Conservation of Angular Momentum in the Navier-Stokes equations using the Mimetic Spectral Element Method

Jaep Koning
Masterthesis Aerospace Engineering | Aerodynamics and Wind Energy

Delft University of Technology
6-3-2026

Supervisor M.I. Gerritsma

Contents

1 Introduction

Since ancient times people have looked at flowing media hoping to understand its behavior. Many of the familiar great minds of the past have given the problem their attention. Archimedes devised one of the first laws related to fluid behavior with his law describing buoyancy. Leonardo da Vinci was also captivated by fluid flows, his drawings of flows show in great detail what turbulent and chaotic flows look like. Both Newton and Euler spent significant time trying to describe fluids, the invention of calculus considerably aiding in the effort and this culminated in the discovery of the Navier-Stokes equations by of course Navier and Stokes. With the advent of the Navier-Stokes equations the motion of fluids were now completely described by a purely mathematical formulation. The study of these equations however raised many more questions than they answered. To this day the Navier-Stokes equations do not have a general solution but for the most simplest of cases. In fact, at the time of writing this work, a general solution (or even the proof that it exists) for the (incompressible) Navier-Stokes equations is one of the 6 remaining millennium problems.

The advent of the general purpose computer after the second world war has enabled the numerical study of the Navier-Stokes equations. The Navier Stokes equations in essence embody only three fundamental principles, The conservation of mass, momentum and energy. Through evermore elaborate numerical schemes and increased computational power more and more progress is made in understanding the most complex behaviors in fluid dynamics. However, in most formulations the conserved properties are only weakly conserved, being only fully conserved in the limit of the mesh size going to zero. In practice of course, this mesh size is never achieved, far from it usually. A crude but effective example is a finite difference code. With finite differences the solution that is produced can be understood to be the solution to the so called modified equations and error analysis of schemes is dependent on the difference between the modified equations and the real equations. For the very basic methods it can be shown that finite volume formulations and finite elements reduce to the same discretized system and thus suffer from the same problem.

This raises a very deep question in regards to computational fluid dynamics, are the problems that are solved, even the right problems to solve? A new class of methods has been in the works for quite some time. The mimetic methods aim to mimic the structure of the partial differential equations at the numerical level through discrete differential forms. And through the use of these discrete differential forms some success has already been achieved [32, 37]. Formulations have been created that exactly satisfy the conservation of mass and momentum, regardless of the order of the method. However some properties still escape these formulations. One of the more interesting properties to add to the list of preserved quantities would be the conservation of angular momentum. It takes only a single look at a modern DNS of a boundary layer to see that it is full of vortices on all scales. As will be seen later that even with the conservation of linear momentum satisfied it is still possible to not conserve angular momentum. Not conserving angular momentum is equivalent to the application of a phantom torque to the flow, leading potentially to stronger or weaker vortices than what would be realistic.

Given the potential of the mimetic methods to conserve physical quantities, the question arises wether a formulation using a mimetic spectral element can be found that indeed does conserve angular momentum in addition to mass and linear momentum.

The current state of using the mimetic spectral element method for conservation of angular momentum is that a formulation exists for the Stokes equations on a square domain. In order to take this formulation and create an angular momentum conserving method for the Navier-Stokes equations several steps need to be undertaken. As an introduction, two standard formulations are explored one for the Poisson equations and one for the Stokes equations using the mimetic spectral element method. Next, the formulation that conserves angular momentum on square grids is discussed in section 7. From there on there will be three sections dedicated to constructing a Navier-Stokes formulation. Firstly the angular momentum conserving formulation needs to be extended to arbitrary meshes. The amount of problems that are solved on only square domains is very limited and thus it is almost required for the method to be modified such that it can be used on arbitrary curved domains. The expansion to arbitrary grids will be done in section 8. Secondly the convective term needs to be taken into account to transform the formulation from a Stokes solver to a Navier-Stokes solver which will be done in section 10. Thirdly, the inclusion of the convective terms will make the formulation non linear. This makes the problem necessarily iterative on a modern computer. One way of performing the iterative process to a steady state solution is to evolve the system in time until it reaches a steady state. It thus makes sense to create a formulation that allows for the inclusion of time derivatives. The creation of general time derivative in the mimetic spectral element method is done in section 9 using the Poisson equation as the basis.

2 Summary

In this report an attempt is made to produce a spectral mimetic method for the Navier-Stokes equations that strongly conserves angular momentum. To do so first the relevant literature concerning mimetic methods is reviewed in section 3. These methods aim to use algebraic discrete vector calculus to solve partial differential equations in order to strongly conserve the underlying conservation laws inherently. Furthermore the relevance of conserving angular momentum is reviewed by making it apparent that under contemporary numerical methods angular momentum is not conserved and as a result several transport problems calculated with non angular momentum conserving methods produce nonphysical results. Then an overview of the intended test cases for the name method is reviewed. Most of these methods are standardized problems from literature which allows for effective benchmarking.

In order to support the mathematical concepts in this report section 4 contains the fundamental math required for the spectral mimetic method. In this section important concepts like co- and contra-variant vectors, differential forms, maps, wedge products and Hodge operators are summarized. In order to effectively define parametrized functions on domains basis functions are presented that have easily calculated derivatives using simple and sparse incidence matrices. Finally numerical integration is discussed to produce mass matrices. Both incidence and mass matrices are heavily used in the mimetic spectral method.

To get familiar with the method in section 5 a Poisson formulation of the mimetic spectral element method is discussed. Here the idea of splitting sequential operators into their components and appropriate basis functions for the field variables are presented. This formulation is then tested using a manufactured solution and shown to converge both when the order of the method is increased or when the mesh is refined.

Next, a non angular momentum conserving mimetic method for the stokes equations is presented in section 6. This formulation is then validated using the method of manufactured solutions to show that it converges and tested against several benchmarks such as Poisseuille flow, lid driven cavity flow and a backwards facing step.

The formulation presented in section 7 is a formulation created by Fisser[18] that strongly conserves angular momentum through explicitly incorporating the fluid stresses, using a specific set of basis functions and projecting to compatible basis functions. This formulation is also shown to converge on a manufactured solution and tested on the Poisseuille and lid driven cavity flows.

In section 8 an attempt is made to use the formulation by Fisser on curved grids. It is then shown that while on a curved grid the solution appears close to the uncurved grid, it is not correct. Furthermore it is shown that especially the symmetry of the stress tensor is broken compared to the orthogonal grid formulation, showing that this exact formulation does not provide a way to conserve angular momentum on arbitrary grids.

Next, section 9 describes a method to incorporate the time derivative in the spectral mimetic using semi-discretisation principles. This method is shown to make integrating a time derivative relatively easy despite the mathematical complexity of the mimetic meth-

ods. The method is shown to be convergent for a backwards Euler method on the Poisson equations, opening up the possibilities to incorporate more complicated time integration methods.

Then in section 10 transport equations are discussed and a time dependent version of Fisser's formulation is presented that converges to the steady state solution of a lid driven cavity for the Stokes equations. This time dependent formulation is then used in a moving mesh Lagrangian formulation simulating the full Navier Stokes equations. This formulation, however, is not stable and results in the simulation blowing up. This instability is likely caused by the problems encountered in the transformation process of section 8

Lastly in the conclusion, it is determined that transforming Fisser's formulation for the use in a Navier-Stokes solver has been unsuccessful. This can largely be attributed to the unsuccessful attempt to create a formulation that allows for a curved domain.

3 Literature Review

In this section relevant literature for this project will be reviewed, starting with an overview of what the mimetic method is. Next the conservation of angular momentum in other implementations and their effects are discussed, making the case for the importance of this property to be conserved. Lastly a look will be taken at meaningful test cases that can be used to test flow problem.

3.1 Mimetic Methods

The mimetic discretization technique is relatively new approach to the solution of partial differential equations. In mimetic methods the discretization of a partial differential equation is done by associating the quantities that interact on specific geometric objects. An example of such a discretization is associating the flux with a surface in three dimensions. The first to work on mimetic discretization techniques were Tonti, [41] and Dodziuk, [14]. As stated by Bochev and Hyman, [6] the realisation was made that while separate mimetic finite differences, finite volumes and finite element methods had been achieved, what they shared on a more deeply underlying level was a algebraic version of discrete vector calculus. In order to link physical quantities and geometry the ideas of exterior calculus are used. In exterior calculus differential forms and their interactions are studied and for instance, includes a generalisation of Stokes's theorem, The divergence theorem and the gradient. The use of the exterior calculus and differential forms has allowed these methods to exactly conserve quantities. Since the realisation that exterior calculus can be used to create mimetic numerical methods numerous methods have been proposed. Bossavits, [7] proposed a finite element method based on differential forms for the equations governing electromagnetism. Finite difference methods for the same set of equations were for instance developed by Hyman and Shashkov, [28] and Brezzi and Buffa, [9]. Methods for the linear elastisicy equations were for instance created by Da Veiga,[11] and very recently by Yi, Fisser and Gerritsma, [45]. The Stokes equations also have been tackled successfully, such as by Kreeft and Gerritsma,[32].

3.2 Conservation of Angular Momentum

When properly discretized, mimetic methods allow for much stricter conservation of physical quantities at a discrete level. The usual three, mass, momentum and energy are the basis of most fluid solvers. Usually in fluid dynamics the conservation of angular momentum is not explicitly conserved because on a continuous level this is not even necessary. At the continuous level the conservation of linear momentum implies the conservation of angular momentum, if the stress tensor is symmetric. As described in, [12] the angular momentum can be defined as in Equation 1

$$\boldsymbol{w} = \boldsymbol{u} \times \boldsymbol{x}. \tag{1}$$

Where $\boldsymbol{w}$ is the angular momentum, $\boldsymbol{u}$ the velocity and $\boldsymbol{x}$ the location in the domain. Creating an conservation law for this quantity the following is obtained,

$$\partial_t(\rho \boldsymbol{w}) + \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u} \times \boldsymbol{x}) + \nabla \times (p\boldsymbol{x}) = 0. \tag{2}$$

Where ρ is the density and p the pressure. This equation becomes redundant when the linear momentum equation (Equation 3) is filled into Equation 2

$$\partial_t(\rho \boldsymbol{u}) + \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u}) + \nabla p = 0. \tag{3}$$

Thus the angular momentum is usually not included in numerical schemes because in the limit on a fine mesh the angular momentum is conserved if linear momentum is conserved and a symmetric Cauchy stress tensor is applied. In practice however the fine mesh limit is not really reached, and errors are present in both the conservation of linear momentum and by extension the angular momentum. On true Cartesian meshes the problem is less severe but nonetheless still existent, [12].

3.2.1 Rotating Shell Problem

To show how significantly the conservation of angular momentum can influence fluid simulations Després et al., [12] considered a set of problems using numerical methods. The first is the rotating shell problem where a fluid ring is rotating around a central axis. This problem is equivalent to a pure convection problem where no viscosity is involved, thus simply letting the fluid rotate on its own. The only boundary conditions applied are that there is no flow normal to the walls on the inside and outside of the ring, sealing the material inside it. The formulation used is a Lagrangian formulation, moving the mesh as a means to incorporate the convective term in this convection problem. If the conservation of angular momentum is not included in the formulation the results do not even reach the total prescribed time for the simulation, instead the simulation breaks down after 2.56 seconds out of a full rotation in 2π seconds. The reason for the breakdown is easily visible in Figure 1 where the mesh at the boundary is incredibly skewed as the cumulative result of calculating the velocity slightly wrong at each time step. This skewness makes it hard for a numerical method to accurately represent the solution on the mesh, and as a result the simulation will blow up. This is in stark contrast to what can be viewed in Figure 2 where the conservation of angular momentum is taken into account explicitly. It can be seen that the simulation not only completed the entire rotation but that the resulting mesh is almost identical to the initial mesh, suggesting that the addition of the conservation of angular momentum is crucial for the correct convection of the medium.

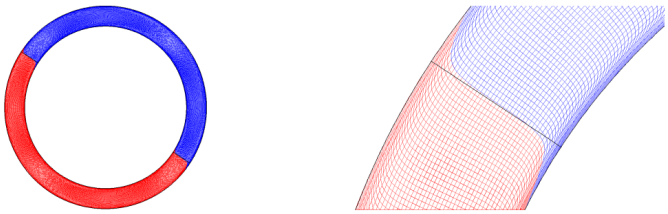

Figure 1: Rotating Shell problem, non angular momentum conservative, [12]. Here it can clearly be seen that the cells near the edges are skewed and the simulation crashes due to the deformation of the mesh, disallowing further computation.

3.2.2 Implosion Problem

It appears that not only in the Lagrangian convection method but also on flux based methods the conservation of angular momentum can make an impact on the solution. Despres et al.

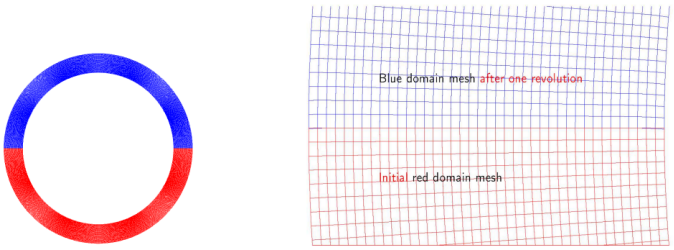

Figure 2: Rotating Shell Problem, angular momentum conservative. [12]. Here the angular momentum conservative scheme is used and the difference in results is readily apparent, not only does the simulation finished as planned, the accuracy of the solution is remarkable.

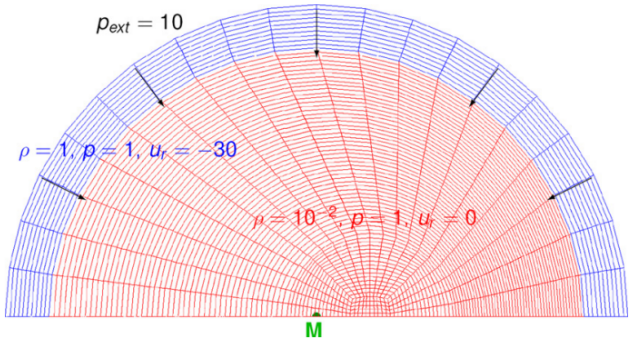

Figure 3: The implosion problem setup and mesh from,[12].

also considered the implosion of heavy gas on to a light gas on a fixed but skewed semicircular grid. Here the skewness can be seen in Figure 3 where it can be seen that the inner semicircle has its center moved to the right in order to create non alignment of the flow with the mesh, considering that an implosion problem is a radially symmetric problem.

The difference for this implosion problem between an angular momentum conserving scheme versus one that does not is most easily seen in the first order method that Després presents. The difference can be seen in Figure 4, where it is visible that the implosion was directed towards the center of the deformed mesh in the scheme that was not conserving angular momentum. In the scheme that does conserve angular momentum the implosion is simply directed to the actual center of the geometry as should be the case for an implosion case with symmetric boundary conditions and initial conditions.

3.3 Test Cases

In order to test new numerical methods it is standard practice to start with a simple problem. The reasons are of course obvious to do this, it takes much less time to construct a method for a simple problem and thus many more iterations can be made before committing to a more complex problem. Almost all new numerical methods start with formulations for the

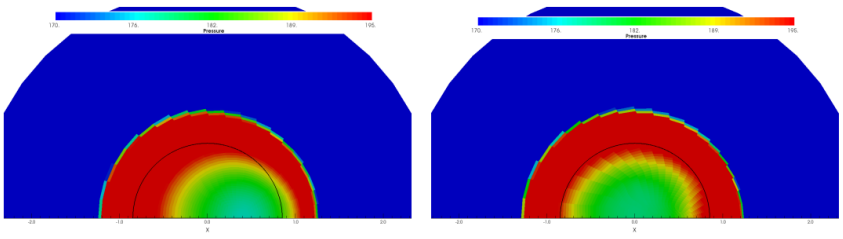

Figure 4: Mesh Imprint of a first order method comparing schemes that do (on the right) and don't (on the left) conserve angular momentum, [12].

Poisson problem, which has simple exact solutions and is only a single scalar equation. The next steps usually is taking the new formulation to the intended equations and creating a so called manufactured solution. This is a solution that does not necessarily has any semblance to a physically meaningful solution, but is intended to test whether the method is implemented correctly. Finally, if the equations are a familiar problem in the scientific community it is likely that reference cases for the problem exist. These reference problems have been studied with well tested methods and some may even be calculated using much more computational power than required for the problem to be considered solved.

3.3.1 Poisson

The Poisson equations are good a test for new numerical methods to solve and with very good reason. The Poisson equations are linear and for simple geometric domains has an exact solution. The fact that the solution to the problem can be written down unambiguously means that very exact error analysis can be performed on the new numerical method. In the case of developing a new method, the results of the formulation can usually be checked thoroughly in order to understand why (or why not) the method is working. The mimetic spectral element method has thus also been applied to the Poisson equations, both scalar and vector version have been tested, [37]. The Poisson problem is also useful in that some flows reduce to a Poisson equation such as Darcy flow, which is flow through a porous medium. A solution can be seen in Figure 5 where flow is considered through a medium with varying porosity.

3.3.2 Method of manufactured solutions

The method of manufactured solution, [38] is a method to test the correct implementation of the method for the set of equations. An analytical solution is created, which does not even have to be a physical solution to the equations. The analytical solution is then used to calculate the appropriate boundary conditions and forcing terms such that the solution to the problem is the analytical solution. All these conditions are then applied to the new numerical method and the method should then produce a solution that is close to the analytical solution. This method can also be used to make error analysis of the problem due the availability of an exact solution.

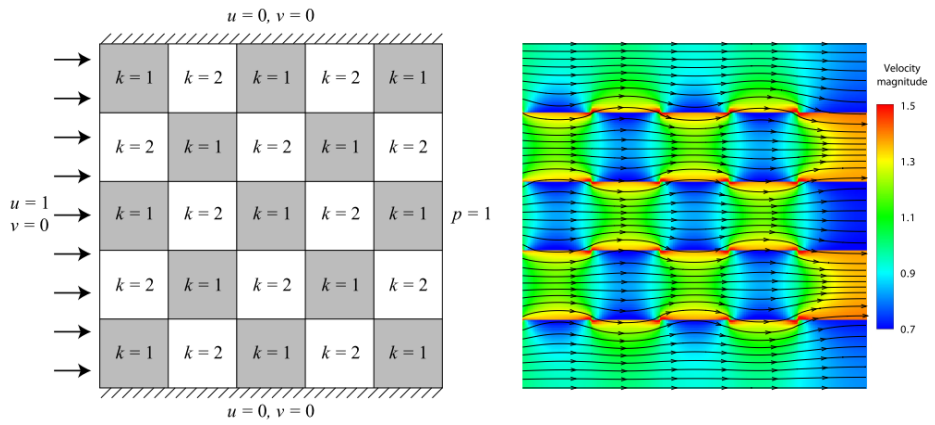

Figure 5: Checkerboard pattern Darcy flow using mimetic discretization. On the left: problem set up. On the right: Solution using the mimetic discretization. Figure taken from [37].

3.3.3 Plane Poiseuille Flow

The Plane Poiseuille flow is probably one of the simplest flow problems with real life application that can be considered. Poiseuille flow in general considers steady flow through non-changing geometry, of which the plane Poiseuille is even one of the simpler variants. The flow is a two dimensional flow between 2 parallel solid walls. In both the Stokes equations and the Navier-Stokes equations the solution is an exact solution and they are the same. To see this starting from the steady two dimensional Navier-Stokes equations.

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \frac{1}{\rho}\frac{\partial p}{\partial x} + \mu(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}), \tag{4}$$

$$u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = \frac{1}{\rho}\frac{\partial p}{\partial y} + \mu(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}), \tag{5}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \tag{6}$$

The assumption can be made that the flow does not vary in the horizontal direction and from this alone the equations reduce to.

$$\frac{\partial p}{\partial x} = \mu\frac{\partial^2 u}{\partial y^2}, \tag{7}$$

$$\frac{\partial p}{\partial y} = 0. \tag{8}$$

Where it can be seen that the convective term has dropped out, making the Stokes solution equal to the Navier-Stokes solution. The solution itself is of course in general.

$$u = \frac{1}{\mu}(-\frac{\partial p}{\partial x})\frac{y^2}{2} + C_1 y + C_2. \tag{9}$$

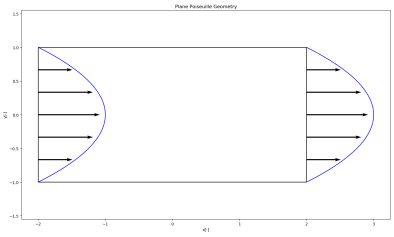

Figure 6: Geometry for the Plane Poiseuille flow, note the inflow and outflow parabolic profile

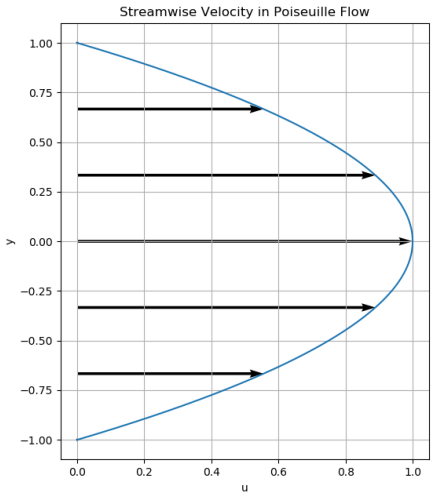

Figure 7: Stream wise velocity plot for the Poiseuille flow solution using a normalized profile such that the maximum velocity is 1

With the well known boundary conditions of no slip walls the solution is a parabolic profile in the entire domain with both at the top and bottom wall u = 0.

3.3.4 Lid Driven Cavity Flow

The lid driven cavity flow is a more interesting test for both the Stokes and Navier-Stokes equations. The simple geometry usually consists of a square or rectangle usually mean that simple Cartesian grids can be used and thus new numerical methods are relatively easy to implement in this way. Also the boundary conditions are simple, no slip wall conditions on all walls except for the upper wall. The upper wall has a unit velocity boundary condition to either the left or right. Because of these boundary conditions for instance the conservation of mass is very easy to check, there is no inflow and there is no outflow in the domain so the same mass should be in the domain at all times. The Stokes equations have a nice symmetrical result and is an excellent target for a new numerical method, and many formulation have been created for this test problem, [32, 40] and the mimetic method result can be seen in Figure 8. For the Navier-Stokes equations the lid driven cavity has also been studied extensively, [5, 8, 17] though the exact behavior at high Reynolds number is still up for debate, [16] solutions can be created for both the steady and unsteady problem.

At high Reynolds numbers the flow starts showing small vortical structures in the corners. The vortices also induce separation. The presence of these vortices make this test case a very good candidate for comparison using the conservation of angular momentum. It is extraordinary how such simple geometry and boundary conditions create incredibly complex behaviour in fluid flow, which make this an ideal test case. The geometry is summarized in Figure 9 and an example result at high Reynolds numbers is given in Figure 10.

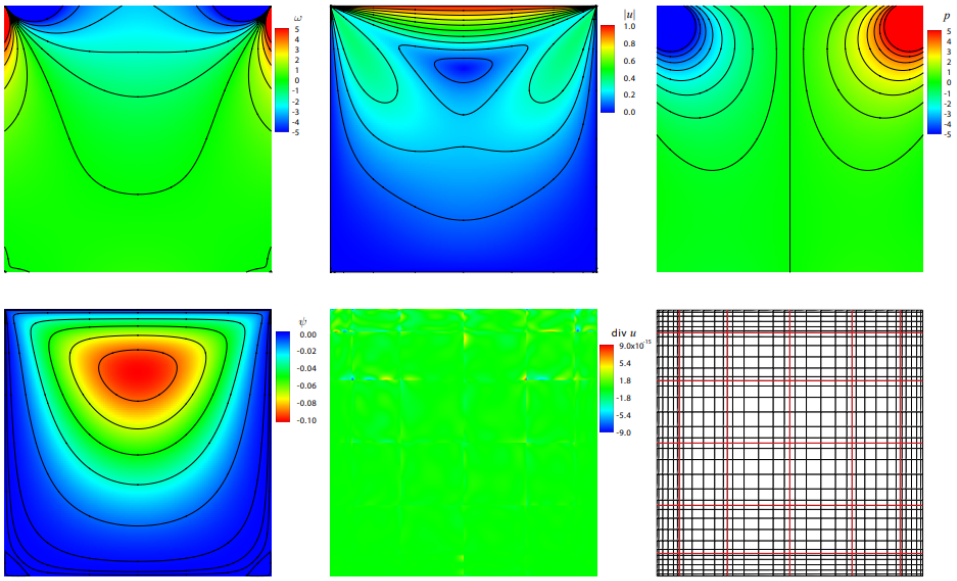

Figure 8: Stokes solution to the lid driven cavity problem using the mimetic spectral element method, Figure taken from [32]

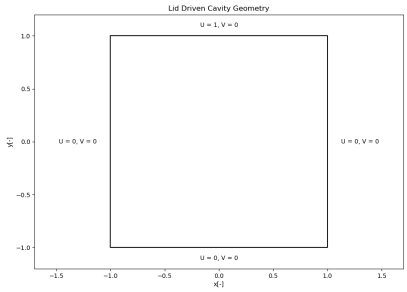

Figure 9: Geometry and boundary conditions for the lid driven cavity flow

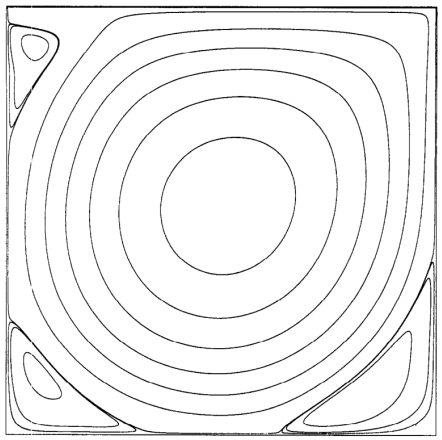

Figure 10: Streamlines for the lid driven cavity flow at a Reynolds number of 5000. Figure taken from [5]

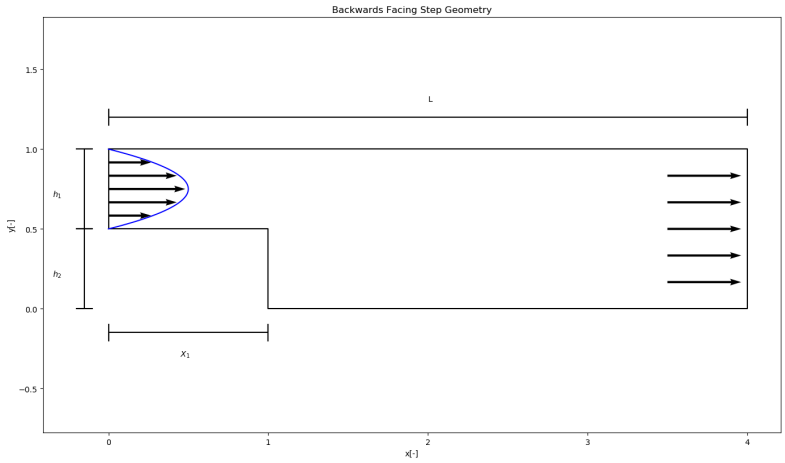

Figure 11: Example geometry definition for the backwards facing step problem with $L = 4.0$, $h_1 = 0.5$, $h_2 = 0.5$, $X_1 = 1.0$. Note the parabolic shaped velocity inflow condition on the left that is in based on the Poiseuille flow and the outflow boundary on the right

3.3.5 Backwards Facing Step

The backwards facing step is similar to the lid driven cavity in the sense that it has a very simple geometry, but produces a very complex flow at high Reynolds numbers. The step itself incurs separation at almost any Reynolds number due to being infinitely sharp, and thus the correct prediction of reattachment and other effects further downstream are incredibly important for practical internal flows. Examples of such practical flows include turbines, diameter changing pipes and diffusers, [4]. Just like the lid-driven cavity flow many reference solutions exist and many numerical schemes have been implemented to investigate the behavior of flow of this problem, [15, 21, 25]. An example for the geometry of the backwards facing step can be found in Figure 11 and solutions at multiple Reynolds numbers can be seen in Figure 12

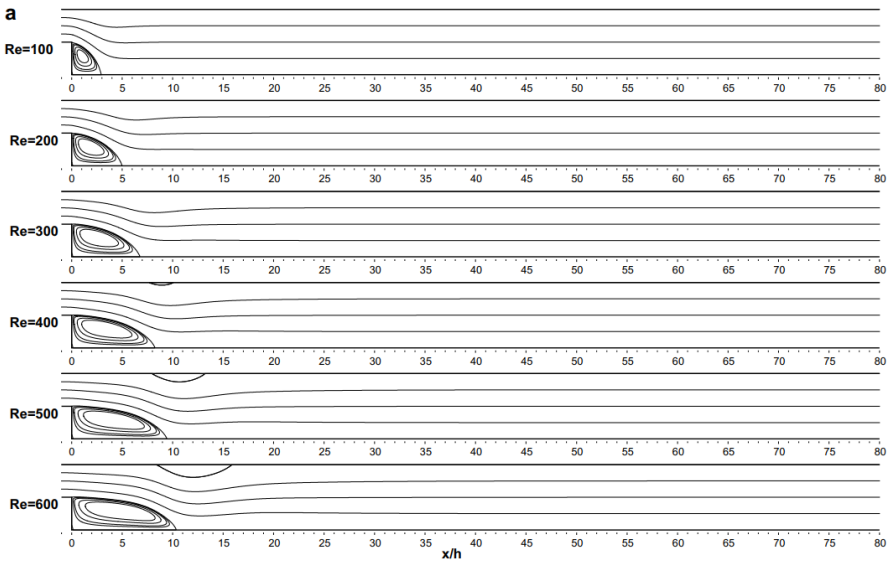

Figure 12: Example results for the backwards facing step. Note the separated and recirculating area. Figure taken from [15]

4 Supporting Math

In this section the supporting math for the mimetic spectral element method will be considered.

