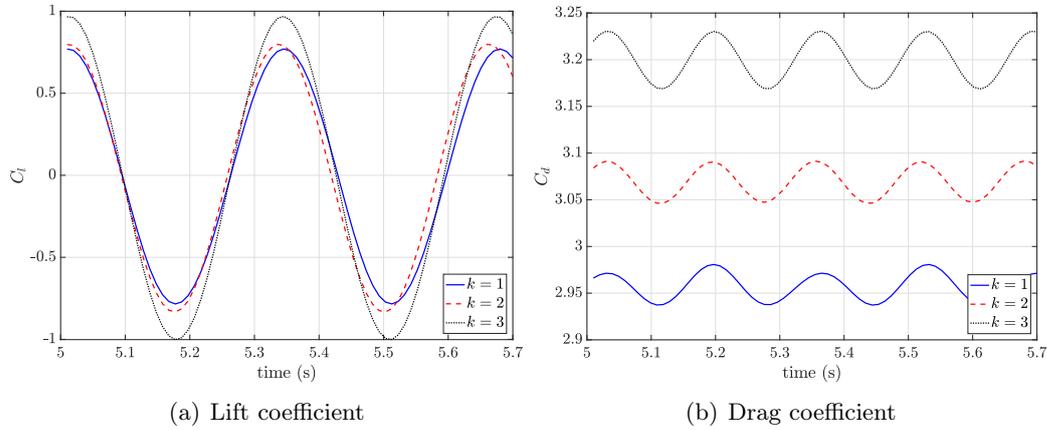
Figure 4.23: *Two-dimensional transient laminar flow around circle: comparison of solution using various degrees of approximation $k = 1, 2, 3$, employed time-marching scheme is BDF2, $\Delta t = 0.01$ s, mesh consists of 856 triangular elements, $Re = 100$ and $\hat{\tau} = 1$.*

4.5   Adaptive transient simulations

In this section adaptive techniques will be presented for transient flows. Two different techniques will be discussed. The first one is related to the degree of approximation of the elements and the second one to the timestep size.

Regarding the degree of approximation, so far in transient simulations a uniform degree has been used throughout the whole mesh. Nevertheless, since time-dependent phenomena occur, for instance the creation of vortices and their movement through the domain, it is reasonable that higher accuracy is desired in the vicinity of these phenomena in order to capture them precisely. In this regard, a high degree of approximation on a global level might be redundant and only necessary in certain zones of the mesh.

Similarly, in transient simulations presented so far, the timestep size has been constant. It is possible to change the size of the timestep during a transient simulation in order to adjust to a certain level of temporal accuracy. This can be achieved by using

(a) Lift coefficient

(b) Drag coefficient
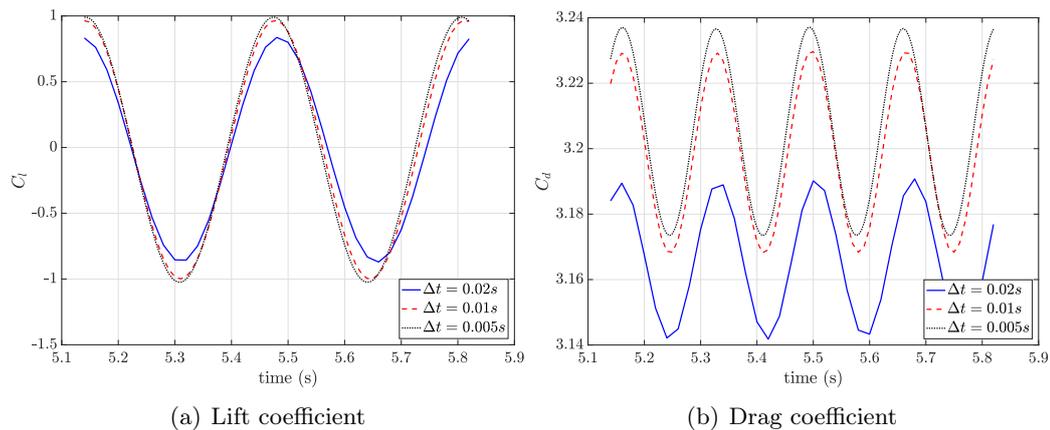
Figure 4.24: *Two-dimensional transient laminar flow around circle: comparison of solution using various timestep sizes Δt, employed time-marching scheme is BDF2, mesh consists of 856 triangular elements of $k = 3$, $Re = 100$ and $\hat{\tau} = 1$.*

appropriate error estimators that can estimate the level of temporal error. Afterwards, the timestep size may be increased or decreased automatically so that the solution meets the temporal accuracy requirements.

4.5.1 Degree adaptivity in transient simulations

Based on the degree adaptivity procedure described in Section 4.3 for steady incompressible laminar flows, a similar procedure for transient flows can be constructed. This procedure is described in Algorithm 3. Since it's a transient simulation, the degree distribution should be adapted with a certain frequency, for example every N_{adapt} timesteps. The expected frequency of the transient phenomenon should be taken into consideration when choosing the frequency in which p-adaptivity is done. For instance, in the case of a wave, given its oscillating nature, the p-adaptivity should be performed 10 times or more per cycle. In this way, the movement of the wave can be captured and the elements' degree can be adapted accordingly.

In Figure 4.25 the estimated local error of the unsteady flow around a circle in 2D at two instants is shown. These results are taken for a simulation with uniform degree $k = 2$ throughout the whole mesh and a relatively coarse mesh of 856 triangles. The estimator appears to be coherent with the movement of the vortices and the interaction with the boundary layer.

Next, the simulation of the transient flow in which the degree of the elements is adapted every N_{adapt} timesteps is presented. The mesh has 856 triangular elements, which is a rather coarse mesh and is, in fact, chosen on purpose, so that the effectiveness of p-adaptivity is demonstrated. Initially, the mesh has a uniform degree $k = 2$, ensuring this way the correct approximation of the curved boundaries. For time marching the third-order ESDIRK3/2 method is used, with a timestep size $\Delta t = 0.01$ s. Given the fact that the period of shedding is approximately $T = 0.33$ s and the chosen timestep

---

Algorithm 3 Degree-adaptivity process for the transient incompressible Navier-Stokes equations using the proposed HDG method

---

1: Load mesh
2: Read initial degree map k_e
3: Read desired accuracy ϵ
4: Read curved elements
5: Read the frequency of degree adaptivity, in number of timesteps N_{adapt}
6: $i = 1$
7: **while** Final time is not reached **do**
8:      Solve the problem and calculate $\mathbf{L}, \mathbf{u}, \mathbf{p}, \rho, \hat{\mathbf{u}}$
9:      **if** $i == N_{adapt}$ **then**
10:         Calculate the post-processed velocity field $\mathbf{u}^{\star}$
11:         Estimate the local error E_e using Equation (4.24)
12:         Calculate the new degree map k_e^{new} using Equation (4.26)
13:         **if** curved elements **then**
14:            minimum degree in curved elements $k = 2$
15:         **end if**
16:         Build new mesh based on k_e^{new}
17:         Project solution $\mathbf{L}, \mathbf{u}, \mathbf{p}, \rho, \hat{\mathbf{u}}$ on the new mesh
18:         **if** $max(E_e) \gg \epsilon \bar{U}$ **then**
19:            Repeat last timestep
20:         **end if**
21:         $i = 1$
22:      **else**
23:         $i = i + 1$
24:      **end if**
25: **end while**

(a) Velocity magnitude in m/s, $t = 6\,\text{s}$ (b) Estimated spatial error in m/s, $t = 6\,\text{s}$

(c) Velocity magnitude in m/s, $t = 6.2\,\text{s}$ (d) Estimated spatial error in m/s, $t = 6.2\,\text{s}$
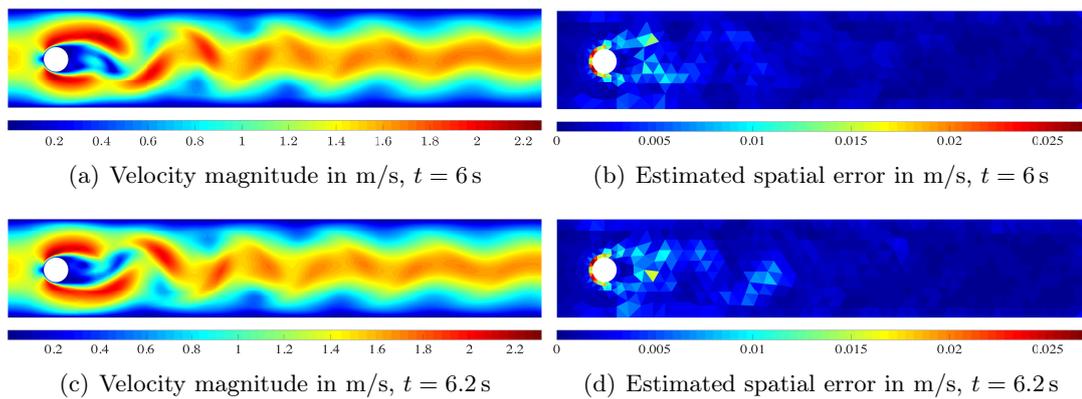
Figure 4.25: *Two-dimensional transient laminar flow around circle: velocity magnitude and estimated local error at two instants when uniform degree $k = 2$ is used, employed time-marching scheme is ESDIRK3/2, $\Delta t = 0.01\,\text{s}$, mesh consists of 856 triangular elements, $Re = 100$ and $\hat{\tau} = 1$.*

(a) Velocity magnitude, $t = 3.5\,\text{s}$ (b) Degree map, $\epsilon = 5 \times 10^{-3}$, $t = 3.5\,\text{s}$

(c) Velocity magnitude, $t = 6.3\,\text{s}$ (d) Degree map, $\epsilon = 5 \times 10^{-4}$, $t = 6.3\,\text{s}$
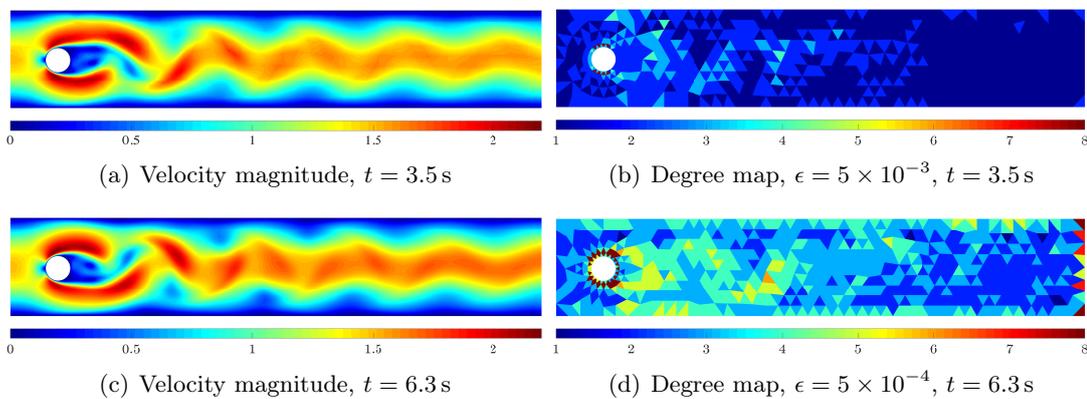
Figure 4.26: *Adaptive-degree two-dimensional transient laminar flow around circle: velocity magnitude and degree map for two levels of accuracy ϵ, employed time-marching scheme is ESDIRK3/2, $\Delta t = 0.01\,\text{s}$, mesh consists of 856 triangular elements, $Re = 100$ and $\hat{\tau} = 1$.*

size $\Delta t = 0.01\,\text{s}$, then if the the degree is adapted every $N_{adapt} = 2$ timesteps, this is roughly equivalent to adapting the mesh 16 times per cycle.

Two values of desired accuracy are studied, namely $\epsilon = 5 \times 10^{-3}$ and $\epsilon = 5 \times 10^{-4}$. In Figure 4.26 the degree maps that arise after using each value of accuracy for two instants are shown. In both cases it is observed that the degree of the elements is increased downstream of the circle and especially in the elements where the separation of the flow takes place as well as the in boundary layer. Moreover, it is reasonable that a stricter value of accuracy leads to higher element degrees.

The maximum lift coefficient, maxmimum drag coefficient and pressure drop when lift is minimum are reported in Figure 4.27. The adaptive-degree results with the two levels of accuracy and the uniform-degree results are compared with one another. The superiority of the adapted meshes is clear, especially the adapted mesh of accuracy $\epsilon = 5 \times 10^{-3}$ appears to capture the quantities of interest more accurately than the uniform-degree meshes of the same number of degrees of freedom.

(a) Lift coefficient   (b) Drag coefficient
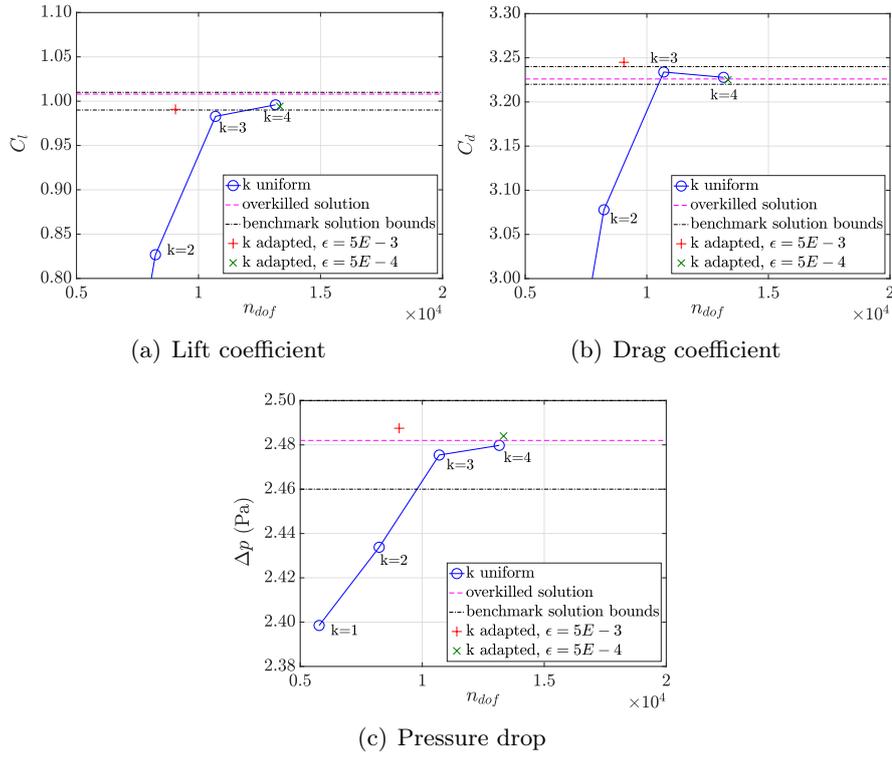
(c) Pressure drop

Figure 4.27: *Adaptive-degree two-dimensional transient laminar flow around circle: calculation of maximum lift, maximum drag and pressure drop when lift is minimum for adaptive-degree and uniform-degree simulations, time-marching scheme is ESDIRK3/2, timestep size* $\Delta t = 0.005\,\mathrm{s}$, $Re = 100$ *and* $\hat{\tau} = 1$.

4.5.2   Timestep size control

While using single-step temporal schemes, a temporal error estimator can be obtained in each time step through the following steps (a scalar variable y is considered for simplicity):

1. advance from t^n to t^{n+1} with timestep Δt_1 and compute the numerical approximation of y, $\tilde{y}^{n+1}_{\Delta t_1}$;

2. repeat, using a smaller timestep, e.g. $\Delta t_2 = \Delta t_1/2$, and compute $\tilde{y}^{n+1}_{\Delta t_2}$;

3. evaluate the difference between the two solutions $\left| \tilde{y}^{n+1}_{\Delta t_1} - \tilde{y}^{n+1}_{\Delta t_2} \right|$;

4. calculate the optimal timestep size Δt_* that would satisfy the tolerance criterion;

5. if $\Delta t_* < \Delta t_1$, then repeat the timestep using Δt_*.

Nevertheless, the aforementioned procedure increases dramatically the cost of single-step schemes because whenever the timestep size is to be adapted, additional calculations have to be performed, therefore single-step schemes are inefficient for problems

with adaptive timestep size. Furthermore, as far as the backward differentiation formula of order two (BDF2) is concerned, it becomes complicated to use a non-constant step size, since the solution of the two previous time levels is used.

The singly-diagonally implicit Runge-Kutta methods usually include an embedded stage of lower (or sometimes higher) order, which can serve as an error estimator. The singly-diagonally implicit Runge-Kutta methods with an explicit first stage (ESDIRK) that were presented in Section 4.4.1 include an embedded stage that will be used for this purpose. In order to explain the timestep adaptivity using a method of order k with embedded stage of order $k-1$, first, the local truncation error of a scalar variable y is written:

$$
\begin{aligned}
y^{n+1} - \tilde{y}_k^{n+1} &= C_k\,\Delta t^{k+1} + \mathcal{O}(\Delta t^{k+2}) \\
y^{n+1} - \tilde{y}_{k-1}^{n+1} &= C_{k-1}\,\Delta t^k + \mathcal{O}(\Delta t^{k+1}),
\end{aligned}
\tag{4.49}
$$

where $\tilde{y}_k^{n+1}$ is the solution obtained at the stage of kth-order and $\tilde{y}_{k-1}^{n+1}$ at the stage of $(k-1)$th-order, both at time $n+1$. Next, the first expression of the truncation error is subtracted from the second and is divided by the computed solution $\tilde{y}_k^{n+1}$, yielding the normalized estimated local error:

$$
E_y = \frac{\tilde{y}_k^{n+1} - \tilde{y}_{k-1}^{n+1}}{\tilde{y}_k^{n+1}} \approx C\,\Delta t^k
\tag{4.50}
$$

The optimal timestep size Δt_* is the one that, using the method of order k, produces a local error equal to the prescribed normalized tolerance TOL

$$
TOL \approx \frac{C_{k-1}\,\Delta t_*^k}{\tilde{y}_k^{n+1}}
\tag{4.51}
$$

Combining Equations (4.50) and (4.51) it is possible to express the optimal timestep size as

$$
\Delta t_* =
\begin{cases}
S\Delta t \sqrt[k]{\left|\frac{TOL}{E_y}\right|}, & TOL \geq E_y \\[2mm]
S\Delta t \sqrt[k-1]{\left|\frac{TOL}{E_y}\right|}, & TOL < E_y
\end{cases}
\tag{4.52}
$$

where S, according to [178], is a parameter with a value lower than 1, usually $S = [0.90, 0.98]$. In [179] more sophisticated timestep size controllers are used for ESDIRK methods. The aforementioned procedure holds also for a vector variable, the only difference being that in the local error estimator of Equation (4.50), both nominator and denominator have to be expressed in a norm, usually the $\mathcal{L}_2$-norm is preferred. Moreover, since in our study there are more than one variables, it is

reasonable to evaluate the local error estimator for each variable and, afterwards, adapt the timestep based on the largest error.

Following, the timestep adaptivity procedure will be applied to a transient flow with analytical expression of the velocity and pressure field as well as the transient flow around a circle.

Adaptive timestep size in transient analytical flow

Timestep control will be performed in the analytical flow whose velocity and pressure fields are expressed in Equations (4.47) and (4.48). The simulation will be carried out from $t^0 = 0\,\mathrm{s}$ until $t^{end} = 1.5\,\mathrm{s}$ with initial timestep $\Delta t = 0.2\,\mathrm{s}$ and using the ESDIRK3/2 method. In each timestep, the temporal error is estimated based on the solution at $n + 1$ and the solution of the embedded stage. Next, if this error is larger than the prescribed tolerance, the timestep is adapted according to Equation (4.52) and the timestep is repeated. It has to be highlighted that the computational mesh should be fine, both in mesh size and degree, minimizing in this way the spatial error. In this study, the mesh is uniform and consists of fourth-order quadrilaterals with size $h = 1/16$.

