



**EFFICIENT DISCRETISATION AND
DOMAIN DECOMPOSITION
PRECONDITIONERS FOR
INCOMPRESSIBLE FLUID MECHANICS**

BY

Michał Bosy

Supervised by

Victorita Dolean

Gabriel R. Barrenechea

*A thesis submitted to the
Department of Mathematics and Statistics
University of Strathclyde
in partial fulfilment of the require for the degree of*

Doctor of Philosophy

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

Signed:

Date:

Abstract

Solving the linear elasticity and Stokes equations by an optimal domain decomposition method derived algebraically involves the use of non standard interface conditions whose discretisation is not trivial. For this reason the use of approximation methods such as hybrid discontinuous Galerkin appears as an appropriate strategy: on the one hand they provide the best compromise in terms of the number of degrees of freedom in between standard continuous and discontinuous Galerkin methods, and on the other hand the degrees of freedom used in the non standard interface conditions are naturally defined at the boundary between elements. In this manuscript we present the coupling between a well chosen discretisation method (hybrid discontinuous Galerkin) and a novel and efficient domain decomposition method to solve the Stokes system. An analysis of the boundary value problem with non standard condition is provided as well as the numerical evidence showing the advantages of the new method. Furthermore, we present and analyse a stabilisation method for the presented discretisation that allows the use of the same polynomial degrees for velocity and pressure discrete spaces. The original definition of the domain decomposition preconditioners is one-level, this is, the preconditioner is built only using the solution of local problems. This has the undesired consequence that the results are not scalable, it means that the number of iterations needed to reach convergence increases with the number of subdomains. This is the reason why we have also introduced, and tested numerically, two-level preconditioners. Such preconditioners use a coarse space in their construction. We consider two finite element discretisations, namely, the hybrid discontinuous Galerkin and Taylor-Hood discretisations for the nearly incompressible elasticity problems and Stokes equations.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisors Dr Victorita Dolean and Dr Gabriel Barrenechea for the continuous support of my Ph.D study and related research, for their patience, motivation, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis.

Secondly, I am grateful to Prof. Frédéric Nataf, who provided me with an opportunity to join his team, as well as to Dr Pierre-Henri Tournier who shared with me his experience and long days at Laboratoire Jacques-Louis Lions in Paris. I am thankful to both for their useful comments and suggestions.

I extend my gratitude to the Centre for Numerical Analysis and Intelligent Software (NAIS) that supported this work. I would also like to acknowledge my thesis committee: Dr Sébastien Loisel and Dr John MacKenzie, for their insightful comments and encouragement, but also for the challenging questions which allowed me to look on my research from various perspectives. In addition, I am thankful to friends and staff members from the Department of Mathematics and Statistics at University of Strathclyde for their help and advice.

Last but clearly not the least, I would like to thank my parents, Beata and Jarosław, for supporting me spiritually throughout writing this thesis and my life in general. I would like also to express my gratitude to Teresa, my girlfriend, for her continued support and encouragement. This thesis would just not have been possible without them.

Contents

Introduction	xiii
Motivation	xiii
Content	xvi
1 Hybrid discontinuous Galerkin methods	1
1.1 State of the art	1
1.2 Notation	3
1.3 Stokes equations	4
1.4 Inverse and trace inequalities	5
1.5 Approximation results	6
1.6 TVNF boundary conditions	8
1.6.1 The discrete problem	8
1.6.2 Well-posedness of the discrete problem	11
1.6.3 Error analysis	17
1.6.4 Convergence validation	19
1.7 NVTF boundary conditions	25
1.7.1 The discrete problem	25
1.7.2 Well-posedness of the discrete problem	27
1.7.3 Error analysis	29
1.7.4 Convergence validation	29
1.8 Summary	32

2	Stabilised hybrid discontinuous Galerkin methods	34
2.1	State of the art	34
2.2	TVNF boundary conditions	36
2.2.1	The discrete problem	36
2.2.2	Well-posedness of the discrete problem	37
2.2.3	Error analysis	42
2.2.4	Convergence validation	43
2.3	NVTF boundary conditions	46
2.3.1	The discrete problem	47
2.3.2	Well-posedness of the discrete problem	47
2.3.3	Error analysis	49
2.3.4	Convergence validation	50
2.4	Summary	53
3	Domain decomposition methods	54
3.1	State of the art	54
3.2	Overlapping Schwarz methods	56
3.3	The domain decomposition preconditioner	58
3.4	Partition of unity	59
3.5	Preconditioners comparison	60
3.6	Summary	68
4	Coarse spaces	70
4.1	State of the art	70
4.2	Symmetrised Modified Restricted Additive Schwarz	72
4.3	Two-level methods	72
4.4	Numerical results	74
4.4.1	Stokes equation	74
4.4.2	Nearly incompressible elasticity	81
4.5	Summary	89

<i>CONTENTS</i>	vii
Conclusions	91
Summary	91
Future work	92
A Matlab implementations	94
A.1 One dimensional hdG method for elliptic problem	94
A.2 Two dimensional hdG method for elliptic problem	98
B FreeFem++ implementations	103
B.1 Taylor-Hood method for Stokes problem	103
B.2 hdG method for Stokes problem	105

List of Figures

1	Comparison of degrees of freedom for polynomial degree one for three kinds of finite element methods	xiv
1.1	Analytic solution - Example 1	20
1.2	Error convergence of the hdG method with TVNF boundary conditions - Example 1	21
1.3	Analytic solution - Example 2	21
1.4	Error convergence of the hdG method with TVNF boundary conditions - Example 2	22
1.5	Analytic solution - Example 3	22
1.6	Error convergence of the hdG method with TVNF boundary conditions - Example 3	23
1.7	hdG solution with TVNF boundary conditions - Example 4	23
1.8	Mesh - Example 4	24
1.9	Error convergence of the hdG method with TVNF boundary conditions - Example 4	24
1.10	Error convergence of the hdG method with NVTF boundary conditions - Example 1	30
1.11	Error convergence of the hdG method with NVTF boundary conditions - Example 2	31
1.12	Error convergence of the hdG method with NVTF boundary conditions - Example 3	31
1.13	hdG solution with NVTF boundary conditions - Example 4	32
1.14	Error convergence of the hdG method with NVTF boundary conditions - Example 4	33

2.1	Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 1	44
2.2	Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 2	44
2.3	Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 3	45
2.4	hdG solution with TVNF boundary conditions - Example 4	46
2.5	Error convergence of the hdG method with TVNF boundary conditions - Exam- ple 4	46
2.6	Error convergence of the stabilised hdG method with NVTF boundary conditions - Example 1	50
2.7	Error convergence of the stabilised hdG method with NVTF boundary conditions - Example 2	51
2.8	Error convergence of the stabilised hdG method with NVTF boundary conditions - Example 3	52
2.9	hdG solution with NVTF boundary conditions - Example 4	52
2.10	Error convergence of the hdG method with NVTF boundary conditions - Exam- ple 4	53
3.1	Original geometry of domain	56
3.2	Partition of unity	57
3.3	Convergence of error for uniform decomposition into the 8×8 subdomains - Example 2	61
3.4	Convergence of error for METIS decomposition into the 64 subdomains - Example 2	62
3.5	Convergence of error for uniform decomposition into the 12×12 subdomains - Example 3	63
3.6	Convergence of error for METIS decomposition into the 144 subdomains - Ex- ample 3	63
3.7	Numerical solution of the driven cavity problem - Example 5	64
3.8	Convergence of error for uniform decomposition into the 14×14 subdomains - Example 5	65
3.9	Convergence of error for METIS decomposition into the 196 subdomains - Ex- ample 5	65

3.10	Convergence of error for uniform decomposition into the 16×16 subdomains - Example 1	66
3.11	Convergence of error for METIS decomposition into the 256 subdomains - Example 1	67
3.12	Numerical solution - Example 4	67
3.13	Convergence of error for METIS decomposition into the 800 subdomains - Example 4	68
4.1	Increase of size of the plateau region - Example 2	71
4.2	Eigenvalues on one of the floating subdomains in the case of uniform decomposition and hdG discretisation - Example 5	75
4.3	Eigenvalues on one of the floating subdomains in the case of METIS decomposition and hdG discretisation - Example 4	77
4.4	Eigenvalues on one of the floating subdomains in the case of uniform decomposition and Taylor-Hood discretisation - Example 5	80
4.5	Eigenvalues on one of the floating subdomains in the case of METIS decomposition and Taylor-Hood discretisation - Example 4	80
4.6	Numerical solution of the problem - Example 6	83
4.7	Eigenvalues on one of the floating subdomains in the case of uniform decomposition and hdG discretisation - Example 6	84
4.8	Heterogeneous beam	85
4.9	Eigenvalues on one of the floating subdomains in the case of METIS decomposition and hdG discretisation - Example 7	85
4.10	Eigenvalues on one of the floating subdomains in the case of uniform decomposition and Taylor-Hood discretisation - Example 6	88
4.11	Eigenvalues on one of the floating subdomains in the case of METIS decomposition and Taylor-Hood discretisation - Example 7	88
A.1	One dimensional Matlab implementation - error convergence of the hdG method with strongly imposed boundary conditions	97
A.2	One dimensional Matlab implementation - error convergence of the hdG method with boundary conditions imposed by penalisation	98
A.3	One dimensional Matlab implementation - error convergence of the dG method	98

A.4	Two dimensional Matlab implementation - error convergence of the hdG method with strongly imposed boundary conditions	102
A.5	Two dimensional Matlab implementation - error convergence of the hdG method with boundary conditions imposed by penalisation	102
B.1	Example of the mesh of the unit square	104
B.2	Taylor-Hood solution	105
B.3	Hybrid discontinuous Galerkin solution	108

List of Tables

3.1	Preconditioners comparison - Example 2	61
3.2	Preconditioners comparison - Example 3	62
3.3	Preconditioners comparison - Example 5	64
3.4	Preconditioners comparison for stabilised hdG - Example 5	65
3.5	Preconditioners comparison - Example 1	66
3.6	Preconditioners comparison - Example 4	68
4.1	Comparison of preconditioners for hdG discretisation - Example 5	76
4.2	Comparison of preconditioners for stabilised hdG discretisation - Example 5	77
4.3	Comparison of preconditioners for hdG discretisation - Example 4	78
4.4	Comparison of preconditioners for Taylor-Hood discretisation - Example 5	79
4.5	Comparison of preconditioners for Taylor-Hood discretisation - Example 4	81
4.6	Comparison of preconditioners for hdG discretisation - Example 6	84
4.7	Comparison of preconditioners for hdG discretisation - Example 7	86
4.8	Comparison of preconditioners for Taylor-Hood discretisation - Example 6	87
4.9	Comparison of preconditioners for Taylor-Hood discretisation - Example 7	89

Introduction

Motivation

Viscous fluid flows (e.g. water flow) are modeled mathematically by the Navier-Stokes equations. When the density is constant, that is the fluid is incompressible, the underlying partial differential equations defined on the domain Ω become

$$(1) \quad \begin{cases} \frac{\partial \mathbf{u}}{\partial t} + -\nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases},$$

where \mathbf{u} is the flow velocity, p the pressure, t the time, ν the viscosity and \mathbf{f} is an external body force.

The various applications were the motivation to develop numerical methods and solution strategies for this kind of problems over the past decades. To solve an unsteady problem it is common to use a time discretisation. Although, we focus only on a steady-state version of the Navier-Stokes equations. Since equation (1) is non-linear, its solution requires linearisation. One possibility is to use Newton's method for which, at each step, a linear problem (called the Oseen problem) of the following type

$$(2) \quad \begin{cases} -\nu \Delta \mathbf{u} + (\mathbf{b} \cdot \nabla) \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases},$$

where \mathbf{b} is the convective velocity, needs to be solved.

When the Reynolds number is small, that is when the convection is negligible with respect to the diffusion, we can further simplify (2) and get Stokes equation

$$(3) \quad \begin{cases} -\nu \Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases}.$$

As a general rule, studying Stokes and Oseen equations is the first step before dealing with the difficulties of the numerical simulation of the Navier-Stokes equations.

We are interested in continuous and discontinuous variational approximations of Stokes equation of finite element type. That means we replace an infinite dimensional problem by its finite dimensional approximation on appropriate spaces. The major difficulty in building this kind of spaces is taking into account the incompressibility condition and the vector nature of the problem. Indeed, it is difficult to find divergence-free finite element spaces. Furthermore, in the Stokes equation the variables associated with velocity and pressure are coupled. Thus, the discrete problem to be solved is a mixed problem of the following type

Find $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times P_h$ such that for all $(\mathbf{v}_h, q_h) \in \mathbf{V}_h \times P_h$

$$\begin{cases} a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, dx \\ b(\mathbf{u}_h, q_h) = 0 \end{cases},$$

where $a : \mathbf{V}_h \times \mathbf{V}_h \rightarrow \mathbb{R}$ is a bilinear form associated only with velocity space and $b : \mathbf{V}_h \times P_h \rightarrow \mathbb{R}$ is associated with velocity and pressure spaces. It implies that different but coupled spaces have to be used for velocity and pressure. In addition, it is important to prove the well-posedness of the discretised problem. For continuous finite element methods, this aspect is well known and the theory is well understood.

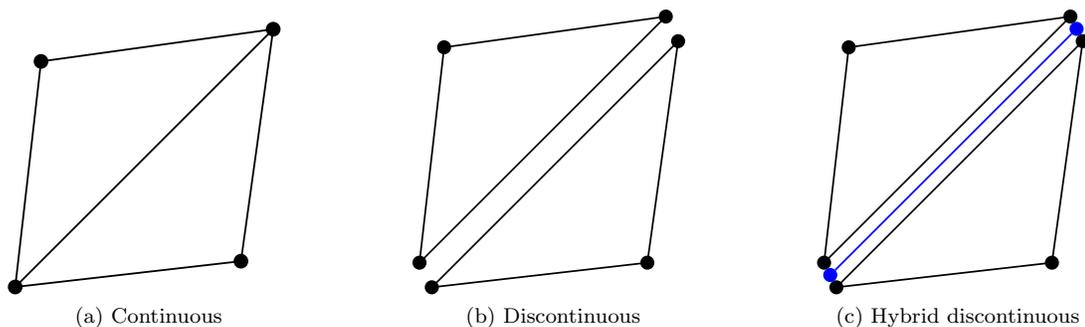


Figure 1: Comparison of degrees of freedom for polynomial degree one for three kinds of finite element methods

In the latest years, a new class of methods has been introduced, namely, discontinuous Galerkin methods. They combine the advantages of finite element methods such as high order approximation or compatibility while allowing more flexibility such as discontinuity of the fields across the mesh interfaces. However, discontinuous Galerkin methods lead to larger linear systems than continuous finite element methods, since degrees of freedom are not shared by neighbouring elements (see Figure 1). A partial solution for this problem is given by the hybrid discontinuous Galerkin method, where partial continuity is enforced across the interfaces between elements by introducing an additional variable associated with edges, as shown in Figure 1. This leads to a potentially lower number of degrees of freedom, since additional continuity is enforced across the edges. Moreover, it maintains a flexible treatment of different fields, since the discrete spaces are still discontinuous. Therefore, hybrid discontinuous Galerkin methods appear as the

best compromise in terms of the number of degrees of freedom between standard continuous and discontinuous Galerkin method.

After discretisation by a discontinuous or continuous finite element method, the underlying linear systems are usually big and difficult to solve. There are two big classes of solvers for linear systems: direct and iterative. The first are very robust and provide the exact solution (up to the precision of the machine) after a finite numbers of steps. Unfortunately they are limited by memory requirements, which make the solution of the linear system beyond a given threshold impossible to obtain in practice. For this reason in case of large linear systems iterative solvers are used. They generate a sequence that approximates the solution of the problem. Hence, the convergence to the appropriate solution depends on the properties of the matrix such as the condition number. The remedy for this is the use of a preconditioner that is instead of solving

$$\mathbf{A}\mathbf{U} = \mathbf{F} \qquad \text{we solve} \qquad \mathbf{M}^{-1}\mathbf{A}\mathbf{U} = \mathbf{M}^{-1}\mathbf{F}.$$

If \mathbf{M}^{-1} is a good approximation of \mathbf{A}^{-1} , then the condition number of $\mathbf{M}^{-1}\mathbf{A}$ is much lower than \mathbf{A} .

In this thesis, we will focus on domain decomposition preconditioners. The main idea is to split the problem defined on the global domain into local problems on smaller subdomains, which can be solved independently, in parallel, and then communicate the results to the other domains. We distinguish two kinds of decompositions into domains: overlapping and non-overlapping. In the overlapping case some subdomains have in common more than just the interface (the artificial boundary created by the decomposition), which can lead to better convergence. On the other side, non-overlapping decomposition allows an easier treatment of problems of different nature (different physical models, different discretisations etc.). In this work we will focus on overlapping domain decomposition preconditioners.

Domain decomposition preconditioners for scalar equations are widely analysed. For the Poisson equation there seem to be consensus of which interface conditions need to be used. These are usually Dirichlet or Neumann type. But for systems of partial differential equations such as elasticity or Stokes problems, it has been derived algebraically, using Smith factorisation, that normal velocity-tangential flux or tangential velocity-normal flux interface conditions should be superior to the pure velocity (Dirichlet like) or pure stress (Neumann like) ones. Thanks to the Smith factorisation, authors of [DNR09] showed the equivalence of the scalar bi-Laplacian operator domain decomposition algorithm with the vectorial one for Stokes system. Hence, our aim is to solve the Stokes equations by an optimal domain decomposition method. By optimal we refer to a method that displays the same properties as the best domain decomposition method designed and applied in the case of a scalar equation.

Due to the difficulty of implementing these non standard interface conditions previous numerical tests were restricted to decompositions where boundaries of subdomains are rectilinear so that the normal to the interface is easy to define. Fortunately, the degrees of freedom used in the

non standard interface conditions are naturally defined at the boundary between elements in case of hybrid discontinuous Galerkin. For this reason and the ones mentioned before, the use of approximation methods such as hybrid discontinuous Galerkin appears as an appropriate strategy.

Content

Since this manuscript is a combination of appropriate hybrid discontinuous Galerkin discretisation and the associated domain decomposition methods using non standard interface conditions, it brings some contributions to the fields of finite element discretisations for Stokes equations and domain decomposition preconditioners. We shortly describe here the main contribution of each of the chapters of the manuscript.

Chapter 1 presents the hybridisation of an interior penalty Galerkin method that allows us to impose the tangential velocity-normal flux and normal velocity-tangential flux boundary conditions in quite a natural way. The formulation is similar to the one from [LS16] with Dirichlet boundary conditions. In addition to different kinds of boundary conditions, we include the projection to reduce the number of degrees of freedom. We prove the existence and uniqueness of the discrete solution in both cases of the non standard boundary conditions. We provide the error estimates and numerical experiments associated with the error convergence of the solution obtained from the hybrid discontinuous Galerkin methods.

Chapter 2 deals with the stabilisation methods presented in [DB04] and [BDG06] that was introduced for continuous finite element methods for Stokes problem with Dirichlet boundary conditions. This approach allows the use of polynomials of equal-order for the pair of velocity and pressure discrete spaces. We apply the same idea to the hybrid discontinuous Galerkin methods for Stokes problem with non standard boundary conditions that we have introduced in the previous chapter. We prove the well-posedness and analysed the error convergence of the stabilised hybrid discontinuous Galerkin method. Furthermore, we test the same examples as in previous chapter.

Chapter 3 discusses domain decomposition preconditioners. To solve the discretised problem we introduced two different kinds of preconditioners with non standard interface conditions. Their optimality has been proved by algebraic techniques in [DNR09]. Unfortunately, numerical experiments that were made previously were restricted to decompositions in rectangular subdomains. Thanks to the hybrid discontinuous Galerkin discretisation, we are able to solve the problem decomposed onto general shaped subdomains, even on the non-structured meshes. We compare the newly introduced preconditioners to the more standard Restricted Additive Schwarz preconditioner in conjunction with different kinds of discretisations (hybrid discontinuous Galerkin and Taylor-Hood).

Chapter 4 continues with domain decomposition preconditioners, but this time we consider two-level preconditioners with the coarse space associated with the general eigenvalue problems. This approach aims at improving the convergence of the Schwarz preconditioners presented in previous chapter. This allows us to obtain the results that are scalable with the respect to the number of subdomains. A similar approach was developed in [HJN15] for the symmetrised variant of preconditioner with Robin interface conditions. We introduce the symmetrised variants of the new preconditioners that were introduced in the previous chapter. Furthermore, we compare preconditioners with non standard interface conditions to the one with the Robin interface conditions.

Appendix A includes implementations of hybrid discontinuous Galerkin methods for a scalar elliptic problem. We consider there examples of one- and two-dimensional problems. The implementation presented in this appendix can be easily written e.g. in Matlab. We test also these examples by plotting the convergence at the end of the sections.

Appendix B is a little introduction to FreeFem++, which is a free software specialised in variational discretisations of partial differential equations. We start with explaining the implementation of well known Taylor-Hood method for Stokes equation in this open source software. Later, we present our implementation of hybrid discontinuous Galerkin methods for the Stokes problem. Both sections end with the plot of the solutions given by each of the discretisations.

Publication Chapters 1 and 3 lead to the following publication

Barrenechea, G. R., Bosy, M., Dolean, V., Nataf, F. and Tournier, P-H.

Hybrid discontinuous Galerkin discretisation and preconditioning of the Stokes problem with non standard boundary conditions

<https://arxiv.org/abs/1610.09207>, submitted, 2016.

Chapter 1

Hybrid discontinuous Galerkin methods

In this chapter, we present the hybrid discontinuous Galerkin discretisation. We start with the state of the art in Section 1.1. In Section 1.2 we introduce the notation used in this manuscript. In Section 1.3 we present the Stokes problem with non standard boundary conditions. The preliminary results necessary for further analysis are the content of Sections 1.4 and 1.5. The analysis of the method is split over two sections. In Section 1.6 we analyse tangential-velocity and normal-flux boundary value problem and in Section 1.7 - normal-velocity and tangential-flux boundary value problem. Each part has the following respectively subsections presenting a discrete problem (Sections 1.6.1 and 1.7.1), analysing the formulation (Sections 1.6.2 and 1.7.2) and the error of the discrete solution (Sections 1.6.3 and 1.7.3), and the last Subsections 1.6.4 and 1.7.4 contain numerical examples. Finally, a summary is outlined in Section 1.8.

1.1 State of the art

Discontinuous Galerkin (dG) methods were first introduced in the early 1970s [RH73] and they have benefited from a wide interest from the scientific community. The main advantages of these methods are their generality and flexibility as they can be used for a large variety of partial differential equations on unstructured meshes. Moreover, they can preserve local properties such as mass and momentum, while ensuring a high order accuracy. However, the cost of these advantages is a large number of degrees of freedom in comparison to the continuous Galerkin methods [EG04] for the same approximation order.

A good compromise between the previous methods, while preserving the high order, are the hybridised versions of dG such as Raviart-Thomas (RT) and Brezzi-Douglas-Marini (BDM) methods [BBF13]. These methods are a subset of the hybrid discontinuous Galerkin (hdG)

methods introduced in [CGL09] for a second-order elliptic problem. This paper contains a detailed comparison, analysis and numerical experiments of a unified hdG, mixed and continuous Galerkin methods for second-order elliptic problems.

An error analysis of hdG methods for second-order elliptic problems is presented in [CGS10]. The new technique introduced there relies on the use of a projection that depends on the choice of numerical trace of the method. In the same time, B. Cockburn provided an overview of mixed hdG methods in [Coc10]. The first a posteriori error analysis of the hdG methods for second-order elliptic problems has been presented in [CNP10]. The conditions for superconvergence with a priori error analysis is shown in [CQS12].

This method was also introduced for convection-diffusion problems in the two papers [NPC09a] and [NPC09b] of N.C. Nguyen, J. Peraire and B. Cockburn which present analysis of mixed formulation for steady and unsteady, linear and non-linear convection-diffusion problems. Meanwhile, H. Egger and J. Schöberl in [ES10] present the whole analysis and a priori error analysis of hybrid-mixed discontinuous Galerkin for convection-diffusion problems. The method is the combination of a mixed method for the elliptic and a discontinuous Galerkin method for the hyperbolic part of the problem.

The hdG methods for the three-dimensional Stokes equation were first introduced in [CG09]. The authors present a mixed formulation of hdG methods defined locally on each element. They consider many types of boundary conditions that involve pressure, normal and tangential velocities, and tangential stress. The formulations of the methods are similar, the only difference is the choice of the numerical traces. A few of these methods have been discussed in [NPC10] and [CNP10]. Both papers not only compare the formulations and the convergence of the methods, but also present some implementation issues.

The hdG methods for the Stokes equation with Dirichlet boundary conditions have been analysed in [CGN⁺11] where the authors show the optimal convergence of the error for hdG methods and present different possibilities to obtain superconvergence. The conditions for the dependence of superconvergence on the approximation projections with a priori error analysis is shown in [CS13]. B. Cockburn and F.-J. Sayas proposed in [CS14a] the divergence-conforming hdG methods. Since it is a modification of previously presented ones, this approach takes advantage of the properties shown before what is discussed in [CS14a]. For the Stokes equation, B. Cockburn and K. Shi also prepared an overview [CS14b] of the above formulations of hdG methods for Stokes problem with Dirichlet boundary conditions.

On the other hand, in [EW13a], a hdG method for two- or three-dimensional Stokes equation with Dirichlet boundary conditions which is hybridisation of a symmetric interior penalty Galerkin method [SST02] is presented and analysed. In a further paper [EW13b] this approach is extended to Darcy, and coupled Darcy-Stokes flows. The new formulation includes different degrees of polynomials for finite element spaces associated with different variables.

In [LS16] the authors consider the Navier-Stokes problem, which can be seen as an extension

of the Stokes and Stokes-Brinkman problems. To obtain the global mixed formulation, the authors choose $H(\text{div})$ -conforming finite elements. Moreover, they introduce the formulation that includes a projection onto a space of lower polynomial degree. Such a modification allowed them to use fewer degrees of freedom. In addition, it helped also to establish a connection between the hybrid high-order [DPEL16] and the hdG methods [Coc16] that presented authors of both methods in their joint paper [CDPE15].

B. Cockburn, N. C. Nguyen and J. Peraire worked also on the hdG methods for Oseen and Navier-Stokes equations. In the joint paper with A. Cesmelioglu [CCNP13] they propose an a priori error analysis of hdG method to numerically solve the Oseen equations. Oseen equations can be interpreted as a linearised version of the incompressible Navier-Stokes equations. Hence, Picard iterations can be used to approximate the Navier-Stokes flow. Such an approach, similar to the one presented by Ch. Lehrenfeld and J. Schöberl in [LS16], is a natural extension of solving convection-diffusion equations and Stokes problems. This extension was shown and analysed earlier in [NPC11].

These authors introduced also some formulations of hdG methods for compressible Euler and Navier-Stokes equation [PNC10]. A wide overview of formulations of the hdG methods for variety continuum mechanics problems are provided in [NP12]. It consists of a large amount of numerical examples to compare the performance of hdG and continuous Galerkin methods.

1.2 Notation

Let Ω be an open polygon domain in \mathbb{R}^2 with Lipschitz boundary $\Gamma := \partial\Omega$. We use **bold and italics** for tensor or vector variables, for example \mathbf{u} is a velocity vector field. The scalar variables will be *italics* for example p denotes pressure scalar value. We define the stress tensor and the flux as $\boldsymbol{\sigma} := \nu \nabla \mathbf{u} - p \mathbf{I}$ and $\boldsymbol{\sigma}_n := \boldsymbol{\sigma} \mathbf{n}$, respectively. In addition we denote normal and tangential components as follows $u_n := \mathbf{u} \cdot \mathbf{n}$, $u_t := \mathbf{u} \cdot \mathbf{t}$, $\sigma_{nn} := \boldsymbol{\sigma}_n \cdot \mathbf{n}$, $\sigma_{nt} := \boldsymbol{\sigma}_n \cdot \mathbf{t}$, where \mathbf{n} is the outward unit normal vector to the boundary Γ and \mathbf{t} is the tangential vector such that $\mathbf{n} \cdot \mathbf{t} = 0$.

For $D \subset \Omega$, we use the standard $L^2(D)$ space with the following norm

$$\|f\|_D^2 := \int_D f^2 \, dx \quad \text{for all } f \in L^2(D).$$

We define the following Sobolev spaces

$$\begin{aligned} H^m(D) &:= \{v \in L^2(D) : \forall |\boldsymbol{\alpha}| \leq m, \partial^{\boldsymbol{\alpha}} v \in L^2(D)\} \text{ for } m \in \mathbb{N}, \\ H(\text{div}, D) &:= \{\mathbf{v} \in [L^2(D)]^2 : \nabla \cdot \mathbf{v} \in L^2(D)\}, \end{aligned}$$

where, for $\boldsymbol{\alpha} = (\alpha_1, \alpha_2) \in \mathbb{N}^2$ and $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2$, we denote $\partial^{\boldsymbol{\alpha}} = \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2}}$. In addition, we

will use the following standard semi-norm and norm for the Sobolev space $H^m(D)$ for $m \in \mathbb{N}$

$$|f|_{H^m(D)}^2 := \sum_{|\alpha|=m} \|\partial^\alpha f\|_D^2 \quad \text{and} \quad \|f\|_{H^m(D)}^2 := \sum_{k=0}^m |f|_{H^k(D)}^2, \quad \text{for all } f \in H^m(D).$$

1.3 Stokes equations

We consider the two dimensional Stokes problem

$$(1.1) \quad \begin{cases} -\nu \Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases},$$

where $\mathbf{u} : \bar{\Omega} \rightarrow \mathbb{R}^2$ is the unknown velocity field, $p : \Omega \rightarrow \mathbb{R}$ the pressure, $\nu > 0$ the viscosity which is considered to be constant and $\mathbf{f} \in [L^2(\Omega)]^2$ is a given function. For $g \in L^2(\Gamma)$ we consider two types of boundary conditions

- tangential-velocity and normal-flux (TVNF) boundary conditions

$$(1.2) \quad \begin{cases} u_t = 0 & \text{on } \Gamma \\ \sigma_{nn} = g & \text{on } \Gamma \end{cases},$$

- normal-velocity and tangential-flux (NVTF) boundary conditions

$$(1.3) \quad \begin{cases} u_n = 0 & \text{on } \Gamma \\ \sigma_{nt} = g & \text{on } \Gamma \end{cases},$$

which, together with (1.1), define two boundary value problems. We consider the formulation of TVNF boundary value problem

$$(1.4) \quad \begin{cases} -\nu \Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \\ \sigma_{nn} = g & \text{on } \Gamma \\ u_t = 0 & \text{on } \Gamma \end{cases},$$

and NVTF boundary value problem

$$(1.5) \quad \begin{cases} -\nu \Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \\ \sigma_{nt} = g & \text{on } \Gamma \\ u_n = 0 & \text{on } \Gamma \end{cases}.$$

The choice of boundary conditions is mainly motivated by domain decomposition algorithms, as was explained in the introduction. We will not address in this manuscript the physical

relevance of these boundary conditions, but they can be seen as a linearised version of slip boundary conditions (see [TM05, Chapter 9.1] for details).

1.4 Inverse and trace inequalities

Let $\{\mathcal{T}_h\}_{h>0}$ be a regular family of triangulations of $\bar{\Omega}$. For each triangulation \mathcal{T}_h , \mathcal{E}_h denotes the set of its edges. In addition, for each of element $K \in \mathcal{T}_h$, $h_K := \text{diam}(K)$, and we define $h := \max_{K \in \mathcal{T}_h} h_K$. The following results are useful for the following discussion.

Lemma 1.1 (Local inverse inequality). *There exists $C > 0$, independent of h_K , such that for all $K \in \mathcal{T}_h$ and polynomial functions v in K the following inequality holds*

$$|v|_{H^s(K)} \leq Ch_K^{m-s} |v|_{H^m(K)}, \quad 0 \leq m \leq s.$$

Proof. See [EG04, Lemma 1.138] □

Lemma 1.2 (Discrete trace inequality). *There exists $C_{max} > 0$, independent of h_K , such that for all $K \in \mathcal{T}_h$ and polynomial function v in K the following inequality holds*

$$h_K^{\frac{1}{2}} \|v\|_{\partial K} \leq C_{max} \|v\|_K.$$

Proof. See [DPE12, Lemma 1.46]. □

Lemma 1.3 (Local trace inequality). *There exists $C > 0$, independent of h_K , such that for any $v \in H^1(K)$, the following local trace inequality holds*

$$(1.6) \quad \|v\|_{\partial K} \leq C \left(h_K^{-\frac{1}{2}} \|v\|_K + h_K^{\frac{1}{2}} |v|_{H^1(K)} \right).$$

Proof. Mapping our function to a reference unit triangle \hat{K} and using trace inequality on it (see [BS08, Theorem 1.6.6]) we obtain following result

$$(1.7) \quad \|\hat{v}\|_{\partial \hat{K}}^2 \leq \hat{C} \|\hat{v}\|_{\hat{K}} \|\hat{v}\|_{H^1(\hat{K})} \leq \hat{C} \|\hat{v}\|_{H^1(\hat{K})}^2.$$

By transforming back (see [EG04, Lemma 1.113]) to the triangle K we arrive to (1.6). □

It is worth mentioning that a trace estimate independent of h_K is not achievable. In fact it is enough to consider $v = 1$ in K .

1.5 Approximation results

We define the following Sobolev spaces on the triangulation \mathcal{T}_h and the set of all edges in \mathcal{E}_h

$$\begin{aligned} L^2(\mathcal{E}_h) &:= \{v : v|_E \in L^2(E) \ \forall E \in \mathcal{E}_h\}, \\ H^m(\mathcal{T}_h) &:= \{v \in L^2(\Omega) : v|_K \in H^m(K) \ \forall K \in \mathcal{T}_h\} \text{ for } m \in \mathbb{N}. \end{aligned}$$

Moreover, for $D \subset \Omega$, we use the following notation $\mathbb{P}_k(D)$ for a set of the all polynomials of degree k on the set D .

Now we will introduce the finite element spaces that discretise the above spaces. To discretise the velocity \mathbf{u} we use one of the following Brezzi-Douglas-Marini spaces (see [BBF13, Section 2.3.1])

$$\begin{aligned} \mathbf{BDM}_h^k &:= \left\{ \mathbf{v}_h \in H(\operatorname{div}, \Omega) : \mathbf{v}_h|_K \in [\mathbb{P}_k(K)]^2 \ \forall K \in \mathcal{T}_h \right\}, \\ \mathbf{BDM}_{h,0}^k &:= \left\{ \mathbf{v}_h \in H(\operatorname{div}, \Omega) : \mathbf{v}_h|_K \in [\mathbb{P}_k(K)]^2 \ \forall K \in \mathcal{T}_h \wedge (\mathbf{v}_h)_n = 0 \text{ on } \Gamma \right\}. \end{aligned}$$

These spaces are only $H(\operatorname{div})$ -conforming. This means that only the normal component across the edges is continuous, but not the tangential one. Hence, functions belonging to the \mathbf{BDM}_h^k space are, in general, discontinuous. In addition, $\mathbf{BDM}_{h,0}^k$ is naturally a subspace of

$$H_0(\operatorname{div}, \Omega) := \{ \mathbf{v} \in H(\operatorname{div}, \Omega) : v_n = 0 \text{ on } \Gamma \}.$$

The trace theorem justifying the definition of this space can be find in [GR86, Theorem 1.5]. It is important to remark that we shall consider approximation in $\mathbf{BDM}_{h,0}^k$ of functions in $H^1(\Omega)$ whose normal component vanishes. The existence of such a normal component is a consequence of the standard trace theorem.

In addition, for $1 \leq m \leq k+1$ and for each $K \in \mathcal{T}_h$ we define $\Pi_K^k : [H^m(K)]^2 \rightarrow [\mathbb{P}_k(K)]^2$ the local BDM projection as follows. For every $\mathbf{v} \in [H^m(K)]^2$, $\Pi_K^k(\mathbf{v})$ is the element of $[\mathbb{P}_k(K)]^2$ satisfying

$$(1.8) \quad \int_E \Pi_K^k(\mathbf{v}) \cdot \mathbf{n} w_h \, ds = \int_E v_n w_h \, ds \quad \forall w_h \in \mathbb{P}_k(E) \ \forall E \in (\mathcal{E}_h \cap \partial K),$$

$$(1.9) \quad \int_K \Pi_K^k(\mathbf{v}) \mathbf{w}_h \, d\mathbf{x} = \int_K \mathbf{v} \mathbf{w}_h \, d\mathbf{x} \quad \forall \mathbf{w}_h \in [\mathbb{P}_{k-2}(K)]^2 \text{ and } k \geq 2,$$

and we denote $\Pi^k : [H^m(\Omega)]^2 \rightarrow \mathbf{BDM}_h^k$, where $\Pi^k|_K := \Pi_K^k$ for all $K \in \mathcal{T}_h$ as the BDM projection (see [BBF13, Section 2.5]). Thanks to the (1.8) we can be sure that the projections preserves the normal-velocity boundary conditions.

Lemma 1.4 (Local Brezzi-Douglas-Marini approximation). *There exist $C_0^\Pi, C_1^\Pi > 0$, independent of h_K , such that for all $\mathbf{v} \in [H^m(K)]^2$, $1 \leq m \leq k+1$, the following interpolation*

estimates hold

$$\begin{aligned} \|\mathbf{v} - \Pi^k(\mathbf{v})\|_K &\leq C_0^\Pi h_K^m |\mathbf{v}|_{H^m(K)}, \\ |\mathbf{v} - \Pi^k(\mathbf{v})|_{H^1(K)} &\leq C_1^\Pi h_K^{m-1} |\mathbf{v}|_{H^m(K)}. \end{aligned}$$

Proof. Let us start with the first inequality. We use mapping to a reference unit triangle \hat{K} , Bramble-Hilbert Lemma (see [EG04, Lemma B.68]) and transforming back (see [EG04, Lemma 1.101]) to the triangle K to obtain

$$\|\mathbf{v} - \Pi^k(\mathbf{v})\|_K \leq c_0 h_K \left\| \hat{\mathbf{v}} - \hat{\Pi}^k(\hat{\mathbf{v}}) \right\|_{\hat{K}} \leq \hat{C}_0^\Pi h_K |\hat{\mathbf{v}}|_{H^m(\hat{K})} \leq C_0^\Pi h_K h_K^{m-1} |\mathbf{v}|_{H^m(K)}.$$

In the case of the second inequality we use the same approach

$$|\mathbf{v} - \Pi^k(\mathbf{v})|_{H^1(K)} \leq c_1 \left| \hat{\mathbf{v}} - \hat{\Pi}^k(\hat{\mathbf{v}}) \right|_{H^1(\hat{K})} \leq \hat{C}_1^\Pi |\hat{\mathbf{v}}|_{H^m(\hat{K})} \leq C_1^\Pi h_K^{m-1} |\mathbf{v}|_{H^m(K)}.$$

In both cases the constants C_0^Π, C_1^Π are independent of h_K . \square

The pressure is discretised using one of the following discontinuous spaces

$$\begin{aligned} Q_h^{k-1} &:= \{q_h \in L^2(\Omega) : q_h|_K \in \mathbb{P}_{k-1}(K) \ \forall K \in \mathcal{T}_h\}, \\ Q_{h,0}^{k-1} &:= \left\{ q_h \in L^2(\Omega) : q_h|_K \in \mathbb{P}_{k-1}(K) \ \forall K \in \mathcal{T}_h \wedge \int_\Omega q_h \, d\mathbf{x} = 0 \right\}. \end{aligned}$$

Moreover, in this case we define the local $L^2(K)$ -projection $\Psi_K^{k-1} : L^2(K) \rightarrow \mathbb{P}_{k-1}(K)$ for each $K \in \mathcal{T}_h$ defined as follows. For every $w \in L^2(K)$, $\Psi_K^k(w)$ is the element of $\mathbb{P}_{k-1}(K)$ satisfying

$$(1.10) \quad \int_K \Psi_K^k(w) v_h \, d\mathbf{x} = \int_K w v_h \, d\mathbf{x} \quad \forall v_h \in \mathbb{P}_{k-1}(K).$$

Lemma 1.5 (Local L^2 -projection approximation). *There exist $C_0^\Psi, C_1^\Psi > 0$, independent of h_K , such that for all $v \in H^m(K)$, $1 \leq m \leq k+1$, the following interpolation estimates hold*

$$\begin{aligned} \|v - \Psi_K^k(v)\|_K &\leq C_0^\Psi h_K^m |v|_{H^m(K)}, \\ |v - \Psi_K^k(v)|_{H^1(K)} &\leq C_1^\Psi h_K^{m-1} |v|_{H^m(K)}. \end{aligned}$$

Proof. The proof is similar to the one of Lemma 1.4 (see the proof of [EG04, Theorem 1.103]). \square

In order to have fewer degrees of freedom, we discretise the Lagrange multiplier \tilde{u} using one of the following spaces

$$M_h^{k-1} := \{\tilde{v}_h \in L^2(\mathcal{E}_h) : \tilde{v}_h|_E \in \mathbb{P}_{k-1}(E) \ \forall E \in \mathcal{E}_h\},$$

$$M_{h,0}^{k-1} := \{ \tilde{v}_h \in L^2(\mathcal{E}_h) : \tilde{v}_h|_E \in \mathbb{P}_{k-1}(E) \ \forall E \in \mathcal{E}_h \wedge \tilde{v}_h = 0 \text{ on } \Gamma \}.$$

The function in M_h^{k-1} are defined edge-wise, and hence, are in general discontinuous.

Furthermore, we introduce for all $E \in \mathcal{E}_h$ the $L^2(E)$ -projection $\Phi_E^{k-1} : L^2(E) \rightarrow \mathbb{P}_{k-1}(E)$ defined as follows. For every $\tilde{w} \in L^2(E)$, $\Phi_E^{k-1}(\tilde{w})$ is the element of $\mathbb{P}_{k-1}(E)$ satisfying

$$(1.11) \quad \int_E \Phi_E^{k-1}(\tilde{w}) \tilde{v}_h \, ds = \int_E \tilde{w} \tilde{v}_h \, ds \quad \forall \tilde{v}_h \in \mathbb{P}_{k-1}(E),$$

and we denote $\Phi^{k-1} : L^2(\mathcal{E}_h) \rightarrow M_h^{k-1}$, where $\Phi^{k-1}|_E := \Phi_E^{k-1}$ for all $E \in \mathcal{E}_h$.

Lemma 1.6 (Trace L^2 -projection approximation). *There exists $C^\Phi > 0$, independent of h_K , such that for all $v \in H^m(K)$, $1 \leq m \leq k+1$, the following interpolation estimate holds*

$$(1.12) \quad \|v - \Phi^k(v)\|_{\partial K} \leq C^\Phi h_K^{m-\frac{1}{2}} |v|_{H^m(K)}.$$

Proof. We use the local trace inequality (Lemma 1.3)

$$\|v - \Phi^k(v)\|_{\partial K} \leq \|v - \Psi^k(v)\|_{\partial K} \leq c_1 \left(h_K^{-\frac{1}{2}} \|v - \Psi^k(v)\|_K + h_K^{\frac{1}{2}} |v - \Psi^k(v)|_{H^1(K)} \right)$$

Thanks to the above local L^2 -projection approximation (Lemma 1.5) we arrive to (1.12). \square

1.6 TVNF boundary conditions

We begin by discretising the Stokes problem with TVNF boundary conditions defined by (1.4). In Section 1.6.2 we prove the well-posedness of the discrete problem obtained in Section 1.6.1. Later, we estimate the error of the hdG method for (1.4) in Section 1.6.3. We conclude with the numerical experiments that validate the theory of the previous sections.