4.1 Vectors and Differential Forms

To understand differential forms it is best to start with a short recap of functions and vectors. A function is defined on a domain, and in the case of the mimetic spectral element method each element is a sub-domain. In its most basic form, a vector is a collection of numbers, [19]. The collection of numbers can also be a function of the domain, creating a vector field. A vector can thus be defined as a set of numbers multiplied with a set of basis vectors. These basis vectors span the local vector space at a point in the domain. Beyond this the basis vectors of the local vector space can vary throughout the domain.

$$\boldsymbol{X} = \sum_{i}^{n} X^i \boldsymbol{e}_i, \boldsymbol{X} \in \mathbb{R}^n. \tag{10}$$

Where X^i are the vector components and $\boldsymbol{e}_i$ are the basis vectors. Vectors are also known as contravariant vectors, mainly because when the vectors are transformed using a map the components vary in a way that is against the scaling of the mapping. When taking the exterior derivative of a function on a domain a different kind of vector emerges, the co-vector. The exterior derivative is given by, [19].

$$da^{(0)} := \sum_{i} \frac{\partial a}{\partial x^i} dx^i. \tag{11}$$

Which holds for all forms. A scalar function is considered a 0-form and taking the exterior derivative will increase the order of the form to a 1-form. This 1-form is the same as a co-vector field. More generally, taking an n-form and applying the exterior derivative produces an $(n+1)$-form. For a 1-form in 3 dimensional euclidean space with local coordinate system (x^1,x^2,x^3) the exterior derivative becomes.

$$da^{(1)} = \left(\frac{\partial a_2}{\partial x^3} - \frac{\partial a_3}{\partial x^2}\right) dx^2 \wedge dx^3 + \left(\frac{\partial a_1}{\partial x^3} - \frac{\partial a_3}{\partial x^1}\right) dx^3 \wedge dx^1 + \left(\frac{\partial a_2}{\partial x^1} - \frac{\partial a_1}{\partial x^2}\right) dx^1 \wedge dx^2. \tag{12}$$

An important theoretical result is that applying the exterior derivative twice on any form produces zero.

$$dda^{(k)} := 0 \; \forall a^{(k)} \in \Lambda^k(\mathcal{M}). \tag{13}$$

4.2 Maps

A map is a mathematical object that relates two locations on two different domains to each other. In the case of the mimetic spectral element method the most important use is the

map between the physical domain and the computational domain. Applying the map thus results in the transfer from one to the other. In two dimensions this means

$$\Phi^* : a(x, y) \to a(\xi, \eta), \tag{14}$$

and inversely, the inverse map maps back from the physical domain to the computational domain.

$$\Phi^{-*} : a(\xi, \eta) \to a(x, y). \tag{15}$$

The Jacobian of the mapping is of great use, because it allows the transformation of the differential basis. The Jacobian of the map is given by.

$$\boldsymbol{\Phi}^* = \boldsymbol{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix}. \tag{16}$$

Its inverse is given by.

$$\boldsymbol{\Phi}^{-*} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} = \frac{1}{|\Phi^*|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \eta} \end{bmatrix}. \tag{17}$$

Applying the Jacobian to the differential basis of the physical domain results in,

$$\Phi^*(\mathrm{d}x) = \frac{\partial x}{\partial \xi}\mathrm{d}\xi + \frac{\partial x}{\partial \eta}\mathrm{d}\eta, \tag{18}$$

$$\Phi^*(\mathrm{d}y) = \frac{\partial y}{\partial \xi}\mathrm{d}\xi + \frac{\partial y}{\partial \eta}\mathrm{d}\eta. \tag{19}$$

and the inverse applied tot the computational domain basis,

$$\Phi^{-*}(\mathrm{d}\xi) = \frac{1}{|\Phi^*|}(\frac{\partial y}{\partial \eta}\mathrm{d}x - \frac{\partial x}{\partial \eta}\mathrm{d}y), \tag{20}$$

$$\Phi^{-*}(\mathrm{d}\eta) = \frac{1}{|\Phi^*|}(-\frac{\partial y}{\partial \xi}\mathrm{d}x + \frac{\partial x}{\partial \xi}\mathrm{d}y). \tag{21}$$

linking the two domains together. Another operation that is important for differential forms is the wedge product. The wedge product takes two differential n- and k-forms and combines them to an $(n + k)$-form and is skew symmetric.

4.3 Hodge-$\star$ Operator

The Hodge-$\star$ operator is most easily understood as the operator that creates the dual to a chain of forms, with the dual to the a k form being a $(n-k)$-form with n being the dimension of the space. In 2 dimensions the Hodge -$\star$ operator applied to simple forms become.

$$\star 1 = dx \wedge dy,\ \star dx = dy,\ \star dy = -dx,\ \star(dx \wedge dy) = 1. \tag{22}$$

To achieve this operation the Hodge-$\star$ operator is defined as an inner product of 2 forms.

$$a^{(k)} \wedge \star b^{(k)} := (a^{(k)}, b^{(k)})\mathrm{vol}^n. \tag{23}$$

4.4 Numerical Integration

Numerical integration in one dimension is usually achieved using quadrature rules. These quadrature rules are typically defined on the domain [-1,1] and work by evaluating a function at certain locations and using a weighted sum of those evaluations. Effectively, most quadrature rules are of the form.

$$\int_{-1}^{1} f(x)dx \approx \sum_{i}^{n} w_i f(x_i). \tag{24}$$

where x_i is the node of the function to be evaluated, and w_i is the weight that is associated with the node. In the mimetic spectral element method the use of Gauss-Lobatto-Legendre quadrature is preferred due to this being a quadrature rule with the highest accuracy while including the endpoints of the function to be evaluated. The Gauss-Lobatto-Legendre nodes are related to the Legendre polynomial, [20, 27] in that their nodes are the roots of the function,

$$(1-\xi^2)L'_N(\xi), \tag{25}$$

where $L'_N(\xi)$ is the derivative of the N-th Legendre polynomial, [29]. The weights associated with the Gauss-Lobatto-Legendre quadrature are then, [1].

$$w_i = \frac{2}{N(N+1)(L_{N-1}(x_i))^2}. \tag{26}$$

From the one dimensional definition of a quadrature rule it is possible to create quadrature rules for higher dimensions by creating a multi-dimensional grid from the nodes of the rule. With the weight of a node x_i corresponding to $\mathrm{w}(x_i)$ the rule for example in a 2 dimensional evaluation of an integral becomes.

$$\int_{-1}^{1}\int_{-1}^{1} f(x,y) \approx \sum_{i}^{n}\sum_{j}^{n} f(x_i,y_j)w(x_i)w(y_j). \tag{27}$$

4.5 Basis Functions

The mimetic spectral element method uses a very specific set of basis functions. Starting with the nodal degrees of freedom being defined as the value of the function at the nodes, the nodal basis functions (that are associated with 0-forms) are constructed as the Lagrange polynomials on the Gauss-Lobbato-Legendre nodes. For any element of any dimension these nodal basis functions can all be constructed from the one dimensional Lagrange basis functions. The Lagrange Basis functions are constructed from.

$$f_i(x) = \prod_{k=1,k\neq i}^{p+1} \frac{x-x_k}{x_i-x_k}. \tag{28}$$

Which have the very useful property that they are exactly 1 on one of the nodes of the polynomial and zero on all the other nodes.

$$f_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{if j} = \text{i} \\ 0 & \text{if j} \neq \text{i} \end{cases}. \tag{29}$$

Combining the definition of the Lagrange polynomials with the definition for the nodes of the Gauss-Lobatto-Legendre polynomials the nodal basis functions in one dimension become, [29].

$$h_i(\xi) = \frac{(\xi^2 - 1)L'_n(\xi)}{N(N+1)L_n(\xi_i)(\xi - xi_i)}, i = 1, ..., N. \tag{30}$$

As such, if $\mathcal{N}_i^0$ are considered the degrees of freedom at the nodes, then

$$\mathcal{N}_i^0(h_j) = \delta_{ij}. \tag{31}$$

In a one dimensional element it is also useful to define basis functions on the line segments or edges that connect the nodes of the grid. These functions will be associated with the 1-form in a one dimensional element. It is important to note that these edge basis function, [23, 29], are associated with degrees of freedom that use integral values of a function on that piece of line.

$$\mathcal{N}_i^1(p^h) := \int_{\xi_{i-1}}^{\xi_i} p^h(\xi) d\xi \tag{32}$$

with p^h being a polynomial. The quickest way to define polynomials that satisfy this property is to construct them from the previous nodal basis functions.

$$e_j(\xi) = -\sum_{k=0}^{j-1} \frac{dh_k}{d\xi}(\xi) \tag{33}$$

and from the proof for this,

$$\int_{\xi_{i-1}}^{\xi_i} e_j(\xi) d\xi = -\sum_{k=0}^{j-1} \int_{\xi_{i-1}}^{\xi_i} \frac{dh_k}{d\xi}(\xi) = -\sum_{k=0}^{j-1} [h_k(\xi_i) - h_k(\xi_{i-1})] = \delta_{ij}, \tag{34}$$

it is possible to deduce that the derivative of any nodal basis function can be constructed from only 2 edge functions.

$$\frac{dh_j}{d\xi} = e_j(\xi) - e_{j+1}(\xi). \tag{35}$$

The nodal and edge basis functions are used to construct the primal grid basis functions in any dimension, in Figure 13 these basis functions are shown for $N = 4$.

The mimetic numerical methods are based on the double DeRham sequence, see Figure 14 and the above basis functions are for the primal grid. This means that also basis functions for the dual grid must be constructed. A good way of doing this is using an inner product, [29]. This inner product results in a mass matrix that can be calculated using the Gauss-Lobatto-Legendre numerical integration method described above. The construction of all

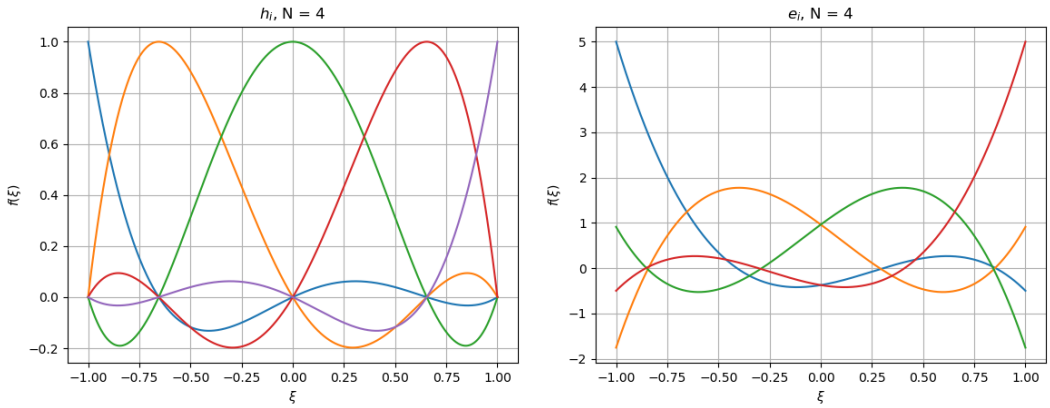

Figure 13: left: $h_i(\xi)$ Basis functions, note the points where basis functions intersect on the y axis, the Gauss-Lobatto-Legendre points. Right: $e_i'(\xi)$ Basis functions

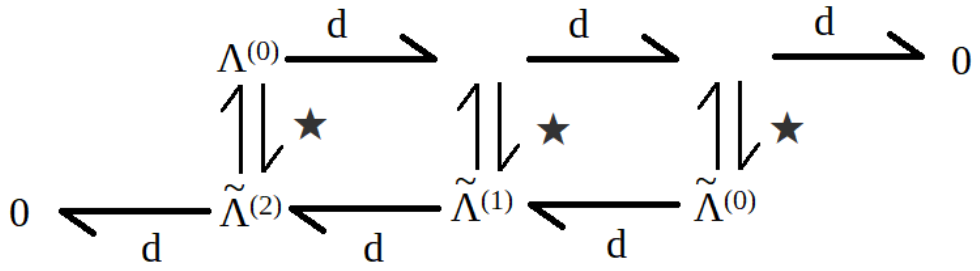

Figure 14: Double DeRham sequence used in mimetic discretization

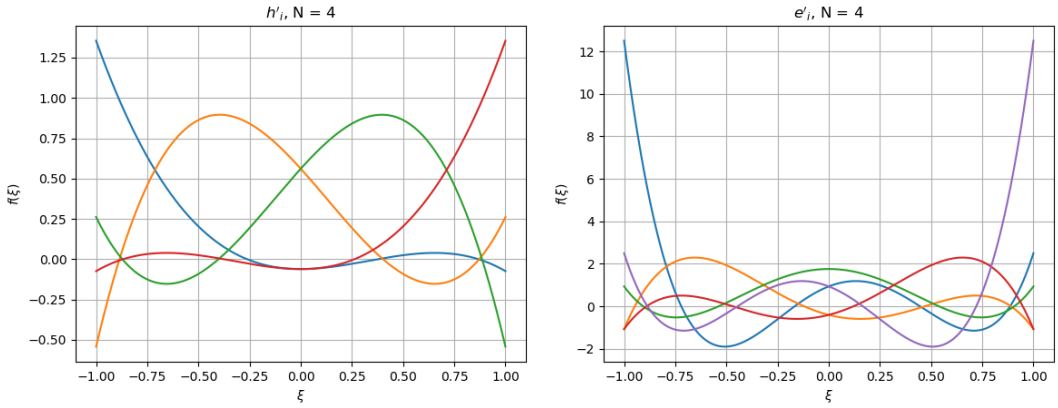

Figure 15: left: $h'_i(\xi)$ Dual Nodal Basis functions. Right: $e_i(\xi)$ edge Dual Basis functions

the basis functions start with the construction of the one dimensional basis functions. The mass matrix for the nodal basis functions is calculated according to.

$$\mathbb{M}^{(0)}_{i,j} = \sum_{k=1}^{n+1} w_k h_i(\xi_k) h_j(\xi_k), \tag{36}$$

which through the useful Dirac function like property of the Lagrange Polynomials at their nodes (seeEquation 29) results in a sparse diagonal matrix with the weights of the Gauss-Lobatto-Legendre method on the diagonal. For the mass matrix of the edge based function the same procedure applies and the resulting mass matrix is.

$$\mathbb{M}^{(1)}_{i,j} = \sum_{k=1}^{n+1} w_k e_i(\xi_k) e_j(\xi_k) \tag{37}$$

To produce the dual basis functions these mass matrices are inverted and applied to the vector that contains the basis functions. Their appearance can be found in Figure 15.

$$e'_i(\xi) = (\mathbb{M}^{(0)})^{-1} h_i(\xi), \tag{38}$$

$$h'_i(\xi) = (\mathbb{M}^{(1)})^{-1} e_i(\xi) \tag{39}$$

4.6 Multi Dimensional Grids

To create a multi dimensional mimetic spectral element the grid points of the one dimensional Gauss-Lobatto-Legendre points are arranged in a multidimensional orthogonal grid for the computational geometry. This work will mainly consider the two dimensional grid and an example of a 2 dimensional grid can be found in Figure 16. In the two dimensional case it can also clearly be seen that there are three distinct different geometrical objects, points, lines and surfaces. In the two dimensional grids the degrees of freedom are assigned

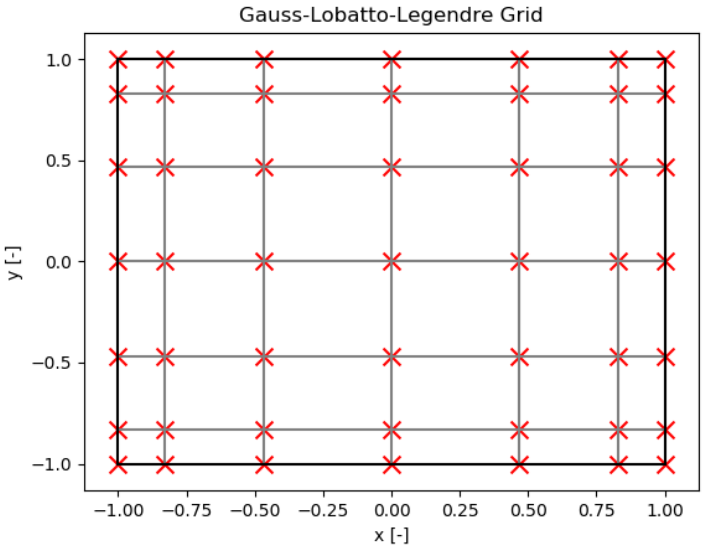

Figure 16: 2 dimensional Gauss-Lobatto-Legendre grid with 7 nodes per dimension

to these objects. Higher dimensional grids will of course use appropriate geometric objects for their dimension, [34].

These degrees of freedom need their own basis functions. The simplest basis functions are the nodal basis functions. The multidimensional dimensional basis functions are a tensor product of the one-dimensional nodal basis functions, in two dimensions the result is,

$$f(\xi,\eta) = \sum_{i=0}^{n+1}\sum_{j=0}^{n+1} C_{i,j} h_i(\xi) h_j(\eta), \tag{40}$$

with the $C_{i,j}$ being the expansions coefficient or the degrees of freedom. This is considered the 0-form in 2 dimensions, and the 1-form is most easily obtained through taking the exterior derivative, [19, 34] of the nodal functions.

$$df(\xi,\eta) = \sum_{i=1}^{n+1}\sum_{j=0}^{n+1}(C_{i,j} - C_{i-1,j}) e_i(\xi) h_j(\eta) d\xi + \sum_{i=0}^{n+1}\sum_{j=1}^{n+1}(C_{i-1,j} - C_{i,j}) h_i(\xi) e_j(\eta) d\eta. \tag{41}$$

Where the difference between the original expansion coefficient of the nodal functions$(C_{i,j} - C_{i-1,j})$ and $(C_{i-1,j} - C_{i,j})$ define a matrix refered to as an incidence matrix. For the exterior derivative from a 0 to a 1-form, the matrix is named $\mathbb{E}^{(1,0)}$ defined in Equation 42 for $n = 2$. The basis for functions on a 1 form is thus $\{e_i(x)h_j(y), h_i(x)e_j(y)\}$ which are called the edge functions because they are defined over an edge, this can be seen in Figure 17.

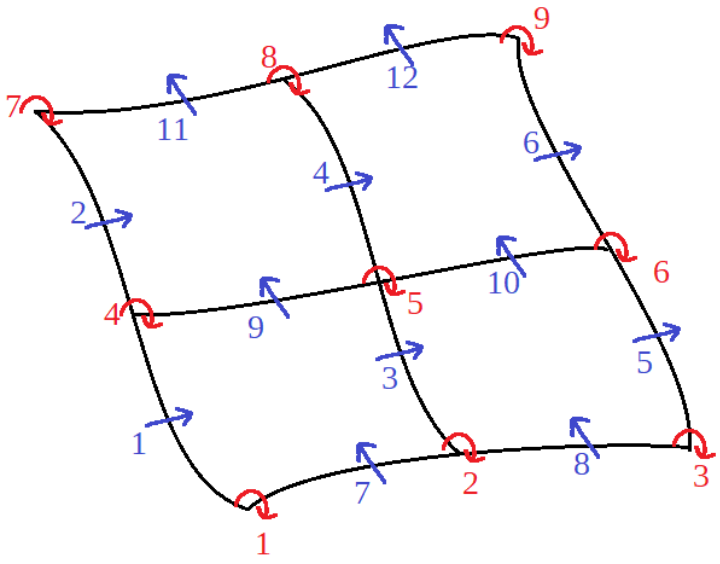

Figure 17: Numbering used to define the $\mathbb{E}^{(1,0)}$ matrix, red numbers are corners, blue numbers are edges, $n = 2$

$$\vec{E} = \mathbb{E}^{(1,0)}\vec{P}, \quad \begin{bmatrix} E_1 \\ E_2 \\ E_3 \\ E_4 \\ E_5 \\ E_6 \\ E_7 \\ E_8 \\ E_9 \\ E_{10} \\ E_{11} \\ E_{12} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \\ P_9 \end{bmatrix} \tag{42}$$

Applying the exterior derivative to a 1-form expanded with the basis functions $\{e_i(x)h_j(y), h_i(x)e_j(y)\}$ results in a 2-form with expansions using the surface basis functions of the form.

$$f(x,y) = \sum_{i=1}^{n+1}\sum_{j=1}^{n+1} C_{i,j}e_i(x)e_j(y)dx \wedge dy. \tag{43}$$

An incidence matrix can be defined to calculate the expansion coefficients of this 2-form resulting from the exterior derivative applied to the 1-form. This incidence matrix is then called $\mathbb{E}^{(2,1)}$ and is defined in Equation 44 with the numbering defined in Figure 18 for the

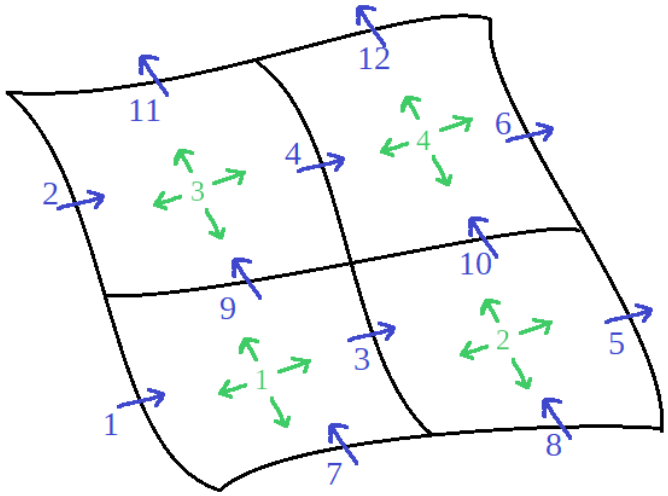

Figure 18: Numbering used to define the $\mathbb{E}^{(2,1)}$ matrix, blue numbers are edges, green numbers are surfaces. n = 2

case of $n = 2$. The basis functions dual to these set of basis functions can then be created by making a mass matrix of the basis functions and applying the inverse of it to the appropriate set of basis functions.

$$\vec{S} = \mathbb{E}^{(2,1)}\vec{E}, \qquad \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \\ E_3 \\ E_4 \\ E_5 \\ E_6 \\ E_7 \\ E_8 \\ E_9 \\ E_{10} \\ E_{11} \\ E_{12} \end{bmatrix} \tag{44}$$

4.7 Mass Matrices

With the basis functions defined on multidimensional grids there is also a requirement to create the dual to these basis function. Just like in the one-dimensional case, the dual can be constructed using an inner product. In two dimensions there are 3 forms, the 0, 1 and

2-form. Each has its distinct mass matrix. To construct any mass matrix, the following definition can be used.

$$(\tilde{a}, a) = \int_{\Omega} \tilde{a} \wedge \star a \tag{45}$$

Because of the properties of the Hodge-$\star$ operator this integral will always be defined. With a 0-form being expanded in the nodal basis functions the Hodge of the basis functions is

$$\star a = a \, \mathrm{d}x \wedge \mathrm{d}y \tag{46}$$

with a being the 0-form. Using this the mass matrix of a 0-form can be calculated as.

$$(\tilde{a}, a) = \int_{\Omega} \tilde{a} \wedge \star a = \int_{\Omega} \sum_{i=1}^{n+1} \sum_{j=1}^{n+1} \sum_{k=1}^{n+1} \sum_{l=1}^{n+1} \tilde{a}_{i,j} a_{k,l} h_i(x) h_j(y) h_k(x) h_l(y) \mathrm{d}x \wedge \mathrm{d}y. \tag{47}$$

and thus using a numerical integration the mass matrix can be calculated as

$$\mathbb{M}^{(0)}_{ijkl} = \sum_{r=1}^{n+1} \sum_{s=1}^{n+1} h_i(x_r) h_j(y_s) h_k(x_r) h_l(y_s) w(x_r) w(y_s), \tag{48}$$

with x_r and y_s being the nodal locations of the integration method $w(x_r)$ and $w(y_s)$ being the weight of a corresponding numerical integration node. The mass matrix associated with a 1-form is a little more involved due to the multiple components that a 1-form has. With a 1-form being expressed as.

$$b = \sum_{i=1}^{n+1} \sum_{j=1}^{n} b^1_{i,j} h_i(x) e_j(y) \mathrm{d}x + \sum_{i=1}^{n} \sum_{j=1}^{n+1} b^2_{i,j} e_i(x) h_j(y) \mathrm{d}y \tag{49}$$

and thus $\star$ b becomes.

$$\star b = \sum_{i=1}^{n+1} \sum_{j=1}^{n} b^1_{i,j} h_i(x) e_j(y) \mathrm{d}y - \sum_{i=1}^{n} \sum_{j=1}^{n+1} b^2_{i,j} e_i(x) h_j(y) \mathrm{d}x \tag{50}$$

Applying the wedge product will then produce terms like dx$\wedge$ dx and dy$\wedge$ dy which are zero. Because the wedge product is skew symmetric the term dy$\wedge$ dx will become -dx$\wedge$ dy, and thus the mass matrix become a block diagonal matrix. If the components of the expansions are stacked in a vector like.

$$\tilde{\boldsymbol{b}} = \begin{bmatrix} \tilde{b}^1_{i,j} \\ \tilde{b}^2_{i,j} \end{bmatrix}, \tag{51}$$

$$\boldsymbol{b} = \begin{bmatrix} b^1_{i,j} \\ b^2_{i,j} \end{bmatrix} \tag{52}$$

The mass matrix can be calculated as a single sum like.

$$(\tilde{b} \wedge \star b) = \int_\Omega \sum_{i=1}^{n+1} \sum_{j=1}^{n} \sum_{k=1}^{n+1} \sum_{l=1}^{n} \tilde{b}^1_{i,j} b^1_{k,l} h_i(x) e_j(y) h_k(x) e_l(y) \mathrm{d}x \wedge \mathrm{d}y + \tag{53}$$

$$\int_\Omega \sum_{i=1}^{n} \sum_{j=1}^{n+1} \sum_{k=1}^{n} \sum_{l=1}^{n+1} \tilde{b}^2_{i,j} b^2_{k,l} e_i(x) h_j(y) e_k(x) h_l(y) \mathrm{d}x \wedge \mathrm{d}y. \tag{54}$$

The upper left block of the mass matrix can be can be calculated using the following.

$$\mathbb{M}^{(1)}_{i,j,k,l} = \sum_{r}^{n+1} \sum_{s}^{n+1} h_i(x_r) e_j(y_s) h_k(x_r) e_l(y_s) w(x_r) w(y_s). \tag{55}$$

The lower right block of the mass matrix can be calculated according to.

$$\mathbb{M}^{(1)}_{ijkl} = \sum_{r}^{n+1} \sum_{s}^{n+1} e_i(x_r) h_j(y_s) e_k(x_r) h_l(y_s) w(x_r) w(y_s). \tag{56}$$

The mass matrix of the 2-forms are very similar to the 0-form. The inner product

$$(\tilde{c}, c) = \int_\Omega \tilde{c} \wedge \star c. \tag{57}$$

with c being a 2-form expressed as.

$$c = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{i,j} e_i(x) e_j(y) \mathrm{d}x \wedge \mathrm{d}y. \tag{58}$$

The mass matrix components become

$$\mathbb{M}^{(2)}_{ijkl} = \sum_{r}^{n+1} \sum_{s}^{n+1} e_i(x_r) e_j(y_s) e_k(x_r) e_l(y_s) w(x_r) w(y_s). \tag{59}$$

for additional information on how these mass matrices change under transformations see Appendix B

5 Poisson

The Poisson equations are a good way to introduce the mimetic spectral element method. The equation has exact solutions, particularly in two dimensions in a square or rectangle. With the spectral element being constructed from quadrilateral elements testing solutions to the problem is straightforward.

5.1 Problem Definition

The Poisson equations come in 2 forms, the scalar and vector equations. In this chapter the scalar equations will be discussed and a mimetic spectral element method will be presented for the scalar Poisson equation. For the scalar problem in a domain Ω with Dirichlet boundary conditions the following holds.

$$\Delta p = f \in \Omega, \tag{60}$$
$$p = g \in \partial\Omega. \tag{61}$$

5.1.1 Mathematical Definition

The Poisson problem as a mathematical problem is encountered so widespread through engineering and mathematics that it even has its own operator in the form of Δ, also known as the Laplace operator. Δ can be composed as the sequence $\nabla \cdot \nabla$ and the Poisson problem can also be written as.

$$\nabla \cdot \nabla p = f \in \Omega. \tag{62}$$

For a mimetic method however it is useful to keep the formulation in Equation 62 in mind because this sequence of operation is directly linked with the mimetic method. Formulating the problem in this 2 stage problem immediately links the scalar Poisson problem to 2 of the three fundamental vector calculus operators in three dimensions: the divergence $\nabla\cdot$ and the gradient ∇. To exploit this a little more it is possible to rewrite the scalar Poisson problem in the following way by introducing an intermediate variable $\boldsymbol{q}$

$$\boldsymbol{q} - \nabla p = 0 \in \Omega, \tag{63}$$
$$\nabla \cdot \boldsymbol{q} = f \in \Omega. \tag{64}$$

And in this way the scalar Poisson problem is split into 2 equations each only containing a fundamental vector calculus operation. And interestingly, a Neumann boundary condition can now be specified by enforcing the value of the intermediate variable by setting q to $\hat{p}$ at the boundaries.

$$\boldsymbol{q} = \hat{q} \in \partial\Omega. \tag{65}$$

Multiplying Equation 64 with test functions $\tilde{q}$ and $\tilde{p}$ results in the weak formulation of the Poisson equation.

$$(\tilde{\boldsymbol{q}}, \boldsymbol{q})_\Omega - (\tilde{\boldsymbol{q}}, \nabla p)_\Omega = 0, \tag{66}$$
$$(\tilde{p}, \nabla \cdot \boldsymbol{q})_\Omega = (\tilde{p}, f)_\Omega. \tag{67}$$

with $(...,...)_\Omega$ being the inner product on Ω. Applying integration by parts to these equations leads to the following.

$$(\tilde{\boldsymbol{q}}, \boldsymbol{q})_\Omega + (p, \nabla \cdot \tilde{\boldsymbol{q}})_\Omega = \int_{\partial\Omega} \hat{q}\tilde{\boldsymbol{q}} \cdot \boldsymbol{n} d\Gamma_\Omega, \tag{68}$$

$$(\tilde{p}, \nabla \cdot \boldsymbol{q})_\Omega = (\tilde{p}, f)_\Omega. \tag{69}$$

It is also possible to get the same definition from a functional, [30].

$$\mathcal{L}(p, \boldsymbol{q}, f, \hat{p}) := \int_\Omega \frac{1}{2}|\boldsymbol{q}|^2 d\Omega + \int_\Omega p(div\ \boldsymbol{q} - f) - \int_\Omega \hat{q}(\boldsymbol{q} \cdot \boldsymbol{n}) d\Omega. \tag{70}$$

Taking variations with respect to both q and p the same system is recovered.