The estimator appears to predict sufficiently the $\mathcal{L}_2$-norm of the temporal error, as seen in Figure 4.28 for velocity, pressure and mixed variable. Even though, the error is overestimated, the actual error manages to stay below the specified normalized tolerance. It can be also observed that for this simulation the pressure and the mixed variable show the largest errors and the timestep size is adapted based on the error levels of these variables. In Figure 4.28 (d) the evolution of Δt is plotted, where it can be seen that the initial $\Delta t = 0.2$ was too large and after 2 iterations it was finally reduced to approximately $\Delta t = 0.12$. Later, it is increases gradually and in the last timestep it is decreased once again.

Adaptive timestep size in transient flow around circle

The transient flow around a circle with Reynolds number $Re = 100$ will be studied using an adaptive timestep. The same case was previously presented in Section 4.4.2, but with a fixed timestep size. In order to minimize the error due to spatial discretization, a relatively fine mesh with 2,345 second-order triangular elements is employed. At this point, the ESDIRK3/2 method, thanks to its embedded stage, will be used for the estimation of the temporal error and the adaptation of the timestep size during the simulation. Three different value of tolerance TOL will be considered and the effect on the resulting timestep evolution as well as lift will be studied.

The different levels of tolerance yield a different evolution of timestep size, especially when a loose tolerance, such as 5×10^{-2}, is used. In this case, according to Figure 4.29, Δt is so large that the shedding phenomenon cannot be captured. This is no longer the case when a stricter tolerance is used, for example 5×10^{-3} or 5×10^{-4}. In both
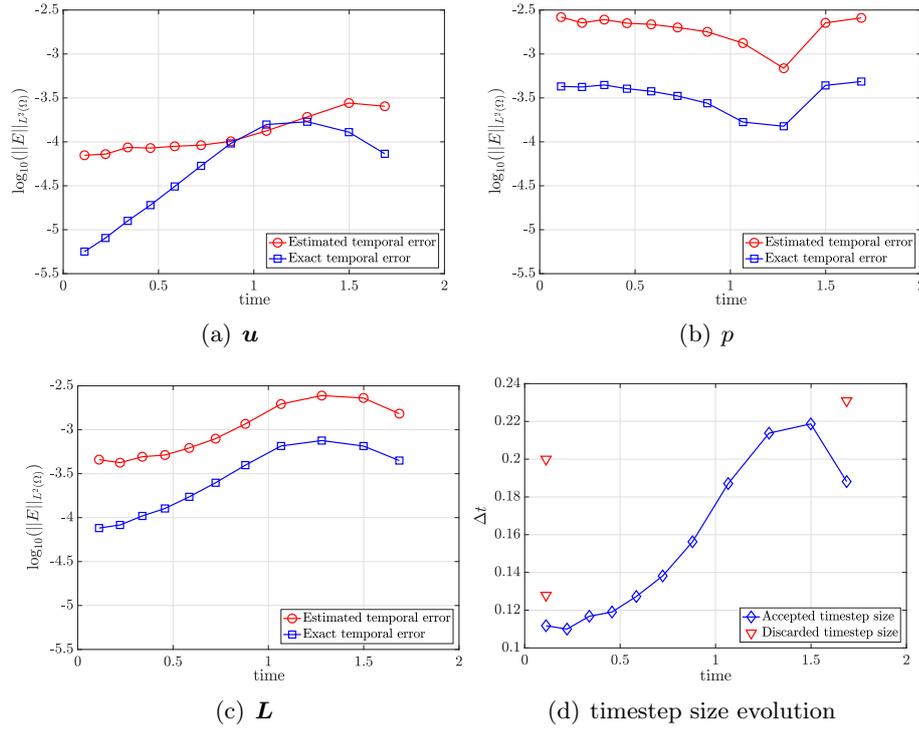
Figure 4.28: *Transient analytical flow: estimated and exact temporal error for transient analytical flow, desired normalized tolerance $TOL = 5E - 2$, time-marching scheme is ESDIRK3/2, initial time $t^0 = 0$ and final time $t^{end} = 1.5$, fourth-order quadrilateral mesh is used, $\nu = 0.33$ and $\hat{\tau} = 0.5$.*

cases, the timestep size initially increases until the shedding starts, then decreases and eventually converges around $0.015\,\text{s}$ for tolerance 5×10^{-3} and $0.006\,\text{s}$ for tolerance 5×10^{-4}. Last, the shedding pattern and lift are captured correctly. Recall that the overkilled Strouhal number was 0.3030, which leads to a period $T \approx 0.33\,\text{s}$, while the overkilled maximum lift was 1.008.

4.6 Conclusion

In this chapter a hybridizable discontinuous Galerkin method using the Voigt notation for the Cauchy formulation of the steady and transient incompressible Navier-Stokes is presented. The symmetry of the stress tensors is strongly enforced and the point-wise conservation of angular momentum is satisfied. The tractions that have physical meaning are imposed directly on Neumann boundaries.

Initially, steady-state flows were studied. The optimal convergence properties of the HDG were demonstrated through numerical tests. In particular, the convergence order $k + 1$ of all primal and mixed variables as well as the order $k + 2$ of the post-processed variable was shown. The proposed HDG method was found to accurately estimate flow quantities, such as drag and lift caused by the two-dimensional flow around a circle. Moreover, the method performed well in a realistic case of a microvalve with

(a) Δt, $TOL = 5 \times 10^{-2}$

(b) C_l, $TOL = 5 \times 10^{-2}$

(c) Δt, $TOL = 5 \times 10^{-3}$

(d) C_l, $TOL = 5 \times 10^{-3}$

(e) Δt, $TOL = 5 \times 10^{-4}$

(f) C_l, $TOL = 5 \times 10^{-4}$
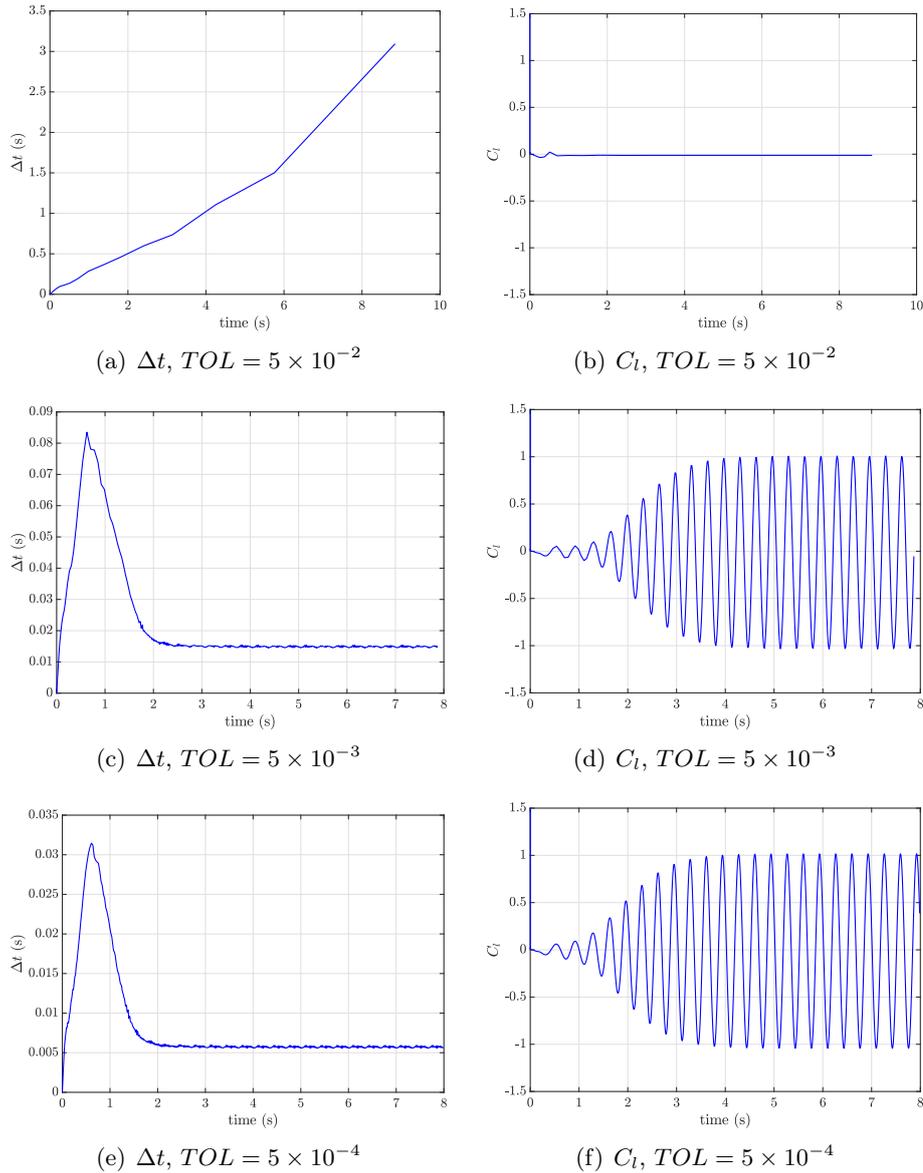
Figure 4.29: *Adaptive-timestep two-dimensional transient laminar flow around circle: lift coefficient and timestep size evolution for different tolerance levels, time-marching scheme is ESDIRK3/2, second-order triangular mesh with 2,345 elements is used, $Re = 100$ and $\hat{\tau} = 0.5$.*

a more complex geometry and for Reynolds number in the steady and incompressible flow limits. The superconvergent property of the proposed method gives the ability to construct spatial estimators and perform degree-adaptivity locally. Degree-adapted meshes were shown to perform better than uniform-degree meshes, in terms of spatial error versus degrees of freedom.

Transient flows were studied by adding high-order time-marching schemes to HDG. High-order temporal schemes are necessary if the high-order spatial accuracy offered by HDG is to be preserved. Several transient flows were studied, through which a comparison between the various temporal schemes was made and their suitability for

a high-order spatial method, such as HDG, was demonstrated. Furthermore, the possibility of adaptive simulations was explored. On one hand, meshes whose degree may be adapted locally in the course of the transient flow were shown to accurately approximate important flow features, such as boundary layers and vortices. On the other hand, certain time integration methods give the possibility to construct temporal error estimators and, consequently, enable the timestep size control, which takes place during the simulation. Tests showed the ability of the method to adapt to the transient phenomena.

Chapter 5

Preconditioning for incompressible flows using the HDG method [1]

An integral part of every simulation is the solution of a linear system that arises from the discretization of PDEs. The various numerical methods differ in the way this system is formed and the final structure of the left and right-hand side matrices. Nevertheless, the solution of a system is a common step among all solvers. When small problems of some thousands degrees of freedom (DOF) are studied, given today's computational resources, the solution of the final system can be usually done using direct solvers. Such solvers might be based on the LU decomposition, QR factorization and Cholesky factorization. Some examples of well-known direct solvers include the MUltifrontal Massively Parallel sparse direct Solver (MUMPS) [180, 181], the PARDISO package [182] and UMFPACK package [183].

In practical settings, a common obstacle to overcome is the size of the linear system. This becomes more pronounced in three-dimensional simulations, as the number of DOF increases dramatically with the number of elements. Direct solvers require large amounts of memory and, on top of that, the required memory scales poorly with DOFs, especially in three dimensions. Under these circumstances, it is almost impossible to proceed to the solution of the linear system using a direct solver and, therefore, iterative solvers need to be employed. Iterative solvers use an initial guess and, making approximations based on previous iterations, they gradually converge to the solution. The initial guess and the desired accuracy of the solution, in the form of stopping criterion of the iterative solver, affect the number of iterations in which the solver convergences.

In spite of that, what largely affects the number of iterations and speed of iterative solvers is the distribution of eigenvalues of the final linear system. In particular, systems that have their eigenvalues clustered near 0 converge more slowly. It is very common to transform the linear system that is to be solved with the aim of solving a system with a better distributed eigenvalues. This process is called preconditioning and the

---

[1]This chapter is a modified version of the in-preparation article: A. Karkoulias, A. Viguerie, F. Auricchio, M. Giacomini, A. Huerta, Effective preconditioning for the solution of incompressible flows using the HDG method

preconditioned system is more suitable than the original system, when iterative solvers are employed.

This chapter focuses on the preconditioning of the final system of equations that arises from the HDG method. In Section 5.1, the preconditioned strategy for incompressible flows using HDG is presented. The proposed preconditioning technique used with an iterative solver is compared with the direct solver for the solution of Stokes flow and Navier-Stokes equations in Section 5.2. In Section 5.3, the performance of the preconditioner over a range of Reynolds numbers is studied. In Section 5.4, three-dimensional cases that can be solved thanks to the preconditioned system and the use of iterative solvers are presented. A comparison of the iterative HDG solver with a velocity-pressure formulation of the Navier-Stokes employed in an open-source solver using the same iterative solver and preconditioner is performed in Section 5.5. Finally, Section 5.6 summarises the main points of this chapter.

5.1   Preconditioning strategy

5.1.1   General considerations

The discretization of the weak form of the incompressible Navier-Stokes equations using the HDG method gives the following saddle point problem

$$
\begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}} & \widehat{\mathbf{K}}_{\hat{u}\rho} \\ \widehat{\mathbf{K}}_{\hat{u}\rho}^T & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \widehat{\boldsymbol{u}} \\ \rho \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{f}_{\hat{u}} \\ \boldsymbol{f}_{\rho} \end{Bmatrix}.
\tag{5.1}
$$

Performing left preconditioning, we introduce the matrix $\mathbf{P}$ and solve the following system

$$
\mathbf{P}^{-1} \begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}} & \widehat{\mathbf{K}}_{\hat{u}\rho} \\ \widehat{\mathbf{K}}_{\hat{u}\rho}^T & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \widehat{\boldsymbol{u}} \\ \rho \end{Bmatrix} = \mathbf{P}^{-1} \begin{Bmatrix} \boldsymbol{f}_{\hat{u}} \\ \boldsymbol{f}_{\rho} \end{Bmatrix}.
\tag{5.2}
$$

We desire $\mathbf{P}^{-1}$ to be an approximation of $\begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}} & \widehat{\mathbf{K}}_{\hat{u}\rho} \\ \widehat{\mathbf{K}}_{\hat{u}\rho}^T & \mathbf{0} \end{bmatrix}^{-1}$. A common class of preconditioners for the solution of the saddle-point problem given by the incompressible Navier–Stokes equations has been previously presented in [184, 185, 186] and has the form

$$
\mathbf{P}^{-1} = \begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}} & \widehat{\mathbf{K}}_{\hat{u}\rho} \\ \mathbf{0} & -\widetilde{\mathbf{S}} \end{bmatrix}^{-1} = \begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}}^{-1} & \widehat{\mathbf{K}}_{\hat{u}\hat{u}}^{-1} \widehat{\mathbf{K}}_{\hat{u}\rho} \widetilde{\mathbf{S}}^{-1} \\ \mathbf{0} & -\widetilde{\mathbf{S}}^{-1} \end{bmatrix},
\tag{5.3}
$$

where $\widetilde{\mathbf{S}}$ is the approximation of the Schur complement $\mathbf{S} = \widehat{\mathbf{K}}_{\hat{u}\rho}^T \, \widehat{\mathbf{K}}_{\hat{u}\hat{u}}^{-1} \, \widehat{\mathbf{K}}_{\hat{u}\rho}$. The inverse matrix of $\widehat{\mathbf{K}}_{\hat{u}\hat{u}}$ makes the computation of the Schur complement difficult and costly. Therefore, an approximation of the Schur complement is commonly used.

Within the finite element framework, a good approximation of the Schur complement for the Stokes problem is a mass matrix on the pressure space, as already proposed in [187, 185]. In HDG a suitable mass matrix in the pressure space of the global problem has to be found. We recall that the pressure variable of the final system in HDG is ρ, which is defined element-wise, while the corresponding pressure space is $\mathbb{R}^{\mathtt{n_{el}}}$, so the approximation of the Schur complement has to be a square matrix of dimension $\mathtt{n_{el}}$.

A diagonal matrix may be chosen as the approximation of the Schur complement, based on the fact that the pressure variable ρ of the global problem has one single value for each element. Moreover, the elements' characteristic length may appear in the diagonal of the matrix, offering, in this way, a scaling to the mass matrix in the pressure space. So far, the proposed Schur complement approximation for the Navier–Stokes problem has the following form

$$\widetilde{\mathbf{S}} = C \operatorname{diag}(\boldsymbol{h}_e), \tag{5.4}$$

where h_e denotes the local element size and C is a scaling constant. This constant has to be selected in such a way that it approximates the scaling of the Schur complement. In various studies [187, 188] it was shown that the Schur complement approximation could be scaled with the viscosity ν. In HDG the parameter τ plays an important role in the stabilization of the method while it multiplies many terms in the weak form of the equations, thus it may be also included in the scaling factor of the Schur complement. The following scaling factor is proposed, where the viscosity ν is divided by the reference length $\bar{L}$ of the problem in order to match the units of τ. Recall that τ has a part due to viscosity and other due to velocity, as shown in Equation (4.9). In this sense, a change in Reynolds number will also cause a change in the scaling factor.

$$\widetilde{\mathbf{S}} = \bar{L}\left(\frac{\nu}{\bar{L}} + \tau\right)^{-1} \operatorname{diag}(\boldsymbol{h}_e). \tag{5.5}$$

The inverse and the multiplication by the characteristic length $\bar{L}$ is used so that the scaling factor is consistent with the approximation of the Schur complement agrees in units with the definition of Schur complement, refer also to Appendix A.3.3. The ν term is divided by the problem's characteristic length in order to be consistent in units with τ. It is important to underline that the aforementioned approximation of the Schur complement is only a suggestion and that countless approximations might arise likewise. Nonetheless, this approximation was found to give satisfying results, some of which are reported in the following sections of this chapter.

5.1.2 Implementation details

In order for the solver to be effective, the preconditioner must be implemented efficiently. At each GMRES iteration the following preconditioned system is solved:

$$\mathbf{P}\,\boldsymbol{x} = \boldsymbol{r} \Leftrightarrow \begin{bmatrix} \widehat{\mathbf{K}}_{\hat{u}\hat{u}} & \widehat{\mathbf{K}}_{\hat{u}\rho} \\ \mathbf{0} & -\mathbf{S} \end{bmatrix} \begin{Bmatrix} \widetilde{\boldsymbol{x}} \\ \widetilde{\boldsymbol{y}} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \end{Bmatrix}, \tag{5.6}$$

where $\boldsymbol{x}$ is the vector that contains the two variables and $\boldsymbol{r}$ is the right-hand side vector. Initially, at the first GMRES iteration, the vector $\boldsymbol{r}$ is equal to $\boldsymbol{r} = \boldsymbol{f} - \mathbf{K}\boldsymbol{x}_0$, where $\boldsymbol{f}$ and $\mathbf{K}$ are the right-hand side vector and left-hand side matrix of Equation (5.1), whereas $\boldsymbol{x}_0$ is the initial guess provided to the GMRES method. The solution from the previous nonlinear iteration is taken as $\boldsymbol{x}_0$, hence speeding up the convergence of GMRES. Afterwards, the right-hand side vector $\boldsymbol{r}$ is updated inside GMRES. The advantage of a block-triangular preconditioner is that the above system can be solved fast in the following way

$$\begin{cases} \widetilde{\boldsymbol{y}} = -\overset{\sim}{\mathbf{S}}^{-1} \boldsymbol{r}_2, \\ \widetilde{\boldsymbol{x}} = \widehat{\mathbf{K}}_{\hat{u}\hat{u}}^{-1}\big(\boldsymbol{r}_1 - \widehat{\mathbf{K}}_{\hat{u}\rho}\widetilde{\boldsymbol{y}}\big), \end{cases} \tag{5.7}$$

without forming matrix $\mathbf{P}$ explicitly. Recall that the approximation of the Schur complement, as calculated in Equation (5.5), is a diagonal matrix, hence solution of the first subequation of (5.7) is straightforward. In the second subequation, it is impractical to calculate the matrix $\widehat{\mathbf{K}}_{\hat{u}\hat{u}}^{-1}$, especially for large problems. In such instances, it may be convenient to use an approximation of $\widehat{\mathbf{K}}_{\hat{u}\hat{u}}$, such as an incomplete LU decomposition. Moreover, we observe that this matrix does not change during the iterative solver, hence the decomposed parts remain constant till the iterative solver converges.

