UiO : Department of Mathematics
University of Oslo

# Plane Wave Semi-Continuous Galerkin method for the Helmholtz equation

Anders Matheson

Master's Thesis, Spring 2015

**Abstract**

The Plane Wave Semi-Continuous Galerkin method is an example of a method where some of the expected structure of the solution is included in the finite element space. The idea is that this will lead to a more accurate method than the standard methods on problems where the solutions do have this structure.

First, this thesis establishes the necessary theory for partial differential equations. Next, some of the theory behind continuous and discontinuous Galerkin methods is established, emphasizing the difference in how these two methods handle the interfaces between elements. Using this, the semi-continuous nature of the Plane Wave Semi-Continuous Galerkin method is established.

Finally, the thesis provides a posteriori error estimates for the method, comparing it to the standard $Q_1$ method. In the provided result the method proves promising for methods having solutions behaving like plane waves locally.

## Acknowledgements

First, I would like to thank my supervisor Snorre Harald Christiansen for guiding me trough the entire process of writing this thesis. All the way from defining the problem to figuring out the last minute details, his input has been invaluable.

I would also like to thank Torquil Macdonald Sørensen for helping me with problems I encountered while implementing the numerical methods. Without his input, both figuring out some parts of the GetFEM++ library and debugging the implementation would have taken much longer, which would have made it much more difficult to finish the thesis on time.

Finally, I would like to thank my friends and family for helping me keep the motivation up throughout this year of writing. Without their support, the thesis would most likely never have been finished.

Oslo, 26 May, 2015
Anders Matheson

# Contents

# Chapter 1

# Introduction

Solving general partial differential equations is notoriously hard, both analytically and numerically. The solutions of these equations are often very complicated, and most of them are impossible to write out explicitly, and in many cases, even finding the most basic properties of the solutions is near impossible. There are classes of equations, however, for which we have extensive understanding of the behaviour. Probably not coincidentally, these are also often the same kind of equations that arise in many practical problems.

As with algebraic equations, differential equations can have any number of solutions, including both infinitely many and none. One major part of the analytical study of PDEs is finding the number and characteristic properties of solutions. Studying these properties can give vital insight, which can be used to understand the physical or abstract behaviour of the process the equation describes, even without actually solving the equation.

Even though we know there exists a solution to a particular partial differential equation, finding it can be a lot more difficult. Often the only option is to seek an approximate solution trough the use of a numerical method. These methods range from simple and intuitive methods such as the simplest finite difference schemes, to more abstract and sophisticated methods making use of the insight provided by the analytical studies. Common for all the methods is that they convert the continuous problem into a discrete problem which can be solved explicitly using only arithmetic operations which can be executed by computers. Loosing information in this process is unavoidable, and this is why such methods only generate approximations to solutions.

While analytical results alone can provide valuable insight in many practical problems, they also play a vital role when using numerical methods. For instance, trying to use a numerical method to approximate a solution that does not exist may cause problems. In many cases the method will, after a lot of calculation, detect that the equation has no solution and fail.

Other methods, however, lack the ability to detect failure and may return a solution, even if no solution actually exists. The situation can be just as bad when solving an equation which has an infinite number of solutions. Again, some methods will detect this and fail, but other methods may return a function which in some way approximates a solution locally, but it may be a different solution in different areas, resulting in a function which behaves nothing like a true solution.

One of the more popular methods utilizing insight gained trough the analytical studies of the equations is the finite element method, or FEM for short. It consists of splitting the solution into pieces, each defined over a small subset of the domain of definition of the equation. On each of these parts, we assume the solution has a particularly simple form, usually a linear combination of a finite number of predefined polynomials. We can then use Galerkin approximation to select a linear combination which closely approximates a solution of the equation, and the result is a method which can, for large classes of PDEs, be proven to give arbitrarily good approximations if we just make the discrete problem large enough.

While this seems to have pretty much solved our problem of finding solutions to PDEs, this is far from the case. In reality, computers, with their finite memory and speed, limit the size of the discrete problem. For many of the trickier problems, not even supercomputers can hope to find acceptable approximation in an acceptable amount of time using naive finite element methods.

This is the reason we need FEMs which are particularly good at solving just the kind of problem we want to solve. In this thesis we will introduce the Plane Wave Semi-Continuous Galerkin method, which is a finite element method designed to work particularly well with problems where the solutions have plane wave-like behaviour locally. This is done by replacing the polynomials used to approximate the solution on the elements with plane waves, the idea being that this will allow for more of the behaviour of the continuous problem to be preserved in the discrete problem.

## 1.1 About the thesis

The theory of PDEs includes a lot of results with long derivations, using different techniques from different fields of mathematics. Since the main focus of this thesis is to derive and test the Plane Wave Semi-Continuous Galerkin method, I will only derive and state the simplest versions of the results from PDE and Galerkin theory, since this will be enough to reason about the expected properties of the relevant Galerkin methods.

In chapter 2, we start by looking at the classical theory for some partial differential equations. This chapter lay the groundwork for our continued treatment of these equations by stating some basic results and introducing the notation which will be used later when discussing these equations.

In chapter 3, we will look at Galerkin methods and introduce the finite element framework at a rather low level, emphasizing how the methods behave on the intersections between elements since this is an important aspect of describing the semi-continuous nature of the plane wave method which is introduced in chapter 4.

Chapter 4 uses the notion of finite elements introduced in chapter 3 to derive the Plane Wave Semi-Continuous Galerkin method. In the second part of this chapter, we look at some of the high-level aspects of the implementation of the method.

Then in chapter 5, we look at how this method behaves numerically by running the implementation for some interesting cases, and making some a posteriori error estimates comparing the method to standard polynomial elements.

## 1.2 Code

A considerable part of this project was writing an implementation of the Plane Wave Semi-Continuous Galerkin method. The implementation is based on the GetFEM++ [10] finite element framework. Since understanding most of the code requires an understanding of the GetFEM++ library, I have chosen not to include any actual code in the thesis. Instead, I have included some more high-level notes on the implementation in the last part of chapter 4 and the results gained by running the code in chapter 5.

For readers interested in diving into the code, it is available on GitHub: `github.com/ANerd/PWSCG`.

# Chapter 2

# Introduction of relevant equations

Before we look at finite element methods, we need to introduce some of the classical theory of PDEs. In this chapter, we will first look at a concrete example of a PDE which will have a special role in deriving the plane wave-methods discussed later. Next we will look at a more general class of problems and develop the notation and some results which will be useful when working with the numerical methods.

## 2.1 Helmholtz equation

The methods described in this thesis will mostly have advantages for equations with solutions of wave-like form. As an example of an equation that has wave-like solutions, we look at the Helmholtz equation

$$\begin{cases} \Delta u + k^2 u = f & \text{in } \Omega \\ u = u_0 & \text{on } \partial\Omega \end{cases} \tag{2.1}$$

where $\Omega \in \mathbb{R}^d$ is of $C^1$ class [9, p. 710], $k : \Omega \to \mathbb{R}$, $u \in C^2(\Omega)$, $u_0 \in C^2(\partial\Omega)$ and $f \in C(\Omega)$.