5.1.2 Numerical Representation in 2 Dimensions

If we choose the expansions of our solution carefully we can represent the mathematical system of Equation 68 andEquation 69 using the mimetic method. Choosing p to be in the dual points and q in the primal edges the expansions for the basis functions become.

$$u = \sum_{i=0}^{N(N+1)} q_i h(x) e(y), \tag{71}$$

$$v = \sum_{i=N(N+1)}^{2N(N+1)} q_i e(x) h(y), \tag{72}$$

$$p = \sum_{i=0}^{N^2} p_i h'(x) h'(y). \tag{73}$$

with u and v being the components of the flux in the x and y direction respectively. The inner product of the test function $\tilde{\boldsymbol{q}}$ and the flux vector $\boldsymbol{q}$ can be represented by the mass matrix $\mathbb{M}^{(1)}$ and the divergence of $\boldsymbol{q}$ can be represented with the matrix $\mathbb{E}^{2,1}$ because the exterior derivative of a 1-form in 2 dimensions is equal to the divergence. The numerical system then becomes.

$$\begin{bmatrix} \mathbb{M}^{(1)} & (\mathbb{E}^{2,1})^T \\ \mathbb{E}^{2,1} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ p \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} \\ \boldsymbol{F} \end{bmatrix}. \tag{74}$$

where $\boldsymbol{G}$ and $\boldsymbol{F}$ represent the vectors that encode the boundary condition and forcing functions respectively. Using a Schur complement method to eliminate the fluxes creates a more compact formulation. Solving the system just for p (the Lagrange multiplier) results in the following.

$$\mathbb{E}^{(2,1)}(\mathbb{M}^{(1)})^{-1}(\mathbb{E}^{(2,1)})^T p = \mathbb{E}^{(2,1)}(\mathbb{M}^{(1)})^{-1} G - F. \tag{75}$$

And here we can see the power of the mimetic method through the lens of the double DeRham sequence, see Figure 19. The operator, or the matrix that arises when constructing

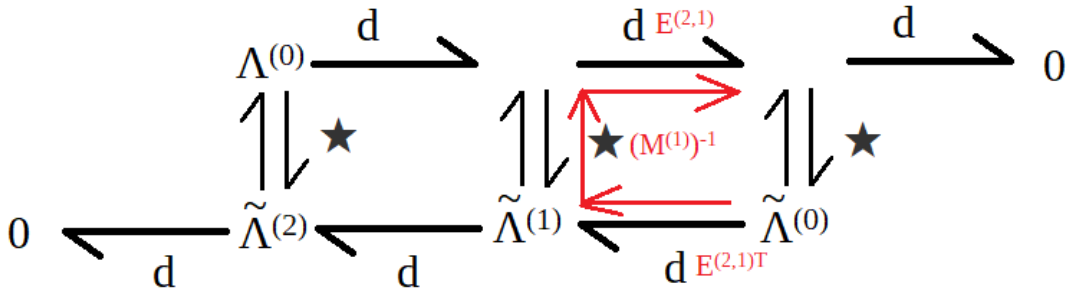

Figure 19: Double DeRham sequence with scalar Laplace path shown in red

the operator can also be constructed by choosing p to be a variable defined on dual points and following the exterior derivative once using the $(\mathbb{E}^{(2,1)})^T$ operator which represents the discrete analog the exterior derivative operator d. Because the exterior operator d can only be applied once (applying it twice results in 0) switching back to the primal space using the inverse of the mass matrix $(M^1)^{-1}$ and applying the discrete exterior operator $\mathbb{E}^{(2,1)}$ again is equivalent to applying the gradient and divergence on the variable p sequentially.

5.1.3 Multiple Element Chaining

A single element is usually not the simplest way to describe complex geometries. It is useful to be able to chain multiple elements together such that these more complex geometries can be more easily created. For the mixed Poisson formulation it is sufficient to couple the fluxes. This can be achieved in 2 ways. For 2 adjacent elements the connection can be made by taking the edges in the mesh that overlap and reducing the variables that these 2 edges represent to a single variable, making the whole domain continuous. Another approach would be to add an equation containing a Lagrange multiplier that forces the difference between the 2 variables to 0, effectively making them one. This is called the hybrid approach. The hybrid method is used extensively in the mimetic spectral element method because rearranging the global variables in the continuous approach greatly increases the complexity in calculating the correct incidence and mass matrices and would result in global dual polynomials instead of local ones. Using the hybrid approach the same procedure can be applied internally to every element for the creation of the mass and incidence matrices and requires only for the elements to be coupled afterwards. Especially when using large geometries and multi core computers the parallelization potential for the hybrid approach is readily apparent. By collecting the new Lagrange multiplier in a separate set of equation the formulation in Equation 74 can be modified with an extra block C like in Equation 76 to connect the elements together.

$$\begin{bmatrix} \mathbb{M}^{(1)} & (\mathbb{E}^{2,1})^T & C^T \\ \mathbb{E}^{2,1} & 0 & 0 \\ C & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ p \\ \lambda_c \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} \\ \boldsymbol{F} \\ 0 \end{bmatrix}. \tag{76}$$

It is noted that the structure of this matrix is very regular in that there are only entries in the first row and the first column of the matrix. Because of this it should be possible

to create a formulation for this problem using a nested Schur complement method similar to Equation 75 but applied twice. Once for the outer blocks C and C^T and once for the inner blocks $(\mathbb{E}^{2,1})^T$ and $\mathbb{E}^{2,1}$. Using this method it is possible to solve for the Lagrange multipliers first and then use that information to calculate the solution element wise.

5.2 Results

To check the implementation of this problem a manufactured solution was created, [38]. The manufactured solution for this problem was chosen to be,

$$p = cos(\pi x)cos(\pi y) + 1, \tag{77}$$

on the domain $[-1, 1]^2$. This function, due to the cosines, does not have a finite polynomial expansion. Because of this the solution can be used to check the approximation made by the use of polynomials without reaching a limit where the solution can be exactly interpolated by those polynomial. This results in the following exact solutions for the other variables.

$$f = \Delta p = -2\pi^2 cos(\pi x)cos(\pi y), \tag{78}$$

$$u = \frac{\partial p}{\partial x} = -\pi sin(\pi x)cos(\pi y), \tag{79}$$

$$v = \frac{\partial p}{\partial y} = -\pi cos(\pi x)sin(\pi y). \tag{80}$$

5.2.1 Convergence

An important aspect of any numerical scheme is its convergence to the analytical solution. Traditionally a method is used to solve a problem on a progressively finer mesh and the error with respect to the true solution is measured. This method of decreasing the mesh spacing is sometimes called h-refinement due to the mesh spacing usually being referred to as "h". The order of the method can then be deduced from the rate at which the numerical solution converges to the true solution. An example of h-refinement can be found in Figure 21 where there is an increase in the amount of elements that form the mesh, but the order of the elements stays the same. For methods that allow for multiple orders of convergence through the arbitrary selection of the order of the method there exists another metric on which to compare the convergence which is called p-refinement. In p-refinement the error is plotted against the order of the method while keeping the spacing of the mesh the same. An example of p-refinement can be found in Figure 22. Because the mimetic method allows for both h- and p- refinement the result for both types of refinement are presented. The exact solution and a calculated solution are presented next to each other in Figure 20.

In Figure 23 the results for the p-convergence in the L_2 error,

$$||e|| = \sqrt{\int_\Omega (p_{exact}(x, y) - p_{discrete}(x, y))^2 \mathrm{d}\Omega}. \tag{81}$$

In Figure 24 the convergence with respect to h is shown.

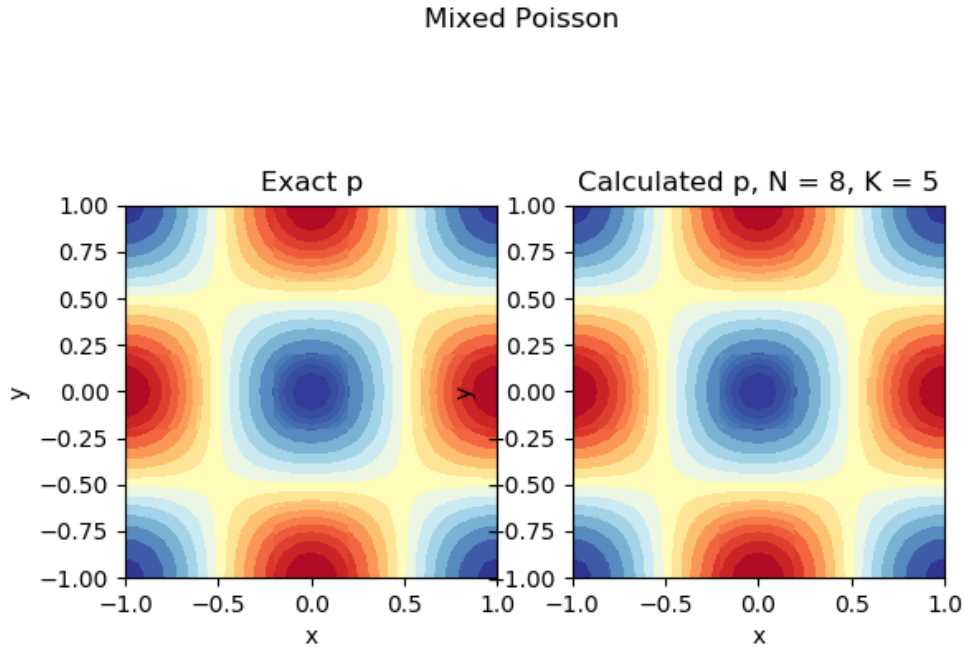

Figure 20: Exact manufactured solution (left) of p vs calculated solution using 25 elements (K = 5) 8th order elements

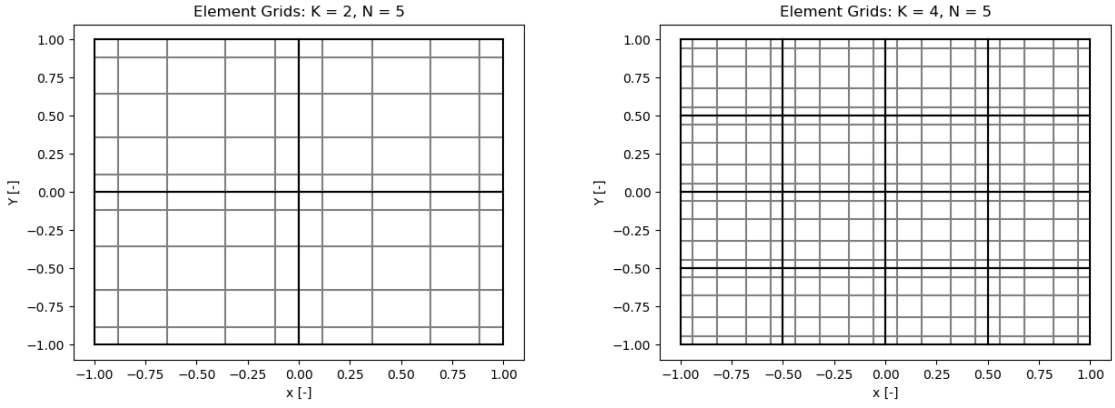

Figure 21: h-refinement of a mesh, note that the black borders are the element borders and the gray line are the internal grid lines for an element

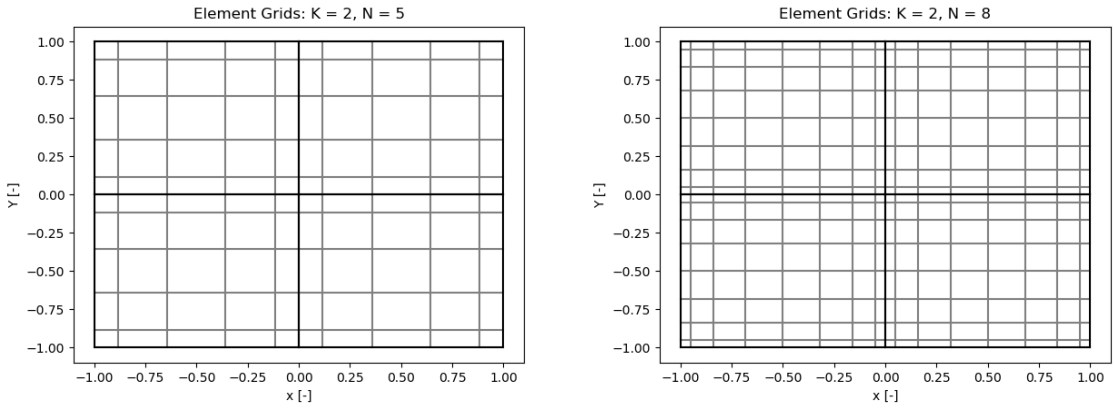

Figure 22: p-refinement of a mesh, again note that the black borders are the element borders and the gray line are the internal grid lines for an element

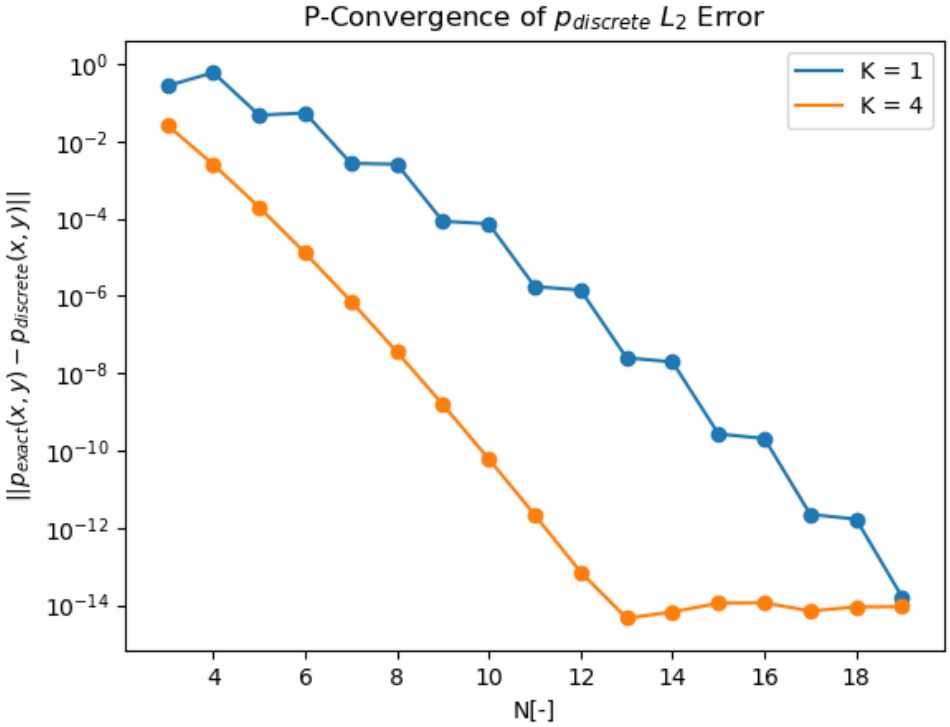

Figure 23: p-refinement effect on the error for the manufactured solution.

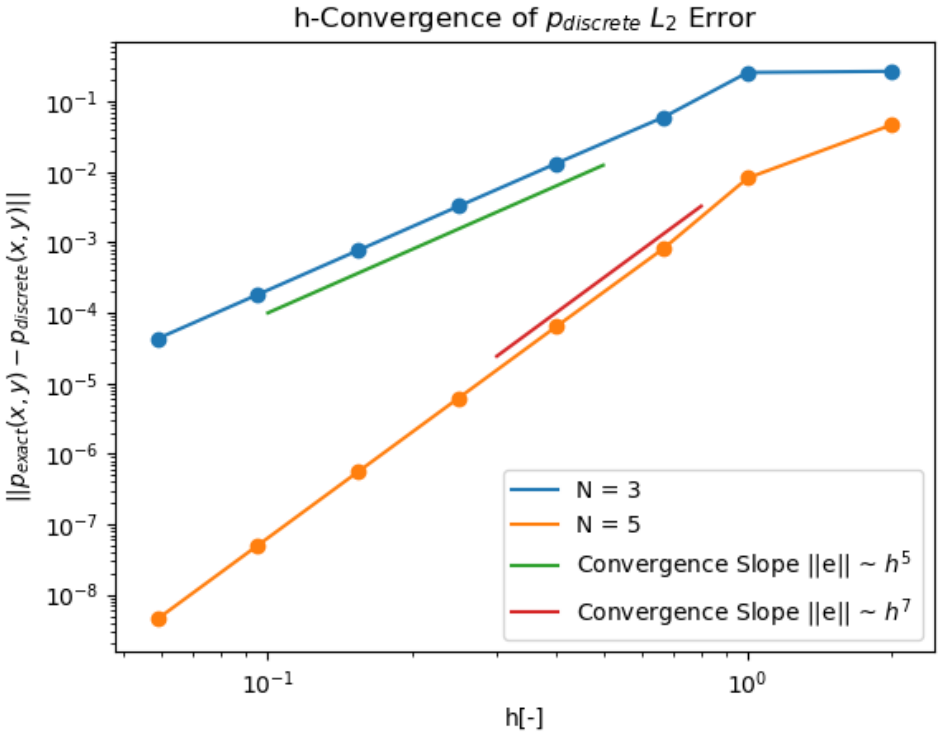

Figure 24: h-refinement effect on the error for the manufactured solution

6 Stokes Equations

The Stokes equations are the equations that govern flows where convection of the flow has negligible influence on the flow. This is the case in flows with low Reynolds numbers with either very low speeds, very high viscosity or very small length scales, [31]. As such, these flows are sometimes also referred to as creeping flows.

6.1 Problem Definition

From a mathematical point of view the Stokes equations are only a small step up from the Poisson equations. This makes building on the formulation for the Poisson equations using the mimetic spectral element method a logical next step in the direction of fluid flows.

6.1.1 Mathematical Definition

The Stokes equation can be derived from the Cauchy momentum equation, [2] the full equation,

$$\rho \frac{Du_i}{Dt} = \frac{\partial T_{ij}}{\partial x_j} + \rho f_i, \tag{82}$$

is an equation that fully describes the momentum balance of a continuous medium such as water or air. Here the material derivative is the combination of the time derivative and the convective term. T_{ij} is the Cauchy stress tensor of the medium. For a Newtonian fluid the deviatoric stress tensor in two dimensions is given by, [3].

$$\sigma_{xx} = \lambda(\nabla \cdot \boldsymbol{u}) + 2\mu \frac{\partial u}{\partial x}, \tag{83}$$

$$\sigma_{xy} = \sigma_{yx} = \mu \frac{\partial v}{\partial x} + \nu \frac{\partial u}{\partial y}, \tag{84}$$

$$\sigma_{yy} = \lambda(\nabla \cdot \boldsymbol{u}) + 2\mu \frac{\partial v}{\partial y}. \tag{85}$$

$$\tag{86}$$

Where λ is the bulk viscosity and μ the dynamic viscosity. Combining this with the action of the pressure term the Cauchy stress tensor can be written as.

$$T_{ij} = \sigma_{ij} - \delta_{ij} p. \tag{87}$$

And from these definitions for the stresses the Navier-Stokes equations directly follow. To obtain the Stokes equations a further set of assumptions must be made. Starting at Equation 82 the main assumptions are a steady flow and assuming that the viscous forces dominate the problem to the extend that the convective term in the equation can be neglected, essentially this sets the entire left hand side of Equation 82 to zero. The resulting equation is.

$$0 = \frac{\partial T_{ij}}{\partial x_j} + \rho f_i. \tag{88}$$

Next we make the assumption that the fluid is incompressible.

$$\nabla \cdot \boldsymbol{u} = 0. \tag{89}$$

This allows us to take the stress equations from Equation 83 to Equation 85, noting that the bulk viscosity term disappear from 2 of the equations under the incompressibility constraint and substituting them into Equation 88 the following equation emerges.

$$\mu \Delta \boldsymbol{u} - \nabla p + \rho f_i = 0. \tag{90}$$

This equation, divided by the density will then combine with the incompressibility constraint to create the full system.

$$\nu \Delta \boldsymbol{u} - \frac{\nabla p}{\rho} + f_i = 0, \tag{91}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{92}$$

Usually for numerical experiments ρ and ν are chosen to be 1 such that the equations reduces even further to.

$$\Delta \boldsymbol{u} - \nabla p + f_i = 0, \tag{93}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{94}$$

In this set of equations we see the appearance of the Laplacian of $\boldsymbol{u}$. From vector calculus it is known that.

$$\Delta \boldsymbol{A} = \nabla(\nabla \cdot \boldsymbol{A}) - \nabla \times (\nabla \times \boldsymbol{A}). \tag{95}$$

Together with the incompressibility constraint the first term of this equation can be dropped and the system can be reduced to.

$$-\nabla \times (\nabla \times \boldsymbol{u}) - \nabla p + f_i = 0, \tag{96}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{97}$$

Introducing the explicit vorticity as the curl of the velocity,

$$\nabla \times \boldsymbol{u} = \omega. \tag{98}$$

a new formulation arises as the following.

$$\omega - \nabla \times \boldsymbol{u} = 0, \tag{99}$$

$$\nabla \times \omega + \nabla p = f_i, \tag{100}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{101}$$

This formulation is sometimes called the VVP formulation [32] of the Stokes equations due to the degrees of freedom being the vorticity, velocity and pressure, respectively. To create a finite element method that the mimetic spectral element method uses we multiply

the system above with an appropriate set of test functions to gain the weak formulation of the Stokes equations.

$$(\tilde{\omega}, \omega)_\Omega - (\tilde{\omega}, \nabla \times \boldsymbol{u})_\Omega = 0, \tag{102}$$
$$(\tilde{q}, \nabla \times \omega)_\Omega + (\tilde{q}, \nabla p)_\Omega = (\tilde{q}, f_i), \tag{103}$$
$$(\tilde{p}, \nabla \cdot \boldsymbol{u})_\Omega = 0. \tag{104}$$

This can then be written in the language of differential forms, [32].

$$(\tilde{\omega}, \omega)_\Omega - (\tilde{\omega}, d^\star q)_\Omega = 0, \tag{105}$$
$$(\tilde{q}, d\omega)_\Omega + (\tilde{q}, d^\star p)_\Omega = (\tilde{q}, f_i), \tag{106}$$
$$(\tilde{p}, dq)_\Omega = 0. \tag{107}$$

Where the flux q is the dual to the velocity u. To create a formulation that has easy implementation for boundary conditions the following identities are used [32].

$$(\tilde{\omega}, d^\star q)_\Omega = (d\tilde{\omega}, q)_\Omega - \int_{\partial\Omega} \mathrm{tr}(\tilde{\omega}) \wedge \mathrm{tr}(\star q), \tag{108}$$
$$(\tilde{q}, d^\star p)_\Omega = (d\tilde{q}, p)_\Omega - \int_{\partial\Omega} \mathrm{tr}(\tilde{q}) \wedge \mathrm{tr}(\star p). \tag{109}$$

This transforms the differential form system to.

$$(\tilde{\omega}, \omega)_\Omega - (d\tilde{\omega}, q)_\Omega = \int_{\partial\Omega} \mathrm{tr}(\tilde{\omega}) \wedge \mathrm{tr}(\star q), \tag{110}$$
$$(\tilde{q}, d\omega)_\Omega + (d\tilde{q}, p)_\Omega = (\tilde{q}, f_i) + \int_{\partial\Omega} \mathrm{tr}(\tilde{q}) \wedge \mathrm{tr}(\star p), \tag{111}$$
$$(\tilde{p}, dq)_\Omega = 0. \tag{112}$$

6.1.2 Numerical Representation in 2 Dimensions

The final differential form system can be discretized using the mimetic spectral element method. First the right spaces must be chosen for the forms. The flux q must still be chosen as a 1-form due to the fact that it is a vector quantity. It is however not obvious in what space the vorticity or pressure should be chosen considering that they are both scalars and scalars can be chosen to be in either a 0- of 2-form. The formulation itself however suggests that from the term,

$$(d\tilde{\omega}, q)_\Omega. \tag{113}$$

is only a viable term if the result from $d\tilde{\omega}$ is also in a 1-form. recalling the property that the exterior derivative elevates an n-form to an $(n+1)$-form, ω should be a 0-form. Using a similar argument for the term,

$$(d\tilde{q}, p)_\Omega. \tag{114}$$

suggests that p is a 2-form. If all the forms are located on the primal grid of the double DeRham complex the numerical system to solve is, [32].

$$\begin{bmatrix} \mathbb{M}^{(0)} & (\mathbb{E}^{(1,0)})^T\mathbb{M}^{(1)} & 0 \\ \mathbb{M}^{(1)}\mathbb{E}^{(1,0)} & 0 & (\mathbb{E}^{(2,1)})^T\mathbb{M}^{(2)} \\ 0 & \mathbb{M}^{(2)}\mathbb{E}^{(2,1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{q} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{b} \\ \mathbb{M}^{(1)}\boldsymbol{f} + \boldsymbol{g} \\ 0 \end{bmatrix}. \tag{115}$$

There is however a modification to this scheme that can be made to make the system sparser, this is both from a storage and computational point of view desirable. Especially the $\mathbb{M}^{(2)}$ matrix is a completely full matrix. Luckily it can be removed by choosing the pressure to be in a dual 0-form, which is the dual to its natural primal 2-form. The new resulting numerical system is, [32].

$$\begin{bmatrix} \mathbb{M}^{(0)} & (\mathbb{E}^{(1,0)})^T\mathbb{M}^{(1)} & 0 \\ \mathbb{M}^{(1)}\mathbb{E}^{(1,0)} & 0 & (\mathbb{E}^{(2,1)})^T \\ 0 & \mathbb{E}^{(2,1)} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{q} \\ \boldsymbol{p'} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{b} \\ \mathbb{M}^{(1)}\boldsymbol{f} + \boldsymbol{g} \\ 0 \end{bmatrix}. \tag{116}$$

Where p' is now the new pressure as a dual 0-form instead of a 2-form.The expansions for the numerical representation of the Stokes problem are thus very similar to the previous Poisson problem with the inclusion of the extra vorticity field.

$$\omega = \sum_{i=0}^{(N+1)^2} \omega_i h(x)h(y), \tag{117}$$

$$u = \sum_{i=0}^{N(N+1)} q_i h(x)e(y), \tag{118}$$

$$v = \sum_{i=N(N+1)}^{2N(N+1)} q_i e(x)h(y), \tag{119}$$

$$p' = \sum_{i=0}^{N^2} p_i h'(x)h'(y). \tag{120}$$

$$\tag{121}$$

The boundary terms in the formulation of course need to be calculated the top most term $\boldsymbol{b}$ is equivalent to the term,

$$\int_{\partial\Omega} tr(\tilde{\omega}) \wedge tr(\star\boldsymbol{q}) = \int_{\partial\Omega} tr(\tilde{\omega}) \wedge tr(u), \tag{122}$$

and represents the tangential velocity boundary condition. Expanding the right hand side then gives.

$$= -\sum_{i=1}^{n+1}\sum_{j=1}^{n+1} \tilde{\omega}_{ij}\ h_i(x)h_j(y)\ u^{ex}(x,y)\mathrm{d}y - \sum_{i=1}^{n+1}\sum_{j=1}^{n+1} \tilde{\omega}_{ij}\ h_i(x)h_j(y)\ v^{ex}(x,y)\mathrm{d}x \tag{123}$$

Which on the left boundary for instance would result in,

$$\int_{\partial\Omega_L} tr(\tilde{\omega}) \wedge tr(u) \approx -\sum_{k=1}^{n+1}\sum_{i=1}^{n+1}\sum_{j=1}^{n+1} \tilde{\omega}_{ij} h_j(x_k) w_k v^{external}(-1, y_k). \tag{124}$$

The pressure boundary term,

$$g = \int_{\partial\Omega} tr(\tilde{q}) \wedge tr(p), \tag{125}$$

is handled similarly and can be written as.

$$\approx \sum_{k=1}^{n+1}\sum_{i=1}^{n} (\tilde{q_x})_{1i} e(y_k) p^{external}(y_k) w_k. \tag{126}$$

Finally the normal velocity boundary conditions are dealt with separately. With the degrees of freedom q discretized on the edges of the element representing the flux through those edges the normal boundary conditions can simply be calculated as the integral over the line edge piece. To set this equation in the matrix either the associated momentum equation (second line in Equation 116) can be replaced with zeros and a 1 at the degree of freedom associated with the edge and an appropriate right hand side with result of the aforementioned integral. Another way is by creating a Lagrange multiplier for the degree of freedom. The integral on the right hand side for a boundary on the left will then be.

$$g_i = \sum_{k=1}^{n+1} g^{external}(-1, x_k) w_k. \tag{127}$$

6.1.3 Multiple Element Chaining

Multiple elements can be chained using a technique similar to what has been described in the previous section for the Poisson equation. For the VVP formulation however the connection is not only on the fluxes q, but also on the vorticity. The resulting system is.

$$\begin{bmatrix} \mathbb{M}^{(0)} & (\mathbb{E}^{(1,0)})^T \mathbb{M}^{(1)} & 0 & C_\omega^t & 0 \\ \mathbb{M}^{(1)}\mathbb{E}^{(1,0)} & 0 & (\mathbb{E}^{(2,1)})^T & 0 & C_q^t \\ 0 & \mathbb{E}^{(2,1)} & 0 & 0 & 0 \\ C_\omega & 0 & 0 & 0 & 0 \\ 0 & C_q & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{q} \\ \boldsymbol{p'} \\ \lambda_\omega \\ \lambda_q \end{bmatrix} = \begin{bmatrix} -\boldsymbol{b} \\ \mathbb{M}^{(1)}\boldsymbol{f} + \boldsymbol{g} \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{128}$$

Due to the vorticity being nodally discretized the corners of this formulation become a problem. If the four blocks that surround a corner are connected naively there will be 4 connections on the corner. One connection between both top blocks, one connection between the bottom blocks, and the same for the left and right blocks. The problem here is that to define this connection only three of these connections are required with the fourth allowing this problem to have multiple non-physical solutions. One way to solve this problem would be to only create three of the connections. If one however uses an iterative solver this problem is inconsequential. Even though the problem allows multiple solutions, the idea behind an iterative solver is to minimize the residual $Ax - b$, which will still be small because the extra equation does not add an error to this residual.

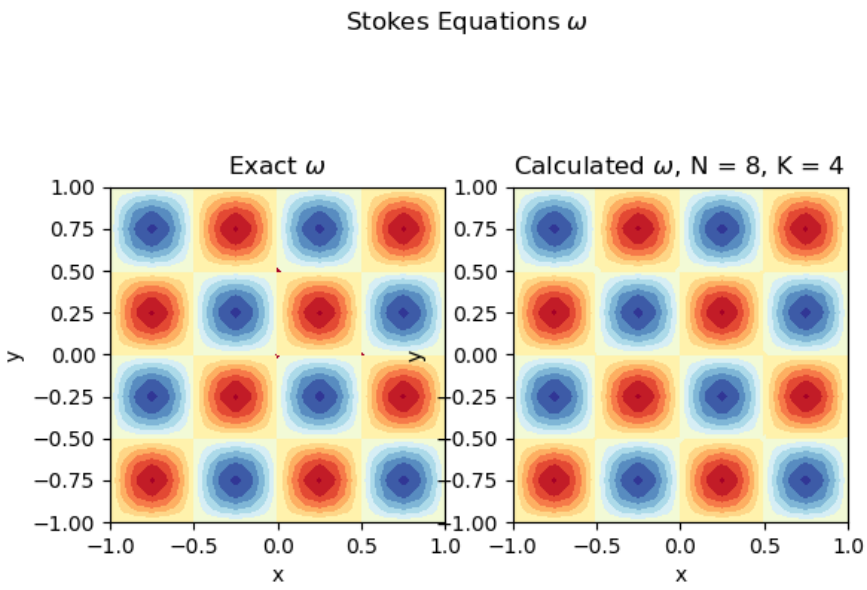

Figure 25: Visualisation of ω with the exact solution on the left and the calculated solution on the right

6.2 Results

6.2.1 Convergence

To verify that this code is working the code was tested against a manufactured solution, [38]. The exact solution is used is,

$$\omega = -4\pi sin(2\pi x)sin(2\pi y), \tag{129}$$
$$u = -sin(2\pi x)cos(2\pi y), \tag{130}$$
$$v = cos(2\pi x)sin(2\pi y), \tag{131}$$
$$p = sin(\pi x)sin(\pi y), \tag{132}$$
$$f_x = \pi cos(\pi x)sin(\pi y) - 8\pi^2 sin(2\pi x)cos(\pi y), \tag{133}$$
$$f_y = \pi sin(\pi x)cos(\pi y) + 8\pi^2 cos(2\pi x)sin(\pi y). \tag{134}$$

which has also previously been used by, [18, 32]. The visualisation of this and an example solution using the VVP formulation can be seen in Figure 25 to Figure 28

Using again the same p and h refinement as in the case of the Poisson formulation the results are presented in Figure 29 and Figure 30.