Using the UMFPACK package [189], the matrix $\widehat{\mathbf{K}}_{\hat{u}\hat{u}}$ can be decompossed into parts, namely $\mathbf{L}, \mathbf{U}, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{R}$ which correspond to lower triangular, upper triangular, two permutation matrices and a diagonal matrix of row scaling factors, such that $\mathbf{Q}_1 * (\mathbf{R}^{-1}\widehat{\mathbf{K}}_{\hat{u}\hat{u}}) * \mathbf{Q}_2 = \mathbf{L} * \mathbf{U}$. This package is optimised for the decomposition of unsymmetric sparse matrices, such the ones that arise from the discretisation of the HDG method. Then, the above system is solved according to the following steps, see also [183].

$$\begin{aligned} \widetilde{\boldsymbol{y}} &= -\overset{\sim}{\mathbf{S}}^{-1} \boldsymbol{r}_2, \\ \boldsymbol{b} &= \big(\boldsymbol{r}_1 - \widehat{\mathbf{K}}_{\hat{u}\rho}\widetilde{\boldsymbol{y}}\big), \\ \boldsymbol{c} &= \mathbf{Q}_1 * (\mathbf{R}^{-1}\boldsymbol{b}), \\ \widetilde{\boldsymbol{x}} &= \mathbf{Q}_2 * \big(\mathbf{U}^{-1}(\mathbf{L}^{-1}\boldsymbol{c})\big). \end{aligned} \tag{5.8}$$

The iterative solver that is employed in the following experiments is the GMRES method [190]. We continue the iterations until the relative residual is less than 1×10^{-8}. The computational grids are generated with GiD software [191], unless otherwise specified. The HDG solver is developed with MATLAB software [192]. All the experiments

are performed on a Dell PowerEdge R630 Octa-Core Xeon E5-2667 v4 @ 3.2 GHz.

5.2 Comparison between direct solver and iterative solver with preconditioning

In this subsection, two tests are performed; a three-dimensional steady flow around a sphere in the Stokes limit and a two-dimensional steady flow around a circle. Both flows are solved with a direct solver and a preconditioned iterative solver, using the HDG method described in Chapter 3 and 4 respectively. The direct solver solves the system in (4.16) by performing a LU decomposition of the matrix in the left-hand side. On the other hand, the GMRES iterative solver is implemented as suggested in Section 5.1.

5.2.1 Three-dimensional flow around a sphere in the Stokes limit

The three-dimensional flow around a sphere in the Stokes limit is a common benchmark problem for the Stokes flow. Due to the high number of degrees of freedom, such computations are usually memory-demanding for a direct solver. This flow was previously studied in Section 3.3.3, where the drag on the surface of the sphere was calculated numerically and was compared to the exact value.

First and second-order isoparametric curved meshes and their respective polynomial orders of approximation are considered. Moreover, three levels of h–refinement are used, namely 3,107, 10,680 and 43,682 tetrahedral elements. The corresponding number of degrees of freedom (DOFs) can be seen in Table 5.1 for degrees $k = 1$ and $k = 2$ and all three levels of refinement.

No-slip boundary condition is set on the surface of the sphere, homogeneous Neumann boundary on the outlet and on the rest of the domain walls the following Dirichlet boundary condition, which describes the analytical flow around a sphere, is prescribed:

$$\boldsymbol{u}(\boldsymbol{x}) = \begin{Bmatrix} U_\infty \left(1 - \frac{3R^3 x_1^2}{2r^5} + \frac{R^3}{2r^3} \right) \\ -U_\infty \frac{3R^3 x_1 x_2}{2r^5} \\ -U_\infty \frac{3R^3 x_1 x_3}{2r^5} \end{Bmatrix} \tag{5.9}$$

where R is the sphere radius, r the distance from the center of the sphere, U_∞ the uniform velocity. In this problem, $R = 1\,\mathrm{m}$, $U_\infty = 1\,\mathrm{m/s}$, the dynamic viscosity $\mu = 1\,\mathrm{kg/m-s}$, density $\bar{\rho} = 1\,\mathrm{kg/m^3}$ and the dimensionless parameter $\hat{\tau}$ is set to 2.

The simulation is run twice for every mesh, the first time using a direct solver during the solution of the final system and the second time using a preconditioned iterative GMRES method with the proposed precondition method and stopping criterion of relative residual 1×10^{-8}. In Table 5.1, the timings for the direct and the iterative solver are reported, as well as the reduction percentage of the iterative compared to the direct solver and, finally, the drag error.

k	DOFs	Direct solver (s)	Iterative solver with preconditioning (s)	Time reduction %	Relative drag error with direct solver	Relative drag error with iterative solver
1	58,925	80.6	19.0	76.4 %	$1.95 \cdot 10^{-1}$	$1.95 \cdot 10^{-1}$
1	202,083	1,160	71.6	93.8 %	$1.03 \cdot 10^{-1}$	$1.03 \cdot 10^{-1}$
1	828,158	—	669	—	—	$4.32 \cdot 10^{-2}$
2	114,743	495	37.6	92 %	$6.52 \cdot 10^{-3}$	$6.52 \cdot 10^{-3}$
2	393,486	12,872	192	98.5 %	$5.18 \cdot 10^{-3}$	$5.18 \cdot 10^{-3}$
2	1,612,634	—	3,616	—	—	$1.96 \cdot 10^{-3}$

Table 5.1: *Three-dimensional steady potential flow around a sphere: wall–clock times of the direct and iterative solver, percentage of time decrease for the iterative solver and relative drag error, the dash indicates that the simulation failed to finish due to insufficient memory.*
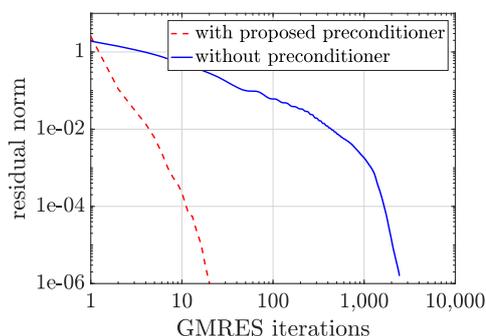
Figure 5.1: *Three-dimensional steady potential flow around a sphere: convergence history of the GMRES iterative solver with and without the use of preconditioner, degree $k = 1$, $DOF = 202,083$.*

The first and most important observation is that the direct solver fails to solve the final system when the finest of the grids are employed due to insufficient memory (the limit is set to 200 GB). However, this does not occur when the preconditioned iterative solver is used. The second observation has to do with the timing, where the preconditioned GMRES solver reduces the required time to by up to 93.8% for the first-order approximation and up to 98.5% for the second-order one.

The relative drag error for both direct and iterative calculations is computed by comparing the computed drag on the surface of the sphere with the analytical value $F_D = 6\pi\mu U_\infty R$. It is worth noting that no difference is observed between the error in the drag computed using the direct and the iterative solver. This is due to the fact that the solution of the iterative solver matches the one provided by the direct solver in its first seven digits.

Next, it should be noted that second-order approximations perform better in terms of error versus time. A second-order approximation on a coarse grid outperforms a first-order approximation on a fine grid. The second-order curvilinear elements approximate the surface of the sphere more accurately than the linear first-order elements, therefore they lead to a reduction of the geometric error and improve the overall accuracy of the calculation of the drag.

While memory limitations can be overcome by solving the system with an iterative

(a) $k = 1, 2$ (b) $k = 3, 4$
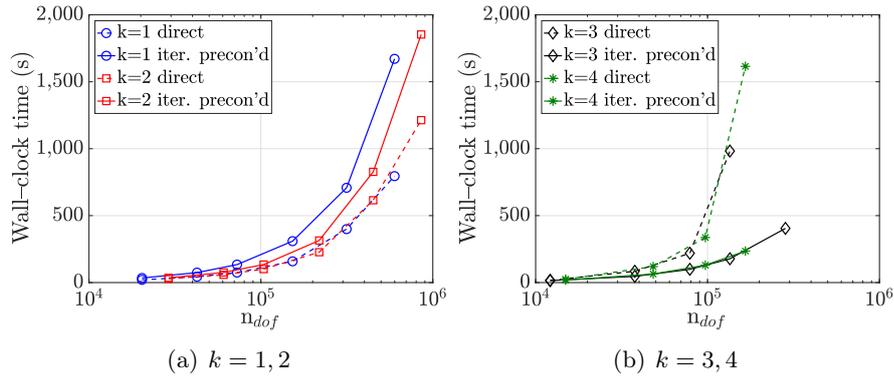
Figure 5.2: *Two-dimensional steady laminar flow around circle: wall-clock times versus degrees of freedom, Re=20.*

solver, the speed in which this solver converges is crucial. For this reason, the convergence history of the iterative solver with and without the use of the precoditioner expressed in terms of iterative solver's residual norm versus number of GMRES iterations is studied. Both residual histories can be seen in Figure 5.1 and the effectiveness of the preconditioner is apparent. While the iterative solver without any precoditioner would require around 2,500 iterations to converge, the preconditioning speeds up the convergence so that the solver converges in this case in less than 20 iterations.

5.2.2 Two-dimensional steady flow around a circle

Next, the low-Reynolds, steady flow around a circle in 2D is studied. This flow was previously presented in detail in Section 4.2.2. In the current study, the scaling behaviour of the wall–clock time for the direct solver and the GMRES iterative solver using the proposed preconditioner is investigated. The comparison is performed for various levels of mesh and degree refinement. The domain, boundary conditions and flow properties are the ones mentioned in Section 4.2.2, giving a Reynolds number $Re = 20$. The stabilization parameter is set $\hat{\tau} = 2$.

It is shown in Figure 5.2 that the observed scaling behaviour is different depending on the approximation degree. For instance, for relatively low orders, $k = 1, 2$, the direct solver appears to be faster than the iterative one, even for a bigger problem size. When linear polynomials are employed the direct solver can be up to 48% faster than the iterative one. This decreases to 27% for the quadratic approximation. However, the iterative solver has a clear advantage in higher orders. The iterative method can save up to 82% wall-clock time with respect to the direct method for a degree order equal to three and 85% if we choose fourth order.

It is also of great interest to observe the accuracy of the solution using low and high-order approximations and the timings of the direct and the iterative solver. The wall-clock time (in seconds) that is required in order to achieve a desired accuracy (relative error) in the quantities of interest for the different degrees is reported in Table 5.2.

	k=1		k=2		k=3		k=4		Desired accuracy
	Dir.	It.	Dir.	It.	Dir.	It.	Dir.	It.	
C_d	20.6	34.6	8.9	11.8	16.5	14.8	9.4	9.0	1×10^{-3}
C_l	—	—	1213	1853	84.4	46.2	24.7	18.4	1×10^{-2}
Δp	795	1671	1213	1853	16.5	14.8	24.7	18.4	1×10^{-3}

Table 5.2: *Two-dimensional steady laminar flow around circle: required wall-clock time (s) to achieve a desired accuracy in the different quantities of interest, both direct (Dir.) and iterative (It.) solvers considered, degree varying from $k = 1$ to $k = 4$, $Re = 20$, dash indicates that the accuracy could not be attained with the available meshes.*

The quantities of interest are the drag coefficient C_d, lift coefficient C_l and the pressure difference between the leading and trailing point of the circle Δp. As a reference solution, an overkilled solution with degree $k = 5$ and 1,625,480 degrees of freedom is used. The different timings arise from the fact that each degree of approximation k requires a different level of mesh refinement in order to achieve the desired accuracy, thereby leading to different wall-clock times.

The superiority of the high degrees ($k = 3, 4$) of approximation is demonstrated. As in the previous simulation, the error introduced by the approximation of the geometry plays an important role in the correct calculation of the quantities of interest. High degrees of approximation lead to more precise results, which are also obtained faster, than low degrees.

5.3   Scaling with Reynolds number

In this subsection the behaviour of the preconditioner for different values of Reynolds numbers is studied. A common drawback of some preconditioners is that they start behaving poorly as the Reynolds number increases. The required number of GMRES iterations to reach convergence increases dramatically, leading to an increase in computational time. An effecient preconditioner should ensure its performance over a range of Reynolds numbers.

For this purpose, the flow inside a microvalve will be studied for different Reynolds numbers. This incompressible flow was presented thoroughly in Section 4.2.3 and it is known that for Reynolds number up to 300 the flow inside the valve remains steady and laminar. The geometry, boundary conditions and flow properties are the same as the ones presented in Section 4.2.3. The Reynolds number lies in the interval [50,300] and is controlled through the velocity at inlet. It is reminded that a parabolic velocity field is specified at inlet and that the mean velocity is used for the calculation of the Reynolds number of the flow. The quantity of interest of this simulation is the diodicity, which measures the effectiveness of this type of valve, refer to Equation (4.23) for its definition. The numerical meshes that are used for this study consist of 5,141 triangular elements of degree two and the polynomial approximation is also of second

(a) Error   (b) Time
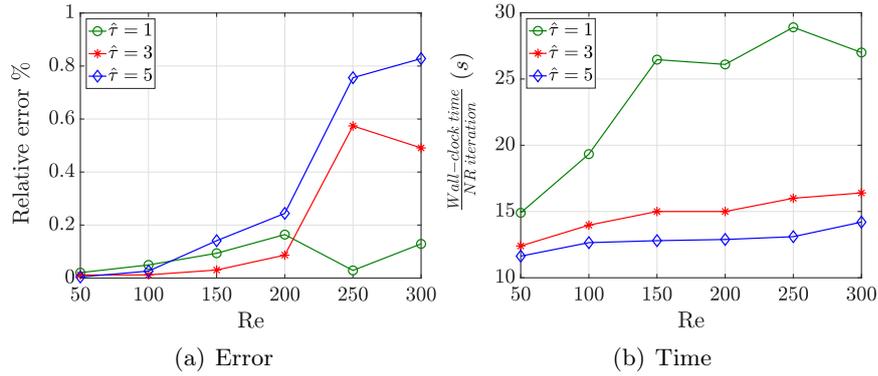
Figure 5.3: *Flow inside a Tesla valve T45-R: error and wall-clock time for the calculation of diodicity using the GMRES solver with the proposed preconditioning and for various values of the stabilization parameter $\hat{\tau}$, Re ranges from 50 to 300.*

order and the reference value was taken from an overkilled solution on a mesh with 37,865 isoparametric second-order triangular elements.

Among the objectives of this simulation is to examine how the preconditioner performs as the Reynolds number increases, namely it can vary from 50 to 300. In Figure 5.3, the percentage error of the diodicity and the measured wall–clock time with respect to Re and for various values of the HDG stabilization parameter $\hat{\tau}$ are reported. The influence of $\hat{\tau}$ on the required average time per non-linear iteration is clear. A small $\hat{\tau}$ appears to scale badly with Reynolds number, while a bigger $\hat{\tau}$ scales significantly better. Observing the percent error on the left part of the same figure, it is evident that a small $\hat{\tau}$ leads to smaller numerical error. Comparing the two plots, the reader can understand that there is a trade-off between accuracy and speed of the iterative solver. Nonetheless, an intermediate value can show an acceptable behaviour in terms of accuracy and speed.

The required iterations for the convergence of the GMRES method in every non-linear iteration depend on the Reynolds number and the parameter $\hat{\tau}$, as shown in Figure 5.4. The initial guess is zero field for velocity and pressure, hence the first non-linear iteration is essentially a Stokes problem, and the iterative solver converges in a small number of iterations. In the remaining non-linear iterations, the number of GMRES iterations generally increases as Reynolds number rises, while the value of $\hat{\tau}$ does play an important role in the overall convergence behaviour. For instance, a value of $\hat{\tau} = 1$ leads to a relatively slow convergence, which is also deteriorated as we move to higher Reynolds, while a value of $\hat{\tau} = 5$ appears to be more robust.

The effect of the scaling term of the Schur complement approximation is also studied for this case. In Table 5.3 the average number of GMRES iterations of all non-linear iterations is reported, along with the required wall–clock time and the error, for various Reynolds numbers and values of the scaling term. Three different scaling terms are used; first, an identity matrix, second a diagonal matrix whose elements are the
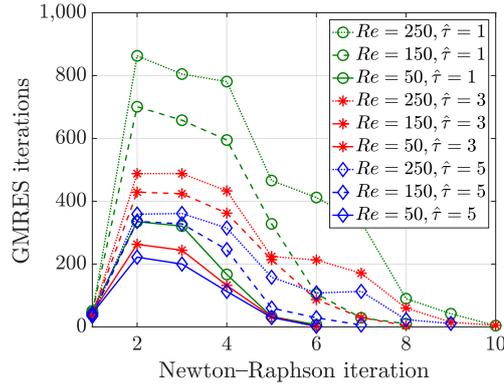
Figure 5.4: *Flow inside a Tesla valve T45-R in forward direction: evolution of number of GMRES iteration for each non-linear Newton–Raphson iteration for various values of Reynolds number and stabilization parameter $\hat{\tau}$.*

	Re=50		Re=150		Re=250	
	Avg	Solve time	Avg	Solve time	Avg	Solve time
Preconditioner scaling	GMRES	/ non-linear	GMRES	/ non-linear	GMRES	/ non-linear
	iters	iters (s)	iters	iters (s)	iters	iters (s)
$\widetilde{\boldsymbol{S}} = \mathbf{I}_{\mathrm{n_{el}}}$	146	15.8	195	17.2	222	18.4
$\widetilde{\boldsymbol{S}} = \mathrm{diag}(\boldsymbol{h}_e)$	108	13.2	145	14.5	166	15.2
$\widetilde{\boldsymbol{S}} = \bar{L}\left(\frac{\nu}{L}+\tau\right)^{-1}\mathrm{diag}(\boldsymbol{h}_e)$	102	12.9	128	12.8	140	13.0

Table 5.3: *Flow inside a Tesla valve T45-R in forward direction: effect of preconditioner scaling on the number of GMRES iterations for various values of Reynolds number, $\hat{\tau} = 5$.*

elements' size and third a diagonal matrix that is used so far and that is presented in Section 5.1. In general, an increase of the GMRES iterations count and the solver time is observed as the Reynolds number increases. However, according to the table, this increase is bigger when the identity matrix is chosen as the scaling of the Schur complement approximation. As for the average solve time per non-linear iteration, the third scaling exhibits robustness as the Reynolds number varies, whereas the other two options see their corresponding solve time increasing.

5.4   Three-dimensional Navier-Stokes simulations

In this subsection, the steady laminar incompressible Navier-Stokes equations in three dimensions using the HDG method will be solved. Solving three-dimensional flows has been so far non-feasible due to the memory requirements, which the solution of the system with a direct solver implies. Therefore, at this point the GMRES iterative solver with the proposed preconditioning of the linear system will be employed. Two cases are studied: the steady laminar flow around a cylinder in three-dimensions, which is a well-known benchmark problem for incompressible Navier-Stokes, as well as the flow inside a brain vessel of a patient with aneurysm, which is a practical case of hemodynamics interest.
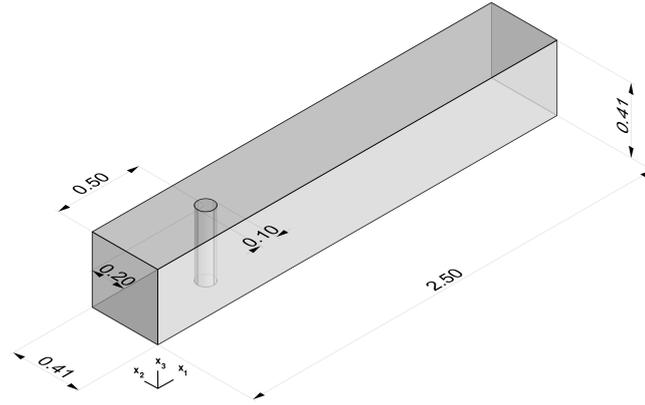
Figure 5.5: *Three-dimensional flow around cylinder: computational domain, dimensions in* m.