1.6.1 The discrete problem

We start with a short discussion on the approach of the Lagrange multiplier and give a brief motivation on the idea to approximate the tangential component of \mathbf{u} by a Lagrange multiplier. The Stokes equation can be interpreted as the following minimisation problem

$$\min J(\mathbf{v}) := \frac{1}{2} \int_{\Omega} \nabla \mathbf{v} : \nabla \mathbf{v} \, d\mathbf{x} - \int_{\Omega} \mathbf{f} \mathbf{v} \, d\mathbf{x},$$

subject to $\nabla \cdot \mathbf{v} = 0$.

Then, to take into account the incompressibility condition $\nabla \cdot \mathbf{u} = 0$ in Ω , the pressure p is introduced as a Lagrange multiplier. In this manuscript we consider the discontinuous Galerkin approach that involves integration by parts. This process allows us to incorporate the traces of the normal and tangential component to the variational formulation. Although the saddle-point structure of such methods reflects well the above structure, since bilinear form $a_h(\mathbf{u}, \cdot)$ is a discrete version of $\nu \Delta \mathbf{u}$, $b_h(\cdot, p)$ of ∇p , and $b_h(\mathbf{u}, \cdot)$ of $\nabla \cdot \mathbf{u}$. Further more, since our approach is associated with the $H(\text{div})$ -conforming space for velocity, it provides us the continuity of the normal component of the velocity. The continuity of the tangential component needs to be fulfilled. If we consider the following saddle-point problem

$$\min J_h(\mathbf{v}_h) := \frac{1}{2} a_h(\mathbf{v}_h, \mathbf{v}_h) - \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x},$$

subject to $b_h(\mathbf{v}_h, q_h) = 0 \, \forall q_h \in Q_h$ and $(\mathbf{v}_h)_t$ is continuous across all edges

then the pressure p or the trace of the tangential component of the velocity can be introduced as a Lagrange multiplier. We have decided to use the trace of the tangential component of the velocity.

Let us denote $\mathbf{V}_h := \mathbf{BDM}_h^k \times M_{h,0}^{k-1}$. From now on we will use ∇ to denote the element-wise gradient. First, we multiply the first equation from (1.1) by a test function $\mathbf{v}_h \in \mathbf{BDM}_h^k$ and integrate by parts. This gives

$$(1.13) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \nu \partial_{\mathbf{n}} \mathbf{u} \mathbf{v}_h \, ds + \int_{\partial K} p (\mathbf{v}_h)_n \, ds \right).$$

Since the normal and tangential vectors are perpendicular ($\mathbf{n} \cdot \mathbf{t} = 0$) we can split (1.13) as

$$(1.14) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \sigma_{nt} (\mathbf{v}_h)_t \, ds - \int_{\partial K} \sigma_{nn} (\mathbf{v}_h)_n \, ds \right).$$

Using the definition of stress $\boldsymbol{\sigma} := \nu \nabla \mathbf{u} - p \mathbf{I} = (\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)^T$, we can rewrite the first equation of the Stokes problem (1.1) as $-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f}$. Since $\mathbf{f} \in L^2(\Omega)$, we know that $\nabla \cdot \boldsymbol{\sigma} = (\nabla \cdot \boldsymbol{\sigma}_1, \nabla \cdot \boldsymbol{\sigma}_2)^T$ and each $\boldsymbol{\sigma}_i \in H(\text{div}, \Omega)$, for $i = 1, 2$. That is why, $\boldsymbol{\sigma}_n$ is continuous across all interior edges. Moreover, since $\mathbf{v}_h \in \mathbf{BDM}_h^k$ and $H(\text{div})$ -conforming spaces preserve the continuity of the normal component of the velocity across the edges, then $(\mathbf{v}_h)_n$ is continuous across all interior edges. Then we can rewrite (1.14) as follows

$$(1.15) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} \right)$$

$$- \int_{\partial K} \sigma_{nt} (\mathbf{v}_h)_t ds \Big) - \int_{\Gamma} \sigma_{nn} (\mathbf{v}_h)_n ds.$$

Furthermore, since σ_n is continuous across all interior edges, then $\sum_{K \in \mathcal{T}_h} \int_{\partial K} \sigma_{nt} \tilde{v}_h ds = 0$, for all $\tilde{v}_h \in M_{h,0}^{k-1}$, and we can subtract this from (1.15) to get

$$(1.16) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h d\mathbf{x} \right. \\ \left. - \int_{\partial K} \sigma_{nt} ((\mathbf{v}_h)_t - \tilde{v}_h) ds \right) - \int_{\Gamma} \sigma_{nn} (\mathbf{v}_h)_n ds.$$

Denoting $\tilde{u} = u_t$ on \mathcal{E}_h , then $(u_t - \tilde{u}) = \Phi^{k-1}(u_t - \tilde{u}) = 0$ on \mathcal{E}_h and applying the boundary conditions (1.2) we can rewrite (1.16) as

$$(1.17) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h d\mathbf{x} \right. \\ \left. - \int_{\partial K} \nu (\partial_n \mathbf{u})_t ((\mathbf{v}_h)_t - \tilde{v}_h) ds \right. \\ \left. \pm \int_{\partial K} \nu (u_t - \tilde{u}) (\partial_n \mathbf{v}_h)_t ds \right. \\ \left. + \nu \frac{\tau}{h_K} \int_{\partial K} \Phi^{k-1}(u_t - \tilde{u}) \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) ds \right) \\ - \int_{\Gamma} g (\mathbf{v}_h)_n ds,$$

where $\tau > 0$ is a stabilisation parameter. In this process we have added terms that vanish for the exact solution. These terms are added for symmetry and consistency considerations, and will become relevant (i. e. non-zero) in the discrete formulation. Hence, we define the velocity bilinear form $a : \mathbf{V}_h \times \mathbf{V}_h \rightarrow \mathbb{R}$ as

$$(1.18) \quad a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h)) := \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{w}_h : \nabla \mathbf{v}_h d\mathbf{x} - \int_{\partial K} \nu (\partial_n \mathbf{w}_h)_t ((\mathbf{v}_h)_t - \tilde{v}_h) ds \right. \\ \left. + \varepsilon \int_{\partial K} \nu ((\mathbf{w}_h)_t - \tilde{w}_h) (\partial_n \mathbf{v}_h)_t ds \right. \\ \left. + \nu \frac{\tau}{h_K} \int_{\partial K} \Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h) \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) ds \right),$$

where $\varepsilon \in \{-1, 1\}$ and $\tau > 0$ is a stabilisation parameter and $b : \mathbf{V}_h \times Q_h^{k-1} \rightarrow \mathbb{R}$ as

$$(1.19) \quad b((\mathbf{v}_h, \tilde{v}_h), q_h) := - \sum_{K \in \mathcal{T}_h} \int_K q_h \nabla \cdot \mathbf{v}_h d\mathbf{x}.$$

With these definitions we propose the hybrid discontinuous Galerkin (hdG) method for the TVNF boundary value problem (1.4):

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^{k-1}$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^{k-1}$

$$(1.20) \quad \begin{cases} a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g(\mathbf{v}_h)_n \, ds \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) = 0 \end{cases}.$$

1.6.2 Well-posedness of the discrete problem

The discrete problem presented in (1.20) is a mixed formulation. Thus, to prove the existence and uniqueness of the solution we will use the Brezzi's theorem [Bre74]. Below, we present the formulation of this theorem using our notation.

Theorem 1.1 (Brezzi's theorem). *Let us assume that*

B1 \mathbf{V}_h, Q_h^{k-1} are Hilbert spaces;

B2 $a : \mathbf{V}_h \times \mathbf{V}_h \rightarrow \mathbb{R}, b : \mathbf{V}_h \times Q_h^{k-1} \rightarrow \mathbb{R}$ are continuous bilinear forms,

B3 $a : \mathbf{V}_h \times \mathbf{V}_h \rightarrow \mathbb{R}$ is coercive on $\text{Ker}(b) := \{\mathbf{v}_h \in \mathbf{V}_h : b(\mathbf{v}_h, q_h) = 0 \, \forall \, q_h \in Q_h^{k-1}\}$,

B4 $b : \mathbf{V}_h \times Q_h^{k-1} \rightarrow \mathbb{R}$ fulfils the inf-sup condition.

If all of the above assumptions are fulfilled, then the discrete problem (1.20) has unique solution $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^{k-1}$.

Before proving that assumptions are satisfied let us consider following semi-norm

$$(1.21) \quad |||(\mathbf{w}_h, \tilde{w}_h)|||^2 := \nu \sum_{K \in \mathcal{T}_h} \left(|\mathbf{w}_h|_{H^1(K)}^2 + h_K \|\partial_n \mathbf{w}_h\|_{\partial K}^2 + \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h)\|_{\partial K}^2 \right).$$

Lemma 1.7 (hdG norm). *The semi-norm $||| \cdot |||$ defined by (1.21) is a norm on \mathbf{V}_h .*

Proof. Since $||| \cdot |||$ is a semi-norm, we only need to show that

$$|||(\mathbf{w}_h, \tilde{w}_h)||| = 0 \Rightarrow \mathbf{w}_h = \mathbf{0} \text{ and } \tilde{w}_h = 0.$$

Let us suppose $(\mathbf{w}_h, \tilde{w}_h) \in \mathbf{V}_h$ and $|||(\mathbf{w}_h, \tilde{w}_h)||| = 0$. Then $\nabla \mathbf{w}_h = 0$ in all $K \in \mathcal{T}_h$, and thus $\mathbf{w}_h|_K = \mathbf{C}_K$ for all $K \in \mathcal{T}_h$. Now, since $\mathbf{w}_h \in [\mathbb{P}_0(K)]^2$ in every K

$$\|\Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h)\|_{\partial K} = 0 \Rightarrow (\mathbf{w}_h)_t = \tilde{w}_h \text{ in each } E \in \mathcal{E}_h.$$

Since \tilde{w}_h is single-valued on all the edges in \mathcal{E}_h , then $(\mathbf{w}_h)_t$ is continuous in Ω . Moreover, since \mathbf{w}_h belongs to \mathbf{BDM}_h^k , $(\mathbf{w}_h)_n$ is also continuous in Ω . Then, \mathbf{w}_h is continuous in Ω , and thus

$\mathbf{w}_h = \mathbf{C} \in \mathbb{R}^2$ in Ω . Finally,

$$(\mathbf{w}_h)_t = (\mathbf{C})_t = 0 \text{ on } \Gamma \Rightarrow \mathbf{w}_h = \mathbf{0} \text{ in } \Omega,$$

which, since $\tilde{w}_h = (\mathbf{w}_h)_t$ on every edge, finishes the proof. \square

The first assumption **B1** is fulfilled, because these are discretisations of Hilbert spaces. Let us now prove the assumption **B2**.

Lemma 1.8 (Continuity of bilinear forms). *There exists a constant $C > 0$ such that, for all $(\mathbf{w}, \tilde{w}), (\mathbf{v}, \tilde{v}) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\mathcal{E}_h)$ and $q \in L^2(\Omega)$, we have*

$$(1.22) \quad |a((\mathbf{w}, \tilde{w}), (\mathbf{v}, \tilde{v}))| \leq C ||| (\mathbf{w}, \tilde{w}) ||| ||| (\mathbf{v}, \tilde{v}) |||,$$

$$(1.23) \quad |b((\mathbf{w}, \tilde{w}), q)| \leq \sqrt{\frac{2}{\nu}} ||| (\mathbf{w}, \tilde{w}) ||| \|q\|_{\Omega}.$$

Proof. Let us start with (1.22). Using the Cauchy-Schwarz inequality we get

$$\begin{aligned} |a((\mathbf{w}, \tilde{w}), (\mathbf{v}, \tilde{v}))| &\leq 2 ||| (\mathbf{w}, \tilde{w}) ||| ||| (\mathbf{v}, \tilde{v}) ||| \\ &\quad + \sum_{K \in \mathcal{T}_h} (\nu \|\partial_n \mathbf{w}\|_{\partial K} \|v_t - \tilde{v}\|_{\partial K} + \nu \|\partial_n \mathbf{v}\|_{\partial K} \|w_t - \tilde{w}\|_{\partial K}). \end{aligned}$$

Therefore, using the triangle inequality and the trace L^2 -projection approximation (Lemma 1.6) we get

$$(1.24) \quad \begin{aligned} \|\partial_n \mathbf{w}\|_{\partial K} \|v_t - \tilde{v}\|_{\partial K} &\leq \|\partial_n \mathbf{w}\|_{\partial K} \|v_t - \Phi^{k-1}(v_t)\|_{\partial K} + \|\partial_n \mathbf{w}\|_{\partial K} \|\Phi^{k-1}(v_t - \tilde{v})\|_{\partial K} \\ &\leq \sqrt{h_K} \|\partial_n \mathbf{w}\|_{\partial K} \left(\tilde{c}_1 |\mathbf{v}|_{H^1(K)} + \frac{1}{\sqrt{h_K}} \|\Phi^{k-1}(v_t - \tilde{v})\|_{\partial K} \right). \end{aligned}$$

Using the Cauchy-Schwarz inequality

$$\begin{aligned} \nu \|\partial_n \mathbf{w}\|_{\partial K} \|v_t - \tilde{v}\|_{\partial K} &\leq c_1 ||| (\mathbf{w}, \tilde{w}) ||| ||| (\mathbf{v}, \tilde{v}) |||, \\ \nu \|\partial_n \mathbf{v}\|_{\partial K} \|w_t - \tilde{w}\|_{\partial K} &\leq c_2 ||| (\mathbf{v}, \tilde{v}) ||| ||| (\mathbf{w}, \tilde{w}) |||. \end{aligned}$$

Finally, we get (1.22) for $C_a = (2 + c_1 + c_2)$.

Now we get the second inequality by using Cauchy-Schwarz inequality

$$|b((\mathbf{w}, \tilde{w}), p)| \leq \sqrt{2} \sqrt{\sum_{K \in \mathcal{T}_h} \|p\|_K^2} \sqrt{\sum_{K \in \mathcal{T}_h} \|\nabla \mathbf{w}\|_{\partial K}^2} \leq \sqrt{\frac{2}{\nu}} \|p\|_{\Omega} ||| (\mathbf{w}, \tilde{w}) |||.$$

\square

Now we will show the assumption **B3** associated with the bilinear form a .

Lemma 1.9 (Coercivity of the bilinear form a). *There exists $\alpha > 0$ such that for all $(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h$*

$$(1.25) \quad a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) \geq \alpha |||(\mathbf{v}_h, \tilde{v}_h)|||^2.$$

If $\varepsilon = -1$ in the definition (1.18), then this only holds under the additional hypothesis of τ being large enough. If $\varepsilon = 1$ in (1.18), this inequality holds for arbitrary $\tau > 0$.

Proof. First, since $\partial_n \mathbf{v}_h|_E \in [\mathbb{P}_{k-1}(E)]^2$ for all $E \in \mathcal{E}_h$, then

$$(1.26) \quad \begin{aligned} a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) &= \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 - \nu(1-\varepsilon) \int_{\partial K} (\partial_n \mathbf{v}_h)_t \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) ds \right. \\ &\quad \left. + \nu \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)\|_{\partial K}^2 \right). \end{aligned}$$

It only remains to bound the middle term in terms of the other two. For that, we consider two cases.

- if $\varepsilon = 1$, then (1.26) reduces to

$$(1.27) \quad a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) = \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 + \nu \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)\|_{\partial K}^2 \right).$$

It only remains to show that the right hand side of (1.27) is an upper bound (up to a constant) for the norm $||| \cdot |||$ given by (1.21). Using the discrete trace inequality (Lemma 1.2) we get

$$\sum_{K \in \mathcal{T}_h} h_K \|\partial_n \mathbf{v}_h\|_{\partial K}^2 \leq \sum_{K \in \mathcal{T}_h} C_{max}^2 |\mathbf{v}_h|_{H^1(K)}^2,$$

and then

$$(1.28) \quad |||(\mathbf{v}_h, \tilde{v}_h)|||^2 \leq (1 + C_{max}^2) \sum_{K \in \mathcal{T}_h} \nu \left(|\mathbf{v}_h|_{H^1(K)}^2 + \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)\|_{\partial K}^2 \right),$$

which proves (1.25) with $\alpha = \frac{1}{1+C_{max}^2}$.

- if $\varepsilon = -1$, then (1.26) becomes

$$\begin{aligned} a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) &= \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 - 2\nu \int_{\partial K} (\partial_n \mathbf{v}_h)_t \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) ds \right. \\ &\quad \left. + \nu \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)\|_{\partial K}^2 \right). \end{aligned}$$

Using the Cauchy-Schwarz inequality

$$a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) \geq \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 - 2\nu \|\partial_n \mathbf{v}_h\|_{\partial K} \|\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)\|_{\partial K} \right)$$

$$+\nu \frac{\tau}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 \Big).$$

Since $\mathbf{v}_h \in \mathbf{BDM}_h^k$ is a piecewise polynomial, we can apply the discrete trace inequality (Lemma 1.2) to the second term to arrive at

$$\begin{aligned} a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) &\geq \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 - 2\nu \frac{C_{max}}{\sqrt{h_K}} |\mathbf{v}_h|_{H^1(K)} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K} \right. \\ &\quad \left. + \nu \frac{\tau}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 \right). \end{aligned}$$

Next, we apply Young's inequality $ab \leq \frac{a^2}{2} + \frac{b^2}{2}$ with $b = 2\sqrt{\nu} \frac{C_{max}}{\sqrt{h_K}} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}$ and $a = \sqrt{\nu} |\mathbf{v}_h|_{H^1(K)}$ to get

$$\begin{aligned} a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) &\geq \sum_{K \in \mathcal{T}_h} \left(\nu |\mathbf{v}_h|_{H^1(K)}^2 - \frac{\nu}{2} |\mathbf{v}_h|_{H^1(K)}^2 \right. \\ &\quad \left. - 2\nu \frac{C_{max}^2}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 + \nu \frac{\tau}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 \right) \\ &= \sum_{K \in \mathcal{T}_h} \left(\frac{\nu}{2} |\mathbf{v}_h|_{H^1(K)}^2 + \nu \frac{\tau - 2C_{max}^2}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 \right) \\ &\geq \nu C \sum_{K \in \mathcal{T}_h} \left(|\mathbf{v}_h|_{H^1(K)}^2 + \frac{\tau}{h_K} \left\| \Phi^{k-1} ((\mathbf{v}_h)_t - \tilde{v}_h) \right\|_{\partial K}^2 \right). \end{aligned}$$

Finally, if we suppose $\tau > 2C_{max}^2$, then $C := \min \left\{ \frac{1}{2}, \frac{\tau - 2C_{max}^2}{\tau} \right\} > 0$, using (1.28) we get (1.25) for $\alpha = \frac{C}{1 + C_{max}^2}$. \square

We only left with the last assumption **B4** to prove.

Lemma 1.10 (Inf-sup condition for bilinear form b). *There exists $\beta > 0$ independent of h_K such that*

$$\sup_{(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h} \frac{b((\mathbf{v}_h, \tilde{v}_h), q_h)}{\|(\mathbf{v}_h, \tilde{v}_h)\|} \geq \frac{\beta}{\sqrt{\nu}} \|q_h\|_{\Omega} \quad \forall q_h \in Q_h^{k-1}.$$

Proof. We use the idea of the Fortin criterion, see [EG04, Lemma 4.19]. We need to prove that there exists a Fortin operator $\mathbf{\Pi} : [H^1(\Omega)]^2 \rightarrow \mathbf{V}_h$ such that for every $\mathbf{v} \in [H^1(\Omega)]^2$ the following conditions hold

$$(1.29) \quad b((\mathbf{v}, \tilde{v}), q_h) = b(\mathbf{\Pi}(\mathbf{v}), q_h) \quad \forall q_h \in Q_h^{k-1},$$

$$(1.30) \quad \|(\mathbf{\Pi}(\mathbf{v}))\| \leq C\sqrt{\nu} \|\mathbf{v}\|_{H^1(\Omega)}.$$

If we prove that (1.29) and (1.30) hold, then the discrete inf-sup condition is satisfied. This follows because, in [BBF13, Section 1.2] it is shown that for given $q_h \in Q_h$, there exists $\mathbf{v} \in$

$H_0^1(\Omega)$ such that

$$(1.31) \quad \sum_{K \in \mathcal{T}_h} \int_K q_h \nabla \cdot \mathbf{v} \, d\mathbf{x} = \|q_h\|_{\Omega}^2,$$

$$(1.32) \quad \|\mathbf{v}\|_{H^1(\Omega)} \leq c_1 \|q_h\|_{\Omega}^2.$$

In fact, if we use the Fortin operator properties (1.29) and (1.30), and regularity results (1.31) and (1.32) to obtain

$$\sup_{(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h} \frac{b((\mathbf{v}_h, \tilde{v}_h), q_h)}{\|(\mathbf{v}_h, \tilde{v}_h)\|} \geq \frac{b(\mathbf{\Pi}(\mathbf{v}), q_h)}{\|\mathbf{\Pi}(\mathbf{v})\|} \geq \frac{b((\mathbf{v}, \tilde{v}), q_h)}{C\sqrt{\nu}\|\mathbf{v}\|_{H^1(\Omega)}} \geq \frac{1}{Cc_1\sqrt{\nu}} \frac{\|q_h\|_{\Omega}^2}{\|q_h\|_{\Omega}},$$

we prove the discrete inf-sup condition with $\beta = \frac{1}{Cc_1}$.

That is why, in this proof we focus on the existence of the Fortin operator. Let $\mathbf{v} \in [H^1(\Omega)]^2$ and let us consider the operator $\mathbf{\Pi}(\mathbf{v}) := (\Pi^k(\mathbf{v}), \Phi^{k-1}(v_t))$. Let us note that the bilinear form b does not depend on the \tilde{v} or $\Phi^{k-1}(v_t)$. Moreover, in [BBF13, Section 2.5] it is shown that for all $\mathbf{v} \in [H_0^1(\Omega)]^2$ the following holds

$$\sum_{K \in \mathcal{T}_h} \int_K \nabla \cdot \mathbf{v} q_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \int_K \nabla \cdot \Pi^k(\mathbf{v}) q_h \, d\mathbf{x} \quad \forall q_h \in Q_h^{k-1},$$

which gives

$$b((\mathbf{v}, \tilde{v}), q_h) = \sum_{K \in \mathcal{T}_h} \int_K \nabla \cdot \mathbf{v} q_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \int_K \nabla \cdot \Pi^k(\mathbf{v}) q_h \, d\mathbf{x} = b(\mathbf{\Pi}(\mathbf{v}), q_h) \quad \forall q_h \in Q_h^{k-1}.$$

Hence, (1.29) is satisfied. To prove (1.30) we denote $(\mathbf{w}_h, \tilde{w}_h) := \mathbf{\Pi}(\mathbf{v})$. Then using the discrete trace inequality (Lemma 1.2) and the fact that Φ^{k-1} projection is a bounded operator, we get

$$(1.33) \quad \begin{aligned} \|(\mathbf{w}_h, \tilde{w}_h)\|^2 &= \sum_{K \in \mathcal{T}_h} \nu \left(|\mathbf{w}_h|_{H^1(K)}^2 + h_K \|\partial_{\mathbf{n}} \mathbf{w}_h\|_{\partial K}^2 + \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h)\|_{\partial K}^2 \right) \\ &\leq \sum_{K \in \mathcal{T}_h} \nu \left((1 + C_{max}^2) |\mathbf{w}_h|_{H^1(K)}^2 + \frac{\tau}{h_K} \|(\mathbf{w}_h)_t - \tilde{w}_h\|_{\partial K}^2 \right). \end{aligned}$$

Now applying the triangle inequality for the last term of (1.33) we arrive at

$$(1.34) \quad \begin{aligned} \|(\mathbf{w}_h, \tilde{w}_h)\|^2 &\leq \sum_{K \in \mathcal{T}_h} \nu \left((1 + C_{max}^2) |\mathbf{w}_h|_{H^1(K)}^2 + \frac{2\tau}{h_K} \left(\|(\mathbf{w}_h)_t - v_t\|_{\partial K}^2 + \|v_t - \tilde{w}_h\|_{\partial K}^2 \right) \right) \\ &=: \sum_{K \in \mathcal{T}_h} \nu \left((1 + C_{max}^2) \mathfrak{F}_1^K + \frac{2\tau}{h_K} (\mathfrak{F}_2^K + \mathfrak{F}_3^K) \right). \end{aligned}$$

Using the stability of Π^k we get

$$(1.35) \quad \mathfrak{F}_1^K = |\Pi^k(\mathbf{v})|_{H^1(K)}^2 \leq c_1 |\mathbf{v}|_{H^1(K)}^2.$$

Using Lemma 1.4 and the local trace inequality (Lemma 1.3), then

$$\begin{aligned}
\mathfrak{I}_2^K &\leq \| \mathbf{v} - \mathbf{w}_h \|_{\partial K}^2 \leq \tilde{c}_1 \left(\frac{1}{h_K} \| \mathbf{v} - \mathbf{w}_h \|_K^2 + h_K | \mathbf{v} - \mathbf{w}_h |_{H^1(K)}^2 \right) \\
(1.36) \qquad &\leq \tilde{c}_1 \left(\tilde{c}_2 h_K | \mathbf{v} |_{H^1(K)}^2 + \tilde{c}_3 h_K | \mathbf{v} |_{H^1(K)}^2 \right) \leq \tilde{c}_1 (\tilde{c}_2 + \tilde{c}_3) h_K | \mathbf{v} |_{H^1(K)}^2.
\end{aligned}$$

Finally, using the trace L^2 -projection approximation (Lemma 1.6) for the third term we get

$$(1.37) \qquad \mathfrak{I}_3^K \leq \tilde{c}_4 h_K | \mathbf{v} |_{H^1(K)}^2.$$

Then collecting (1.35), (1.36) and (1.37), we obtain (1.30) with

$$C := \sqrt{\left((1 + C_{max}^2) c_1 + 2\tau \tilde{c}_1 (\tilde{c}_2 + \tilde{c}_3) + 2\tau \tilde{c}_4 \right)},$$

which finishes the proof of existence of Fortin operator, and then finishes the proof. \square

Since all assumption are fulfilled, according to Brezzi's theorem (Theorem 1.1), problem (1.20) is well-posed. Moreover, the discrete problem (1.20) has one more property which is consistency.

Lemma 1.11 (Consistency). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\Omega)$ be the solution of the problem (1.4) and $\tilde{u} = u_t$ on all edges of \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^{k-1}$ solves (1.20), then for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^{k-1}$ the following holds*

$$a((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h), q_h) + b((\mathbf{v}_h, \tilde{v}_h), p - p_h) = 0.$$

Proof. Since we consider the divergence-free Stokes problem, for all $q \in L^2(\Omega)$

$$(1.38) \qquad b((\mathbf{u}, \tilde{u}), q) = - \sum_{K \in \mathcal{T}_h} \int_K q \nabla \cdot \mathbf{u} \, dx = 0.$$

Let us consider an arbitrary $(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h$. Using integration by parts for the left hand side of the first equation of the problem (1.20) and the assumption about \tilde{u} as a trace of \mathbf{u} , we get

$$\begin{aligned}
a((\mathbf{u}, \tilde{u}), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p) &= \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, dx - \int_K p \nabla \cdot \mathbf{v}_h \, dx \right. \\
&\quad - \nu \int_{\partial K} (\partial_n \mathbf{u})_t ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \\
&\quad + \varepsilon \nu \int_{\partial K} (u_t - \tilde{u}) (\partial_n \mathbf{v}_h)_t \, ds \\
&\quad \left. + \nu \frac{\tau}{h_K} \int_{\partial K} (\Phi^{k-1}(u_t - \tilde{u})) (\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)) \, ds \right) \\
&= \sum_{K \in \mathcal{T}_h} \left(-\nu \int_K \nabla \cdot (\nabla \mathbf{u}) \mathbf{v}_h \, dx + \int_K \nabla p \mathbf{v}_h \, dx \right)
\end{aligned}$$

$$(1.39) \quad + \int_{\Gamma} \sigma_{nn}(\mathbf{v}_h)_n \, ds + \int_{\Gamma} \sigma_{nt} \tilde{v}_h \, ds.$$

Since the TVNF boundary conditions (1.2) and the fact that $(\tilde{v}_h)_t = 0$ on Γ , we arrive at

$$(1.40) \quad a((\mathbf{u}, \tilde{u}), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} \sigma_{nn}(\mathbf{v}_h)_n \, ds.$$

The proof is finished by adding (1.38) and (1.40). \square

1.6.3 Error analysis

In this section we present the error estimates for the method. At the beginning, let us denote the following norm

$$(1.41) \quad |||(\mathbf{u}, \tilde{u}, p)|||_h := |||(\mathbf{u}, \tilde{u})||| + \frac{1}{\sqrt{\nu}} \|p\|_{\Omega}.$$

The first step is the following version of Cea's lemma [EG04, Lemma 2.28].

Lemma 1.12 (Cea's Lemma). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\Omega)$ be the solution of (1.4) and $\tilde{u} = u_t$ on all edges in \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^{k-1}$ solves (1.20), then there exists $C > 0$, independent of h and ν , such that*

$$(1.42) \quad |||(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)|||_h \leq C \inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^{k-1}} |||(\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h)|||_h.$$

Proof. Let us denote

$$B((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) := a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), r_h) + b((\mathbf{w}_h, \tilde{w}_h), q_h).$$

Using Lemma 1.9 and 1.10, and [EG04, Preposition 2.36] we get the following stability for B . There exists $\beta_B > 0$, independent of h and ν , such that

$$(1.43) \quad \sup_{(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{V}_h \times Q_h^{k-1}} \frac{B((\mathbf{v}_h, \tilde{v}_h, q_h), (\mathbf{w}_h, \tilde{w}_h, r_h))}{|||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h} \geq \beta_B |||(\mathbf{v}_h, \tilde{v}_h, q_h)|||_h$$

for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^{k-1}$. Now using Lemma 1.8, we get continuity of B .

There exists $C_B > 0$, such that

$$(1.44) \quad |B((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))| \leq C_B |||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h |||(\mathbf{v}_h, \tilde{v}_h, q_h)|||_h.$$

Now let $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h$, then using the triangle inequality, we get

$$\begin{aligned} |||(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)|||_h &\leq |||(\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h)|||_h \\ &\quad + |||(\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h)|||_h. \end{aligned}$$

There exists $(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{V}_h \times Q_h^{k-1}$ such that $|||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h = 1$ and which satisfies (1.43). Then, using Lemma 1.11 and (1.44), we arrive at

$$\begin{aligned} |||(\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h)|||_h &\leq \frac{1}{\beta_B} B((\mathbf{v}_h - \mathbf{u}, \tilde{v}_h - \tilde{u}, q_h - p), (\mathbf{w}_h, \tilde{w}_h, r_h)) \\ &\leq \frac{C_B}{\beta_B} |||(\mathbf{v}_h - \mathbf{u}, \tilde{v}_h - \tilde{u}, q_h - p)|||_h. \end{aligned}$$

Thus, we get (1.42) with $C := 1 + \frac{C_B}{\beta_B}$. \square

Using standard interpolation estimates, the following error estimate is proved.

Lemma 1.13 (hdG error). *Let \mathcal{T}_h be a shape regular mesh and $k \geq 1$. Moreover, we assume $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^{k+1}(\mathcal{T}_h)]^2 \times H^k(\mathcal{T}_h)$ is the solution of (1.4) and $\tilde{u} = u_t$ on all edges in \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^{k-1}$ solves the mixed problem (1.20), then there exists $C > 0$, independent of h , such that*

$$(1.45) \quad |||(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)|||_h \leq Ch^k \left(\sqrt{\nu} \|\mathbf{u}\|_{H^{k+1}(\mathcal{T}_h)} + \frac{1}{\sqrt{\nu}} \|p\|_{H^k(\mathcal{T}_h)} \right).$$

Proof. Let us consider the Fortin operator $\mathbf{\Pi}$ defined by (1.29), (1.30). If $\mathbf{\Pi}(\mathbf{u}) = (\mathbf{w}_h, \tilde{w}_h)$, then by using the triangle inequality and boundedness of the projection Φ^{k-1}

$$\begin{aligned} |||(\mathbf{u} - \mathbf{w}_h, \tilde{u} - \tilde{w}_h)|||^2 &= \sum_{K \in \mathcal{T}_h} \nu \left(|\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 + h_K \|\partial_n(\mathbf{u} - \mathbf{w}_h)\|_{\partial K}^2 \right. \\ &\quad \left. + \frac{\tau}{h_K} \|\Phi^{k-1}((\mathbf{u} - \mathbf{w}_h)_t - (\tilde{u} - \tilde{w}_h))\|_{\partial K}^2 \right) \\ &\leq \sum_{K \in \mathcal{T}_h} \nu \left(|\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 + h_K \|\partial_n(\mathbf{u} - \mathbf{w}_h)\|_{\partial K}^2 \right. \\ &\quad \left. + \frac{2c_1\tau}{h_K} \left(\|\mathbf{u} - \mathbf{w}_h\|_{\partial K}^2 + \|\tilde{u} - \tilde{w}_h\|_{\partial K}^2 \right) \right) \\ (1.46) \quad &=: \sum_{K \in \mathcal{T}_h} \nu \left(\mathfrak{T}_1^K + h_K \mathfrak{T}_2^K + \frac{2c_1\tau}{h_K} (\mathfrak{T}_3^K + \mathfrak{T}_4^K) \right). \end{aligned}$$

For the first term from (1.46), we use the BDM approximation (Lemma 1.4) to get

$$(1.47) \quad \mathfrak{T}_1^K \leq c_2 h_K^{2k} |\mathbf{u}|_{H^{k+1}(K)}^2.$$

Next we use the local trace inequality (Lemma 1.3) to get

$$(1.48) \quad \mathfrak{T}_2^K \leq c_3 \left(\frac{1}{h_K} |\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 + h_K |\mathbf{u} - \mathbf{w}_h|_{H^2(K)}^2 \right).$$

Let $\mathcal{L}^k \mathbf{u}$ be the usual Lagrange interpolant of degree k of \mathbf{u} (see [EG04, Example 131]). Using the triangle inequality followed by the local inverse inequality (Lemma 1.1), the local Lagrange

approximation [EG04, Example 1106] and Lemma 1.4, (1.48) become

$$\begin{aligned}\mathfrak{I}_2^K &\leq c_3 \left(\frac{1}{h_K} |\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 + 2h_K |\mathbf{u} - \mathcal{L}^k \mathbf{u}|_{H^2(K)}^2 + 2h_K |\mathcal{L}^k \mathbf{u} - \mathbf{w}_h|_{H^2(K)}^2 \right) \\ &\leq c_3 \left((c_4 + 2c_5) h_K^{2k-1} |\mathbf{u}|_{H^{k+1}(K)}^2 + \frac{2c_6}{h_K} |\mathcal{L}^k \mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 \right).\end{aligned}$$

Thanks to the hypothesis $\mathbf{u} \in [H^1(\Omega) \cap H^{k+1}(\mathcal{T}_h)]^2$, we have $\mathbf{u} \in [C^0(\bar{\Omega})]^2$ (see [EG04, Corollary B.43]), which justifies the use of the Lagrange interpolation projector. Now, using the triangle inequality, the local Lagrange approximation and Lemma 1.4 once again, we get

$$(1.49) \quad \begin{aligned}\mathfrak{I}_2^K &\leq c_3 \left((c_4 + 2c_5) h_K^{2k-1} |\mathbf{u}|_{H^{k+1}(K)}^2 + \frac{4c_6}{h_K} |\mathcal{L}^k \mathbf{u} - \mathbf{u}|_{H^1(K)}^2 + \frac{4c_6}{h_K} |\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 \right) \\ &\leq c_3 (c_4 + 2c_5 + 4c_6(c_7 + c_8)) h_K^{2k-1} |\mathbf{u}|_{H^{k+1}(K)}^2.\end{aligned}$$

For the third term in (1.46), we use the local trace inequality (Lemma 1.3) and Lemma 1.4 to get

$$(1.50) \quad \mathfrak{I}_3^K \leq c_9 \left(\frac{1}{h_K} \|\mathbf{u} - \mathbf{w}_h\|_K^2 + h_K |\mathbf{u} - \mathbf{w}_h|_{H^1(K)}^2 \right) \leq c_9 c_{10} h_K^{2k+1} |\mathbf{u}|_{H^{k+1}(K)}^2.$$

The last term in (1.46) is bounded using Lemma 1.6 as follows

$$(1.51) \quad \mathfrak{I}_4^K \leq c_{11} h_K^{2k+1} |\mathbf{u}|_{H^{k+1}(K)}^2.$$

Finally, the local L^2 -projection approximation (Lemma 1.5) gives

$$(1.52) \quad \inf_{q \in Q_h^{k-1}} \|p - q_h\|_\Omega = \|p - \Psi_h^{k-1}(p)\|_\Omega \leq \tilde{c}_1 h_K^k \|p\|_{H^k(\mathcal{T}_h)}.$$

Thus, putting together (1.46) with (1.47), (1.49), (1.50), (1.51), (1.52) and the shape regularity of the mesh we get

$$\inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h} \|(\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h)\|_h \leq \hat{C} h^k \left(\sqrt{\nu} \|\mathbf{u}\|_{H^{k+1}(\mathcal{T}_h)} + \frac{1}{\sqrt{\nu}} \|p\|_{H^k(\mathcal{T}_h)} \right),$$

with

$$\hat{C} := \max \left\{ \sqrt{c_2 + c_3(c_4 + 2c_5 + 4c_6(c_7 + c_8)) + 2\tau c_1 c_9 c_{10} + 2\tau c_1 c_{11}}, \tilde{c}_1 \right\},$$

and the result (1.45) follows from Lemma 1.12. \square

1.6.4 Convergence validation

The computational domain for three first test cases considered here is the unit square $\Omega = (0, 1)^2$. We present the results for $k = 1$, that is the discrete space is given by $\mathbf{BDM}_h^1 \times M_{h,0}^0 \times Q_h^0$. We test both the symmetric method ($\varepsilon = -1$) and the non-symmetric method ($\varepsilon = 1$).

For all cases we have followed the recommendation given in [Leh10, Section 2.5.2] and taken $\tau = 6$. All examples aim to verify the formulation with TVNF boundary conditions (1.20).

Example 1. We choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} [100(1 - \cos((1 - x)^2)) \sin(x^2) \sin(y^2) (1 - \cos((1 - y)^2))], \quad p = \tan(xy).$$

The analytic solution is depicted in Figure 1.1.

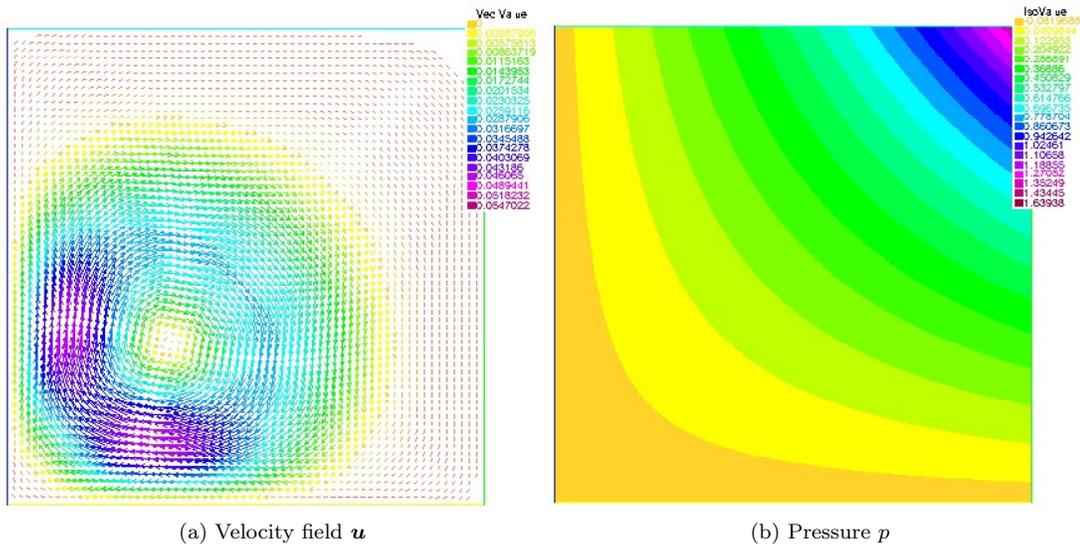


Figure 1.1: Analytic solution - Example 1

In Figures 1.2a and 1.2b we show the results of the usual convergence order tests for the symmetric case and the non-symmetric case by plotting the error as a function of the size of the mesh using a log-log scale. We notice that they validate the theory from Section 1.6.3. In addition, an optimal h^2 convergence rate is observed for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$. The proof of this fact is lacking, but it does not seem to be an easy task due to the nature of the boundary conditions of problem (1.1).

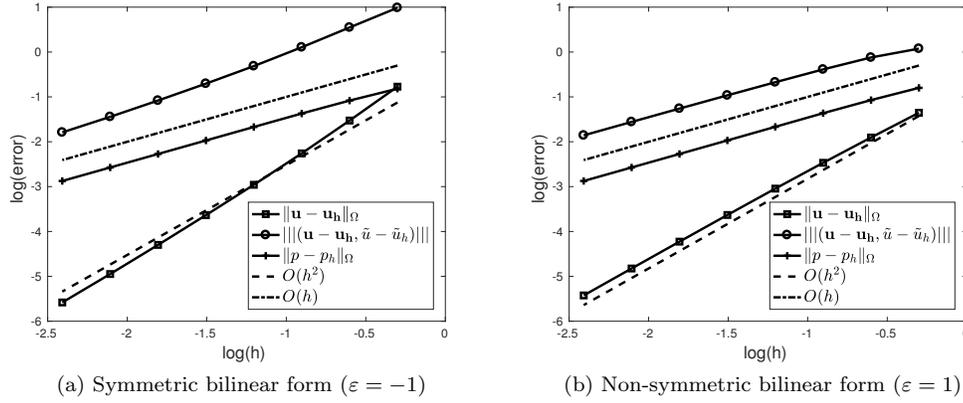


Figure 1.2: Error convergence of the hdG method with TVNF boundary conditions - Example 1

Example 2. We choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[x^2 (1-x)^2 y^2 (1-y)^2 \right], \quad p = x^2 - y^2.$$

The analytic solution is depicted in Figure 1.3.