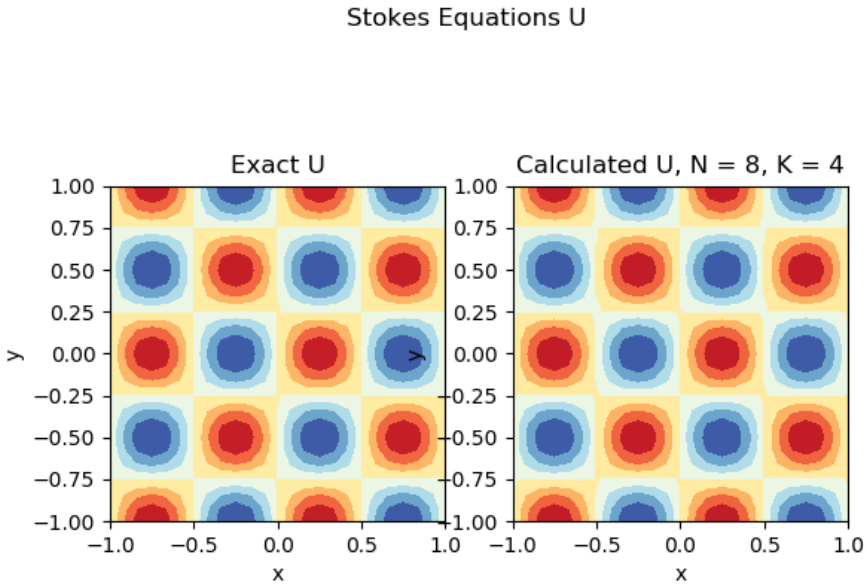

Figure 26: Visualisation of velocity in x-direction with the exact solution on the left and the calculated solution on the right

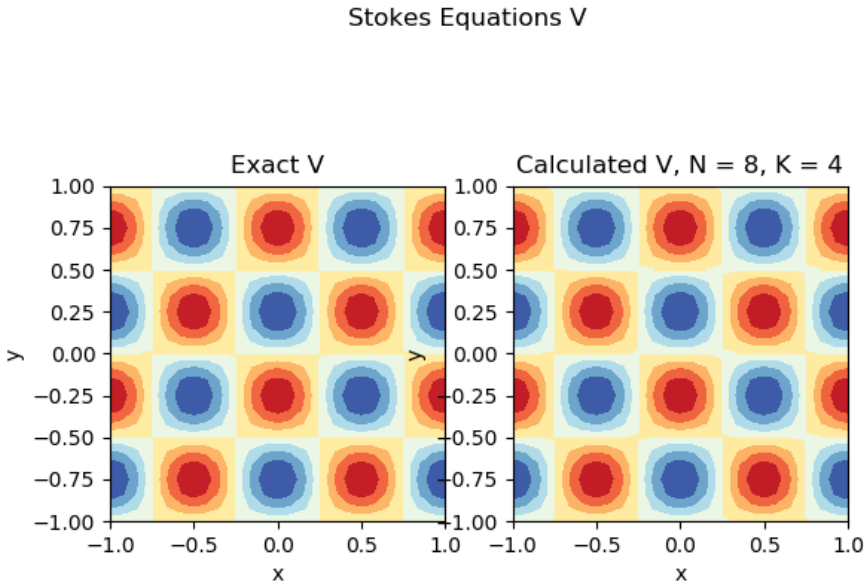

Figure 27: Visualisation of velocity in y-direction with the exact solution on the left and the calculated solution on the right

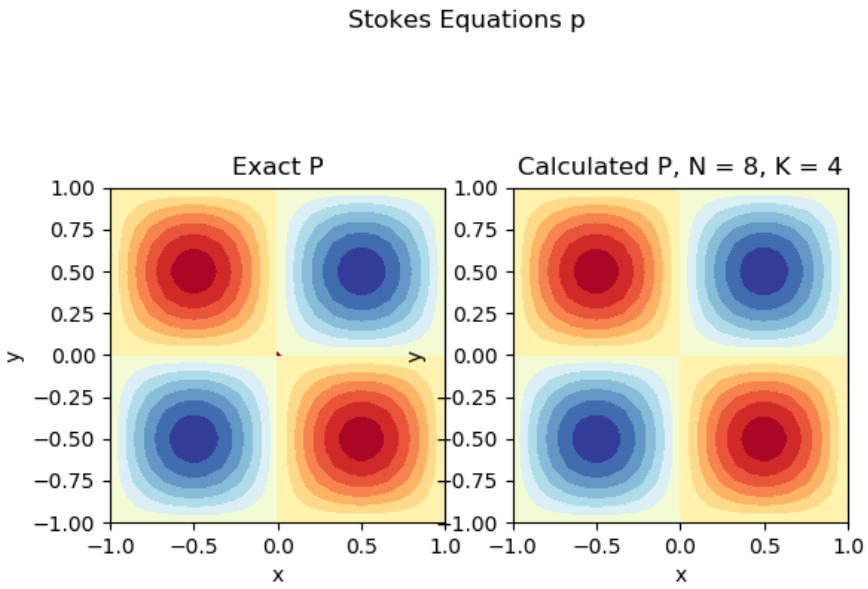

Figure 28: Visualisation of the pressure with the exact solution on the left and the calculated solution on the right

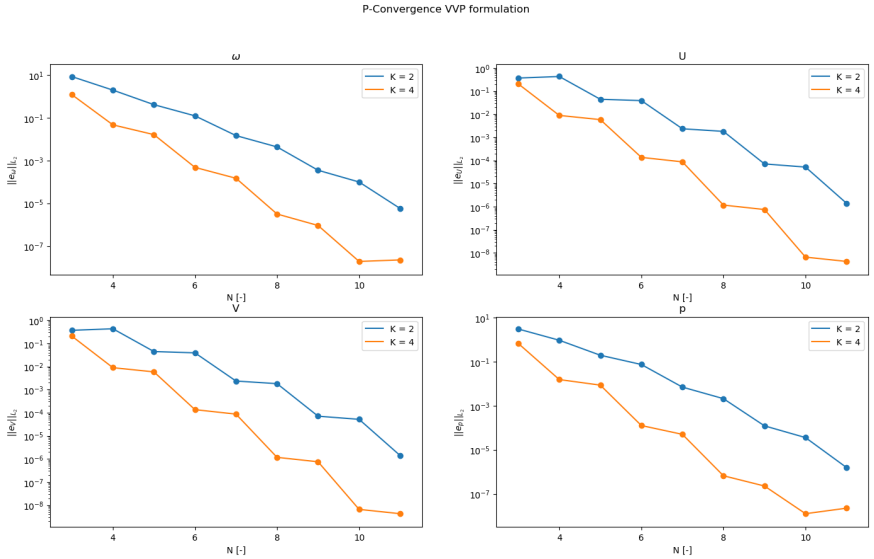

Figure 29: p-Convergence of the VVP formulation using the L_2 norm

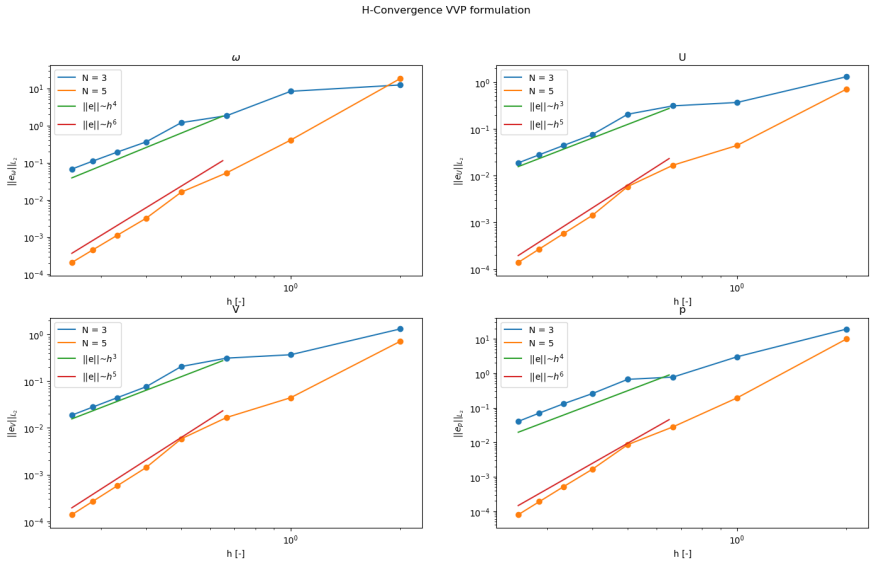

Figure 30: h-Convergence of the VVP formulation using the L_2 norm, note the slopes of the convergence that is expected for the order of the method.

6.2.2 Poiseuille Flow

For the very basic planar Poiseuille flow only a single element is required, here the domain is chosen to be $[-1, 1]^2$ with the viscosity and density set to 1. With the solution to the planar Poiseuille flow being a parabolic profile for the velocity in the x-direction and a linear profile for both the pressure and vorticity the problem should be completely described by a single element of order 2. The element used has an order of 20 however. This shows that the method is stable in the sense that when given greater representation power than is required it does not produce spurious solutions. In Figure 31 the results for the flow are presented. Note that the vertical velocity v appears to contain some spurious results, this is however only due to v being exactly 0 in the exact solution and the result shown is of course a numerical result with an accuracy up to machine precision.

6.2.3 Lid Driven Cavity Flow

With the Poiseuille flow verifying that the VVP formulation works for physical flow problems the next step is to test the formulation with the lid driven cavity flow. The results are shown in Figure 32. These results compare favourably with reference solutions like, [40] where a central difference approach is used on the bi-harmonic formulation of the Stokes equation. The calculated center of the vortex using the present method is at a y-level 0.470 below the top edge, exactly the same as the reference result 0.470.

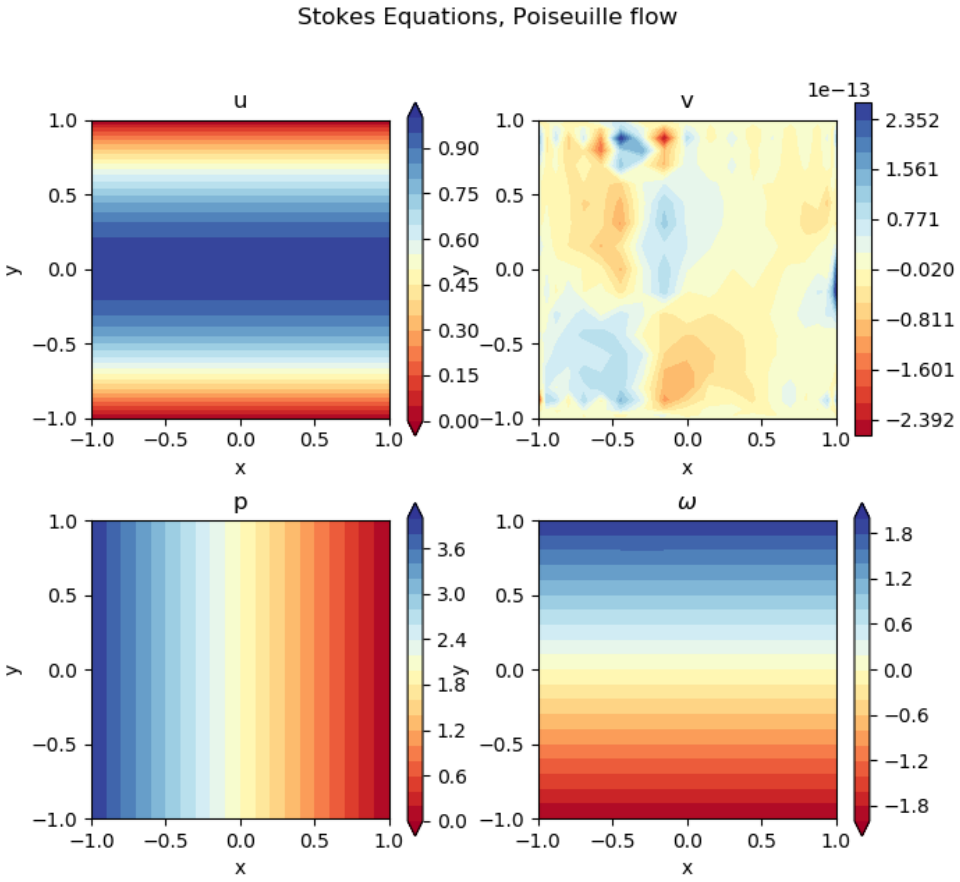

Figure 31: Solution to the planar Poiseuille flow problem using the VVP formulation. Single element, $N = 20$

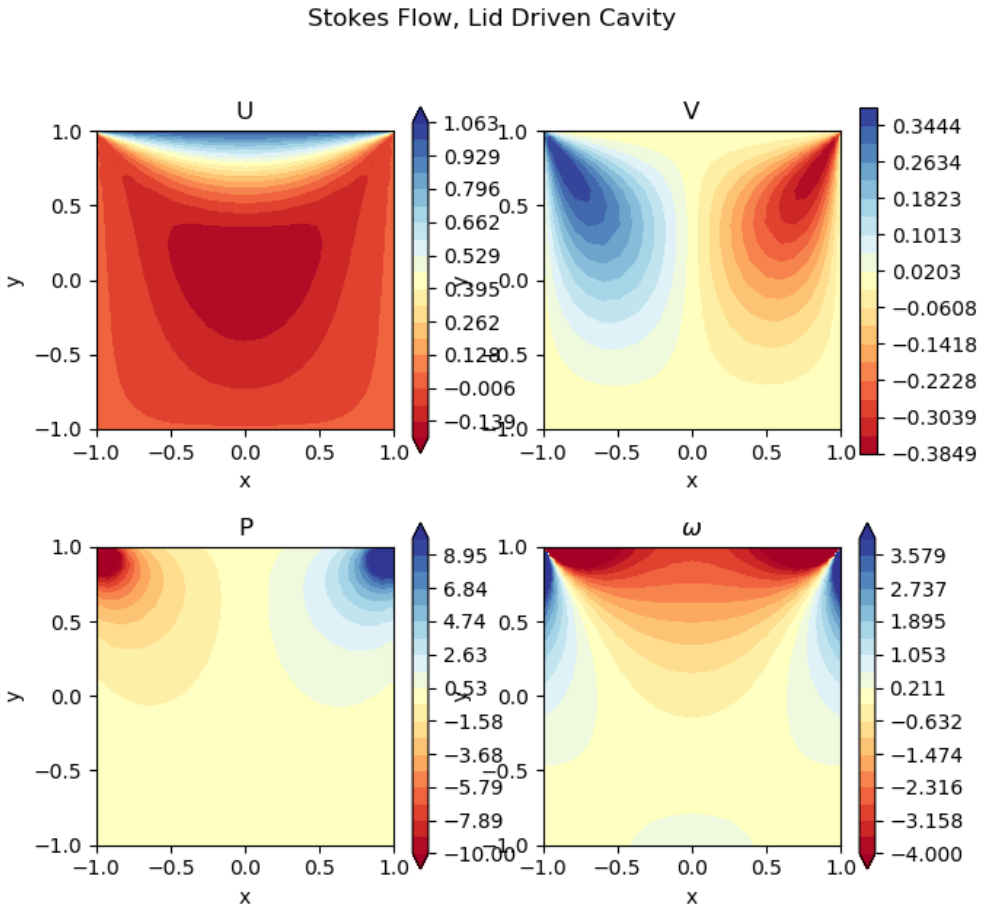

Figure 32: Solution to the lid driven cavity problem using the VVP formulation. Single element, $N = 24$

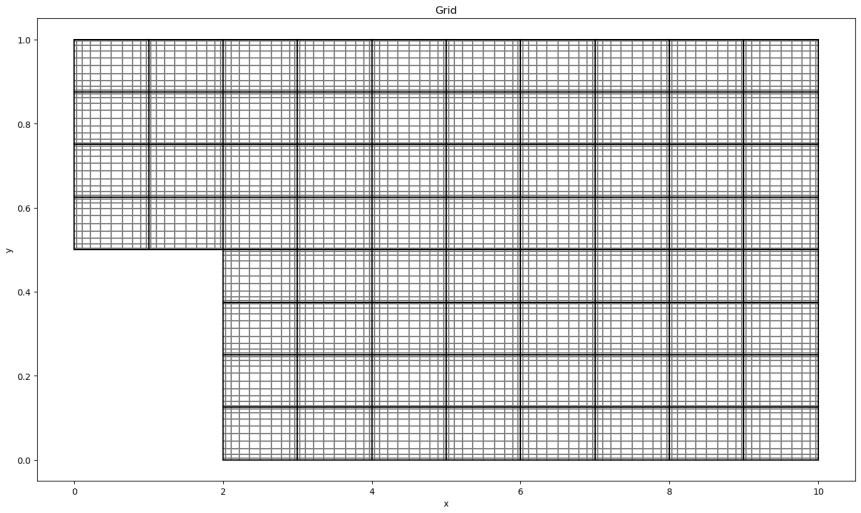

Figure 33: The grid used in the calculation of flow over a backwards facing step using the VVP formulation. $N = 10$, 72 Elements

6.2.4 Backwards Facing Step

The grid used for the backwards facing step can be found in Figure 33 and the solution to the backwards facing step given in Figure 34 compares very well with the results from [18]. The inflow boundary condition on the left boundary is.

$$u(y) = -16(y - \frac{1}{2})(y - 1). \tag{135}$$

Which gives a normalized maximum value of the inflow velocity of 1.

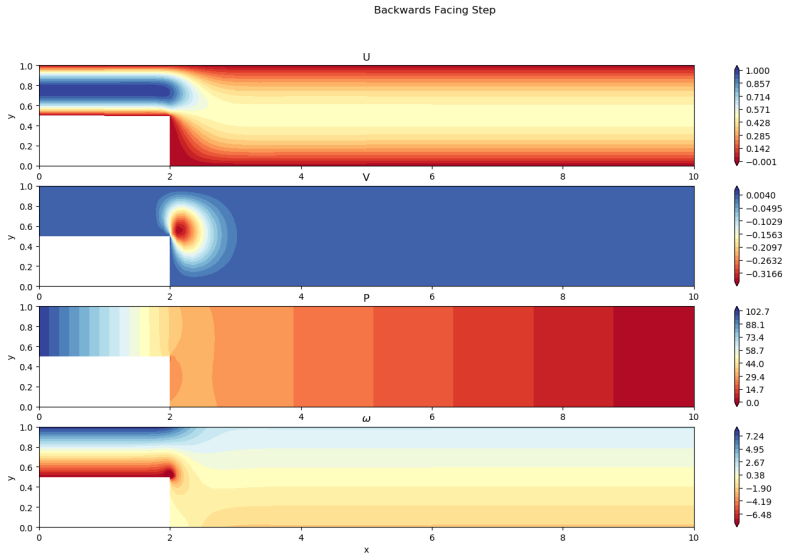

Figure 34: The solution of flow over a backwards facing step using the VVP formulation. $N = 10$, 72 Elements

7 A new Formulation

To improve upon the VVP formulation a new formulation has been developed, [18]. This formulation strongly satisfies the conservation of angular momentum.

7.1 Problem definition

At a continuous level, if the solution to the Stokes equations satisfies the linear momentum equations and has a symmetric stress tensor the solution also satisfies the conservation of angular momentum. To show this, first consider the angular momentum, [18].

$$\nabla \cdot (\boldsymbol{r} \times \underline{\underline{\sigma}}) + \boldsymbol{r} \times f = 0. \tag{136}$$

expanding this equation will lead to.

$$0 = \nabla \cdot \left(\begin{bmatrix} x \\ y \end{bmatrix} \times \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{bmatrix} \right) + \begin{bmatrix} x \\ y \end{bmatrix} \times \begin{bmatrix} f_x \\ f_y \end{bmatrix} \tag{137}$$

$$= \nabla \cdot \begin{bmatrix} x\sigma_{yx} & -y\sigma_{xx} \\ x\sigma_{yy} & -y\sigma_{xy} \end{bmatrix} + xf_y - yf_x \tag{138}$$

$$= \frac{\partial}{\partial x}(x\sigma_{yx} - y\sigma_{xx}) + \frac{\partial}{\partial y}(x\sigma_{yy} - y\sigma_{xy}) + xf_y - yf_x \tag{139}$$

$$= \sigma_{yx} + x\frac{\partial \sigma_{yx}}{\partial x} - y\frac{\partial \sigma_{xx}}{\partial x} + x\frac{\partial \sigma_{yy}}{\partial y} - \sigma_{xy} - y\frac{\partial \sigma_{xy}}{\partial y} + xf_y - yf_x \tag{140}$$

$$= \sigma_{yx} - \sigma_{xy} + \boldsymbol{r} \times (\nabla \cdot \underline{\underline{\sigma}} + \boldsymbol{f}). \tag{141}$$

With the last term in parentheses in the last line of the above equations being the conservation of linear momentum and thus being 0, only the term $\sigma_{yx} - \sigma_{xy}$ remains. Thus with a symmetric stress tensor a solution to the linear momentum equation also satisfies the conservation of angular momentum. Using this information it becomes possible to create a formulation for the Stokes equations that strongly satisfies the conservation of angular momentum.

7.1.1 Mathematical Definition

By explicitly including the stresses in the formulation of the problem it becomes possible to directly use these stresses in order to achieve the desired properties of the method. For this formulation, it is important to start with the Lagrangian functional.

$$\mathcal{L}(\underline{\underline{\sigma}}, \boldsymbol{u}, \omega, p, \boldsymbol{f}, \boldsymbol{u}_p) = \int_\Omega \frac{1}{2}\underline{\underline{\sigma}}^T \frac{1}{2\nu}\underline{\underline{\sigma}} \, \mathrm{d}\Omega + \int_\Omega \boldsymbol{u}(\nabla \cdot \underline{\underline{\sigma}} + \boldsymbol{f}) \, \mathrm{d}\Omega + \int_\Omega \omega(\sigma_{xy} - \sigma_{yx}) \, \mathrm{d}\Omega + \int_\Omega p(\nabla \cdot \boldsymbol{u}) \, \mathrm{d}\Omega - \int_{\partial\Omega} \underline{\underline{\sigma}}(\boldsymbol{u}_p \cdot \boldsymbol{n}) \, \mathrm{d}\Gamma. \tag{142}$$

Which is where we can see the most important physics in this formulation and how the resulting numerical method will impose the constraints on the solution. At the heart is the minimisation of the energy created by the stresses denoted by the first term. Secondly there are 3 Lagrange multipliers prescribed. The velocity is the Lagrange multiplier that

enforces the conservation of linear momentum $\nabla \cdot \underline{\underline{\sigma}} + \boldsymbol{f}$ explicitly. The pressure is the Lagrange multiplier assigned to restrict the solution of the velocity to be divergence free and the vorticity is the Lagrange multiplier that enforces the stress tensor to be symmetric. Rewriting this formulation slightly brings it to the very useful form.

$$\begin{aligned}\mathcal{L}(\underline{\underline{\sigma}}, \boldsymbol{u}, \omega, p, \boldsymbol{f}, \boldsymbol{u}_p, (\sigma_{xy})_P, (\sigma_{yx})_P) = \int_\Omega \frac{1}{2}\underline{\underline{\sigma}}^T \frac{1}{2\nu}\underline{\underline{\sigma}} \, \mathrm{d}\Omega + \int_\Omega u(\frac{\partial \sigma_{xx}}{\partial x} + f_x) \, \mathrm{d}\Omega \\ - \int_\Omega \frac{\partial u}{\partial y}\sigma_{xy} \, \mathrm{d}\Omega + \int_\Omega v(\frac{\partial \sigma_{yy}}{\partial y} + f_y) \, \mathrm{d}\Omega - \int_\Omega \frac{\partial v}{\partial x}\sigma_{yx} \, \mathrm{d}\Omega + \int_\Omega \omega(\sigma_{yx} - \sigma_{xy}) \, \mathrm{d}\Omega \\ - \int_\Omega \nabla p \cdot \boldsymbol{u} \, \mathrm{d}\Omega + \int_{\partial\Omega} p, \boldsymbol{u}_p \cdot \boldsymbol{n} \mathrm{d}\Gamma + \\ \int_\Omega -u_p \sigma_{xx} n_x + u(\sigma_{xy})_p n_y + v(\sigma_{yx})_P n_x - v_p \sigma_{yy} n_y \, \mathrm{d}\Omega. \quad (143)\end{aligned}$$

This form, when taking variation will create the system.

$$\left(\tilde{\underline{\underline{\sigma}}}, \frac{1}{2\nu}\underline{\underline{\sigma}}\right)_\Omega + \int_\Omega u\frac{\partial \tilde{\sigma}_{xx}}{\partial x} - \frac{\partial u}{\partial y}\tilde{\sigma}_{xy} + v\frac{\partial \tilde{\sigma}_{yy}}{\partial y} - \frac{\partial v}{\partial x}\tilde{\sigma}_{yx} \, \mathrm{d}\Omega + (\omega, (\tilde{\sigma}_{yx} - \tilde{\sigma}_{xy}))_\Omega = B_\sigma, \quad (144)$$

$$\int_\Omega \tilde{u}\frac{\partial \sigma_{xx}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\sigma_{xy} + \tilde{v}\frac{\partial \sigma_{yy}}{\partial y} - \frac{\partial \tilde{v}}{\partial x}\sigma_{yx} \, \mathrm{d}\Omega - (\tilde{\boldsymbol{u}}, \nabla p)_\Omega = -(\tilde{\boldsymbol{u}}, \boldsymbol{f})_\Omega - B_u, \quad (145)$$

$$(\tilde{\omega}(\sigma_{yx} - \sigma_{xy}))_\Omega = 0, \quad (146)$$

$$(\nabla \tilde{p}, \boldsymbol{u})_\Omega = B_p. \quad (147)$$

The boundary terms are:

$$B_\sigma = \int_{\partial\Omega} u_p \tilde{\sigma}_{xx} n_x + v_p \tilde{\sigma}_{yy} n_y \mathrm{d}\Gamma, \quad (148)$$

$$B_u = \int_{\partial\Omega} \tilde{u}(\sigma_{xy})_p n_y + \tilde{v}(\sigma_{yx})_p n_x \mathrm{d}\Gamma, \quad (149)$$

$$B_p = \int_{\partial\Omega} \tilde{p}, \boldsymbol{u}_p \mathrm{d}\Gamma. \quad (150)$$

Using integration by parts and assuming a sufficiently smooth solution this then produces the set of partial differential equations:

$$\frac{1}{2\nu}\underline{\underline{\sigma}} = \nabla \boldsymbol{u} - \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}, \quad (151)$$

$$\nabla \cdot \underline{\underline{\sigma}} - \nabla p + \boldsymbol{f} = 0, \quad (152)$$

$$\sigma_{yx} - \sigma_{xy} = 0, \quad (153)$$

$$\nabla \cdot \boldsymbol{u} = 0. \quad (154)$$

7.1.2 Numerical Representation

The mathematical system Equation 147 can be discretized into the system

$$\begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 \\ \mathbb{E}_\sigma & 0 & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T \\ \mathbb{R} & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma} \\ \boldsymbol{u} \\ \boldsymbol{\omega} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ -\boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \end{bmatrix}. \tag{155}$$

Here the most important change with regards to the previous numerical system lies in the expansion of the basis functions. In order for the symmetry of the stress tensor to be numerically achieved, both σ_{yx} and σ_{xy} must be expressed in a basis that have the same polynomial order. The obvious choice here would be the nodal $h(x)h(y)$ basis functions to expand both shear stresses in. This however will not work, in the second line of the above matrix the divergence of the stress tensor is calculated for the conservation of linear momentum, the stress σ_{xy} will have a partial derivative in the y-direction and the result will be expanded in the basis functions $h(x)e(y)$. To complete the divergence of the stress tensor in the x-direction the partial derivative of the normal stress σ_{xx} in the x-direction needs to be expanded in the same $h(x)e(y)$ basis functions to able to sum the results. This would require σ_{xx} to be expanded in a set of basis functions in 1D to be the anti-derivative of the set of basis functions in $h(x)$. The canonical mimetic spectral element method does not have such a set of basis functions. Furthermore, All the geometric entities have already been assigned a set of basis functions in the canonical mimetic spectral element method and adding the anti-derivative of the nodal functions as a set of basis functions would require a free geometric object for them to be assigned to. The other basis functions that have the same polynomial order in both directions $e(x)e(y)$ (discretized on surfaces) can be discarded almost immediately on the basis that the derivative of an edge function is also not defined in the canonical mimetic spectral element method.