5.4.1   Three-dimensional steady laminar flow around cylinder

The steady incompressible flow around a cylinder in three dimensions is a well-known benchmark problem for Navier-Stokes equations. The use of an iterative solver for the solution of the final linear system enables the simulation of the three-dimensional problem, which would be impossible with a direct solver. In this thesis, the case "3D-1Z" presented in [167] will be studied. The computational domain consists of a square duct of $\Omega = ([0, L] \times [0, H] \times [0, H])$ in m, where L and H are respectively the duct length $L = 2.5\,\mathrm{m}$ and the duct size $H = 0.41\,\mathrm{m}$. A cylinder of diameter $D = 0.1\,\mathrm{m}$, whose axis is located at $(0.5, 0.2)$ and is parallel to the z-axis, removed, refer to Figure 5.5.

The boundary conditions are chosen in the following way; on the cylinder surface and on the lateral planes ($x_2 = 0$, $x_2 = H$, $x_3 = 0$, $x_3 = H$) zero velocity is set, on the outlet plane ($x_1 = L$) a homogeneous Neumann boundary condition is selected and on the inlet plane ($x_1 = 0$) the following parabolic velocity field is prescribed:

$$\boldsymbol{u}(\boldsymbol{x}) = \left\{ \begin{array}{c} 16 U_m \frac{x_2 x_3 (H - x_2)(H - x_3)}{H^4} \\ 0 \\ 0 \end{array} \right\}. \tag{5.10}$$

For the characteristic velocity the value $\bar{U} = 0.2\,\mathrm{m/s}$ is chosen, which is related to the maximum velocity at the center of the inlet plane $(0, H/2, H/2)$ as $U_m = 9\bar{U}/4$. The kinematic viscosity is $\nu = 1 \times 10^{-3}\,\mathrm{m^2/s}$, the density $\bar{\rho} = 1.0\,\mathrm{kg/m^3}$, thus the Reynolds number is $Re = 20$, while the dimensionless stabilization parameter is chosen $\hat{\tau} = 2$. The computed velocity magnitude and the pressure field are shown in Figure 5.6. A horizontal slice at the center of the duct cuts the domain except for the cylinder, hence the flow inside the duct is visible.

The calculated quantities of this flow are the lift coefficient in the y-axis, drag coefficient and the pressure drop between the leading edge and the trailing edge of the cylinder. These have a similar definition as in the two-dimensional case, refer to

(a) Velocity magnitude contour   (b) Pressure contour
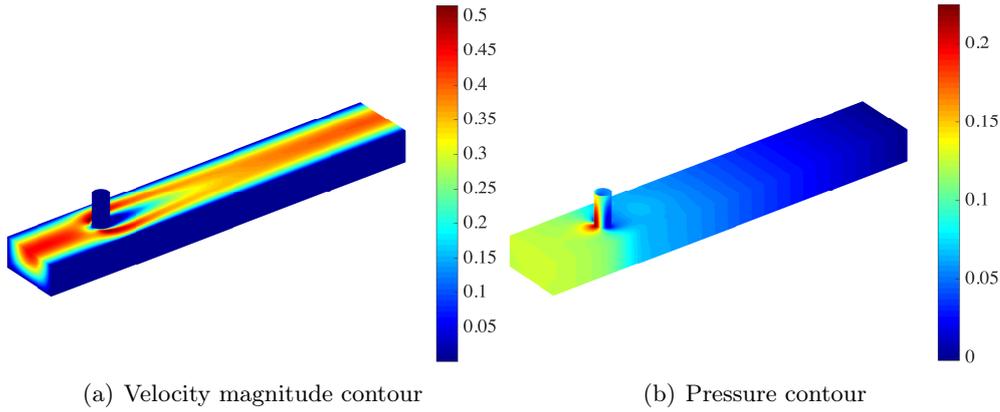
Figure 5.6: *Three-dimensional steady laminar flow around cylinder: velocity magnitude and pressure contours using the third refinement level and second order of approximation, $Re = 20$, $\hat{\tau} = 2$, velocity is expressed in $\mathrm{m/s}$ and pressure in Pa.*

Section 4.2.2, but adapted to the geometry of the three-dimensional cylinder. The lift and drag coefficients are then

$$C_L = \frac{2F_A}{\bar{\rho}\bar{U}^2 DH}, \qquad C_D = \frac{2F_W}{\bar{\rho}\bar{U}^2 DH}, \tag{5.11}$$

where the lift force in y-axis F_A and drag force F_W on the cylinder surface S are

$$F_A = -\int_S \bar{\rho} \mathbf{N}^T \boldsymbol{\sigma}_V \cdot [0, 1, 0]^T \, dS, \qquad F_W = -\int_S \bar{\rho} \mathbf{N}^T \boldsymbol{\sigma}_V \cdot [1, 0, 0]^T \, dS. \tag{5.12}$$

It should be reminded that thanks to the symmetric stress tensor $\boldsymbol{\sigma}_V$, the calculation of the forces on the cylinder is straightforward. Next, the pressure drop is measured between two points

$$\Delta P = \bar{\rho} \left(p(x_{1,a}, x_{2,a}, x_{3,a}) - p(x_{1,e}, x_{2,e}, x_{3,e}) \right), \tag{5.13}$$

where $(x_{1,a}, x_{2,a}, x_{3,a}) = (0.15, 0.2, 0.205)$ and $(x_{1,e}, x_{2,e}, x_{3,e}) = (0.25, 0.2, 0.205)$ are points on the leading edge and the trailing edge, respectively.

For the solution of the flow with approximation degree $k = 1$, meshes with 9,264, 16,146, 32,184, 51,492 and 98,496 elements and a maximum number of 1,834,596 degrees of freedom are considered. For degree $k = 2$ meshes with 9,264, 16,146 and 32,184 isoparametric curved elements and a maximum number of 1,156,140 degrees of freedom are studied. The computational grids, both of first and second order, are generated with the Gmsh software [193]. Table 5.4 summarizes the flow quantities for different levels of mesh refinement and degrees of approximation. There are no exact, analytical values and consequently the results of this study are compared with the bounds reported in [167], which in turn are based on the results of the research groups that participated in this experiment. Out of all measured quantities, lift coefficient takes more degrees of freedom to converge to the interval defined by the benchmark solutions. The superiority of the second-order curvilinear mesh and degree of approximation is clear.

k	n_{el}	n_{dof}	C_L	C_D	ΔP(Pa)	Wall-clock time (s)	Avg GMRES iters
	9,264	169,086	0.00094	6.2120	0.1635	660	184
	16,146	295,884	0.03311	6.2102	0.1681	1,647	249
1	32,184	594,162	0.00617	6.2280	0.1718	5,792	285
	51,492	954,210	0.00584	6.2072	0.1713	15,906	353
	98,496	1,834,596	0.01030	6.1972	0.1709	52,155	450
	9,264	328,908	-0.00017	6.1842	0.1712	2,130	160
2	16,146	576,622	0.00896	6.1936	0.1709	5,493	195
	32,184	1,156,140	0.00918	6.1939	0.1709	20,810	245
	lower bound [167]		0.0080	6.0500	0.1650		
	upper bound [167]		0.0100	6.2500	0.1750		

Table 5.4: *Three-dimensional steady laminar flow around cylinder: quantities of interest and timings for first and second order of approximation as well as different refinement levels, $Re = 20$ and $\hat{\tau} = 2$.*

5.4.2 Three-dimensional brain aneurysm

At this point, the blood flow inside a brain vessel with aneurysm will be simulated. The data are obtained from the open database ANEURISK [194] and correspond to case 32. In this experiment the computed wall shear stress (WSS) is compared to the time-averaged benchmark wall shear stress. The scope is to assess the effectiveness of the preconditioner and the HDG method when applied to a real case of medical interest.

A parabolic Poiseuille velocity profile is prescribed at inflow with a flow rate of $Q = 0.4005$ ml/s. No-slip boundary conditions are prescribed on the vessel walls and homogeneous Neumann boundary conditions at the four outflows. The dynamic viscosity is set to $\mu = 0.04$ g/cm $-$ s, yielding a Reynolds number $Re \approx 40$. For the imposition of the inlet boundary condition we rely on [195]. As for the numerical parameter τ, it is chosen $\hat{\tau} = 3$. Various refinement levels are considered. The finest computational mesh consists of 175,652 first-order tetrahedral elements and 3,197,618 total degrees of freedom. The input geometry, being of STL file format, consists only of straight faces, hence, only first-order meshes are considered for this study.

In this problem, what is of interest is the WSS on the wall of the blood vessel. WSS is one of the factors for the development of cerebral aneurysms, especially in the aneurysm region where it is significantly lower [196]. Using the proposed formulation of the HDG method, the calculation of WSS is facilitated by the fact that the symmetric stress tensor $\boldsymbol{L}$ is one of the variables. Its evaluation is performed following a quite straightforward procedure. First, the traction on the wall nodes i is obtained

$$\boldsymbol{t}_i = \mathbf{N}^T \boldsymbol{\sigma}_{\mathrm{V}}^i. \tag{5.14}$$
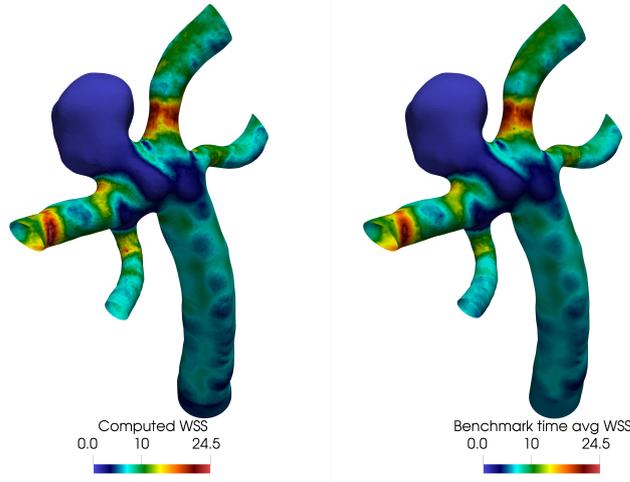
Figure 5.7: *Flow inside a brain vessel with aneurysm: computed wall shear stress versus benchmark values using the highest level of refinement, units in* $\mathrm{dyn/cm^2}$

This vector has a normal and a tangent component to the wall, therefore, the next step is to identify the normal component by calculating the following

$$t_i^n = t_i n_i, \tag{5.15}$$

where n_i is the normal vector to the wall at node i. Last, the WSS is obtained

$$\boldsymbol{T}_{S,i} = \bar{\rho}\sqrt{(\boldsymbol{t}_i)^2 - (\boldsymbol{t}_i^n)^2}, \tag{5.16}$$

subtracting the contribution of the normal component guarantees that $\boldsymbol{T}_{S,i}$ accounts only for the tangent part of the wall stress. It is reminded the stress tensor $\boldsymbol{\sigma}_{\mathrm{V}}$ was initially divided by the constant density $\bar{\rho}$. As a result and in order to obtain the correct units for WSS, density $\bar{\rho}$ has to multiply the above product.

The computed wall shear stress is compared to the benchmark results [194]. It is important to mention that the benchmark results are time-averaged and, due to the low Reynolds number, the flow is considered to be steady and therefore a steady-state computation is performed. In Figure 5.7 it is observed that the computed wall shear stress contour is in good agreement with the time-averaged benchmark shear stress contour. The wall shear stress can be approximated accurately in the main vessel, in the area of the aneurysm as well as in the outflow branches, especially in the narrow part in the upper right branch.

The wall shear stress in the center of the blood vessel along the curve shown in Figure 5.8 of the current computation is compared with the benchmark computation. This curve is formed by the intersection of the vessel wall with the a plane originating from point (3.74, 3.29, 1.39) and being normal to yz plane. Meanwhile, the starting and ending points of the curve are (3.74, 3.29, 1.30) and (3.74, 3.29, 1.04) respectively. The resulting wall shear stress curves are reported in Figure 5.9 for the current solution and the benchmark solution. An agreement between the two solutions is seen, as the
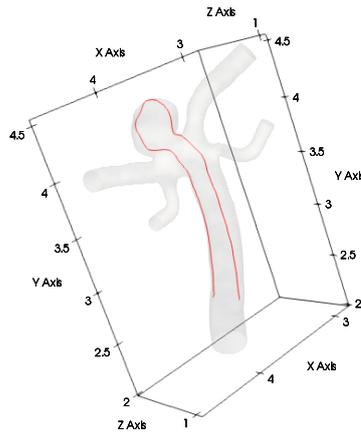
Figure 5.8: *Flow inside a brain vessel with aneurysm: curve upon which the computed shear stress is compared to the benchmark solution*
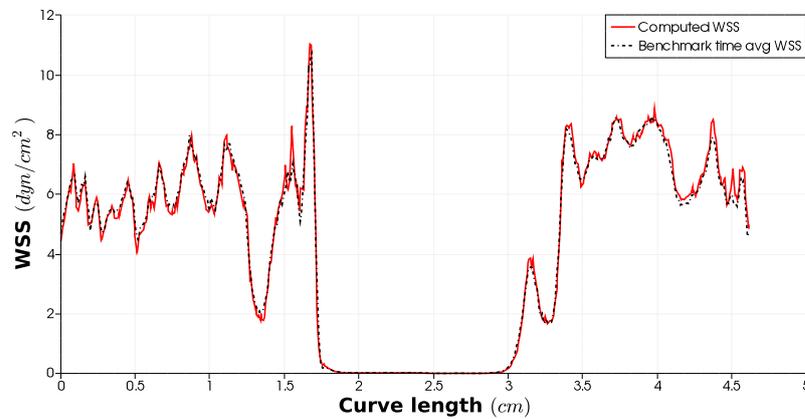
Figure 5.9: *Flow inside a brain vessel with aneurysm: comparison of the computed wall shear stress with the benchmark values along a prescribed curve using the highest level of refinement*

HDG method is able to properly capture the peaks, the troughs of the wall-shear stress and, finally, the aneurysm zone.

Conclusions about the performance of the preconditioner can be deducted by looking in Table 5.5. The finest mesh has 48.4 times more degrees of freedom than the coarsest mesh, while the average number of GMRES iterations increases only by 25%. This suggests that the preconditioner exhibits robust performance as the size of the problem increases.

5.5 Comparison with Finite Element solver

In this subsection we compare our preconditioned iterative HDG method with a classical velocity-pressure formulation (two-variable formulation) of the Navier-Stokes equations. In the latter formulation, the same GMRES iterative method with the preconditioner found in [187], which is structurally similar to the one used so far for the HDG solver,

n_{el}	n_{dof}	Wall-clock time (s)	Avg GMRES iters
4,034	66,071	195	202
16,901	290,240	823	208
50,036	883,571	5,870	215
72,804	1,296,705	8,724	224
175,652	3,197,618	36,975	253

Table 5.5: *Flow inside a brain vessel with aneurysm: degrees of freedom and timings for the different refinement levels that are studied, $\hat{\tau} = 3$.*

is used. For the implementation of the two-variable formulation, the open-source solver FreeFem++ is used [197]. The purpose of this experiment is to compare the first-order HDG Stress-Velocity-Pressure formulation with the classical Velocity-Pressure formulation of the incompressible Navier-Stokes. It is unquestionable that the solvers cannot be compared against one another, since they are developed in different programming languages, not both are parallelized or support the same order, especially high order, approximations. Nonetheless, the formulations can still be compared and in addition, their scaling (Time vs Dofs) can be studied. Two steady incompressible flows are considered: the two-dimensional flow around a circle and the three-dimensional flow around a cylinder.

5.5.1   Two-dimensional steady flow around a circle

The steady flow around a circle in 2D that is solved with an iterative solver after the preconditioning of the system was presented in Section 5.2.2. The same domain and flow settings are used. The evolution of the relative error of the characteristic flow quantities versus the degrees of freedom is reported. There is no analytic value to these quantities, therefore the reference values are obtained through a reference solution. This reference solution is computed using the HDG solver and a fine mesh, consisting of 85,640 isoparametric triangular elements of degree $k = 5$, giving a total number of 1,625,480 degrees of freedom.

In this experiment, the employed meshes vary in degree of approximation ($k = 1$ to 4) and number of degrees of freedom (up to 855,560). In FreeFem++ second-order polynomial space is used for velocity and first-order for pressure.

According to Figure 5.10, high-order approximations (second order or more) appear to estimate lift, drag and pressure drop more accurately than low-order ones. In addition, for a given number of degrees of freedom, the HDG method is more accurate (up to 4 orders of magnitude) than the velocity-pressure formulation employed in FreeFem++. It is also necessary to study how the required time of computation scales with the number of degrees of freedom. In Figure 5.11(a), the respective timings for

(a) Drag coefficient   (b) Lift coefficient

(c) Pressure drop
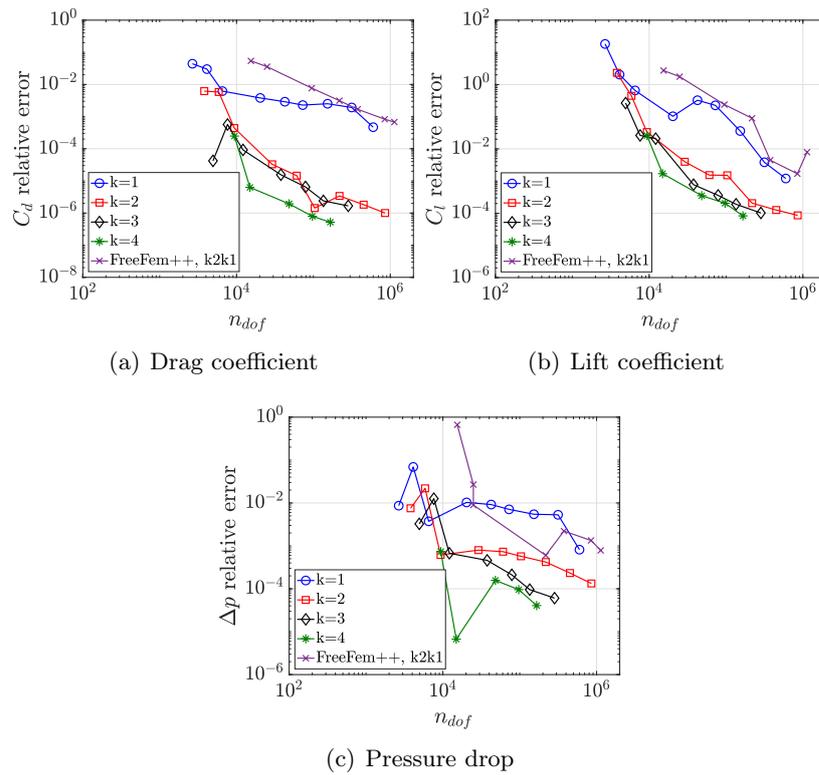
Figure 5.10: *Two-dimensional steady laminar flow around circle: relative error of quantities of interest versus degrees of freedom, $Re = 20$, $\hat{\tau} = 2$*

(a) Two-dimensional flow around circle   (b) Three-dimensional flow around cylinder
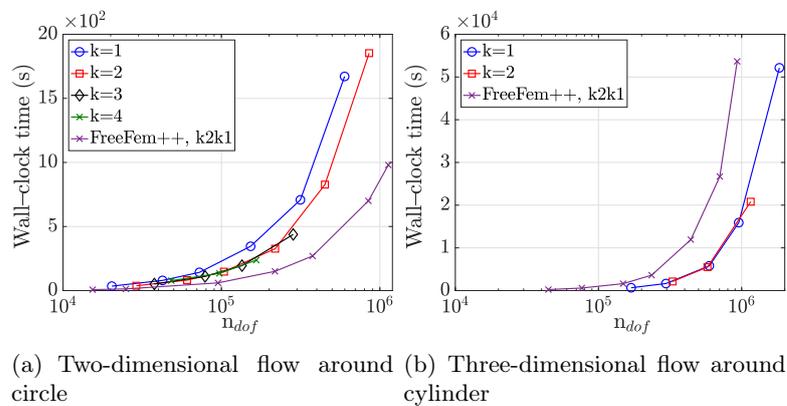
Figure 5.11: *Wall–clock time scaling behaviour versus the number of degrees of freedom for the steady Navier–Stokes equations*

the four approximation degrees using HDG and the velocity-pressure formulation are reported. It is shown that high degrees scale better than low degrees for HDG and that the two-variable formulation using FreeFem++ scales a bit better than the HDG formulation.