### 2.1.1 Applications

The Helmholtz equation helps describing the behaviour of waves in multiple fields of physics, including acoustics, electromagnetic radiation and seismology. One way to arrive at Helmholtz equation is to look at the linear wave equation

$$\frac{\partial^2 u}{\partial t^2} = c^2 \Delta u \tag{2.2}$$

If we use separation of variables and assume

$$u(\boldsymbol{x}, t) = T(t)X(\boldsymbol{x})$$

where $T \in C^2(\mathbb{R})$ and $X \in C^2(\Omega)$, we get

$$\frac{1}{c^2}\frac{\partial^2 T}{\partial t^2}X = \Delta XT$$
$$\frac{1}{c^2 T}\frac{\partial^2 T}{\partial t^2} = \frac{\Delta X}{X}$$

and since the left hand and right hand side depend solely on $t$ and $\boldsymbol{x}$ respectively, they must both be constant.

$$\frac{1}{c^2 T}\frac{\partial^2 T}{\partial t^2} = c = \frac{\Delta X}{X}$$

We then find a $k \in \mathbb{R}$ such that $c = -k^2$ and we arrive at the homogeneous Helmholtz equation in $X(x)$

$$\frac{\Delta X}{X} = -k^2 \quad \Rightarrow \quad \Delta X + k^2 X = 0$$

The wave equation describes multiple physical phenomena. One example is the propagation of acoustic waves trough a 3 dimensional medium where $u$ represents the pressure in the medium. Other applications include waves on a 2 dimensional elastic membrane where $u$ then represents the displacement of the membrane in normal direction of the undisturbed membrane. Another describe the vibration of a string in 1 dimension, where again $u$ represents the displacement [4, p. 4]. There are also other equations in physics which can be reduced to Helmholtz equation, including the Schrödinger equation and some aspects of Maxwell's equations.

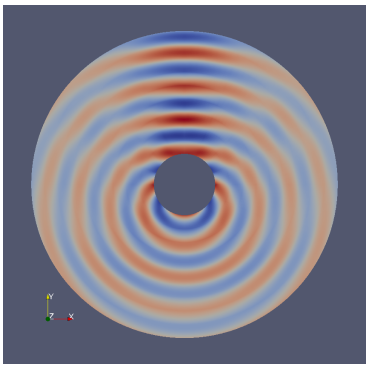


Figure 2.1: A solution of Helmholtz equation representing a wave bending around a circular obstacle.

### 2.1.2 Analytical solutions

The solutions of Helmholtz equation are in general very complex and can usually not be written explicitly. There are exceptions, however, some of which will be presented next.

### Plane waves

For the first solution we need $\boldsymbol{k}$ to be a constant vector. Let $\boldsymbol{k} \in \mathbb{R}^d$, and set $k$ from the equation such that $k^2 = \boldsymbol{k}^2$. If we then insert $u = e^{i\boldsymbol{k}\cdot\boldsymbol{x}}$ into the interior part of (2.1), we get

$$\begin{aligned} \Delta\left(e^{i\boldsymbol{k}\cdot\boldsymbol{x}}\right) + \boldsymbol{k}^2 e^{i\boldsymbol{k}\cdot\boldsymbol{x}} &= 0 \\ -\boldsymbol{k}^2 e^{i\boldsymbol{k}\cdot\boldsymbol{x}} + \boldsymbol{k}^2 e^{i\boldsymbol{k}\cdot\boldsymbol{x}} &= 0 \end{aligned}$$

which holds. This kind of function is known as a plane wave along $\boldsymbol{k}$. We also know that since $(-1)^2 = 1$ then $u = e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ must also be a solution. Since the equation is linear, any linear combinations of these solutions will also be solutions. Also, we may combine solutions with different directions of $\boldsymbol{k}$. While this may seem like a lot of flexibility, it is still not possible to satisfy all boundary conditions by linear combinations of these functions. Also, assuming constant $\boldsymbol{k}$ excludes a lot of useful solutions.

### Radial waves

Another interesting function is

$$u = \frac{e^{ikr}}{r}$$

where $r = |\boldsymbol{x} - \boldsymbol{x}_0|$ for some $\boldsymbol{x}_0 \in \mathbb{R}^d$. For this to be a solution of (2.1) we need to assume $\Omega \subset \mathbb{R}^3$ and that there exists a small neighborhood around $\boldsymbol{x}_0$ which is not included in $\Omega$. We also have to assume constant $k$. Using the Laplace operator in spherical coordinates [14, p. 111] we get

$$\begin{aligned} \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\left(\frac{e^{ikr}}{r}\right)\right) + k^2\frac{e^{ikr}}{r} &= \frac{1}{r^2}\frac{\partial}{\partial r}\left(e^{ikr}(ikr-1)\right) + k^2\frac{e^{ikr}}{r} \\ &= -k^2\frac{e^{ikr}}{r} + k^2\frac{e^{ikr}}{r} \\ &= 0 \end{aligned}$$

which is well formed since $\Omega$ does not include $x_0$.

Since this function is radial it can only satisfy boundary conditions which are also radial. An example of a problem which is solved by this kind of function is when $\Omega$ is on the form

$$\Omega = \{x \in \mathbb{R}^d \,:\, 0 < \theta < |x| < R\}$$

for some $\theta, R \in \mathbb{R}$ and the Dirichlet condition enforces a constant value on the inner boundary and another constant value on the outer boundary.

A way to relate this function to plane waves is to write

$$u = \frac{e^{i\boldsymbol{k}\cdot\boldsymbol{x}}}{|\boldsymbol{x}|}$$

with

$$\boldsymbol{k} = k\frac{\boldsymbol{x}}{|\boldsymbol{x}|}$$

What prevents this from being a true plane wave is that a plane wave has constant $\boldsymbol{k}$. One thing this form does provide is an indication that radial waves may have behaviour similar to plane waves locally.

## 2.2 Second-order elliptic boundary-value problems

To look at Galerkin methods, we must first establish some basic notation and results for the equations we seek to solve. A class of problems that usually works quite nicely with Galerkin methods are second-order elliptic boundary-value problems.

We will always assume the complex-valued functions unless otherwise specified. This means a function $v \in C(\Omega)$ will be $v : \Omega \to \mathbb{C}$ even tough one usually defines these functions to be $v : \Omega \to \mathbb{R}$. This also means we will use the complex $L^2$ inner products when constructing the weak forms. We denote the complex conjugate of $v$ as $\overline{v}$.

We will look at problems on the form

$$\begin{cases} Lu = f & \text{in } \Omega \\ u = u_0 & \text{on } \partial\Omega \end{cases} \tag{2.3}$$

for $\Omega \subset \mathbb{R}^d$, $f \in C(\Omega)$, $u_0 \in C^2(\partial\Omega)$ given, $u \in C^2(\Omega)$ the unknown and the operator $L$ defined as

$$Lu = -\operatorname{div}(\alpha(x)\operatorname{grad} u) + \boldsymbol{\beta}(x)\cdot\operatorname{grad} u + \gamma(x)u$$

where $\alpha \in [L^\infty(\Omega)]^{d\times d}$, $\boldsymbol{\beta} \in [L^\infty(\Omega)]^d$ and $\gamma \in L^\infty(\Omega)$ are known coefficient functions. We will also assume $\alpha(x)$ is symmetric.