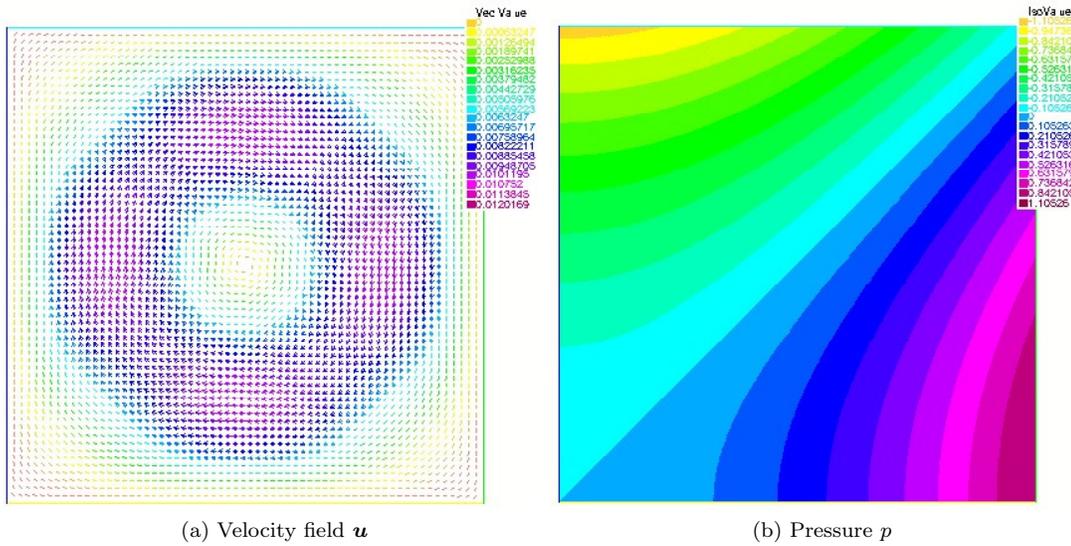


Figure 1.3: Analytic solution - Example 2

The error convergence with respect to the size of the mesh is depicted on the log-log plots for the symmetric case and the non-symmetric case in Figures 1.4a and 1.4b, respectively. We can see that they not only validate the theory from Section 1.6.3, but also perform an optimal h^2

convergence rate for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

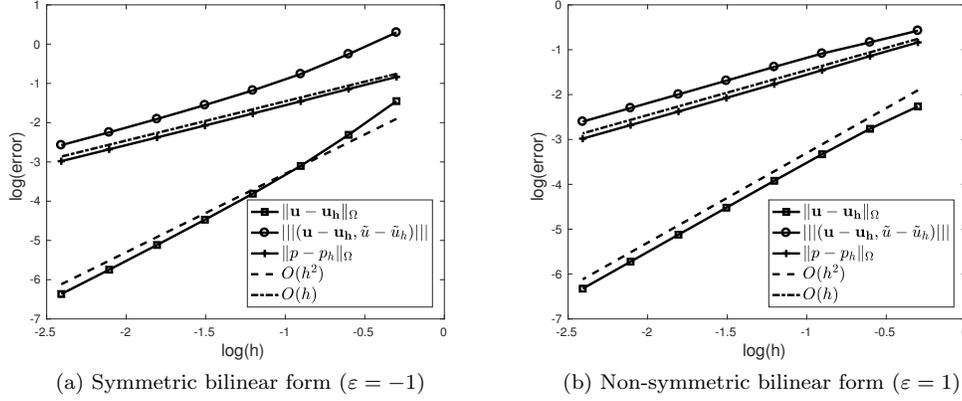


Figure 1.4: Error convergence of the hdG method with TVNF boundary conditions - Example 2

Example 3. We consider Poiseuille problem and we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = [4y(1 - y), 0]^T, \quad p = 4 - 8x.$$

The analytic solution is depicted in Figure 1.5.

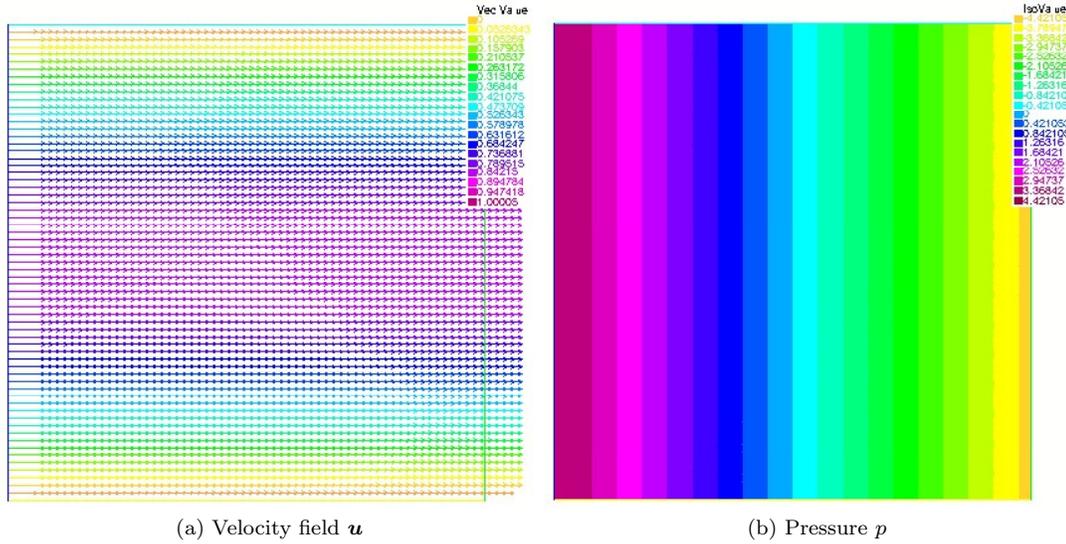


Figure 1.5: Analytic solution - Example 3

The conclusion from Figures 1.6a and 1.6b remains the same as in previous examples.

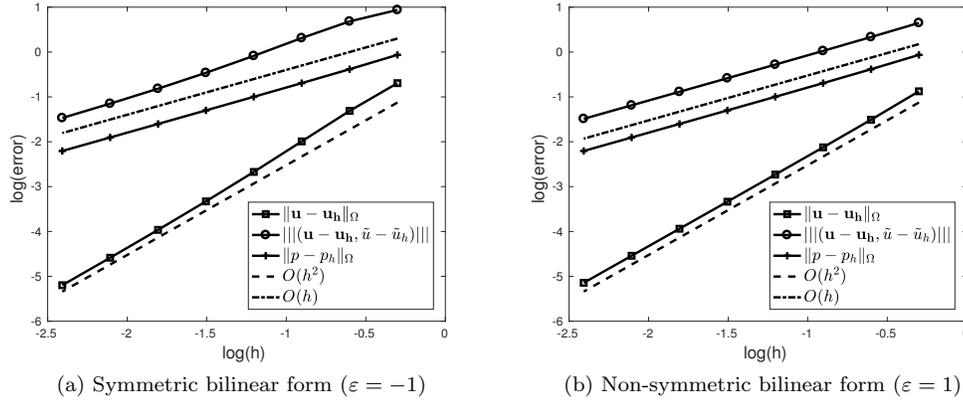


Figure 1.6: Error convergence of the hdG method with TVNF boundary conditions - Example 3

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions given by

$$(1.53) \quad \begin{cases} \mathbf{u}(x, y) = (4y(1 - y), 0)^T & \text{if } x = 0 \\ \sigma_{nn}(x, y) = 0, \quad u_t(x, y) = 0 & \text{if } x = 1.5 \\ \mathbf{u}(x, y) = (0, 0)^T & \text{otherwise} \end{cases} .$$

The solution of the symmetric hdG discretisation on a mesh containing 4 712 triangles is depicted in Figure 1.7. Since the analytic solution of this problem is unknown, we solved the

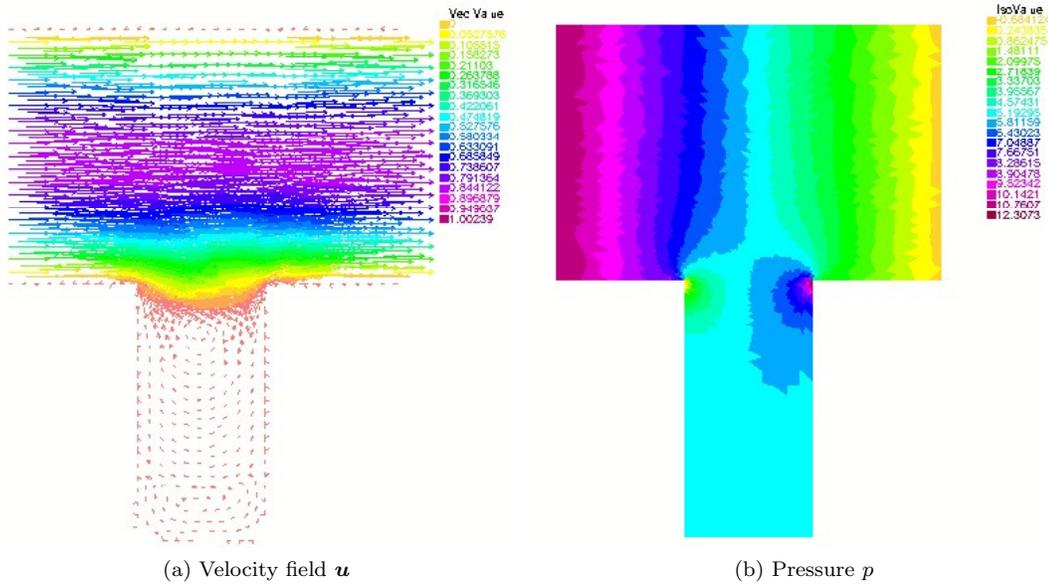


Figure 1.7: hdG solution with TVNF boundary conditions - Example 4

problem using the the lowest order Taylor-Hood discretisation [GR86, Chapter II, Section 4.2] on a mesh containing 2 046 150 triangles. The obtained solution is used as a reference solution $(\mathbf{u}_{ref}, p_{ref})$ to calculate the error of the hdG methods. Since the domain contains two re-entrant corners in points $(0.5, 0)$ and $(1, 0)$, we can expect some unstable behaviour at these points. To prevent it we refine the mesh in the neighbourhood of these re-entrant corners (see Figure 1.8). The refinement does not use any automatic mesh adaptivity, it doubles the number of triangles in the neighbourhood of these re-entrant corners.

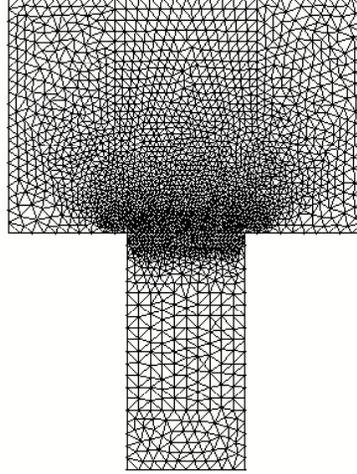


Figure 1.8: Mesh - Example 4

In Figures 1.9a and 1.9b we show the results of the convergence order tests for the error of the symmetric and non-symmetric hdG method by plotting in log-log scale the error as a function of the size of the mesh. For the reminder, $h = \max_{K \in \mathcal{T}_h} h_K$ denotes the maximum diameter of all triangles.

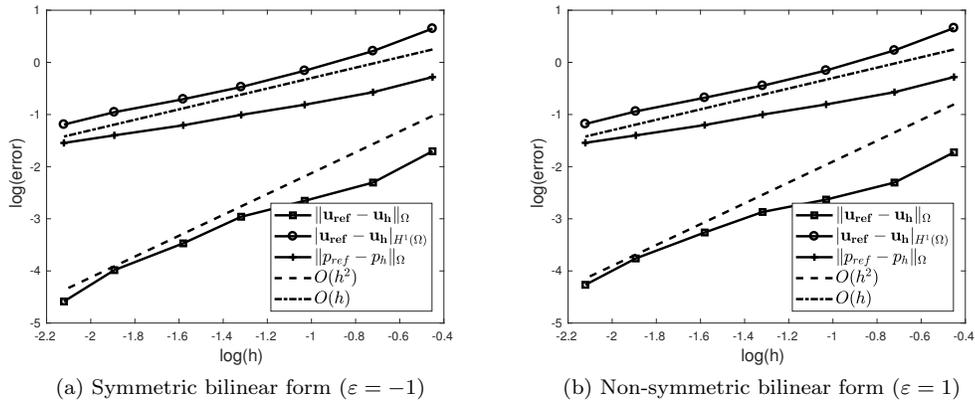


Figure 1.9: Error convergence of the hdG method with TVNF boundary conditions - Example 4

1.7 NVTF boundary conditions

In this section we discretise the Stokes problem with NVTF boundary conditions defined by (1.5). As it was in case of TVNF boundary conditions we start with presentation of the discrete problem in Section 1.7.1. Since proving the existence, uniqueness and error estimates of the discrete solution is similar to the case with TVNF boundary conditions, Sections 1.7.2 and 1.7.3 contain only proves that are different than those from Section 1.6. At the end, the same numerical experiments as in Section 1.6.4 validate the theory of the Section 1.7.

1.7.1 The discrete problem

In this section we consider other boundary conditions and thus we denote $\mathbf{V}_h := \mathbf{BDM}_{h,0}^k \times M_h^{k-1}$. Once again we multiply the first equation from (1.1) by a test function $\mathbf{v}_h \in \mathbf{BDM}_{h,0}^k$ and integrate by parts. This gives

$$(1.54) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \nu \partial_{\mathbf{n}} \mathbf{u} \mathbf{v}_h \, ds + \int_{\partial K} p (\mathbf{v}_h)_n \, ds \right).$$

Since the normal and tangential vectors are perpendicular ($\mathbf{n} \cdot \mathbf{t} = 0$) we can split (1.54) as

$$(1.55) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \sigma_{nt} (\mathbf{v}_h)_t \, ds - \int_{\partial K} \sigma_{nn} (\mathbf{v}_h)_n \, ds \right).$$

Using the definition of stress $\boldsymbol{\sigma} := \nu \nabla \mathbf{u} - p \mathbf{I} = (\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)^T$, we can rewrite the first equation of the Stokes problem (1.1) as $-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f}$. Since $\mathbf{f} \in L^2(\Omega)$, we know that $\nabla \cdot \boldsymbol{\sigma} = (\nabla \cdot \boldsymbol{\sigma}_1, \nabla \cdot \boldsymbol{\sigma}_2)^T$ and each $\boldsymbol{\sigma}_i \in H(\text{div}, \Omega)$, for $i = 1, 2$. That is why, $\boldsymbol{\sigma}_n$ is continuous across all interior edges. Moreover, since $\mathbf{v}_h \in \mathbf{BDM}_h^k$ and $H(\text{div})$ -conforming spaces preserve the continuity of the normal component of the velocity across the edges, then $(\mathbf{v}_h)_n$ is continuous across all interior edges. Then we can rewrite (1.14) as follows

$$(1.56) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \sigma_{nt} (\mathbf{v}_h)_t \, ds \right).$$

Moreover, since $\boldsymbol{\sigma}_n$ is continuous across all interior edges and NVTF boundary conditions (1.3), then

$$\sum_{K \in \mathcal{T}_h} \int_{\partial K} \sigma_{nt} \tilde{v}_h \, ds = \int_{\Gamma} g \tilde{v}_h \, ds \quad \forall \tilde{v}_h \in M_h^{k-1},$$

and we can subtract this from (1.56) to get

$$(1.57) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \sigma_{nt} ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \right) - \int_{\Gamma} g \tilde{v}_h \, ds.$$

Denoting $\tilde{u} = u_t$ on \mathcal{E}_h , then $(u_t - \tilde{u}) = \Phi^{k-1}(u_t - \tilde{u}) = 0$ on \mathcal{E}_h and applying NVTf boundary conditions (1.3) we can rewrite (1.57) as

$$(1.58) \quad - \int_{\Omega} \nabla \cdot (\nu \nabla \mathbf{u}) \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla p \cdot \mathbf{v}_h \, d\mathbf{x} = \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_K p \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \nu (\partial_n \mathbf{u})_t ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \pm \int_{\partial K} \nu (u_t - \tilde{u}) (\partial_n \mathbf{v}_h)_t \, ds + \nu \frac{\tau}{h_K} \int_{\partial K} \Phi^{k-1}(u_t - \tilde{u}) \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) \, ds - \int_{\Gamma} g \tilde{v}_h \, ds, \right)$$

where $\tau > 0$ is a stabilisation parameter. Again we have added terms that vanish for the exact solution, but will be relevant (i. e. non-zero) in the discrete formulation. The same idea is used in deriving discontinuous Galerkin methods. Hence, we define the velocity bilinear form $a : \mathbf{V}_h \times \mathbf{V}_h \rightarrow \mathbb{R}$ as

$$(1.59) \quad a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h)) := \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{w}_h : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_{\partial K} \nu (\partial_n \mathbf{w}_h)_t ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds + \varepsilon \int_{\partial K} \nu ((\mathbf{w}_h)_t - \tilde{w}_h) (\partial_n \mathbf{v}_h)_t \, ds + \nu \frac{\tau}{h_K} \int_{\partial K} \Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h) \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \right),$$

where $\varepsilon \in \{-1, 1\}$ and $\tau > 0$ is a stabilisation parameter and $b : \mathbf{V}_h \times Q_{h,0}^{k-1} \rightarrow \mathbb{R}$ as

$$(1.60) \quad b((\mathbf{v}_h, \tilde{v}_h), q_h) := - \sum_{K \in \mathcal{T}_h} \int_K q_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x}.$$

With these definitions we propose the hybrid discontinuous Galerkin (hdG) method for the NVTf boundary value problem (1.5):

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$

$$(1.61) \quad \begin{cases} a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g \tilde{v}_h \, ds \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) = 0 \end{cases}.$$

1.7.2 Well-posedness of the discrete problem

In this case once again, the discrete problem presented in (1.61) is a mixed formulation. Thus, to prove the existence and uniqueness of the solution we will use the Theorem 1.1.

We start by showing that semi-norm defined by (1.21) is either a norm for this choice of spaces.

Lemma 1.14 (hdG norm). *The semi-norm $||| \cdot |||$ defined by (1.21) is a norm on \mathbf{V}_h .*

Proof. Since $||| \cdot |||$ is a semi-norm, we only need to show that

$$||| (\mathbf{w}_h, \tilde{w}_h) ||| = 0 \Rightarrow \mathbf{w}_h = \mathbf{0} \text{ and } \tilde{w}_h = 0.$$

Let us suppose $(\mathbf{w}_h, \tilde{w}_h) \in \mathbf{V}_h$ and $||| (\mathbf{w}_h, \tilde{w}_h) ||| = 0$. Then $\nabla \mathbf{w}_h = 0$ in all $K \in \mathcal{T}_h$, and thus $\mathbf{w}_h|_K = \mathbf{C}_K$ for all $K \in \mathcal{T}_h$. Now, since \mathbf{w}_h belongs to $\mathbf{BDM}_{h,0}^k$, $(\mathbf{w}_h)_n$ is continuous in Ω . And as we have shown in proof of Lemma 1.7, \mathbf{w}_h is continuous in Ω , and thus $\mathbf{w}_h = \mathbf{C} \in \mathbb{R}^2$ in Ω . Finally,

$$(\mathbf{w}_h)_n = (\mathbf{C})_n = 0 \text{ on } \Gamma \Rightarrow \mathbf{w}_h = \mathbf{0} \text{ in } \Omega,$$

which, since $\tilde{w}_h = (\mathbf{w}_h)_t$ on every edge, finishes the proof. \square

The first assumption **B1** is fulfilled, because of the same reason as previously that these are discretisations of Hilbert spaces. To prove assumption **B2** we consider the following lemma that is the same as in previous section.

Lemma 1.15 (Continuity of bilinear forms). *There exists a constant $C > 0$ such that, for all $(\mathbf{w}, \tilde{w}), (\mathbf{v}, \tilde{v}) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\mathcal{E}_h)$ and $q \in L^2(\Omega)$, we have*

$$(1.62) \quad |a((\mathbf{w}, \tilde{w}), (\mathbf{v}, \tilde{v}))| \leq C ||| (\mathbf{w}, \tilde{w}) ||| ||| (\mathbf{v}, \tilde{v}) |||,$$

$$(1.63) \quad |b((\mathbf{w}, \tilde{w}), q)| \leq \sqrt{\frac{2}{\nu}} ||| (\mathbf{w}, \tilde{w}) ||| \|q\|_{\Omega}.$$

Proof. See proof of Lemma 1.8. \square

The next is the assumption **B3**. There is no difference in the proof of the following lemma.

Lemma 1.16 (Coercivity of the bilinear form a). *There exists $\alpha > 0$ such that for all $(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h$*

$$(1.64) \quad a((\mathbf{v}_h, \tilde{v}_h), (\mathbf{v}_h, \tilde{v}_h)) \geq \alpha ||| (\mathbf{v}_h, \tilde{v}_h) |||^2.$$

If $\varepsilon = -1$ in the definition (1.59), then this only holds under the additional hypothesis of τ being large enough. If $\varepsilon = 1$ in (1.59), this inequality holds for arbitrary $\tau > 0$.

Proof. See proof of Lemma 1.9. \square

Now we are only left with the last assumption **B4** to prove. Again, there is no difference in the proof of the following lemma.

Lemma 1.17 (Inf-sup condition for bilinear form b). *There exists $\beta > 0$ independent of h_K such that*

$$\sup_{(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h} \frac{b((\mathbf{v}_h, \tilde{v}_h), q_h)}{\|(\mathbf{v}_h, \tilde{v}_h)\|} \geq \frac{\beta}{\sqrt{\nu}} \|q_h\|_{\Omega} \quad \forall q_h \in Q_{h,0}^{k-1}.$$

Proof. Since the BDM projection defined in (1.8) and (1.9) preserves the normal-velocity boundary condition, the different boundary conditions do not affect the proof (see proof of Lemma 1.10). \square

Since all assumptions are fulfilled, according to Brezzi's theorem (Theorem 1.1) problem (1.61) is well-posed. Moreover, the discrete problem (1.61) has one more property which is consistency.

Lemma 1.18 (Consistency). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L_0^2(\Omega)$ be the solution of the problem (1.5) and $\tilde{u} = u_t$ on all edges of \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$ solves (1.61), then for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$ the following holds*

$$a((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h), q_h) + b((\mathbf{v}_h, \tilde{v}_h), p - p_h) = 0.$$

Proof. Since we consider the divergence-free Stokes problem, for all $q \in L^2(\Omega)$

$$(1.65) \quad b((\mathbf{u}, \tilde{u}), q) = - \sum_{K \in \mathcal{T}_h} \int_K q \nabla \cdot \mathbf{u} \, dx = 0.$$

Let us consider an arbitrary $(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{V}_h$. Using integration by parts for the left hand side of the first equation of the problem (1.61) and the assumption about \tilde{u} as a trace of \mathbf{u} , we get

$$(1.66) \quad \begin{aligned} a((\mathbf{u}, \tilde{u}), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p) &= \sum_{K \in \mathcal{T}_h} \left(\int_K \nu \nabla \mathbf{u} : \nabla \mathbf{v}_h \, dx - \int_K p \nabla \cdot \mathbf{v}_h \, dx \right. \\ &\quad - \nu \int_{\partial K} (\partial_{\mathbf{n}} \mathbf{u})_t ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \\ &\quad + \varepsilon \nu \int_{\partial K} (u_t - \tilde{u}) (\partial_{\mathbf{n}} \mathbf{v}_h)_t \, ds \\ &\quad \left. + \nu \frac{\tau}{h_K} \int_{\partial K} (\Phi^{k-1}(u_t - \tilde{u})) (\Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h)) \, ds \right) \\ &= \sum_{K \in \mathcal{T}_h} \left(-\nu \int_K \nabla \cdot (\nabla \mathbf{u}) \mathbf{v}_h \, dx + \int_K \nabla p \mathbf{v}_h \, dx \right) \\ &\quad + \int_{\Gamma} \sigma_{nn}(\mathbf{v}_h)_{\mathbf{n}} \, ds + \int_{\Gamma} \sigma_{nt} \tilde{v}_h \, ds. \end{aligned}$$

Since the NVTF boundary conditions (1.3), we arrive at

$$(1.67) \quad a((\mathbf{u}, \tilde{u}), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, dx + \int_{\Gamma} g \tilde{v}_h \, ds.$$

The proof is finished by adding (1.65) and (1.67). \square

1.7.3 Error analysis

We use once again the norm $||| \cdot |||_h$ defined in (1.41) and begin with the following version of Cea's lemma.

Lemma 1.19 (Cea's Lemma). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L_0^2(\Omega)$ be the solution of (1.5) and $\tilde{u} = u_t$ on all edges in \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$ solves (1.61), then there exists $C > 0$, independent of h and ν , such that*

$$(1.68) \quad |||(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)|||_h \leq C \inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}} |||(\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h)|||_h.$$

Proof. Since the BDM projection defined in (1.8) and (1.9) preserves the normal-velocity boundary condition, the different boundary conditions do not affect the proof (see proof of Lemma 1.12). \square

The following error estimate requires the same approach as before.

Lemma 1.20 (hdG error). *Let \mathcal{T}_h be a shape regular mesh and $k \geq 1$. Moreover, we assume $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^{k+1}(\mathcal{T}_h)]^2 \times H^k(\mathcal{T}_h)$ is the solution of (1.5) and $\tilde{u} = u_t$ on all edges in \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_{h,0}^{k-1}$ solves the mixed problem (1.61), then there exists $C > 0$, independent of h , such that*

$$(1.69) \quad |||(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)|||_h \leq Ch^k \left(\sqrt{\nu} \|\mathbf{u}\|_{H^{k+1}(\mathcal{T}_h)} + \frac{1}{\sqrt{\nu}} \|p\|_{H^k(\mathcal{T}_h)} \right).$$

Proof. Since the BDM projection defined in (1.8) and (1.9) preserves the normal-velocity boundary condition, the different boundary conditions do not affect the proof (see proof of Lemma 1.13). \square

1.7.4 Convergence validation

We consider the same examples as in Section 1.6.4. The computational domain for the first three test cases is the unit square $\Omega = (0, 1)^2$. We present the results for $k = 1$, the discrete space is given by $\mathbf{BDM}_{h,0}^1 \times M_h^0 \times Q_{h,0}^0$. We test both the symmetric method ($\varepsilon = -1$) and the non-symmetric method ($\varepsilon = 1$). For both cases we have followed the recommendation given in [Leh10, Section 2.5.2] and taken $\tau = 6$. All examples aim to verify the formulation with NVTF boundary conditions (1.61).

Example 1. We choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[100 (1 - \cos((1 - x)^2)) \sin(x^2) \sin(y^2) (1 - \cos((1 - y)^2)) \right], \quad p = \tan(xy).$$

In Figures 1.10a and 1.10b we show the results of the usual convergence order tests for the symmetric case and the non-symmetric case by plotting the error as a function of the size of the mesh using a log-log scale. We notice that they validate the theory from Section 1.7.3. In addition, an optimal h^2 convergence rate is observed for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

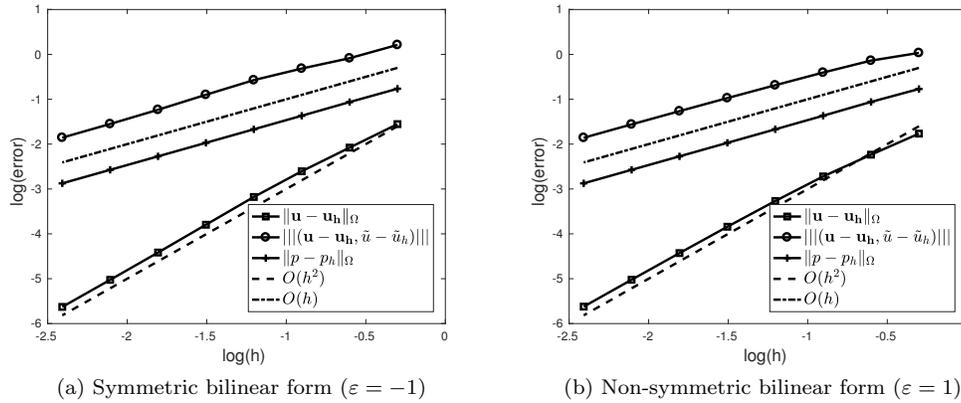


Figure 1.10: Error convergence of the hdG method with NVTf boundary conditions - Example 1

Example 2. We choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[x^2 (1 - x)^2 y^2 (1 - y)^2 \right], \quad p = x^2 - y^2.$$

The analytic solution is depicted in Figure 1.3.

The error convergence with respect to the size of the mesh is depicted on the log-log plots for the symmetric case and the non-symmetric case in Figures 1.11a and 1.11b, respectively. We can see that they not only validate the theory from Section 1.7.3, but also perform an optimal h^2 convergence rate for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

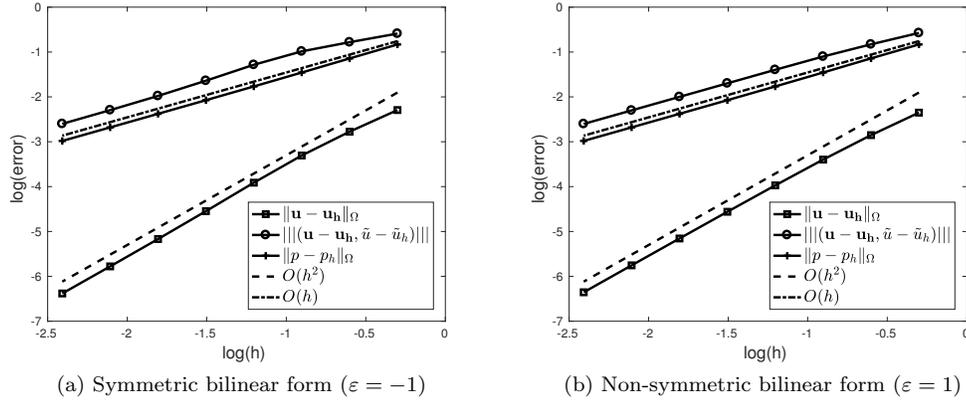


Figure 1.11: Error convergence of the hdG method with NVTf boundary conditions - Example 2

Example 3. We consider Poiseuille problem and we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = [4y(1 - y), 0]^T, \quad p = 4 - 8x.$$

The analytic solution is depicted in Figure 1.5.

The conclusion from Figures 1.12a and 1.12b remains the same as in previous examples.

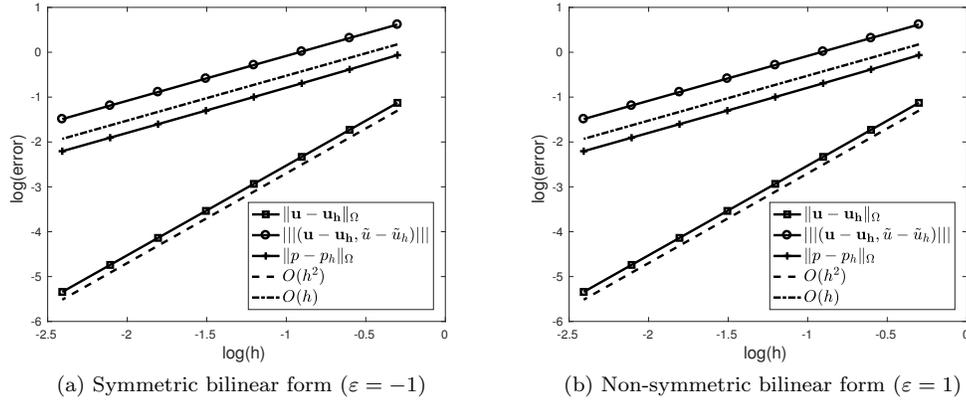


Figure 1.12: Error convergence of the hdG method with NVTf boundary conditions - Example 3

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions given by

$$(1.70) \quad \begin{cases} \mathbf{u}(x, y) = (4y(1 - y), 0)^T & \text{if } x = 0 \\ \sigma_{nn}(x, y) = 0, \quad u_t(x, y) = 0 & \text{if } x = 1.5 \\ \mathbf{u}(x, y) = (0, 0)^T & \text{otherwise} \end{cases} .$$

The solution of the symmetric hdG discretisation on a mesh containing 4 712 triangles is depicted in Figure 1.13. Since the analytic solution of this problem is unknown, we solved the

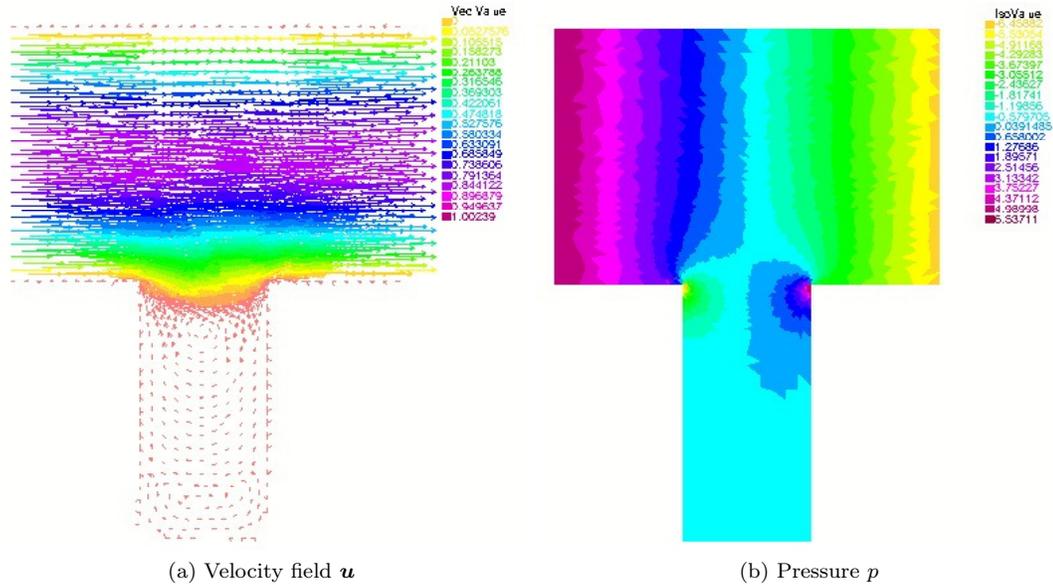


Figure 1.13: hdG solution with NVTF boundary conditions - Example 4

problem using the the lowest order Taylor-Hood discretisation on a mesh containing 2 046 150 triangles. The obtained solution is used as a reference solution $(\mathbf{u}_{ref}, p_{ref})$ to calculate the error of the hdG methods. Since the domain contains two re-entrant corners in points $(0.5, 0)$ and $(1, 0)$, we can expect some unstable behaviour at these points. To prevent it we refine the mesh in the neighbourhood of these re-entrant corners (see Figure 1.8).

In Figures 1.14a and 1.14b we show the results of the convergence order tests for the error of the symmetric and non-symmetric hdG method by plotting in log-log scale the error as a function of the size of the mesh ($h = \max_{K \in \mathcal{T}_h} h_K$).

1.8 Summary

In this chapter we introduced hdG methods for Stokes equations that naturally discretises non standard boundary value problems such as those with TVNF and NVTF boundary conditions. The interest of problem with these boundary conditions can also be found outside the domain decomposition idea. In fact, these boundary conditions (especially NVTF) can be seen as a "linearised" Tresca-like boundary conditions. As a matter of fact, the authors of [ABGS14] considered a Stokes equation with NVTF boundary conditions. This approach can be extended naturally to the case of incompressible, or nearly incompressible, elasticity. We will take advantage of this possibility in Chapter 4.

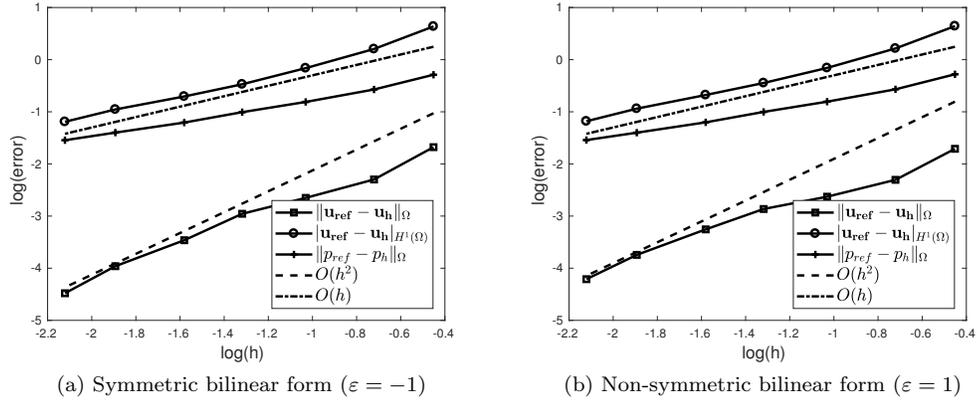


Figure 1.14: Error convergence of the hdG method with NVTf boundary conditions - Example 4

We proved the well-posedness and convergence with respect to the norm (1.41) of the discrete solution. We tested some numerical experiments in Sections 1.6.4 and 1.7.4 that validated the theory. In addition, we were able to observe the h^2 optimal convergence of the velocity error with respect to the L^2 norm. Unfortunately, the regularity of the boundary value problems with TVNF and NVTf boundary conditions is not well understood, which makes the use of duality arguments difficult. Thus, a formal proof of the h^2 order of convergence for $\|\mathbf{u} - \mathbf{u}_h\|_{\Omega}$ is lacking. This task shall be investigated more in the future.

Since the analysis of the hdG method at least in the case of symmetric formulation depends on the value of the stabilisation parameter, we can confirm that our numerical experiments performed the optimal convergence with the parameter value $\tau = 6$ as it was recommended in [LS16].

Finally, it is worth mentioning that we have not limited ourselves to showing that the discrete problem needed in domain decomposition algorithms is well-posed, but we have written the whole error analysis of the method, which is of interest in its own right.

Chapter 2

Stabilised hybrid discontinuous Galerkin methods

This chapter is motivated by the following observation. It has been observed in the stabilised finite element methods community that, for some cases, when increasing the order of polynomial approximation for the pressure space, the convergence order increases (although this can not be proved, in general). This increase in polynomial order requires the introduction of stabilising terms, since the finite element pairs used do not stability the inf-sup condition. Then, in this chapter we apply the stabilisation approach first proposed in [DB04] to the hybrid discontinuous Galerkin discretisation introduced in the last chapter. We start with the state of the art in Section 2.1. We do not present new notation, since we carry on with the one presented in Section 1.2. As before, we split the analysis in two parts. First we study the problem with TVNF boundary conditions in Section 2.2 and NVTF boundary conditions in Section 2.3. In each we prove stability and convergence results, and give numerical experiments confirming them.

2.1 State of the art

Stabilised finite element methods add terms to the usual weak formulation. This results in a saddle point problem being replaced by an elliptic one. It makes possible to avoid the inf-sup condition, and then allows the use of interpolation of equal order for velocity and pressure variables (see [FHS93]).

The term that a stabilised finite element method adds to the formulation can be either residual or non-residual. In [BP84] the first stabilised formulation was proposed. The authors added a mesh-dependent term penalising the gradient of the pressure to the formulation. Later, in [HFB86] this method was restricted and reinterpreted as a Petrov-Galerkin scheme leading

to the first consistent stabilised method. The authors of above papers continue analysis the method in further papers [BD88] and [HF87].

The development of residual methods was based on the choice of the stabilisation parameters. A critical study of these parameters have been presented in [FP92]. Another interesting study is the subject of [BBF93], where the equivalence between the Galerkin least squares method and the continuous Galerkin method enriched with bubble functions is shown.

Later, the optimal parameter for Stokes flow was characterised in [Sil94]. D. Silvester presents that there is a relation between optimal approximation and optimal conditioning. Furthermore, he proposed the mechanism for choosing the parameter in the case of the Galerkin least squares type methods. For a review of a different stabilised methods see [BBGS04] where the analysis of these methods and comparison between a different kinds of stabilised methods are also included.

Besides these different approaches, some of these methods present problems with consistency for low order interpolation spaces. The extension of [FHS93] by including the inertia terms in the momentum equation that solves this problem is presented in [BFT93]. Unfortunately, this approach leads to a bigger linear system as it entails more unknowns. A different possibility is presented in [BG04]. In there, the methods recast the pressure Poisson method as the result of the use of a discrete Laplacian.

Another alternative to enhance consistency for low order methods is to use a weak higher order derivative. The stabilisation operator that reconstruct the higher derivative information for implementations with polynomial lower order was introduced in [JCWS99]. Unfortunately, this approach, the same as the pressure-Poisson one, requires using the global L^2 projection.

Residual methods include unphysical couplings to the formulation, and modify all the entries of the stiffness matrix. Another type of methods are non-residual methods that do not preserve the consistency. The examples are the pressure gradient projection [CB00] and local pressure gradient stabilisation [BB01]. The idea in both cases is to project the pressure gradient to the velocity space globally and locally, respectively. For a unified analysis of different alternatives of pressure gradients projections, see [Bur08]. One issue that appears in most pressure gradient projection methods is the use of a second mesh. To avoid this complication, the local pressure gradient stabilisation has been also presented on the same mesh in [GMT08].

Methods that involve the fluctuation of the gradient of a pressure are not applicable in the case of discrete pressure spaces consisting of piecewise constant functions. Alternatively, instead of subtracting the pressure gradient, pressure jumps can be used as was presented in [SK90]. Since the requirement of the non standard data structures or specification of the mesh parameters may be limiting, the authors [DB04] present an approach that is based on polynomial-pressure-projection. This method works for low order of polynomials as was shown in [BDG06] and preserves symmetry of the original equation.

Another example of non-residual methods is the penalty finite element methods for incompressible fluids that are well presented in [HLB79]. The main aim is to uncouple pressure from

velocity. It is possible by relaxing the continuity equation. The authors present also a technique for stabilising the convection in the case of the Navier-Stokes equation.

Due to the above points we decided to use a non-residual method. Finally, due to the simplicity of their implementation, we have decided to analyse the polynomial-pressure-projection method from [DB04] for the hdG method introduced in previous chapter.

2.2 TVNF boundary conditions

We start with a stabilised hdG method for the Stokes problem with TVNF boundary conditions defined by (1.4). The formulation of the discrete problem is presented in Section 2.2.1. Sections 2.2.2 and 2.2.3 that discuss the well-posedness of the discrete problem and error estimates of the discrete solution respectively, use some of the results obtained in Section 1.6. We conclude with the numerical experiments that validate the theory of the Section 2.2.

2.2.1 The discrete problem

Our approach is to write the discrete problem with the same degree of polynomials for velocity and pressure spaces. In other words we want to use following spaces $\mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ instead of $\mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^{k-1}$ as it was in (1.20). To do this, we need the proper stabilisation term, because this choice of spaces does not guarantee inf-sup stability.

Let us define the continuous projection into the piecewise polynomial space with degree $k \geq 1$

$$(2.1) \quad \Psi^k : L^2(\Omega) \rightarrow Q_h^k$$

such that $\Psi^k|_K = \Psi_K^k$ for all $K \in \mathcal{T}_h$, which was defined in (1.10). Now we can write the discrete problem for the Stokes boundary value problem (1.4):

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$

$$(2.2) \quad \begin{cases} a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) &= \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g(\mathbf{v}_h)_n \, ds \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) - s(p_h, q_h) &= 0, \end{cases}$$

where

$$(2.3) \quad s(p_h, q_h) := \frac{1}{\nu} \int_{\Omega} (p_h - \Psi^{k-1} p_h) (q_h - \Psi^{k-1} q_h) \, d\mathbf{x}.$$

The above problem can be rewritten as follows:

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$

$$(2.4) \quad A((\mathbf{u}_h, \tilde{u}_h, p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g(\mathbf{v}_h)_n \, ds,$$

where

$$\begin{aligned} A((\mathbf{u}_h, \tilde{u}_h, p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) := & a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) \\ & + b((\mathbf{u}_h, \tilde{u}_h), q_h) - s(p_h, q_h). \end{aligned}$$

2.2.2 Well-posedness of the discrete problem

We use once again the norm $||| \cdot |||_h$ defined in (1.41). Since the order of the pressure space is the same as the one of the velocity space, the inf-sup condition for bilinear form b is no longer valid. Fortunately as we mentioned before, stabilised finite element methods replace a mixed formulation by elliptic problem. Thus, to prove the existence and uniqueness of the solution of (2.4) we will use the following generalised Lax-Milgram theorem [EG04, Chapter 2.1].

Theorem 2.1 (Generalised Lax-Milgram). *Let us assume that*

LM1 \mathbf{V}_h, Q_h^k are Hilbert spaces;

LM2 $A : (\mathbf{V}_h \times Q_h^k) \times (\mathbf{V}_h \times Q_h^k) \rightarrow \mathbb{R}$ is a continuous bilinear form,

LM3 $A : (\mathbf{V}_h \times Q_h^k) \times (\mathbf{V}_h \times Q_h^k) \rightarrow \mathbb{R}$ fulfills the following inf-sup conditions for $\alpha_1, \alpha_2 > 0$

$$\begin{aligned} \inf_{(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{V}_h \times Q_h^k} \sup_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^k} \frac{A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))}{|||(\mathbf{v}_h, \tilde{v}_h, q_h)|||_h |||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h} & \geq \alpha_1, \\ \inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{V}_h \times Q_h^k} \sup_{(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{V}_h \times Q_h^k} \frac{A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))}{|||(\mathbf{v}_h, \tilde{v}_h, q_h)|||_h |||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h} & \geq \alpha_2. \end{aligned}$$

If all of the above assumptions are fulfilled, then the discrete problem (2.4) has unique solution $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{V}_h \times Q_h^k$.