Instead, for this formulation a completely new set of basis functions was devised, the so called mixed primal-dual basis functions. This mixed basis set takes 1 set of basis function from the primal grid in one direction and take a set of basis functions from the dual grid for the other direction, and use the tensor product of the two to create a set of two dimensional basis function. The expansions for the basis functions of the stress components in this formulation then become.

$$\sigma_{xx} = \sum_{i=1}^{n+1}\sum_{j=1}^{n+1}(\sigma_{xx})_{ij}\ h_i(x)e'_j(y), \tag{156}$$

$$\sigma_{xy} = \sum_{i=1}^{n}\sum_{j=1}^{n}(\sigma_{xy})_{ij}\ e_i(x)h'_j(y), \tag{157}$$

$$\sigma_{yx} = \sum_{i=1}^{n}\sum_{j=1}^{n}(\sigma_{yx})_{ij}\ h'_i(x)e_j(y), \tag{158}$$

$$\sigma_{yy} = \sum_{i=1}^{n+1}\sum_{j=1}^{n+1}(\sigma_{yy})_{ij}\ e'_i(x)h_j(y). \tag{159}$$

remembering that $e'_i(x)$ is the dual of $h_i(x)$, $h'_i(x)$ is the dual of $e_i(x)$ and that the dual expansion has the same polynomial order as the primal expansion, the basis functions of the

shear stresses are expanded in a basis with the same polynomial order in both directions. The polynomial space is equivalent to the polynomial order of the $e(x)e(y)$ basis function set and they can thus be set to be exactly equal. These basis functions also line up for the calculation of the conservation of linear momentum. The expanded partial derivatives become.

$$\frac{\partial \sigma_{xx}}{\partial x} = \sum_{i=1}^{n}\sum_{j=1}^{n+1}[(\sigma_{xx})_{i+1,j} - (\sigma_{xx})_i, j]\ e_i(x)e'_j(y), \tag{160}$$

$$\frac{\partial \sigma_{xy}}{\partial y} = \sum_{i=1}^{n+1}\sum_{j=1}^{n}((\sigma_{xy})_{i,j} - (\sigma_{xy})_{i,j-1})\ e_i(x)e'_j(y), \tag{161}$$

$$\frac{\partial \sigma_{yx}}{\partial x} = \sum_{i=1}^{n}\sum_{j=1}^{n+1}((\sigma_{yx})_{i,j} - (\sigma_{xy})_{i-1,j})\ e'_i(x)e_j(y), \tag{162}$$

$$\frac{\partial \sigma_{yy}}{\partial y} = \sum_{i=1}^{n+1}\sum_{j=1}^{n}[(\sigma_{xx})_{i,j+1} - (\sigma_{xx})_i, j]\ e'_i(x)e_j(y). \tag{163}$$

$$\tag{164}$$

Here it must be noted that the boundary terms are used in this definition of the partial derivatives for the shear stresses. Where $i = 0$ or $i = n + 1$ for σ_{yx} and $j = 0$ or $j = 1$ for σ_{xy} the stresses used are the prescribed ones on the boundary because the dual space does not include the endpoints. As can be seen from the expansions both terms involved in the divergence of the stress tensor in the x-direction ($\frac{\partial \sigma_{xx}}{\partial x}$ and $\frac{\partial \sigma_{xy}}{\partial y}$) are expanded in the $e(x)e'(y)$ basis. Similarly, the divergence of the stress tensor in the y direction is also expanded in the $e'(x)e(y)$ basis. The above operation to move from the degrees of freedom for the stress to the expressions for the partial derivatives can be collected in the matrix $\mathbb{E}_\sigma$ that is an incidence matrix containing only the entries -1, 0, 1. Applying this incidence matrix is thus the equivalent of taking the divergence of the stress. With this expansion configuration it is therefor possible to both satisfy the conservation of linear momentum and the symmetry of the stress tensor. It must however be noted that the expansion of the shear stresses is in the correct polynomial space, but the expansion is not the same. In order to constrict both shear stresses to be equal a projection is made to the basis functions $e(x)e(y)$. The projection is as follows.

$$(\sigma_{xy} - \sigma_{yx})_{i,j} = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n}\int_\Omega h'_i(x)h'_j(y)[((\sigma_{xy})_{kl}e_k(x)h'_l(y) - \sigma_{yx})_{kl}h'_k(x)e_l(y)]\ \mathrm{d}\Omega. \tag{165}$$

For an orthogonal domain this matrix $\mathbb{R} = [\mathbb{R}_{xy}, \mathbb{R}_{yx}]$ simplifies to

$$(\mathbb{R}_{xy})_{ijkl} = \sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h'_i(x_r)h'_j(y_s)e_k(x_r)h'_l(y_s)w_r w_s, \tag{166}$$

$$(\mathbb{R}_{yx})_{ijkl} = -\sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h'_i(x_r)h'_j(y_s)h'_k(x_r)e_l(y_s)w_r w_s. \tag{167}$$

It follows from the Lagrangian that the vorticity must be dual to the expansion of this expansion for the 2 stresses because it is the Lagrange multiplier that enforces it, which results in the vorticity being expressed as.

$$\omega = \sum_{i=1}^{n}\sum_{j=1}^{n} \omega_{ij}\ h_i'(x)h_j'(y). \tag{168}$$

The matrix $\mathbb{M}_\sigma$ is the mass matrix in this system. To understand how to construct this matrix it must be noted that the mimetic spectral element method does not have an inherent way to express tensors. In order to mimic a tensor the stress is constructed as 2 separate 1-forms, each representing the stress in a Cartesian direction.

$$\sigma = \begin{bmatrix}\sigma_x \\ \sigma_y\end{bmatrix} = \begin{bmatrix}\sigma_{xx}\mathrm{d}y - \sigma_{xy}\mathrm{d}x \\ \sigma_{yx}\mathrm{d}y - \sigma_{yy}\mathrm{d}x\end{bmatrix}. \tag{169}$$

In the previous work, [18] a very thorough derivation is made for general expressions of the stress tensor. In this work only the simpler form will be considered because fluids have a diagonal compliance matrix. With that simplification, the matrix $\mathbb{M}_\sigma$ becomes a diagonal block matrix with 2 matrices equivalent to a $\mathbb{M}^{(1)}$ matrix for transformation purposes with of course the new mixed basis functions as input.

$$\begin{bmatrix}\mathbb{M}_{\sigma_x} & 0 \\ 0 & \mathbb{M}_{\sigma_y}\end{bmatrix}. \tag{170}$$

From the Lagrangian for this formulation the velocities can be seen to be the dual to the expansion of the linear momentum. From this is follows that.

$$u = \sum_{i=1}^{n}\sum_{j=1}^{n+1} u_{i,j}\ h_i'(x)h_j(y), \tag{171}$$

$$v = \sum_{i=1}^{n+1}\sum_{j=1}^{n} v_{i,j}\ h_i(x)h_j'(y). \tag{172}$$

In order to calculate the divergence the partial derivatives of u in the x-direction and v in the y-direction is necessary. The derivatives on an orthogonal mesh become

$$\frac{\partial u}{\partial x} = \sum_{i=1}^{n}\sum_{j=1}^{n+1}[u_{i,j} - u_{i-1,j}]e_i'(x)h_j(y), \tag{173}$$

$$\frac{\partial v}{\partial y} = \sum_{i=1}^{n+1}\sum_{j=1}^{n}[v_{i,j} - v_{i,j-1}]h_i(x)e_j'(y). \tag{174}$$

Again, because of the use of the dual space the endpoints $i = 0$ and $i = n + 1$ are a prescribed velocity from the boundary in the x-direction and for $j = 0$ and $j = n + 1$ in the y-direction. These terms are found in the $\boldsymbol{g}_p$ terms in the Equation 155. Just like the

incidence matrix for the divergence of the stresses the partial derivatives for the velocity can also be reduced to an application of an incidence matrix to the velocity degrees of freedom. This matrix $\mathbb{E}_{div\boldsymbol{u}}$ can be constructed from the 2 separate incidence matrices defined by the above equations. The total incidence matrix becomes.

$$\mathbb{E}_{div\boldsymbol{u}} = \begin{bmatrix} (\mathbb{E}_{div\boldsymbol{u}})_u & 0 \\ 0 & (\mathbb{E}_{div\boldsymbol{u}})_v \end{bmatrix}. \tag{175}$$

As can be seen similarly to the expansions of the shear stresses, the partial derivative expansions for the divergence are in the same polynomial space, but not in the same expansion. In order to calculate the divergence, the expansions are projected into an e'(x)e'(y) expansion, and from here are set equal. The matrix $\mathbb{P} = [\mathbb{P}_x, \mathbb{P}_y]$ can be constructed very similarly to the matrix $\mathbb{R}$ according to.

$$(\mathbb{P}_x)_{ijkl} = \sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i(x_r)h_j(y_s)e'_k(x_r)h_l(y_s)w_r w_s, \tag{176}$$

$$(\mathbb{P}_y)_{ijkl} = -\sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i(x_r)h_j(y_s)h_k(x_r)e'_l(y_s)w_r w_s. \tag{177}$$

Multiplying $\mathbb{P}$ with $\mathbb{E}_{div\boldsymbol{u}}$ then creates the operator that sums the derivative $\frac{\partial u}{\partial x}$ and $\frac{\partial v}{\partial y}$. Finally, the pressure itself is dual to the expansion of the divergence, again because it is the Lagrange multiplier that enforces the divergence, and thus it can be expressed as.

$$p = \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}h_i(x)h_j(y). \tag{178}$$

7.2 Results

The new formulation has been shown to converge and later the new formulation has been applied to the Poiseuille flow test case and the lid driven cavity test problem.

7.2.1 Convergence

In order to verify that the formulation is working correctly the same problem considered with the VVP formulation of section 6 is considered. The solution to this problem is given by the equations.

$$\omega = -2\pi sin(2\pi x)sin(2\pi y), \tag{179}$$

$$u = -sin(2\pi x)cos(2\pi y), \tag{180}$$

$$v = cos(2\pi x)sin(2\pi y), \tag{181}$$

$$p = sin(\pi x)sin(\pi y), \tag{182}$$

$$\sigma_{xx} = -2cos(\pi x)cos(2\pi y), \tag{183}$$

$$\sigma_{xy} = 0, \tag{184}$$

$$\sigma_{yx} = 0, \tag{185}$$

$$\sigma_{yy} = 2cos(\pi x)cos(2\pi y). \tag{186}$$

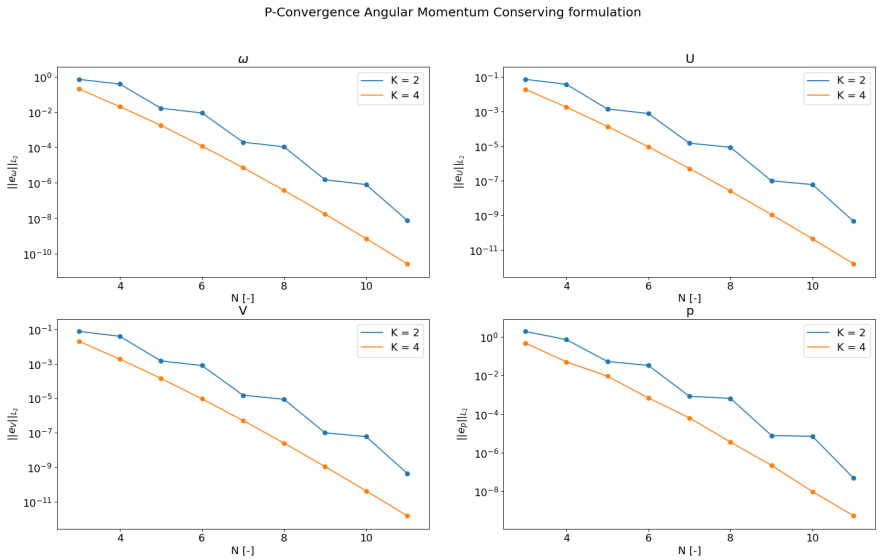

Figure 35: Velocity, pressure and vorticity p-convergence

with forcing functions

$$f_x = \pi cos(\pi x) sin(\pi x) - 4\pi^2 sin(2\pi x) cos(2\pi y), \tag{187}$$

$$f_y = \pi sin(\pi x) cos(\pi y) + 4\pi^2 cos(2\pi x) sin(2\pi y). \tag{188}$$

Again the p and h convergence is both considered. For this problem the domain is restricted to $[0, 1]^2$. In Figure 35 the p-convergence for the vorticiy(ω), U, V and pressure using K = 2(4 elements) and K = 4(16 elements). The method converges as the polynomial order N is increased. Interestingly, with the case K = 2, the method has a staircase like convergence pattern. This is likely due to the use of sines and cosines in the analytical solution. The polynomial expansions of both sines and cosines have either odd or even powered coefficients and thus when an element spans a whole or half period of a wave, adding some polynomials does not create a better approximation. In Figure 36 the convergence for the stresses can be seen to be almost equivalent to the vorticiy(ω), U, V and pressure.

Increasing the amount of elements instead of the polynomial order also converges. In Figure 37 the convergence is plotted for the vorticiy(ω), U, V and pressure with both a polynomial order of 3 and 5 for each element and increased amounts of elements, leading to a smaller step size per element. In Figure 38 the convergence rates for all the stresses are plotted. From the plots it can be seen that while the method is either third or fifth order, only the velocity converges at that rate, and all other fields converge at an order lower than the element. This indicates that the other fields are restricted in their convergence by the constraints imposed by the methods.

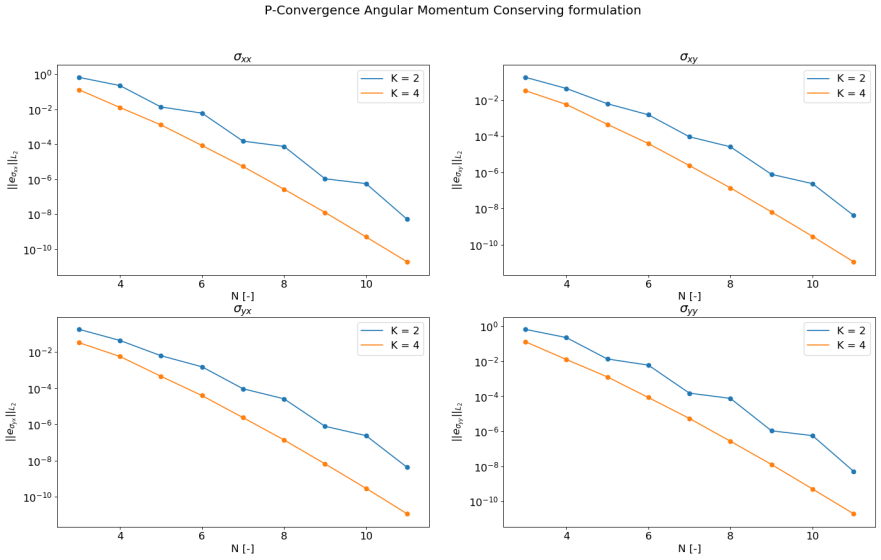

Figure 36: normal and shear stress p-convergence

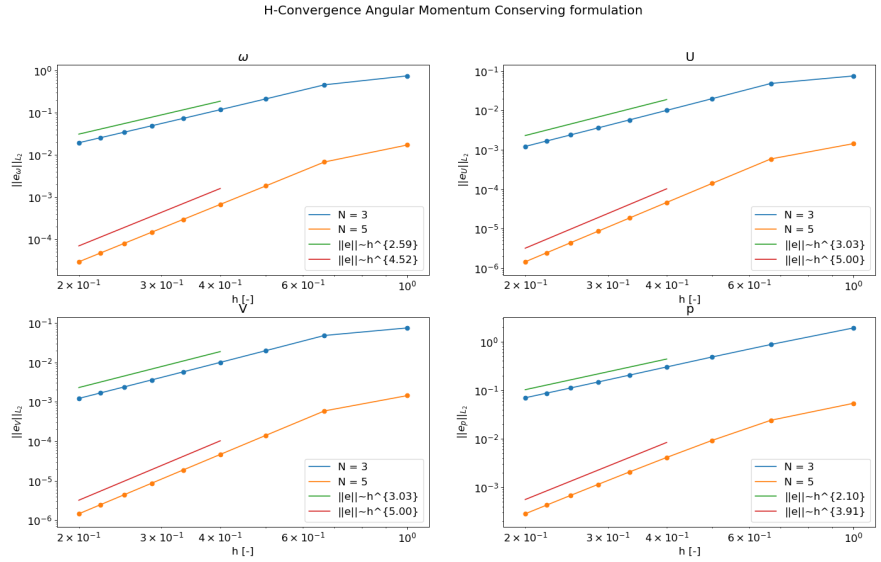

Figure 37: Velocity, pressure and vorticity h-convergence

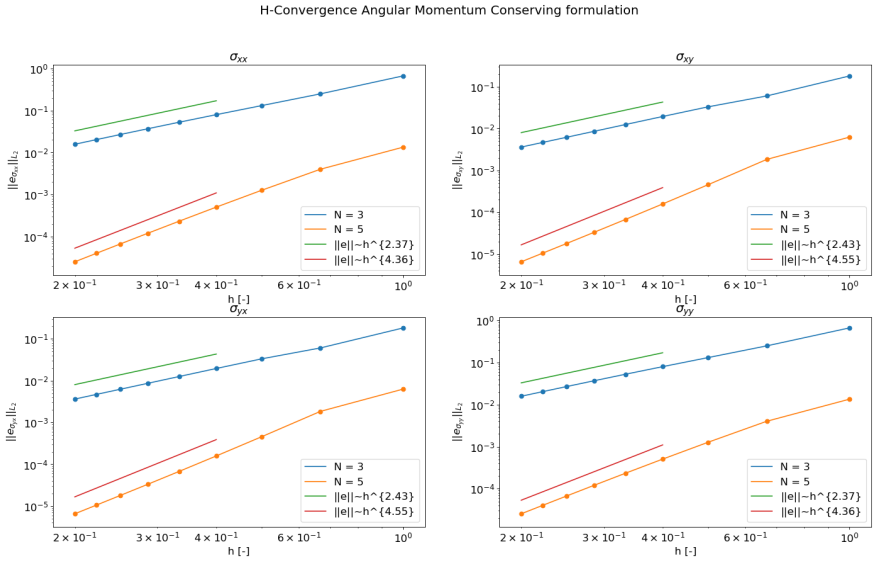

Figure 38: normal and shear stress h-convergence

7.2.2 Poiseuille Flow

The results for the steady Poiseuille flow can be found in Figure 39 and Figure 40.

The exact results for the stresses are given by

$$\sigma_{xx} = \frac{\partial u}{\partial x} = 0, \tag{189}$$

$$\sigma_{xy} = \frac{\partial v}{\partial x} - \omega = -y, \tag{190}$$

$$\sigma_{yx} = \frac{\partial u}{\partial y} + \omega = -y, \tag{191}$$

$$\sigma_{yy} = \frac{\partial v}{\partial y} = 0. \tag{192}$$

And it can be seen that the new formulation is on the exact solution within machine precision if the polynomial order for the approximation is higher than 2 and the solution is stable, the solution is calculated with a 20^{th} order element and thus the higher order representing power of the element does not create nonphysical errors in the solution.

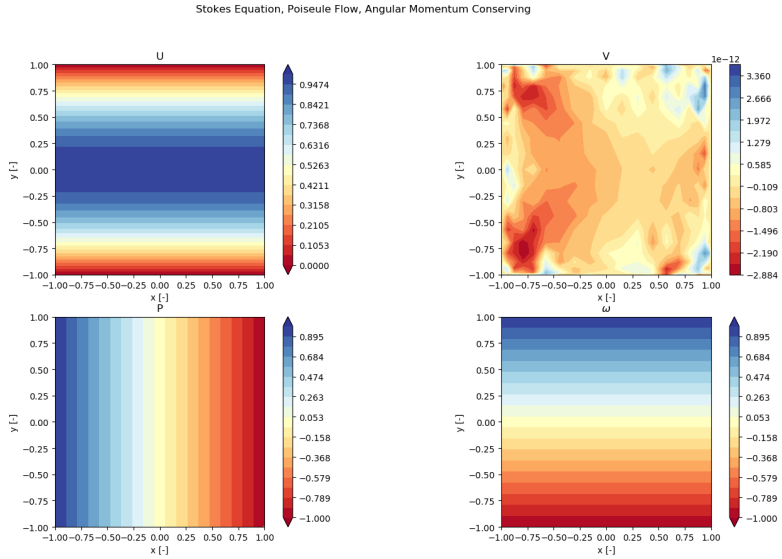

Figure 39: Velocity, pressure and vorticity results for the new formulation in Poiseuille flow, N = 20

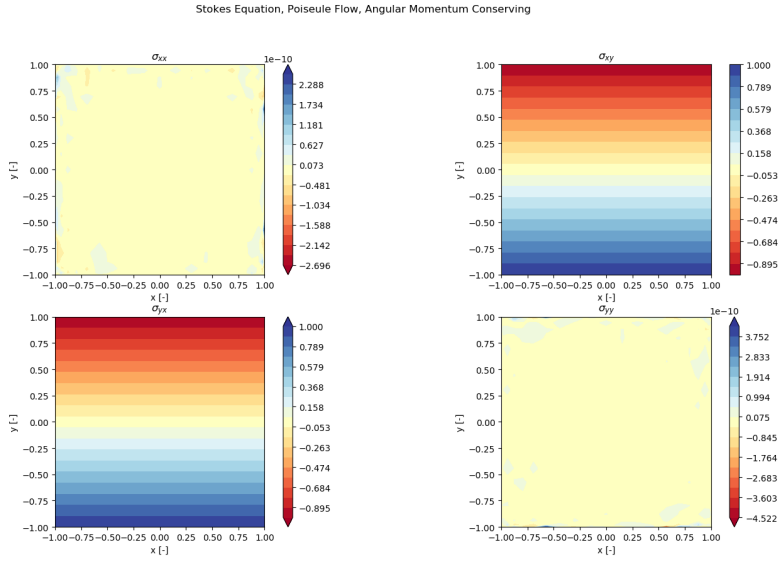

Figure 40: Stress results for the new formulation from the Poiseuille flow, $N = 20$

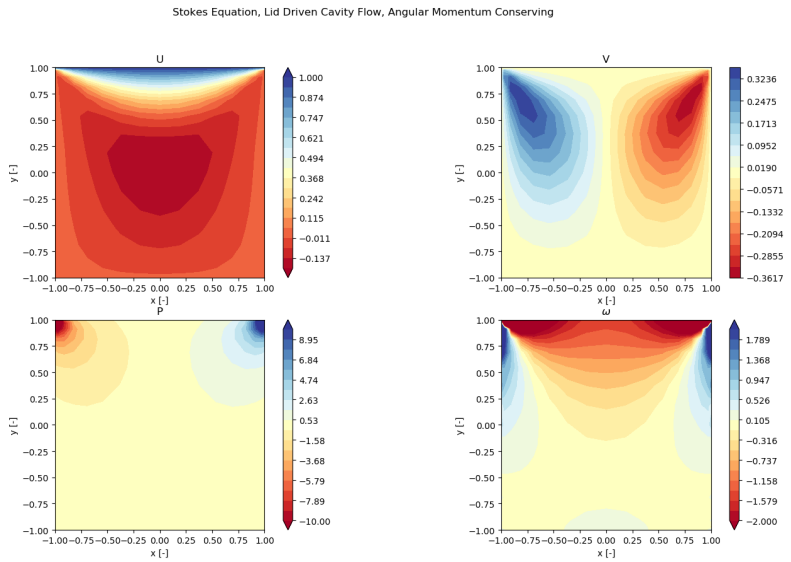

Figure 41: Velocity, Pressure and Vorticity results for the new formulation in lid driven cavity flow. $N = 16$

7.2.3 Lid Driven Cavity Flow

A more testing case is the steady lid driven cavity flow. The results for the lid driven cavity flow can be seen in Figure 41 and Figure 42

It is also interesting to look at the difference in the symmetry of the stress tensor for the old and new formulation in the lid driven cavity flow problem. In Figure 43 the difference in the shear stresses is shown, clearly indicating that conservation of angular momentum is not achieved, because this requires both conservation of linear momentum and symmetry of the stress tensor. In Figure 44 the conservation of angular momentum for the new formulation is displayed. As can be seen, the difference could not be greater between the VVP formulation and the present formulation. The present formulation conserves the angular momentum exactly down to machine precision while the error in the conservation of angular momentum can be as large as $O(10^2)$ for the VVP formulation from,[32].

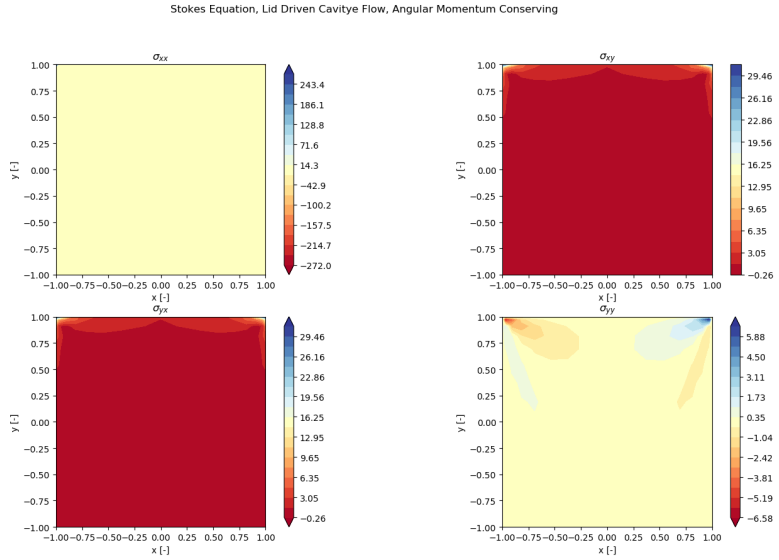

Figure 42: Stress results for the new formulation in lid driven cavity flow. $N = 16$

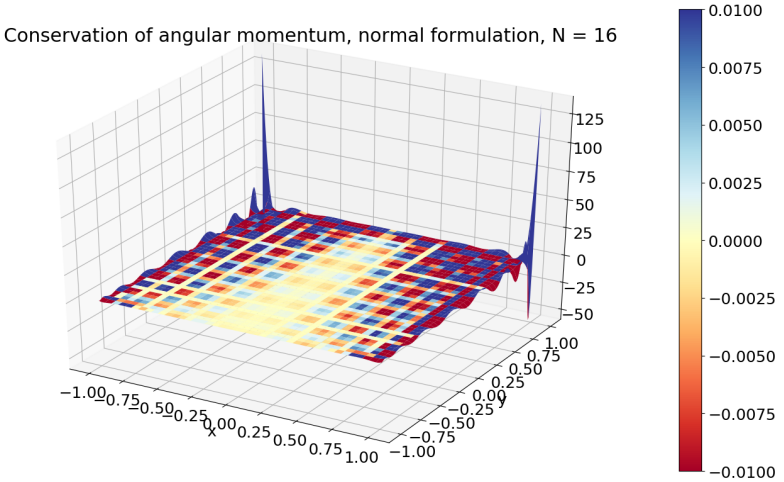

Figure 43: Error in the conservation of angular momentum in the VVP formulation

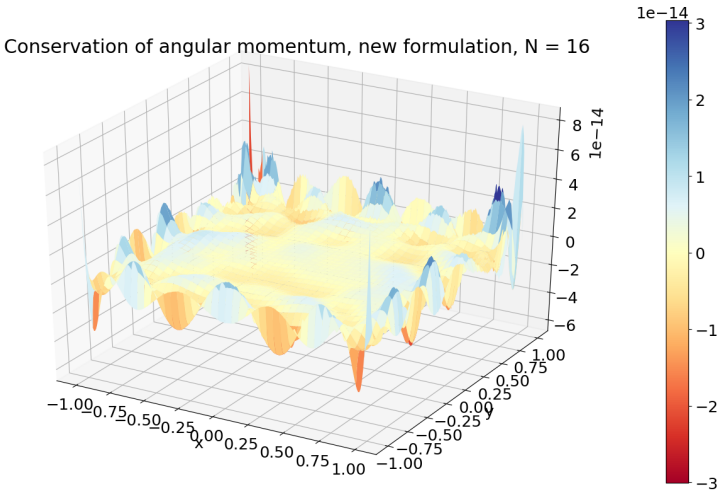

Figure 44: Symmetry of the stress tensor in new formulation

8 Transforming the new formulation

The new formulation in the previous section was derived for a square grid. In this section will be discussed how to transform this previous formulation to more general grids. It will be shown that the derivation on a square grid has had some consequences that can not be mitigated without some change to the formulation, especially with regards to the treatment of the divergence term.

8.1 New Transformations

In this section the modifications to the new formulations will be discussed in order for the formulation to work on curved meshes.

8.1.1 The Divergence

In the previous formulation the velocity is expanded in the terms

$$u = \sum_{i=1}^{n} \sum_{j=1}^{n+1} u_{i,j}\ h_i'(x) h_j(y), \tag{193}$$

$$v = \sum_{i=1}^{n+1} \sum_{j=1}^{n} v_{i,j}\ h_i(x) h_j'(y). \tag{194}$$

Which essentially discretizes the velocity components of the velocity as two separate 0-forms on the mesh. Because the 0-form is only a function on the mesh, it is not a vector or co-vector quantity, it can be transformed with only the map of the transformation. Here it is important to grasp that the mimetic spectral element method is initially defined on what is called the computational domain, which is $(\xi, \eta) \in [-1, 1]^2$ in two dimensions. From this computational domain it is then transformed to fit an element in the mesh that is overlaid on the computational domain of interest, which is known as the physical domain. The implications for a 0-form is that if one knows the map,

$$f(\xi, \eta) \to f(x, y). \tag{195}$$

from the computational domain to physical domain the inverse is also known. Given the 0-forms of the velocity it is possible to simply take the velocity and transform it back to the computational domain.

$$u(x, y) \to u(\xi, \eta), \tag{196}$$

$$v(x, y) \to v(\xi, \eta). \tag{197}$$

$$\tag{198}$$

On the computational domain we can then simply apply the exterior derivative d to these two 0-forms to get.

$$\mathrm{d}u = \frac{\partial u}{\partial \xi}\mathrm{d}\xi + \frac{\partial u}{\partial \eta}\mathrm{d}\eta, \tag{199}$$

$$\mathrm{d}v = \frac{\partial v}{\partial \xi}\mathrm{d}\xi + \frac{\partial v}{\partial \eta}\mathrm{d}\eta. \tag{200}$$

Where we see that we now have partial derivatives with respect to the computational domain, not the physical domain. To get back to the physical domain we use the transformation properties, [19] and from this we find that the derivatives of interested become.

$$\frac{\partial u}{\partial x} = \frac{\partial u}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial u}{\partial \eta}\frac{\partial \eta}{\partial x}, \tag{201}$$

$$\frac{\partial u}{\partial y} = \frac{\partial u}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial u}{\partial \eta}\frac{\partial \eta}{\partial y}, \tag{202}$$

$$\frac{\partial v}{\partial x} = \frac{\partial v}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial v}{\partial \eta}\frac{\partial \eta}{\partial x}, \tag{203}$$

$$\frac{\partial v}{\partial y} = \frac{\partial v}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial v}{\partial \eta}\frac{\partial \eta}{\partial y}. \tag{204}$$

$$\tag{205}$$

Which is the gradient of the velocity vector. This can be summarized in the matrix equation.

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial u}{\partial \eta} \\ \frac{\partial v}{\partial \xi} & \frac{\partial v}{\partial \eta} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{bmatrix}. \tag{206}$$

On the right we see the inverse Jacobian of the mapping.

$$\begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix}^{-1} = \frac{1}{|det(\mathbf{\Phi}^*)|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{bmatrix}. \tag{207}$$

And thus.

$$\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial u}{\partial \eta} \\ \frac{\partial v}{\partial \xi} & \frac{\partial v}{\partial \eta} \end{bmatrix} \cdot \frac{1}{|det(\mathbf{\Phi}^*)|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{bmatrix}. \tag{208}$$

The divergence of the velocity is nothing more than the trace of the gradient of the velocity.

$$\nabla \cdot \boldsymbol{u} = tr\left(\begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix} \right) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}. \tag{209}$$

from Equation 208 and Equation 209 it is possible to write the divergence as.

$$\nabla \cdot \boldsymbol{u} = \frac{1}{|det(\mathbf{\Phi}^*)|}\left(\frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi} - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta} + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi}\right). \tag{210}$$

This divergence is free from Christoffel symbols as is explained in Appendix A

8.1.2 Numerical Divergence

Starting from the expansions for the velocity.

$$u = \sum_{i=1}^{n} \sum_{j=1}^{n+1} u_{i,j}\ h_i'(x) h_j(h), \tag{211}$$

$$v = \sum_{i=1}^{n+1} \sum_{j=1}^{n} v_{i,j}\ h_i(x) h_j'(h). \tag{212}$$

The inverse mapping of a 0-form directly lets us define the velocity on the computational domain as

$$u = \sum_{i=1}^{n} \sum_{j=1}^{n+1} u_{i,j}\ h_i'(\xi) h_j(h), \tag{213}$$

$$v = \sum_{i=1}^{n+1} \sum_{j=1}^{n} v_{i,j}\ h_i(\xi) h_j'(h). \tag{214}$$

Taking the exterior derivative using an incidence matrix takes the form of

$$\mathrm{d}u = \sum_{i=1}^{n} \sum_{j=1}^{n+1} [u_{i,j} - u_{i-1,j}] e_i'(\xi) h_j(\eta) \mathrm{d}\xi + \sum_{i=1}^{n+1} \sum_{j=1}^{n} [u_{i,j} - u_{i,j-1}] h_i'(x) e_j(y) \mathrm{d}\eta, \tag{215}$$

$$\mathrm{d}v = \sum_{i=1}^{n+1} \sum_{j=1}^{n} [v_{i,j} - v_{i-1,j}] e_i(\xi) h_j'(\eta) \mathrm{d}\xi + \sum_{i=1}^{n} \sum_{j=1}^{n+1} [v_{i,j} - v_{i,j-1}] h_i(x) e_j'(y) \mathrm{d}\eta. \tag{216}$$

This operation is naturally able to be collected into a new incidence matrix $\mathbb{E}_{div}$ that represents the entire application of the gradient operator on the velocity Here the most important part of the new formulation is introduced, the above equations are of course 1-forms due to them being the result of the application of the exterior derivative to 0-forms. However, they are 1-forms on the computational domain. The computational domain is orthonormal, thus the basis that is used for the expansion of the partial derivative can be changed relatively easily, [33, 36] using a projection. Figure 45 shows how the same polynomial can be expanded in 1D in two different bases, one edge based, one nodal based. Most importantly, the dual mesh is also orthogonal on the computational domain and thus it is possible to transfer both dual and primal expansions to a common basis.