**Definition 2.1** (Strong solution)**.** Assume $\alpha \in [C^1(\Omega)]^{d\times d}$, $\boldsymbol{\beta} \in [C(\Omega)]^d$ and $\gamma \in C(\Omega)$. If (2.3) holds for some $u \in C^2(\Omega)$ then $u$ is a *strong solution* of (2.3).

Second-order means that the highest order derivatives of $u$ included in the equation is second derivatives, and boundary-value problem is a problem defined by an equation on a domain and some condition on the behaviour on the boundary. Both of these properties are implicit in the definition of (2.3), but ellipticity needs to be defined explicitly

**Definition 2.2** (Ellipticity)**.** The partial differential operator $L$ is *elliptic* if there exist a constant $\theta > 0$ such that

$$\zeta^T \alpha(x) \zeta > \theta \left|\zeta\right|^2 \tag{2.4}$$

for all $\zeta \in \mathbb{R}^d$ and almost every $x \in \Omega$.

**Corollary 2.3.** *The Helmholtz equation is a second-order elliptic boundary value-problem.*

*Proof.* We can write the Helmholtz equation (2.1) on the form (2.3) with $\alpha(x) = I$, $\boldsymbol{\beta}(x) = 0$ and $\gamma(x) = k^2$, which means it's a second order boundary-value problem, and since

$$\zeta^T \alpha(x) \zeta = \zeta^T I \zeta = \left|\zeta\right|^2 > \theta \left|\zeta\right|^2$$

for any $\theta < 1$, it is also elliptic. □

The form (2.3) is called the *strong form* of the equation, and for it to be well formed we need $u \in C^2(\Omega)$ which leads to $f \in C(\Omega)$. This is a strong requirement which turns out to exclude many useful cases. This is why we in the next section introduce another form of the equation.

### 2.2.1 Weak form

We will now introduce the weak form of the problem. A way to handle inhomogeneous Dirichlet boundary conditions will be presented in section 2.2.3, but for now we will assume $u = 0$ on $\partial\Omega$. To derive the weak form of (2.3) we multiply it by the complex conjugate of $v \in C_0^\infty(\Omega)$ and integrate over $\Omega$. This gives us

$$\int_\Omega \alpha(x) \operatorname{grad} u \cdot \overline{\operatorname{grad} v} \,\mathrm{d}x + \int_\Omega \boldsymbol{\beta}(x) \cdot \operatorname{grad} u \overline{v} \,\mathrm{d}x + \int_\Omega \gamma(x) u \overline{v} \,\mathrm{d}x = \int_\Omega f \overline{v} \,\mathrm{d}x + \int_{\partial\Omega} \operatorname{grad} u \cdot n \overline{v} \,\mathrm{d}s$$

and since $v|_{\partial\Omega} = 0$ the boundary term disappears. Using this formulation as a starting point, we can make another definition of what it means to solve (2.3)

**Definition 2.4** (Weak solution)**.** $u \in H_0^1(\Omega)$ is a *weak solution* of (2.3) if

$$\int_\Omega \alpha(x) \operatorname{grad} u \cdot \overline{\operatorname{grad} v} \, \mathrm{d}x + \int_\Omega \boldsymbol{\beta}(x) \cdot \operatorname{grad} u \overline{v} \, \mathrm{d}x + \int_\Omega \gamma(x) u \overline{v} \, \mathrm{d}x = \langle f, \overline{v} \rangle \quad (2.5)$$

for all $v \in H_0^1(\Omega)$, where $\alpha \in [L^\infty(\Omega)]^{d \times d}$, $\boldsymbol{\beta} \in [L^\infty(\Omega)]^d$, $\gamma \in L^\infty(\Omega, \mathbb{R})$, $f \in H^{-1}(\Omega)$ and $\langle \cdot, \cdot \rangle$ is the pairing of $H^{-1}(\Omega)$ and $H_0^1(\Omega)$.

From this definition we introduce the bilinear and linear forms

$$\begin{aligned} a(u, v) &= \int_\Omega \alpha(x) \operatorname{grad} u \cdot \overline{\operatorname{grad} v} \, \mathrm{d}x + \int_\Omega \boldsymbol{\beta}(x) \cdot \operatorname{grad} u \overline{v} \, \mathrm{d}x + \int_\Omega \gamma(x) u \overline{v} \, \mathrm{d}x \\ l(v) &= \langle f, \overline{v} \rangle \end{aligned}$$

which give us the shorthand; find $u \in H_0^1(\Omega)$ such that

$$a(u, v) = l(v) \quad \forall v \in H_0^1(\Omega) \quad (2.6)$$

This form is called the *weak form* of the problem. Here, the requirement that $u \in C^2(\Omega)$ is replaced by the much weaker $u \in H_0^1(\Omega)$, and allowing $f \in H^{-1}(\Omega)$ means we have a well formed problem even for a very irregular $f$. In this form we call $u$ the *trial function* and $v$ the *test function*. This form also fits very nicely into the framework of Galerkin methods which will be presented in the next chapter.

Since $C_0^2(\Omega) \in H_0^1(\Omega)$, we have the following relation between strong and weak solutions

**Proposition 2.5.** *Assume $\alpha \in [C^1(\Omega)]^{d \times d}$, $\boldsymbol{\beta} \in [C(\Omega)]^d$ and $\gamma \in C(\Omega)$, $f \in C(\Omega)$. Then for $u \in C_0^2(\Omega)$ the following are equivalent*

*(i) $u$ is a* strong solution *of* (2.3)

*(ii) $u$ is a* weak solution *of* (2.3)

*Proof.* Assume $u$ is a strong solution. We multiply (2.3) by any test function $v \in C_0^\infty(\Omega)$, and integrate both sides of the equation over $\Omega$.

$$\int_\Omega Lu \overline{v} \, \mathrm{d}x = \int_\Omega f \overline{v} \, \mathrm{d}x$$

Since $u \in C_0^2(\Omega)$ we can perform the integration by parts without introducing boundary terms

$$\int_\Omega - \operatorname{div} (\alpha(x) \operatorname{grad} u) \, \overline{v} \, \mathrm{d}x = \int_\Omega \alpha(x) \operatorname{grad} u \cdot \operatorname{grad} \overline{v} \, \mathrm{d}x$$

Since this holds for any $v \in C^\infty(\Omega)$, it must also hold in the closure. Hence

$$a(u, v) = l(v) \quad \forall v \in H_0^1(\Omega)$$

and $u$ is a weak solution. To prove the converse we use the same steps in reverse to arrive at

$$\int_\Omega Lu\overline{v}\,\mathrm{d}x = \int_\Omega f\overline{v}\,\mathrm{d}x \quad \forall v \in C_0^\infty(\Omega)$$

and since this holds for all $v \in C_0^\infty(\Omega)$, we know that that $Lu = f$, hence $u$ is a strong solution. □

### 2.2.2 Existence of solutions

By using the weak form we can now make a sufficient condition for the existence of an unique solution. While this condition turns out to be too strict for many problems, it gives insight into what kind of properties well formed problems should have. It is also used as a starting point to develop more sophisticated existence theorems.