To prove the inf-sup condition for bilinear form A we first show the following weak inf-sup bound.

Lemma 2.1 (Weak inf-sup bound). *There exist constants $C_1, C_2 > 0$, independent of h_K and ν , such that*

$$(2.5) \quad \sup_{(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1}} \frac{b((\mathbf{v}_h, \tilde{v}_h), q_h)}{|||(\mathbf{v}_h, \tilde{v}_h)|||} \geq C_1 \|q_h\|_\Omega - C_2 \|q_h - \Psi^{k-1} q_h\|_\Omega \quad \forall q_h \in Q_h^k.$$

Proof. We consider an arbitrary $q_h \in Q_h^k$. Let $\tilde{\Omega}$ be a convex open set such that $\Omega \subset \tilde{\Omega}$, and let us consider following extension

$$\hat{q}_h := \begin{cases} q_h & \text{in } \Omega \\ 0 & \text{in } \tilde{\Omega} \setminus \Omega \end{cases}.$$

Then we define adjoint problem

$$\begin{cases} -\Delta\phi &= \hat{q}_h & \text{on } \tilde{\Omega} \\ \phi &= 0 & \text{on } \partial\Omega \end{cases}.$$

Since $\tilde{\Omega}$ is convex, then $\phi \in H^2(\tilde{\Omega})$. Then $\mathbf{w} := \nabla\phi|_{\Omega}$ belongs to $[H^1(\Omega)]^2$, and for $\tilde{w} := w_t$,

$$(2.6) \quad b((\mathbf{w}, \tilde{w}), q_h) = \|q_h\|_{\Omega}^2 \quad \forall q_h \in Q_h^k.$$

Then applying regularity results, see [BBF13, Section 1.2], we get

$$(2.7) \quad \|\mathbf{w}\|_{H^1(\Omega)} \leq \|\nabla\phi\|_{H^1(\tilde{\Omega})} \leq c_1 \|q_h\|_{\Omega}.$$

Let $(\mathbf{w}_h, \tilde{w}_h) := \mathbf{\Pi}(\mathbf{w})$, then since the BDM interpolation operator preserves the discrete divergence

$$b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), r_h) = 0 \quad \forall r_h \in Q_h^{k-1}.$$

Hence thanks to (2.6) and the continuity of b (Lemma 1.8)

$$\begin{aligned} b((\mathbf{w}_h, \tilde{w}_h), q_h) &= b((\mathbf{w}, \tilde{w}), q_h) - b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), q_h) \\ &= \|q_h\|_{\Omega}^2 - b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), q_h - \Psi^{k-1}q_h) \\ &\geq \|q_h\|_{\Omega}^2 - c_2 \sqrt{\sum_{K \in \mathcal{T}_h} |\mathbf{w}_h - \mathbf{w}|_{H^1(K)}^2} \|q_h - \Psi^{k-1}q_h\|_{\Omega}. \end{aligned}$$

Since we apply the BDM approximation (Lemma 1.4) and (2.7)

$$\begin{aligned} b((\mathbf{w}_h, \tilde{w}_h), q_h) &\geq \|q_h\|_{\Omega}^2 - c_2 c_3 |\mathbf{w}|_{H^1(\Omega)} \|q_h - \Psi^{k-1}q_h\|_{\Omega} \\ &\geq \left(\frac{1}{c_1} \|q_h\|_{\Omega} - c_2 c_3 \|q_h - \Psi^{k-1}q_h\|_{\Omega} \right) |\mathbf{w}|_{H^1(\Omega)}. \end{aligned}$$

Finally, thanks to the estimate of Fortin operator $\mathbf{\Pi}(\mathbf{w})$ in (1.30) we get

$$b((\mathbf{w}_h, \tilde{w}_h), q_h) \geq (C_1 \|q_h\|_{\Omega} - C_2 \|q_h - \Psi^{k-1}q_h\|_{\Omega}) \|(\mathbf{w}_h, \tilde{w}_h)\|,$$

where $C_1 = \frac{1}{C\sqrt{\nu}c_1}$ and $C_2 = \frac{c_2 c_3}{C\sqrt{\nu}}$ □

The first assumption **LM1** is fulfilled, because these are discretisations of Hilbert spaces. Let us now prove the assumption **LM2**.

Lemma 2.2 (Continuity of bilinear form A). *There exists a constant $C > 0$ such that, for all $(\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1}$ and $r_h, q_h \in Q_h^k$, we have*

$$(2.8) \quad |A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))| \leq C \|(\mathbf{w}_h, \tilde{w}_h, r_h)\|_h \|(\mathbf{v}_h, \tilde{v}_h, q_h)\|_h.$$

Proof. Thanks to the continuity of the bilinear forms (Lemma 1.8), Cauchy-Schwarz inequality and the fact that the projection is a bounded operator we get

$$\begin{aligned}
 |A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))| &\leq |a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h))| + |b((\mathbf{v}_h, \tilde{v}_h), r_h)| \\
 &\quad + |b((\mathbf{w}_h, \tilde{w}_h), q_h)| + \frac{1}{\nu} \|r_h - \Psi^{k-1}r_h\|_{\Omega} \|q_h - \Psi^{k-1}q_h\|_{\Omega} \\
 &\leq C_a ||| (\mathbf{w}_h, \tilde{w}_h) ||| ||| (\mathbf{v}_h, \tilde{v}_h) ||| + \sqrt{\frac{2}{\nu}} ||| (\mathbf{v}_h, \tilde{v}_h) ||| \|r_h\|_{\Omega} \\
 &\quad + \sqrt{\frac{2}{\nu}} ||| (\mathbf{w}_h, \tilde{w}_h) ||| \|q_h\|_{\Omega} + \frac{1}{\sqrt{\nu}} \|r_h\|_{\Omega} \frac{1}{\sqrt{\nu}} \|q_h\|_{\Omega}.
 \end{aligned}$$

Finally, we get (2.8) for $C = 4 \max\{C_a, \sqrt{2}\}$. \square

We only left with the last assumption **LM3** to prove. Since the proof is based on the coercivity of the bilinear form a , it is enough to show just the following lemma.

Lemma 2.3 (Inf-sup condition for bilinear form A). *There exists $\beta > 0$ independent of h_K such that for all $(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ we have*

$$(2.9) \quad \sup_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k} \frac{A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))}{||| (\mathbf{v}_h, \tilde{v}_h, q_h) |||_h} \geq \beta ||| (\mathbf{w}_h, \tilde{w}_h, r_h) |||_h.$$

Proof. Let $(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$. The idea of the proof is to construct an appropriate $(\mathbf{v}_h, \tilde{v}_h, q_h)$ such that

$$A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) \geq c ||| (\mathbf{w}_h, \tilde{w}_h, r_h) |||_h ||| (\mathbf{v}_h, \tilde{v}_h, q_h) |||_h.$$

Let $(\mathbf{v}_h, \tilde{v}_h, q_h) = (\mathbf{w}_h, \tilde{w}_h, -r_h)$, then using the coercivity of a (Lemma 1.9) we get

$$\begin{aligned}
 A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{w}_h, \tilde{w}_h, -r_h)) &= a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{w}_h, \tilde{w}_h)) + \int_{\Omega} (r_h - \Psi^{k-1}r_h)^2 \, dx \\
 &\geq \alpha_a ||| (\mathbf{w}_h, \tilde{w}_h) |||^2 + \|r_h - \Psi^{k-1}r_h\|_{\Omega}^2.
 \end{aligned}$$

Now, to control the second term associated with r_h and $\Psi^{k-1}r_h$, we apply Lemma 2.1 to conclude there exists $(\mathbf{t}_h, \tilde{t}_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1}$ such that

$$(2.10) \quad b((\mathbf{t}_h, \tilde{t}_h), r_h) \geq (C_1 \|r_h\|_{\Omega} - C_2 \|r_h - \Psi^{k-1}r_h\|_{\Omega}) ||| (\mathbf{t}_h, \tilde{t}_h) |||.$$

We choose this $(\mathbf{t}_h, \tilde{t}_h)$ such that it satisfies

$$(2.11) \quad ||| (\mathbf{t}_h, \tilde{t}_h) ||| = \frac{\sqrt{\alpha_a}}{C_a} \|r_h\|_{\Omega}$$

and we see that for $(\mathbf{v}_h, \tilde{v}_h, q_h) = (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)$, where $\lambda > 0$, we have

$$\begin{aligned} A((\mathbf{w}_h, \tilde{w}_h, r_h), (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)) &= -\lambda a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{t}_h, \tilde{t}_h)) + b((-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h), r_h) \\ &\geq -\lambda |a((\mathbf{w}_h, \tilde{w}_h), (\mathbf{t}_h, \tilde{t}_h))| + \lambda b((-\mathbf{t}_h, -\tilde{t}_h), r_h). \end{aligned}$$

Applying the continuity of a (Lemma 1.8) we get

$$A((\mathbf{w}_h, \tilde{w}_h, r_h), (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)) \geq -\lambda C_a |||(\mathbf{w}_h, \tilde{w}_h)||| |||(\mathbf{t}_h, \tilde{t}_h)||| + \lambda b((-\mathbf{t}_h, -\tilde{t}_h), r_h).$$

Using (2.10) and (2.11) we arrive at

$$b((-\mathbf{t}_h, -\tilde{t}_h), r_h) \geq C_1 \|r_h\|_\Omega^2 - C_2 \|r_h - \Psi^{k-1} r_h\|_\Omega \|r_h\|_\Omega.$$

Thus

$$\begin{aligned} A((\mathbf{w}_h, \tilde{w}_h, r_h), (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)) &\geq -\lambda C_a |||(\mathbf{w}_h, \tilde{w}_h)||| |||(\mathbf{t}_h, \tilde{t}_h)||| + \lambda C_1 \|r_h\|_\Omega^2 \\ &\quad - \lambda C_2 \|r_h - \Psi^{k-1} r_h\|_\Omega \|r_h\|_\Omega. \end{aligned}$$

Again using (2.11) we get

$$\begin{aligned} A((\mathbf{w}_h, \tilde{w}_h, r_h), (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)) &\geq -\lambda \sqrt{\alpha_a} |||(\mathbf{w}_h, \tilde{w}_h)||| \|r_h\|_\Omega + \lambda C_1 \|r_h\|_\Omega^2 \\ &\quad - \lambda C_2 \|r_h - \Psi^{k-1} r_h\|_\Omega \|r_h\|_\Omega. \end{aligned}$$

And now Young's inequality with constant $\varepsilon = \frac{C_1}{2}$ for the first and the last terms

$$\begin{aligned} A((\mathbf{w}_h, \tilde{w}_h, r_h), (-\lambda \mathbf{t}_h, -\lambda \tilde{t}_h, 0)) &\geq -\lambda \left(\frac{\alpha_a}{C_1} |||(\mathbf{w}_h, \tilde{w}_h)|||^2 + \frac{C_1}{4} \|r_h\|_\Omega^2 \right) + \lambda C_1 \|r_h\|_\Omega^2 \\ &\quad - \lambda \left(\frac{C_2}{C_1} \|r_h - \Psi^{k-1} r_h\|_\Omega + \frac{C_1}{4} \|r_h\|_\Omega \right). \end{aligned}$$

As a result, for $(\mathbf{v}_h, \tilde{v}_h, q_h) = (\mathbf{w}_h - \lambda \mathbf{t}_h, \tilde{w}_h - \lambda \tilde{t}_h, -r_h)$ we get

$$\begin{aligned} A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{w}_h - \lambda \mathbf{t}_h, \tilde{w}_h - \lambda \tilde{t}_h, -r_h)) &\geq \alpha_a \left(1 - \frac{\lambda}{C_1} \right) |||(\mathbf{w}_h, \tilde{w}_h)|||^2 + \frac{\lambda C_1}{2} \|r_h\|_{L^2(\Omega)}^2 \\ &\quad + \left(1 - \frac{\lambda C_2}{C_1} \right) \|r_h - \Psi^{k-1} r_h\|_\Omega^2. \end{aligned}$$

The choice $\hat{\lambda} = \min\{\frac{C_1}{2}, \frac{C_1}{2C_2}\}$ guarantees

$$1 - \frac{\hat{\lambda}}{C_1} \geq \frac{1}{2} \qquad 1 - \frac{\hat{\lambda} C_2}{C_1} \geq \frac{1}{2}.$$

Thus

$$A\left((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{w}_h - \hat{\lambda} \mathbf{t}_h, \tilde{t}_h - \hat{\lambda} \tilde{t}_h, -r_h)\right) \geq \frac{1}{2} \left(\alpha_a \|\| (\mathbf{w}_h, \tilde{w}_h) \|\|^2 + \hat{\lambda} C_1 \|r_h\|_{L^2(\Omega)}^2 + \|r_h - \Psi^{k-1} r_h\|_{\Omega}^2 \right).$$

Since $(a + b + c)^2 \leq 3(a^2 + b^2 + c^2)$ we get

$$\begin{aligned} A\left((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{w}_h - \hat{\lambda} \mathbf{t}_h, \tilde{w}_h - \hat{\lambda} \tilde{t}_h, -r_h)\right) &\geq \frac{1}{6} \left(\sqrt{\alpha_a} \|\| (\mathbf{w}_h, \tilde{w}_h) \|\| + \sqrt{\hat{\lambda} C_1} \|r_h\|_{L^2(\Omega)} \right. \\ &\quad \left. + \|r_h - \Psi^{k-1} r_h\|_{\Omega} \right)^2 \\ (2.12) \qquad \qquad \qquad &\geq C_4 \|\| (\mathbf{w}_h, \tilde{w}_h, r_h) \|\|^2. \end{aligned}$$

Finally using once again (2.11)

$$\begin{aligned} \|\| (\mathbf{v}_h, \tilde{v}_h, q_h) \|\|_h &= \|\| (\mathbf{w}_h - \hat{\lambda} \mathbf{t}_h, \tilde{w}_h - \hat{\lambda} \tilde{t}_h, -r_h) \|\|_h \\ &\leq \|\| (\mathbf{w}_h, \tilde{w}_h) \|\| + \hat{\lambda} \|\| (\mathbf{t}_h, \tilde{t}_h) \|\| + \|r_h\|_{\Omega} \\ &\leq \|\| (\mathbf{w}_h, \tilde{w}_h) \|\| + \frac{\hat{\lambda} \sqrt{\alpha_a}}{C_a} \|r_h\|_{\Omega} + \|r_h\|_{\Omega} \\ &\leq \|\| (\mathbf{w}_h, \tilde{w}_h) \|\| + \left(1 + \frac{\hat{\lambda} \sqrt{\alpha_a}}{C_a} \right) \|r_h\|_{\Omega} \\ (2.13) \qquad \qquad \qquad &\leq C_5 \|\| (\mathbf{w}_h, \tilde{w}_h, r_h) \|\|_h. \end{aligned}$$

Then collecting (2.12) and (2.13) we obtain (2.9) with $C := \frac{C_4}{C_5}$ □

Since all assumption are fulfilled, according to generalised Lax-Milgram theorem (Theorem 2.1), problem (2.4) is well-posed. Because of the stabilisation term we lost the consistency. However according to [BDG06], it should not be viewed as a serious flaw, as this consistency error can be bounded in an optimal way. The following result is the first step towards that goal.

Lemma 2.4 (Weak consistency). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\Omega)$ be the solution of the problem (1.4) and $\tilde{u} = u_t$ on all edges of \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ solves (2.4), then for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ the following holds*

$$(2.14) \qquad A((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) = s(p - p_h, q_h).$$

Proof. Since we have proved consistency in Lemma 1.11, then it is easy to see that the only remaining part is the added stabilisation term $s(p - p_h, q_h)$. □

2.2.3 Error analysis

We start by proving a variant of Cea's lemma [EG04, Lemma 2.28] for this stabilised Stokes problem.

Lemma 2.5 (Cea' Lemma). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L^2(\Omega)$ be a solution of (1.1) with TVNF boundary conditions (1.2), $\tilde{u} = u_t$ on all edges in \mathcal{E}_h , and $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ solves the discrete problem (2.4). Then there exists $C > 0$, independent of h and ν , such that*

$$(2.15) \quad \begin{aligned} \|\| (\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h) \|\|_h &\leq C \inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k} \|\| (\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h) \|\|_h \\ &+ \frac{C}{\sqrt{\nu}} \|p - \Psi^{k-1}p\|_{\Omega}. \end{aligned}$$

Proof. Let $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$. Using the triangle inequality we get

$$\begin{aligned} \|\| (\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h) \|\|_h &\leq \|\| (\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h) \|\|_h \\ &+ \|\| (\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h) \|\|_h. \end{aligned}$$

We estimate the second term by using the inf-sup condition of A (Lemma 2.3) and the weak consistency of A (Lemma 2.4). In fact, there exists $(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1}$ such that $\|\| (\mathbf{t}_h, \tilde{t}_h, r_h) \|\|_h = 1$ and

$$\begin{aligned} \|\| (\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h) \|\|_h &\leq \frac{1}{\beta_A} A((\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h), (\mathbf{t}_h, \tilde{t}_h, r_h)) \\ &\leq \frac{1}{\beta_A} A((\mathbf{v}_h - \mathbf{u}, \tilde{v}_h - \tilde{u}, q_h - p), (\mathbf{t}_h, \tilde{t}_h, r_h)) \\ &\quad + \frac{1}{\beta_A} A((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h), (\mathbf{t}_h, \tilde{t}_h, r_h)) \\ &\leq \frac{1}{\beta_A} (A((\mathbf{v}_h - \mathbf{u}, \tilde{v}_h - \tilde{u}, q_h - p), (\mathbf{t}_h, \tilde{t}_h, r_h)) + s(p, r_h)). \end{aligned}$$

Using Cauchy-Schwarz's inequality and the fact that the projection is a bounded operator we get

$$\begin{aligned} s(p, r_h) &= \frac{1}{\nu} \int_{\Omega} (p - \Psi^{k-1}p) (r_h - \Psi^{k-1}r_h) \, d\mathbf{x} \leq \frac{1}{\nu} \|p - \Psi^{k-1}p\|_{\Omega} \|r_h - \Psi^{k-1}r_h\|_{\Omega} \\ &\leq \frac{1}{\sqrt{\nu}} \|p - \Psi^{k-1}p\|_{\Omega} \frac{1}{\sqrt{\nu}} \|r_h\|_{\Omega}. \end{aligned}$$

Using $\|\| (\mathbf{t}_h, \tilde{t}_h, r_h) \|\|_h = 1$ we get

$$s(p, r_h) \leq \frac{1}{\sqrt{\nu}} \|p - \Psi^{k-1}p\|_{\Omega}.$$

Hence, using the boundedness of A (Lemma 2.2), we get

$$\|(\mathbf{v}_h - \mathbf{u}_h, \tilde{v}_h - \tilde{u}_h, q_h - p_h)\|_h \leq \frac{C_A}{\beta_A} \|(\mathbf{v}_h - \mathbf{u}, \tilde{v}_h - \tilde{u}, q_h - p)\|_h \frac{1}{\beta_A \sqrt{\nu}} \|p - \Psi^{k-1} p\|_\Omega.$$

Finally, we arrive at (2.15) with $C := \max\{1 + \frac{C_A}{\beta_A}, \frac{1}{\beta_A}\}$. \square

We finally estimate the hdG error.

Lemma 2.6 (hdG error). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times H^k(\Omega)$ be a solution of (1.1) with TVNF boundary conditions (1.2), $\tilde{u} = u_t$ on all edges in \mathcal{E}_h , and $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^k \times M_{h,0}^{k-1} \times Q_h^k$ solves the discrete problem (2.4). Then there exists $C > 0$, independent of h and ν , such that*

$$\|(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)\|_h \leq Ch^k \left(\sqrt{\nu} \|\mathbf{u}\|_{H^{k+1}(\mathcal{T}_h)} + \frac{1}{\sqrt{\nu}} \|p\|_{H^k(\mathcal{T}_h)} \right).$$

Proof. It is a combination of Lemmas 1.13 and 2.5 with the local L^2 -projection approximation (Lemma 1.5) \square

2.2.4 Convergence validation

We consider the same examples as in Section 1.6.4. The computational domain for three first test cases considered here is the unit square $\Omega = (0, 1)^2$. We present the results for $k = 1$, this is the discrete space is given by $\mathbf{BDM}_h^1 \times M_{h,0}^0 \times Q_h^1$. We test both the symmetric method ($\varepsilon = -1$) and the non-symmetric method ($\varepsilon = 1$). For all cases we have followed the recommendation given in [Leh10, Section 2.5.2] and taken $\tau = 6$. All examples aims at verifying the formulation with TVNF boundary conditions (2.2).

Example 1. Once again we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} [100 (1 - \cos((1 - x)^2)) \sin(x^2) \sin(y^2) (1 - \cos((1 - y)^2))], \quad p = \tan(xy).$$

In Figures 2.1a and 2.1b we show the results of the usual convergence order tests for the symmetric case and the non-symmetric case by plotting in log-log scale the error as a function of the size of the mesh. We notice that they validate the theory from Section 2.2.3. And again, an optimal h^2 convergence rate is observed for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

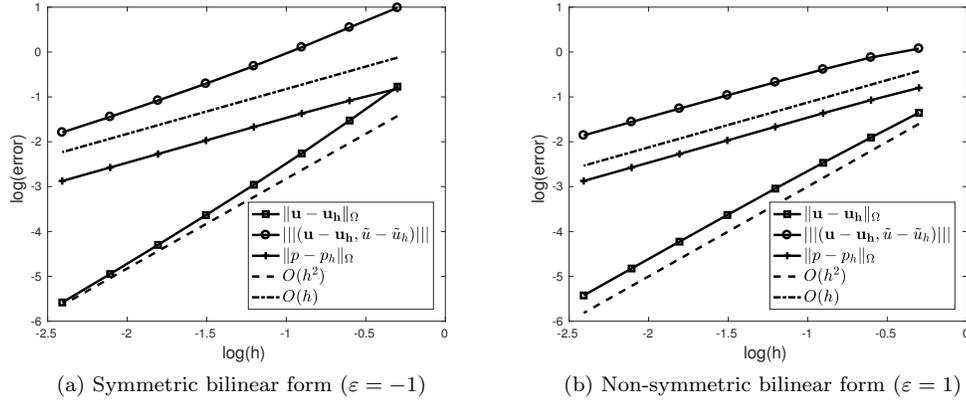


Figure 2.1: Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 1

Example 2. Once again we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[x^2 (1-x)^2 y^2 (1-y)^2 \right], \quad p = x^2 - y^2.$$

The analytic solution is depicted in Figure 1.3.

The error convergence with respect to the size of the mesh is depicted on the log-log plots for the symmetric case and the non-symmetric case in Figures 2.2a and 2.2b, respectively. We can see that they not only validate the theory from Section 2.2.3, but also perform an optimal h^2 convergence rate for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

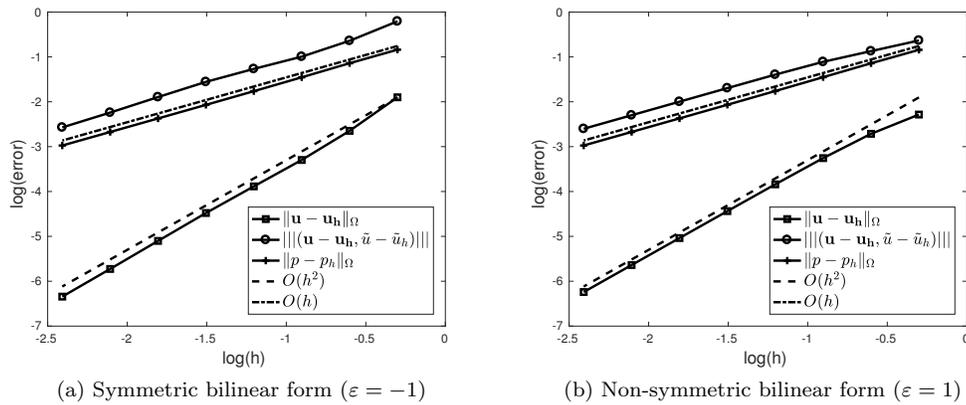


Figure 2.2: Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 2

Example 3. Once again we consider Poiseuille problem and we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = [4y(1 - y), 0]^T, \quad p = 4 - 8x.$$

The analytic solution is depicted in Figure 1.5.

The conclusion from Figures 2.3a and 2.3b remains the same as in previous examples.

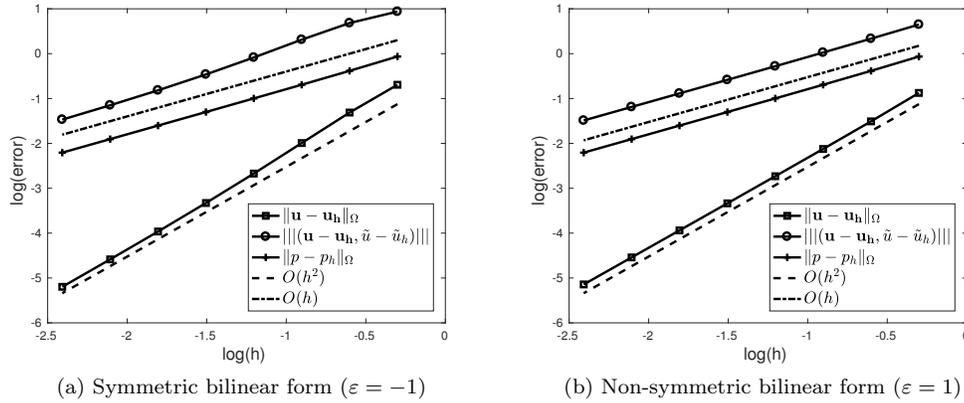


Figure 2.3: Error convergence of the stabilised hdG method with TVNF boundary conditions - Example 3

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions (1.53). The solution of the symmetric hdG discretisation on a mesh containing 4 712 triangles is depicted in Figure 2.4. Since the analytic solution of this problem is unknown, we solved the problem using the the lowest order Taylor-Hood discretisation on a mesh containing 2 046 150 triangles. The obtained solution is used as a reference solution ($\mathbf{u}_{ref}, p_{ref}$) to calculate the error of the hdG methods. Since the domain contains two re-entrant corners in points $(0.5, 0)$ and $(1, 0)$, we can expect some unstable behaviour at these points. To prevent it we refine the mesh in the neighbourhood of these re-entrant corners (see Figure 1.8).

In Figures 2.5a and 2.5b we show the results of the convergence order tests for the error of the symmetric and non-symmetric hdG method by plotting in log-log scale the error as a function of the size of the mesh.

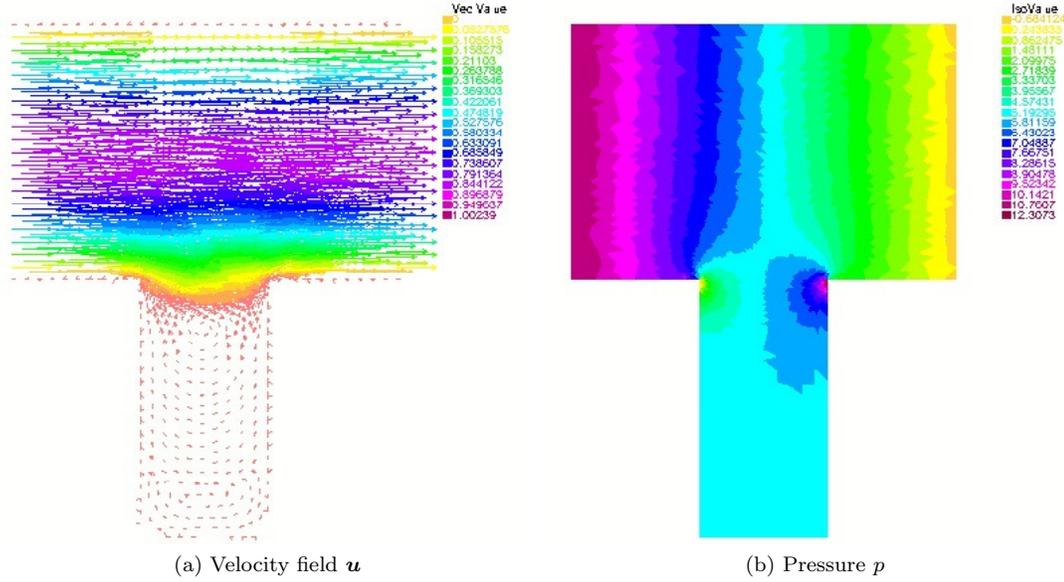


Figure 2.4: hdG solution with TVNF boundary conditions - Example 4

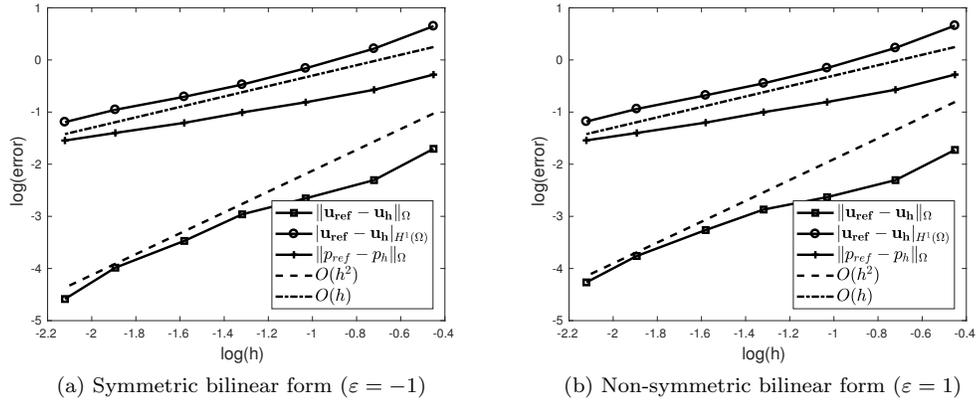


Figure 2.5: Error convergence of the hdG method with TVNF boundary conditions - Example 4

2.3 NVTF boundary conditions

This section discusses the stabilisation hdG method for the Stokes problem with NVTF boundary conditions defined by (1.5). As in the case of TVNF boundary conditions, we start with presentation of the discrete problem in Section 2.3.1. Since proving the existence, uniqueness and error estimates of the discrete solution is similar to the case with TVNF boundary conditions and the BDM projection defined in (1.8) and (1.9) preserves the normal-velocity boundary condition, Sections 2.3.2 and 2.3.3 contain only proofs that are different from those in Section 2.2. At the end, the theory is validated.

2.3.1 The discrete problem

Our approach is to write the discrete problem with the same degree of polynomials for velocity and pressure spaces. In other words we want to use following spaces $\mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^k$ instead of $\mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^{k-1}$ as it was in (1.61). To do this, we need the proper stabilisation term, because this choice of spaces does not guarantee inf-sup stability.

Let us define continuous projection into the piecewise polynomial space with degree $k \geq 1$

$$(2.16) \quad \Psi^k : L^2(\Omega) \rightarrow Q_{h,0}^k$$

such that $\Psi^k|_K = \Psi_K^k$ for all $K \in \mathcal{T}_h$, which was defined in (1.10). Now we can write the discrete problem for the Stokes boundary value problem (1.5):

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^k$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^k$

$$(2.17) \quad \begin{cases} a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g \tilde{v}_h \, ds \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) - s(p_h, q_h) = 0, \end{cases}$$

where once again

$$(2.18) \quad s(p_h, q_h) = \frac{1}{\nu} \int_{\Omega} (p_h - \Psi^{k-1} p_h) (q_h - \Psi^{k-1} q_h) \, d\mathbf{x}.$$

The above problem can be rewritten as follows:

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^k$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_{\mathbf{h},\mathbf{0}}^k \times M_h^{k-1} \times Q_{h,0}^k$

$$(2.19) \quad A((\mathbf{u}_h, \tilde{u}_h, p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} + \int_{\Gamma} g \tilde{v}_h \, ds,$$

where once again

$$\begin{aligned} A((\mathbf{u}_h, \tilde{u}_h, p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) &= a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) \\ &\quad + b((\mathbf{u}_h, \tilde{u}_h), q_h) - s(p_h, q_h). \end{aligned}$$

2.3.2 Well-posedness of the discrete problem

We use once again the norm $\|\cdot\|_h$ defined in (1.41). To prove the well-posedness of the discrete problem (2.19) we use the Theorem 2.1. We start with the following weak inf-sup bound.

Lemma 2.7 (Weak inf-sup bound). *There exist constants $C_1, C_2 > 0$, independent of h_K and*

ν , such that

$$(2.20) \quad \sup_{(\mathbf{v}_h, \tilde{v}_h) \in \mathbf{BDM}_{h,0}^k \times M_{h,0}^{k-1}} \frac{b((\mathbf{v}_h, \tilde{v}_h), q_h)}{|||(\mathbf{v}_h, \tilde{v}_h)|||} \geq C_1 \|q_h\|_\Omega - C_2 \|q_h - \Psi^{k-1} q_h\|_\Omega \quad \forall q_h \in Q_{h,0}^k.$$

Proof. We use the fact that the space for discrete pressure is a subspace of $L_0^2(\Omega)$. Thanks to [GR86, Section I.5.1], there exists $\mathbf{w} \in [H_0^1(\Omega)]^2$ and $\tilde{w} := w_t$ such that

$$(2.21) \quad b((\mathbf{w}, \tilde{w}), q_h) = \|q_h\|_\Omega^2 \quad \forall q_h \in Q_{h,0}^k$$

$$(2.22) \quad |\mathbf{w}|_{H^1(\Omega)} \leq c_1 \|q_h\|_\Omega.$$

Let $(\mathbf{w}_h, \tilde{w}_h) := \mathbf{\Pi}(\mathbf{w})$, then since the BDM interpolation operator preserves the discrete divergence

$$b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), r_h) = 0 \quad \forall r_h \in Q_{h,0}^{k-1}.$$

Hence thanks to (2.21) and the continuity of b (Lemma 1.15)

$$\begin{aligned} b((\mathbf{w}_h, \tilde{w}_h), q_h) &= b((\mathbf{w}, \tilde{w}), q_h) - b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), q_h) \\ &= \|q_h\|_\Omega^2 - b((\mathbf{w} - \mathbf{w}_h, \tilde{w} - \tilde{w}_h), q_h - \Psi^{k-1} q_h) \\ &\geq \|q_h\|_\Omega^2 - c_2 \sqrt{\sum_{K \in \mathcal{T}_h} |\mathbf{w}_h - \mathbf{w}|_{H^1(K)}^2} \|q_h - \Psi^{k-1} q_h\|_\Omega. \end{aligned}$$

Since we apply the BDM approximation (Lemma 1.4) and (2.22)

$$\begin{aligned} b((\mathbf{w}_h, \tilde{w}_h), q_h) &\geq \|q_h\|_\Omega^2 - c_2 c_3 |\mathbf{w}|_{H^1(\Omega)} \|q_h - \Psi^{k-1} q_h\|_\Omega \\ &\geq \left(\frac{1}{c_1} \|q_h\|_\Omega - c_2 c_3 \|q_h - \Psi^{k-1} q_h\|_\Omega \right) |\mathbf{w}|_{H^1(\Omega)}. \end{aligned}$$

Finally, thanks to the estimate of Fortin operator $\mathbf{\Pi}(\mathbf{w})$ in (1.30) we get

$$b((\mathbf{w}_h, \tilde{w}_h), q_h) \geq (C_1 \|q_h\|_\Omega - C_2 \|q_h - \Psi^{k-1} q_h\|_\Omega) |||(\mathbf{w}_h, \tilde{w}_h)|||,$$

where $C_1 = \frac{1}{C\sqrt{\nu}c_1}$ and $C_2 = \frac{c_2 c_3}{C\sqrt{\nu}}$ □

The first assumption **LM1** is fulfilled, because of the same reason as previously that these are discretisations of Hilbert spaces. To prove assumption **LM2** we consider the following lemma which proof is the same as previously.

Lemma 2.8 (Continuity of bilinear form A). *There exists a constant $C > 0$ such that, for all $(\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h) \in \mathbf{BDM}_{h,0}^k \times M_{h,0}^{k-1}$ and $r_h, q_h \in Q_{h,0}^k$, we have*

$$(2.23) \quad |A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))| \leq C |||(\mathbf{w}_h, \tilde{w}_h, r_h)|||_h |||(\mathbf{v}_h, \tilde{v}_h, q_h)|||_h.$$

Proof. See the proof of Lemma 2.2 □

Now we only left with the last assumption **LM3** to prove.

Lemma 2.9 (Inf-sup condition for bilinear form A). *There exists $\beta > 0$ independent of h_K such that for all $(\mathbf{w}_h, \tilde{w}_h, r_h) \in \mathbf{BDM}_{h,0}^k \times M_h^{k-1} \times Q_{h,0}^k$ we have*

$$(2.24) \quad \sup_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_{h,0}^k \times M_h^{k-1} \times Q_{h,0}^k} \frac{A((\mathbf{w}_h, \tilde{w}_h, r_h), (\mathbf{v}_h, \tilde{v}_h, q_h))}{\|(\mathbf{v}_h, \tilde{v}_h, q_h)\|_h} \geq \beta \|(\mathbf{w}_h, \tilde{w}_h, r_h)\|_h.$$

Proof. See the proof of Lemma 2.3. The only difference is in applying weak inf-sup bound from this section (Lemma 2.7 instead of Lemma 2.1). \square

Since all assumption are fulfilled, according to generalised Lax-Milgram theorem (Theorem 2.1), problem (2.19) is well-posed. The next result is the analogous of Lemma 2.4.

Lemma 2.10 (Weak consistency). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L_0^2(\Omega)$ be the solution of the problem (1.4) and $\tilde{u} = u_t$ on all edges of \mathcal{E}_h . If $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_{h,0}^k \times M_h^{k-1} \times Q_{h,0}^k$ solves (2.19), then for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^k \times M_h^{k-1} \times Q_{h,0}^k$ the following holds*

$$(2.25) \quad A((\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h), (\mathbf{v}_h, \tilde{v}_h, q_h)) = s(p - p_h, q_h).$$

Proof. Since we have proved consistency in Lemma 1.11, then it is easy to see that the only remaining part is the added stabilisation term $s(p - p_h, q_h)$. \square

2.3.3 Error analysis

We start by proving a variant of Cea's lemma for this stabilised Stokes problem.

Lemma 2.11 (Cea' Lemma). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times L_0^2(\Omega)$ be a solution of (1.1) with NVTF boundary conditions (1.3), $\tilde{u} = u_t$ on all edges in \mathcal{E}_h , and $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_{h,0}^k \times M_h^{k-1} \times Q_{h,0}^k$ solves the discrete problem (2.19). Then there exists $C > 0$, independent of h and ν , such that*

$$(2.26) \quad \begin{aligned} \|(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)\|_h \leq & C \inf_{(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_{h,0}^k \times M_h^{k-1} \times Q_{h,0}^k} \|(\mathbf{u} - \mathbf{v}_h, \tilde{u} - \tilde{v}_h, p - q_h)\|_h \\ & + \frac{C}{\sqrt{\nu}} \|p - \Psi^{k-1}p\|_\Omega. \end{aligned}$$

Proof. See the proof of Lemma 2.5. \square

We finally estimate the hdG error.

Lemma 2.12 (hdG error). *Let $(\mathbf{u}, p) \in [H^1(\Omega) \cap H^2(\mathcal{T}_h)]^2 \times H^k(\Omega)$ be a solution of (1.1) with NVTF boundary conditions (1.3), $\tilde{u} = u_t$ on all edges in \mathcal{E}_h , and $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_{h,0}^k \times$*

$M_h^{k-1} \times Q_{h,0}^k$ solves the discrete problem (2.19). Then there exists $C > 0$, independent of h and ν , such that

$$\|(\mathbf{u} - \mathbf{u}_h, \tilde{u} - \tilde{u}_h, p - p_h)\|_h \leq Ch^k \left(\sqrt{\nu} \|\mathbf{u}\|_{H^{k+1}(\mathcal{T}_h)} + \frac{1}{\sqrt{\nu}} \|p\|_{H^k(\mathcal{T}_h)} \right).$$

Proof. It is a combination of Lemmas 1.20 and 2.11 with the local L^2 -projection approximation (Lemma 1.5) □

2.3.4 Convergence validation

We consider the same examples as in Section 1.6.4. The computational domain for three first test cases considered here is the unit square $\Omega = (0, 1)^2$. We present the results for $k = 1$, this is, the discrete space is given by $BDM_{h,0}^1 \times M_h^0 \times Q_{h,0}^1$. We test both the symmetric method ($\varepsilon = -1$) and the non-symmetric method ($\varepsilon = 1$). For all cases we have followed the recommendation given in [Leh10, Section 2.5.2] and taken $\tau = 6$. All examples aims at verifying the formulation with NVTf boundary conditions (2.17).

Example 1. Once again we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} [100 (1 - \cos((1 - x)^2)) \sin(x^2) \sin(y^2) (1 - \cos((1 - y)^2))], \quad p = \tan(xy).$$

In Figures 2.6a and 2.6b we show the results of the usual convergence order tests for the symmetric case and the non-symmetric case by plotting the error as a function of the size of the mesh using a log-log scale. We notice that they validate the theory from Section 2.3.3. And again, an optimal h^2 convergence rate is observed for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

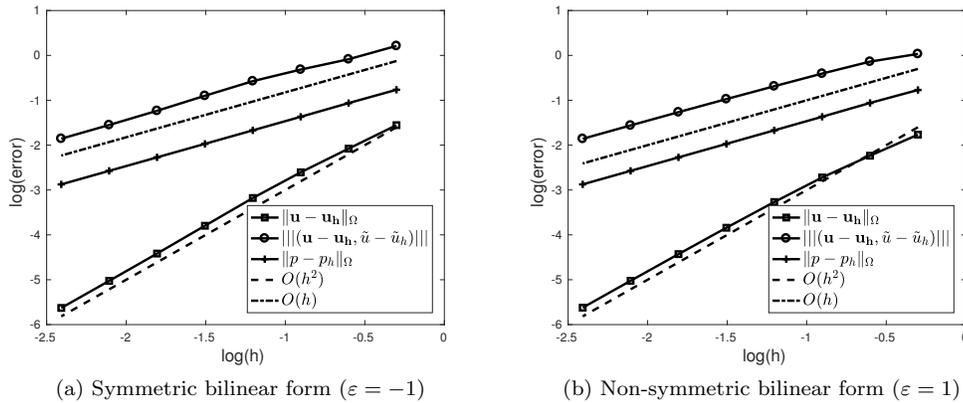


Figure 2.6: Error convergence of the stabilised hdG method with NVTf boundary conditions - Example 1

Example 2. Once again we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[x^2 (1-x)^2 y^2 (1-y)^2 \right], \quad p = x^2 - y^2.$$

The analytic solution is depicted in Figure 1.3.

The error convergence with respect to the size of the mesh is depicted on the log-log plots for the symmetric case and the non-symmetric case in Figures 2.7a and 2.7b, respectively. We can see that they not only validate the theory from Section 2.3.3, but also perform an optimal h^2 convergence rate for $\|\mathbf{u} - \mathbf{u}_h\|_\Omega$.