(a) Drag coefficient

(b) Lift coefficient

(c) Pressure drop
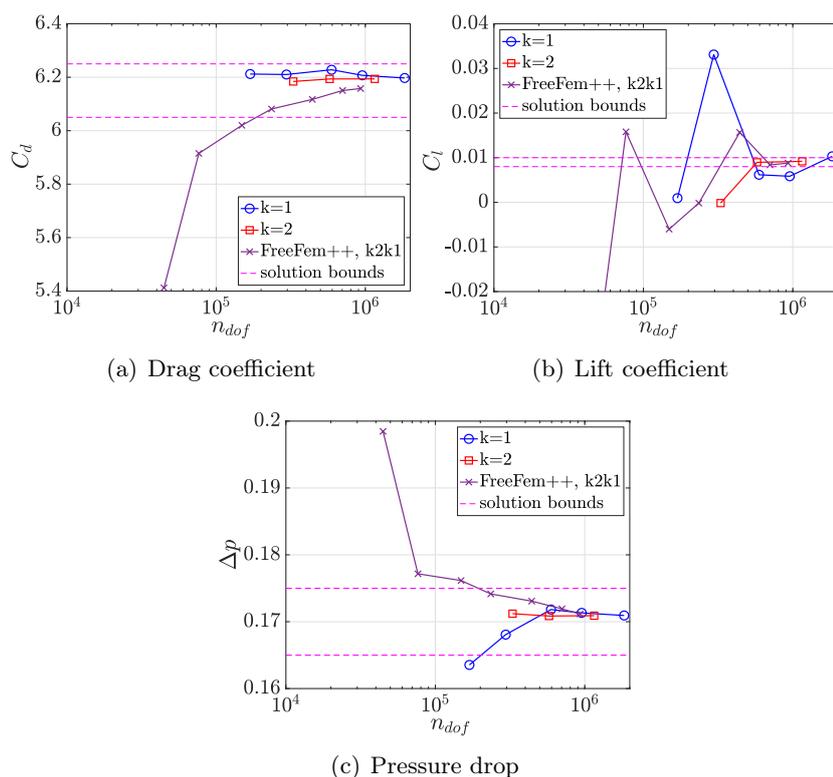
Figure 5.12: *Computed quantities of interest versus degrees of freedom for the calculation of the quantities of interest of the 3D incompressible flow around a cylinder, $Re = 20$*

5.5.2   Three-dimensional flow around a cylinder

The second case is the steady incompressible flow around a cylinder in three dimensions. This flow was presented in Section 5.4.1 and for the current study the same geometry, boundary conditions and flow properties will be used. The computational grids that are employed consist only of tetrahedral elements. For the solution of the flow with the two-variable (Velocity-Pressure) formulation and FreeFem++ the following levels of mesh refinement are considered: 9,264, 16,146, 32,184, 46,536, 98,496, 158,268 and 210,300 elements with a maximum number of 930,332 degrees of freedom. Second-order polynomial space is used for velocity and first-order for pressure. As for the HDG formulation, for degree $k = 1$, meshes with 9,264, 16,146, 32,184 and 46,536 elements and a maximum number of 1,834,596 degrees of freedom are considered. For degree $k = 2$ meshes with 9,264, 16,146 and 32,184 isoparametric curved elements and a maximum number of 1,156,140 degrees of freedom are studied. The polynomial spaces for all variables coincide with the mesh degree, that is first-order polynomials for first-order meshes and second-order for second-order meshes.

The lift and drag coefficients, as well as the pressure drop are evaluated and compared to the benchmark values given by [167], the results of which are shown in Figure 5.12. It is observed that the HDG formulation with $k = 2$ gives satisfying results already with a mesh of 32,184 elements. The two-variable formulation solved with

FreeFem++ solver will eventually converge to the solution of the HDG solver, but only with a very fine mesh. As far as the required wall–clock time with respect to the degrees of freedom is concerned, it is seen in Figure 5.11(b) that the HDG solver scales better for both degrees $k = 1, 2$ when compared to the (Velocity-Pressure) formulation of Navier-Stokes solved with FreeFem++.

5.6   Conclusion

A preconditioning method for the solution of the steady incompressible Navier–Stokes equations with the use of the hybridizable discontinuous Galerkin method has been presented in this section. The preconditioner was built by adapting the block triangular preconditioning approach, common in the standard velocity-pressure formulation of the problem, to the HDG setting.

This preconditioner allows one to employ iterative solvers effectively, making the HDG method and the arising linear systems tractable. This is particularly important for three-dimensional problems. Its effectiveness is demonstrated in a variety of experiments. The preconditioner was found to scale well with respect to problem size and investigated on the way the HDG parameter $\hat{\tau}$ affects the convergence speed of the iterative solver for a range of Reynolds number. This approach was also demonstrated to be competitive with standard methods in terms of computational time and accuracy. Given the advantages that HDG methods offer in accuracy, this is significant. The preconditioned HDG method was also shown to be appropriate for three-dimensional simulations of medical interest.

Chapter 6

Conclusions and Future research

In this PhD thesis, an adaptive high-order solver for the solution of steady and transient engineering incompressible viscous flows was developed. The solver was based on the hybridizabe discontinuous Galerkin (HDG) method, where the symmetry of the strain tensor was enforced strongly. This allowed to impose boundary tractions and calculate wall stresses in a straightforward manner. Moreover, due to the pointwise balance of angular momentum of the velocity, it was possible to construct a superconvergent velocity field and, subsequently, perform a p-adaptivity procedure for steady and unsteady incompressible flows. Large systems that derive from practical problems were also handled with effective preconditioning, which, given the accuracy properties of HDG, makes it a valuable alternative approach for the solution of CFD problems in practical contexts.

6.1 Contributions

Following, the main contributions of this thesis are summarized:

1. **A hybridizable discontinuous Galerkin method for linear elasticity with optimal accuracy properties**. A new HDG formulation for the linear elastic problem with symmetric stress tensor has been proposed. Utilizing the Voigt notation, the symmetric second-order tensors were expressed as vectors with the diagonal and half of the off-diagonal terms. The number of degrees of freedom was therefore reduced, while the symmetry of the stress tensor was enforced.

   Through 2D and 3D simulations, the optimal convergence properties of the displacement field and the stress tensor were demonstrated. In particular, both fields converged optimally with order $k + 1$ in the $\mathcal{L}_2$-norm for approximation polynomials of degree $k \geq 1$. This improved the sub-optimal behaviour of the stress tensor for low-order approximations using equal-order approximations in tradicional HDG methods, without necessitating special enrichment techniques.

   Given the optimal convergence properties of the strain tensor, a post-processed displacement field of degree $k+1$ was constructed through an element-by-element

procedure. After an appropriate treatment of the boundary conditions, the under-determination of the problem was resolved and the post-processed displacement field converged optimally with order $k + 2$.

In addition, attention was given to the study of nearly incompressible materials and the employed method was found to be locking-free while maintaining the convergence rates for all variables.

2. **A hybridizable discontinuous Galerkin method for the Cauchy formulation of the Stokes equation with optimal convergence properties**. The aforementioned HDG formulation was extended to the Stokes problem, that is to the incompressible limit. Voigt notation for symmetric tensors and equal-order approximation for all primal, mixed and hybrid variables were used for the solution of the Cauchy formulation of the Stokes equation.

Regarding the convergence properties, it was verified through various numerical simulations in two and three dimensions that all primal and mixed variables converged with order $k + 1$ in the $\mathcal{L}_2$-norm for approximations polynomials of degree $k \geq 1$. Numerical studies, in two and three dimensions, showed the optimal convergence characteristics of the method for different element types. Moreover, a post-processed velocity field that superconverges with order $k + 2$ was obtained without modifying the discrete spaces in which the variable was sought.

Finally, the drag force on a sphere was calculated and it was shown that using the symmetric stress tensor leads to a straightforward calculation of important qualities of the flow.

3. **Solution of the transient laminar incompressible Navier-Stokes equations using a hybridizable discontinuous Galerkin method**. The afore-mentioned formulation was extended to Navier-Stokes case by introducing a non-linear convective term. First, optimal convergence and superconvergence properties were verified on steady test cases. The method was also able to accurately estimate the quantities of interest for the benchmark flow around a circle in 2D. The use of high-order geometrical and functional approximations clearly outperformed low-order approaches. The HDG solver was then utilized to simulate the flow inside a microvalve, namely a Tesla valve, at different Reynolds numbers.

Exploiting the primal and postprocessed velocity fields, an inexpensive local error indicator was obtained to drive p-adaptivity. Its effectiveness was tested on a variety of cases and it was shown that degree-adapted approximations perform better than uniform-degree ones in terms of accuracy for a given number of degrees of freedom.

Then, the HDG spatial solver was coupled with high-order time integration schemes to achieve high accuracy in space and time. Among the considered

schemes, the family of high-order explicit first stage singly-diagonal implicit Runge-Kutta (ESDIRK) methods could deliver the fastest and most reliable results. The accuracy of the resulting method was verified using classic benchmark tests in 2D.

Eventually, adaptive transient simulations were studied. Two different concepts of adaptivity were explored. The first one was p-adaptivity for transient flows which allowed to adapt the mesh locally in the course of the transient flow, increasing and decreasing the polynomial degree locally, depending on the instant of the calculation and the movement of the flow. The second idea of adaptivity in transient flows dealt with timestep size control, using appropriate estimates of the temporal error. Numerical tests showed the ability of the method to adapt to transient phenomena that take place in the computational domain.

4. **Effective preconditioning for the solution of incompressible flows using a hybridizable discontinuous Galerkin method**. A preconditioning method for the HDG formulation for solution of incompressible Navier-Stokes was presented. The preconditioner was based on the adaptation of a block triangular preconditioner, which is common in standard two-variable formulations, to the HDG global problem. The key element is the use of a scaled diagonal mass matrix in the pressure sparce as approximation of the Schur complement of the global matrix. In this context, the element size is used to locally scale this diagonal matrix.

The preconditioner gave the opportunity to employ iterative solvers effectively and permitted the solution of large-scale problems using the HDG method. Through various experiments it was demonstrated that the preconditioner scaled well with the problem size as well as with the Reynolds number.

The proposed approach was compared with standard methods in terms of computational time and accuracy highlighting its competitiveness. It was also used for the calculation of the blood flow inside a vessel with aneurysm where the ability to calculate quantities of medical interest on the wall of the vessel was exploited.

6.2   On-going research and future developments

Reducing the computational cost of solving the global system remains critical, especially for transient simulations. In this regard, a first topic of investigation is the extension of the preconditioning strategy to the case of transient simulations with the aim of efficiently solving large-scale flow problems using HDG. Some preliminary research has been conducted and possible expressions of the approximation of the Schur complement, taking into consideration the different structure of the final matrix due to the transient term, have been investigated. In order to be able to tackle larger 3D problems, a

framework that is suitable for large-scale parallel computations, such as PETSc or Trilinos, can be used for the implementation of the preconditioner and the method as a whole.

Fast iterative solvers will also allow to perform simulations using higher-order polynomials. In this respect, generation of high-order curved meshes of complex 3D geometries and the effective integration of the NURBS-enhanced finite element framework represent open topics of investigation.

In addition, alternative degree-adaptive strategies remain to be explored, e.g. driven by indicators accounting for the error in quantities of engineering interest. Of course, in order for the presented methodology to be applied to realistic flow problems, turbulence modelling needs to be taken into account. In this framework, HDG of classical closure models, for instance Spalart-Allmaras and $k - \omega$ sst, provide interesting extensions to the presented approach.

Appendices

Appendix A

Implementation details[1]

A standard isoparametric formulation is considered, where the approximation of the primal and mixed variables, $\boldsymbol{u}^h$, p^h and $\boldsymbol{L}^h$, is defined in a reference element $\widetilde{\Omega}$, with local coordinates $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_{\mathrm{n_{sd}}})$, and the approximation of the hybrid variable, $\widehat{\boldsymbol{u}}^h$, is defined in a reference face $\widetilde{\Gamma}$, with local coordinates $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_{\mathrm{n_{sd}}-1})$, as

$$\boldsymbol{u}^h(\boldsymbol{\xi}) = \sum_{j=1}^{\mathrm{n_{en}}} \mathbf{u}_j N_j(\boldsymbol{\xi}), \qquad p^h(\boldsymbol{\xi}) = \sum_{j=1}^{\mathrm{n_{en}}} \mathbf{p}_j N_j(\boldsymbol{\xi}), \qquad \boldsymbol{L}^h(\boldsymbol{\xi}) = \sum_{j=1}^{\mathrm{n_{en}}} \mathbf{L}_j N_j(\boldsymbol{\xi}),$$

$$\hat{u}^h(\boldsymbol{\eta}) = \sum_{j=1}^{\mathrm{n_{fn}}} \hat{\mathbf{u}}_j \hat{N}_j(\boldsymbol{\eta}),$$

where $\mathrm{n_{en}}$ and $\mathrm{n_{fn}}$ denote the number of element and face nodes respectively and N_j and $\hat{N}_j$ are the shape functions used to define the approximation within the reference element and face respectively.

The isoparametric transformation is used to relate local and Cartesian coordinates, namely

$$\boldsymbol{x}(\boldsymbol{\xi}) = \sum_{k=1}^{\mathrm{n_{en}}} \boldsymbol{x}_k N_k(\boldsymbol{\xi}),$$

where $\{\boldsymbol{x}_k\}_{k=1,\ldots,\mathrm{n_{en}}}$ denote the elemental nodal coordinates.

The following matrices are introduced in two dimensions

$$\mathbf{E}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \qquad \mathbf{E}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}^T.$$

Similarly, in three dimensions, the following matrices are defined

$$\mathbf{E}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T \qquad \mathbf{E}_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T \qquad \mathbf{E}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T.$$

These matrices are used to express, in compact form, the matrices $\boldsymbol{\nabla}_{\text{S}}$ and $\mathbf{N}$, defined in Equations (2.8) and (2.12) respectively, as

$$\boldsymbol{\nabla}_{\text{S}} = \sum_{k=1}^{\text{n}_{\text{sd}}} \mathbf{E}_k \frac{\partial}{\partial x_k}, \qquad \mathbf{N} = \sum_{k=1}^{\text{n}_{\text{sd}}} \mathbf{E}_k n_k$$

In addition, the following compact form of the shape functions is introduced

$$\boldsymbol{\mathcal{K}} = \begin{bmatrix} N_1 & N_2 & \dots & N_{\text{n}_{\text{en}}} \end{bmatrix}^T, \qquad \widehat{\boldsymbol{\mathcal{K}}} = \begin{bmatrix} \hat{N}_1 & \hat{N}_2 & \dots & \hat{N}_{\text{n}_{\text{en}}} \end{bmatrix}^T,$$

$$\boldsymbol{\mathcal{N}} = \begin{bmatrix} N_1 \mathbf{I}_{\text{n}_{\text{sd}}} & N_2 \mathbf{I}_{\text{n}_{\text{sd}}} & \dots & N_{\text{n}_{\text{en}}} \mathbf{I}_{\text{n}_{\text{sd}}} \end{bmatrix}^T, \qquad \boldsymbol{\mathcal{M}} = \begin{bmatrix} N_1 \mathbf{I}_{\text{m}_{\text{sd}}} & N_2 \mathbf{I}_{\text{m}_{\text{sd}}} & \dots & N_{\text{n}_{\text{en}}} \mathbf{I}_{\text{m}_{\text{sd}}} \end{bmatrix}^T,$$

$$\boldsymbol{\mathcal{K}}_k = \begin{bmatrix} \frac{\partial N_1}{\partial x_k} \boldsymbol{E}_k^T \mathbf{E} & \frac{\partial N_2}{\partial x_k} \boldsymbol{E}_k^T \mathbf{E} & \dots & \frac{\partial N_{\text{n}_{\text{en}}}}{\partial x_k} \boldsymbol{E}_k^T \mathbf{E} \end{bmatrix}^T, \quad \text{for } k = 1, \dots, \text{n}_{\text{sd}},$$

$$\widehat{\boldsymbol{\mathcal{K}}}_k^n = \begin{bmatrix} \hat{N}_1 n_k \boldsymbol{E}_k^T \mathbf{E} & \hat{N}_2 n_k \boldsymbol{E}_k^T \mathbf{E} & \dots & \hat{N}_{\text{n}_{\text{en}}} n_k \boldsymbol{E}_k^T \mathbf{E} \end{bmatrix}^T, \quad \text{for } k = 1, \dots, \text{n}_{\text{sd}},$$

$$\boldsymbol{\mathcal{N}}_k = \begin{bmatrix} \frac{\partial N_1}{\partial x_k} \boldsymbol{E}_k^T \mathbf{D}^{1/2} & \frac{\partial N_2}{\partial x_k} \boldsymbol{E}_k^T \mathbf{D}^{1/2} & \dots & \frac{\partial N_{\text{n}_{\text{en}}}}{\partial x_k} \boldsymbol{E}_k^T \mathbf{D}^{1/2} \end{bmatrix}^T, \quad \text{for } k = 1, \dots, \text{n}_{\text{sd}},$$

$$\boldsymbol{\mathcal{N}}_k^n = \begin{bmatrix} N_1 n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} & N_2 n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} & \dots & N_{\text{n}_{\text{fn}}} n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} \end{bmatrix}^T, \quad \text{for } k = 1, \dots, \text{n}_{\text{sd}}.$$

$$\widehat{\boldsymbol{\mathcal{N}}} = \begin{bmatrix} \hat{N}_1 \mathbf{I}_{\text{n}_{\text{sd}}} & \hat{N}_2 \mathbf{I}_{\text{n}_{\text{sd}}} & \dots & \hat{N}_{\text{n}_{\text{fn}}} \mathbf{I}_{\text{n}_{\text{sd}}} \end{bmatrix}^T, \qquad \widehat{\boldsymbol{\mathcal{N}}}_\tau = \begin{bmatrix} \hat{N}_1 \boldsymbol{\tau} & \hat{N}_2 \boldsymbol{\tau} & \dots & \hat{N}_{\text{n}_{\text{fn}}} \boldsymbol{\tau} \end{bmatrix}^T,$$

$$\widehat{\boldsymbol{\mathcal{N}}}_k^n = \begin{bmatrix} \hat{N}_1 n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} & \hat{N}_2 n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} & \dots & \hat{N}_{\text{n}_{\text{fn}}} n_k \boldsymbol{E}_k^T \mathbf{D}^{1/2} \end{bmatrix}^T, \quad \text{for } k = 1, \dots, \text{n}_{\text{sd}}.$$

A.1   Linear Elasticity

The matrices and vectors resulting from the discretization using the HDG method of Equation (2.30a) of the local problem of linear elasticity are (refer also to Equation (2.32) for the employed notation)

$$[\mathbf{A}_{LL}]_e = -\sum_{\text{g}=1}^{\text{n}_{\text{ip}}^{\text{e}}} \boldsymbol{\mathcal{M}}(\boldsymbol{\xi}_{\text{g}}^{\text{e}}) \boldsymbol{\mathcal{M}}^T(\boldsymbol{\xi}_{\text{g}}^{\text{e}}) |\mathbf{J}(\boldsymbol{\xi}_{\text{g}}^{\text{e}})| w_{\text{g}}^{\text{e}},$$

$$[\mathbf{A}_{Lu}]_e = \sum_{k=1}^{\text{n}_{\text{sd}}} \sum_{\text{g}=1}^{\text{n}_{\text{ip}}^{\text{e}}} \boldsymbol{\mathcal{N}}_k(\boldsymbol{\xi}_{\text{g}}^{\text{e}}) \boldsymbol{\mathcal{N}}^T(\boldsymbol{\xi}_{\text{g}}^{\text{e}}) |\mathbf{J}(\boldsymbol{\xi}_{\text{g}}^{\text{e}})| w_{\text{g}}^{\text{e}},$$

$$[\mathbf{A}_{L\hat{u}}]_e = \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}} \sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \boldsymbol{\mathcal{N}}_k^n(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \widehat{\boldsymbol{\mathcal{N}}}^T(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) (1 - \chi_{\Gamma_D}(f)),$$

$$[\mathbf{f}_L]_e = \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}} \sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \boldsymbol{\mathcal{N}}_k^n(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \boldsymbol{u}_D\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) \chi_{\Gamma_D}(f),$$

where $\mathrm{n}_{\mathrm{fa}}^e$ is the number of faces, $\Gamma_{e,j}$ for $j = 1, \ldots, \mathrm{n}_{\mathrm{fa}}^e$ of the element Ω_e and χ_{Γ_D} is the indicator function of Γ_D, i.e.

$$\chi_{\Gamma_D}(f) = \begin{cases} 1 & \text{if } \Gamma_{e,j} \cap \Gamma_D \neq \emptyset \\ 0 & \text{otherwise} \end{cases}.$$

In the above expressions, $\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{e}}$ and $w_{\mathrm{g}}^{\mathrm{e}}$ are the $\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}$ integration points and weights defined on the reference element and $\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}$ and $w_{\mathrm{g}}^{\mathrm{f}}$ are the $\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}$ integration points and weights defined on the reference face.