**Theorem 2.6** (Lax-Milgram)**.** *Let $H$ be a Hilbert space and assume*

$$a : H \times H \to \mathbb{R}$$
$$l : H \to \mathbb{R}$$

*are linear functionals. Then, if there exists $c_1, c_2, c_3 > 0$ such that*

*(i)* $|a(u, v)| \leq c_1\|u\|\|v\| \quad \forall u, v \in H$ *(continuity of $a$)*

*(ii)* $a(u, u) \geq c_2\|u\|^2 \quad \forall u \in H$ *(coercivity of $a$)*

*(iii)* $|l(v)| \leq c_3\|v\|^2 \quad \forall v \in H$ *(continuity of $l$)*

*then there exists a unique element $u \in H$ such that*

$$a(u, v) = l(v) \quad \forall v \in H$$

*Proof.* Let $(\cdot, \cdot)$ be an inner product over $H$. For any $u \in H$ the mapping $v \mapsto a(u, v)$ is a bound linear functional. From Riesz Representation Theorem [9, p. 722] we know there exists an unique element $w \in H$ such that

$$a(u, v) = (w, v) \quad \forall v \in H$$

and we write $Au = w$ such that

$$a(u, v) = (Au, v) \quad \forall v \in H \tag{2.7}$$

First we show that $A : H \to H$ is linear. For any $v \in H$ we have

$$\begin{aligned}(A(\lambda_1 u_1 + \lambda_2 u_2), v) &= a(\lambda_1 u_1 + \lambda_2 u_2, v) \quad \text{from (2.7)} \\ &= \lambda_1 a(u_1, v) + \lambda_2 a(u_2, v) \quad \text{by linearity of } a \\ &= \lambda_1 (Au_1, v) + \lambda_2 (Au_2, v) \quad \text{by (2.7) again} \\ &= (\lambda_1 Au_1 + \lambda_2 Au_2, v) \quad \text{by linearity of the inner product}\end{aligned}$$

Since this holds for all $v \in H$, we know $A$ is linear. Furthermore

$$\begin{aligned}\|Au\|^2 &= (Au, Au) \\ &= a(u, Au) \quad \text{from (2.7)} \\ &\leq c_1 \|u\| \|Au\| \quad \text{from property (i) in the theorem}\end{aligned}$$

and hence $\|Au\| \leq c_1 \|u\|$ and $A$ is bounded. Next we observe that property (ii) gives us

$$c_2 \|u\|^2 \leq a(u, u) = (Au, u) \leq \|Au\| \|u\|$$

hence $c_2 \|u\| \leq \|Au\|$ which implies the two properties

$$\begin{cases} A \text{ is injective} \\ \text{The range of } A \text{ (denoted im}A\text{) is closed in H} \end{cases}$$

Using this we can prove that

$$\text{im}A = H$$

by contradiction. Since im$A$ is closed there would exist a nonzero element $w \in H$ with $w \in \text{im}A^{\perp}$, but since

$$c_2 \|w\|^2 \leq a(w, w) = (Aw, w) = 0$$

this is a contradiction. Lastly, from property (iii) in the theorem, $l$ is a bounded linear functional and we can use Riesz Representation to find the unique $w \in H$ such that $(w, v) = l(v)$ $\forall v \in H$. Since $A$ is bijective, there exists exactly one $u \in H$ such that $Au = w$ and this gives us

$$l(v) = (w, v) = (Au, v) = a(u, v) \quad \forall v \in H$$

□

Since $H_0^1$ is a Hilbert space, Lax-Milgram gives a sufficient condition for existence and uniqueness of solutions of (2.6) given the three properties. It turns out, however, it is not that simple. The first and third property holds since

$$\begin{aligned} a(u,v) &\leq \|\alpha\|_{L^\infty(\Omega)} \int_\Omega |\operatorname{grad} u|\,|\operatorname{grad} v|\,\mathrm{d}x \\ &\quad + \|\boldsymbol{\beta}\|_{L^\infty(\Omega)} \int_\Omega |\operatorname{grad} u|\,|v|\,\mathrm{d}x + \|\gamma\|_{L^\infty(\Omega)} \int_U |u|\,|v|\,\mathrm{d}x \\ &\leq \|\alpha\|_{L^\infty(\Omega)} \|\operatorname{grad} u\|_{L^2(\Omega)} \|\operatorname{grad} v\|_{L^2(\Omega)} \\ &\quad + \|\boldsymbol{\beta}\|_{L^\infty(\Omega)} \|\operatorname{grad} u\|_{L^2(\Omega)} \|v\|_{L^2(\Omega)} + \|\gamma\|_{L^\infty(\Omega)} \|u\|_{L^2(\Omega)} \|v\|_{L^2(\Omega)} \\ &\leq c_1 \|u\|_{H_0^1(\Omega)} \|v\|_{H_0^1(\Omega)} \end{aligned} \tag{2.8}$$

and

$$l(v) = \langle f, \overline{v} \rangle \leq \|f\|_{H^{-1}(\Omega)} \|v\|_{H^1(\Omega)} \leq c_3 \|v\|_{H^1(\Omega)} \tag{2.9}$$

but if we try to verify coercivity we get

$$\begin{aligned} \theta \int_\Omega |\operatorname{grad} u|^2\,\mathrm{d}x &\leq \int_\Omega \alpha(x) \operatorname{grad} u \cdot \overline{\operatorname{grad} u}\,\mathrm{d}x \quad \text{from ellipticity (2.4)} \\ &\leq a(u,u) - \int_\Omega (\boldsymbol{\beta}(x) \cdot \operatorname{grad} u\overline{u} + \gamma(x) u\overline{u})\,\mathrm{d}x \\ &\leq a(u,u) + \|\boldsymbol{\beta}\|_{L^\infty(\Omega)} \int_\Omega |\operatorname{grad} u|\,|u|\,\mathrm{d}x + \|\gamma\|_{L^\infty(\Omega)} \|u\|_{L^2(\Omega)}^2 \end{aligned}$$

and from Cauchy's inequality with $\epsilon$ we have

$$\int_\Omega |\operatorname{grad} u|\,|u|\,\mathrm{d}x \leq \epsilon \int_\Omega |\operatorname{grad} u|^2\,\mathrm{d}x + \frac{1}{4\epsilon} \int_\Omega |u|^2\,\mathrm{d}x$$

and choosing $\epsilon > 0$ such that

$$\epsilon \|\boldsymbol{\beta}\|_{L^\infty(\Omega)} < \frac{\theta}{2}$$

gives

$$\frac{\theta}{2} \int_\Omega |\operatorname{grad} u|^2 \leq a(u,u) + C\|u\|_{L^2(\Omega)}$$

and from Poincaré's inequality we can make the semi norm on the left side into a full norm for appropriate constants $c_2, c_3 > 0$ such that

$$c_2 \|u\|_{H_0^1(\Omega)}^2 \leq a(u,u) + c_3 \|u\|_{L^2(\Omega)}^2$$

which is the closest we get, but not exactly what we need to use the Lax-Milgram theorem. Existence and uniqueness for the general second order

elliptic equation can be shown using that the highest order term in $a$ is coercive and that the lower order terms can be interpreted as a compact perturbation of this. Using the Fredholm theory for compact operators we get insight into what is needed of the problem for it to have a unique solution [9, p. 321]. The proof of this is rather long and outside the scope of this introduction.