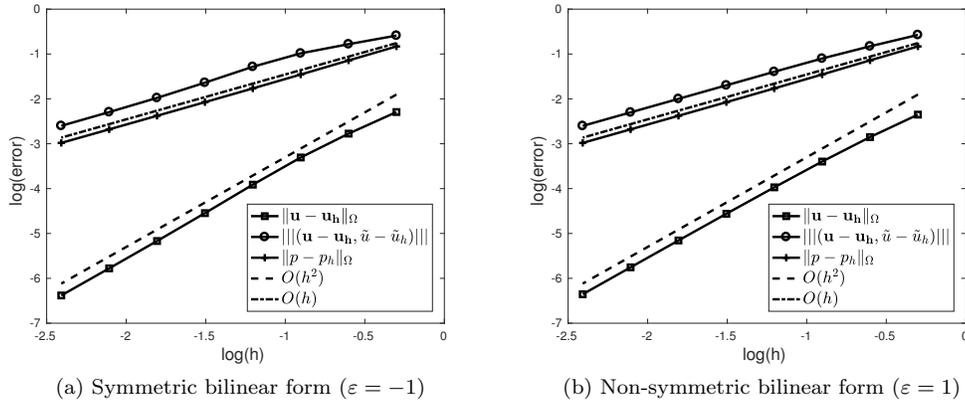


Figure 2.7: Error convergence of the stabilised hdG method with NVTF boundary conditions - Example 2

Example 3. Once again we consider Poiseuille problem and we choose the right hand side \mathbf{f} and the boundary condition g such that the exact solution is given by

$$\mathbf{u} = [4y(1-y), 0]^T, \quad p = 4 - 8x.$$

The analytic solution is depicted in Figure 1.5.

The conclusion from Figures 2.8a and 2.8b remains the same as in previous examples.

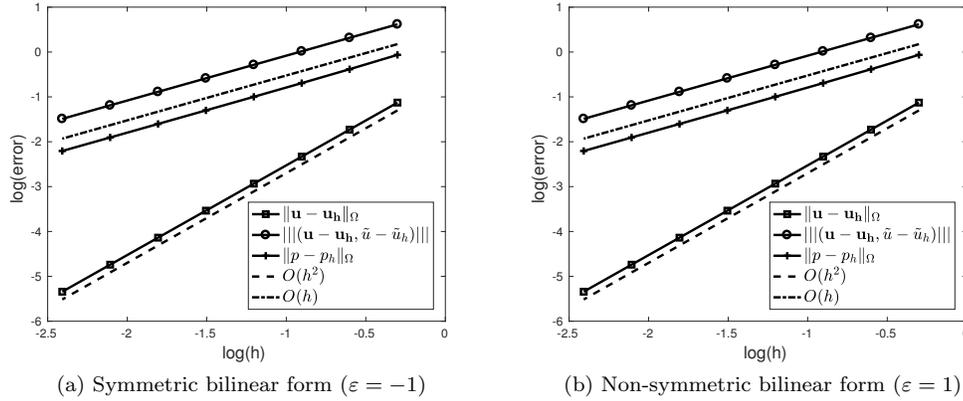


Figure 2.8: Error convergence of the stabilised hdG method with NVTF boundary conditions - Example 3

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions (1.70). The solution of the symmetric hdG discretisation on a mesh containing 4 712 triangles is depicted in Figure 2.9. Since the analytic solution of this

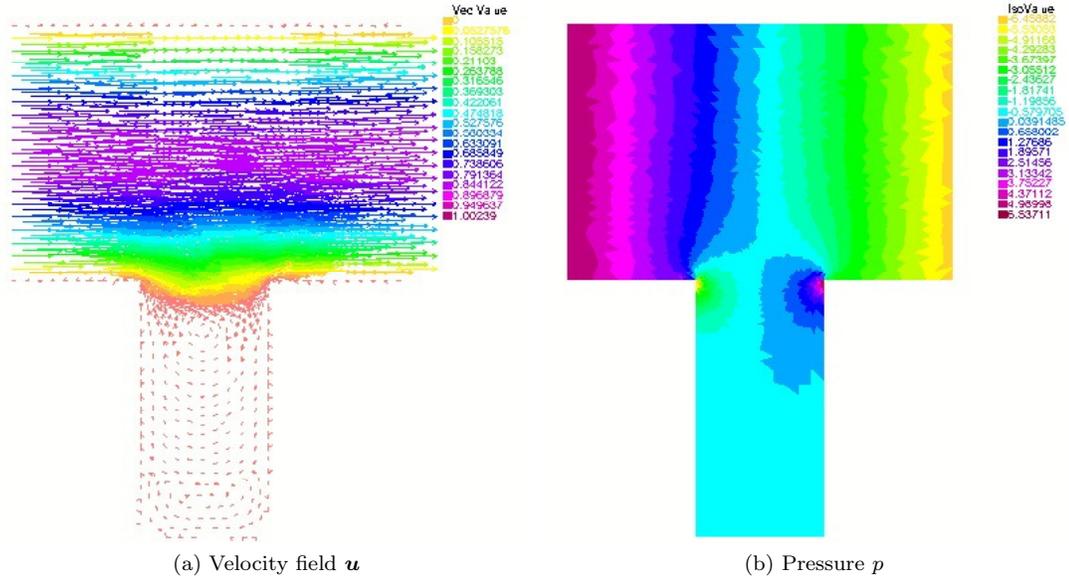


Figure 2.9: hdG solution with NVTF boundary conditions - Example 4

problem is unknown, we solved the problem using the the lowest order Taylor-Hood discretisation on a mesh containing 2 046 150 triangles. The obtained solution is used as a reference solution ($\mathbf{u}_{ref}, p_{ref}$) to calculate the error of the hdG methods. Since the domain contains two re-entrant corners in points $(0.5, 0)$ and $(1, 0)$, we can expect some unstable behaviour at these points. To prevent it we refine the mesh in the neighbourhood of these re-entrant corners (see

Figure 1.8).

In Figures 2.10a and 2.10b we show the results of the convergence order tests for the error of the symmetric and non-symmetric hdG method by plotting in log-log scale the error as a function of the size of the mesh.

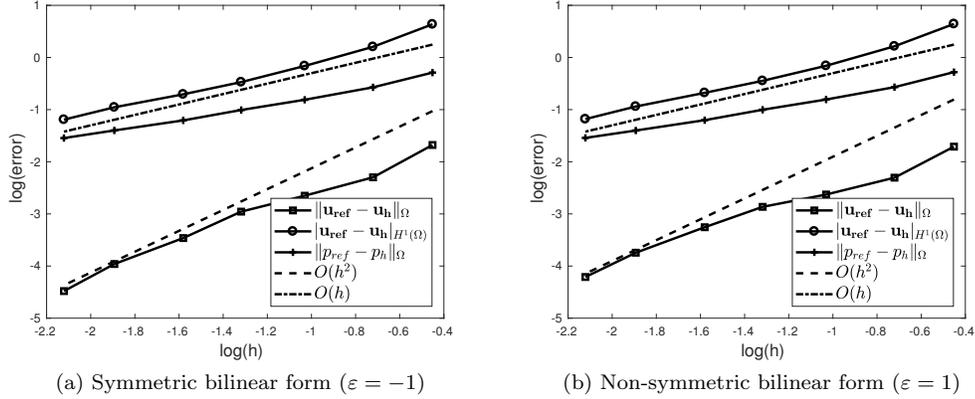


Figure 2.10: Error convergence of the hdG method with NVTF boundary conditions - Example 4

2.4 Summary

The idea of stabilisation of the finite element methods for Stokes problem from [DB04] is presented in the case of continuous pressure space. We introduced stabilised hdG methods for Stokes equations with TVNF and NVTF boundary conditions that are associated with the discontinuous pressure space. Fortunately, thanks to continuity of the normal component of the velocity on element interfaces we were able to prove the well-posedness and convergence with respect to the norm (1.41).

Furthermore, we tested the same examples as in Chapter 1 that demonstrated good stability of the new method. Unfortunately, we did not observe the improvement of the convergence in the low order case. Future testing using higher order discretisations is needed to assess whether this approach provides an increase of the convergence rate for the pressure. This approach can be also applied to other discontinuous Galerkin methods that deal with Stokes or nearly incompressible elasticity problems. The numerical tests with higher order of polynomials for discontinuous finite methods is interest for further research to look for the improvement of the convergence.

Chapter 3

Domain decomposition methods

We proposed the hdG method with projection that allows us to decrease the number of degrees of freedom. However, in the case of big problems it still leads to linear systems that are too big for direct solvers. That is why, we present in this chapter more advance techniques such as domain decomposition methods that allow us to treat this problem. We begin with a literature review in Section 3.1 and a short introduction to the domain decomposition methods in Section 3.2. In Section 3.3, the Additive Schwarz methods are defined at the algebraic level. Since partition of unity plays an important role in the preconditioners presented by us, in Section 3.4 we give more details about it. Section 3.5 contains the numerical simulation results. Finally, we summarise this chapter in Section 3.6.

3.1 State of the art

Let us start with a historical overview of domain decomposition methods. The classical Schwarz method was introduced in [Sch70]. It has been shown that this method is equivalent to a block Gauss Seidel type iteration where each of blocks corresponds to one of the subdomains. With the advent of the parallel computers these methods regained a lot of interest. P.-L. Lions introduced in [Lio88] a modification of the original Schwarz method that yields a fully parallel algorithm whose algebraic counterpart is equivalent to a block-Jacobi method.

Since the seminal works of Lions, there is a rich of literature on domain decomposition methods. We can mention the books on the topic that contain an overview of the state of the art at that time. They are quite complementary in scope and approach. The first [SBG96] presents the methods from an algebraic point of view and more oriented on different applications. The second [QV99] is more oriented on the continuous analysis of these methods as applied directly to PDEs.

The Restricted Additive Schwarz (RAS) preconditioner was introduced by X.-Ch. Cai and

M. Sarkis in [CS99]. The analysis of this method has been presented in [CDS03] and the convergence theory was presented in [FS01]. X.-Ch. Cai and M. Sarkis in [CS99] proposed also a Restricted Multiplicative Schwarz (RMS) preconditioner whose convergence has been analysed in [NS02]. There is also a continuous interpretation at the matrix level of the RAS in [EG03] that helps to explain why this method converges faster than AS.

Another reference book by Toselli and Widlund [TW05] discusses the domain decomposition methods in the context of finite element discretisation. Moreover, they present an overview of the preconditioning approach of these methods for a variety continuum mechanics problems.

Another class of methods are the Optimized Schwarz methods introduced in [Lio90] that use more effective transmission conditions at the interfaces between the subdomains than the usual Dirichlet or Neumann boundary conditions. Again a rich literature can be found on this topic, with applications to various domains and equations. The effective application of these methods as preconditioners are shown in [SCGT07]. These new methods are known as optimized version of the previous e.g. Optimized RAS (ORAS), Optimized MS (OMS) and Optimized AS (OAS) preconditioners.

The newest book from V. Dolean, P. Jolivet and F. Nataf [DJN15], in addition to [TW05], includes also the optimized methods, new advances in coarse spaces and provides implementations in an open-source finite element software.

For systems of partial differential equations such as elasticity or Stokes problems, it has been envisioned that normal velocity-tangential flux (NVTF) or tangential velocity-normal flux (TVNF) interface conditions should be superior to the pure velocity (Dirichlet like) or pure stress (Neumann like) interface conditions, see [DJN15, Section 6.6] and references therein.

In [GR06] reviewed most non-overlapping domain decomposition methods. The authors of this paper presented different computational frameworks and obtained the above non standard interface conditions by considering symmetric linear elasticity problem. Using symbolic techniques known as the Smith factorization [Smi61] for the Stokes equation the authors of [DNR09] obtained also such interface conditions. The further analysis for Cauchy-Navier and Oseen equation [CDNQ13], and scalar symmetric positive definite second order, advection-diffusion plate and shell problems [CDNQ12] validated this choice of non standard interface conditions. Similar attempts to derive more intrinsic interface conditions to the nature of the equation to solve were derived in [DN06] for the Euler system.

The difficulties surrounding the implementations of these interface conditions mean that historical numerical tests were restricted to decompositions where boundaries of subdomains are rectilinear. This is so the normal to the interface is easy to define. The underlying domain decomposition method was a Schur complement method. That is mainly the reason we have considered and analysed a specific hdG method where this kind of degrees of freedom are naturally present.

The combination of the discretisation and domain decomposition methods is meant to provide

a competitive solving strategy for this kind of partial differential equation system. A different, but somewhat related, approach can be found in [AdDBM⁺14] where a dG type discretisation is coupled to a discrete Helmholtz decomposition to propose new preconditioners.

3.2 Overlapping Schwarz methods

The very first domain decomposition method was due to H. Schwarz, as he was interested in proving existence and uniqueness of the solution of the Poisson equation with given function f on the generally shaped domain Ω presented on Figure 3.1a

$$(3.1) \quad \begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega. \end{cases}$$

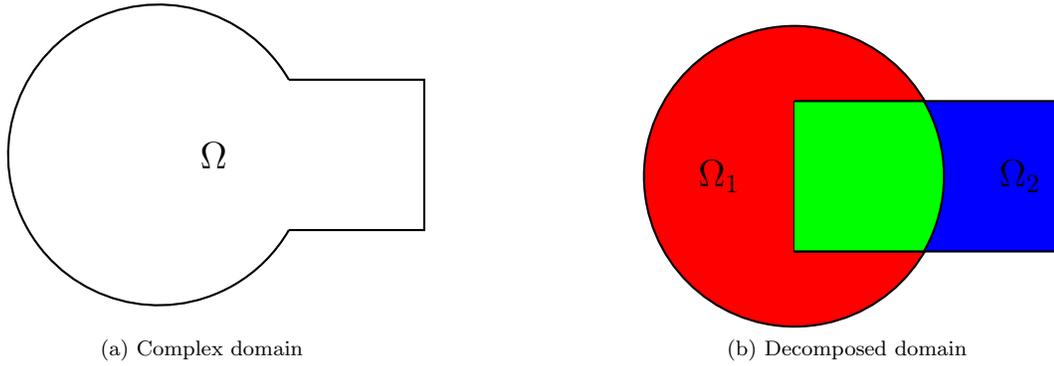


Figure 3.1: Original geometry of domain

To solve problem (3.1), H. Schwarz considered first a domain which is the union of circle (subdomain Ω_1) and rectangle (subdomain Ω_2) as it is shown on Figure 3.1b on which the solution was available by using Fourier series. He used then an iterative method to prove the existence of the solution on the whole domain. This method is commonly known as the classical Schwarz method and can be described as follows: For an initial guess u_2^0 one solves

$$(3.2) \quad \begin{cases} -\Delta u_1^{n+1} = f & \text{in } \Omega_1 \\ u_1^{n+1} = 0 & \text{on } \partial\Omega \cap \partial\Omega_1 \\ u_1^{n+1} = u_2^n & \text{on } \partial\Omega_1 \setminus \partial\Omega \end{cases} \quad \begin{cases} -\Delta u_2^{n+1} = f & \text{in } \Omega_2 \\ u_2^{n+1} = 0 & \text{on } \partial\Omega \cap \partial\Omega_2 \\ u_2^{n+1} = u_1^{n+1} & \text{on } \partial\Omega_2 \setminus \partial\Omega. \end{cases}$$

It is worth noting that the solution on subdomain Ω_2 at $n+1$ iteration depends on the solution on subdomain Ω_1 . That is why this algorithm is not parallel. Over a century later P.-L. Lions modified the classical Schwarz method and proposed fully parallel algorithm that for given pair

(u_1^0, u_2^0) solves

$$(3.3) \quad \begin{cases} -\Delta u_1^{n+1} = f & \text{in } \Omega_1 \\ u_1^{n+1} = 0 & \text{on } \partial\Omega \cap \partial\Omega_1 \\ u_1^{n+1} = u_2^n & \text{on } \partial\Omega_1 \setminus \partial\Omega \end{cases} \quad \begin{cases} -\Delta u_2^{n+1} = f & \text{in } \Omega_2 \\ u_2^{n+1} = 0 & \text{on } \partial\Omega \cap \partial\Omega_2 \\ u_2^{n+1} = u_1^n & \text{on } \partial\Omega_2 \setminus \partial\Omega. \end{cases}$$

As a general rule this kind of method can be generalized for a boundary value problem defined by the partial differential operator \mathcal{L} that leads to a well-posed problem

$$(3.4) \quad \begin{cases} \mathcal{L}u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega. \end{cases}$$

If we consider again the two subdomains decomposition $\Omega = \Omega_1 \cup \Omega_2$, then for $i = 1, 2$, we define the extension operator

$$E_i(\omega_i) = \begin{cases} \omega_i & \text{in } \Omega_i \\ 0 & \text{otherwise.} \end{cases}$$

Furthermore, for $i = 1, 2$, we define the partition of unity as a non-negative function $\chi_i : \Omega \rightarrow \mathbb{R}$ such that $\chi_i = 0$ on $\partial\Omega_i \setminus \partial\Omega$ and $\omega = E_1(\chi_1\omega|_{\Omega_1}) + E_2(\chi_2\omega|_{\Omega_2})$. The example of such partition of unity for the complex geometry is depicted in Figure 3.2.

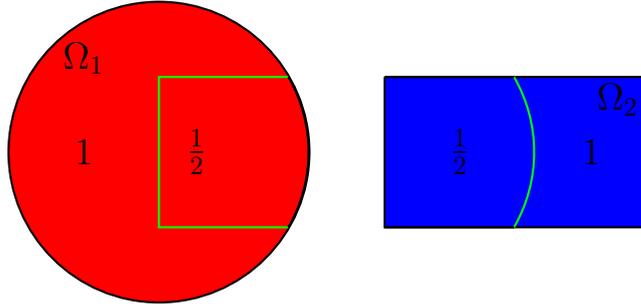


Figure 3.2: Partition of unity

For such definition the generalized Schwarz method for given initial value u^0 solves in parallel local problems

$$(3.5) \quad \begin{cases} \mathcal{L}v_i^{n+1} = f & \text{in } \Omega_i \\ v_i^{n+1} = 0 & \text{on } \partial\Omega \cap \partial\Omega_i \\ v_i^{n+1} = u_{3-i}^n & \text{on } \partial\Omega_i \setminus \partial\Omega \end{cases}$$

and then glues the global solution

$$u^{n+1} = E_1(\chi_1 v_1^{n+1}) + E_2(\chi_2 v_2^{n+1}).$$

In Figure 3.2 we can see that the decomposition created the artificial boundary, named interface (green lines). This kind of boundaries we define as $\bigcup_{i=1}^N \partial\Omega_i \setminus \partial\Omega$, for N subdomains. Since they are not associated with the physical boundary, we can choose any kind of interface conditions that lead to a local well-posed problem and a convergent algorithms. In the case of (3.2), (3.3) and (3.5) there are Dirichlet interface conditions. Although, it was shown in [Lio90] that Robin kind of interface conditions can bring the improvement in the convergence of an iterative method for Poisson equation. Extension of the Lions' algorithm for the Helmholtz problem is presented in [Des93] and for time-harmonic Maxwell equation in [DJR92].

The interesting remark is that if we consider non-overlapping decomposition, the Schwarz method defined by (3.3) is equivalent to the block Jacobi algorithm (see [DJN15, Chapter 1.2]), which block matrix is associated with local problem on each subdomain. Since we are interested in preconditioners given by overlapping domain decomposition methods, we now present the notation associated with this kind of approach.

3.3 The domain decomposition preconditioner

Let us assume that we have to solve the following linear system

$$(3.6) \quad \mathbf{A}\mathbf{U} = \mathbf{F},$$

where \mathbf{A} is the matrix arising from discretisation of the Stokes equations on the domain Ω , \mathbf{U} is the vector of unknowns and \mathbf{F} is the right hand side. To accelerate the performance of an iterative Krylov method applied to this system we will consider domain decomposition preconditioners which are naturally parallel [DJN15, Chapter 3]. They are based on an overlapping partition of the computational domain.

Let $\{\mathcal{T}_{h,i}\}_{i=1}^N$ be a partition of the triangulation \mathcal{T}_h . For an integer value $l \geq 0$, we define an overlapping decomposition $\{\mathcal{T}_{h,i}^l\}_{i=1}^N$ such that $\mathcal{T}_{h,i}^l$ is a set of all triangles from $\mathcal{T}_{h,i}^{l-1}$ and all triangles from $\mathcal{T}_h \setminus \mathcal{T}_{h,i}^{l-1}$ that have non-empty intersection with $\mathcal{T}_{h,i}^{l-1}$, and $\mathcal{T}_{h,i}^0 = \mathcal{T}_{h,i}$. With this definition the width of the overlap will be of $2l$. Furthermore, if W_h stands for the finite element space associated with \mathcal{T}_h , $W_{h,i}^l$ is the local finite element spaces on $\mathcal{T}_{h,i}^l$ that is a triangulation of Ω_i .

Let \mathcal{N} be the set of indices of degrees of freedom of W_h and \mathcal{N}_i^l the set of indices of degrees of freedom of $W_{h,i}^l$ for $l \geq 0$. Moreover, we define the restriction operator $\mathbf{R}_i : W_h \rightarrow W_{h,i}^l$ as a rectangular matrix $|\mathcal{N}_i^l| \times |\mathcal{N}|$ such that if \mathbf{V} is the vector of degrees of freedom of $v_h \in W_h$, then $\mathbf{R}_i \mathbf{V}$ is the vector of degrees of freedom of W_h in Ω_i . Abusing notation we denote by \mathbf{R}_i both the operator, and its associated matrix. The extension operator from $W_{h,i}^l$ to W_h and its associated matrix are both then given by \mathbf{R}_i^T . In addition we introduce a partition of unity \mathbf{D}_i

as a diagonal matrix $|\mathcal{N}_i^l| \times |\mathcal{N}_i^l|$ such that

$$(3.7) \quad \mathbf{Id} = \sum_{i=1}^N \mathbf{R}_i^T \mathbf{D}_i \mathbf{R}_i,$$

where $\mathbf{Id} \in \mathbb{R}^{|\mathcal{N}| \times |\mathcal{N}|}$ is the identity matrix.

We are ready to present the RAS preconditioner [CS99], given by

$$(3.8) \quad \mathbf{M}_{RAS}^{-1} = \sum_{i=1}^N \mathbf{R}_i^T \mathbf{D}_i (\mathbf{R}_i \mathbf{A} \mathbf{R}_i^T)^{-1} \mathbf{R}_i.$$

We also introduce a new preconditioner that is modification of the one above. The modification is similar to the Optimized RAS [SCGT07], however we do not use Robin interface conditions. For this, let \mathbf{B}_i be the matrix associated to a discretisation of (1.1) in Ω_i where we impose either TVNF (1.2) or NVTF (1.3) boundary conditions in Ω_i . Then, the preconditioner reads

$$(3.9) \quad \mathbf{M}_{MRAS}^{-1} = \sum_{i=1}^N \mathbf{R}_i^T \mathbf{D}_i \mathbf{B}_i^{-1} \mathbf{R}_i.$$

As we mentioned before we will be solving the preconditioned system using the iterative methods. The most basic one is the fixed-point iteration

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \mathbf{M}^{-1} (\mathbf{F} - \mathbf{A} \mathbf{U}^n),$$

where \mathbf{M}^{-1} depends on the method used (RAS or MRAS). We can then see that the solution of this iteration is given in a space spanned by powers of the matrix $\mathbf{Id} - \mathbf{M}^{-1} \mathbf{A}$. This issue can be accelerated by Krylov space [DJN15, Chapter 3] that basis form such space.

3.4 Partition of unity

The above definitions of the preconditioners can be associated with any discretisation of the problem. However, each discretisation involves the construction of a relevant partition of unity \mathbf{D}_i , $i = 1, \dots, N$. We discuss here the construction of \mathbf{D}_i when the problem (1.1) is discretised by the hdG method in the case $k = 1$, either with TVNF boundary conditions (1.20), or NVTF boundary conditions (1.61). Let us introduce the piecewise linear functions $\tilde{\chi}_i^l$ of \mathcal{T}_h such that

$$\tilde{\chi}_i^l = \begin{cases} 1 & \text{at all nodes of } \mathcal{T}_{h,i}^0, \\ 0 & \text{at other nodes.} \end{cases}$$

Now we define the piecewise linear functions χ_i^l of $\mathcal{T}_{h,i}^l$ as follows

$$\chi_i^l := \frac{\tilde{\chi}_i^l}{\sum_{j=1}^N \tilde{\chi}_j^l}.$$

Obviously $\sum_{i=1}^N \chi_i^l = 1$. We define the partition of unity matrix \mathbf{D}_i as a block diagonal matrix where the first block \mathbf{D}_i^{BDM} is associated with BDM_h^1 , the second \mathbf{D}_i^M with M_h^0 and the third \mathbf{D}_i^Q with Q_h^0 . The degrees of freedom of the BDM elements are associated with the normal components on the edges of the mesh. For these finite elements, the diagonal of \mathbf{D}_i^{BDM} is a vector obtained by interpolating χ_i^l at the two points of the edges. The degrees of freedom of the Lagrange multiplier finite elements are associated with the edges of the mesh. For these finite elements, the diagonal of \mathbf{D}_i^M is a vector obtained by interpolating χ_i^l at the midpoints of the edges. For pressure finite elements, the diagonal of \mathbf{D}_i^Q is a vector obtained by interpolating χ_i^l at the midpoints of the elements.

3.5 Preconditioners comparison

In this section we compare the standard RAS preconditioner (3.8) with the newly introduced preconditioners, that is the ones based on non standard interface conditions. We call them MRAS preconditioners (3.9) and more precisely TVNF-MRAS for which \mathbf{B}_i is the matrix of discretisation of (1.1) in Ω_i with interface conditions (1.2) on $\partial\Omega_i$, and NVTF-MRAS for which \mathbf{B}_i is the matrix of discretisation of (1.1) in Ω_i with interface conditions (1.3) on $\partial\Omega_i$. For a fairer comparison we add also similar preconditioners but based on a more standard discretisation that is the lowest order Taylor-Hood discretisation [GR86, Chapter II, Section 4.2].

All cases are used in conjunction with a Krylov iterative solver like GMRES [SS86]. Tables in this chapter show the number of iterations needed to achieve a relative L^2 norm of error smaller than 10^{-6} , $\frac{\|\mathbf{U} - \mathbf{U}_n\|_\Omega}{\|\mathbf{U} - \mathbf{U}_0\|_\Omega} < 10^{-6}$, where \mathbf{U} is the one domain solution and \mathbf{U}_m denotes the approximation of \mathbf{U} at the m -th iteration of the iterative solver. The overlapping decomposition into subdomains can be uniform (Unif) or generated by METIS (MTS) [KK98]. In addition, N stands for the number of subdomains in all tables.

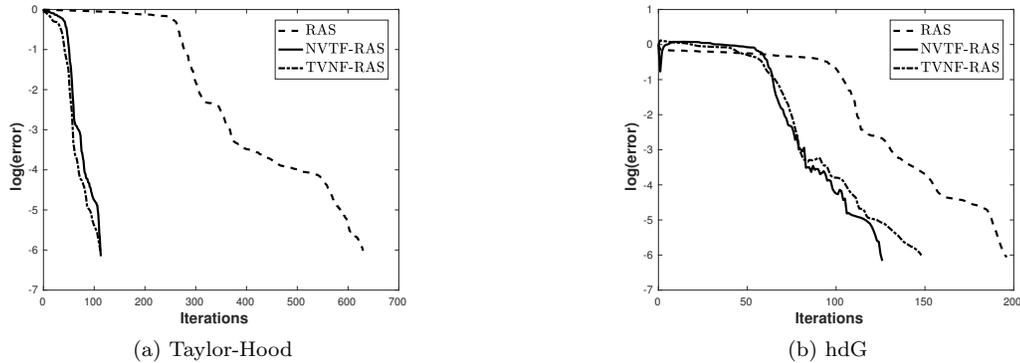
Example 2. We consider the same example as in the Section 1.6.4 where the analytic solution is depicted in Figure 1.3. The problem (1.4) is discretised by Taylor-Hood and the symmetric formulation of hdG discretisations. The mesh is uniform and contains 125 000 triangles for a total of 565 003 degrees of freedom for the Taylor-Hood discretisation and 689 000 degrees of freedom for the hdG discretisation. We use a random initial guess for the GMRES iterative solver and it has two layers of mesh size h in the overlap.

The first thing that we notice from results in Table 3.1 is a significant convergence improvement in the case of RAS applied to a system resulting from a hdG discretisation in comparison to the RAS applied to the system resulting from the Taylor-Hood discretisation. Even in spite

N	Taylor Hood						hdG					
	RAS		NVTF-MRAS		TVNF-MRAS		RAS		NVTF-MRAS		TVNF-MRAS	
	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
4	133	311	40	39	37	37	58	95	41	45	53	50
9	336	563	58	58	52	60	94	131	62	66	69	81
16	315	691	60	76	59	73	101	151	68	85	80	100
25	427	774	76	93	71	90	127	186	77	100	103	119
64	630	1132	113	147	112	132	196	280	126	172	148	183
100	769	1246	136	174	132	169	247	348	151	205	175	228
144	929	1434	158	201	155	192	306	408	178	228	192	259
196	1000	1637	180	239	168	224	354	480	198	326	212	299
256	1133	1805	201	265	183	286	403	536	226	358	233	341

Table 3.1: Preconditioners comparison - Example 2

of the fact that the number of degrees of freedom is slightly bigger in the first case and the discretisation order is similar. A change in discretisation presumably leads to better conditioned systems to solve. Also the MRAS preconditioner with both discretisations perform better than the standard RAS method which fully justifies the use of the new interface conditions no matter the discretisation method. Moreover, as expected the iteration number increases with respect to the number of the subdomains and this behaviour is common to the three preconditioners. It is worth noting that this increase is slower than the expected linear one.

Figure 3.3: Convergence of error for uniform decomposition into the 8×8 subdomains - Example 2

Figures 3.3 and 3.4 show the convergence of the error for the different discretisations. We observe that in all cases the MRAS preconditioner defined by (3.9) shortens the plateau region in the convergence curves significantly which leads, automatically, to a significant reduction in the number of iterations.

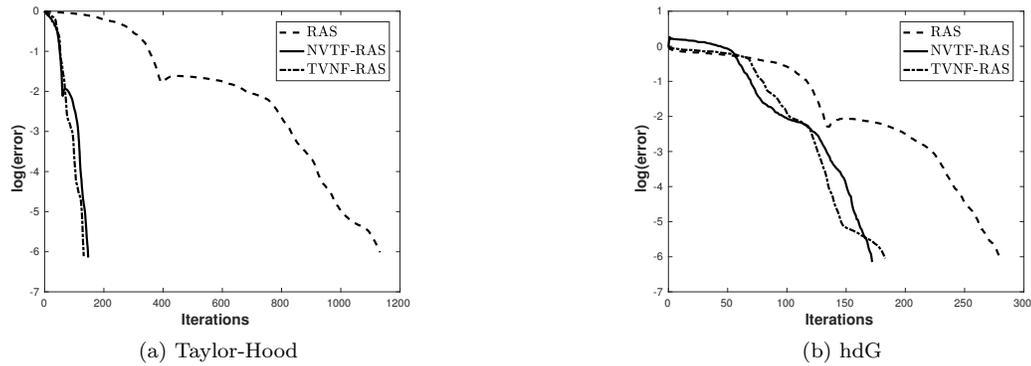


Figure 3.4: Convergence of error for METIS decomposition into the 64 subdomains - Example 2

Example 3. We consider the same example as in the Section 1.6.4 where the analytic solution is depicted in Figure 1.5. The problem (1.4) is discretised by Taylor-Hood and symmetric hdG discretisations. The mesh is again uniform and contains 125 000 triangles for a total of 565 003 degrees of freedom for the Taylor-Hood discretisation and 689 000 degrees of freedom for the hdG discretisation. We use a random initial guess for the GMRES iterative solver and it has three layers of mesh size h in the overlap.

N	Taylor Hood						hdG					
	RAS		NVTF-MRAS		TVNF-MRAS		RAS		NVTF-MRAS		TVNF-MRAS	
	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
4	117	220	36	39	38	36	58	95	39	47	54	48
9	294	421	63	60	54	54	103	129	66	67	77	78
16	236	510	59	73	61	68	98	153	65	83	74	94
25	300	642	68	89	72	83	120	184	77	103	88	115
64	454	916	102	144	100	122	188	279	117	160	120	165
100	559	1088	122	173	116	154	225	349	140	198	138	215
144	940	1251	176	195	145	215	342	395	198	231	183	232
196	781	1346	166	230	146	242	325	486	191	277	173	284
256	881	1553	189	269	159	272	368	538	210	316	195	309

Table 3.2: Preconditioners comparison - Example 3

Looking at the Table 3.2 we see that the conclusions remain the same as in previous example. But it is worthy to mention that we can notice the little improvement in comparison to the previous example (Example 2) since we consider a bigger size of the overlap. These differences are bigger in the case of RAS preconditioners than MRAS ones. We observe in Figures 3.5 and 3.6 that once again the MRAS preconditioner defined by (3.9) reduces the number of iterations in all cases.

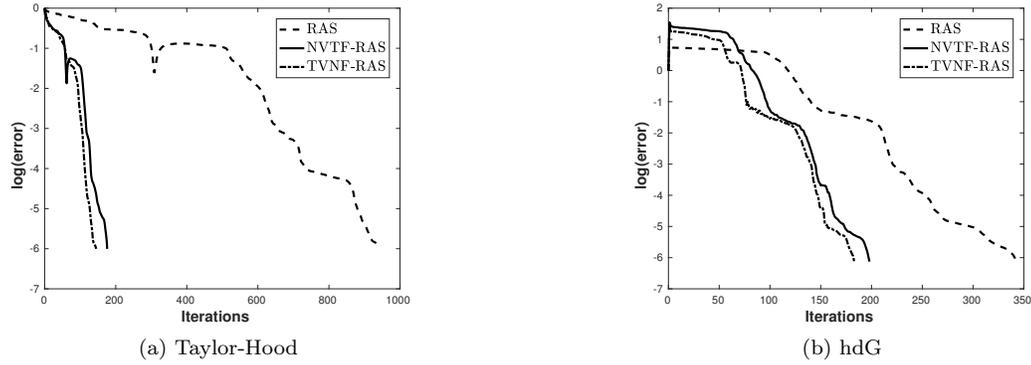


Figure 3.5: Convergence of error for uniform decomposition into the 12×12 subdomains - Example 3

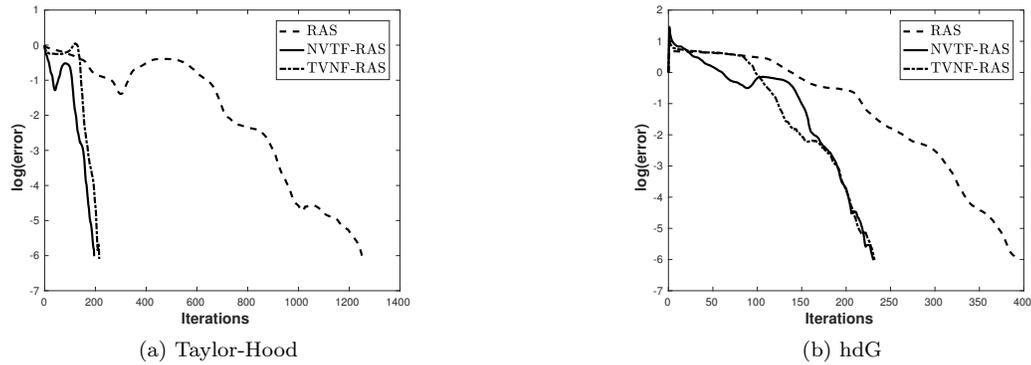


Figure 3.6: Convergence of error for METIS decomposition into the 144 subdomains - Example 3

Example 5. The next test case is the driven cavity. We consider the following problem on the unit square $\Omega = (0, 1)^2$

$$(3.10) \quad \begin{cases} -\Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega \\ -\nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \\ \mathbf{u}(x, y) = (1, 0)^T & \text{on } \partial\Omega \cap \{y = 1\} \\ \mathbf{u}(x, y) = (0, 0)^T & \text{on } \partial\Omega \setminus \{y = 1\} \end{cases} .$$

In Figure 3.7 we plot the vector field and pressure, after solving numerically the problem. In this case, we used a uniform mesh containing 245 000 triangles, which gives a linear system of size 1 106 003 for the Taylor-Hood discretisation and 1 349 600 for the hdG discretisation. We use a random initial guess for the GMRES iterative solver. The overlapping decomposition into subdomains can be uniform or generated by METIS and it has two layers of mesh size h in the overlap.

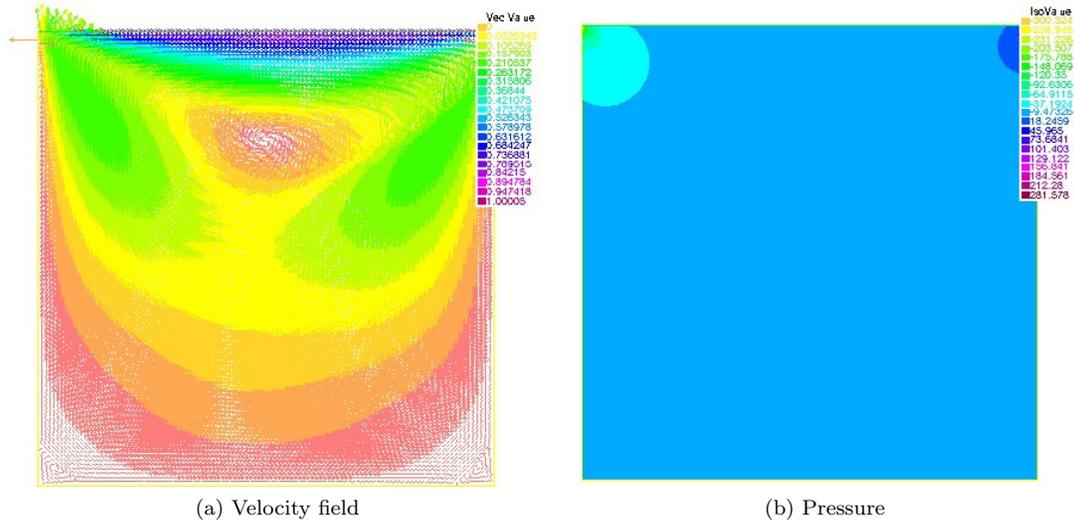


Figure 3.7: Numerical solution of the driven cavity problem - Example 5

N	Taylor Hood						hdG					
	RAS		NVTF-MRAS		TVNF-MRAS		RAS		NVTF-MRAS		TVNF-MRAS	
	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
4	276	323	41	33	37	34	71	93	46	51	64	40
9	350	419	49	49	59	44	86	97	54	55	81	57
16	399	504	63	65	65	59	106	147	68	69	86	74
25	455	574	76	79	82	76	127	142	81	87	109	87
64	656	771	113	125	121	121	206	269	125	133	152	144
100	785	923	136	156	151	156	258	340	150	174	176	193
144	927	1067	161	186	175	177	310	343	179	192	205	220
196	1067	1212	181	208	192	213	357	391	202	246	223	259
256	1212	1340	205	236	208	257	405	443	229	298	245	301

Table 3.3: Preconditioners comparison - Example 5

Table 3.3 shows that the MRAS preconditioners with both discretisations perform better than the standard RAS method. Also, as expected the iteration number increases with respect to the number of the subdomains and this behaviour is common to the three preconditioners. Finally, we also plot the convergence of the error of the different discretisations in Figure 3.8 and 3.9. And again, in all cases the MRAS preconditioner defined by (3.9) shortens the plateau region.

Similar results can be obtained for the stabilised hdG method proposed in the Chapter 2. In fact, in Table 3.4 we repeat the results for the cavity problem with the same mesh containing 245 000 triangles, which gives a linear system of size 1 839 600 for the stabilised hdG discretisation. We only present the results for this case for brevity, since the results for new preconditioners are very similar to those of the unstabilised hdG discretisation from Chapter 1. We do not present the results for RAS preconditioner, since the restricted matrix is singular. This underline the universalism of the new preconditioners.

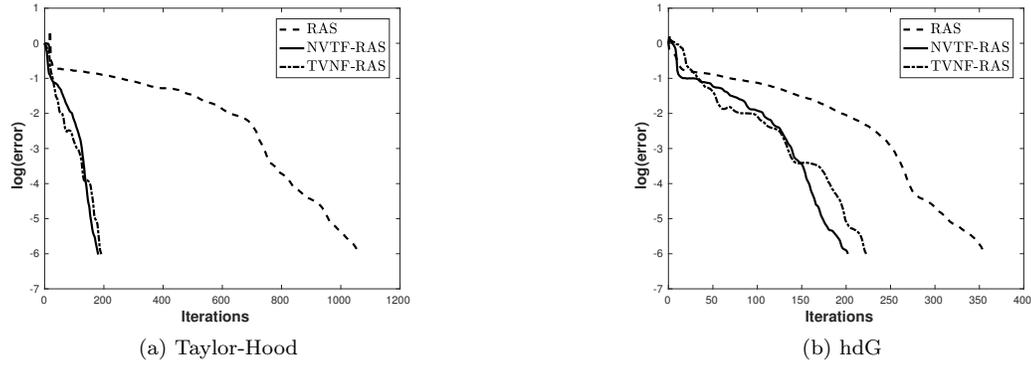


Figure 3.8: Convergence of error for uniform decomposition into the 14×14 subdomains - Example 5

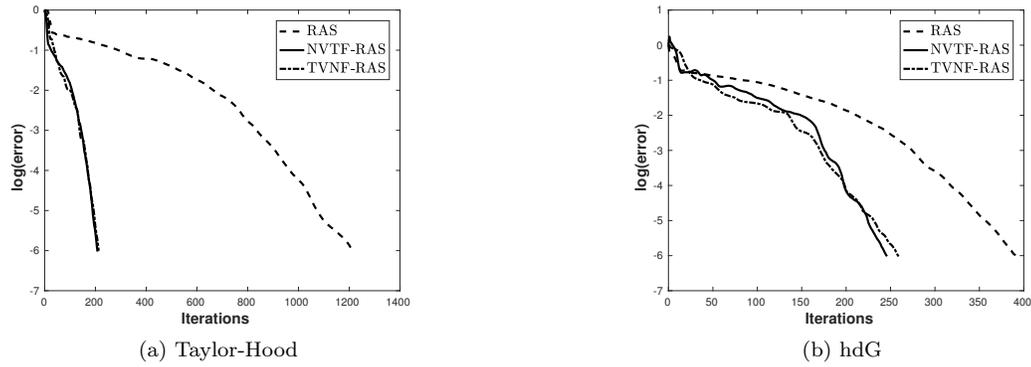


Figure 3.9: Convergence of error for METIS decomposition into the 196 subdomains - Example 5

N	Stabilised hdG			
	NVTF-MRAS		TVNF-MRAS	
	Uniform	METIS	Uniform	METIS
4	61	35	63	41
9	54	54	81	57
16	68	91	86	74
25	82	88	109	88
64	126	133	152	144
100	150	172	179	193
144	179	192	205	221
196	202	244	224	259
256	229	297	245	301

Table 3.4: Preconditioners comparison for stabilised hdG - Example 5

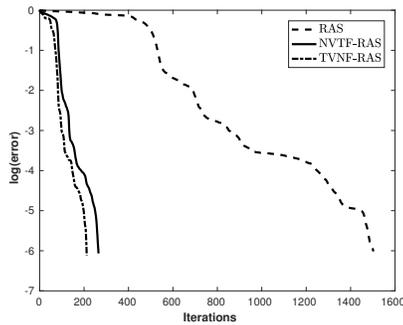
Example 1. We consider the same example as in the Section 1.6.4 where the analytic solution is depicted in Figure 1.1. The problem (1.4) is discretised by Taylor-Hood and symmetric hdG discretisation. In this case, we used a uniform mesh containing 245 000 triangles, which gives a linear system of size 1 106 003 for the Taylor-Hood discretisation and 1 349 600 for the hdG

discretisation. We use a random initial guess for the GMRES iterative solver and it has one layer of mesh size h in the overlap.