Similarly, the matrices and vectors resulting from the discretization of Equation (2.30b) of the local problem of linear elasticity are

$$[\mathbf{A}_{uu}]_e = \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \widehat{\boldsymbol{\mathcal{N}}}_\tau^T(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}},$$

$$[\mathbf{A}_{u\hat{u}}]_e = \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \widehat{\boldsymbol{\mathcal{N}}}_\tau^T(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) (1 - \chi_{\Gamma_D}(f)),$$

$$[\mathbf{f}_u]_e = \sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{e}}) \boldsymbol{f}\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{e}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{e}})| w_{\mathrm{g}}^{\mathrm{e}} + \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \boldsymbol{\tau} \boldsymbol{u}_D\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) \chi_{\Gamma_D}(f).$$

Finally, the matrices and vectors resulting from the discretization of Equation (2.31) of the global problem of linear elasticity are (refer also to Equation (2.33) for the employed notation)

$$[\mathbf{A}_{\hat{u}\hat{u}}]_e = -\sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \widehat{\boldsymbol{\mathcal{N}}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \widehat{\boldsymbol{\mathcal{N}}}_\tau^T(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) (1 - \chi_{\Gamma_D}(f)),$$

$$[\mathbf{f}_{\hat{u}}]_e = -\sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e} \left(\sum_{\mathrm{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}} \widehat{\boldsymbol{\mathcal{N}}}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \boldsymbol{t}\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) \chi_{\Gamma_N}(f),$$

where χ_{Γ_N} is the indicator function of Γ_N.

A.2 Stokes flow

The matrices and vectors resulting from the discretization of Equation (3.17a) of the local problem of Stokes flow have the same definition as in the case of linear elasticity. Therefore, for the definition of $[\mathbf{A}_{LL}]_e$, $[\mathbf{A}_{Lu}]_e$, $[\mathbf{A}_{L\hat{u}}]_e$ and $[\mathbf{f}_L]_e$ the reader is referred to Appendix A.1. The reader is also referred to Equation (3.18) for information on the employed notation.

Next, as far as the discretization of Equation (3.17b) of the local problem of Stokes flow is concerned, matrices $[\mathbf{A}_{uu}]_e$ and $[\mathbf{A}_{u\hat{u}}]_e$ are defined in the same way as in Appendix A.1. The rest of the matrices and vectors are

$$[\mathbf{A}_{up}]_e = \sum_{k=1}^{\mathrm{n_{sd}}} \sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^e}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}}) \boldsymbol{\mathcal{K}}_k^T(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}})| w_{\mathsf{g}}^{\mathsf{e}},$$

$$[\mathbf{f}_u]_e = \sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^e}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}}) \boldsymbol{s}\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{e}})| w_{\mathsf{g}}^{\mathsf{e}} + \sum_{f=1}^{\mathrm{n_{fa}^e}} \left(\sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^f}} \boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}}) \boldsymbol{\tau} \boldsymbol{u}_D\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})| w_{\mathsf{g}}^{\mathsf{f}} \right) \chi_{\Gamma_D}(f).$$

The discretization of Equation (3.17c) leads to the formation of the following matrix and vector

$$[\mathbf{A}_{p\hat{u}}]_e = \sum_{f=1}^{\mathrm{n_{fa}^e}} \left(\sum_{k=1}^{\mathrm{n_{sd}}} \sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^f}} \boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}}) \widehat{\boldsymbol{\mathcal{K}}}_k^n(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})| w_{\mathsf{g}}^{\mathsf{f}} \right) \left(1 - \chi_{\Gamma_D}(f)\right),$$

$$[\mathbf{f}_p]_e = \sum_{f=1}^{\mathrm{n_{fa}^e}} \left(\sum_{k=1}^{\mathrm{n_{sd}}} \sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^f}} \widehat{\boldsymbol{\mathcal{K}}}_k^n(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}}) \boldsymbol{u}_D\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})\right) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})| w_{\mathsf{g}}^{\mathsf{f}} \right) \chi_{\Gamma_D}(f).$$

In the last equation of the local problem Equation (3.17d) the following matrix is used

$$[\boldsymbol{a}_{\rho p}]_e = \frac{1}{|\partial\Omega_e|} \sum_{f=1}^{\mathrm{n_{fa}^e}} \sum_{\mathsf{g}=1}^{\mathrm{n_{ip}^f}} \widehat{\boldsymbol{\mathcal{K}}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathsf{f}})| w_{\mathsf{g}}^{\mathsf{f}}.$$

The matrices and vectors resulting from the discretization of Equation (3.19a) of the global problem of Stokes flow are $[\mathbf{A}_{L\hat{u}}^T]_e$, $[\mathbf{A}_{u\hat{u}}^T]_e$ and $[\mathbf{A}_{p\hat{u}}^T]_e$, whose transpose matrices have been previously defined, as well as the matrix $[\mathbf{A}_{\hat{u}\hat{u}}]_e$ and the vector $[\mathbf{f}_{\hat{u}}]_e$, which have the same definition as in the linear elasticity problem (refer to Appendix A.1). The reader is also referred to Equation (3.20) for more information on the employed notation.

The last equation of the global problem, Equation (3.19b), involves the following matrix and vector

$$[\mathbf{A}_{\rho\hat{u}}]_e = \sum_{f=1}^{n_{\mathrm{fa}}^e} \left(\sum_{k=1}^{n_{\mathrm{sd}}} \sum_{\mathrm{g}=1}^{n_{\mathrm{ip}}^{\mathrm{f}}} \widehat{\mathcal{K}}_k^n(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) \left(1 - \chi_{\Gamma_D}(f) \right),$$

$$[\mathbf{f}_\rho]_e = -\sum_{f=1}^{n_{\mathrm{fa}}^e} \left(\sum_{k=1}^{n_{\mathrm{sd}}} \sum_{\mathrm{g}=1}^{n_{\mathrm{ip}}^{\mathrm{f}}} \widehat{\mathcal{K}}_k^n(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \boldsymbol{u}_D \left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}}) \right) |\mathbf{J}(\boldsymbol{\xi}_{\mathrm{g}}^{\mathrm{f}})| w_{\mathrm{g}}^{\mathrm{f}} \right) \chi_{\Gamma_D}(f).$$

A.3 Navier-Stokes equations

The main difficulty concerning the Navier-Stokes equations concerns the non-linearity of the momentum equation. In this section, the linearization procedure of the Navier-Stokes equations is presented. The procedure for both steady and transient case of the Navier-Stokes equations is presented.

A.3.1 Linearization of steady Navier-Stokes equations

Let us first recall the weak from of the momentum equation of the steady Navier-Stokes, Equation (4.27b). The equations has two non-linear terms, namely $(\boldsymbol{u}_e^h \otimes \boldsymbol{u}_e^h)_V$ and $(\widehat{\boldsymbol{u}} \otimes \widehat{\boldsymbol{u}})_V)$. Considering the vector $\mathbf{x} = \begin{bmatrix} \mathbf{L} & \mathbf{u} & \mathbf{p} & \hat{\mathbf{u}} & \rho \end{bmatrix}^T$, the equation might be rewritten as

$$\mathbf{F}_2(\mathbf{x}) = \mathbf{0} \tag{A.1}$$

Following, the linearization according to Newton-Raphson method consists in solving the following linearized equation

$$\mathbf{F}_2(\mathbf{x}^r) + \left. \frac{\partial \mathbf{F}_2(\mathbf{x})}{\partial \mathbf{x}} \right|_r \Delta\mathbf{x}^r = \mathbf{0} \Leftrightarrow$$

$$\begin{bmatrix} \frac{\partial \mathbf{F}_2}{\partial \mathbf{L}} & \frac{\partial \mathbf{F}_2}{\partial \boldsymbol{u}} & \frac{\partial \mathbf{F}_2}{\partial p} \end{bmatrix} \begin{Bmatrix} \Delta\mathbf{L}^r \\ \Delta\mathbf{u}^r \\ \Delta\mathbf{p}^r \end{Bmatrix} = - \begin{bmatrix} \frac{\partial \mathbf{F}_2}{\partial \widehat{\boldsymbol{u}}} & \frac{\partial \mathbf{F}_2}{\partial \rho} \end{bmatrix} \begin{Bmatrix} \Delta\hat{\mathbf{u}}^r \\ \Delta\rho^r \end{Bmatrix} - \mathbf{F}_2(\mathbf{x}^r) \tag{A.2}$$

where $\Delta\mathbf{x}^r = \mathbf{x}^{r+1} - \mathbf{x}^r$, the difference of vector $\mathbf{x}$ between two successive non-linear iterations r, $r+1$.

After substituting $\Delta\mathbf{x}$ into Equation (A.2) and adding the rest of the equations of the local problem, Equations (4.27a), (4.27c) and (4.27d), the discretized local problem has the matrix form of Equation (4.11). The submatrices with superscript r need to be evaluated in every non-linear iteration and their value will depend on $\mathbf{x}^r$. The exact definitions are

$$[\mathbf{A}_{uu}]_e^r = -\sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e}\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\boldsymbol{\tau}\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{\mathcal{N}}}_{\tau}^{T}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}} + \sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}}\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}}\boldsymbol{\nabla}_{\!\mathrm{S}}\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\boldsymbol{u}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}}))\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|w_{\mathsf{g}}^{\mathrm{e}}$$

$$+ \sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}}\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}}\boldsymbol{\mathcal{K}}_k(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\boldsymbol{u}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}}))\mathbf{I}_{\mathrm{n}_{\mathrm{sd}}}\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|w_{\mathsf{g}}^{\mathrm{e}},$$

$$[\mathbf{A}_{u\hat{u}}]_e^r = \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e}\left(-\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{\mathcal{N}}}_{\tau}^{T}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}} + \sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}}\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\widehat{\boldsymbol{u}}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{n}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\boldsymbol{\nabla}_{\!\mathrm{S}}\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{\mathcal{K}}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}}\right.$$

$$\left.+ \sum_{k=1}^{\mathrm{n}_{\mathrm{sd}}}\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{u}}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{n}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{\mathcal{K}}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}}\right)(1 - \chi_{\Gamma_D}(f)),$$

$$[\mathbf{f}_u]_e^r = -\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}}\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\boldsymbol{s}\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\right)|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|w_{\mathsf{g}}^{\mathrm{e}} + \sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}}\boldsymbol{\nabla}_{\!\mathrm{S}}\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\boldsymbol{u}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}}))\boldsymbol{u}^{r,T}(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}}))|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|w_{\mathsf{g}}^{\mathrm{e}}$$

$$+ \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e}\left(\sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\boldsymbol{u}_D^{T}(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{n}(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\boldsymbol{u}_D(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}} - \sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{f}}}\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\boldsymbol{\tau}\boldsymbol{u}_D\left(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\right)|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}}\right)\chi_{\Gamma_D}(f)$$

$$- \sum_{f=1}^{\mathrm{n}_{\mathrm{fa}}^e}\widehat{\boldsymbol{u}}^{r,T}(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{n}(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))\boldsymbol{\mathcal{N}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})\widehat{\boldsymbol{u}}^r(\boldsymbol{x}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}}))|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{f}})|w_{\mathsf{g}}^{\mathrm{f}}(1 - \chi_{\Gamma_D}(f)).$$

A.3.2 Time-dependent Navier-Stokes equations

When the time derivative is added to Navier-Stokes equation, the mass matrix $\mathbf{M}_{uu}$ appears, as shown in Equation (4.28). Independently of the employed time-marching scheme, this mass matrix has the following definition

$$[\mathbf{M}_{uu}]_e = \sum_{\mathsf{g}=1}^{\mathrm{n}_{\mathrm{ip}}^{\mathrm{e}}}\boldsymbol{\mathcal{K}}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})\boldsymbol{\mathcal{K}}^{T}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|\mathbf{J}(\boldsymbol{\xi}_{\mathsf{g}}^{\mathrm{e}})|w_{\mathsf{g}}^{\mathrm{e}}.$$

A.3.3 Scaling of the approximation of the Schur complement

The Schur complement is defined as $\boldsymbol{S} = \widehat{\boldsymbol{K}}_{\hat{u}\rho}^{T}\,\widehat{\boldsymbol{K}}_{\hat{u}\hat{u}}^{-1}\,\widehat{\boldsymbol{K}}_{\hat{u}\rho}$, with the submatrices of the global problem in Equation (4.16). The goal is to find the scaling of the Schur complement. In Equation (4.15), we see that the right-hand side of Equation (4.15a) has units $\mathrm{m}^5/\mathrm{s}^3$. In fact, velocity $\widehat{\boldsymbol{w}}$ has units $\mathrm{m/s}$, traction $\boldsymbol{t}$ has units $\mathrm{m}^2/\mathrm{s}^2$ in three dimensions (is divided by density) and the surface is expressed in units m^2.

Therefore, the left-hand side should have the same units as the right-hand side, that is $\mathrm{m}^5/\mathrm{s}^3$. The submatrix $\widehat{\boldsymbol{K}}_{\hat{u}\hat{u}}$ is thus expressed in $\mathrm{m}^4/\mathrm{s}^2$ while the submatrix

$\widehat{\boldsymbol{K}}_{\hat{u}\rho}^{T}$ in m^3/s. From Equation (4.15b) the submatrix $\widehat{\boldsymbol{K}}_{\hat{u}\rho}$ has units m^2. Finally, the Schur complement has units:

$$\boldsymbol{S} = \widehat{\boldsymbol{K}}_{\hat{u}\rho}^{T}\,\widehat{\boldsymbol{K}}_{\hat{u}\hat{u}}^{-1}\,\widehat{\boldsymbol{K}}_{\hat{u}\rho} \rightarrow \mathrm{m}^2\frac{\mathrm{s}^2}{\mathrm{m}^4}\frac{\mathrm{m}^3}{\mathrm{s}} = \mathrm{m}\cdot\mathrm{s}. \qquad (\mathrm{A.3})$$

Bibliography

[1] N. Kroll, H. Bieler, H. Deconinck, V. Couaillier, H. van der Ven, K. Sorensen, ADIGMA–A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications: Results of a Collaborative Research Project Funded by the European Union, 2006-2009, Vol. 113, Springer Science & Business Media, 2010. `doi:10.1007/978-3-642-03707-8`.

[2] Z. J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H. T. Huynh, et al., High-order CFD methods: current status and perspective, International Journal for Numerical Methods in Fluids 72 (8) (2013) 811–845. `doi:/10.1002/fld.3767`.

[3] Z. Wang, High-order methods for the Euler and Navier–Stokes equations on unstructured grids, Progress in Aerospace Sciences 43 (1-3) (2007) 1–41. `doi:10.1016/j.paerosci.2007.05.001`.

[4] P. Solin, K. Segeth, I. Dolezel, Higher-order finite element methods, Chapman and Hall/CRC, 2003.

[5] A. Huerta, A. Angeloski, X. Roca, J. Peraire, Efficiency of high-order elements for continuous and discontinuous Galerkin methods, International Journal for Numerical Methods in Engineering 96 (9) (2013) 529–560. `doi:10.1002/nme.4547`.

[6] Z. Wang, A perspective on high-order methods in computational fluid dynamics, Science China Physics, Mechanics & Astronomy 59 (1) (2016) 614701. `doi:10.1007/s11433-015-5706-3`.

[7] F. Khatami, E. Weide, H. Hoeijmakers, Numerical issues in higher-order accurate simulations of flows with vortex cavitation, in: Proceedings of the 6th European Conference on Computational Fluid Dynamics, 20-25 July 2014, Barcelona, Spain.

[8] M. Dumbser, U. Iben, C.-D. Munz, Efficient implementation of high order unstructured WENO schemes for cavitating flows, Computers & Fluids 86 (2013) 141–168. `doi:10.1016/j.compfluid.2013.07.011`.

[9] Y. Bazilevs, V. M. Calo, Y. Zhang, T. J. Hughes, Isogeometric fluid–structure interaction analysis with applications to arterial blood flow, Computational Mechanics 38 (4-5) (2006) 310–322. `doi:10.1007/s00466-006-0084-3`.

[10] A. Van Zuijlen, A. de Boer, H. Bijl, Higher-order time integration through smooth mesh deformation for 3D fluid–structure interaction simulations, Journal of Computational Physics 224 (1) (2007) 414–430. `doi:10.1016/j.jcp.2007.03.024`.

[11] A. Corsini, F. Rispoli, A. Santoriello, A variational multiscale higher-order finite element formulation for turbomachinery flow computations, Computer methods in applied mechanics and engineering 194 (45-47) (2005) 4797–4823. `doi:10.1016/j.cma.2004.11.013`.

[12] G. Ashcroft, K. Heitkamp, E. Kuegeler, High-order accurate implicit runge-kutta schemes for the simulation of unsteady flow phenomena in turbomachinery, in: 5th European Conference on Computational Fluid Dynamics, 14-17 June 2010, Lisbon, Portugal.

[13] T. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Courier Corporation, 2012.

[14] O. C. Zienkiewicz, R. L. Taylor, P. Nithiarasu, J. Zhu, The finite element method, Vol. 3, McGraw-hill London, 1977.

[15] B. Szabó, I. Babuška, Finite Element Analysis, John Wiley & Sons, New York, 1991.

[16] B. Cockburn, G. E. Karniadakis, C.-W. Shu, Discontinuous Galerkin Methods, Springer Berlin Heidelberg, 2000. `doi:10.1007/978-3-642-59721-3`.

[17] D. Di Pietro, A. Ern, Mathematical aspects of discontinuous Galerkin methods, Vol. 69 of Mathématiques & Applications, Springer, Berlin, Heidelberg, 2012. `doi:10.1007/978-3-642-22980-0`.

[18] R. J. Guyan, Reduction of stiffness and mass matrices, AIAA journal 3 (2) (1965) 380–380. `doi:10.2514/3.2874`.

[19] B. Fraeijs de Veubeke, Displacement and equilibrium models in the finite element method, Stress analysis (1965) 145–197`doi:http://hdl.handle.net/2268/79430`.

[20] B. Cockburn, J. Gopalakrishnan, A characterization of hybridized mixed methods for second order elliptic problems, SIAM J. Numer. Anal. 42 (1) (2004) 283–301. `doi:10.1137/S0036142902417893`.

[21] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems, SIAM Journal on Numerical Analysis 47 (2) (2009) 1319–1365. doi:10.1137/070706616.

[22] B. Cockburn, J. Guzmán, H. Wang, Superconvergent discontinuous Galerkin methods for second-order elliptic problems, Mathematics of Computation 78 (2009) 1–24. doi:10.1090/S0025-5718-08-02146-7.