Helmholtz equation is not coercive and consequently is not covered by Lax-Milgram. If we instead look at the equation where the sign of the terms in (2.1) are opposite, we get a unique solution. This equation arises from looking at the spatial part of the heat equation.

**Corollary 2.7.** *The equation*

$$\begin{cases} -\Delta u + k^2 u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{2.10}$$

*has a unique weak solution for all* $f \in H^{-1}(\Omega)$.

*Proof.* The weak form of (2.10) becomes

$$\begin{aligned} a(u,v) &= \int_\Omega \operatorname{grad} u \cdot \overline{\operatorname{grad} v}\,\mathrm{d}x + \int_\Omega \left(k(x)^2 u\overline{u}\right)\,\mathrm{d}x \\ l(v) &= \int_\Omega fv\,\mathrm{d}x \end{aligned}$$

Since property (i) and (iii) of the Lax-Milgram theorem is always satisfied, we only need to show property (ii), the coercivity of $a$.

$$\begin{aligned} \theta \int_\Omega |\operatorname{grad} u|^2\,\mathrm{d}x &\leq \int_\Omega \operatorname{grad} u \cdot \overline{\operatorname{grad} u}\,\mathrm{d}x \\ &\leq a(u,u) - \int_\Omega k(x)^2 u\overline{u}\,\mathrm{d}x \\ &\leq a(u,u) - \|ku\|^2_{L^2(\Omega)} \\ &\leq a(u,u) \end{aligned}$$

and using Poincaré's inequality we get the result. □

### 2.2.3 Boundary function

When introducing weak solutions we assumed the Dirichlet boundary condition to be homogeneous ($u = 0$ on $\partial\Omega$). One of the advantages of this is that the $H^1$ seminorm is equivalent to the full $H^1$ norm. This is especially useful

when showing coercivity since ellipticity of second order equations can be used to bound the seminorm of the function by the bilinear form applied to the function. One way to use the same analysis as we did above on problems with non-homogeneous Dirichlet boundary is to use boundary functions. The weak formulation of problem (2.3) with inhomogeneous Dirichlet boundary condition is to find $u \in H^1(\Omega)$ such that

$$\begin{cases} a(u,v) = l(v) & \forall v \in H^1(\Omega) \\ Tu = u_0 \end{cases} \tag{2.11}$$

for $u_0 \in H^{1/2}(\partial\Omega)$ where $T$ is the trace operator [9, p. 272]. The idea of boundary functions is to find a function $u_b$ such that

$$u - u_b \in H_0^1(\Omega)$$

and then use $u_{\text{int}} = u - u_b$ as the unknown in a problem with homogeneous Dirichlet conditions.

When proving existence of solution of the weak formulation with homogeneous Dirichlet conditions (2.6) we did not consider the existence of functions satisfying the boundary condition since $H_0^1(\Omega)$ obviously contains functions which satisfy $Tu = 0$. Now, however, we have to show that for any $u_0 \in H^{1/2}(\partial\Omega)$ there exists a function $u \in H^1(\Omega)$ such that $Tu = u_0$. To show this we need a result from functional analysis [8, p. 130]

**Proposition 2.8.** *Let $\Omega$ be a $C^1$ class open set; then the image of the trace map on $W^{1,p}(\Omega)$ satisfies*

$$T\left(W^{1,p}(\Omega)\right) = W^{1-1/p,p}(\partial\Omega)$$

Here the notation $T(X)$ means $T(X) = \text{im}T$ when $T : X \to Y$. Since we want $u_b \in H^1(\Omega)$ we use that

$$T\left(H^1(\Omega)\right) = H^{1/2}(\partial\Omega)$$

and we can formulate the needed result.

**Corollary 2.9.** *For any $u_0 \in H^{1/2}(\partial\Omega)$ we can find a function $u_b \in H^1(\Omega)$ such that $Tu_b = u_0$.*

*Proof.* The result follows directly from proposition 2.8. □

We can now transform problem (2.11) to a form where we will be able to apply the analysis from the previous sections. We find $u_b$ such that $Tu_b = u_0$ and set $u_{\text{int}} = u - u_b$. The equation from (2.11) then becomes

$$\begin{aligned} a(u_{\text{int}} + u_b, v) &= l(v) \\ a(u_{\text{int}}, v) &= l(v) - a(u_b, v) \end{aligned}$$

and by defining $\widehat{a} : H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}$ as the restriction of $a$ and

$$\widehat{l} : \begin{cases} H_0^1(\Omega) \to \mathbb{R} \\ v \mapsto l(v) - a(u_b, v) \end{cases}$$

we have reduced the problem to finding $u_{\text{int}} \in H_0^1(\Omega)$ such that

$$\widehat{a}(u_{\text{int}}, v) = \widehat{l}(v) \quad \forall H_0^1(\Omega) \tag{2.12}$$

which is on the form we have studied. The solution of (2.11) will then be $u = u_{\text{int}} + u_b$. Now we can state a existence and uniqueness result which do not require coercivity on all of $H^1(\Omega)$.

**Proposition 2.10.** *Let $a$ from problem* (2.11) *restricted to $H_0^1(\Omega)$ be coercive. Then* (2.11) *has a unique solution.*

*Proof.* First we show existence. Let $l$ and $a$ be the linear and bilinear form from (2.11). We have from (2.8) and (2.9) that $a$ and $l$ are bounded on $H^1(\Omega)$. This implies that the mapping $v \mapsto l(v) - a(u, v)$ is bounded for all $u \in H^1(\Omega)$, hence $\widehat{l}$ from (2.12) is continuous. $\widehat{a}$ is the restriction of $a$ to $H_0^1(\Omega)$ which we assumed to be coercive, and it is obviously also continuous. Hence, we know from theorem 2.6 that (2.12) has a unique solution for every $u_b \in H^1$. Since we know from corollary 2.9 that a suitable $u_b$ can always be found, we know we always have a solution.

To show uniqueness we assume $u_1$ and $u_2$ are two solutions of (2.11). Then

$$T(u_1 - u_2) = Tu_1 - Tu_2 = u_0 - u_0 = 0$$

hence $(u_1 - u_2) \in H_0^1(\Omega)$. This means we can use coercivity of $a$

$$\begin{aligned} c\|u_1 - u_2\|_{H^1(\Omega)} &\leq a(u_1 - u_2, u_1 - u_2) \\ &\leq a(u_1, u_1 - u_2) - a(u_2, u_1 - u_2) \\ &\leq l(u_1 - u_2) - l(u_1 - u_2) \\ &\leq 0 \end{aligned}$$

since both functions are solutions. Hence, $u_1 = u_2$ and we have uniqueness. □

# Chapter 3

# Galerkin methods

In this chapter, we will look at Galerkin methods for second-order elliptic boundary-value problems. First we will look at the standard conforming Galerkin method which poses restrictions on the finite function spaces used by the method, making the calculations easier from both an analytical and numerical point of view. Next we will look at discontinuous Galerkin methods which do not impose the same requirements, gaining flexibility at the cost of complexity.