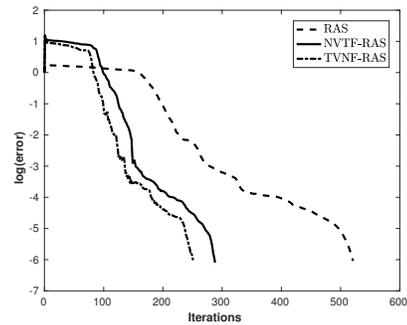
N	RAS		NVTF-MRAS		TVNF-MRAS		RAS		NVTF-MRAS		TVNF-MRAS	
	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
4	194	313	46	46	40	40	83	110	49	55	59	55
9	409	533	66	64	51	60	129	137	74	76	80	79
16	475	692	78	85	79	73	145	179	84	97	85	108
25	576	837	95	102	93	86	173	206	102	118	100	128
64	834	1120	146	163	136	135	253	315	160	187	163	191
100	1013	1410	176	201	155	167	317	376	190	228	190	239
144	1207	1585	209	229	175	206	390	452	234	267	214	278
196	1343	1745	232	259	194	245	461	528	258	303	240	326
256	1503	1932	265	302	212	282	521	609	288	368	251	349

Table 3.5: Preconditioners comparison - Example 1

We arrive with the same conclusions as in previous examples. However, it is worthy to mention that we can notice that the results in Table 3.5 are worse than in the previous example (Table 3.3) since we consider a smaller size of overlap. These differences are bigger in the case of RAS preconditioners than MRAS ones. The plots in Figure 3.10 and 3.11 show that the MRAS preconditioner defined by (3.9) reduces the number of iterations.



(a) Taylor-Hood



(b) hdG

Figure 3.10: Convergence of error for uniform decomposition into the 16×16 subdomains - Example 1

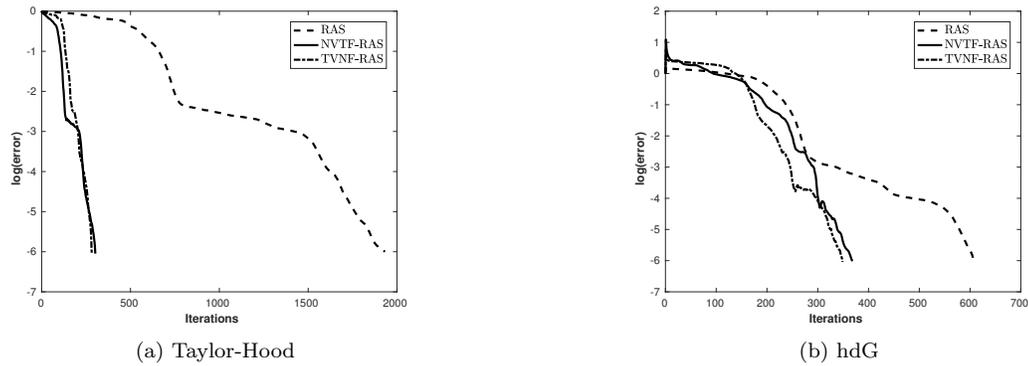


Figure 3.11: Convergence of error for METIS decomposition into the 256 subdomains - Example 1

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions given by

$$(3.11) \quad \mathbf{u}(x, y) = \begin{cases} (4y(1 - y), 0)^T & \text{if } x = 0 \text{ or } x = 1.5 \\ (0, 0)^T & \text{otherwise.} \end{cases}$$

The numerical solution of this problem is depicted in Figure 3.12. In this case, we used a

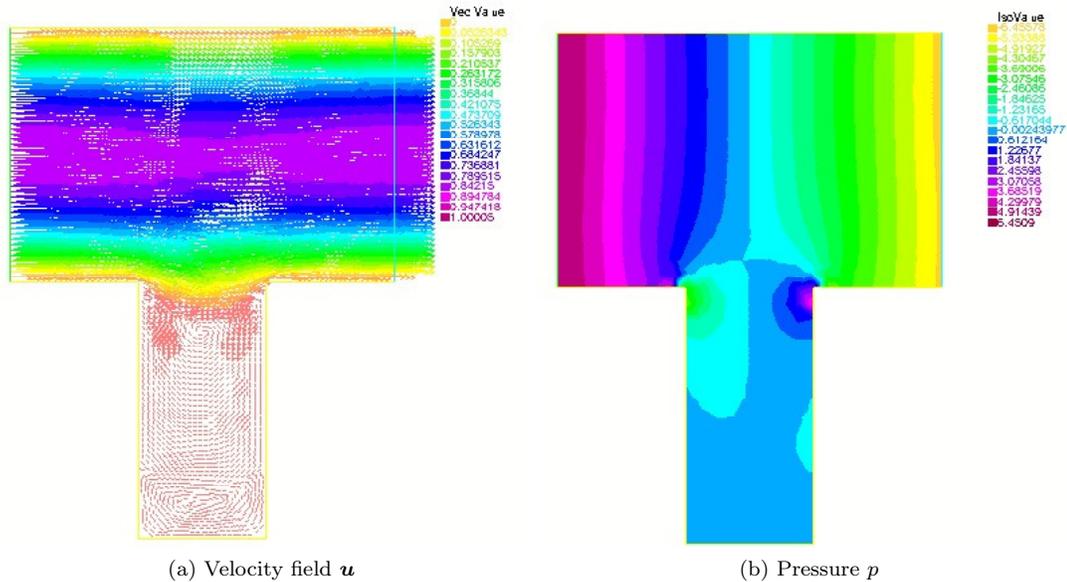


Figure 3.12: Numerical solution - Example 4

mesh containing 379 402 triangles, which gives linear systems of a size 1 712 352 for the Taylor-Hood discretisation and 2 089 735 for the hdG discretisation. The initial guess in the GMRES

iterative solver is zero. The overlapping decomposition into subdomains is generated by METIS and it has two layers of mesh size h in the overlap.

N	Taylor-Hood			hdG		
	RAS	NVTF-MRAS	TVNF-MRAS	RAS	NVTF-MRAS	TVNF-MRAS
50	752	121	105	209	132	135
100	903	175	147	307	190	197
200	1272	245	211	441	264	281
400	1747	341	342	613	366	399
800	2433	469	417	863	650	549

Table 3.6: Preconditioners comparison - Example 4

In this case the conclusions remain the same, that is the standard RAS method performs far better when applied to a hdG discretisation with respect to a Taylor-Hood one and the MRAS preconditioners are better than the standard RAS preconditioner for both discretisations (see Table 3.6). Figure 3.13 again confirms the superiority of the MRAS preconditioner defined

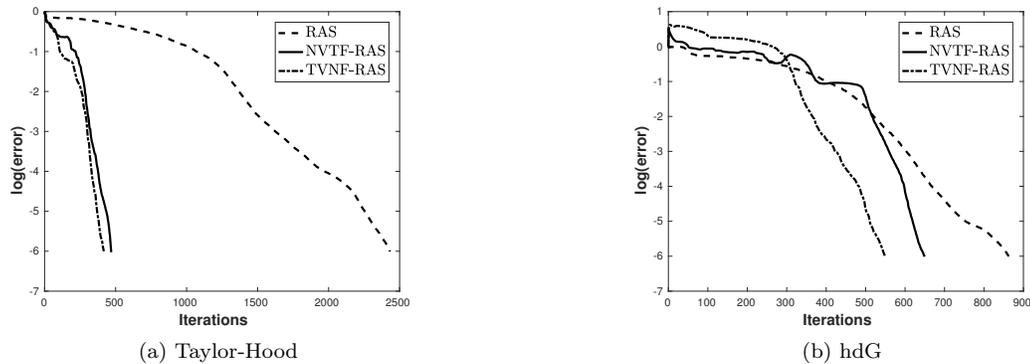


Figure 3.13: Convergence of error for METIS decomposition into the 800 subdomains - Example 4

by (3.9) over the RAS preconditioner defined by (3.8).

3.6 Summary

To solve the discretised Stokes problem we introduced two different kinds of preconditioners with non standard boundary conditions. The simulations have been made for two different finite element methods that are hdG and Taylor-Hood discretisations. We compared the newly introduced preconditioners to the more standard RAS preconditioner. The numerical experiments from Section 3.5 clearly show their superiority for different test cases in two space dimensions. The difference in the case of the Taylor-Hood discretisation is more visible than when using the hdG discretisation. This is due to the RAS preconditioner applied to the system resulting

from the hdG discretisation already performs far better than to the one resulting from the Taylor-Hood discretisation.

We consider two pairs of examples that were defined on the same mesh size with different size of the overlaps. Despite that the problems are different, the same mesh leads to the system of the same degrees of freedom. Hence, we could also observed the improvement that is caused by the increase of the size of the overlap.

The biggest disadvantage of the one-level preconditioners is lack of scalability with respect to the number of subdomains. We see in all cases that by taking more subdomains, even that the problem does not get larger, the number of iterations increases. As we already noted, the increase in case of our preconditioners is slower than the linear one in RAS preconditioner case. However, this is far from results that we would name as scalable. In the next chapter we consider two-level methods that bring the desired improvement.

Chapter 4

Coarse spaces

We observed, as expected, that Schwarz preconditioners are not scalable with respect to the number of subdomains. However, this can be fixed by using appropriate coarse spaces [DJN15, Chapter 4]. An example of a coarse space is presented in this chapter. We begin with the state of the art in Section 4.1. Later we introduce symmetrised versions of the preconditioners from Chapter 3.3. In Section 4.3, the new two-level preconditioners with coarse spaces associated to generalised eigenvalue problems are defined. Section 4.4 contains numerical simulations illustrating weak scaling for Stokes problems and nearly incompressible elasticity. Finally, a summary is outlined in Section 4.5.

4.1 State of the art

As we have seen in the previous chapter, splitting the problem into a larger number of subdomains leads to an increase of size of the plateau region in the convergence of an iterative method (see Figure 4.1). This is caused by the lack of global information, as subdomains can only communicate with their neighbours. Hence, when the number of subdomains increases in one direction, the length of the plateau also increases. Even in cases when the local problems are of the same size, the iterations count grows with the increase of the number of subdomain. This can be also observe in all experiments in this chapter in case of one-level methods.

The remedy for this is the usage of a second level in the preconditioner or a coarse space correction that adds the necessary global information. Two-level algorithms have been analysed for several classes of problems in [TW05]. The analysis is based on the idea of *stable splitting* which leads to convergence estimates independent of the parameters of the problems when the number of degrees of freedom per domain is kept constant.

In many applications we have to deal with strongly heterogeneous problems. For this reason, building two-level methods which are also robust with respect to the physical coefficients is very

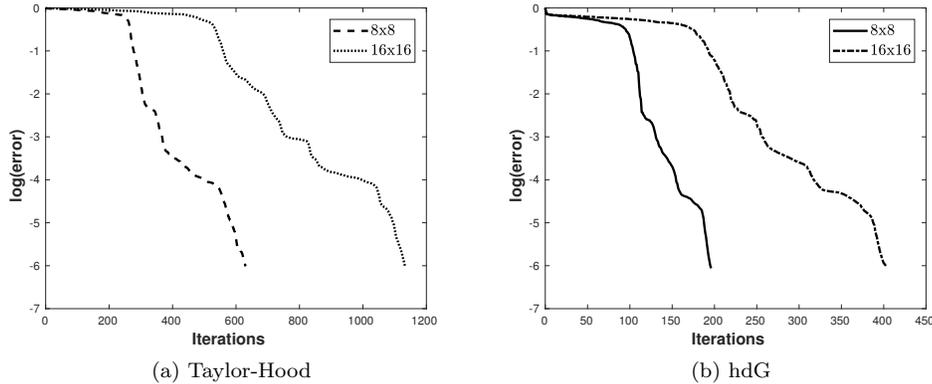


Figure 4.1: Increase of size of the plateau region - Example 2

important. This idea was introduced for the first time in [BHMV99] in the case of multigrid methods. This approach relies on solving local generalised eigenvalue problems to generate suitable vectors for the coarse space.

For overlapping domain decomposition preconditioners, a similar idea was introduced in the case of Darcy equations in [GE10a] by solving local generalised eigenvalue problems in the overlaps. Their solutions were used to identify vanishing eigenvalues and adding the associated eigenvectors to the coarse space. Unfortunately, since the right hand side of each problem involves a local mass matrix, this can lead to a large coarse space. That is why, the same authors redefined this approach by using a multiscale partition of unity in [GE10b].

The authors of [NXD10] consider also the heterogeneous Darcy equation and presented a different generalised eigenvalue problem based on local Dirichlet-to-Neumann maps. This approach comes down to the solution of problems only on the interfaces of the subdomains. The method has been analysed in [DNSS12] and proved to be very robust in the case of small overlaps. The same idea was extended numerically to the heterogeneous Helmholtz problem in [CDKN14]. The authors of [LNS15] apply the coarse space associated with low-frequency eigenfunctions of the subdomain Dirichlet-to-Neumann maps for the generalisation of the optimised Schwarz methods, named 2-Lagrange multiplier methods. The optimisation is associated in usage the local Robin boundary value problems instead of the local Dirichlet or Neumann ones as it is in the previous cases.

There were some attempts to extend this spectral approach to general symmetric positive definite problems. The authors of [EGLW12] develop an abstract framework that is an extension of [GE10a] and [GE10b]. Since some of the assumptions of the previous framework are hard to fulfil, authors of [SDH⁺14] proposed slightly different approach for symmetric positive definite problems. Their idea of constructing partition of unity operator associated with degrees of freedom allows to work with various finite element spaces.

An overview of different kinds of two-level methods can be found in [DJN15, Chapters 5 and 7]. Despite the fact that all these approaches provide satisfactory results, there is no universal treatment to build efficient coarse spaces in the case of non definite problems such as Stokes equations.

The coarse spaces that we are using in this work are inspired by those proposed in [HJN15]. The authors introduced and tested numerically symmetrised two-level preconditioners for overlapping algorithms which use Robin interface conditions between the subdomains (see (4.13) below for details). They applied these preconditioners to solve saddle point problems such as nearly incompressible elasticity and Stokes discretised by Taylor-Hood finite elements. In our case, we use non standard interface conditions. Therefore the use of spectral coarse spaces could lead to an important gain.

4.2 Symmetrised Modified Restricted Additive Schwarz

The main goal of this chapter is to develop a computational framework for the two-level overlapping Schwarz methods based on local problems containing non standard interface conditions. As no theory is available for the kind of saddle point problems presented in [HJN15], we will restrict our investigation to numerical results. We will consider a symmetrised version of the MRAS preconditioner

$$(4.1) \quad \mathbf{M}_{MRAS}^{-1} = \sum_{i=1}^N \mathbf{R}_i^T \mathbf{D}_i \mathbf{B}_i^{-1} \mathbf{R}_i,$$

where \mathbf{B}_i is the matrix associated to a discretisation of (1.1) in Ω_i where we impose either TVNF (1.2) or NVTF (1.3) boundary conditions in Ω_i . The symmetrised variant of above preconditioner is given by

$$(4.2) \quad \mathbf{M}_{SMRAS}^{-1} = \sum_{i=1}^N \mathbf{R}_i^T \mathbf{D}_i \mathbf{B}_i^{-1} \mathbf{D}_i \mathbf{R}_i.$$

4.3 Two-level methods

As we already mentioned the remedy for lack of scalability consists in introducing the coarse space that couples all subdomains at each iteration. The classical or Nicolaidis coarse space was presented in [Nic87] for the symmetric positive definite kind of problems. We define \mathbf{Z}_{Nic} as the matrix such that for all $1 \leq i \leq N$ the i -th column looks like following

$$\mathbf{Z}_i := \mathbf{R}_i^T \mathbf{D}_i \mathbf{R}_i \mathbf{1},$$

where $\mathbf{1}$ is the vector of dimension N full of ones. Furthermore, we define $\mathbf{R}_0 := \mathbf{Z}_{Nic}^T$ and then we combine additively this coarse space with one level preconditioner. Such approach can derives us to the two-level additive Schwarz preconditioner is defined as follows

$$(4.3) \quad \mathbf{M}_{AS,2}^{-1} = \mathbf{R}_0^T \left(\mathbf{R}_0 \mathbf{A} \mathbf{R}_0^T \right)^{-1} \mathbf{R}_0 + \sum_{i=1}^N \mathbf{R}_i^T \left(\mathbf{R}_i \mathbf{A} \mathbf{R}_i^T \right)^{-1} \mathbf{R}_i.$$

The above approach works fine for the symmetric positive definite kind of problems that do not include any jumps of parameters inside the domain. However for other kind of problems there is no universal treatment to build efficient coarse spaces. That is why, we test the spectral coarse space approach for Stokes-like problems.

Our coarse space for two-level SMRAS and MRAS algorithm will be based on the following local generalised eigenvalue problems:

find $(\mathbf{V}_{jk}, \lambda_{jk}) \in \mathbb{R}^{|\mathcal{N}_j|} \setminus \{0\} \times \mathbb{R}$ such that

$$(4.4) \quad \tilde{\mathbf{A}}_j \mathbf{V}_{jk} = \lambda_{jk} \mathbf{B}_j \mathbf{V}_{jk},$$

where $\tilde{\mathbf{A}}_j$ are local matrices associated to a discretisation of local Neumann boundary value problem in Ω_j . Let $\theta > 0$ be a user-defined threshold. We define $Z_{GenEO} \subset \mathbb{R}^{|\mathcal{N}|}$ as the vector space spanned by the family of vectors $(\mathbf{R}_j^T \mathbf{D}_j \mathbf{V}_{jk})_{\lambda_{jk} < \theta}$, $1 \leq j \leq N$, corresponding to eigenvalues smaller than θ . Since the choice of θ depends on a problem and a preconditioner, alternatively, we can use just a fixed number of eigenvectors. We are now ready to introduce the two-level method with coarse space Z_{GenEO} .

Let \mathbf{P}_0 be the \mathbf{A} -orthogonal projection onto the coarse space Z_{GenEO} . The two-level SMRAS preconditioner is defined as

$$(4.5) \quad \mathbf{M}_{SMRAS,2}^{-1} = \mathbf{P}_0 \mathbf{A}^{-1} + (\mathbf{Id} - \mathbf{P}_0) \mathbf{M}_{SMRAS}^{-1} (\mathbf{Id} - \mathbf{P}_0^T).$$

Furthermore, using the similar notation as in case of Nicolaides coarse space, if \mathbf{Z}_0 is a matrix whose columns are a basis of the coarse space Z_{GenEO} , and $\mathbf{R}_0 := \mathbf{Z}_0^T$, then

$$\mathbf{P}_0 \mathbf{A}^{-1} = \mathbf{R}_0^T \left(\mathbf{R}_0 \mathbf{A} \mathbf{R}_0^T \right)^{-1} \mathbf{R}_0.$$

Beside the use of different one-level preconditioners the two-level methods defined by (4.3) and (4.5), we can notice that the second one includes the projector outside the coarse space $(\mathbf{Id} - \mathbf{P}_0)$ and its transpose $(\mathbf{Id} - \mathbf{P}_0^T)$. This is named as two-level hybrid Schwarz preconditioner. In similar way, we can introduce the two-level MRAS preconditioner

$$(4.6) \quad \mathbf{M}_{MRAS,2}^{-1} = \mathbf{P}_0 \mathbf{A}^{-1} + (\mathbf{Id} - \mathbf{P}_0) \mathbf{M}_{MRAS}^{-1} (\mathbf{Id} - \mathbf{P}_0^T).$$

4.4 Numerical results

In this section we will compare a two-level SMRAS preconditioner with the coarse space Z_{GenEO} presented above with the two-level SORAS preconditioner with the coarse space Z_{GenEO} presented in [HJN15]. Moreover, we compare the results with the one-level methods that have been introduced in Chapter 3. We consider the partial differential equation model for nearly incompressible elasticity and Stokes flow as problems of similar mixed formulation. Each of these problems is discretised by using the hdG discretisation introduced in Chapter 1 and Taylor-Hood methods.

We will perform some typical weak scaling experiments, that is increase the size of global problem such that $\frac{H}{h}$ remains constant, where H is maximum diameter of the subdomains and $h := \max_{K \in \mathcal{T}_h} h_K$. We use GMRES [SS86], but it possible to use any other Krylov iterative solver. Generalized eigenvalue problems to generate the coarse space are solved using ARPACK [LSY98]. We define the coarse space by collecting the eigenvectors resulting from the solutions of the local eigenvalue problems. The easiest way to build a coarse space is to incorporate zero energy modes in it. By the zero energy mode we understand the eigenvector associated with the zero eigenvalue on the floating subdomain, that is subdomain that does not have any Dirichlet boundary conditions.

The overlapping decomposition into subdomains can be uniform (Unif) or generated by METIS (MTS). In each of the examples in this chapter we consider decompositions with two layers of mesh size h in the overlap. Tables in this chapter show the number of iterations needed to achieve a relative L^2 norm of the error smaller than 10^{-6} , $\frac{\|\mathbf{U} - \mathbf{U}_n\|_\Omega}{\|\mathbf{U} - \mathbf{U}_0\|_\Omega} < 10^{-6}$, where \mathbf{U} is the one domain solution and \mathbf{U}_m denotes the approximation of \mathbf{U} at the m -th iteration of the iterative solver. In addition, DOF stands for number of degrees of freedom and N for the number of subdomains in all tables.

4.4.1 Stokes equation

In this section we discretise Stokes flow by using the lowest order hdG discretisation introduced in Chapter 1 and the low order Taylor-Hood discretisation that will be described in next section.

hdG discretisation

As a reminder we present the discrete problem:

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^1 \times M_h^0 \times Q_{h,0}^0$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^1 \times M_h^0 \times Q_{h,0}^0$

$$(4.7) \quad \begin{cases} a((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) = \int_\Omega \mathbf{f} \mathbf{v}_h \, dx \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) = 0, \end{cases}$$

where a is defined by (1.18) and b is defined by (1.19).

Since we consider the preconditioners with various interface conditions we need to comment the way of imposing them. In the case of ORAS and SORAS we consider the Robin interface conditions as in [HJN15]. This means, the weak formulation of the Stokes problem contains the following term

$$(4.8) \quad \int_{\partial\Omega_i \setminus \partial\Omega} \boldsymbol{\sigma}_n(\mathbf{v}_h)_n \, ds + \int_{\partial\Omega_i \setminus \partial\Omega} \frac{4}{3} \nu \alpha \mathbf{u}_h \mathbf{v}_h \, ds$$

where $\alpha = 10$. Fortunately, the MRAS and SMRAS preconditioners are parameter-free.

We start with the well known example that was discussed in previous chapter.

Example 5. The test case is the driven cavity from Section 3.5 defined in (3.10). In this experiment of the preconditioners for the GMRES iterative solver we use a random initial guess.

We start with the two energy modes only (see Figure 4.2). This already provides some improvement. Then, we add more eigenvectors to see if they bring bigger improvement. In the case of the Stokes equations we can observe only two zero eigenvalues. These two zeros are associated with two constants as eigenvectors.

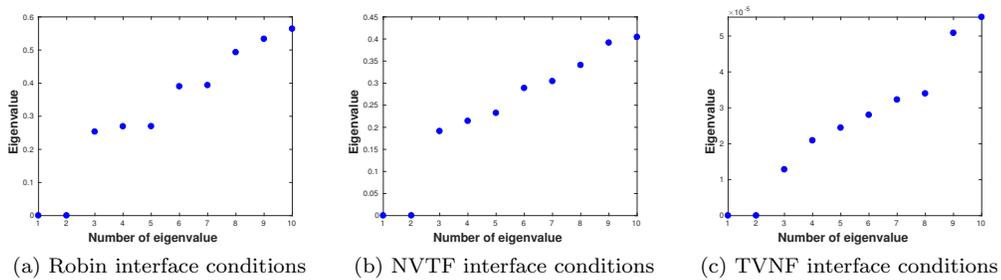


Figure 4.2: Eigenvalues on one of the floating subdomains in the case of uniform decomposition and hdG discretisation - Example 5

The results of Table 4.1 show a clear improvement in the scalability of the two-level preconditioners over the one-level ones. We cannot conclude that SMRAS preconditioners are much better than SORAS, since both provide fully scalable results. It means that, as expected, the iteration number does not increase with respect to the number of the subdomains. Although Table 4.1 shows that coarse spaces containing five eigenvectors seem to decrease the number of iterations even in the case of non-symmetric preconditioners (ORAS and MRAS) that are not fully scalable. The need of bigger size of coarse spaces than the basic one including two eigenvectors in each subdomain is unexpected, since the test case has a constant parameter on the whole domain.

DOF	N	One-level											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	13	15	27	26	17	18	30	28	20	17	30	33
70 880	16	32	44	58	61	39	51	68	73	44	47	73	76
159 120	36	213	130	94	101	60	139	122	130	72	82	129	136
282 560	64	252	343	128	143	88	128	190	199	104	123	192	207
635 040	144	402	>1000	217	231	153	193	337	345	170	246	330	374
1 128 320	256	>1000	>1000	314	316	225	374	498	494	245	285	478	729

DOF	N	Two-level (2 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	9	11	24	20	12	12	23	20	21	15	26	23
70 880	16	19	26	33	32	26	26	37	38	45	34	45	44
159 120	36	42	47	42	46	41	48	60	67	60	42	60	63
282 560	64	273	136	47	53	57	74	88	89	64	45	63	71
635 040	144	517	590	52	63	100	104	141	123	68	56	65	82
1 128 320	256	982	>1000	53	61	154	149	228	174	72	61	66	83

DOF	N	Two-level (5 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	8	10	20	18	12	12	21	19	23	15	22	20
70 880	16	16	19	27	24	21	21	30	24	45	28	30	28
159 120	36	28	33	31	28	23	33	33	34	58	33	33	35
282 560	64	48	45	34	31	27	37	36	35	62	35	34	37
635 040	144	143	112	38	34	32	42	36	40	69	47	35	40
1 128 320	256	269	188	40	34	78	51	70	51	73	64	35	41

DOF	N	Two-level (7 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	7	9	19	17	12	20	20	18	24	13	21	19
70 880	16	16	18	25	22	21	18	30	22	49	29	28	24
159 120	36	24	30	27	25	24	28	39	29	65	32	31	29
282 560	64	35	40	28	27	24	31	35	30	67	35	31	30
635 040	144	75	83	29	31	23	37	34	32	79	49	32	32
1 128 320	256	116	133	29	29	47	39	69	42	87	60	32	31

Table 4.1: Comparison of preconditioners for hdG discretisation - Example 5

Similar results can be obtained for the stabilised hdG method proposed in the Chapter 2. In fact, in Table 4.2 we repeat the results for the cavity problem with the same meshes. We only present the results for this case for brevity, since the conclusions stay the same as in case of hdG discretisation.

Example 4. We consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions (3.11). The numerical solution of this problem is depicted in Figure 3.12. In this experiment of the preconditioners for the GMRES iterative solver we use zero as an initial guess. The overlapping decomposition into subdomains is generated by METIS.

In this case we motivate the size of the coarse space by observing a clustering of small eigenvalues of generalised eigenvalue problem defined in (4.4) (see Figure 4.3).

DOF	N	One-level											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
24 240	4	13	16	28	26	17	18	30	30	20	17	31	34
96 480	16	32	45	61	64	39	51	70	75	44	47	77	78
216 720	36	214	130	102	104	60	139	127	132	73	82	132	136
384 960	64	252	343	135	149	88	128	193	203	105	123	195	207
865 440	144	401	>1000	224	233	153	193	347	351	173	216	337	375
1 537 920	256	>1000	>1000	325	335	226	374	510	505	247	288	482	734

DOF	N	Two-level (2 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
24 240	4	9	11	22	21	12	12	24	20	22	15	23	23
96 480	16	19	26	33	33	27	28	42	42	45	34	46	45
216 720	36	42	47	41	49	45	49	66	69	60	42	67	65
384 960	64	278	136	47	55	68	77	94	100	65	50	69	72
865 440	144	513	593	52	59	105	118	147	141	69	56	72	85
1 537 920	256	>1000	>1000	59	58	163	156	376	178	72	61	72	86

DOF	N	Two-level (5 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
24 240	4	8	10	19	18	21	11	22	19	23	15	20	19
96 480	16	16	19	27	25	21	21	37	24	45	30	29	29
216 720	36	28	33	32	28	31	31	49	38	58	33	34	36
384 960	64	48	45	36	31	32	52	53	65	62	36	35	39
865 440	144	134	117	40	33	41	49	46	44	69	46	36	42
1 537 920	256	280	188	44	35	69	75	108	85	75	62	36	42

DOF	N	Two-level (7 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
24 240	4	7	9	18	17	20	21	20	23	23	14	19	18
96 480	16	16	18	23	22	38	19	32	23	49	27	27	24
216 720	36	24	30	26	24	36	28	54	31	64	33	31	28
384 960	64	33	41	28	27	32	39	49	55	67	35	32	31
865 440	144	76	79	30	30	29	47	49	41	79	50	32	33
1 537 920	256	120	133	32	29	52	60	90	75	86	68	33	33

Table 4.2: Comparison of preconditioners for stabilised hdG discretisation - Example 5

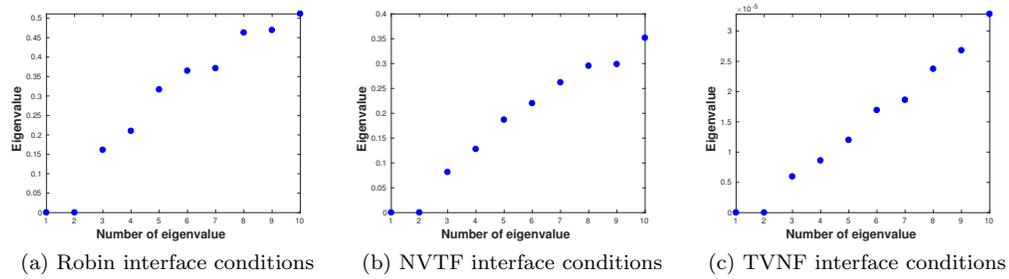


Figure 4.3: Eigenvalues on one of the floating subdomains in the case of METIS decomposition and hdG discretisation - Example 4

In this case scalable results can be only observed for the preconditioners associated with the non standard interface conditions (MRAS and SMRAS), and when using a coarse space which is sufficiently big (see Table 4.3). In the case of ORAS or SORAS, one possibility is to choose a

		One-level					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
38 803	4	22	45	36	49	22	51
154 606	16	111	98	83	172	83	182
311 369	32	265	144	133	262	130	266
616 772	64	568	238	212	410	195	412
1 246 136	128	>1000	494	333	665	313	602
2 451 365	256	>1000	712	464	>1000	477	889

		Two-level (2 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
38 803	4	16	35	31	37	21	38
154 606	16	113	69	73	75	38	75
311 369	32	254	99	103	176	93	162
616 772	64	510	153	171	273	121	140
1 246 136	128	>1000	221	242	252	155	138
2 451 365	256	>1000	286	343	515	189	231

		Two-level (5 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
38 803	4	14	30	27	27	28	30
154 606	16	155	54	54	45	25	44
311 369	32	159	55	72	59	29	52
616 772	64	426	88	106	83	37	76
1 246 136	128	955	113	115	99	43	72
2 451 365	256	>1000	182	138	101	54	73

		Two-level (7 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
38 803	4	13	26	25	24	14	28
154 606	16	85	46	46	38	24	38
311 369	32	168	51	35	44	29	45
616 772	64	382	82	75	62	40	59
1 246 136	128	896	129	86	74	40	64
2 451 365	256	>1000	158	47	71	53	61

Table 4.3: Comparison of preconditioners for hdG discretisation - Example 4

different parameter α , but the proof of this, and even the question of whether this would have a positive impact, are open problems.

Taylor-Hood discretisation

We now turn to the following Taylor-Hood discretisation

$$\begin{aligned} \mathbf{TH}_h^3 &= \{ \mathbf{v}_h \in [H^1(\Omega)]^2 : \mathbf{v}_h|_K \in [\mathbb{P}_3(K)]^2 \quad \forall K \in \mathcal{T}_h \}, \\ R_{h,0}^2 &:= \left\{ q_h \in L^2(\Omega) : q_h \in \mathbb{P}_2(\Omega) \wedge \int_{\Omega} q_h \, d\mathbf{x} = 0 \right\}. \end{aligned}$$

Applying above discretisation to the Stokes equation leads to the following discrete problem:

find $(\mathbf{u}_h, p_h) \in \mathbf{TH}_h^3 \times R_{h,0}^2$ such that for all $(\mathbf{v}_h, q_h) \in \mathbf{TH}_h^3 \times R_{h,0}^2$

$$(4.9) \quad \begin{cases} \int_{\Omega} \nu \nabla \mathbf{u}_h : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_{\Omega} p_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} \\ - \int_{\Omega} \nabla \cdot \mathbf{u}_h q_h \, d\mathbf{x} = 0 \end{cases}.$$

Once again we start with the standard example for Stokes problem.

Example 5. We consider the driven cavity from Chapter 3.5 defined as a problem (3.10). In this experiment of the preconditioners for the GMRES iterative solver we use a random initial guess.

In Figure 4.4, as expected, we can see two zero eigenvalues corresponding to the two zero energy modes that are a motivation for the size of the coarse space.

DOF	N	One-level											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	11	12	19	20	14	13	18	20	13	14	19	19
69 891	16	23	28	41	47	29	32	46	53	30	33	46	50
156 579	36	34	43	66	75	45	55	78	95	48	53	74	88
277 763	64	48	59	93	109	65	109	120	184	69	85	105	130
623 619	144	72	92	151	168	117	184	205	322	118	136	177	301
1 107 459	256	98	292	207	247	173	277	305	485	174	231	263	460

DOF	N	Two-level (2 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	8	9	14	13	10	10	16	13	13	13	14	14
69 891	16	14	17	19	20	16	19	22	25	24	23	20	22
156 579	36	18	22	22	25	27	25	32	36	31	32	23	29
277 763	64	21	25	24	29	28	69	28	63	35	35	23	34
623 619	144	27	36	16	28	42	63	47	74	37	43	24	36
1 107 459	256	32	57	27	33	58	135	61	99	41	51	25	44

DOF	N	Two-level (5 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	6	8	11	12	8	9	13	12	13	12	13	13
69 891	16	11	14	14	14	11	13	14	15	26	22	16	17
156 579	36	14	17	15	15	17	18	19	18	36	30	16	18
277 763	64	17	20	16	16	12	29	15	32	40	35	16	19
623 619	144	21	28	17	18	21	36	23	39	41	42	17	20
1 107 459	256	25	32	17	19	28	50	27	52	45	43	17	22

DOF	N	Two-level (7 eigenvectors)											
		ORAS		SORAS		NVTF-MRAS		NVTF-SMRAS		TVNF-MRAS		TVNF-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	6	7	11	12	8	9	12	12	15	13	16	13
69 891	16	10	13	13	13	11	13	14	14	36	27	18	16
156 579	36	15	16	14	14	12	14	15	15	55	31	18	16
277 763	64	17	20	15	14	12	19	14	18	64	36	18	18
623 619	144	22	27	16	16	14	21	15	20	75	42	18	19
1 107 459	256	27	31	14	16	15	24	17	22	78	42	18	20

Table 4.4: Comparison of preconditioners for Taylor-Hood discretisation - Example 5

Table 4.4 shows a significant improvement of the convergence that is brought by the two-level

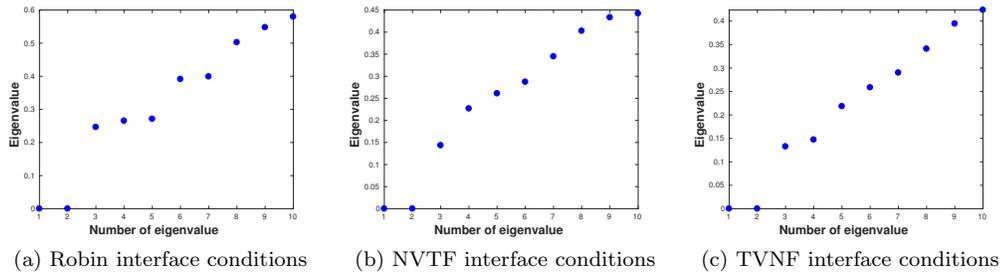


Figure 4.4: Eigenvalues on one of the floating subdomains in the case of uniform decomposition and Taylor-Hood discretisation - Example 5

methods. The results are somewhat less conclusive than for the hdG discretisation, in terms of number of iterations. On the other hand, we state the fact that the new preconditioners are parameter-free, which makes them more universal. Moreover, as expected, the iteration number does not increase with respect to the number of the subdomains. The size of coarse space seems to be sufficient as adding some more eigenvectors does not bring an improvement, while the computational time increases.

Example 4. Finally, we consider a T-shaped domain $\Omega = (0, 1.5) \times (0, 1) \cup (0.5, 1) \times (-1, 1)$, and we impose mixed boundary conditions (3.11). The numerical solution of this problem is depicted in Figure 3.12. We use zero as an initial guess for the GMRES iterative solver. The overlapping decomposition into subdomains is generated by METIS.

Once again a clustering of small eigenvalues of generalised eigenvalue problem defined in (4.4) is a motivation of the size of the coarse space (see Figure 4.5).

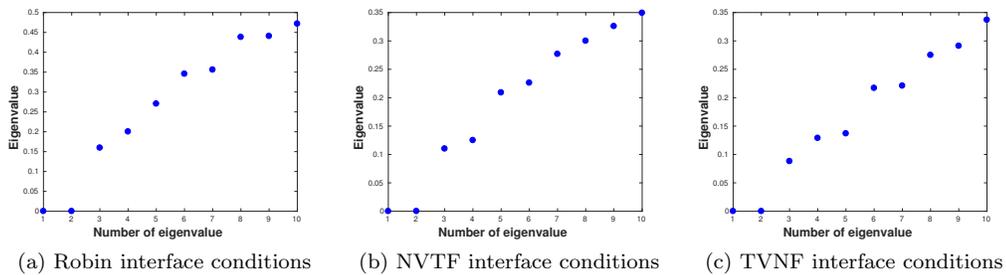


Figure 4.5: Eigenvalues on one of the floating subdomains in the case of METIS decomposition and Taylor-Hood discretisation - Example 4

The same as in all examples for Taylor-Hood discretisation we notice a significant improvement of the convergence when using two-level methods. Although from Table 4.5 we can see that the coarse spaces containing five eigenvectors seem to be sufficient. And again this is unexpected, since the viscous is not changing in any part of the domain.

		One-level					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
33 269	4	13	20	12	19	13	19
138 316	16	36	51	33	52	31	45
269 567	32	59	85	52	85	49	75
553 103	64	92	132	83	136	78	115
1 134 314	128	146	208	132	223	117	188
2 201 908	256	232	328	209	357	189	293

		Two-level (2 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
33 269	4	10	14	9	15	12	15
138 316	16	21	27	19	24	22	24
269 567	32	29	35	30	38	25	30
553 103	64	35	45	34	43	33	35
1 134 314	128	42	52	47	58	34	41
2 201 908	256	47	56	69	76	38	45

		Two-level (5 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
33 269	4	8	13	8	13	12	14
138 316	16	15	16	14	16	20	18
269 567	32	14	19	20	22	24	19
553 103	64	16	20	18	19	29	20
1 134 314	128	17	22	23	24	30	22
2 201 908	256	16	21	34	37	35	24

		Two-level (7 eigenvectors)					
DOF	N	ORAS	SORAS	NVTF-MRAS	NVTF-SMRAS	TVNF-MRAS	TVNF-SMRAS
33 269	4	8	13	8	12	12	15
138 316	16	15	15	15	16	23	19
269 567	32	18	16	14	16	23	19
553 103	64	23	17	16	17	26	21
1 134 314	128	30	21	17	18	31	20
2 201 908	256	36	22	19	20	32	22

Table 4.5: Comparison of preconditioners for Taylor-Hood discretisation - Example 4

4.4.2 Nearly incompressible elasticity

From a mathematical point of view the nearly incompressible elasticity problem is very similar to the Stokes equation. The difference is that instead of considering the gradient $\nabla \mathbf{v}$ we use the symmetric gradient $\varepsilon(\mathbf{v}) := \frac{1}{2}(\nabla \mathbf{v} + \nabla^T \mathbf{v})$. We want to solve the following two dimensional problem

$$(4.10) \quad \begin{cases} -2\mu \nabla \cdot \varepsilon(\mathbf{u}) + \nabla p = \mathbf{f} & \text{in } \Omega \\ -\nabla \cdot \mathbf{u} = \frac{1}{\lambda} p & \text{in } \Omega \end{cases},$$

where $\mathbf{u} : \bar{\Omega} \rightarrow \mathbb{R}^2$ is the unknown displacement field, $p : \Omega \rightarrow \mathbb{R}$ the pressure, λ and μ are the Lamé coefficients defined by

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

where E is the Young modulus and ν the Poisson ratio. We define the stress tensor as

$$\boldsymbol{\sigma}^{sym} := 2\mu\boldsymbol{\varepsilon}(\mathbf{u}) - p\mathbf{I}, \quad \boldsymbol{\sigma}_n^{sym} := \boldsymbol{\sigma}^{sym}\mathbf{n}.$$

hdG discretisation

Using the hdG discretisation for the problem (4.10) with $k = 1$ leads to the following discrete problem:

find $(\mathbf{u}_h, \tilde{u}_h, p_h) \in \mathbf{BDM}_h^1 \times M_h^0 \times Q_h^0$ such that for all $(\mathbf{v}_h, \tilde{v}_h, q_h) \in \mathbf{BDM}_h^1 \times M_h^0 \times Q_h^0$

$$(4.11) \quad \begin{cases} a_s((\mathbf{u}_h, \tilde{u}_h), (\mathbf{v}_h, \tilde{v}_h)) + b((\mathbf{v}_h, \tilde{v}_h), p_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, dx \\ b((\mathbf{u}_h, \tilde{u}_h), q_h) + c(p_h, q_h) = 0, \end{cases}$$

where

$$(4.12) \quad \begin{aligned} a_s((\mathbf{w}_h, \tilde{w}_h), (\mathbf{v}_h, \tilde{v}_h)) := & \sum_{K \in \mathcal{T}_h} \left(\int_K 2\mu\boldsymbol{\varepsilon}(\mathbf{w}_h) : \boldsymbol{\varepsilon}(\mathbf{v}_h) \, dx - \int_{\partial K} 2\mu(\boldsymbol{\varepsilon}_n(\mathbf{w}_h))_t ((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \right. \\ & - \int_{\partial K} 2\mu((\mathbf{w}_h)_t - \tilde{w}_h) (\boldsymbol{\varepsilon}_n(\mathbf{v}_h))_t \, ds \\ & \left. + 2\mu \frac{\tau}{h_K} \int_{\partial K} \Phi^{k-1}((\mathbf{w}_h)_t - \tilde{w}_h) \Phi^{k-1}((\mathbf{v}_h)_t - \tilde{v}_h) \, ds \right), \end{aligned}$$

b is defined by (1.19), and

$$c(r_h, q_h) := -\frac{1}{\lambda} \int_{\Omega} r_h q_h \, ds.$$

In the case of ORAS and SORAS we choose $\alpha = 10$ as in [HJN15] for the Robin interface conditions (4.13)

$$(4.13) \quad \int_{\partial\Omega_i \setminus \partial\Omega} \boldsymbol{\sigma}_n^{sym}(\mathbf{v}_h)_n \, ds + \int_{\partial\Omega_i \setminus \partial\Omega} 2\alpha \frac{\mu(2\mu + \lambda)}{\lambda + 3\mu} \mathbf{u}_h \mathbf{v}_h \, ds.$$

Fortunately, the MRAS and SMRAS preconditioners are parameter-free. We already know that for the Stokes equation these preconditioners are associated with NVTf and TVNF interface conditions. In the case of nearly incompressible elasticity we refer to them as normal-displacement and tangential-normal-stress (NDTNS) and tangential-displacement and normal-normal-stress (TDNNS) interface conditions. The second type of the boundary conditions has been already introduced for linear elasticity equation in [PS11].