[23] N. C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for linear convection-diffusion equations, Journal of Computational Physics 228 (9) (2009) 3232–3254. doi:10.1016/j.jcp.2009.01.030.

[24] N. Nguyen, J. Peraire, B. Cockburn, A hybridizable discontinuous Galerkin method for Stokes flow, Computer Methods in Applied Mechanics and Engineering 199 (9-12) (2010) 582–597. doi:10.1016/J.CMA.2009.10.007.

[25] N. C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier-Stokes equations, Journal of Computational Physics 230 (4) (2011) 1147–1170. doi:10.1016/j.jcp.2010.10.032.

[26] J. Peraire, N. Nguyen, B. Cockburn, A hybridizable discontinuous Galerkin method for the compressible Euler and Navier-Stokes equations, 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Expositiondoi:10.2514/6.2010-363.

[27] S. Adams, B. Cockburn, A mixed finite element method for elasticity in three dimensions, Journal of Scientific Computing 25 (3) (2005) 515–521. doi:10.1007/s10915-004-4807-3.

[28] D. N. Arnold, G. Awanou, R. Winther, Finite elements for symmetric tensors in three dimensions, Mathematics of Computation 77 (263) (2008) 1229–1251. doi:10.1090/S0025-5718-08-02071-1.

[29] D. Arnold, R. Falk, R. Winther, Finite element exterior calculus, homological techniques, and applications, Acta Numerica 15 (2006) 1–155. doi:10.1017/S0962492906210018.

[30] S. C. Brenner, L. R. Scott, The mathematical theory of finite element methods, 3rd Edition, Vol. 15 of Texts in Applied Mathematics, Springer, New York, 2008. doi:10.1007/978-0-387-75934-0.

[31] D. Braess, Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics, Cambridge University Press, 2001.

[32] S. Brenner, L.-Y. Sung, Linear finite element methods for planar linear elasticity, Mathematics of Computation 59 (200) (1992) 321–338. doi:10.2307/2153060.

[33] M. Crouzeix, P.-A. Raviart, Conforming and nonconforming finite element methods for solving the stationary Stokes equations. I, Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge 7 (R-3) (1973) 33–75.

[34] E. Reissner, On a variational theorem in elasticity, J. Math. Phys. Camb. 29 (1-4) (1950) 90–95. doi:10.1002/sapm195029190.

[35] P. Hansbo, M. Larson, Discontinuous Galerkin methods for incompressible and nearly incompressible elasticity by Nitsche's method, Computer Methods in Applied Mechanics and Engineering 191 (17-18) (2002) 1895–1908. doi:10.1016/S0045-7825(01)00358-9.

[36] P. Hansbo, M. Larson, Discontinuous Galerkin and the Crouzeix-Raviart element: application to elasticity, M2AN. Mathematical Modelling and Numerical Analysis 37 (1) (2003) 63–72. doi:10.1051/m2an:2003020.

[37] D. N. Arnold, Mixed finite element methods for elliptic problems, Computer Methods in Applied Mechanics and Engineering 82 (1-3) (1990) 281–300, reliability in computational mechanics (Austin, TX, 1989). doi:10.1016/0045-7825(90)90168-L.

[38] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers, ESAIM: Mathematical Modelling and Numerical Analysis 8 (R2) (1974) 129–151.
URL http://eudml.org/doc/193255

[39] D. Boffi, F. Brezzi, M. Fortin, Reduced symmetry elements in linear elasticity, Communications on Pure and Applied Analysis 8 (1) (2009) 95–121. doi:10.3934/cpaa.2009.8.95.

[40] B. Fraeijs de Veubeke, Stress function approach, Proceedings of the world congress on finite element methods in structural mechanics, Rapport du LTAS, Université de Liège, http://hdl.handle.net/2268/205875 (1975).
URL http://hdl.handle.net/2268/205875

[41] M. Amara, J. M. Thomas, Equilibrium finite elements for the linear elastic problem, Numerische Mathematik 33 (4) (1979) 367–383. doi:10.1007/BF01399320.

[42] D. N. Arnold, R. S. Falk, A new mixed formulation for elasticity, Numerische Mathematik 53 (1-2) (1988) 13–30. doi:10.1007/BF01395876.

[43] E. Stein, R. Rolfes, Mechanical conditions for stability and optimal convergence of mixed finite elements for linear plane elasticity, Computer Methods in Applied Mechanics and Engineering 84 (1) (1990) 77–95. `doi:10.1016/0045-7825(90)90090-9`.

[44] D. N. Arnold, F. Brezzi, J. Douglas, Jr., PEERS: a new mixed finite element for plane elasticity, Japan Journal of Applied Mathematics 1 (2) (1984) 347–367. `doi:10.1007/BF03167064`.

[45] M. E. Morley, A family of mixed finite elements for linear elasticity, Numerische Mathematik 55 (6) (1989) 633–666. `doi:10.1007/BF01389334`.

[46] R. Stenberg, On the construction of optimal mixed finite element methods for the linear elasticity problem, Numerische Mathematik 48 (4) (1986) 447–462. `doi:10.1007/BF01389651`.

[47] R. Stenberg, A family of mixed finite elements for the elasticity problem, Numerische Mathematik 53 (5) (1988) 513–538. `doi:10.1007/BF01397550`.

[48] R. Stenberg, Two low-order mixed methods for the elasticity problem, in: The mathematics of finite elements and applications, VI (Uxbridge, 1987), Academic Press, London, 1988, pp. 271–280.

[49] F. Brezzi, J. Douglas, Jr., L. D. Marini, Recent results on mixed finite element methods for second order elliptic problems, in: Vistas in applied mathematics, Transl. Ser. Math. Engrg., Optimization Software, New York, 1986, pp. 25–43.

[50] M. Farhloul, M. Fortin, Dual hybrid methods for the elasticity and the Stokes problems: a unified approach, Numerische Mathematik 76 (4) (1997) 419–440. `doi:10.1007/s002110050270`.

[51] D. N. Arnold, R. Winther, Mixed finite elements for elasticity, Numerische Mathematik 92 (3) (2002) 401–419. `doi:10.1007/s002110100348`.

[52] A. Pechstein, J. Schöberl, Tangential-displacement and normal-normal-stress continuous mixed finite elements for elasticity, Mathematical Models and Methods in Applied Sciences 21 (8) (2011) 1761–1782. `doi:10.1142/S0218202511005568`.

[53] F. Brezzi, M. Fortin, Mixed and hybrid finite elements methods, Springer series in computational mathematics, Springer-Verlag, 1991.

[54] D. Arnold, R. Winther, Nonconforming mixed elements for elasticity, Mathematical Models and Methods in Applied Sciences 13 (3) (2003) 295–307, dedicated to Jim Douglas, Jr. on the occasion of his 75th birthday.

[55] H.-Y. Man, J. Hu, Z.-C. Shi, Lower order rectangular nonconforming mixed finite element for the three-dimensional elasticity problem, Mathematical Models and Methods in Applied Sciences 19 (1) (2009) 51–65. `doi:10.1142/S0218202509003358`.

[56] D. Arnold, G. Awanou, R. Winther, Nonconforming tetrahedral mixed finite elements for elasticity, Mathematical Models and Methods in Applied Sciences 24 (4) (2014) 783–796. `doi:10.1142/S021820251350067X`.

[57] J. Gopalakrishnan, J. Guzmán, Symmetric nonconforming mixed finite elements for linear elasticity, SIAM Journal on Numerical Analysis 49 (4) (2011) 1504–1520. `doi:10.1137/10080018X`.

[58] T. Wihler, Locking-free DGFEM for elasticity problems in polygons, IMA Journal of Numerical Analysis 24 (1) (2004) 45–75. `doi:10.1093/imanum/24.1.45`.

[59] T. Wihler, Locking-free adaptive discontinuous Galerkin FEM for linear elasticity problems, Mathematics of Computation 75 (255) (2006) 1087–1102. `doi:10.1090/S0025-5718-06-01815-1`.

[60] D. Di Pietro, S. Nicaise, A locking-free discontinuous Galerkin method for linear elasticity in locally nearly incompressible heterogeneous media, Applied Numerical Mathematics. An IMACS Journal 63 (2013) 105–116. `doi:10.1016/j.apnum.2012.09.009`.

[61] J. Bramwell, L. Demkowicz, J. Gopalakrishnan, W. Qiu, A locking-free hp DPG method for linear elasticity with symmetric stresses, Numerische Mathematik 122 (4) (2012) 671–707. `doi:10.1007/s00211-012-0476-6`.

[62] B. Cockburn, D. Schötzau, J. Wang, Discontinuous Galerkin methods for incompressible elastic materials, Computer Methods in Applied Mechanics and Engineering 195 (25-28) (2006) 3184–3204. `doi:10.1016/j.cma.2005.07.003`.

[63] S.-C. Soon, B. Cockburn, H. K. Stolarski, A hybridizable discontinuous Galerkin method for linear elasticity, International Journal for Numerical Methods in Engineering 80 (8) (2009) 1058–1092. `doi:10.1002/nme.2646`.

[64] G. Fu, B. Cockburn, H. Stolarski, Analysis of an HDG method for linear elasticity, International Journal for Numerical Methods in Engineering 102 (3-4) (2015) 551–575. `doi:10.1002/nme.4781`.

[65] B. Cockburn, J. Gopalakrishnan, J. Guzmán, A new elasticity element made for enforcing weak stress symmetry, Mathematics of Computation 79 (271) (2010) 1331–1349. `doi:10.1090/S0025-5718-10-02343-4`.

[66] W. Qiu, J. Shen, K. Shi, An HDG method for linear elasticity with strong symmetric stresses, Mathematics of Computation 87 (309) (2018) 69–93. `doi:10.1090/mcom/3249`.

[67] D. Di Pietro, A. Ern, A hybrid high-order locking-free method for linear elasticity on general meshes, Computer Methods in Applied Mechanics and Engineering 283 (2015) 1–21. `doi:10.1016/j.cma.2014.09.009`.

[68] Cockburn, Bernardo, Di Pietro, Daniele A., Ern, Alexandre, Bridging the hybrid high-order and hybridizable discontinuous galerkin methods, ESAIM: Mathematical Modelling and Numerical Analysis 50 (3) (2016) 635–650. `doi:10.1051/m2an/2015051`.

[69] J. Moitinho de Almeida, J. Teixeira de Freitas, Alternative approach to the formulation of hybrid equilibrium finite elements, Computers and Structures 40 (4) (1991) 1043–1047. `doi:10.1016/0045-7949(91)90336-K`.

[70] J. Moitinho de Almeida, O. Almeida Pereira, A set of hybrid equilibrium finite element models for the analysis of three-dimensional solids, International Journal for Numerical Methods in Engineering 39 (16) (1996) 2789–2802. `doi:10.1002/(SICI)1097-0207(19960830)39:16<2789::AID-NME976>3.0.CO;2-J`.

[71] E. Maunder, J. Moitinho de Almeida, O. Almeida Pereira, The stability of stars of simplicial hybrid equilibrium finite elements for solid mechanics, International Journal for Numerical Methods in Engineering 107 (8) (2016) 633–668. `doi:10.1002/nme.5179`.

[72] J. Moitinho de Almeida, E. Maunder, Equilibrium Finite Element Formulations, Wiley, 2017. `doi:10.1002/9781118925782`.

[73] D. N. Arnold, F. Brezzi, Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates, RAIRO Modélisation Mathématique et Analyse Numérique 19 (1) (1985) 7–32. `doi:10.1051/m2an/1985190100071`.

[74] B. Cockburn, K. Shi, Superconvergent HDG methods for linear elasticity with weakly symmetric stresses, IMA Journal of Numerical Analysis 33 (3) (2013) 747–770. `doi:10.1093/imanum/drs020`.

[75] B. Cockburn, G. Fu, Devising superconvergent HDG methods with symmetric approximate stresses for linear elasticity by M-decompositions, IMA Journal of Numerical Analysis 38 (2018) 566–604. `doi:10.1093/imanum/drx025`.

[76] J. S. Hesthaven, T. Warburton, Nodal discontinuous Galerkin methods, Vol. 54 of Texts in Applied Mathematics, Springer, New York, 2008, algorithms, analysis, and applications. `doi:10.1007/978-0-387-72067-8`.

[77] B. Rivière, Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations, Society for Industrial and Applied Mathematics, 2008.

[78] X. Feng, O. Karakashian, Y. Xing (Eds.), Recent developments in discontinuous Galerkin finite element methods for partial differential equations, Vol. 157 of The IMA Volumes in Mathematics and its Applications, Springer, Cham, 2014, 2012 John H. Barrett Memorial Lectures, Selected papers from the workshop held at the University of Tennessee, Knoxville, TN, May 9–11, 2012. `doi:10.1007/978-3-319-01818-8`.

[79] A. Cangiani, Z. Dong, E. H. Georgoulis, P. Houston, hp-Version Discontinuous Galerkin Methods on Polygonal and Polyhedral Meshes, Springer International Publishing, 2017. `doi:10.1007/978-3-319-67673-9`.

[80] A. Cesmelioglu, B. Cockburn, W. Qiu, Analysis of a hybridizable discontinuous Galerkin method for the steady-state incompressible Navier-Stokes equations, Mathematics of Computation 86 (306) (2016) 1643–1670. `doi:10.1090/mcom/3195`.

[81] B. Cockburn, N. C. Nguyen, J. Peraire, A comparison of HDG methods for Stokes flow, Journal of Scientific Computing 45 (1-3) (2010) 215–237. `doi:10.1007/s10915-010-9359-0`.

[82] B. Cockburn, J. Cui, An analysis of HDG methods for the vorticity-velocity-pressure formulation of the Stokes problem in three dimensions, Mathematics of Computation 81 (279) (2012) 1355–1368. `doi:10.1090/S0025-5718-2011-02575-5`.

[83] B. Cockburn, K. Shi, Devising HDG methods for Stokes flow: An overview, Computers & Fluids 98 (2014) 221–229. `doi:10.1016/j.compfluid.2013.11.017`.

[84] J. Carrero, B. Cockburn, D. Schötzau, Hybridized globally divergence-free LDG methods. I. The Stokes problem, Mathematics of Computation 75 (254) (2006) 533–563. `doi:10.1090/S0025-5718-05-01804-1`.

[85] P. Hansbo, M. G. Larson, Piecewise divergence-free discontinuous Galerkin methods for Stokes flow, Communications in Numerical Methods in Engineering with Biomedical Applications 24 (5) (2008) 355–366. `doi:10.1002/cnm.975`.

[86] J. Peraire, P.-O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, SIAM Journal on Scientific Computing 30 (4) (2008) 1806–1824. `doi:10.1137/070685518`.

[87] A. Montlaur, S. Fernández-Méndez, A. Huerta, Discontinuous Galerkin methods for the Stokes equations using divergence-free approximations, International Journal for Numerical Methods in Fluids 57 (9) (2008) 1071–1092. `doi:10.1002/fld.1716`.

[88] A. Montlaur, S. Fernandez-Mendez, J. Peraire, A. Huerta, Discontinuous Galerkin methods for the Navier-Stokes equations using solenoidal approximations, International Journal for Numerical Methods in Fluids 64 (5) (2010) 549–564. `doi:10.1002/fld.2161`.

[89] B. Cockburn, J. Gopalakrishnan, Incompressible finite elements via hybridization. I. The Stokes system in two space dimensions, SIAM Journal on Numerical Analysis 43 (4) (2005) 1627–1650. `doi:10.1137/04061060X`.

[90] B. Cockburn, J. Gopalakrishnan, Incompressible finite elements via hybridization. II. The Stokes system in three space dimensions, SIAM Journal on Numerical Analysis 43 (4) (2005) 1651–1672. `doi:10.1137/040610659`.

[91] B. Cockburn, B. Dong, J. Guzmán, A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems, Mathematics of Computation 77 (2008) 1887–1916. `doi:10.1090/S0025-5718-08-02123-6`.

[92] B. Cockburn, J. Gopalakrishnan, The derivation of hybridizable discontinuous Galerkin methods for Stokes flow, SIAM Journal on Numerical Analysis 47 (2) (2009) 1092–1125. `doi:10.1137/080726653`.

[93] B. Cockburn, J. Gopalakrishnan, N. C. Nguyen, J. Peraire, F.-J. Sayas, Analysis of HDG methods for Stokes flow, Mathematics of Computation 80 (2011) 723–760. `doi:10.1090/S0025-5718-2010-02410-X`.

[94] B. Cockburn, K. Shi, Conditions for superconvergence of HDG methods for Stokes flow, Mathematics of Computation 82 (282) (2013) 651–671. `doi:10.1090/S0025-5718-2012-02644-5`.

[95] G. Giorgiani, S. Fernández-Méndez, A. Huerta, Hybridizable discontinuous Galerkin with degree adaptivity for the incompressible Navier–Stokes equations, Comp. Fluids 98 (2014) 196–208. `doi:10.1016/j.compfluid.2014.01.011`.

[96] R. Sevilla, A. Huerta, HDG-NEFEM with degree adaptivity for stokes flows, Journal of Scientific Computing 77 (3) (2018) 1953–1980. `doi:10.1007/s10915-018-0657-2`.

[97] B. Cockburn, G. Fu, F. J. Sayas, Superconvergence by M-decompositions. Part I: General theory for HDG methods for diffusion, Mathematics of Computation 86 (2017) 1609–1641. `doi:10.1090/mcom/3140`.

[98] B. Cockburn, G. Fu, Superconvergence by M-decompositions. Part II: Construction of two-dimensional finite elements, ESAIM. Mathematical Modelling and Numerical Analysis 51 (1) (2017) 165–186. `doi:10.1051/m2an/2016016`.

[99] B. Cockburn, G. Fu, Superconvergence by M-decompositions. Part III: Construction of three-dimensional finite elements, ESAIM. Mathematical Modelling and Numerical Analysis 51 (1) (2017) 365–398. `doi:10.1051/m2an/2016023`.

[100] B. Cockburn, G. Fu, W. Qiu, A note on the devising of superconvergent HDG methods for Stokes flow by M-decompositions, IMA Journal of Numerical Analysis 37 (2) (2017) 730–749. `doi:10.1093/imanum/drw029`.

[101] I. Oikawa, Analysis of a reduced-order HDG method for the Stokes equations, Journal of Scientific Computing 67 (2) (2016) 475–492. `doi:10.1007/s10915-015-0090-8`.

[102] W. Qiu, K. Shi, A superconvergent HDG method for the incompressible Navier-Stokes equations on general polyhedral meshes, IMA J. Numer. Anal. 36 (4) (2016) 1943–1967. `doi:10.1093/imanum/drv067`.

[103] C. Lehrenfeld, J. Schöberl, High order exactly divergence-free hybrid discontinuous Galerkin methods for unsteady incompressible flows, Computer Methods in Applied Mechanics and Engineering 307 (2016) 339–361. `doi:10.1016/j.cma.2016.04.025`.

[104] Q. Zhai, R. Zhang, X. Wang, A hybridized weak galerkin finite element scheme for the stokes equations, Science China Mathematics 58 (11) (2015) 2455–2472. `doi:10.1007/s11425-015-5030-4`.

[105] J. Wu, J. Zhu, J. Szmelter, O. Zienkiewicz, Error estimation and adaptivity in Navier-Stokes incompressible flows, Computational mechanics 6 (4) (1990) 259–270. `doi:10.1007/BF00370106`.

[106] K. G. Powell, P. L. Roe, J. Quirk, Adaptive-mesh algorithms for computational fluid dynamics, in: Algorithmic trends in computational fluid dynamics, Springer, 1993, pp. 303–337. `doi:10.1007/978-1-4612-2708-3_18`.

[107] R. Löhner, K. Morgan, O. Zienkiewicz, An adaptive finite element procedure for compressible high speed flows, Computer Methods in Applied Mechanics and Engineering 51 (1-3) (1985) 441–465. `doi:10.1016/0045-7825(85)90042-8`.

[108] J.-F. Remacle, J. E. Flaherty, M. S. Shephard, An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems, SIAM review 45 (1) (2003) 53–72. doi:10.1137/S00361445023830.