## 3.1 Conforming Galerkin methods

Conforming Galerkin methods are usually the easiest and most suitable methods to use on well-behaved problems. They are derived from the weak formulation (2.6) of the problem by limiting the function spaces of the test and trial functions to finite function spaces. We also require the space of trial functions to be the same as the space of test functions. Methods using different spaces for test and trial functions are called Petrov-Galerkin methods [4, p. 54], and will not be covered in this thesis.

A method is *conforming* if the finite function spaces used for test and trial functions are subspaces of the definition spaces of the bilinear form $a$ of the weak formulation. This ensures we can insert the test and trial functions directly into the weak formulation, which is required for the derivation shown here. Non-conforming methods give more flexibility to solve difficult problems, but require more care to ensure the discrete formulation is well posed.

### 3.1.1 Derivation

Deriving the Galerkin method is rather straightforward. Let $\Omega$ be the domain on which the problem is defined. First, we choose a finite dimensional space $X_h \subset H_0^1(\Omega)$, a basis $\text{span}\{\phi_i\} = X_h$, and let $m = \dim X_h$. If we then write the weak formulation (2.6), but instead of using $u, v \in H_0^1$ we use $u_h, v_h \in X_h$, we get the discrete problem of finding $u_h \in X_h$ such that

$$a(u_h, v_h) = l(v_h) \quad \forall v_h \in X_h \tag{3.1}$$

Since $X_h \subset H_0^1$, we know this problem is well posed. Now, since $X_h$ has a finite basis, we can write

$$u_h = \sum_{i=1}^{m} \phi_i c_i$$

and make $m$ equations, one for each $v_h = \phi_j$

$$a(\sum_{i=1}^{m} \phi_i c_i, \phi_j) = l(\phi_j) \quad \forall j \in [1, m]$$

and since $a$ is linear this can be written

$$\sum_{i=1}^{m} a(\phi_i, \phi_j) c_i = l(\phi_j) \quad \forall j \in [1, m]$$

which is a set of linear equations. Written in matrix form for this becomes

$$\begin{bmatrix} a(\phi_1, \phi_1) & a(\phi_2, \phi_1) & \cdots & a(\phi_m, \phi_1) \\ a(\phi_1, \phi_2) & a(\phi_2, \phi_2) & \cdots & a(\phi_m, \phi_2) \\ \vdots & \vdots & \ddots & \vdots \\ a(\phi_1, \phi_m) & a(\phi_2, \phi_m) & \cdots & a(\phi_m, \phi_m) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} = \begin{bmatrix} l(\phi_1) \\ l(\phi_2) \\ \vdots \\ l(\phi_m) \end{bmatrix}$$

which can be solved by numerical methods from linear algebra.

Deriving the method can be done with very little restriction on $X_h$ or the basis used, but while deriving the method is simple, proving it will result in a good approximation to the problem requires us to be more specific about the properties of $X_h$. This is what the elements in the finite element method provide.

### 3.1.2 Finite element methods

The choice of finite element space $X_h$ greatly affects the properties of the method. While doing analysis separately for each $X_h$ is possible, it may not

be very effective. There are some assumptions we can do which makes it possible to generalize some of the analysis to a wide class of methods, only leaving out the parts unique to each method. To analyze a finite element method we first need to define the elements.

### Finite element

In the most general sense, an element can be defined as a triple $(T, X_T, \Sigma_T)$ [4, p. 70].

**Definition 3.1.** A *finite element* is a triple $(T, X_T, \Sigma_T)$ where $T$ is a closed domain, $X_T \subset C(T)$ is a space of continuous functions with $\dim X_T = m_T$, and $\Sigma_T = \{\sigma_T^i\}_{i\in[1,m_T]}$ is an indexed family of linear functionals on $X_T$ called the local *degrees of freedom* on the element. We also require the mapping

$$D : \begin{cases} X_T \to \mathbb{R}^{m_T} \\ \quad v \mapsto [\sigma_T^i(v)]_{i\in[1,m_T]} \end{cases}$$

to be bijective.

We will often use the abbreviation dof for *degrees of freedom*.

A finite element method then consists of defining $n$ elements $\{(T_r, X_{T_r}, \Sigma_{T_r})\}_{r\in[1,n]}$ in such a way that

(i) $\overline{\Omega} = \bigcup_{r=1}^{n} T_r$

(ii) $\dim(T_r \cap T_s) < \dim \Omega \quad \forall r, s \in [1, n],\ r \neq s$

(iii) $X_h = \{u \in C(\Omega) \,:\, r \in [1, n],\ \ u|_{T_r} \in X_{T_r}\}$

where $\overline{\Omega}$ denotes the closure of $\Omega$.

We name the set $\mathcal{T} = \{T_r\}_{r\in[1,n]}$ the *mesh* of the finite element method. Note that we require functions in $X_h$ to be continuous. Since this restricts how we can combine functions from different elements it will impact how we construct the global degrees of freedom. This is where continuous Galerkin diverges from discontinuous Galerkin, which will be discussed in further detail later in this chapter.

Requirement (i) may be hard to accommodate. For example, if $\Omega$ is a circle and $\mathcal{T}$ is a set of triangles, it will be impossible to satisfy this requirement with a finite number of elements. While this may introduce an additional approximation error, it is usually ignored by assuming $\Omega$ can be written as the union of the element domains.

The finite element method also needs an indexed family of global degrees of freedom. We call this set $\Sigma_h = \{\sigma^i\}_{i\in[1,m]}$ and define the mapping

$$D_h : \begin{cases} X_h \to \mathbb{R}^m \\ v \to [\sigma^i(v)]_{i\in[1,m]} \end{cases}$$

which we require to be bijective. The most natural way of defining $\Sigma_h$ would be to include all the local degrees of freedom

$$\sigma \in \Sigma_h \Leftrightarrow \sigma(v) = \sigma_T(v|_T) \text{ where } \sigma_T \in \Sigma_T \text{ for some } T \in \mathcal{T}$$

which would imply $m = \sum_{T\in\mathcal{T}} m_T$, but then in general, the resulting $D_h$ would not be surjective. To show this we first introduce the set

$$\Gamma = \bigcup_{T\in\mathcal{T}} \partial T$$

and introduce the function $v : \Omega \setminus \Gamma \to \mathbb{R}$ on the form

$$v(x) = \begin{cases} v_0(x) & x \in T_0 \setminus \partial T_0 \\ v_1(x) & x \in T_1 \setminus \partial T_1 \\ \vdots & \\ v_n(x) & x \in T_n \setminus \partial T_n \end{cases}$$