Example 6. We consider the following problem on the unit square $\Omega = (0, 1)^2$

$$(4.14) \quad \begin{cases} -2\mu\nabla \cdot \boldsymbol{\varepsilon}(\mathbf{u}) + \nabla p = \mathbf{f} & \text{in } \Omega \\ -\nabla \cdot \mathbf{u} = \frac{1}{\lambda} p & \text{in } \Omega \\ \mathbf{u}(x, y) = (1, 0)^T & \text{on } \partial\Omega \cap \{y = 1\} \\ \mathbf{u}(x, y) = (0, 0)^T & \text{on } \partial\Omega \setminus \{y = 1\} \end{cases}.$$

The physical parameters are $E = 10^8$ and $\nu = 0.3$. In Figure 4.6 we plot the displacement field and pressure, after solving the problem numerically. For this numerical test we use a random initial guess for the GMRES iterative solver.

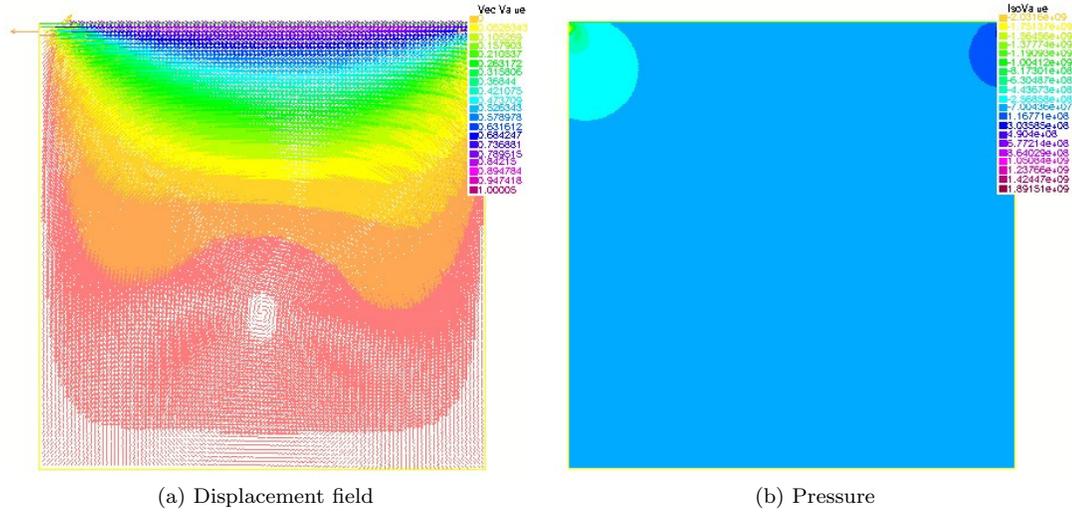


Figure 4.6: Numerical solution of the problem - Example 6

In Figure 4.7 we plot the eigenvalues only for floating subdomains. The clustering of small eigenvalues of the generalised eigenvalue problem defined in (4.4) suggests the number of eigenvectors to be added to the coarse space. The three zero eigenvalues correspond to what we call zero energy modes. In the case of the linear elasticity equation zero they are the rigid body modes, that is translations in the x and y directions, and a rotation.

DOF	N	One-level											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	14	15	35	43	23	24	53	54	26	18	44	48
70 880	16	52	90	79	91	44	50	123	126	108	41	111	113
159 120	36	131	277	123	130	64	72	203	195	253	66	183	172
282 560	64	256	530	166	194	87	104	294	279	487	86	261	260
635 040	144	>1000	>1000	251	288	131	146	487	450	>1000	142	428	424
1 128 320	256	>1000	>1000	362	385	182	197	666	645	>1000	207	597	591

DOF	N	Two-level (3 eigenvectors)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	9	11	27	26	16	15	32	32	17	13	29	32
70 880	16	22	39	40	37	28	26	58	54	45	31	56	52
159 120	36	114	133	43	44	36	35	62	66	88	39	62	65
282 560	64	286	340	47	52	42	37	66	72	167	44	65	73
635 040	144	>1000	>1000	50	61	50	50	67	86	336	63	65	84
1 128 320	256	>1000	>1000	51	63	52	46	66	76	589	69	65	82

DOF	N	Two-level (5 eigenvector)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	8	9	23	24	11	12	25	29	14	12	25	24
70 880	16	17	31	33	33	24	20	47	42	27	27	47	42
159 120	36	43	92	37	36	33	32	55	49	42	36	55	47
282 560	64	152	212	41	41	40	34	58	55	54	42	58	52
635 040	144	492	734	43	44	46	41	60	70	96	55	60	56
1 128 320	256	>1000	>1000	45	46	47	39	62	62	141	61	61	58

DOF	N	Two-level (7 eigenvectors)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 840	4	7	9	20	22	10	11	24	24	12	12	22	22
70 880	16	15	23	29	30	18	18	38	37	25	25	34	35
159 120	36	23	64	33	34	20	24	41	43	38	35	39	39
282 560	64	36	112	35	36	21	25	43	43	50	39	42	42
635 040	144	212	474	40	38	23	33	45	56	80	49	45	44
1 128 320	256	518	979	39	39	21	29	45	45	119	57	46	44

Table 4.6: Comparison of preconditioners for hdG discretisation - Example 6

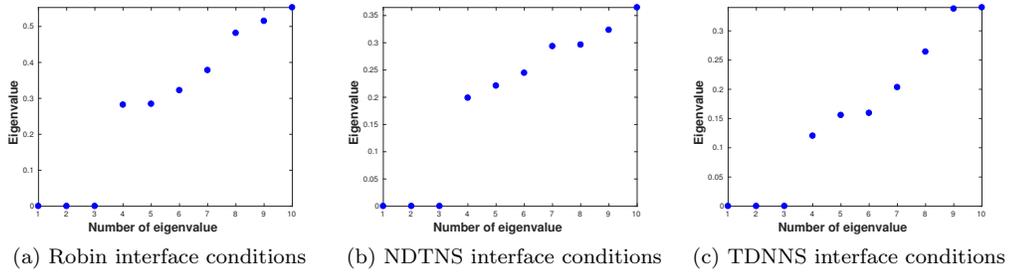


Figure 4.7: Eigenvalues on one of the floating subdomains in the case of uniform decomposition and hdG discretisation - Example 6

In the case of SMRAS and SORAS preconditioners an improvement of convergence caused by the two-level methods is significant as in Example 5 for Stokes equation. Although we can note from the results of Table 4.6 that coarse space improvement is visible for MRAS preconditioners and not for ORAS. For symmetric preconditioners (SMRAS and SORAS) three eigenvectors seem to lead already to satisfactory results. This size of a coarse space, as expected, is sufficient,

since the physical coefficients of the problem are not changing inside domain.

Example 7. We consider a heterogeneous beam with ten layers of steel and rubber. Five layers are made from steel with the physical parameters $E = 210 \cdot 10^9$ and $\nu = 0.3$, and other five are made from rubber with the physical parameters $E = 10^8$ and $\nu = 0.4999$ as is depicted in Figure 4.8a. A similar example was considered in [HJN15]. The computational domain is

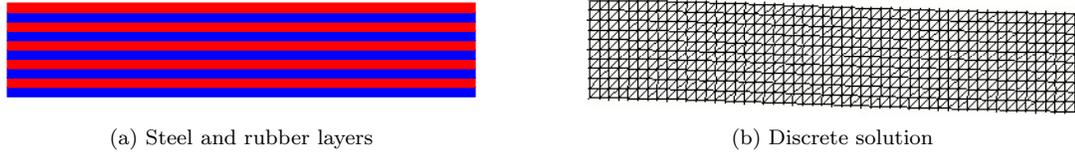


Figure 4.8: Heterogeneous beam

the rectangle $\Omega = (0, 5) \times (0, 1)$. The beam is clamped on its left side, hence we consider the following problem

$$(4.15) \quad \begin{cases} -2\mu \nabla \cdot \varepsilon(\mathbf{u}) + \nabla p & = \mathbf{f} & \text{in } \Omega \\ -\nabla \cdot \mathbf{u} & = \frac{1}{\lambda} p & \text{in } \Omega \\ \mathbf{u}(x, y) & = (0, 0)^T & \text{on } \partial\Omega \cap \{x = 0\} \\ \boldsymbol{\sigma}_n^{sym}(x, y) & = (0, 0)^T & \text{on } \partial\Omega \setminus \{x = 0\} \end{cases}.$$

In Figure 4.8b we plot the mesh of the bent beam, after solving the problem numerically. We use zero as an initial guess for the GMRES iterative solver. The overlapping decomposition into subdomains is generated by METIS.

Unlike in the homogeneous case, we do not notice a clear clustering of the eigenvalues (see Figure 4.9), this is due to the strong heterogeneity of the problem.

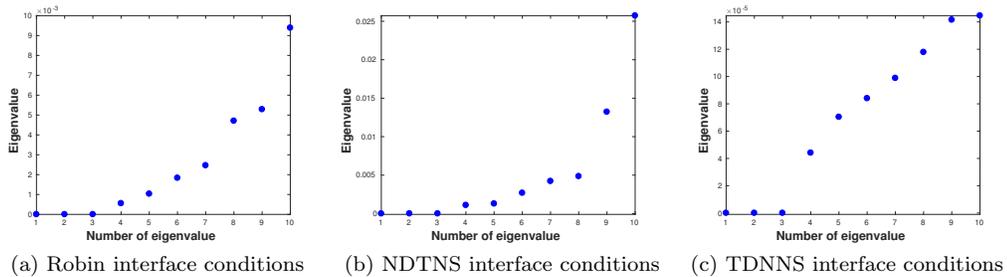


Figure 4.9: Eigenvalues on one of the floating subdomains in the case of METIS decomposition and hdG discretisation - Example 7

We notice an improvement only when using a coarse space which is sufficiently big (see Table 4.7). That is because of the various values of the physical parameters inside domain that can also change across the interfaces. Furthermore, we get a stable number of iterations only

		One-level					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
46 777	8	196	440	189	402	186	463
88 720	16	317	602	330	582	326	666
179 721	32	537	>1000	574	>1000	587	>1000
353 440	64	899	>1000	847	>1000	846	>1000
704 329	128	>1000	>1000	>1000	>1000	>1000	>1000
1 410 880	256	>1000	>1000	>1000	>1000	>1000	>1000

		Two-level (3 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
46 777	8	183	356	178	326	175	374
88 720	16	267	436	293	432	289	472
179 721	32	457	749	515	733	536	853
353 440	64	705	870	720	918	718	>1000
704 329	128	>1000	>1000	>1000	>1000	>1000	>1000
1 410 880	256	>1000	>1000	>1000	>1000	>1000	>1000

		Two-level (5 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
46 777	8	168	255	162	230	161	275
88 720	16	244	313	273	299	262	346
179 721	32	385	525	442	458	469	587
353 440	64	514	444	551	526	590	558
704 329	128	835	557	782	684	765	832
1 410 880	256	>1000	567	>1000	694	844	821

		Two-level (7 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
46 777	8	148	197	149	192	158	231
88 720	16	205	201	286	187	283	273
179 721	32	318	337	385	301	433	419
353 440	64	403	262	397	247	460	389
704 329	128	490	168	447	182	558	443
1 410 880	256	>1000	116	387	138	473	298

Table 4.7: Comparison of preconditioners for hdG discretisation - Example 7

for the symmetric preconditioners (SMRAS and SORAS), and the coarse space improvement in case of ORAS preconditioner is much less visible than in case of MRAS preconditioners. This may be due to the fact we have not chosen an optimal parameter in the Robin interface conditions (4.13).

Taylor-Hood discretisation

Using the Taylor-Hood discretisation of the problem (4.10) leads to the following discrete problem:

find $(\mathbf{u}_h, p_h) \in \mathbf{TH}_h^3 \times R_h^2$ such that for all $(\mathbf{v}_h, q_h) \in \mathbf{TH}_h^3 \times R_h^2$

$$(4.16) \quad \begin{cases} \int_{\Omega} 2\mu \boldsymbol{\varepsilon}(\mathbf{u}_h) : \boldsymbol{\varepsilon}(\mathbf{v}_h) \, d\mathbf{x} - \int_{\Omega} p_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, d\mathbf{x} \\ - \int_{\Omega} \nabla \cdot \mathbf{u}_h q_h \, d\mathbf{x} - \frac{1}{\lambda} \int_{\Omega} p_h q_h \, d\mathbf{x} = 0. \end{cases}$$

Example 6. We consider problem (4.14) and we use a random initial guess for the GMRES iterative solver.

As we already know, the easiest way to build a coarse space is to incorporate zero energy modes in it. Thus, we start looking for the improvement considering three eigenvectors that are associated with zero eigenvalues on the floating subdomain (see Figure 4.10).

DOF	N	One-level											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	8	9	22	25	17	17	29	32	15	17	28	31
69 891	16	15	20	42	53	37	42	59	71	26	35	56	66
156 579	36	24	34	66	75	50	53	92	95	33	58	93	112
277 763	64	30	81	88	92	68	80	122	143	50	77	124	171
623 619	144	59	356	127	138	101	121	193	233	78	145	183	288
1 107 459	256	>1000	>1000	163	201	134	176	265	344	104	187	235	422

DOF	N	Two-level (3 eigenvectors)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	6	7	14	16	9	10	16	18	10	13	17	21
69 891	16	10	13	19	21	17	17	25	27	24	18	25	28
156 579	36	14	16	20	22	18	19	24	29	32	27	28	42
277 763	64	19	27	20	25	19	22	23	36	35	24	28	47
623 619	144	26	32	22	26	20	32	23	44	41	37	29	74
1 107 459	256	33	123	23	31	20	39	23	71	43	58	28	95

DOF	N	Two-level (5 eigenvectors)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	6	6	13	14	7	8	15	16	9	11	14	17
69 891	16	9	11	17	18	13	13	22	22	18	14	19	23
156 579	36	13	17	18	18	16	15	21	23	24	23	22	35
277 763	64	16	19	19	19	16	16	21	25	31	20	22	28
623 619	144	20	26	21	21	18	25	21	38	38	29	24	62
1 107 459	256	26	36	22	22	18	32	21	53	43	46	24	84

DOF	N	Two-level (7 eigenvectors)											
		ORAS		SORAS		NDTNS-MRAS		NDTNS-SMRAS		TDNNS-MRAS		TDNNS-SMRAS	
		Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS	Unif	MTS
17 699	4	5	6	12	13	6	7	13	14	9	10	13	15
69 891	16	8	10	14	17	11	12	20	19	16	13	18	19
156 579	36	11	14	16	16	10	11	17	19	21	22	20	31
277 763	64	15	18	18	17	10	13	17	19	27	19	20	24
623 619	144	21	24	19	18	10	20	17	32	35	24	22	59
1 107 459	256	25	29	20	19	10	28	17	45	41	35	22	76

Table 4.8: Comparison of preconditioners for Taylor-Hood discretisation - Example 6

The conclusions remain the same as for Example 5 for the Stokes equation discretised by Taylor-Hood method since Tables 4.4 and 4.8 show similar results.

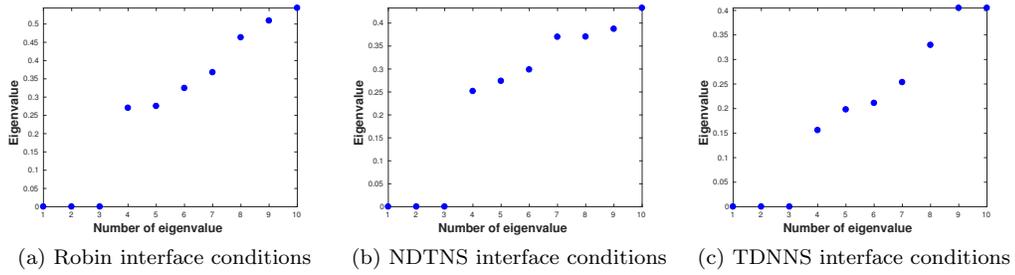


Figure 4.10: Eigenvalues on one of the floating subdomains in the case of uniform decomposition and Taylor-Hood discretisation - Example 6

Example 7. We consider the heterogeneous beam with ten layers of steel and rubber that is defined as a problem (4.15), and we use zero as an initial guess for the GMRES iterative solver. The overlapping decomposition into subdomains is generated by METIS.

As it was observed when this problem was approximated using the hdG discretisation, since the problem has strong heterogeneity, we are unable to see a clear clustering of the eigenvalues (see Figure 4.11).

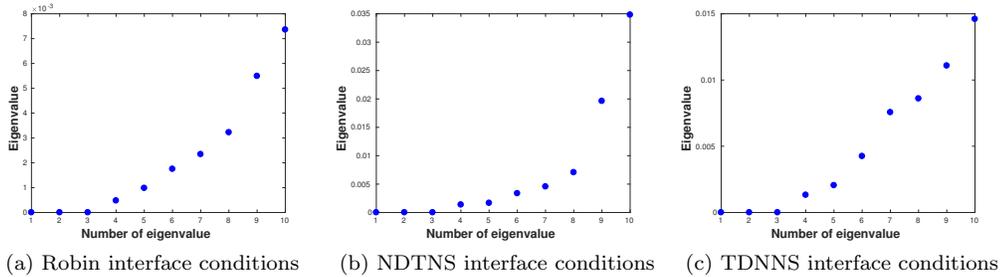


Figure 4.11: Eigenvalues on one of the floating subdomains in the case of METIS decomposition and Taylor-Hood discretisation - Example 7

The same as previous case we notice a significant improvement of the convergence when using two-level methods (see Table 4.9). Although we get a stable number of iterations only while considering coarse space which is sufficiently big.

		One-level					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
44 963	8	168	301	160	267	177	264
87 587	16	226	490	245	462	229	424
177 923	32	373	711	447	684	440	672
347 651	64	615	>1000	728	>1000	746	>1000
707 843	128	973	>1000	>1000	>1000	>1000	>1000
1 385 219	256	>1000	>1000	>1000	>1000	>1000	>1000

		Two-level (3 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
44 963	8	151	216	146	192	166	181
87 587	16	191	342	218	319	202	268
177 923	32	306	467	371	444	386	449
347 651	64	437	616	531	598	615	611
707 843	128	604	765	677	760	860	804
1 385 219	256	756	782	689	831	>1000	805

		Two-level (5 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
44 963	8	109	160	136	147	148	136
87 587	16	136	204	192	200	181	184
177 923	32	193	291	296	275	326	276
347 651	64	260	304	363	282	491	299
707 843	128	412	356	420	369	601	346
1 385 219	256	379	414	448	400	711	317

		Two-level (7 eigenvectors)					
DOF	N	ORAS	SORAS	NDTNS-MRAS	NDTNS-SMRAS	TDNNS-MRAS	TDNNS-SMRAS
44 963	8	76	118	124	115	133	103
87 587	16	106	146	166	138	159	123
177 923	32	157	202	203	185	302	214
347 651	64	178	191	225	170	326	182
707 843	128	140	114	153	112	266	122
1 385 219	256	119	86	118	77	259	94

Table 4.9: Comparison of preconditioners for Taylor-Hood discretisation - Example 7

4.5 Summary

We tested numerically two-level preconditioners with coarse spaces associated to local generalised eigenvalue problems for mixed problems. We chose the equations associated to Stokes (Section 4.4.1), and nearly incompressible elasticity (Section 4.4.2). We considered two finite element discretisations, namely, the hdG and Taylor-Hood discretisations.

The improvement of the convergence in the case of Stokes flow is visible only when using at least five eigenvectors. This is unexpected, since as later can be observed for nearly incompressible elasticity, three eigenvectors should be enough. The reason for that can be associated with the nature of the problems and the incompressibility condition, since it is the main difference between both cases. For Taylor-Hood discretisation taking a sufficiently big coarse space we were able to achieve good scalability for all preconditioners. In the case of the hdG discretisation, coarse spaces applied to our preconditioners gave satisfactory results. Unfortunately, ORAS and SORAS do not perform in an optimal way, this behaviour might be caused by a non optimal choice of the parameter in the Robin interface conditions.

In the case of the homogeneous nearly incompressible elasticity the two-level methods coupled with symmetrised variants of ORAS and MRAS preconditioners allowed us to achieve good scalability results for both discretisations. Moreover, for these symmetric preconditioners coarse spaces containing only three eigenvectors seem to be enough. For the heterogeneous problem we also achieved scalability for two-level SORAS and SMRAS preconditioners, but as expected only in the case when the size of the coarse space is sufficiently big.

We can conclude that the behaviour of the two-level preconditioners associated with non standard interface conditions is at least as good as the two-level ones coupled with Robin interface conditions when the choice of the parameter is optimal. It shows an important advantage of our preconditioners as they are parameter-free.

Numerical tests have shown that the coarse spaces bring a significant improvement in the convergence, but the size of the coarse space depends on the problem. Building as small as possible coarse spaces is important from computational point of view. Thus, it is necessary to investigate what could be an optimal criterion of choosing the eigenvectors for a coarse space.

Conclusions

Summary

In this work we introduced hdG methods for Stokes equation with TVNF and NVTF boundary conditions. The degrees of freedom of these non standard boundary conditions were defined on interfaces of each element in the formulation of the hdG methods. Our approach including the edge projection allowed to further decrease the number of degrees of freedom. We proved well-posedness of the discrete problem and estimated the error of the discrete solution for both boundary value problems. The various numerical experiments confirmed the theory and the h^2 optimal convergence of the velocity error with respect to the L^2 norm. Even if we focused the analysis on the Stokes equations, this approach can be easily applied to the case of incompressible, or nearly incompressible, elasticity.

In addition to the above, we extended the approach and presented the equal-order polynomial approximation spaces for velocity and pressure. The proof of the well-posedness of the discrete problem and error analysis of the discrete solution are provided. This stabilised hdG method was also tested numerically and demonstrated good stability. It is possible to apply this stabilisation to other discontinuous Galerkin methods that use the divergence-conforming spaces.

We introduced new domain decomposition preconditioners that are associated with the non standard interface conditions. Thanks to the hdG discretisation we could consider general shape decompositions, even on the non-structured meshes. Numerical experiments showed the improvement in convergence of the iterative solver due to the non standard interface conditions. Moreover the hybrid discontinuous Galerkin discretisation performed far better than the Taylor-Hood one for the standard Restricted Additive Schwarz preconditioner. The use of Taylor-Hood discretisation required solving the problem of imposing these non standard boundary conditions on interfaces, which is a side effect of this work.

Finally, we investigated two-level methods for the newly introduced preconditioners with non standard interface conditions. We tested them numerically in comparison to the similar approach with Robin interface conditions. Numerical tests demonstrated a significant improvement when using the coarse spaces associated with the local general eigenvalue problems. The symmetrised variants of the preconditioners with non standard and Robin interface conditions

allowed us to achieve good scalability results. In some of the examples we noticed that the additional work on the choice of optimal parameter for Robin interface conditions is required. Hence, we see the advantage of the preconditioners with non standard interface conditions, since they are parameter-free.

Future work

In the future it would be important to prove the well-posedness of the continuous TVNF boundary value problem (1.4) and NVTF one (1.5). This would imply proving the Poincaré inequality for spaces where one of the normal or tangential components is fixed on the boundary of the domain. In this work we were more interested in numerical results since we wanted to test a new kind of domain decomposition preconditioners. Although we can expect that better understanding the nature of these non standard boundary conditions might help in proving the optimal convergence of the L^2 error.

Using the Newton's method we are able to solve the Navier-Stokes equations. This approach requires solving Oseen problems, in addition to Stokes equations. We saw that the stability and error estimates for the Stokes problems depend on the coefficients of the problem. Since Oseen problems contain in addition the convection term that can dominate, we would have to apply the stabilised methods. Further analysis of these methods would be the next step in extending this approach for the Navier-Stokes equations.

In the case of domain decomposition preconditioners there are two directions of further investigation. The first one is to apply presented approach to the three dimensional nearly incompressible elasticity and Stokes problems. It would lead also to the parallel implementation for construction of the preconditioners and coarse spaces. On the other hand, it is very important to obtain improvement of the convergence with a coarse space that is as small as possible. Hence, to optimise the criterion of choosing the eigenvectors that build a coarse space could be an interesting issue.

A theoretical study of multilevel methods requires proving the estimates that control the spectrum of the preconditioned operator. This is done for systems where the global matrix is symmetric positive definite. However, it is not the case of saddle point problems such as nearly incompressible elasticity and Stokes. Since the numerical results showed the good behaviour for such systems, it would be interesting to prove these estimates in the case of the non-symmetric problems.

Appendices

Appendix A

Matlab implementations

A.1 One dimensional hdG method for elliptic problem

Let us consider one dimensional Poisson equation with Dirichlet boundary condition. We present the implementation of the one dimensional hdG method similar to the one presented in [Riv08, Chapter 1] that has been done for the discontinuous Galerkin method. We want to solve the following problem on $\Omega = (0, 1)$ for $f(x) = 2e^{-x^2}(x+1)(2x^2 - 4x + 1)$

$$(1.1) \quad \begin{cases} -u'' &= f & \text{in } (0, 1) \\ u(0) &= 1 \\ u(1) &= 0 \end{cases}.$$

Let $0 = x_0 < x_1 < \dots < x_N = 1$ be partition of our domain $\Omega = (0, 1)$ such that $x_n = x_0 + nh$ and $I_n = (x_n, x_{n+1})$. Hence the triangulation is defined as a set $\mathcal{T}_h = \{I_n\}_{n=0}^{N-1}$, the set of edges is defined as $\mathcal{E}_h = \{x_n\}_{n=0}^N$ and $h = \frac{1}{N}$. We use the following discrete space

$$\begin{aligned} V_h &= \{v_h \in C^2(\Omega) \mid v_h|_{I_n} \in \mathbb{P}_1(I_n) \ \forall 0 \leq n < N\}, \\ \tilde{V}_h^{(g_0, g_N)} &= \{\tilde{v}_h \in L^2(\mathcal{E}_h) \mid \tilde{v}_h(x_n) \in \mathbb{P}_0(x_n) \ \forall 0 \leq n \leq N \wedge \tilde{v}_h(0) = g_0, \tilde{v}_h(1) = g_N\}. \end{aligned}$$

Let us denote $v_h(x_n^+) = \lim_{\varepsilon \rightarrow 0^+} v_h(x_n + \varepsilon)$, $v_h(x_n^-) = \lim_{\varepsilon \rightarrow 0^+} v_h(x_n - \varepsilon)$. We define following bilinear form

$$\begin{aligned} a((w_h, \tilde{w}_h), (v_h, \tilde{v}_h)) &= \sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} w'_h(x) v'_h(x) dx \\ &\quad - (w'_h(x_{n+1}^-) (v_h(x_{n+1}^-) - \tilde{v}_h(x_{n+1})) - w'_h(x_n^+) (v_h(x_n^+) - \tilde{v}_h(x_n))) \\ &\quad + \varepsilon (v'_h(x_{n+1}^-) (w_h(x_{n+1}^-) - \tilde{w}_h(x_{n+1})) - v'_h(x_n^+) (w_h(x_n^+) - \tilde{w}_h(x_n))) \\ &\quad + \frac{\tau}{h} (w_h(x_{n+1}^-) - \tilde{w}_h(x_{n+1})) (v(x_{n+1}^-) - \tilde{v}_h(x_{n+1})) \end{aligned}$$

$$+ \frac{\tau}{h} (w_h(x_n^+) - \tilde{w}_h(x_n)) (v_h(x_n^+) - \tilde{v}_h(x_n)),$$

were $\varepsilon \in \{-1, 1\}$ and τ is a stabilisation parameter. In addition, we define the right hand side as

$$L(v_h) = \int_0^1 f(x) v_h(x) dx$$

Hence, we define the following discrete problem

Find $(u_h, \tilde{u}_h) \in V_h \times \tilde{V}_h^{(1,0)}$ such that for all $(v_h, \tilde{v}_h) \in V_h \times \tilde{V}_h^{(0,0)}$

$$(1.2) \quad a((u_h, \tilde{u}_h), (v_h, \tilde{v}_h)) = L(v_h).$$

Before we build the linear system we need to define the basis functions. That is why, for all $0 \leq n < N$ $\mathbb{P}_1(I_n) = \{\phi_0^n, \phi_1^n\}$ such that for all $x \in I_n$

$$\phi_0^n(x) = \frac{x_{n+1} - x}{h}, \quad (\phi_0^n)'(x) = -\frac{1}{h}, \quad \phi_1^n(x) = \frac{x - x_n}{h}, \quad (\phi_1^n)'(x) = \frac{1}{h}.$$

Therefore the global basis functions Φ_i^n for discrete space V_h and Ψ_i^n for discrete space \tilde{V}_h are following

$$\Phi_i^n = \begin{cases} \phi_i^n(x) & x \in I_n \\ 0 & x \notin I_n \end{cases}, \quad \Psi_i^n = \mathbb{1}_{\{x_{n+i}\}}.$$

For all $x \in (0, 1)$ the solution of (1.2) is following

$$(1.3) \quad u_h(x) = \sum_{m=0}^{N-1} \sum_{i=0}^1 \alpha_i^m \Phi_i^m(x), \quad \tilde{u}_h(x) = \sum_{m=0}^N \sum_{i=0}^1 \beta_i^m \Psi_i^m(x).$$

Hence, on each element $I_n \cup \{x_n, x_{n+1}\}$ for $1 \leq n < N$ we have left hand side defined as

$$\begin{aligned} & \int_{x_n}^{x_{n+1}} (\phi_j^n)'(x) (\phi_i^n)'(x) dx \\ & - (\phi_j^n)'(x_{n+1}^-) (\phi_i^n(x_{n+1}^-) - \psi_1^n(x_{n+1})) + (\phi_j^n)'(x_n^+) (\phi_i^n(x_n^+) - \psi_0^n(x_n)) \\ & + \varepsilon (\phi_j^n(x_{n+1}^-) - \psi_1^n(x_{n+1})) (\phi_i^n)'(x_{n+1}^-) - \varepsilon (\phi_j^n(x_n^+) - \psi_0^n(x_n)) (\phi_i^n)'(x_n^+) \\ & + \frac{\tau}{h} (\phi_j^n(x_{n+1}^-) - \psi_1^n(x_{n+1})) (\phi_i^n(x_{n+1}^-) - \psi_1^n(x_{n+1})) \\ & + \frac{\tau}{h} (\phi_j^n(x_n^+) - \psi_0^n(x_n)) (\phi_i^n(x_n^+) - \psi_0^n(x_n)). \end{aligned}$$

The above local problem we can split for five parts. That is why, we define the 4×4 local matrices that rows and columns are associated with following basis functions $\phi_0^n, \phi_1^n, \psi_0^n, \psi_1^n$. The first matrix is associated with the first integral term

$$(\mathbf{A}_n)_{ij} = \int_{x_n}^{x_{n+1}} (\phi_j^n)'(x) (\phi_i^n)'(x) dx$$

Then the local matrix follows

$$\mathbf{A}_n = \frac{1}{h} \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The next parts are defined as follows

$$\begin{aligned} (\mathbf{B}_n)_{ij} &= (\phi_j^n)'(x_n^+) (\phi_i^n(x_n^+) - \psi_0^n(x_n)) - \varepsilon (\phi_j^n(x_n^+) - \psi_0^n(x_n)) (\phi_i^n)'(x_n^+) \\ &\quad + \frac{\tau}{h} \psi_1^n(x_{n+1}) \psi_1^n(x_{n+1}) + \frac{\tau}{h} (\phi_j^n(x_n^+) - \psi_0^n(x_n)) (\phi_i^n(x_n^+) - \psi_0^n(x_n)), \\ (\mathbf{C}_n)_{ij} &= -(\phi_j^n)'(x_{n+1}^-) \phi_i^n(x_{n+1}^-) + \varepsilon \phi_j^n(x_{n+1}^-) (\phi_i^n)'(x_{n+1}^-) + \frac{\tau}{h} \phi_j^n(x_{n+1}^-) \phi_i^n(x_{n+1}^-), \\ (\mathbf{D}_n)_{ij} &= -\varepsilon \psi_1^n(x_{n+1}) (\phi_i^n)'(x_{n+1}^-) - \frac{\tau}{h} \psi_1^n(x_{n+1}) \phi_i^n(x_{n+1}^-), \\ (\mathbf{E}_n)_{ij} &= (\phi_j^n)'(x_{n+1}^-) \psi_1^n(x_{n+1}) - \frac{\tau}{h} \phi_j^n(x_{n+1}^-) \psi_1^n(x_{n+1}). \end{aligned}$$

And the local matrices follow

$$\begin{aligned} \mathbf{B}_n &= \frac{1}{h} \begin{pmatrix} -1 + \varepsilon + \tau & 1 & -\varepsilon - \tau & 0 \\ -\varepsilon & 0 & \varepsilon & 0 \\ 1 - \tau & -1 & \tau & 0 \\ 0 & 0 & 0 & \tau \end{pmatrix}, & \mathbf{C}_n &= \frac{1}{h} \begin{pmatrix} 0 & -\varepsilon & 0 & 0 \\ 1 & -1 + \varepsilon + \tau & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \\ \mathbf{D}_n &= \frac{1}{h} \begin{pmatrix} 0 & 0 & 0 & \varepsilon \\ 0 & 0 & 0 & -\varepsilon - \tau \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, & \mathbf{E}_n &= \frac{1}{h} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 1 - \tau & 0 & 0 \end{pmatrix}. \end{aligned}$$

Assuming that the vector of unknowns is in following order

$$(\alpha_0^0, \alpha_1^0, \dots, \alpha_0^{N-1}, \alpha_1^{N-1}, \beta_0^0, \beta_1^0 + \beta_0^1, \dots, \beta_1^{N-2} + \beta_0^{N-1}, \beta_0^N)^T,$$

then for $0 \leq n < N$ we can construct the local matrices \mathbf{M}_n as follows

$$\mathbf{M}_n = \mathbf{A}_n + \mathbf{B}_n + \mathbf{C}_{n+1} + \mathbf{D}_{n+1} + \mathbf{E}_{n+1}$$

Then the global matrix \mathbf{M} is constructed in such way that the rows and columns of the local matrix \mathbf{M}_n has following global numbering $[2n + 1, 2n + 2, 2N + n + 1, 2N + n + 2]^2$.

To compute the right hand side, we use the trapezoidal rule

$$\int_0^1 f(x) \Phi_i^n(x) dx \approx \frac{1}{2N} \sum_{j=1}^{N-1} f(x_{n+1}) \Phi(x_{n+1}) + f(x_n) \Phi(x_n).$$

Then the right hand side vector $\mathbf{b} = (\mathbf{b}_0^0, \mathbf{b}_1^0, \mathbf{b}_0^1, \mathbf{b}_1^1, \dots, \mathbf{b}_0^{N-1}, \mathbf{b}_1^{N-1}, \mathbf{b}^{2N+1}, \dots, \mathbf{b}^{3N+1})$, where

$$\forall 0 \leq n < N \quad \forall 0 \leq i \leq 1 \quad \mathbf{b}_i^n = \frac{1}{2N} f(x_{n+i}) \quad \text{and} \quad \forall 0 \leq n \leq N \quad \mathbf{b}^{2N+1+n} = 0.$$

The boundary conditions can be imposed strongly or by penalisation. As we can see on the Figures A.1 and A.2 is only depending on the personal preferences.

Below we present the convergence plots of the error for the above implementation in Matlab. The Figure A.1 depicts the symmetric and non-symmetric formulation of hdG method with strongly imposed boundary conditions. In addition, the Figure A.2 depicts the symmetric and

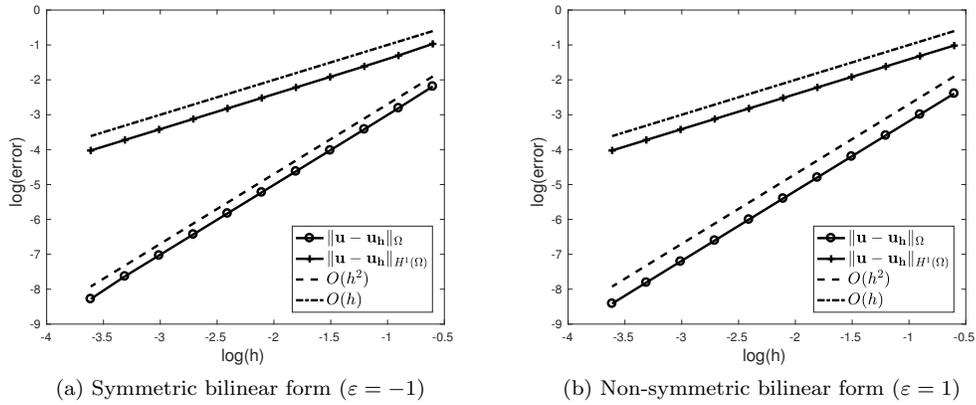


Figure A.1: One dimensional Matlab implementation - error convergence of the hdG method with strongly imposed boundary conditions

non-symmetric formulation of hdG method with boundary conditions imposed by penalisation. For comparison we present also the similar convergence plots of the error of dG implementation in Matlab from [Riv08, Appendix B.1]. The Figure A.3 depicts the symmetric and non-symmetric formulation of dG method. In all cases the penalisation parameter $\tau = 6$.

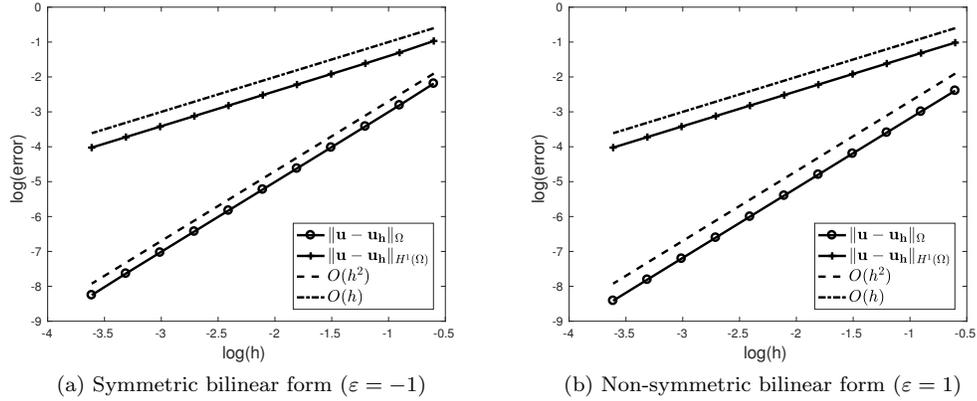


Figure A.2: One dimensional Matlab implementation - error convergence of the hdG method with boundary conditions imposed by penalisation

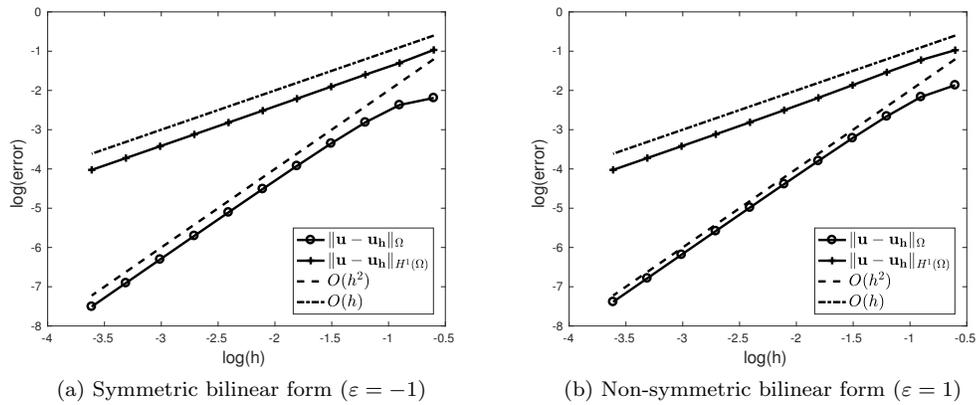


Figure A.3: One dimensional Matlab implementation - error convergence of the dG method

A.2 Two dimensional hdG method for elliptic problem

The model problem is two dimensional Poisson equation with homogeneous Dirichlet boundary condition.

$$(1.4) \quad \begin{cases} -\Delta \mathbf{u} = \mathbf{f} & \text{in } \Omega = (0, 1)^2 \\ \mathbf{u} = \mathbf{0} & \text{on } \partial\Omega \end{cases},$$

where $f(x, y) = 2x(1 - x) + 2y(1 - y)$.

Let \mathcal{T}_h be a given triangulation of Ω and \mathcal{E}_h be a set of edges of triangulation \mathcal{T}_h . Let us consider the following discrete space

$$V_h = \{\mathbf{v}_h \in H^1(\mathcal{T}_h) \mid \forall K \in \mathcal{T}_h \quad \mathbf{v}_h|_K \in \mathbb{P}_1(K)\},$$

$$\tilde{V}_{h,0} = \{\tilde{\mathbf{v}}_h \in L^2(\mathcal{E}_h) \forall E \in \mathcal{E}_h \tilde{\mathbf{v}}_h|_E \in \mathbb{P}_1(E) \wedge \tilde{\mathbf{v}}_h = 0 \text{ on } \partial\Omega\}.$$

Now we define the following bilinear form

$$\begin{aligned} a((\mathbf{w}_h, \tilde{\mathbf{w}}_h), (\mathbf{v}_h, \tilde{\mathbf{v}}_h)) &= \sum_{K \in \mathcal{T}_h} \int_K \nabla \mathbf{w}_h \nabla \mathbf{v}_h \, dx - \int_{\partial K} \partial_n \mathbf{w}_h (\mathbf{v}_h - \tilde{\mathbf{v}}_h) \, ds \\ &+ \varepsilon \int_{\partial K} (\mathbf{w}_h - \tilde{\mathbf{w}}_h) \partial_n \mathbf{v}_h \, ds + \frac{\tau}{h} \int_{\partial K} (\mathbf{w}_h - \tilde{\mathbf{w}}_h) (\mathbf{v}_h - \tilde{\mathbf{v}}_h) \, ds, \end{aligned}$$

where $\varepsilon \in \{-1, 1\}$ and τ is stabilisation parameter. Moreover, we define the right hand side as

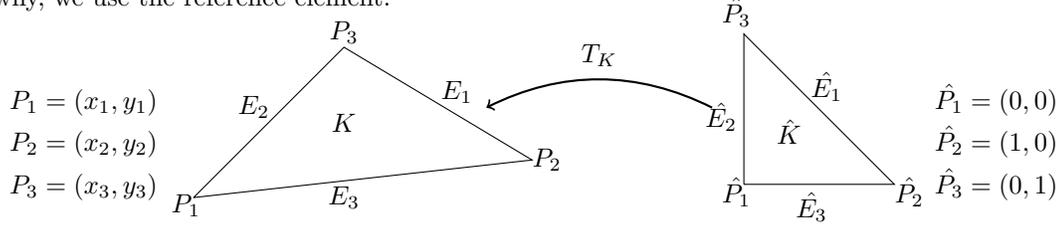
$$L(\mathbf{v}_h) = \int_{\Omega} \mathbf{f} \mathbf{v}_h \, dx.$$

Hence, we define the following discrete problem

Find $(\mathbf{u}_h, \tilde{\mathbf{u}}_h) \in V_h \times \tilde{V}_{h,0}$ such that for all $(\mathbf{v}_h, \tilde{\mathbf{v}}_h) \in V_h \times \tilde{V}_{h,0}$

$$(1.5) \quad a((\mathbf{u}_h, \tilde{\mathbf{u}}_h), (\mathbf{v}_h, \tilde{\mathbf{v}}_h)) = L(\mathbf{v}_h).$$

Computing the integrals on each elements of given triangulation would be very costly. That is why, we use the reference element.