[109] S.-E. Kim, F. Boysan, Application of cfd to environmental flows, Journal of Wind Engineering and Industrial Aerodynamics 81 (1) (1999) 145 – 158. doi:10.1016/S0167-6105(99)00013-6.

[110] R. Löhner, An adaptive finite element scheme for transient problems in CFD, Computer Methods in Applied Mechanics and Engineering 61 (3) (1987) 323–338. doi:10.1016/0045-7825(87)90098-3.

[111] K. J. Fidkowski, D. L. Darmofal, Review of output-based error estimation and mesh adaptation in computational fluid dynamics, AIAA journal 49 (4) (2011) 673–694. doi:10.2514/1.J050073.

[112] P. Díez, A. Huerta, A unified approach to remeshing strategies for finite element h-adaptivity, Computer Methods in Applied Mechanics and Engineering 176 (1-4) (1999) 215–229. doi:10.1016/S0045-7825(98)00338-7.

[113] R. Löhner, J. D. Baum, Adaptive h-refinement on 3D unstructured grids for transient problems, International Journal for Numerical Methods in Fluids 14 (12) (1992) 1407–1419. doi:https://doi.org/10.1002/fld.1650141204.

[114] G. Giorgiani, S. Fernández-Méndez, A. Huerta, Hybridizable discontinuous Galerkin p-adaptivity for wave propagation problems, International Journal for Numerical Methods in Fluids 72 (12) (2013) 1244–1262. doi:10.1002/fld.3784.

[115] T. Tang, Moving mesh methods for computational fluid dynamics, Contemporary mathematics 383 (8) (2005) 141–173. doi:10.0.4.66/conm/383/07162.

[116] C. J. Budd, W. Huang, R. D. Russell, Adaptivity with moving grids, Acta Numerica 18 (2009) 111–241. doi:10.1017/S0962492906400015.

[117] P. Šolín, J. Červený, I. Doležel, Arbitrary-level hanging nodes and automatic adaptivity in the hp-FEM, Mathematics and Computers in Simulation 77 (1) (2008) 117 – 132. doi:10.1016/j.matcom.2007.02.011.

[118] W. Rachowicz, D. Pardo, L. Demkowicz, Fully automatic hp-adaptivity in three dimensions, Computer Methods in Applied Mechanics and Engineering 195 (37) (2006) 4816 – 4842. doi:10.1016/j.cma.2005.08.022.

[119] K. S. Bey, J. T. Oden, hp-version discontinuous Galerkin methods for hyperbolic conservation laws, Computer Methods in Applied Mechanics and Engineering 133 (3-4) (1996) 259–286. doi:10.1016/0045-7825(95)00944-2.

[120] N. Burgess, D. Mavriplis, An hp-adaptive discontinuous Galerkin solver for aero-dynamic flows on mixed-element meshes, in: 49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2011. doi:10.2514/6.2011-490.

[121] M. Woopen, A. Balan, G. May, J. Schütz, A comparison of hybridized and stan-dard DG methods for target-based hp-adaptive simulation of compressible flow, Computers & Fluids 98 (2014) 3–16. doi:10.1016/j.compfluid.2014.03.023.

[122] S. Rhebergen, B. Cockburn, A space–time hybridizable discontinuous Galerkin method for incompressible flows on deforming domains, Journal of Computational Physics 231 (11) (2012) 4185–4204. doi:10.1016/j.jcp.2012.02.011.

[123] M. Ueckermann, P. F. Lermusiaux, Hybridizable discontinuous Galerkin projec-tion methods for Navier–Stokes and Boussinesq equations, Journal of Computa-tional Physics 306 (2016) 390–421. doi:10.1016/j.jcp.2015.11.028.

[124] S. Rhebergen, G. N. Wells, A hybridizable discontinuous Galerkin method for the Navier–Stokes equations with pointwise divergence-free velocity field, Journal of Scientific Computing 76 (3) (2018) 1484–1501. doi:10.1007/s10915-018-0671-4.

[125] J. M. Maljaars, R. J. Labeur, M. Möller, A hybridized discontinuous Galerkin framework for high-order particle–mesh operator splitting of the incompressible Navier–Stokes equations, Journal of Computational Physics 358 (2018) 150–172. doi:10.1016/j.jcp.2017.12.036.

[126] F. Alauzet, P. Frey, P. George, B. Mohammadi, 3D transient fixed point mesh adaptation for time-dependent problems: Application to CFD simulations, Jour-nal of Computational Physics 222 (2) (2007) 592 – 623. doi:10.1016/j.jcp.2006.08.012.

[127] K.-J. Bathe, H. Zhang, A mesh adaptivity procedure for CFD and fluid-structure interactions, Computers & Structures 87 (11-12) (2009) 604–617. doi:j.compstruc.2009.01.017.

[128] E. Hairer, G. Wanner, Solving ordinary differential equations ii: Stiff and differential-algebraic problems second revised edition with 137 figures, Vol. 14, Springer-Verlag, 1996.

[129] E. Hairer, G. Wanner, Stiff differential equations solved by Radau methods, Journal of Computational and Applied Mathematics 111 (1-2) (1999) 93–111. doi:10.1016/S0377-0427(99)00134-X.

[130] J. Rang, Adaptive Timestep Control for Fully Implicit Runge-Kutta Methods of Higher Order, Univ.-Bibl., 2014.
URL http://www.digibib.tu-bs.de/?docid=00055783

[131] K. Kožulović, G. Ashcroft, Temporal error estimation and adaptive time step control in unsteady flow simulations, in: Proceedings of the 6th European Conference on Computational Fluid Dynamics, 20-25 July 2014; Barcelona, Spain.

[132] A. Jaust, J. Schütz, A temporally adaptive hybridized discontinuous Galerkin method for time-dependent compressible flows, Computers & Fluids 98 (2014) 177–185. doi:10.1016/j.compfluid.2014.01.019.

[133] A. Valli, G. Carey, A. Coutinho, Control strategies for timestep selection in simulation of coupled viscous flow and heat transfer, Communications in Numerical Methods in Engineering 18 (2) (2002) 131–139. doi:10.1002/cnm.475.

[134] W. Hackbusch, Multi-grid methods and applications, Vol. 4, Springer Science & Business Media, 2013. doi:10.1007/978-3-662-02427-0.

[135] J. Gopalakrishnan, S. Tan, A convergent multigrid cycle for the hybridized mixed method, Numerical Linear Algebra with Applications 16 (9) (2009) 689–714. doi:10.1002/nla.636.

[136] H. Luo, J. D. Baum, R. Löhner, A p-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids, Journal of Computational Physics 211 (2) (2006) 767–783. doi:10.1016/J.JCP.2005.06.019.

[137] B. Cockburn, O. Dubois, J. Gopalakrishnan, S. Tan, Multigrid for an HDG method, IMA Journal of Numerical Analysis 34 (4) (2014) 1386–1425. doi:10.1093/imanum/drt024.

[138] M. Kronbichler, W. A. Wall, A performance comparison of continuous and discontinuous Galerkin methods with fast multigrid solvers, SIAM Journal on Scientific Computing 40 (5) (2018) A3423–A3448. doi:10.1137/16M110455X.

[139] M. Franciolini, K. Fidkowski, A. Crivellini, P-multigrid preconditioners applied to high-order DG and HDG discretizations, in: Proceedings of the 7th European Conference on Computational Fluid Dynamics: 2018 Jun 11-15; Glasgow, UK, Vol. 50, 2018, pp. 2–5.

[140] S. Yakovlev, D. Moxey, R. M. Kirby, S. J. Sherwin, To CG or to HDG: A Comparative Study in 3D, Journal of Scientific Computing 67 (1) (2016) 192–220. doi:10.1007/s10915-015-0076-6.

[141] S. Rhebergen, G. N. Wells, Preconditioning of a Hybridized Discontinuous Galerkin Finite Element Method for the Stokes Equations, Journal of Scientific Computing 77 (3) (2018) 1936–1952. doi:10.1007/s10915-018-0760-4.

[142] S. Muralikrishnan, M.-B. Tran, T. Bui-Thanh, iHDG: An Iterative HDG Framework for Partial Differential Equations, SIAM Journal on Scientific Computing 39 (5) (2017) S782–S808. `doi:10.1137/16M1074187`.

[143] S. Muralikrishnan, M.-B. Tran, T. Bui-Thanh, An improved iterative HDG approach for partial differential equations, Journal of Computational Physics 367 (2018) 295–321. `doi:10.1016/J.JCP.2018.04.033`.

[144] M. J. Gander, S. Hajian, Analysis of Schwarz Methods for a Hybridizable Discontinuous Galerkin Discretization, SIAM Journal on Numerical Analysis 53 (1) (2015) 573–597. `doi:10.1137/140961857`.

[145] J. Schütz, V. Aizinger, A hierarchical scale separation approach for the hybridized discontinuous Galerkin method, Journal of Computational and Applied Mathematics 317 (2017) 500–509. `doi:10.1016/j.cam.2016.12.018`.

[146] R. Sevilla, M. Giacomini, A. Karkoulias, A. Huerta, A superconvergent hybridisable discontinuous Galerkin method for linear elasticity, International Journal for Numerical Methods in Engineering 116 (2) (2018) 91–116. `doi:10.1002/nme.5916`.

[147] P. Gould, Introduction to Linear Elasticity, Introduction to Linear Elasticity, Springer, 1993.

[148] J. E. Marsden, T. J. R. Hughes, Mathematical foundations of elasticity, Dover Publications, Inc., New York, 1994, corrected reprint of the 1983 original.

[149] P. G. Ciarlet, Mathematical elasticity. Vol. I, Vol. 20 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, 1988, three-dimensional elasticity.

[150] J. Fish, T. Belytschko, A First Course in Finite Elements, John Wiley & Sons, 2007.

[151] N. C. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations, Journal of Computational Physics 228 (23) (2009) 8841–8855. `doi:10.1016/j.jcp.2009.08.030`.

[152] R. Sevilla, A. Huerta, Tutorial on Hybridizable Discontinuous Galerkin (HDG) for second-order elliptic problems, in: J. Schröder, P. Wriggers (Eds.), Advanced Finite Element Technologies, Vol. 566 of CISM International Centre for Mechanical Sciences, Springer International Publishing, 2016, pp. 105–129. `doi:10.1007/978-3-319-31925-4_5`.

[153] B. Cockburn, J. Gopalakrishnan, F.-J. Sayas, A projection-based error analysis of HDG methods, Mathematics of Computation 79 (271) (2010) 1351–1367. `doi:10.1090/S0025-5718-10-02334-3`.

[154] M. Giacomini, A. Karkoulias, R. Sevilla, A. Huerta, A superconvergent HDG method for Stokes flow with strongly enforced symmetry of the stress tensor, Journal of Scientific Computing 77 (3) (2018) 1679–1702. `doi:10.1007/s10915-018-0855-y`.

[155] R. Sevilla, A. Huerta, HDG-NEFEM with degree adaptivity for Stokes flows, J. Sci. Comput. 77 (2018) 1953–1980. `doi:10.1007/s10915-018-0657-2`.

[156] J. Donea, A. Huerta, Finite Element Methods for Flow Problems, Finite Element Methods for Flow Problems, John Wiley & Sons, 2003.
URL `https://books.google.es/books?id=S4URqrTtSXoC`

[157] B. Cockburn, K. Shi, Devising HDG methods for Stokes flow: an overview, Computers & Fluids 98 (2014) 221–229. `doi:10.1016/j.compfluid.2013.11.017`.

[158] R. Stenberg, Some new families of finite elements for the Stokes equations, Numerische Mathematik 56 (8) (1989) 827–838. `doi:10.1007/BF01405291`.

[159] P. G. Ciarlet, The finite element method for elliptic problems, Vol. 40 of Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002, reprint of the 1978 original [North-Holland, Amsterdam].

[160] A. Ern, J.-L. Guermond, Theory and practice of finite elements, Vol. 159 of Applied Mathematical Sciences, Springer-Verlag, New York, 2004. `doi:10.1007/978-1-4757-4355-5`.

[161] C. R. Ethier, D. A. Steinman, Exact fully 3D Navier–Stokes solutions for benchmarking, International Journal for Numerical Methods in Fluids 19 (5) (1994) 369–375. `doi:10.1002/fld.1650190502`.

[162] R. Poya, R. Sevilla, A. J. Gil, A unified approach for a posteriori high-order curved mesh generation using solid mechanics, Computational Mechanics 58 (2016) 457–490. `doi:10.1007/s00466-016-1302-2`.

[163] Z. Q. Xie, R. Sevilla, O. Hassan, K. Morgan, The generation of arbitrary order curved meshes for 3D finite element analysis, Computational Mechanics 51 (2013) 361–374. `doi:10.1007/s00466-012-0736-4`.

[164] G. K. Batchelor, An introduction to fluid dynamics, Cambridge University Press, 2000.

[165] R. Sevilla, O. Hassan, K. Morgan, An analysis of the performance of a high-order stabilised finite element method for simulating compressible flows, Computer Methods in Applied Mechanics and Engineering 253 (2013) 15 – 27. doi:10.1016/j.cma.2012.09.001.

[166] L. Kovasznay, Laminar flow behind a two-dimensional grid, in: Mathematical Proceedings of the Cambridge Philosophical Society, Vol. 44, Cambridge University Press, 1948, pp. 58–62. doi:10.1017/S0305004100023999.

[167] M. Schäfer, S. Turek, F. Durst, E. Krause, R. Rannacher, Benchmark Computations of Laminar Flow Around a Cylinder, in: E. Hirschel (Ed.), Flow Simulation with High-Performance Computers II. Notes on Numerical Fluid Mechanics (NNFM), Vol. 48, Vieweg+Teubner Verlag, 1996, pp. 547–566. doi:10.1007/978-3-322-89849-4_39.

[168] P. Gravesen, J. Branebjerg, O. S. Jensen, Microfluidics-a review, Journal of Micromechanics and Microengineering 3 (4) (1993) 168–182. doi:10.1088/0960-1317/3/4/002.

[169] S. Shoji, M. Esashi, Microflow devices and systems, Journal of Micromechanics and Microengineering 4 (4) (1994) 157–171. doi:10.1088/0960-1317/4/4/001.

[170] K. W. Oh, C. H. Ahn, A review of microvalves, Journal of Micromechanics and Microengineering 16 (5) (2006) R13–R39. doi:10.1088/0960-1317/16/5/R01.

[171] N. Tesla, Valvular conduit, US Patent 1,329,559 (Feb. 3 1920).

[172] F. K. Forster, R. L. Bardell, M. A. Afromowitz, N. R. Sharma, A. Blanchard, Design, fabrication and testing of fixed-valve micro-pumps, in: Proceedings of the ASME Fluids Engineering Division, Vol. 234, ASME, 1995, pp. 39–44.

[173] T. Truong, N. Nguyen, Simulation and Optimization of Tesla Valves, in: Technical Proceedings of the 2003 Nanotechnology Conference and Trade Show, Vol. 1, 2003, pp. 178–181.

[174] S. M. Thompson, B. J. Paudel, T. Jamal, D. K. Walters, Numerical Investigation of Multistaged Tesla Valves, Journal of Fluids Engineering 136 (8) (2014) 081102. doi:10.1115/1.4026620.

[175] U. M. Ascher, Numerical Methods for Evolutionary Differential Equations, Vol. 5, Society for Industrial and Applied Mathematics, 2008. doi:10.1137/1.9780898718911.

[176] C. A. Kennedy, M. H. Carpenter, Diagonally Implicit Runge-Kutta Methods for Ordinary Differential Equations. A Review, Tech. rep., NASA Langley Research Center, Hampton, VA, United States (mar 2016).
URL https://ntrs.nasa.gov/search.jsp?R=20160005923

[177] A. Kværnø, Singly diagonally implicit Runge–Kutta methods with an explicit first stage, BIT Numerical Mathematics 44 (3) (2004) 489–502. doi:10.1023/B:BITN.0000046811.70614.38.

[178] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical recipes 3rd edition: The art of scientific computing, Cambridge university press, 2007.

[179] C. A. Kennedy, M. H. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, Applied Numerical Mathematics 44 (1-2) (2003) 139–181. doi:10.1016/S0168-9274(02)00138-1.

[180] P. R. Amestoy, I. S. Duff, J. Koster, J.-Y. L'Excellent, A fully asynchronous multifrontal solver using distributed dynamic scheduling, SIAM Journal on Matrix Analysis and Applications 23 (1) (2001) 15–41. doi:10.1137/S0895479899358194.

[181] P. R. Amestoy, A. Guermouche, J.-Y. L'Excellent, S. Pralet, Hybrid scheduling for the parallel solution of linear systems, Parallel Computing 32 (2) (2006) 136–156. doi:10.1016/j.parco.2005.07.004.

[182] O. Schenk, K. Gärtner, W. Fichtner, A. Stricker, PARDISO: A high-performance serial and parallel sparse linear solver in semiconductor device simulation, Future Generation Computer Systems 18 (1) (2001) 69–78. doi:10.1016/S0167-739X(00)00076-5.

[183] T. A. Davis, UMFPACK User Guide, VERSION 5.5.1 (2011 (accessed April 10, 2019)).
URL https://users.encs.concordia.ca/~krzyzak/R%20Code-Communications%20in%20Statistics%20and%20Simulation%202014/Zubeh%F6r/SuiteSparse/UMFPACK/Doc/UserGuide.pdf

[184] H. C. Elman, D. J. Silvester, A. J. Wathen, Performance and analysis of saddle point preconditioners for the discrete steady-state Navier-Stokes equations, Numerische Mathematik 90 (4) (2002) 665–688. doi:10.1007/s002110100300.

[185] H. C. Elman, D. J. Silvester, A. J. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, Oxford University Press, USA, 2014.

[186] M. Benzi, G. H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta numerica 14 (2005) 1–137. doi:10.1017/S0962492904000212.

[187] T. Heister, G. Rapin, Efficient augmented Lagrangian-type preconditioning for the Oseen problem using Grad–Div stabilization, International Journal for Numerical Methods in Fluids 71 (1) (2013) 118–134. doi:10.1002/fld.3654.

[188] L. Rebholz, A. Viguerie, M. Xiao, Efficient nonlinear iteration schemes based on algebraic splitting for the incompressible Navier-Stokes equations, Mathematics of Computation 88 (318) (2019) 1533–1557. doi:10.1090/mcom/3411.

[189] T. A. Davis, Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method, ACM Transactions on Mathematical Software 30 (2) (2004) 196–199. doi:10.1145/992200.992206.

[190] Y. Saad, M. H. Schultz, GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 856–869. doi:10.1137/0907058.

[191] A. Melendo, A. Coll, M. Pasenau, E. Escolano, A. Monros, www.gidhome.com, [Online; accessed Jun-2016] (2016).
URL https://www.gidhome.com

[192] MATLAB, version 9.3.0.713579 (R2017b), The Mathworks, Inc., Natick, Massachusetts, 2017.

[193] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering 79 (11) (2009) 1309–1331. doi:10.1002/nme.2579.

[194] Aneurisk-Team, AneuriskWeb project website, http://ecm2.mathcs.emory.edu/aneuriskweb, Web Site (2012).
URL http://ecm2.mathcs.emory.edu/aneuriskweb

[195] J. R. Cebral, M. A. Castro, C. M. Putman, N. Alperin, Flow-area relationship in internal carotid and vertebral arteries., Physiological measurement 29 (5) (2008) 585–94. doi:10.1088/0967-3334/29/5/005.

[196] M. Shojima, M. Oshima, K. Takagi, R. Torii, M. Hayakawa, K. Katada, A. Morita, T. Kirino, Magnitude and role of wall shear stress on cerebral aneurysm: computational fluid dynamic study of 20 middle cerebral artery aneurysms, Stroke 35 (11) (2004) 2500–2505. doi:10.1161/01.STR.0000144648.89172.0f.

[197] F. Hecht, New development in freefem++, Journal of numerical mathematics 20 (3-4) (2012) 251–266. doi:10.1515/jnum-2012-0013.