where $v_i \in X_{T_i}$ $i \in [1, n]$. Since $D_{T_i}$ is bijective, we can uniquely identify $v_i$ by $d_i = D_{T_i} v_i$, so given the values of $d_i$ for all $i \in [1, n]$ we can uniquely identify $v$. Next we let $e = T_r \cap T_s$ for $r, s \in [1, n]$ such that $e \neq \emptyset$, and let $x_0 \in e$. Since we choose the values of $d_r$ and $d_s$ independently, we can make a case where

$$\lim_{\substack{x\to x_0 \\ x\in T_r}} v(x) = v_r(x_0) \neq v_s(x_0) = \lim_{\substack{x\to x_0 \\ x\in T_s}} v(x)$$

hence, there are no $v_c \in C(\Omega)$ such that

$$v_c|_{\Omega\setminus\Gamma} = v$$

and since $X_h \subset C(\Omega)$, there are no $u \in X_h$ for which $D_h u$ will result in this set of dof values, and $D_h$ is not surjective. There are two ways to solve this, one is to extend $X_h$, the other is to restrict the dofs so that $\mathrm{im} D_h = \mathbb{R}^m$ for some $m < \sum_{T\in\mathcal{T}} m_T$. Discontinuous Galerkin, which is discussed later in this chapter, takes the first approach. For now, we will keep $X_h$ the same and take the second approach.

### 3.1.3 Degrees of freedom

We want to define an indexed family of global degrees of freedom $\Sigma_h$ which makes $D_h$ surjective trough reducing the set of all the local degrees of freedom to a smaller set which can only represent continuous functions. To do this we will pose some requirements on how dofs act on the boundaries of elements.

Let $T_r, T_s \in \mathcal{T}$ be element domains such that for $e = T_r \cap T_s$,

$$\dim(e) = \dim\Omega - 1$$

We call any such $e$ an interior edge. Let $v_r \in X_{T_r}$, $v_s \in X_{T_s}$ and let $v : X_{T_r} \cup X_{T_s} \setminus e \to \mathbb{R}$ be defined as

$$v(x) = \begin{cases} v_r(x) & x \in X_{T_r} \setminus e \\ v_s(x) & x \in X_{T_s} \setminus e \end{cases}$$

We will also need the space

$$X_{T_r,e} = \{v|_e : v \in X_{T_r}\}$$

Since $X_{T_q} \subset C(T_q)$ for all $T_q \in \mathcal{T}$, we know that $v$ is continuous everywhere except on $e$, and it will be possible to find a function $v_c \in C(T_r \cup T_s)$ such that $v_c|_{T_r \cup T_s \setminus e} = v$ if and only if

$$v_r(x_0) = v_s(x_0) \quad \forall x_0 \in e \tag{3.2}$$

To ensure this trough our degrees of freedom we must have a set $J_{T_r,e}$ of indices and a set of functionals $\sigma^j_{T_r,e} : X_{T_r,e} \to \mathbb{R}$ such that

$$\sigma^j_{T_r}(v_r) = \sigma^j_{T_r,e}(v_r|_e) \quad \forall j \in J_{T_r,e} \tag{3.3}$$

where the values of $\left[\sigma^j_{T_r,e}(v_r|_e)\right]_{j \in J_{T_r,e}}$ uniquely identify $v_r|_e$. We must also assume there exists a similar $J_{T_s,e}$ and a similar set of $\sigma^j_{T_s,e}$ for $T_s$ and a bijection $E : J_{T_r,e} \to J_{T_s,e}$ such that

$$\sigma^j_{T_r,e}(v_r|_e) = \sigma^{E(j)}_{T_s,e}(v_s|_e) \quad \forall j \in J_{T_r,e} \tag{3.4}$$

if and only if (3.2). Hence, enforcing (3.3) and (3.4) will ensure we can only represent continuous functions.

We may then define the indexed family of $m$ linearly independent global degrees of freedom

$$\Sigma_h = \left\{\sigma^i\right\}_{i \in [1,m]}$$

where

$$\sigma^i \in \left\{\sigma(v) : X_h \to \mathbb{R} \,:\, \sigma(v) = \sigma^j_{T_r}(v|_{T_r}) \;\; \forall \;\; j \in [1, m_{T_r}], r \in [1, n]\right\}$$

Finding a linearly independent subset might not be trivial in general, but using the assumptions we made on the degrees of freedom, we have a natural solution to this. Since we required the local dofs to be linearly independent on the element domain, two linearly dependent dofs must be from different elements. Since a local dof only depend on the function inside its own domain, this may only happen where two domains intersect, which is on the interior edges. From the assumptions above we have that if $e$ is an interior edge and $j \in J_{T_r,e}$ then

$$\sigma_i(v) = \sigma^j_{T_r}(v|_{T_r}) = \sigma^j_{T_r,e}(v|_e) = \sigma^{Ej}_{T_s,e}(v|_e) = \sigma^{Ej}_{T_s}(v|_{T_s}) = \sigma_k(v) \tag{3.5}$$

Hence, provided the mapping $E$, we have a trivial way to collapse local dofs into a linearly independent set of global dofs.

While a local dof can be indexed by the tuple $(r, j)$ where $r$ is the index of the element and $j$ is the index of the local dof, a global dof is indexed with a single integer $i$. We refer to the map $(r, j) \mapsto i$ as the *dof map*. Note that the dof map is surjective but will not be injective in general because of the way we collapse local dofs into global ones. This is not a problem though, since all local dofs mapping to the same global dof will yield the same linear mapping.

The introduction of elements does not restrict the choices of $X_h$ or the basis $u_h = \sum_{i=1}^m \phi_i c_i$ we used when deriving the Galerkin method. Given any $X_h \subset H^1_0(\Omega)$ we may assume a single element ($n = 1$), set $T_1 = \Omega$, $X_{T_1} = X_h$ and the degrees of freedom $\sigma^i_{T_1} = c_i$. Obviously, this does not give us any more insight. The problem of defining $X_h$ directly is that it is tightly coupled with the domain $\Omega$ and there is no general discretization parameter. By instead defining an element $(T, X, \Sigma)$, we can apply it to any domain by splitting it into elements. We will also have flexibility in the size of the elements we use, and by requiring

$$\operatorname{diam} T \leq h \quad \forall T \in \mathcal{T} \tag{3.6}$$

where $\operatorname{diam} T$ is the diameter of $T$, we have a general discretization parameter $h$, the goal being

$$\lim_{h \to 0} \|u - u_h\|_{H^1(\Omega)} = 0$$

where $u$ is a solution of (2.6) and $u_h$ a solution of (3.1). Whether this is true will be discussed in section 3.1.4. Whenever we write $\mathcal{T}_h$ we assume (3.6) holds for each $T \in \mathcal{T}_h$.

Assumption (3.4) gives us one more useful property. While we require function values to be uniquely defined by the dofs values on edges, nothing prevents us from making other quantities uniquely defined as well. We may for instance make the first derivatives uniquely defined which allows us to ensure $X_h \in C^1(\Omega)$.