For the divergence as calculated in Equation 210 it makes the most sense for the divergence to be expanded in the nodal $h(x)h(y)$ expansion. By doing so a costly integral over every line piece between the Gauss-Lobatto-Legendre points can be avoided because of the multiplication of the elements of the Jacobian of the mapping, otherwise the change over the entire edge of the Jacobian would have to be incorporated. The nodal expansions also have the highest polynomial space and can thus accurately express the edge functions, the other way does not work because a lower order polynomial can only approximate a higher order polynomial. The expansion of all the partial derivatives into a nodal expansion also makes it possible to simply add all the contributions at all nodes. Additionally for the creation of

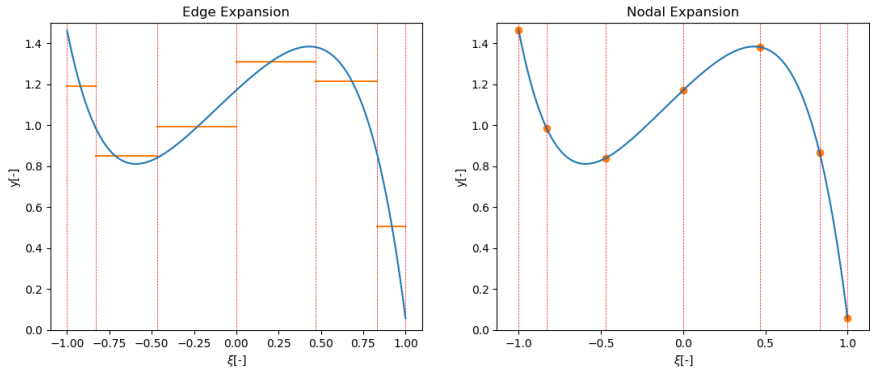

Figure 45: Different expansions of the same polynomial. Edge function based expansion (left), Nodal based expansion based (right)

the mass matrices the Jacobian of the transformation has been calculated at nodal locations anyway. The projection of the partial derivatives into the nodal basis can be done via a matrix. For example when $n = 2$ the basis function $e_0(\xi)h_0'(\eta)$ can be expressed as.

$$\begin{aligned} e_0(\xi)h_0'(\eta) = 1.875h_0(\xi)h_0(\eta) + 0.75h_1(\xi)h_0(\eta) - 0.375h_2(\xi)h_0(\eta) + \\ 0.625h_0(\xi)h_1(\eta) + 0.25h_1(\xi)h_1(\eta) - 0.125h_2(\xi)h_1(\eta) - \\ 0.625h_0(\xi)h_2(\eta) - 0.25h_1(\xi)h_2(\eta) + 0.125h_2(\xi)h_2(\eta). \end{aligned} \tag{217}$$

As mentioned, the Jacobian of the map is calculated at the nodal locations of the mesh, thus the calculation of the divergence can be calculated nodally by applying Equation 210 at each node. Finally, in order for the pressure to be dual to the divergence the pressure has to be expanded as

$$p = \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}e_i'(x)e_j'(y). \tag{218}$$

8.1.3 The Rotation Matrix

The Rotation Matrix is treated in a much simpler way. The implementation for setting the 2 shear stresses to be equal has been described in the previous section for a square domain. Because the stress is defined as two separate 1-forms, one for each stress direction it is possible to transform this to the physical domain and set the dx component of σ_x equal to the dy from σ_y.

$$\sigma = \begin{bmatrix}\sigma_x \\ \sigma_y\end{bmatrix} = \begin{bmatrix}\sigma_{xx}\mathrm{d}\eta - \sigma_{xy}\mathrm{d}\xi \\ \sigma_{yx}\mathrm{d}\eta - \sigma_{yy}\mathrm{d}\xi\end{bmatrix}. \tag{219}$$

Transforming these 1-forms back to the physical domain results in.

$$\sigma_x = -\frac{\sigma_{xx}}{|\Phi^*|}\frac{\partial y}{\partial \xi}dx + \frac{\sigma_{xx}}{|\Phi^*|}\frac{\partial x}{\partial \xi}dy - (\frac{\sigma_{xy}}{|\Phi^*|}\frac{\partial y}{\partial \eta}dx - \frac{\sigma_{xy}}{|\Phi^*|}\frac{\partial x}{\partial \eta}dy), \tag{220}$$

$$\sigma_y = -\frac{\sigma_{yx}}{|\Phi^*|}\frac{\partial y}{\partial \xi}dx + \frac{\sigma_{yx}}{|\Phi^*|}\frac{\partial x}{\partial \xi}dy - (\frac{\sigma_{yy}}{|\Phi^*|}\frac{\partial y}{\partial \eta}dx - \frac{\sigma_{yy}}{|\Phi^*|}\frac{\partial x}{\partial \eta}dy). \tag{221}$$

On the square derivation of the method σ_{xy} and σ_{yx} are associated with dx and dy respectively, thus from the σ_x row the dx terms and from the σ_y row the dy terms need to be set equal.

$$\frac{\sigma_{xx}}{|\Phi^*|}\frac{\partial y}{\partial \xi} + \frac{\sigma_{xy}}{|\Phi^*|}\frac{\partial y}{\partial \eta}dx = \frac{\sigma_{yx}}{|\Phi^*|}\frac{\partial x}{\partial \xi} + \frac{\sigma_{yy}}{|\Phi^*|}\frac{\partial x}{\partial \eta}dy. \tag{222}$$

This results in the matrix $\mathbb{R}$ becoming a 4 component matrix, $\mathbb{R} = [\mathbb{R}_{xx}, \mathbb{R}_{xy}, \mathbb{R}_{xy}\mathbb{R}_{yy}]$ with components.

$$(\mathbb{R}_{xx})_{ijkl} = \sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i'(x_r)h_j'(y_s)h_k(x_r)e_l'(y_s)w_r w_s \frac{1}{|det(\mathbf{\Phi}(x_r, y_s)^*)|}\frac{\partial y}{\partial \xi}(x_r, y_s), \tag{223}$$

$$(\mathbb{R}_{xy})_{ijkl} = \sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i'(x_r)h_j'(y_s)e_k(x_r)h_l'(y_s)w_r w_s \frac{1}{|det(\mathbf{\Phi}(x_r, y_s)^*)|}\frac{\partial y}{\partial \eta}(x_r, y_s), \tag{224}$$

$$(\mathbb{R}_{yx})_{ijkl} = -\sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i'(x_r)h_j'(y_s)h_k'(x_r)e_l(y_s)w_r w_s \frac{1}{|det(\mathbf{\Phi}(x_r, y_s)^*)|}\frac{\partial x}{\partial \xi}(x_r, y_s), \tag{225}$$

$$(\mathbb{R}_{yy})_{ijkl} = -\sum_{r=1}^{n+1}\sum_{s=1}^{n+1} h_i'(x_r)h_j'(y_s)e_k'(x_r)h_l(y_s)w_r w_s \frac{1}{|det(\mathbf{\Phi}(x_r, y_s)^*)|}\frac{\partial x}{\partial \eta}(x_r, y_s). \tag{226}$$

8.2 Results

8.2.1 Lid Driven Cavity on Deformed Mesh

To test the new formulation the lid driven cavity case has been tested on a deformed mesh. The deformed mesh can be seen in Figure 46 and can be created using the following formulas.

$$x = \frac{1}{2}(1 + \xi + c\sin(\pi\xi)\sin(\pi\eta), \tag{227}$$

$$y = \frac{1}{2}(1 + \eta + c\sin(\pi\xi)\sin(\pi\eta). \tag{228}$$

Solving the lid driven cavity problem on the deformed grid does not have the expected results. While the solution does look very much like the solution from section 7 as can be seen in Figure 47. The symmetry of the stress tensor is not preserved, When plotting the result using Equation 222 the error is most certainly not zero, as can be seen in Figure 48. Plotting the polynomials with a higher resolution as can be seen in Figure 49 shows that it is not only at the nodes that the error is not zero, and as a matter of fact is rather large. Equation 222 also explains why the error is likely not possible to be resolved. With all four

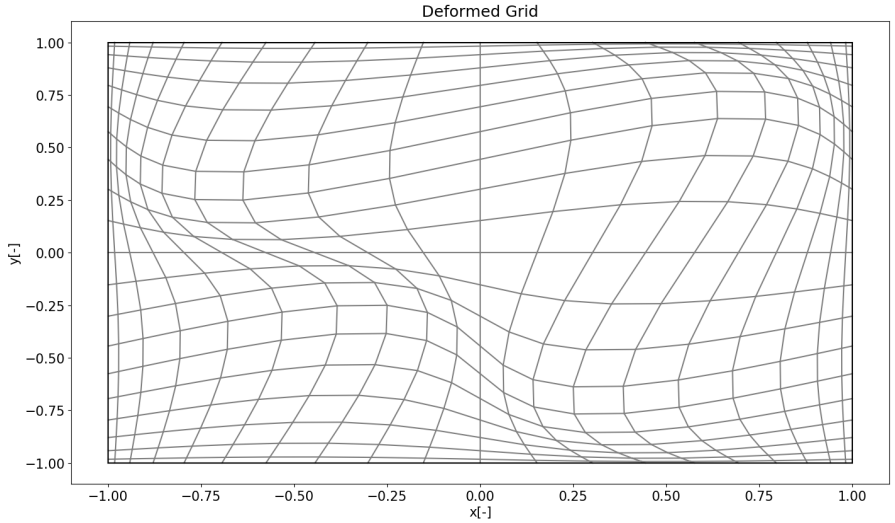

Figure 46: Deformed grid, N = 20, c = 0.2

components of the Jacobian of the transformation present in the equation and multiplying the Jacobian components with the solution polynomial new polynomials of a higher degree are created. With the solution only constrained by a number of degrees comparable to that of the nodes it is impossible to satisfy Equation 222 everywhere in the domain.

Some additional results can be seen in Appendix C

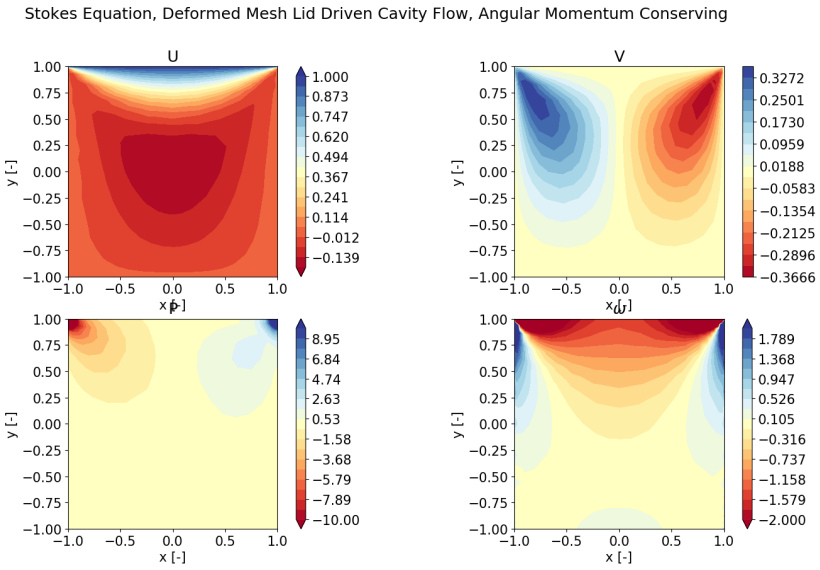

Figure 47: Solution on Deformed mesh, $n = 20$

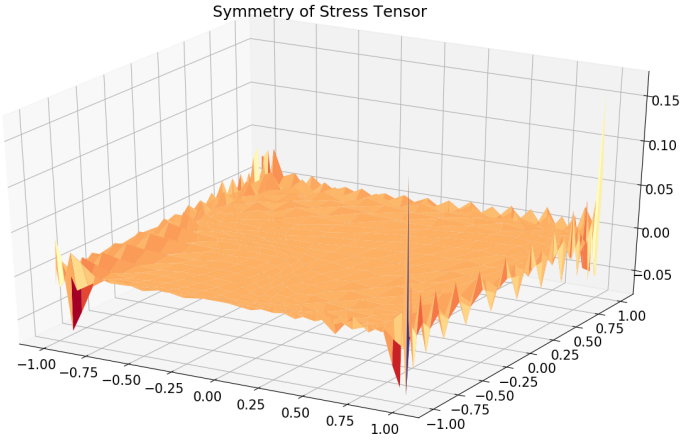

Figure 48: Symmetry error of stress tensor at nodal locations on the Deformed mesh, $n = 20$

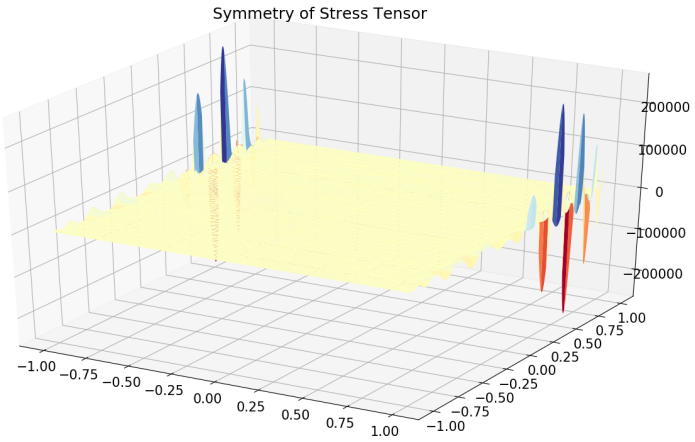

Figure 49: Symmetry error of stress tensor on Deformed mesh shown in polynomial expansion, $n = 20$

9 Time-Dependent Poisson Problems

All the previous formulation have considered the steady state of problems. For many different reasons the dynamic evolution of a problem can be of great interest. To achieve this the methods need to be extended with time derivatives.

9.1 The Parabolic Heat Equation

To start, the parabolic heat equation is considered. This problem is probably most known as the heat equation and is defined as

$$\frac{\partial u}{\partial t} = \Delta u - f \tag{229}$$

Here we see that the problem is of course very similar to the steady Poisson equations from section 5. The only difference is the addition of the time derivative in the equation. In numerical simulation this is a very helpful property because it allows the discretization of the time derivative separately from the spatial discretization, [42]. The mimetic spectral element method is usually a spatial discretization method unless the time dimension is included directly in the geometry of the problem.

9.1.1 Time Stepping

When a problem becomes too large it becomes unfeasible to add an entire dimension for time to the mesh due to storage or computational effort constraints. An option is to perform semi-discretization where the steady problem is discretized using an elaborate method and the time derivative is discretized in a different, usually much simpler way. It is possible to take this idea to the 2 dimensional time-dependent Poisson problem by adding a time derivative term to the steady equations and multiplying this term with the appropriate test functions.

$$(\tilde{\boldsymbol{q}}, \boldsymbol{q})_\Omega + (p, \nabla \cdot \tilde{\boldsymbol{q}})_\Omega = \int_{\partial\Omega} \hat{p}\tilde{\boldsymbol{q}} \cdot \boldsymbol{n} d\Gamma_\Omega, \tag{230}$$

$$(\tilde{p}, \frac{\partial p}{\partial t}) - (\tilde{p}, \nabla \cdot \boldsymbol{q})_\Omega = (\tilde{p}, f)_\Omega. \tag{231}$$

Creating a semi discretized version of this system in the spatial dimensions using a mimetic spectral element method creates the system

$$\begin{bmatrix} 0 \\ (\tilde{p}, \frac{\partial p}{\partial t}) \end{bmatrix} = \begin{bmatrix} \mathbb{M}^(1) & (\mathbb{E}^{2,1})^T \\ \mathbb{E}^{2,1} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ p \end{bmatrix} + \begin{bmatrix} -\boldsymbol{g} \\ -\boldsymbol{f} \end{bmatrix} \tag{232}$$

When chosing a simple backwards Euler discretization for the time derivative the formulation becomes

$$\begin{bmatrix} 0 \\ (\tilde{p}, \frac{p^{n+1}-p^n}{\Delta t}) \end{bmatrix} = \begin{bmatrix} \mathbb{M}^(1) & (\mathbb{E}^{2,1})^T \\ \mathbb{E}^{2,1} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ p^{n+1} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{g} \\ -\boldsymbol{f} \end{bmatrix}. \tag{233}$$

with p in the dual space of the primal surfaces (h'(x)h'(y) expansion) the product of $(\tilde{p}, p)$ becomes the inverse of the $\mathbb{M}^{(2)}$ matrix. This will then result in the numerical discretization.

$$\begin{bmatrix} 0 \\ \frac{(\mathbb{M}^{(2)})^{-1}p^{n+1} - (\mathbb{M}^{(2)})^{-1}p^{n}}{\Delta t}) \end{bmatrix} = \begin{bmatrix} \mathbb{M}(1) & (\mathbb{E}^{2,1})^{T} \\ \mathbb{E}^{2,1} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q}^{n+1} \\ p^{n+1} \end{bmatrix} + \begin{bmatrix} -\boldsymbol{g} \\ -\boldsymbol{f} \end{bmatrix}. \tag{234}$$

which then compacts to

$$\begin{bmatrix} \mathbb{M}(1) & (\mathbb{E}^{2,1})^{T} \\ \mathbb{E}^{2,1} & -\frac{(\mathbb{M}^{(2)})^{-1}}{\Delta t} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{g} \\ -\frac{(\mathbb{M}^{(2)})^{-1}}{\Delta t} p^{n} + \boldsymbol{f} \end{bmatrix}. \tag{235}$$

Which has the very useful property that the system remains symmetric. Using the same method it should be possible to use more advanced time integration techniques on the semi discretized system.

9.1.2 Convergence

To analyse the convergence properties of combining the mimetic spectral element method as semi discrete system with the backwards Euler method a manufactured solution is used. The time-dependent Poisson equation have an exact solution which makes testing this formulation relatively simple. The chosen manufactured solution is a cosine wave in on $[-1, 1]^2$ as initial condition with zero valued Dirichlet boundary conditions on the edges of the domain.

$$U(x, y, 0) = cos(\frac{\pi x}{2})cos(\frac{\pi y}{2}), \tag{236}$$
$$U(1, y, t) = 0, \tag{237}$$
$$U(-1, y, t) = 0, \tag{238}$$
$$U(x, 1, t) = 0, \tag{239}$$
$$U(x, -1, t) = 0. \tag{240}$$
$$\tag{241}$$

The solution for which is given by.

$$U(x, y, t) = e^{-\frac{\pi^2}{2}t}cos(\frac{\pi x}{2})cos(\frac{\pi y}{2}). \tag{242}$$

The convergence is examined for different time steps, mesh sizes and polynomial orders. In Figure 50 the convergence due to decreasing time steps is presented. The method uses fourth order elements, with K, the number of elements in each spatial dimension, being either 2,3 or 4 resulting in 4,9 or 16 elements spanning the domain. Here it is easily seen that the method converges for smaller time steps . As expected however, there is a lower limit to the error that based is on the spatial discretization. This can be seen for the case of K = 2. where the error to the solution no longer decreases significantly after a Δt of 10^{-3} is reached. Increasing the spatial resolution then has the effect of lowering the lower limit of the error. Note that the method also converges with the error proportional to the size

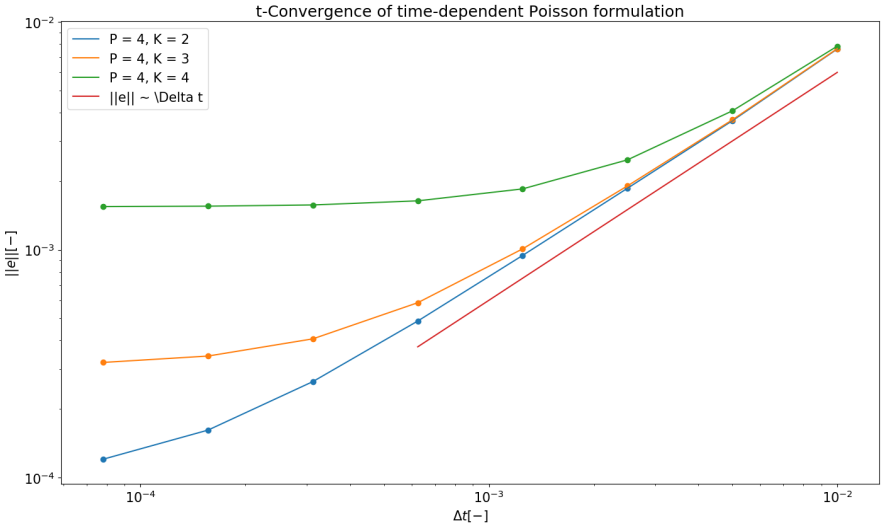

Figure 50: t-convergence of time-dependent formulation

of Δt, so long as the error is dominated by the time resolution. This is exactly as expected because the backwards Euler method is a first order method.

Similarly, in Figure 51 the convergence with respect to the spatial resolution is shown with increasing elements (decreasing h). Here the same behaviour is seen as with the convergence for the time step size, in this case however the plateauing is caused by the time resolution instead of the spatial resolution. Again note that the slope of the convergence in the part that is not restricted by the time resolution is equal to the mesh size cubed, again exactly as expected with the polynomial order of the elements used being 3.

Finally, as can be seen in Figure 52 the method also converges with increasing the polynomial order of the elements. Again the total error is limited due to the resolution in time, as can be clearly seen by the plateaus.

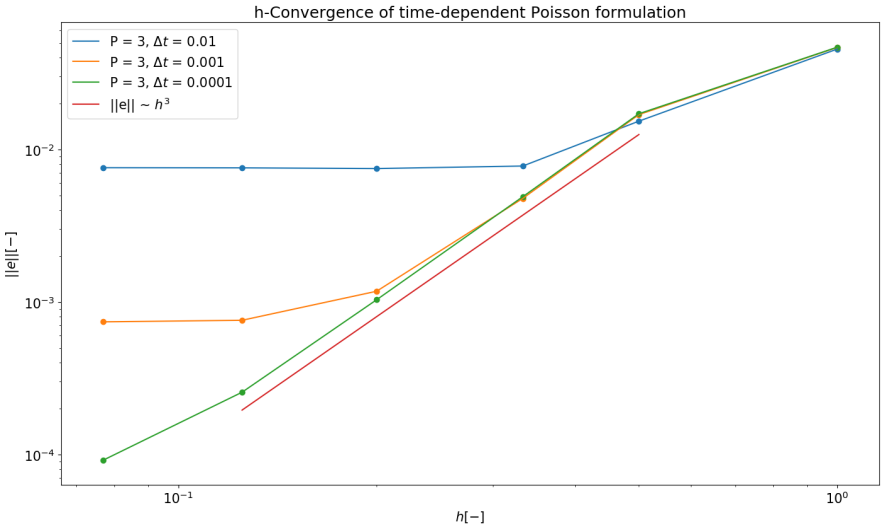

Figure 51: h-convergence of time-dependent formulation

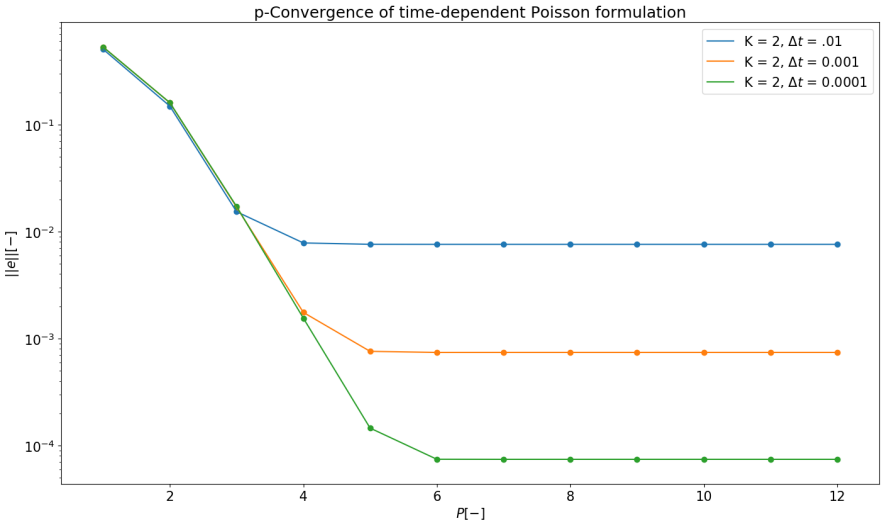

Figure 52: p-convergence of time-dependent formulation

10 Navier-Stokes

Incorporating the convective term into the Stokes equations will result in the Navier-Stokes equations, the equations that govern fluid flow. Many methods exist to incorporate the convective terms.

10.1 Flux Based Methods

In finite difference methods the convective terms can readily be approximated by an approximate derivative and local values. Finite volume methods have more options, most of which are flux based. The simpler methods are, for instance, the central and upwind discretization for the flux. a more elaborate method is a Godunov scheme, [10, 24] where every cell is assumed to have a piecewise constant approximation to the solution. At the cell boundaries the flux is then calculated using (approximate) Riemann solvers. Total variation diminishing schemes are even more elaborate, [43, 44]. Instead of a piecewise constant approximation the TVD schemes usually assume a linear approximation to the solution and then limit the slopes of the piecewise linear approximation. Limiting the slopes of the approximation then results in differences at the cell faces which are then used in an (approximate) Riemann solver to calculate the fluxes. Interestingly, even tough the idea for the total variation diminishing schemes is rooted in the finite volume schemes it is possible to use it in a finite element formulation or a hybrid of finite volume and finite element method, [22, 39]

Another class of methods to calculate the convective fluxes are the ENO/WENO class of methods, [26]. they are related to the TVD schemes in the sense that they generate values on both sides of the cell boundaries in order use these in the (approximate) Riemann solvers to calculate the fluxes. ENO schemes use multiple smaller stencils over a larger stencil and determine which stencil is least likely to contain a discontinuity in the solution. This stencil is then used as input for the fluxes. WENO schemes expand on this idea by creating (non-linear) combinations of all the smaller stencils to create a weighted approximation for the values on both sides of the cell boundary to use as input for the flux calculations.

Because these methods are for a finite volume method they inherently conserve quantities on the global level. The problem lies in the fact that all approximations use a polynomial to approximate a value of the function that is not explicitly tracked as a degree of freedom in the solver. The result is that however small the error of some approximations are, they all have either a dispersive or diffusive error. Based on the dispersive and diffusive errors for all these formulations the decision was made to look to alternative ways of representing the convective terms.

10.2 Lagrangian Convection

The flux based methods of calculating the convective term for the Navier-Stokes equations can not evaluate the convective term without loss of information. There is however a way in which the convective term would not be required. If the mesh itself (and the physical quantities on it) are convected with the flow itself, there is no need for a convective term to get to the Navier-Stokes equations. The formulation would reduce to the Stokes equations on a moving mesh. This is of course not a new idea, [13, 35] but a Lagrangian formulation would allow the use of the mimetic method for the Stokes equation that conserves angular

momentum. To achieve this first, a time dependent formulation of the new formulation should be derived.

10.2.1 Time-Dependent Stokes

Using the semi discretization method as developed in section 9 it is possible to create a time dependent Stokes formulation that conserves angular momentum. Starting from the steady state definition.

$$\frac{1}{2\nu}\underline{\underline{\sigma}} = \nabla \boldsymbol{u} - \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}, \tag{243}$$

$$\nabla \cdot \underline{\underline{\sigma}} - \nabla p + \boldsymbol{f} = 0, \tag{244}$$

$$\sigma_{yx} - \sigma_{xy} = 0, \tag{245}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{246}$$

Here it must be noted that of the four equations here three are conditions or relations. Only the momentum equation (also known as the kinetic equation) can be seen as an equation capable of evolving in time. The time derivative is therefore added only to the momentum equation.

$$\frac{1}{2\nu}\underline{\underline{\sigma}} = \nabla \boldsymbol{u} - \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}, \tag{247}$$

$$\frac{\partial u}{\partial t} = \nabla \cdot \underline{\underline{\sigma}} - \nabla p + \boldsymbol{f}, \tag{248}$$

$$\sigma_{yx} - \sigma_{xy} = 0, \tag{249}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{250}$$

Taking the numerical spacial discretization for the Stokes problem and adding the time derivative to this results in.

$$\begin{bmatrix} 0 \\ -\frac{\partial u}{\partial t} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 \\ \mathbb{E}_\sigma & 0 & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T \\ \mathbb{R} & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma} \\ \boldsymbol{u} \\ \boldsymbol{\omega} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ -\boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \end{bmatrix}. \tag{251}$$

The same procedure as in section 9 is then used to create a backwards Euler method for the new formulation.

$$\begin{bmatrix} 0 \\ -(\tilde{u}, \frac{\partial u}{\partial t}) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 \\ \mathbb{E}_\sigma & 0 & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T \\ \mathbb{R} & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma} \\ \boldsymbol{u} \\ \boldsymbol{\omega} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ -\boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \end{bmatrix}. \tag{252}$$

$$\begin{bmatrix} 0 \\ -(\frac{\mathbb{M}_u^{(1)} u^{n+1} - \mathbb{M}_u^{(1)} u^n}{\Delta t}) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 \\ \mathbb{E}_\sigma & 0 & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T \\ \mathbb{R} & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma^{n+1}} \\ \boldsymbol{u^{n+1}} \\ \boldsymbol{\omega^{n+1}} \\ \boldsymbol{p^{n+1}} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ -\boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \end{bmatrix}. \tag{253}$$

$$\begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 \\ \mathbb{E}_\sigma & -\frac{\mathbb{M}_u}{\Delta t} & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T \\ \mathbb{R} & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma^{n+1}} \\ \boldsymbol{u^{n+1}} \\ \boldsymbol{\omega^{n+1}} \\ \boldsymbol{p^{n+1}} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ \frac{-\mathbb{M}_u u^n}{\Delta t} - \boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \end{bmatrix}. \tag{254}$$

To incorporate the potential for a variable density (remembering that momentum is the product of the density and the velocity) in a fluid the momentum $\boldsymbol{\phi}$ itself can be added to this formulation by introducing it as the dual to the velocity. The formulation becomes.

$$\begin{bmatrix} \mathbb{M}_\sigma & \mathbb{E}_\sigma^T & \mathbb{R}^T & 0 & 0 \\ \mathbb{E}_\sigma & 0 & 0 & (\mathbb{PE}_{div\boldsymbol{u}})^T & \frac{I}{\Delta t} \\ \mathbb{R} & 0 & 0 & 0 & 0 \\ 0 & (\mathbb{PE}_{div\boldsymbol{u}}) & 0 & 0 & 0 \\ 0 & I & 0 & 0 & -\mathbb{M}_u \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}^{n+1} \\ \boldsymbol{u^{n+1}} \\ \boldsymbol{\omega}^{n+1} \\ \boldsymbol{p}^{n+1} \\ \boldsymbol{\phi}^{n+1} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{g}_\sigma \\ -\frac{\phi^n}{\Delta t} - \boldsymbol{f} - \boldsymbol{g}_u \\ 0 \\ \boldsymbol{g}_p \\ 0 \end{bmatrix}. \tag{255}$$

With I being the identity matrix, and with a constant density is equivalent to the more compact formulation. Example results for a lid driven cavity flow from rest can be seen in figures Figure 53 to Figure 55 where it can clearly be seen that the time dependent solution converges to the steady state solution as expected.