Let T_K be a transformation for each triangle $K \in \mathcal{T}_h$ such as

$$\begin{pmatrix} x \\ y \end{pmatrix} = T_K \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} + \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \mathcal{B}_K \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} + \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}.$$

Now we can define local basis function for reference triangle \hat{K} and edges $\{E_1, E_2, E_3\}$

$$\begin{aligned} \lambda_1(\hat{x}, \hat{y}) &= 1 - \hat{x} - \hat{y}, & \lambda_2(\hat{x}, \hat{y}) &= \hat{x}, & \lambda_3(\hat{x}, \hat{y}) &= \hat{y}, \\ \psi_4(x, y) &= \frac{(x_3 - x) + (y_3 - y)}{(x_3 - x_2) + (y_3 - y_2)}, & \psi_5(x, y) &= \frac{(x - x_2) + (y - y_2)}{(x_3 - x_2) + (y_3 - y_2)}, & \text{if } (x, y) \in E_1, \\ \psi_6(x, y) &= \frac{(x_1 - x) + (y_1 - y)}{(x_1 - x_3) + (y_1 - y_3)}, & \psi_7(x, y) &= \frac{(x - x_3) + (y - y_3)}{(x_1 - x_3) + (y_1 - y_3)}, & \text{if } (x, y) \in E_2, \\ \psi_8(x, y) &= \frac{(x_2 - x) + (y_2 - y)}{(x_2 - x_1) + (y_2 - y_1)}, & \psi_9(x, y) &= \frac{(x - x_1) + (y - y_1)}{(x_2 - x_1) + (y_2 - y_1)}, & \text{if } (x, y) \in E_3. \end{aligned}$$

And global basis functions $\mathbb{P}_1(\mathcal{T}_h) = \{\Phi_1^K, \Phi_2^K, \Phi_3^K\}_{K \in \mathcal{T}_h}$ and $\mathbb{P}_1(\mathcal{E}_h) = \{\Psi_1^E, \Psi_2^E\}_{E \in \mathcal{E}_h}$ for

discrete space V_h

$$\Phi_i^K = \begin{cases} \lambda_i \circ T_K^{-1} & \begin{pmatrix} x \\ y \end{pmatrix} \in K \\ 0 & \text{otherwise} \end{cases} \quad \Psi_i^E = \begin{cases} \sum_{k \in \{k', k''\}} \psi_{3+2(k-1)+\gamma(i)} & \begin{pmatrix} x \\ y \end{pmatrix} \in \partial K^{k'} \cap \partial K^{k''} \\ 0 & \text{otherwise} \end{cases}$$

where $\gamma(i) = i$ for $K^{k'}$ and $\gamma(i) = 1 + (i \bmod 2)$ for $K^{k''}$.

Then, the solution of (1.5) is following, for all $\mathbf{x} \in \Omega$,

$$(1.6) \quad \mathbf{u}_h(\mathbf{x}) = \sum_{K \in \mathcal{T}_h} \sum_{i=1}^3 \alpha_i^K \Phi_i^K(\mathbf{x}) \quad \tilde{\mathbf{u}}_h(\mathbf{x}) = \sum_{E \in \mathcal{E}_h} \sum_{i=1}^2 \beta_i^E \Psi_i^E(\mathbf{x})$$

To compute volume integration firstly we change for reference triangle and later use three point quadrature rule. Let assume that $\hat{\mathbf{v}}_h = \mathbf{v}_h \circ T_k$, then $\nabla \hat{\mathbf{v}}_h = \mathcal{B}_K^T \nabla \mathbf{v}_h \circ T_k$ and

$$\int_K \nabla \mathbf{v}_h \mathbf{u}_h \, d\mathbf{x} = |\mathcal{B}_K| \int_{\hat{K}} (\mathcal{B}_K^T)^{-1} \nabla \hat{\mathbf{v}}_h \hat{\mathbf{u}}_h \, d\mathbf{x} \approx |\mathcal{B}_K| \sum_{q=1}^3 w_q (\mathcal{B}_K^T)^{-1} \nabla \hat{\mathbf{v}}_h(s_q) \hat{\mathbf{u}}_h(s_q),$$

where $w = (\frac{1}{6}, \frac{1}{6}, \frac{1}{6})$ and $s = \left(\begin{pmatrix} 1/2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1/2 \end{pmatrix}, \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} \right)$. That is why, the local volume matrix and right hand side vector are defined as for $\{i, j\} \in \{1, 2, 3\}$

$$(\mathbf{A}_K)_{ij} = \int_K \nabla \Phi_j^K \nabla \Phi_i^K \, d\mathbf{x} \approx |\mathcal{B}_K| \sum_{q=1}^3 w_q (\mathcal{B}_K^T)^{-1} \nabla \lambda_j(s_q) (\mathcal{B}_K^T)^{-1} \nabla \lambda_i(s_q),$$

$$(\mathbf{b}_K)_i = \int_K \mathbf{f} \Phi_i^K \, d\mathbf{x} \approx |\mathcal{B}_K| \sum_{j=1}^3 w_j \mathbf{f} \circ T_K(s_q) \lambda_i(s_q).$$

Using the integration by parts and the fact that $\Phi_j^K \in \mathbb{P}_1(\mathcal{T}_h)$ we get

$$\int_{\partial K} \nabla \Phi_j^K \circ \underline{n} \Phi_i^K \, ds = \int_K \nabla \Phi_j^K \nabla \Phi_i^K \, d\mathbf{x}$$

Computing local edges matrix we use Gauss quadrature and the definition of line integral.

$$\begin{aligned} \int_E \mathbf{u}_h \mathbf{v}_h \, ds &= |E| \int_0^1 \mathbf{u}_h(x(t), y(t)) \mathbf{v}_h(x(t), y(t)) \, ds \\ &\approx \frac{|E|}{2} \sum_{q=1}^Q w_q \mathbf{u}_h \left(x \left(\frac{s_q+1}{2} \right), y \left(\frac{s_q+1}{2} \right) \right) \mathbf{v}_h \left(x \left(\frac{s_q+1}{2} \right), y \left(\frac{s_q+1}{2} \right) \right) \\ &\approx \frac{|E|}{2} \sum_{q=1}^Q w_q \mathbf{u}_h(\underline{S}_q) \mathbf{v}_h(\underline{S}_q) \end{aligned}$$

Let $\{E_1, E_2, E_3\}$ be a set of edges of triangle $K \in \mathcal{T}_h$. Then for $i, j, k \in \{1, 2, 3\}$ and $a, b \in \{1, 2\}$

$$\begin{aligned} (\mathbf{B}_K)_{(3+2(k-1)+a)j} &= \int_{E_k} \nabla \Phi_j^K \circ \underline{n}_{E_k} \Psi_a^{E_k} ds = |E_k| \int_0^1 \nabla \lambda_j \circ \underline{n}_{E_k} \psi_{3+2(k-1)+a} dt \\ &\approx \frac{|E_k|}{2} \sum_{q=1}^Q w_q (\mathbf{B}_K^T)^{-1} \nabla \lambda_j (\underline{S}_q) \circ \underline{n}_{E_k} \psi_{3+2(k-1)+a} (\underline{S}_q) \end{aligned}$$

$$\begin{aligned} (\mathbf{C}_K)_{(3+2(k-1)+a)j} &= \int_{E_k} \Phi_j^K \Psi_a^{E_k} ds = |E_k| \int_0^1 \lambda_j \psi_{3+2(k-1)+a} dt \\ &\approx \frac{|E_k|}{2} \sum_{q=1}^Q w_q \lambda_j (\underline{S}_q) \psi_{3+2(k-1)+a} (\underline{S}_q) \end{aligned}$$

$$\begin{aligned} (\mathbf{D}_K)_{ij} &= \sum_{k=1}^3 \int_{E_k} \Phi_j^K \Phi_i^K ds = \sum_{k=1}^3 |E_k| \int_0^1 \lambda_j \lambda_i dt \\ &\approx \sum_{k=1}^3 \frac{|E_k|}{2} \sum_{q=1}^Q w_q \lambda_j (\underline{S}_q) \lambda_i (\underline{S}_q) \end{aligned}$$

$$\begin{aligned} (\mathbf{E}_K)_{(3+2(k-1)+a)(3+2(k-1)+b)} &= \int_{E_k} \Psi_b^{E_k} \Psi_a^{E_k} ds = |E_k| \int_0^1 \psi_{3+2(k-1)+b} \psi_{3+2(k-1)+a} dt \\ &\approx \frac{|E_k|}{2} \sum_{q=1}^Q w_q \psi_{3+2(k-1)+b} (\underline{S}_q) \psi_{3+2(k-1)+a} (\underline{S}_q) \end{aligned}$$

We can build the local matrix \mathbf{M}_K following for $0 \leq n < N$

$$\mathbf{M}_K = \mathbf{A}_K - (1 - \varepsilon) \mathbf{A}_K + \mathbf{B}_K - \varepsilon \mathbf{B}_K^T + \frac{\tau}{h} (\mathbf{D}_K - \mathbf{C}_K - \mathbf{C}_K^T + \mathbf{E}_K)$$

Then the global matrix \mathbf{M} is constructed in such way that the rows and columns of the local matrix \mathbf{M}_K has following global numbering

$$\left[\begin{array}{ccc} 3(K-1)+1 & 3(K-1)+2 & 3(K-1)+3 \\ 3N+2(E_1-1)+\zeta_1^{E_1} & 3N+2(E_1-1)+\zeta_2^{E_1} & 3N+2(E_2-1)+\zeta_1^{E_2} \\ 3N+2(E_2-1)+\zeta_2^{E_2} & 3N+2(E_3-1)+\zeta_1^{E_3} & 3N+2(E_3-1)+\zeta_2^{E_3} \end{array} \right]^2$$

where K is the index of triangle, E_i index of edge, N is an amount of triangles, Nd_i^E index of node of edge E and

$$\zeta_i^E = 1 + \frac{1 + \operatorname{sgn}(Nd_i^E - Nd_{1+(i \bmod 2)}^E)}{2}.$$

The boundary conditions can be imposed strongly or by penalisation. As we can see on the

Figures A.4 and A.5 is only depending on the personal preferences.

Below we present the convergence plots of the error for the above implementation in Matlab. The Figure A.4 depicts the symmetric and non-symmetric formulation of hdG method with strongly imposed boundary conditions.

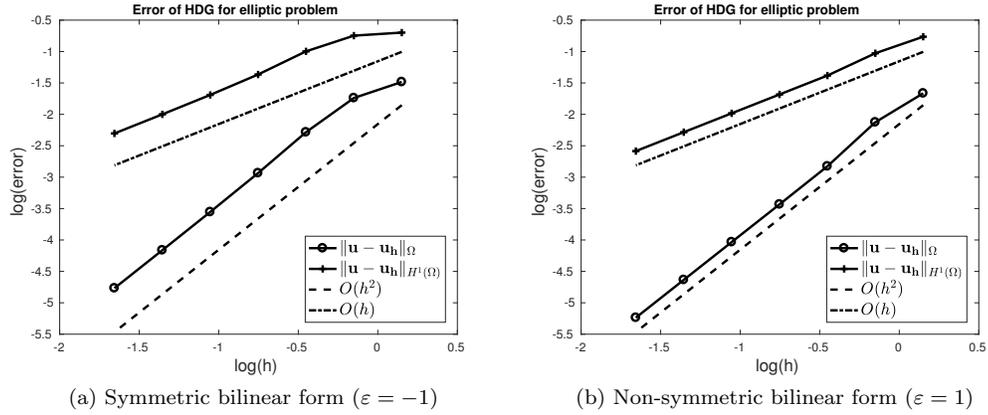


Figure A.4: Two dimensional Matlab implementation - error convergence of the hdG method with strongly imposed boundary conditions

In addition, the Figure A.5 depicts the symmetric and non-symmetric formulation of hdG method with boundary conditions imposed by penalisation.

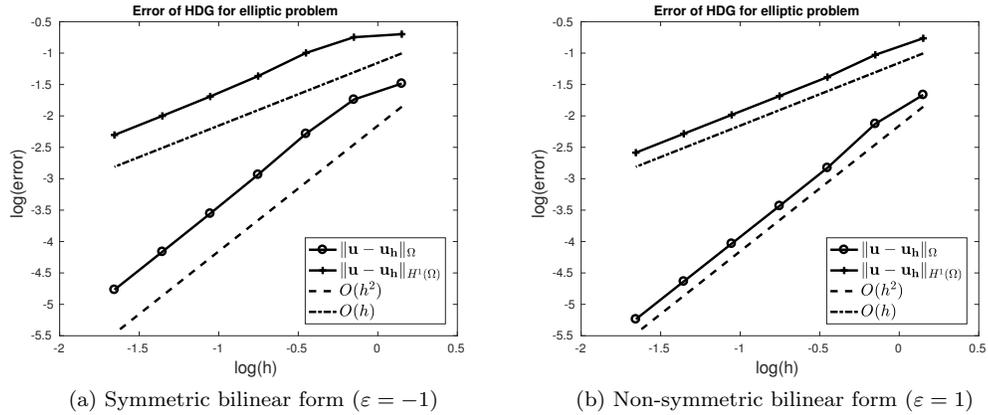


Figure A.5: Two dimensional Matlab implementation - error convergence of the hdG method with boundary conditions imposed by penalisation

Appendix B

FreeFem++ implementations

B.1 Taylor-Hood method for Stokes problem

In this section we discuss the FreeFem++ implementation of some methods for the Stokes problem. We consider the unit square domain $\Omega = (0,1)^2$, choose the right hand side \mathbf{f} , and the TVNF boundary condition such that the exact solution is given by

$$\mathbf{u} = \text{curl} \left[x^2 (1-x)^2 y^2 (1-y)^2 \right], \quad p = x^2 - y^2.$$

At the beginning, we consider an implementation of Taylor-Hood method as the little introduction to the FreeFem++ programming.

```
// Mesh
int m = 20;
int[int] labs = [1,2,1,2];
mesh Th=square(m,m,label=labs);
plot(Th);
```

We use the build-in function `square` to define the mesh that split the unit square for `m` by `m` squares. Each of these smaller squares is split to two triangles. Moreover, we add labels for boundaries by `label` such that the counting is from the bottom boundary. The last line is to plot the mesh and it is depict in Figure B.1.

The next step is to define the discrete spaces.

```
// Discrete space
fespace Vh(Th, [P2,P2,P1]);
Vh [u1,u2,p], [v1,v2,q];
```

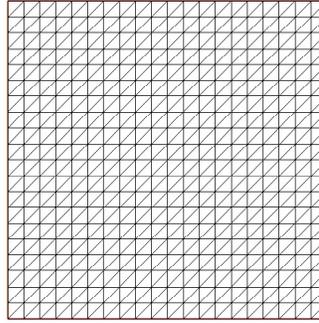


Figure B.1: Example of the mesh of the unit square

We can use the build-in polynomial spaces such like space of all polynomials degree two P2 or space of all polynomials degree one P1. In the last line we introduce here the functions from this spaces. Now we define operators that we will use to pose the problem.

```
// Definition of the operators
macro grad(u) [dx(u),dy(u)] //EOM
macro div(u1,u2) (dx(u1)+dy(u2)) //EOM
```

To do this we use the macro that has to finish with `//`. Moreover we need to impose boundary conditions.

```
// Normal vector
func Nor=[N.x,N.y];
// Normal flux boundary condtions
func sigmaNN1 = -4*x*(-1+x)*y*(-1+y)*(-1+2*y)*(-1+2*x)+x*x-y*y;
func sigmaNN2 = 4*x*(-1+x)*y*(-1+y)*(-1+2*y)*(-1+2*x)+x*x-y*y;
```

It includes defining the normal vector and the formula of the functions on the boundaries. The last thing is to define the right hand side function.

```
// Right hand side
func f = [24*x^4*y+48*x^2*y^3-12*x^4-48*x^3*y-72*x^2*y^2-48*x*y^3
          +24*x^3+48*x^2*y+72*x*y^2+8*y^3-12*x^2-24*x*y-12*y^2+2*x+4*y,
          -48*x^3*y^2-24*x*y^4+48*x^3*y+72*x^2*y^2+48*x*y^3
          +12*y^4-8*x^3-72*x^2*y-48*x*y^2-24*y^3+12*x^2+24*x*y
          +12*y^2-4*x-2*y];
```

We are ready to pose the problem.

```
// Problem
solve ProbTH([u1,u2,p],[v1,v2,q],solver=sparse solver)
= int2d(Th)(grad(u1)'*grad(v1) + grad(u2)'*grad(v2)
```

```

- div(u1,u2)*q - div(v1,v2)*p)
- int2d(Th)(f'*[v1,v2])
+ int1d(Th,1)(sigmaNN1*([v1,v2] '*Nor))
+ int1d(Th,2)(sigmaNN2*([v1,v2] '*Nor))
+on(1,u1=0) +on(2,u2=0);

```

The second line is defined the variables associated with the solution and test functions, and additionally we can choose the solver. The third, fourth and the fifth lines are the weak formulation of the problem. The last three lines are imposing the boundary conditions. The only thing that left is to plot the solution.

```

// Solution
plot([u1,u2], wait=1,value=1);
plot(p, wait=1,fill=1,value=1);

```

The results are depicted in Figure B.2.

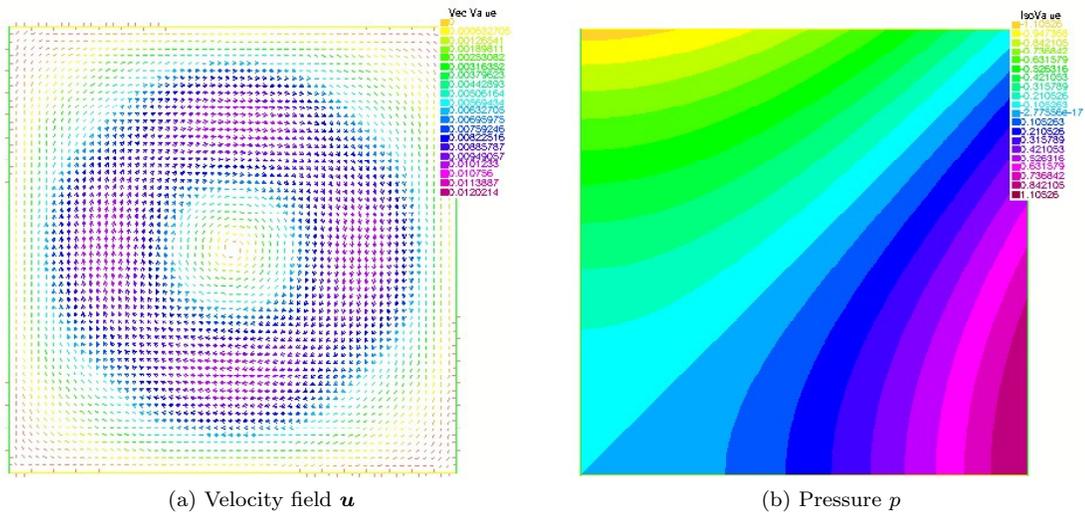


Figure B.2: Taylor-Hood solution

B.2 hdG method for Stokes problem

Now, we present the implementation of the hdG method from Chapter 1. It requires additional terms and packages. We start with including the necessary packages.

```

load "Element_Mixte"
load "Element_PkEdge"
load "Element_NEdge"

```

The last one is not the build-in one, however we need it to have tangential vector associated with the edge and not as it is in FreeFem++ associated with triangle. Once again we define the mesh in the same way as for Taylor-Hood and the discrete spaces.

```
// Mesh
int m = 20;
int[int] labs = [1,2,1,2];
mesh Th=square(m,m,label=labs);
plot(Th);
// Discrete space
fespace Vh(Th,[BDM1,P0edge,P0]);
Vh [ux,uy,ul,p],[vx,vy,vl,q];
```

However this time we use BDM space degree one BDM1 included in `Element_Mixte`, set of polynomials degree zero on the edges `P0edge` included in `Element_PkEdge` and set of polynomials degree zero `P0`. We define operators that we will use to pose the problem.

```
// Definition of the operators
macro grad(u) [dx(u),dy(u)] //EOM
macro gradn(ux,uy) ([grad(ux)']*Nor, grad(uy)']*Nor) //EOM
macro div(ux,uy) (dx(ux)+dy(uy)) //EOM
```

Next we define the normal and tangential vectors.

```
// Normal vector
func Nor=[N.x,N.y];
// Tangential vector
fespace Uh(Th,PNEdge);
Uh [nx,ny];
nx[] = 1;
func Tan=[-ny,nx];
```

And here we take advantage of the created spaces giving the normal vector associated with the edge from `Element_NEdge`. The only what left is stabilisation parameter, right hand side function and normal flux boundary conditions.

```
// Stabilisation parameter
real tau = 6;
// Normal flux boundary condtions
func sigmaNN1 = -4*x*(-1+x)*y*(-1+y)*(-1+2*y)*(-1+2*x)+x*x-y*y;
func sigmaNN2 = 4*x*(-1+x)*y*(-1+y)*(-1+2*y)*(-1+2*x)+x*x-y*y;
// Right hand side
func f = [24*x^4*y+48*x^2*y^3-12*x^4-48*x^3*y-72*x^2*y^2-48*x*y^3
          +24*x^3+48*x^2*y+72*x*y^2+8*y^3-12*x^2-24*x*y-12*y^2+2*x+4*y,
```

```

-48*x^3*y^2-24*x*y^4+48*x^3*y+72*x^2*y^2+48*x*y^3
+12*y^4-8*x^3-72*x^2*y-48*x*y^2-24*y^3+12*x^2+24*x*y
+12*y^2-4*x-2*y];

```

We are ready to pose the problem.

```

// Problem
solve ProbHDG([ux,uy,u1,p],[vx,vy,v1,q],solver=sparseSolver)
= int2d(Th)(grad(ux)'*grad(vx) + grad(uy)'*grad(vy)
- div(ux,uy)*q - div(vx,vy)*p)
- int2d(Th)(f'*[vx,vy])
+ intalldges(Th)(-(Tan'*gradn(ux,uy))*(Tan'*[vx,vy]-(v1))
-(Tan'*gradn(vx,vy))*(Tan'*[ux,uy]-(u1)))
+ intalldges(Th,qforder=1)(tau/hTriangle*(Tan'*[ux,uy]-(u1))
*(Tan'*[vx,vy]-(v1)))
+ int1d(Th,1)(sigmaNN1*([vx,vy]'*Nor))
+ int1d(Th,2)(sigmaNN2*([vx,vy]'*Nor))
+on(1,2, u1=0);

```

Once again the second line is defined the variables associated with the solution and test functions, and additionally we can choose the solver. The third till ninth lines are the weak formulation of the problem. We show here just symmetric formulation, but if in seventh line we change $-$ for $+$ we have the non-symmetric formulation. Furthermore, we have easy way to formulate the projection in the eight line since it is enough to take one quadrature point for the edge instead of two. The last three lines are imposing the boundary conditions. The only thing that left is to plot the solution.

```

// Solution
plot([ux,uy], wait=1,value=1);
plot(p, wait=1,fill=1,value=1);

```

The results are depicted in Figure B.3.

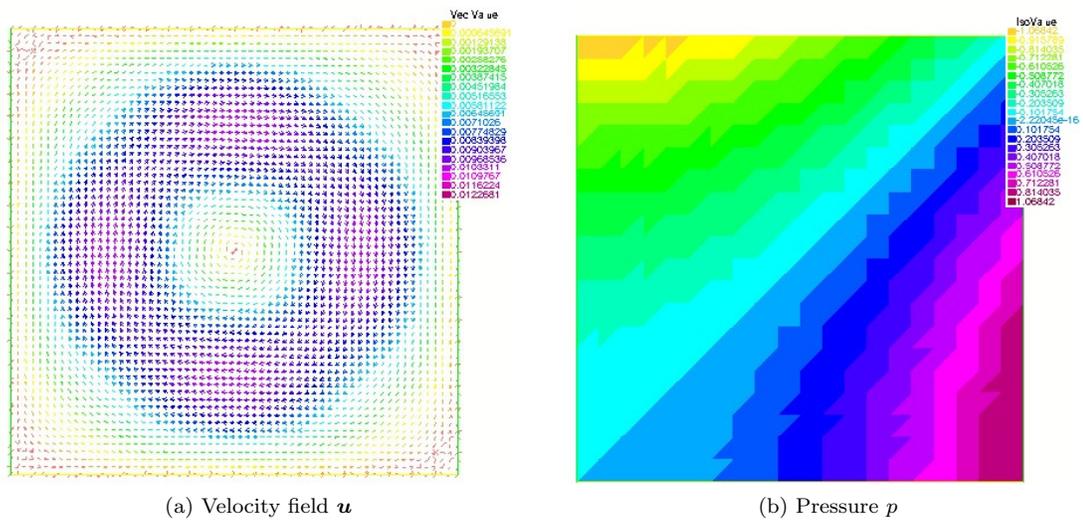


Figure B.3: Hybrid discontinuous Galerkin solution

Bibliography

- [ABGS14] M. Ayadi, L. Baffico, M. K. Gdoura, and T. Sassi. Error estimates for Stokes problem with Tresca friction conditions. *ESAIM Math. Model. Numer. Anal.*, 48(5):1413–1429, 2014.
- [AdDBM⁺14] B. Ayuso de Dios, F. Brezzi, L. D. Marini, J. Xu, and L. Zikatanov. A simple preconditioner for a discontinuous Galerkin method for the Stokes problem. *Journal of Scientific Computing*, 58(3):517–547, 2014.
- [BB01] R. Becker and M. Braack. A finite element pressure gradient stabilization for the Stokes equations based on local projections. *Calcolo*, 38(4):173–199, 2001.
- [BBF93] C. Baiocchi, F. Brezzi, and L. P. Franca. Virtual bubbles and Galerkin-least-squares type methods (Ga.L.S.). *Comput. Methods Appl. Mech. Engrg.*, 105(1):125–141, 1993.
- [BBF13] D. Boffi, F. Brezzi, and M. Fortin. *Mixed finite element methods and applications*, volume 44 of *Springer Series in Computational Mathematics*. Springer, Heidelberg, 2013.
- [BBGS04] T. Barth, P. Bochev, M. Gunzburger, and J. Shadid. A taxonomy of consistently stabilized finite element methods for the Stokes problem. *SIAM J. Sci. Comput.*, 25(5):1585–1607, 2004.
- [BD88] F. Brezzi and Jim Douglas, Jr. Stabilized mixed methods for the Stokes problem. *Numer. Math.*, 53(1-2):225–235, 1988.
- [BDG06] P. B. Bochev, C. R. Dohrmann, and M. D. Gunzburger. Stabilization of low-order mixed finite elements for the Stokes equations. *SIAM J. Numer. Anal.*, 44(1):82–101 (electronic), 2006.
- [BFT93] M. A. Behr, L. P. Franca, and T. E. Tezduyar. Stabilized finite element methods for the velocity-pressure-stress formulation of incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 104(1):31–48, 1993.

- [BG04] P. Bochev and M. Gunzburger. An absolutely stable pressure-Poisson stabilized finite element method for the Stokes equations. *SIAM J. Numer. Anal.*, 42(3):1189–1207, 2004.
- [BHMV99] M. Brezina, C. I Heberton, J. Mandel, and P. Vanek. An iterative method with convergence rate chosen a priori, ucd/ccm report 140. Technical report, Center for Computational Mathematics, University of Colorado at Denver, 1999.
- [BP84] F. Brezzi and J. Pitkäranta. On the stabilization of finite element approximations of the Stokes equations. In *Efficient solutions of elliptic systems (Kiel, 1984)*, volume 10 of *Notes Numer. Fluid Mech.*, pages 11–19. Friedr. Vieweg, Braunschweig, 1984.
- [Bre74] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge*, 8(R-2):129–151, 1974.
- [BS08] S. C. Brenner and L. R. Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer, New York, third edition, 2008.
- [Bur08] E. Burman. Pressure projection stabilizations for Galerkin approximations of Stokes’ and Darcy’s problem. *Numer. Methods Partial Differential Equations*, 24(1):127–143, 2008.
- [CB00] R. Codina and J. Blasco. Analysis of a pressure-stabilized finite element approximation of the stationary Navier-Stokes equations. *Numer. Math.*, 87(1):59–81, 2000.
- [CCNP13] A. Cesmelioglu, B. Cockburn, N. C. Nguyen, and J. Peraire. Analysis of HDG methods for Oseen equations. *J. Sci. Comput.*, 55(2):392–431, 2013.
- [CDKN14] L. Conen, V. Dolean, R. Krause, and F. Nataf. A coarse space for heterogeneous Helmholtz problems based on the Dirichlet-to-Neumann operator. *J. Comput. Appl. Math.*, 271:83–99, 2014.
- [CDNQ12] Th. Cluzeau, V. Dolean, F. Nataf, and A. Quadrat. Preconditioning techniques for systems of partial differential equations based on algebraic methods. Technical Report 7953, INRIA, 2012. <http://hal.inria.fr/hal-00694468>.
- [CDNQ13] Th. Cluzeau, V. Dolean, F. Nataf, and A. Quadrat. Symbolic techniques for domain decomposition methods. In *Domain decomposition methods in science and engineering XX*, pages 27–38. Springer LNCSE, <http://hal.archives-ouvertes.fr/hal-00664092>, 2013.

- [CDPE15] B. Cockburn, D. A. Di Pietro, and A. Ern. Bridging the hybrid high-order and hybridizable discontinuous Galerkin methods. *ESAIM: Math. Model Numer. Anal.(M2AN)*, 2015.
- [CDS03] X.-Ch. Cai, M. Dryja, and M. Sarkis. Restricted additive Schwarz preconditioners with harmonic overlap for symmetric positive definite linear systems. *SIAM J. Numer. Anal.*, 41(4):1209–1231 (electronic), 2003.
- [CG09] B. Cockburn and J. Gopalakrishnan. The derivation of hybridizable discontinuous Galerkin methods for Stokes flow. *SIAM J. Numer. Anal.*, 47(2):1092–1125, 2009.
- [CGL09] B. Cockburn, J. Gopalakrishnan, and R. Lazarov. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. *SIAM J. Numer. Anal.*, 47(2):1319–1365, 2009.
- [CGN⁺11] B. Cockburn, J. Gopalakrishnan, N. C. Nguyen, J. Peraire, and F. J. Sayas. Analysis of HDG methods for Stokes flow. *Math. Comp.*, 80(274):723–760, 2011.
- [CGS10] B. Cockburn, J. Gopalakrishnan, and F. J. Sayas. A projection-based error analysis of HDG methods. *Math. Comp.*, 79(271):1351–1367, 2010.
- [CNP10] B. Cockburn, N. C. Nguyen, and J. Peraire. A comparison of HDG methods for Stokes flow. *J. Sci. Comput.*, 45(1-3):215–237, 2010.
- [Coc10] B. Cockburn. The hybridizable discontinuous Galerkin methods. In *Proceedings of the International Congress of Mathematicians. Volume IV*, pages 2749–2775. Hindustan Book Agency, New Delhi, 2010.
- [Coc16] Bernardo Cockburn. Static Condensation, Hybridization, and the Devising of the HDG methods. In Gabriel R. Barrenechea, Franco Brezzi, Andrea Cangiani, and Emmanuil H. Georgoulis, editors, *Building Bridges: Connections and Challenges in Modern Approaches to Numerical Partial Differential Equations*, pages 129–177. Springer International Publishing, Cham, 2016.
- [CQS12] B. Cockburn, W. Qiu, and K. Shi. Conditions for superconvergence of HDG methods for second-order elliptic problems. *Math. Comp.*, 81(279):1327–1353, 2012.
- [CS99] X.-Ch. Cai and M. Sarkis. A restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.*, 21(2):792–797 (electronic), 1999.
- [CS13] B. Cockburn and K. Shi. Conditions for superconvergence of HDG methods for Stokes flow. *Math. Comp.*, 82(282):651–671, 2013.

- [CS14a] B. Cockburn and F. J. Sayas. Divergence-conforming HDG methods for Stokes flows. *Math. Comp.*, 83(288):1571–1598, 2014.
- [CS14b] B. Cockburn and K. Shi. Devising HDG methods for Stokes flow: an overview. *Comput. & Fluids*, 98:221–229, 2014.
- [DB04] C. R. Dohrmann and P. B. Bochev. A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Internat. J. Numer. Methods Fluids*, 46(2):183–201, 2004.
- [Des93] B. Després. Domain decomposition method and the Helmholtz problem. II. In *Second International Conference on Mathematical and Numerical Aspects of Wave Propagation (Newark, DE, 1993)*, pages 197–206. SIAM, Philadelphia, PA, 1993.
- [DJN15] V. Dolean, P. Jolivet, and F. Nataf. *An introduction to domain decomposition methods*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015. Algorithms, theory, and parallel implementation.
- [DJR92] B. Després, P. Joly, and J. E. Roberts. A domain decomposition method for the harmonic Maxwell equations. In *Iterative methods in linear algebra (Brussels, 1991)*, pages 475–484. North-Holland, Amsterdam, 1992.
- [DN06] V. Dolean and F. Nataf. A new domain decomposition method for the compressible Euler equations. *M2AN Math. Model. Numer. Anal.*, 40(4):689–703, 2006.
- [DNR09] V. Dolean, F. Nataf, and F. Rapin. Deriving a new domain decomposition method for the Stokes equations using the Smith factorization. *Math. Comp.*, 78(266):789–814, 2009.
- [DNSS12] V. Dolean, F. Nataf, R. Scheichl, and N. Spillane. Analysis of a two-level Schwarz method with coarse spaces based on local Dirichlet-to-Neumann maps. *Comput. Methods Appl. Math.*, 12(4):391–414, 2012.
- [DPE12] D. A. Di Pietro and A. Ern. *Mathematical aspects of discontinuous Galerkin methods*, volume 69 of *Mathématiques & Applications (Berlin) [Mathematics & Applications]*. Springer, Heidelberg, 2012.
- [DPEL16] D. A. Di Pietro, A. Ern, and S. Lemaire. A Review of Hybrid High-Order Methods: Formulations, Computational Aspects, Comparison with Other Methods. In Gabriel R. Barrenechea, Franco Brezzi, Andrea Cangiani, and Emmanuil H. Georgoulis, editors, *Building Bridges: Connections and Challenges in Modern Approaches to Numerical Partial Differential Equations*, pages 205–236. Springer International Publishing, Cham, 2016.

- [EG03] E. Efsthathiou and M. J. Gander. Why restricted additive Schwarz converges faster than additive Schwarz. *BIT*, 43(suppl.):945–959, 2003.
- [EG04] A. Ern and J. L. Guermond. *Theory and practice of finite elements*, volume 159 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2004.
- [EGLW12] Y. Efendiev, J. Galvis, R. Lazarov, and J. Willems. Robust domain decomposition preconditioners for abstract symmetric positive definite bilinear forms. *ESAIM Math. Model. Numer. Anal.*, 46(5):1175–1199, 2012.
- [ES10] H. Egger and J. Schöberl. A hybrid mixed discontinuous Galerkin finite-element method for convection-diffusion problems. *IMA J. Numer. Anal.*, 30(4):1206–1234, 2010.
- [EW13a] H. Egger and Ch. Waluga. *hp* analysis of a hybrid DG method for Stokes flow. *IMA J. Numer. Anal.*, 33(2):687–721, 2013.
- [EW13b] H. Egger and Ch. Waluga. A hybrid discontinuous Galerkin method for Darcy-Stokes problems. In Randolph E. Bank, Michael J. Holst, Olof B. Widlund, and Jinchao Xu, editors, *Domain Decomposition Methods in Science and Engineering XX*, volume 91 of *Lecture Notes in Computational Science and Engineering*, pages 663–670. Springer, 2013.
- [FHS93] L. P. Franca, T. J. R. Hughes, and R. Stenberg. Stabilized finite element methods. *Incompressible computational fluid dynamics*, pages 87–107, 1993.
- [FP92] M. Fortin and R. Pierre. Stability analysis of discrete generalized Stokes problems. *Numer. Methods Partial Differential Equations*, 8(4):303–323, 1992.
- [FS01] A. Frommer and D. B. Szyld. An algebraic convergence theory for restricted additive Schwarz methods using weighted max norms. *SIAM J. Numer. Anal.*, 39(2):463–479 (electronic), 2001.
- [GE10a] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high-contrast media. *Multiscale Model. Simul.*, 8(4):1461–1483, 2010.
- [GE10b] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media: reduced dimension coarse spaces. *Multiscale Model. Simul.*, 8(5):1621–1644, 2010.
- [GMT08] S. Ganesan, G. Matthies, and L. Tobiska. Local projection stabilization of equal order interpolation applied to the Stokes problem. *Math. Comp.*, 77(264):2039–2060, 2008.
- [GR86] V. Girault and P. A. Raviart. *Finite element methods for Navier-Stokes equations*, volume 5 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1986. Theory and algorithms.

- [GR06] P. Gosselet and Ch. Rey. Non-overlapping domain decomposition methods in structural mechanics. *Arch. Comput. Methods Engrg.*, 13(4):515–572, 2006.
- [HF87] T. J. R. Hughes and L. P. Franca. A new finite element formulation for computational fluid dynamics. VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces. *Comput. Methods Appl. Mech. Engrg.*, 65(1):85–96, 1987.
- [HFB86] T. J. R. Hughes, L. P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics. V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations. *Comput. Methods Appl. Mech. Engrg.*, 59(1):85–99, 1986.
- [HJN15] R. Haferssas, P. Jolivet, and F. Nataf. An additive Schwarz method type theory for Lions’ algorithm and Optimized Schwarz Methods. preprint, <https://hal.archives-ouvertes.fr/hal-01278347>, December 2015.
- [HLB79] T. J. R. Hughes, W. K. Liu, and A. Brooks. Finite element analysis of incompressible viscous flows by the penalty function formulation. *J. Comput. Phys.*, 30(1):1–60, 1979.
- [JCWS99] K. E. Jansen, S. S. Collis, C. Whiting, and F. Shakib. A better consistency for low-order stabilized finite element methods. *Comput. Methods Appl. Mech. Engrg.*, 174(1-2):153–170, 1999.
- [KK98] G. Karypis and V. Kumar. A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Technical report, University of Minnesota, Department of Computer Science and Engineering, Army HPC Research Center, Minneapolis, MN, 1998.
- [Leh10] Ch. Lehrenfeld. Hybrid discontinuous Galerkin methods for solving incompressible flow problems. Dissertation, Rheinisch-Westfälischen Technischen Hochschule Aachen, June 2010.
- [Lio88] P.-L. Lions. On the Schwarz alternating method. I. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations (Paris, 1987)*, pages 1–42. SIAM, Philadelphia, PA, 1988.
- [Lio90] P.-L. Lions. On the Schwarz alternating method. III. A variant for nonoverlapping subdomains. In *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations (Houston, TX, 1989)*, pages 202–223. SIAM, Philadelphia, PA, 1990.
- [LNS15] Sébastien Loisel, Hieu Nguyen, and Robert Scheichl. Optimized Schwarz and 2-Lagrange multiplier methods for multiscale elliptic PDEs. *SIAM J. Sci. Comput.*, 37(6):A2896–A2923, 2015.

- [LS16] Ch. Lehrenfeld and J. Schöberl. High order exactly divergence-free hybrid discontinuous Galerkin methods for unsteady incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 307:339–361, 2016.
- [LSY98] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK users' guide*, volume 6 of *Software, Environments, and Tools*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods.
- [Nic87] R. A. Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24(2):355–365, 1987.
- [NP12] N. C. Nguyen and J. Peraire. Hybridizable discontinuous Galerkin methods for partial differential equations in continuum mechanics. *J. Comput. Phys.*, 231(18):5955–5988, 2012.
- [NPC09a] N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for linear convection-diffusion equations. *J. Comput. Phys.*, 228(9):3232–3254, 2009.
- [NPC09b] N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for nonlinear convection-diffusion equations. *J. Comput. Phys.*, 228(23):8841–8855, 2009.
- [NPC10] N. C. Nguyen, J. Peraire, and B. Cockburn. A hybridizable discontinuous Galerkin method for Stokes flow. *Comput. Methods Appl. Mech. Engrg.*, 199(9-12):582–597, 2010.
- [NPC11] N. C. Nguyen, J. Peraire, and B. Cockburn. An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 230(4):1147–1170, 2011.
- [NS02] R. Nabben and D. B. Szyld. Convergence theory of restricted multiplicative Schwarz methods. *SIAM J. Numer. Anal.*, 40(6):2318–2336 (electronic) (2003), 2002.
- [NXD10] F. Nataf, H. Xiang, and V. Dolean. A two level domain decomposition preconditioner based on local Dirichlet-to-Neumann maps. *C. R. Math. Acad. Sci. Paris*, 348(21-22):1163–1167, 2010.
- [PNC10] J. Peraire, N. C. Nguyen, and B. Cockburn. A hybridizable discontinuous Galerkin method for the compressible Euler and Navier-Stokes equations. *AIAA paper*, 363:2010, 2010.
- [PS11] A. Pechstein and J. Schöberl. Tangential-displacement and normal-normal-stress continuous mixed finite elements for elasticity. *Math. Models Methods Appl. Sci.*, 21(8):1761–1782, 2011.

- [QV99] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, 1999.
- [RH73] W. H. Reed and T. R. Hill. Triangular mesh methods for the neutron transport for a scalar hyperbolic equation. Technical report, Research Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
- [Riv08] B. Rivière. *Discontinuous Galerkin methods for solving elliptic and parabolic equations*, volume 35 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. Theory and implementation.
- [SBG96] B. F. Smith, P. E. Bjørstad, and W. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.
- [SCGT07] A. St-Cyr, M. J. Gander, and S. J. Thomas. Optimized multiplicative, additive, and restricted additive Schwarz preconditioning. *SIAM J. Sci. Comput.*, 29(6):2402–2425 (electronic), 2007.
- [Sch70] H. A. Schwarz. *Über einen Grenzübergang durch alternirendes Verfahren*. Zürcher u. Furrer, 1870.
- [SDH⁺14] N. Spillane, V. Dolean, P. Hauret, F. Nataf, C. Pechstein, and R. Scheichl. Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps. *Numer. Math.*, 126(4):741–770, 2014.
- [Sil94] D. J. Silvester. Optimal low order finite element methods for incompressible flow. *Comput. Methods Appl. Mech. Engrg.*, 111(3-4):357–368, 1994.
- [SK90] D. J. Silvester and N. Kechkar. Stabilised bilinear-constant velocity-pressure finite elements for the conjugate gradient solution of the Stokes problem. *Comput. Methods Appl. Mech. Engrg.*, 79(1):71–86, 1990.
- [Smi61] H. J. S. Smith. On systems of linear indeterminate equations and congruences. *Philosophical Transactions of the Royal Society of London*, 151:293–326, 1861.
- [SS86] Y. Saad and M. H. Schultz. GMRES: a Generalized Minimal Residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856–869, 1986.
- [SST02] D. Schötzau, Ch. Schwab, and A. Toselli. Mixed *hp*-DGFEM for incompressible flows. *SIAM J. Numer. Anal.*, 40(6):2171–2194 (electronic) (2003), 2002.
- [TM05] R. Temam and A. Miranville. *Mathematical modeling in continuum mechanics*. Cambridge University Press, Cambridge, second edition, 2005.

- [TW05] A. Toselli and O. Widlund. *Domain Decomposition Methods - Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer, 2005.