10.2.2 Lagrangian Moving Mesh

The time-dependent Stokes problem will at all time steps have the velocity as a product of the solution. The mesh can thus be convected at all points with the velocity at that point. Very elaborate numerical integration can be used, but here a simple forward Euler approach is used.

$$x_{i,j}^{n+1} = x_{i,j}^n + \Delta t \; u_{i,j}^{n+1}, \tag{256}$$

$$y_{i,j}^{n+1} = y_{i,j}^n + \Delta t \; v_{i,j}^{n+1}. \tag{257}$$

The moving mesh also implies a new transformation at every time step. Most importantly the transformation needs to be able to represent the motion of all nodal points. To do so the transformation can be represented by the nodal basis functions $h(\xi)h(\eta)$. Due to the Kronecker delta property of the nodal basis functions the map from the computational domain to the physical domain is relatively simple.

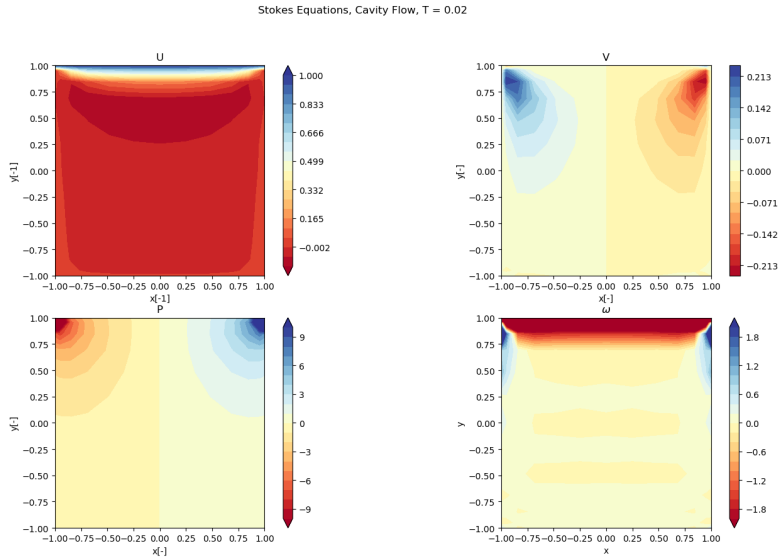

Figure 53: Lid driven cavity started from rest at t = 0.02, $\Delta t = 0.0025$

$$x(\xi,\eta) = \sum_{i=1}^{n+1}\sum_{j=1}^{n+1} x_{i,j} h_i(\xi) h_j(\eta), \tag{258}$$

$$y(\xi,\eta) = \sum_{i=1}^{n+1}\sum_{j=1}^{n+1} y_{i,j} h_i(\xi) h_j(\eta). \tag{259}$$

Where $x_{i,j}$ and $y_{i,j}$ are x- and y-coordinates of the nodal locations of the element. This also allows the Jacobian of the map to be calculated using the familiar and cheap to apply incidence matrices and the edge basis function expansion.

$$\frac{\partial x}{\partial \xi}(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{n+1} (\mathbb{E}_x^{(1,0)}\boldsymbol{x})_{i,j} e_i(\xi) h_j(\eta), \tag{260}$$

$$\frac{\partial x}{\partial \eta}(\xi,\eta) = \sum_{i=1}^{n+1}\sum_{j=1}^{n} (\mathbb{E}_x^{(1,0)}\boldsymbol{x})_{i,j} h_i(\xi) e_j(\eta), \tag{261}$$

$$\frac{\partial y}{\partial \xi}(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{n+1} (\mathbb{E}_x^{(1,0)}\boldsymbol{y})_{i,j} e_i(\xi) h_j(\eta), \tag{262}$$

$$\frac{\partial x}{\partial \eta}(\xi,\eta) = \sum_{i=1}^{n+1}\sum_{j=1}^{n} (\mathbb{E}_x^{(1,0)}\boldsymbol{y})_{i,j} h_i(\xi) e_j(\eta), \tag{263}$$

Where $\boldsymbol{x}$ and $\boldsymbol{y}$ represent the vector containing the x and y values of the nodal locations, and $\mathbb{E}_x^{(1,0)}$ and $\mathbb{E}_y^{(1,0)}$ are the x and y components of the $\mathbb{E}^{(1,0)}$ matrix respectively.

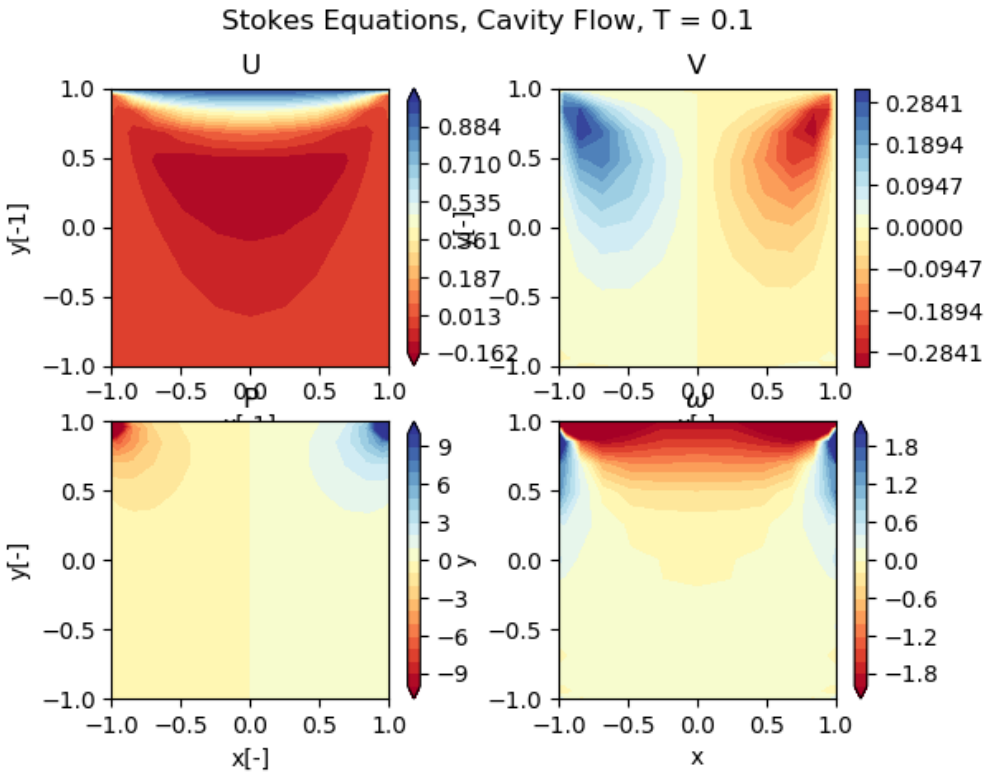

Figure 54: Lid driven cavity started from rest at t = 0.10, $\Delta t = 0.0025$

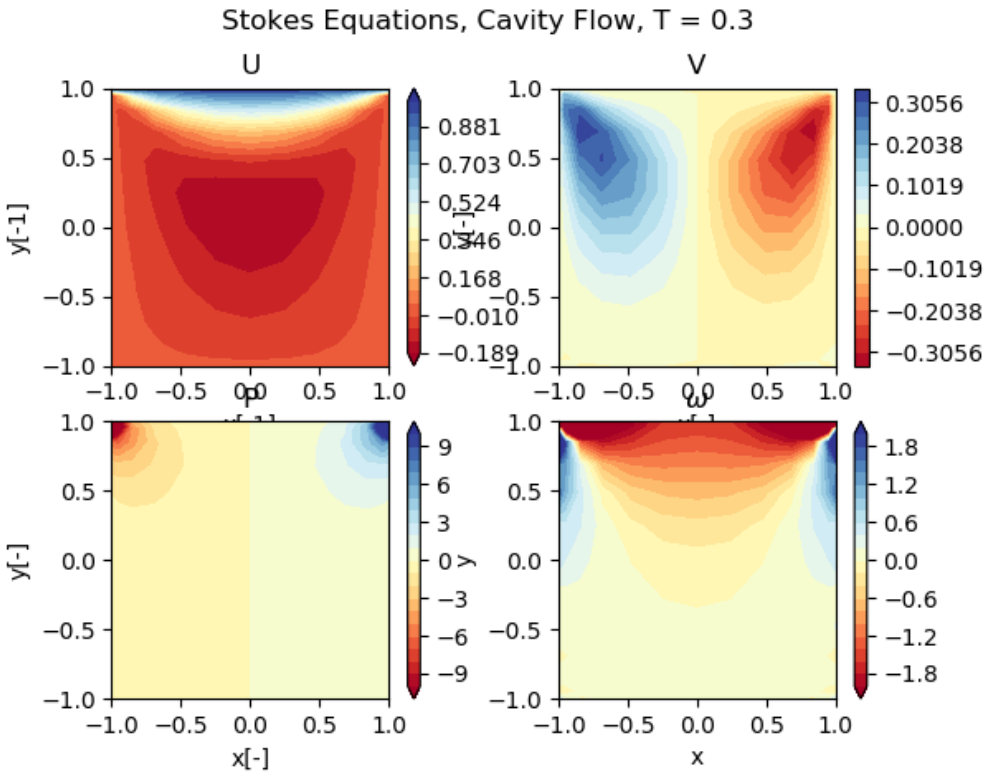

Figure 55: Lid driven cavity started from rest at t = 0.30, $\Delta t = 0.0025$

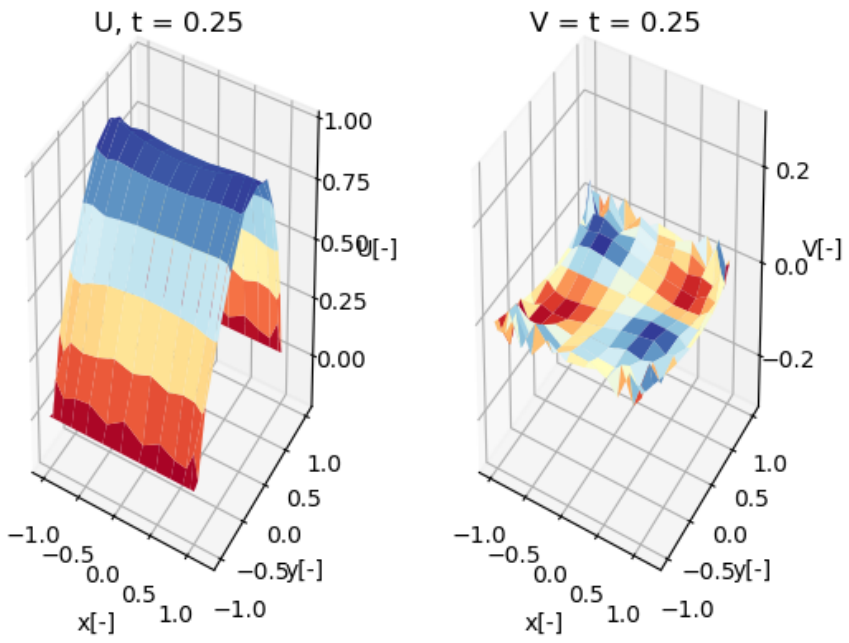

Figure 56: Solution for U(left) and V(right) after 5 time steps in the Navier-Stokes formulation

10.2.3 Navier Stokes

Combining the moving mesh with the velocity of the flow and the time dependant Stokes formulation produces the Navier-Stokes formulation. To test the formulation the problem was tested using the Poiseuille flow. The Poiseuille flow is an exact solution to the laminar flow through a channel and has the same solution in both the Stokes and the Navier-Stokes equations because the convective term reduces to 0. It is also a good test for the Lagrangian formulation because the test should just produce a parabolic element mesh because the solution to the velocity is a parabola. The simple solution is thus also a good first simple test to see if the formulation works. The formulation is however not stable. After only several time steps the simulation blows up as can be seen in Figure 56 and Figure 57. Even setting the time step to a very small step does not improve the stability significantly. The same instability occurs in all the variables instantaneously, making it hard to point to the exact problem. It is however likely that the fact that angular momentum is not conserved as shown in section 8 results in similar behavior as the rotating shell problem by Després et al. [12] shown in Figure 2.

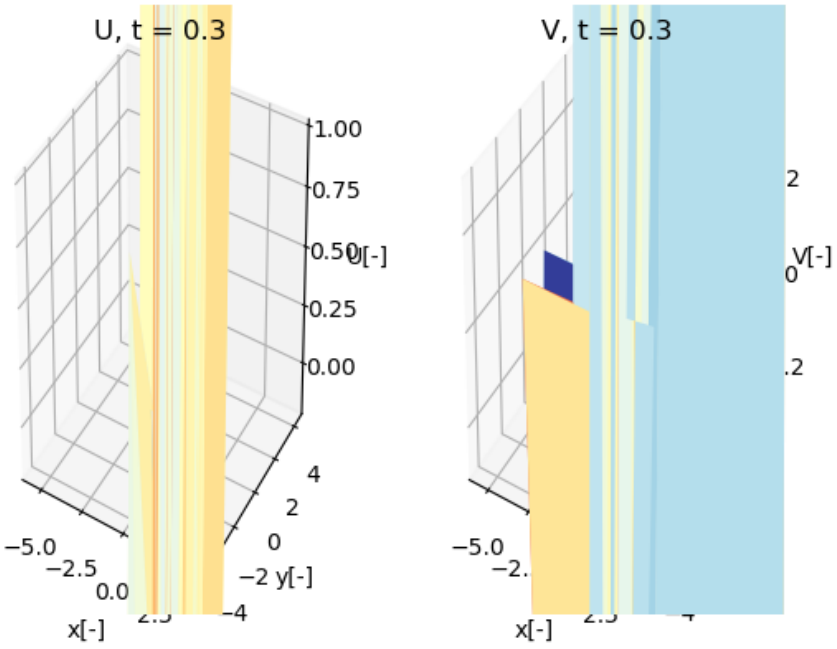

Figure 57: Solution for U(left) and V(right) after 6 time steps in the Navier-Stokes formulation, where the simulation has completely blown up

10.3 Unresolved Problems

The instability of the formulation leads to the question of why the formulation is unstable. Of course, if the answer was obvious the problem would have been solved. Here the instability can lie in a multitude of problems.

10.3.1 Mixed Basis

The source of a lot of these potential problems is the choice of the mixed basis functions. This basis is the tensor product of the basis functions on the primal grid in one direction and the dual basis functions in the other direction. It is however this "other" direction that immediately also is the problem. Due to the dual grid being defined through an inner product, it is not explicitly known. This is not a problem in a square grid for which the original formulation, [18] was derived because it would be orthogonal anyway, it is however a problem in a curvilinear grid. This would most likely not even be a problem if the mixed basis functions were used in just a mimetic spectral element context. The use of mass matrices and incidence matrices alone is likely sufficient to account for this because the only time that curvature is required to be directly addressed is in the mass matrices. In this formulation however multiple projections are made, and thus the location and direction are required. Because the expansions are a mixed basis it is not *exactly* clear where the dual component of the basis is located. It had been hoped the dual basis component could be transformed back to the primal basis with a set of only one dimensional mass matrices. Again, this is possible in a rectangular domain with no curvature. To see why the dual basis functions can never be a multiplication of just one dimensional mass matrices consider the mass matrix $\mathbb{M}^{(1)}$ constructed from the primal basis functions $e(x)h(y)$ and $h(x)e(y)$. This mass matrix is block diagonal on a rectangular grid and this is consistent with a multiplication of a set of 1 dimensional mass matrices, the off diagonal blocks however become filled and dense if the map becomes curvilinear. Something that can only happen due to the intermixing of the components of the co-vector, and thus cannot be the product of a one dimensional mass matrix, because in only one dimension the other component does not exist. To this end, the notation of the canonical basis functions might not be accurate, the functions, assumed to be the tensor product of 2 one dimensional spaces read as,

$$e(x)h(y), h(x)e(y). \tag{264}$$

for the vector components, and its dual components would be expanded as.

$$h'(x)e'(y), e'(x)h'(y). \tag{265}$$

It might be more appropriate to consider these basis functions as more general functions of x and y, especially the dual functions. This would emphasize the transformation to the dual grid is not straightforward and that unless orthogonal grids are used, the dual basis functions should be constructed from the primal basis functions, not as the tensor product of 2 one-dimensional dual basis functions.

$$eh(x,y), he(x,y). \tag{266}$$

And their dual

$$(eh)'(x,y), (he)'(x,y). \tag{267}$$

10.3.2 Projection and Polynomial Order

A projection to the computational domain is made on two separate occasions, in the projection of the shear stress components and in the projection of the divergence. Looking at Equation 210 the divergence is calculated as a multiplication of the polynomial that represents the derivative on the computational domain and the polynomial that represents an element of the Jacobian of the mapping. Both these polynomials can be of the same degree as the representation power of the element itself. The resulting polynomial is then of twice the order that the element is capable of representing which results in the divergence being under constrained. The same applies for the shear stress components because here too is the polynomial of the shear stress multiplied with the polynomial representing a component of the Jacobian.

10.3.3 The divergence of the velocity

A more deep problem lies in the compatibility of the mimetic method with fluid flows. This can already be seen in the continuous formulation o f the Stokes flow.

$$\frac{1}{2\nu}\underline{\underline{\sigma}} = \nabla \boldsymbol{u} - \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix}, \tag{268}$$

$$\nabla \cdot \underline{\underline{\sigma}} - \nabla p + \boldsymbol{f} = 0, \tag{269}$$

$$\sigma_{yx} - \sigma_{xy} = 0, \tag{270}$$

$$\nabla \cdot \boldsymbol{u} = 0. \tag{271}$$

where both the divergence and the gradient of the velocity are required. The mimetic method relies on discretizing the physical quantities to geometric locations and two different locations are required to perform both the gradient and the divergence all at once. This is however impossible because the velocity needs to be discretized on two separate objects, while also being the same. This dual description of the velocity has been noted before by Tonti, [41], and can be seen as the Lagranian and the Eulerian description of the velocity. Tonti however also makes no attempt to connect or convert one description to the other. Another way of looking at this problem is seeing that the divergence of two different quantities is required, both the stress and the velocity. This implies that these two physical quantities should be discretized on the same geometric object. Something that is completely allowed in the mimetic method. The problem however is that there also exists a differential relation between the stress and the velocity, which implies that they can *not* be discretized on the same geometric object, because the differentiated function in the mimetic method is discretized on a different geometric object than the original function.

11 Conclusion

In order to investigate the potential of new formulation for the Stokes equations to be able to calculate solutions for the Navier-Stokes equations which strongly preserve angular momentum a set of modifications and additions to the formulation were made. First the formulation has been modified to allow for a general mesh instead of only being applicable to square or rectangular meshes. Time derivatives were implemented in several problems using a semi discretization method and using the mimetic method in the spatial dimensions. This was combined with a moving mesh method to represent the convective term in the Navier-Stokes equations. However, the modification of the method to general meshes was unsuccessful and revealed some underlying problems that have no readily available solution. As such, a formulation for the Navier-Stokes equations that explicitly conserves angular momentum has not been achieved.

This formulation leaves much to be desired compared to the simplicity of, for instance, the mimetic VVP formulation of the stokes equations. In the VVP formulation only the canonical elements present for a mimetic spectral element method are required, mass and incidence matrices and an appropriate set of boundary equations. The new formulation requires projections on 2 separate occasion, which are likely the reason the method does not produce correct results. Furthermore, the assumption that the basis for the expansions can be represented as a tensor product of both a primal basis function in one direction and a dual in the other direction has shown itself to be incredibly difficult to understand under curvature. This is also not helped by the implicit definition of the dual grid as the result of an inner product.

The formulation for the inclusion of the time-derivatives has been largely successful, with the semi discretization implementation being very close to the structure of the mimetic spectral element method. The success of the semi-discretisation method creates some interesting potential for the use of higher order time discretization methods to be combined with the mimetic discretization.

Especially the final implementation of the Navier-Stokes equations is unstable, with the current transformation rules being applied not being successful. As mentioned above the use of mixed basis functions clearly complicates the analysis due the unclear location of the degrees of freedom making transformations to curvilinear domains very hard. Convection of course can carry the mesh in all sorts of direction in a Lagrangian moving mesh method and thus the transformations must be incredibly robust. Furthermore the fact that the divergence needs to be applied both to the stress and the velocity, which implies that their discretization should be similar while the differential relationship between them suggests that they can not be discretized in the same way is a true paradox for this method.

11.1 Recommendations

The knowledge gained about the nature of the formulations created in this worked leave questions open for future work. Most prominently is the relation between stress and velocity and the need to take the divergence of both is a hard problem to solve. Also an alternative for the basis functions in which these variables were expressed in the present formulation might allow for a correct implementation of the idea to conserve angular momentum. The

formulation of the time dependent problem opens new avenues to be explored, with the semi discretization opening up the method to many different time stepping methods. A list of Items to work on for the future can thus be made in the following way.

- Investigate the mathematical properties of mixed basis functions in order to more thoroughly understand their nature under transformation.
- Create the present method on an explicit dual mesh in contrast to the current implicit mesh.
- Investigate the use of projection as a canonical element in the mimetic spectral element methods, as they have been required in multiple instances. Perhaps a more sophisticated projection can be found that is consistent with the conversion between differential forms.
- Investigate the behavior of the time dependent problem using more sophisticated time integration methods in order to conserve physical quantities in time.
- Investigate the possibility of incorporating flux based convective method in the mimetic spectral element method.

References

[1] Milton Abramowitz and Irene A Stegun. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. National Bureau of Standards Applied Mathematics Series 55. Tenth Printing. 1972.

[2] DJ Acheson. Elementary fluid dynamics: Oxford university press, 1990.

[3] John D. Anderson. *Fundamentals of aerodynamics*. McGraw-Hill, 5th edition, 2011.

[4] Fareed Ahmad Anwar-ul Haque, Shunsuke Yamada, and Sajid Raza Chaudhry. Assessment of turbulence models for turbulent flow over backward facing step. In *Proceedings of the World Congress on Engineering*, volume 2, pages 2–7, 2007.

[5] E Barragy and GF Carey. Stream function-vorticity driven cavity solution using p finite elements. *Computers & Fluids*, 26(5):453–468, 1997.

[6] Pavel B Bochev and James M Hyman. Principles of mimetic discretizations of differential operators. In *Compatible spatial discretizations*, pages 89–119. Springer, 2006.

[7] Alain Bossavit. Whitney forms: A class of finite elements for three-dimensional computations in electromagnetism. *IEE Proceedings A (Physical Science, Measurement and Instrumentation, Management and Education, Reviews)*, 135(8):493–500, 1988.

[8] O Botella and R Peyret. Benchmark spectral results on the lid-driven cavity flow. *Computers Fluids*, 27(4):421–433, 1998.

[9] Franco Brezzi and Annalisa Buffa. Innovative mimetic discretizations for electromagnetic problems. *Journal of computational and applied mathematics*, 234(6):1980–1987, 2010.

[10] Thierry Buffard, Thierry Gallouët, and Jean-Marc Hérard. A sequel to a rough Godunov scheme: application to real gases. *Computers Fluids*, 29(7):813–847, 2000.

[11] Lourenco Beirão Da Veiga. A mimetic discretization method for linear elasticity. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(2):231–250, 2010.

[12] Bruno Després and Emmanuel Labourasse. Angular Momentum preserving cell-centered Lagrangian and Eulerian schemes on arbitrary grids. *Journal of Computational Physics*, 290:28–54, 2015.

[13] Yana Di, Ruo Li, Tao Tang, and Pingwen Zhang. Moving mesh finite element methods for the incompressible Navier–Stokes equations. *SIAM Journal on Scientific Computing*, 26(3):1036–1056, 2005.

[14] Jozef Dodziuk. Finite-difference approach to the hodge theory of harmonic forms. *American Journal of Mathematics*, 98(1):79–104, 1976.

[15] Ercan Erturk. Numerical solutions of 2-D steady incompressible flow over a backward-facing step, Part I: High Reynolds number solutions. *Computers Fluids*, 37(6):633–655, 2008.

[16] Ercan Erturk. Discussions on driven cavity flow. *International journal for numerical methods in fluids*, 60(3):275–294, 2009.

[17] Ercan Erturk, Thomas C Corke, and Cihan Gökçöl. Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers. *International journal for Numerical Methods in fluids*, 48(7):747–774, 2005.

[18] Joël Marius Fisser. Advancing the Mimetic Spectral Element Method, Towards Continuum Mechanics Applications, 2019.

[19] Theodore Frankel. *The geometry of physics: an introduction*. Cambridge university press, 2011.

[20] Daniele Funaro. *Polynomial Approximation of Differential Equations*. Springer Berlin, Heidelberg, 1992.

[21] David K Gartling. A test problem for outflow boundary conditions—flow over a backward-facing step. *International Journal for Numerical Methods in Fluids*, 11(7):953–967, 1990.

[22] Sebastian Geiger, Stephen Roberts, Stephan K Matthäi, Christopher Zoppou, and Adrian Burri. Combining finite element and finite volume methods for efficient multiphase flow simulations in highly heterogeneous and structurally complex geologic media. *Geofluids*, 4(4):284–299, 2004.

[23] Marc Gerritsma. Edge Functions for Spectral Element Methods. In Jan S. Hesthaven and Einar M. Rønquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, pages 199–207, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[24] Sergei Godunov and I Bohachevsky. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. *Matematičeskij sbornik*, 47(3):271–306, 1959.

[25] G Guj and F Stella. Numerical solutions of high-Re recirculating flows in vorticity-velocity form. *International journal for numerical methods in fluids*, 8(4):405–416, 1988.

[26] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. In *Upwind and high-resolution schemes*, pages 218–290. Springer, 1987.

[27] H.T. Huynh, Z.J. Wang, and P.E. Vincent. High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids. *Computers Fluids*, 98:209–220, 2014. 12th USNCCM mini-symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 80th birthday of Professor Antony Jameson.

[28] James M Hyman and Mikhail Shashkov. Mimetic discretizations for maxwell's equations. *Journal of Computational Physics*, 151(2):881–909, 1999.

[29] V. Jain, Y. Zhang, A. Palha, and M. Gerritsma. Construction and application of algebraic dual polynomial representations for finite element methods on quadrilateral and hexahedral meshes. *Computers Mathematics with Applications*, 95:101–142, 2021. Recent Advances in Least-Squares and Discontinuous Petrov–Galerkin Finite Element Methods.

[30] Varun Jain, Yi Zhang, Artur Palha, and Marc Gerritsma. Construction and application of algebraic dual polynomial representations for finite element methods on quadrilateral and hexahedral meshes, 2017.

[31] Sangtae Kim and Seppo J Karrila. *Microhydrodynamics: principles and selected applications*. Courier Corporation, 2013.

[32] Jasper Kreeft and Marc Gerritsma. Mixed mimetic spectral element method for Stokes flow: A pointwise divergence-free solution. *Journal of Computational Physics*, 240:284–309, 2013.

[33] Jasper Kreeft, Artur Palha, and Marc Gerritsma. Mimetic spectral element method for generalized convection-diffusion problems. pages 14–17, 07 2010.

[34] Jasper Kreeft, Artur Palha, and Marc Gerritsma. Mimetic framework on curvilinear quadrilaterals of arbitrary order, 2011.

[35] Ivan Malcevic and Omar Ghattas. Dynamic-mesh finite element method for Lagrangian computational fluid dynamics. *Finite Elements in Analysis and Design*, 38(10):965–982, 2002.

[36] Artur Palha, Pedro Rebelo, and Marc Gerritsma. Mimetic spectral element advection. *Lecture Notes in Computational Science and Engineering*, 95, 04 2013.

[37] Artur Palha, Pedro Pinto Rebelo, René Hiemstra, Jasper Kreeft, and Marc Gerritsma. Physics-compatible discretization techniques on single and dual grids, with application to the poisson equation of volume forms. *Journal of Computational Physics*, 257:1394–1422, 2014. Physics-compatible numerical methods.

[38] Patrick J. Roache. Code Verification by the Method of Manufactured Solutions . *Journal of Fluids Engineering*, 124(1):4–10, 11 2001.

[39] Omid Seyedashraf and Ali Akbar Akhtari. Two-dimensional numerical modeling of dam-break flow using a new TVD finite-element scheme. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 39(11):4393–4401, 2017.

[40] Radhakrishnan Srinivasan. Accurate solutions for steady plane flow in the driven cavity. I. Stokes flow. *Zeitschrift für angewandte Mathematik und Physik ZAMP*, 46(4):524–545, 1995.

[41] Enzo Tonti. *The mathematical structure of classical and relativistic physics*, volume 10. Springer, 2013.

[42] JJIM van Kan, A Segal, and Fredericus Johannes Vermolen. *Numerical methods in scientific computing*. VSSD, 2005.

[43] Bram Van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of computational physics*, 14(4):361–370, 1974.

[44] HC Yee, RF Warming, and A Harten. Implicit total variation diminishing (TVD) schemes for steady-state calculations. *Journal of Computational Physics*, 57(3):327–360, 1985.

[45] Yi Zhang, Joël Fisser, and Marc Gerritsma. A hybrid mimetic spectral element method for three-dimensional linear elasticity problems. *Journal of Computational Physics*, 433:110179, 2021.

A Divergence Christoffel Symbols

In the divergence the possibility for Christoffel symbols arises, however the next derivation will show that in the trace of the gradient no Christoffel symbols appear. The gradient in a curved domain can be calculated as

$$\nabla u = du + P^{-1} dPu, \tag{272}$$
$$d(Pu) = Pdu + dPu \tag{273}$$

where the Christoffel symbols arise from dP.

$$P = \frac{1}{|\phi|}\begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{bmatrix}, \tag{274}$$
$$P^{-1} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix} \tag{275}$$

thus

$$dP = \frac{1}{|\phi|}\begin{bmatrix} \frac{\partial}{\partial \xi}\frac{\partial y}{\partial \eta}\mathrm{d}x + \frac{\partial}{\partial \eta}\frac{\partial y}{\partial \eta}\mathrm{d}y & -\frac{\partial}{\partial \xi}\frac{\partial x}{\partial \eta}\mathrm{d}x - \frac{\partial}{\partial \eta}\frac{\partial x}{\partial \eta}\mathrm{d}y \\ -\frac{\partial}{\partial \xi}\frac{\partial y}{\partial \xi}\mathrm{d}x - \frac{\partial}{\partial \eta}\frac{\partial y}{\partial \xi}\mathrm{d}y & \frac{\partial}{\partial \xi}\frac{\partial x}{\partial \xi}\mathrm{d}x + \frac{\partial}{\partial \eta}\frac{\partial x}{\partial \xi}\mathrm{d}y \end{bmatrix} \tag{276}$$

then

$$Pdu = \frac{1}{|\phi|}\begin{bmatrix} \frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta}\mathrm{d}x - \frac{\partial u}{\partial \xi}\frac{\partial x}{\partial \eta}\mathrm{d}y - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi}\mathrm{d}x + \frac{\partial u}{\partial \eta}\frac{\partial x}{\partial \xi}\mathrm{d}y \\ \frac{\partial v}{\partial \xi}\frac{\partial y}{\partial \eta}\mathrm{d}x - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta}\mathrm{d}y - \frac{\partial v}{\partial \eta}\frac{\partial y}{\partial \xi}\mathrm{d}x + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi}\mathrm{d}y \end{bmatrix} \tag{277}$$

and

$$dPu = \frac{1}{|\phi|}\begin{bmatrix} \frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta}\mathrm{d}x + \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \eta}\mathrm{d}y - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta}\mathrm{d}x - \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \eta}\mathrm{d}y \\ -\frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \xi}\mathrm{d}x - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi}\mathrm{d}y + \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \xi}\mathrm{d}x + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi}\mathrm{d}y \end{bmatrix} \tag{278}$$

Taking the trace of the above means summing the dx part of the upper row and the dy in the lower row, resulting in

$$\nabla \cdot u = \frac{1}{|\phi|}\left(\frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi} - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta} + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi} + \frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta} - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi} + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi}\right) = \tag{279}$$
$$\frac{2}{|\phi|}\left(\frac{\partial u}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial u}{\partial \eta}\frac{\partial y}{\partial \xi} - \frac{\partial v}{\partial \xi}\frac{\partial x}{\partial \eta} + \frac{\partial v}{\partial \eta}\frac{\partial x}{\partial \xi}\right) \tag{280}$$

B Mass Matrix Transformation

The mass matrices transform in the following way. For all form, the mapped inner product can be calculated as.

$$(\tilde{a}, a)_\Omega = \int_{\Phi(\hat{\Omega})} \tilde{a} \wedge \star a \, \mathrm{d}\Omega \tag{281}$$

Where $\hat{\Omega}$ is the reference $[-1, 1]^n$ domain. Using the transformation rules:

$$\begin{aligned}
(\tilde{a}, a)_\Omega &= \int_{\hat{\Omega}} \Phi^*[\tilde{a} \wedge \star a] \, \mathrm{d}\Omega && (282)\\
&= \int_{\hat{\Omega}} \Phi^* \tilde{a} \wedge \Phi^* \star a \, \mathrm{d}\Omega && (283)\\
&= \int_{\hat{\Omega}} \Phi^* \tilde{a} \wedge \Phi^* \star \Phi^{-*} \Phi^* a \, \mathrm{d}\Omega && (284)\\
&= \int_{\hat{\Omega}} \hat{\tilde{a}} \wedge \hat{\star} \hat{a} \, \mathrm{d}\Omega && (285)
\end{aligned}$$

where $\hat{a} = \Phi^* a$ and $\hat{\star} = \Phi^* \star \Phi^{-*}$. Note that $\Phi^{-*}\Phi^* a$ is the identity.

B.1 Mass Matrices for 0-forms

noting that a pullback operator acting on a 0-form does not change to a new form because it it just a function and thus.

$$\star \Phi^{-*} \hat{a}^{(0)} = \sum_{i=1}^{p+1} \sum_{j=1}^{p+1} a_{i,j} h_i(\xi) h_j(\eta) \mathrm{d}x \wedge \mathrm{d}y \tag{286}$$

then applying the pullback operator results in

$$\Phi^* \star \Phi^{-*} \hat{a}^{(0)} = \sum_{i=1}^{p+1} \sum_{j=1}^{p+1} a_{i,j} h_i(\xi) h_j(\eta) \left[\frac{\partial x}{\partial \xi} \mathrm{d}\xi + \frac{\partial x}{\partial \eta} \mathrm{d}\eta \right] \wedge \left[\frac{\partial y}{\partial \xi} \mathrm{d}\xi + \frac{\partial y}{\partial \eta} \mathrm{d}\eta \right] \tag{287}$$

which is then egual to

$$\begin{aligned}
\Phi^* \star \Phi^{-*} \hat{a}^{(0)} &= \sum_{i=1}^{p+1} \sum_{j=1}^{p+1} a_{i,j} h_i(\xi) h_j(\eta) \left[\frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \xi} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \right] \mathrm{d}\xi \wedge \mathrm{d}\eta && (288)\\
&= \sum_{i=1}^{p+1} \sum_{j=1}^{p+1} a_{i,j} h_i(\xi) h_j(\eta) |\boldsymbol{\Phi^*}| \, \mathrm{d}\xi \wedge \mathrm{d}\eta && (289)
\end{aligned}$$

because the inner product then is.

$$\int_\Omega \hat{\tilde{a}} \wedge \hat{\star} \hat{a} \, \mathrm{d}\Omega \approx \int_\Omega \sum_{i=1}^{p+1} \sum_{j=1}^{p+1} \sum_{k=1}^{p+1} \sum_{l=1}^{p+1} a_{i,j} \tilde{a}_{k,l} J(\xi, \eta) h_i(\xi) h_j(\eta) h_k(\xi) h_l(\eta) \, \mathrm{d}\xi \wedge \mathrm{d}\eta \tag{290}$$

With $J(\xi,\eta)$ being the determinant of the Jacobian of the mapping. The above approximation can be rewritten in the matrix form $\hat{a}^T\mathbb{M}a$ and thus the mass matrix $\mathbb{M}$ can be created using the following:

$$M^{(0)}_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} J(\xi_r,\eta_s)h_i(\xi_r)h_j(\eta_s)h_k(\xi_r)h_l(\eta_s)w_r w_s \tag{291}$$

Where ξ_r and η_s are the locations of the of the numerical integration nodes and w_r and w_s are the integration weights.

B.2 Mass Matrices for 1-forms

The procedure starts in the same fashion as the 0-form matrices with

$$(\tilde{a},a)_\Omega = \int_{\hat{\Omega}} \Phi^*[\tilde{a}\wedge\star a]\,\mathrm{d}\Omega \tag{292}$$

$$= \int_{\hat{\Omega}} \hat{\tilde{a}}\wedge\hat{\star}\hat{a}\,\mathrm{d}\Omega \tag{293}$$

and again $\hat{a}=\Phi^* a$ and $\hat{\star}=\Phi^*\star\Phi^{-*}$, starting with a definition of

$$\hat{a}^{(1)} = \bar{u}\mathrm{d}\eta - \bar{v}\mathrm{d}\xi \tag{294}$$

Then $\Phi^*\star\Phi^{-*}$ can be calculated by applying the operations in sequence

$$\Phi^{-*}\hat{a}^{(1)} = \bar{u}\Phi^{-*}\mathrm{d}\eta - \bar{v}\Phi^{-*}\mathrm{d}\xi \tag{295}$$

$$= \frac{1}{|\Phi^*|}\left[\bar{u}\left(-\frac{\partial y}{\partial\xi}\mathrm{d}x + \frac{\partial x}{\partial\xi}\mathrm{d}y\right) - \bar{v}\left(\frac{\partial y}{\partial\eta}\mathrm{d}x - \frac{\partial x}{\partial\eta}\mathrm{d}y\right)\right] \tag{296}$$

$$= \frac{1}{|\Phi^*|}\left[-\left(\bar{u}\frac{\partial y}{\partial\xi} + \bar{v}\frac{\partial y}{\partial\eta}\right)\mathrm{d}x + \left(\bar{u}\frac{\partial x}{\partial\xi} + \bar{v}\frac{\partial x}{\partial\eta}\right)\mathrm{d}y\right] \tag{297}$$

Then applying the Hodge-$\star$ operator which operates such that $\star\mathrm{d}x=\mathrm{d}y, \star\mathrm{d}y=-\mathrm{d}x$.

$$\star\Phi^{-*}\hat{a}^{(1)} = \frac{1}{|\Phi^*|}\left[-\left(\bar{u}\frac{\partial y}{\partial\xi} + \bar{v}\frac{\partial y}{\partial\eta}\right)\mathrm{d}y - \left(\bar{u}\frac{\partial x}{\partial\xi} + \bar{v}\frac{\partial x}{\partial\eta}\right)\mathrm{d}x\right] \tag{298}$$

Applying the pullback operator to this then results in

$$\Phi^*\star\Phi^{-*}\hat{a}^{(1)} = \frac{1}{|\Phi^*|}\left[-\left(\bar{u}\frac{\partial y}{\partial\xi} + \bar{v}\frac{\partial y}{\partial\eta}\right)\Phi^*\mathrm{d}y - \left(\bar{u}\frac{\partial x}{\partial\xi} + \bar{v}\frac{\partial x}{\partial\eta}\right)\Phi^*\mathrm{d}x\right] \tag{299}$$

$$= \frac{1}{|\Phi^*|}\left[-\left(\bar{u}\frac{\partial y}{\partial\xi} + \bar{v}\frac{\partial y}{\partial\eta}\right)\left[\frac{\partial y}{\partial\xi}\mathrm{d}\xi + \frac{\partial y}{\partial\eta}\mathrm{d}\eta\right] - \left(\bar{u}\frac{\partial x}{\partial\xi} + \bar{v}\frac{\partial x}{\partial\eta}\right)\left[\frac{\partial x}{\partial\xi}\mathrm{d}\xi + \frac{\partial x}{\partial\eta}\mathrm{d}\eta\right]\right] \tag{300}$$

This can be simplified as

$$\frac{-1}{|\Phi^*|}\left(\left(\left(\frac{\partial x}{\partial\xi}\right)^2 + \left(\frac{\partial y}{\partial\xi}\right)^2\right)\bar{u} + \left(\frac{\partial y}{\partial\xi}\frac{\partial y}{\partial\eta} + \frac{\partial x}{\partial\xi}\frac{\partial x}{\partial\eta}\right)\bar{v}\right)\mathrm{d}\xi + \tag{301}$$

$$\frac{-1}{|\Phi^*|}\left(\left(\frac{\partial y}{\partial\xi}\frac{\partial y}{\partial\eta} + \frac{\partial x}{\partial\xi}\frac{\partial x}{\partial\eta}\right)\bar{u} + \left(\left(\frac{\partial x}{\partial\eta}\right)^2 + \left(\frac{\partial y}{\partial\eta}\right)^2\right)\bar{v}\right)\mathrm{d}\eta \tag{302}$$

Then shorthanding the terms to

$$(G_{11}\bar{u} + G_{12}\bar{v})\mathrm{d}\xi + (G_{21}\bar{u} + G_{22}\bar{v})\mathrm{d}\eta \tag{303}$$

with

$$G_{11} = \frac{-1}{|\Phi^*|}\left(\left(\frac{\partial x}{\partial \xi}\right)^2 + \left(\frac{\partial y}{\partial \xi}\right)^2\right) \tag{304}$$

$$G_{12} = \frac{-1}{|\Phi^*|}\left(\frac{\partial y}{\partial \xi}\frac{\partial y}{\partial \eta} + \frac{\partial x}{\partial \xi}\frac{\partial x}{\partial \eta}\right) \tag{305}$$

$$G_{21} = \frac{-1}{|\Phi^*|}\left(\frac{\partial y}{\partial \xi}\frac{\partial y}{\partial \eta} + \frac{\partial x}{\partial \xi}\frac{\partial x}{\partial \eta}\right) \tag{306}$$

$$G_{22} = \frac{-1}{|\Phi^*|}\left(\left(\frac{\partial x}{\partial \eta}\right)^2 + \left(\frac{\partial y}{\partial \eta}\right)^2\right) \tag{307}$$

$$\tag{308}$$

now expanding $\bar{u}$ and $\bar{v}$ as

$$\bar{u} = \sum_{i=1}^{p+1}\sum_{j=1}^{p} u_{i,j} h_i(\xi) e_j(\eta), \tag{309}$$

$$\bar{v} = \sum_{i=1}^{1}\sum_{j=1}^{p+1} v_{i,j} e_i(\xi) h_j(\eta) \tag{310}$$

and defining the test functions as

$$\hat{\tilde{a}} = \sum_{k=1}^{p+1}\sum_{l=1}^{k}(a_\xi)_{k,l} h_k(\xi) e_l(\eta)\mathrm{d}\eta - \sum_{k=1}^{p}\sum_{l=1}^{p+1}(a_\eta)_{k,l} e_k(\xi) h_l(\eta)\mathrm{d}\xi \tag{311}$$

The integral for 1-forms becomes

$$\int_{\hat{\Omega}} \hat{\tilde{a}} \wedge \hat{\star}\hat{a}\ \mathrm{d}\Omega = \tag{312}$$

$$\int_{\hat{\Omega}} \sum_{k=1}^{p+1}\sum_{l=1}^{k}(a_\xi)_{k,l} h_k(\xi) e_l(\eta)\left(G_{11}\sum_{i=1}^{p+1}\sum_{j=1}^{p} u_{i,j} h_i(\xi) e_j(\eta) + G_{12}\sum_{i=1}^{1}\sum_{j=1}^{p+1} v_{i,j} e_i(\xi) h_j(\eta)\right)\mathrm{d}\xi \wedge \mathrm{d}\eta + \tag{313}$$

$$\int_{\hat{\Omega}} \sum_{k=1}^{p}\sum_{l=1}^{p+1}(a_\eta)_{k,l} e_k(\xi) h_l(\eta)\left(G_{21}\sum_{i=1}^{p+1}\sum_{j=1}^{p} u_{i,j} h_i(\xi) e_j(\eta) + G_{22}\sum_{i=1}^{1}\sum_{j=1}^{p+1} v_{i,j} e_i(\xi) h_j(\eta)\right)\mathrm{d}\xi \wedge \mathrm{d}\eta \tag{314}$$

Giving rise to a matrix form

$$(\hat{\tilde{a}})^T \mathbb{M}^{(1)} \hat{a} = \begin{bmatrix} (\tilde{a}_\xi)^T M_{11} u & (\tilde{a}_\xi)^T M_{12} v \\ (\tilde{a}_\eta)^T M_{21} u & (\tilde{a}_\eta)^T M_{22} v \end{bmatrix} \tag{315}$$

Where

$$(M_{11})_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} G_{11}(\xi_r,\eta_s)h_i(\xi_r)e_j(\eta_s)h_k(\xi_r)e_l(\eta_s)w_r w_s, \tag{316}$$

$$(M_{12})_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} G_{12}(\xi_r,\eta_s)e_i(\xi_r)h_j(\eta_s)h_k(\xi_r)e_l(\eta_s)w_r w_s, \tag{317}$$

$$(M_{21})_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} G_{21}(\xi_r,\eta_s)h_i(\xi_r)e_j(\eta_s)e_k(\xi_r)h_l(\eta_s)w_r w_s, \tag{318}$$

$$(M_{22})_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} G_{22}(\xi_r,\eta_s)e_i(\xi_r)h_j(\eta_s)e_k(\xi_r)h_l(\eta_s)w_r w_s \tag{319}$$

with ξ_r and η_s the locations of the of the numerical integration nodes and w_r and w_s the integration weights.

B.3 Mass Matrices for 2-forms

Again, the mapping of the forms in the inner product can be achieved through

$$(\tilde{a}, a)_\Omega = \int_{\hat{\Omega}} \Phi^*[\tilde{a} \wedge \star a] \, \mathrm{d}\Omega \tag{320}$$

$$= \int_{\hat{\Omega}} \hat{\tilde{a}} \wedge \hat{\star}\hat{a} \, \mathrm{d}\Omega \tag{321}$$

With $\hat{a} = \Phi^* a$ and $\hat{\star} = \Phi^* \star \Phi^{-*}$, Then $\Phi^* \star \Phi^{-*}$ can be constructed by first considering

$$\Phi^{-*}\hat{a}^{(2)} = \bar{a}\Phi^{-*}(\mathrm{d}\xi) \wedge \Phi^{-*}(\mathrm{d}\eta) \tag{322}$$

$$= \bar{a}\left[\frac{\partial \xi}{\partial x}\mathrm{d}x + \frac{\partial \xi}{\partial y}\mathrm{d}y\right] \wedge \left[\frac{\partial \eta}{\partial x}\mathrm{d}x + \frac{\partial \eta}{\partial y}\mathrm{d}y\right], \tag{323}$$

$$= \bar{a}\left[\frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial y} - \frac{\partial \eta}{\partial x}\frac{\partial \xi}{\partial y}\mathrm{d}x \wedge \mathrm{d}y\right] \tag{324}$$

$$= \bar{a}|\Phi^{-*}|\mathrm{d}x \wedge \mathrm{d}y \tag{325}$$

$$= \bar{a}\frac{1}{|\Phi^*|}\mathrm{d}x \wedge \mathrm{d}y \tag{326}$$

$$\tag{327}$$

Applying the Hodge-$\star$ operation to the above will drop $\mathrm{d}x \wedge \mathrm{d}y$. Applying the pullback operation to the remaining function does nothing because $\Phi^* 1 = 1$. I thus leads to

$$\hat{\star}a^{(2)} = \bar{a}\frac{1}{|\Phi^*|} \tag{328}$$

expanding the 2-form and the test functions as

$$a^{(2)} = \sum_{i=1}^{p}\sum_{j=1}^{p} a_{i,j} e_i(\xi) e_k(\eta) \tag{329}$$

$$\tilde{a}^{(2)} = \sum_{k=1}^{p}\sum_{l=1}^{p} a_{k,l} e_k(\xi) e_l(\eta) \mathrm{d}\xi \wedge \mathrm{d}\eta \tag{330}$$

The inner product becomes

$$\int_\Omega \hat{\tilde{a}}^{(2)} \wedge \star \hat{\tilde{a}}^{(2)} = \int_\Omega \sum_{i=1}^{p}\sum_{j=1}^{p}\sum_{k=1}^{p}\sum_{l=1}^{p} a_{i,j} a_{k,l} \frac{1}{|\Phi^*(\xi,\eta)|} e_i(\xi) e_k(\eta) e_k(\xi) e_l(\eta) \mathrm{d}\xi \wedge \mathrm{d}\eta \tag{331}$$

Which is much more compact in the matrix form

$$\int_\Omega \hat{\tilde{a}}^{(2)} \wedge \star \hat{a}^{(2)} = \left(\hat{\tilde{a}}^{(2)}\right)^T \mathbb{M}^{(2)} \hat{a}^{(2)} \tag{332}$$

For which the elements of $\mathbb{M}^{(2)}$ can be computed as

$$M^{(2)}_{ijkl} = \sum_{r=1}^{p+1}\sum_{s=1}^{p+1} \frac{1}{J(\xi_r,\eta_s)} h_i(\xi_r) h_j(\eta_s) h_k(\xi_r) h_l(\eta_s) w_r w_s \tag{333}$$

where again $J(\xi,\eta)$ is the determinant of the Jacobian of the mapping, ξ_r and η_s are the locations of the of the numerical integration nodes and w_r and w_s are the integration weights.

C Additional results for the new Stokes formulation

C.1 Stretched Lid Driven Cavity

The transformed formulation from section 8 was tested on some stretched grids using the lid driven cavity boundary conditions. Firstly the grid was stretched 3x in only the vertical direction, creating a deep cavity flow. The Results can be found in Figure 58 and Figure 59. The result is as expected, the reference results show the same patterns for the flow, [40], even tough the symmetry of the stress tensor is not maintained.

C.2 Skewed Lid Driven Cavity

Additionally, the lid driven cavity has also been explored in a skewed cavity. The results are in Figure 60 and Figure 61. As can be seen in Figure 60 plot a lot of artifacts are present in the result. While this can be a result of the singularity in the top corners it is much more likely that this is the result of the conservation of angular momentum not being able to be resolved properly.

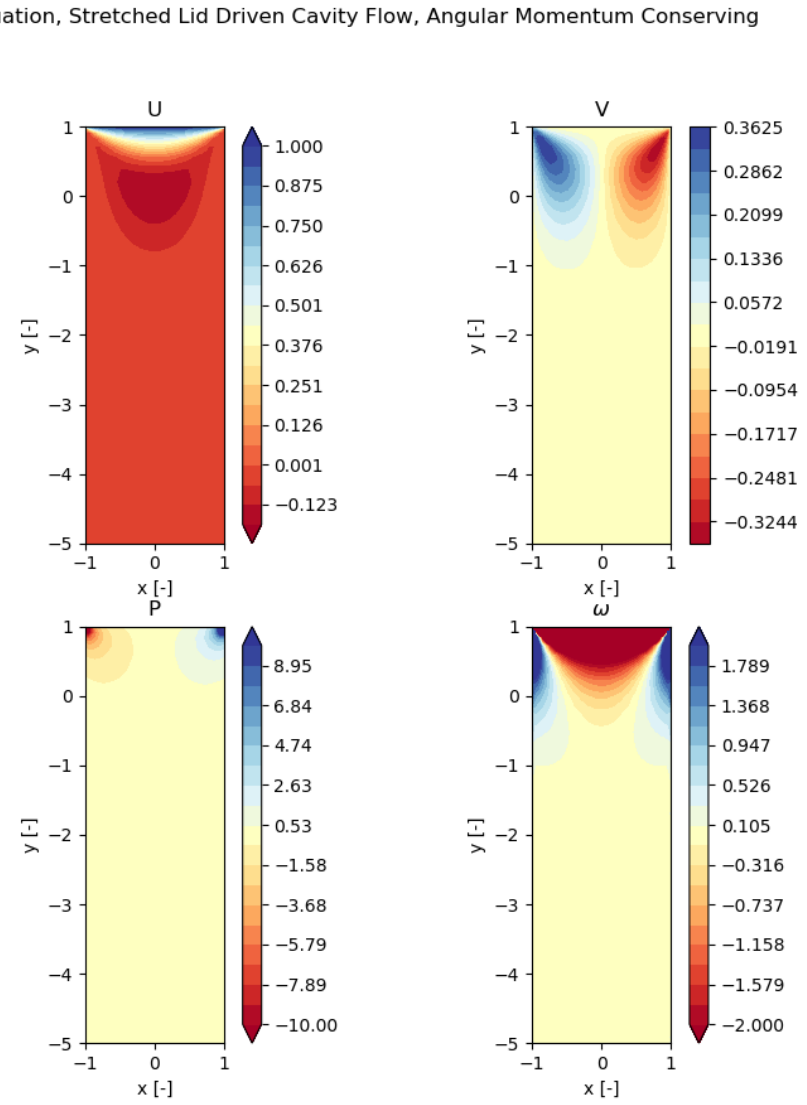

Figure 58: Velocity, pressure and vorticity in a vertically stretched lid driven cavity, N = 30

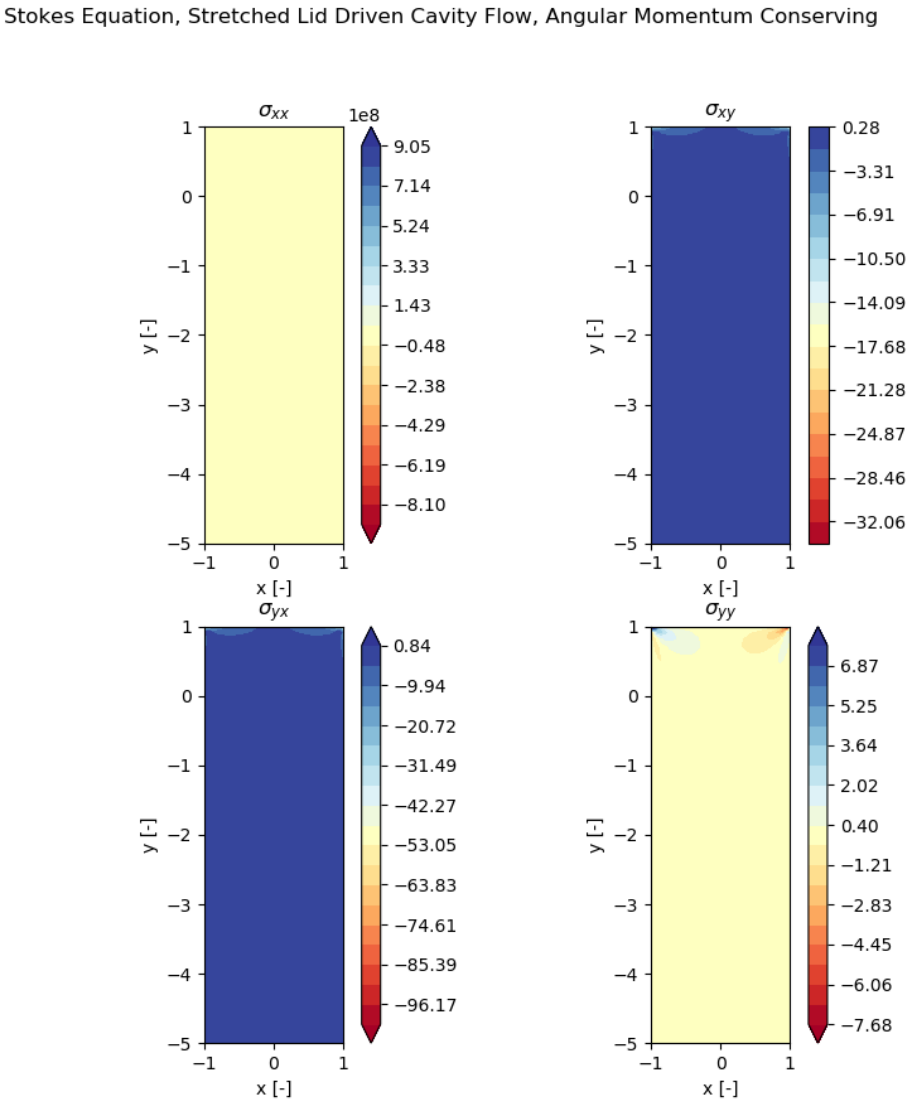

Figure 59: Stresses in a vertically stretched lid driven cavity, N = 30

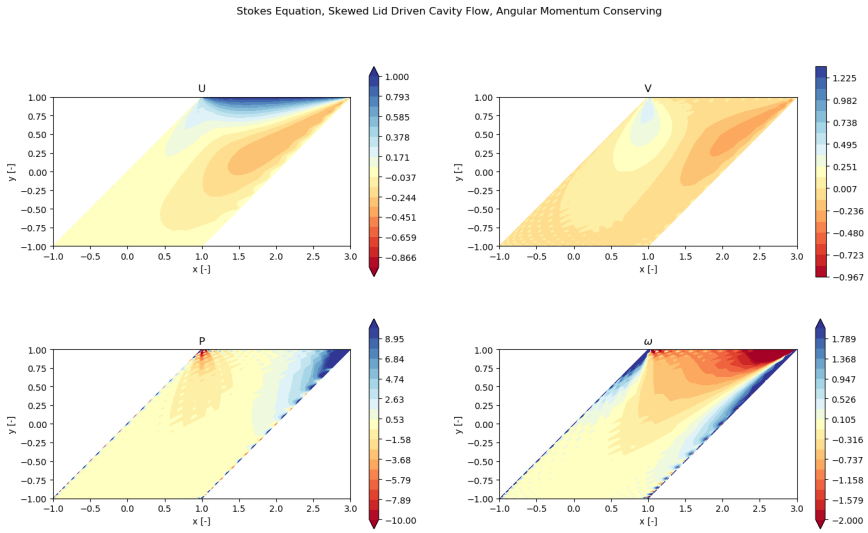

Figure 60: Velocity, pressure and vorticity in a Skewed lid driven cavity, N = 30

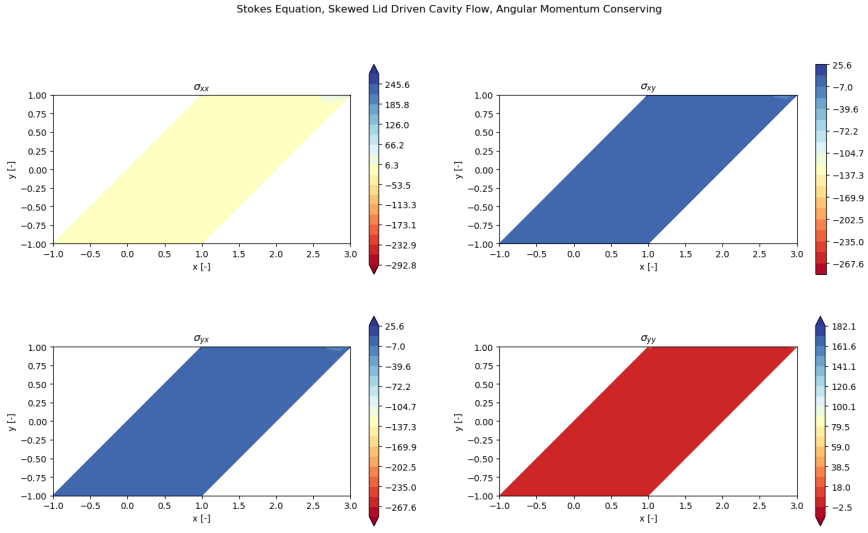

Figure 61: Stresses in a Skewed lid driven cavity, N = 30

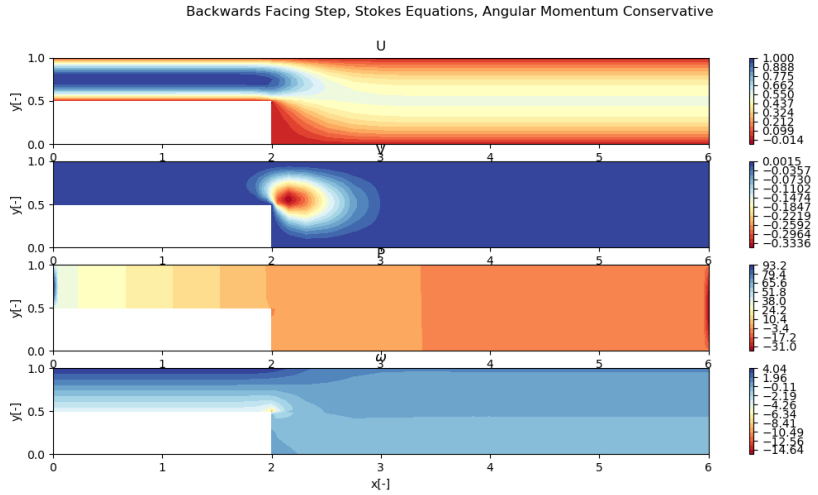

Figure 62: Velocity, pressure and vorticity for the new formulation in a backwards facing step. 5 Elements, N = 12

C.3 Backwards Facing Step using Stretched Elements

Finally, a backwards facing step was created with the new formulation. The results are very similar to the VVP formulation as can be seen in Figure 62. The backwards facing step has been cut off at x = 6.0 due to no significant changes being observed in the remaining part of the domain.