To use the Galerkin method from the previous section we need two things; a space $X_h \in H_0^1(\Omega)$ with a basis $\{\phi_i\}_{i\in[1,m]}$. First, the $X_h$ derived here will be in $H^1(\Omega)$, but in general not in $H_0^1(\Omega)$. This is solved in different ways in different implementations and it will not be covered in this section. Two ways of solving this are presented in section 4.2.4. We can use any basis for $X_h$, but if we have defined the method through elements the most natural basis to use is $\{\phi_i\}_{i\in[1,m]}$ such that

$$v(x) = \sum_{i=1}^{m} \phi_i(x)\sigma^i(v)$$

### Nodal elements

We will now look at some common elements. One simple choice of dof is taking a point value of the function

**Definition 3.2.** If

$$\sigma_T^i(v) = v(x_T^i) \quad i \in [1, m_T]$$

for some $x_T^i \in T$, we say $\sigma_T^i$ is a *nodal* dof in the node $x_T^i$. Elements which only contain nodal dofs are called *nodal elements*.

For nodal elements we find a basis satisfying

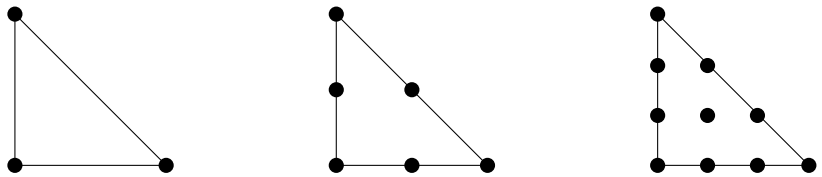
$$\phi_T^i(x_T^j) = \delta_{ij} \quad \forall i, j \in [1, m_T]$$

where $\delta_{ij}$ is the Kronecker delta, defined

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

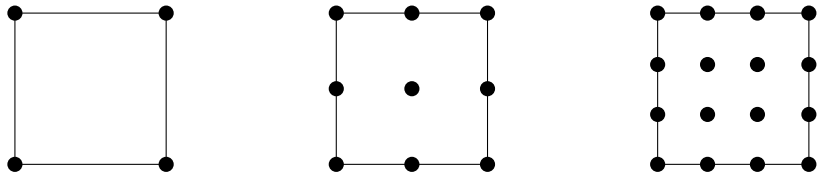
This will ensure $v = \sum_{i=1}^{m} \phi_i\sigma^i(v) \ \forall v \in X_h$. For this finite element space to be continuous, we need conditions (3.3) and (3.4) to hold. Let $e = T_r \cap T_s \neq \emptyset$. The first condition holds if basis functions corresponding to dofs outside the edge has a zero value on $e$, or more formally $\phi_{T_r}^i(x) = 0 \ \forall x \in e$ when $x_{T_r}^i \notin e$. This will ensure that the function values on $e$ only depends on the dofs on $e$. The second condition holds if

$$X_{T_r,e} = X_{T_s,e} \tag{3.7}$$



(a) $P_1$ element (b) $P_2$ element (c) $P_3$ element

Figure 3.1: Examples of $P_k$ elements in 2D. A dot represents a point $x_T^i$.



(a) $Q_1$ element (b) $Q_2$ element (c) $Q_3$ element

Figure 3.2: Examples of $Q_k$ elements in 2D. A dot represents a point $x_T^i$.

and there for every point $x_{T_r}^i \in e$ exists a point $x_{T_s}^j \in e$ such that

$$x_{T_r}^i = x_{T_s}^j \tag{3.8}$$

**Lagrange elements**

One very popular family of elements is the Lagrange elements. These are nodal elements where the nodes $x_T^i$ are arranged in a particular fashion (see figure 3.1) and the function space is the space of polynomials of degree $\leq k$ denoted $X_T = \mathcal{P}_k$. We then want a basis for $\mathcal{P}_k$ satisfying

$$\phi_T^i(x_T^j) = \delta_{ij} \quad \forall i, j \in [1, m_T]$$

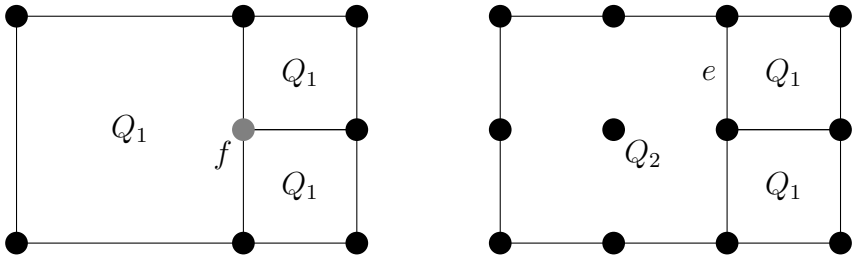
which is exactly the Lagrange polynomials [7, p. 354]

$$\phi_T^i(x) = \prod_{\substack{0<j\leq p \\ j\neq i}} \frac{x - x_T^j}{x_T^i - x_T^j}$$

When the domains are simplexes, these elements are called $P_k$ elements where the $k$ is the degree of the polynomial, and similarly when elements are Cartesian products of 1D $P_k$ elements we call them $Q_k$ elements. For instance, a 2D $Q_k$ element is $Q_k = P_k \otimes P_k$, see figure 3.2.

When connecting multiple elements, we need to ensure the continuity conditions still hold. If we for instance look at $Q_1$ elements, we cannot allow

a corner of one element to intersect the interior of an edge of another element (see figure 3.3a) as this would break condition (3.8). Even in the case where the connecting element have a node in the intersection and (3.8) is satisfied, we still need (3.7), which would not be the case with $Q_k$ and $Q_{k+1}$ elements (see figure 3.3b).




(a) The value at $f$ will be a dof for the two small elements, but not the larger. Such a node is called a *hanging node*.

(b) Here there are no hanging nodes, but edge $e$ has different function spaces on each side.

Figure 3.3: Examples of two invalid compositions of $Q_k$ elements

### 3.1.4 Error estimate

As most other numerical methods, this method finds an approximate solution to our problem. If $u$ is the exact solution to the continuous problem (2.6) and $u_h$ is the solution of (3.1), then we want a bound on the error $\|u_h - u\|$ in some norm $\|\cdot\|$. Since both $u_h$ and $u$ are in $H_0^1(\Omega)$, it is natural to look at the error in $H^1$-norm. It also turns out that because $a(u, v)$ satisfies theorem 2.6 using the $H^1$-norm, this is also the easiest to derive.

Making a $H^1$ error estimate consists of two steps, the first one being Céa's lemma [4, p. 55]

**Lemma 3.3** (Céa's lemma)**.** *Assume the bilinear form $a$ satisfies the conditions of theorem 2.6 with $H^m$-norm and assume $u \in H_0^m(\Omega)$ solves* (2.6) *and $u_h \in X_h \subset H_0^m(\Omega)$ solves* (3.1)*. Then*

$$\|u - u_h\|_{H^m(\Omega)} \leq C \inf_{v \in X_h} \|u - v\|_{H^m(\Omega)}$$

*Proof.* Since $u_h, u$ are solutions, we have

$$\begin{aligned} a(u, w) &= l(w) \quad \forall w \in H_0^m(\Omega) \\ a(u_h, w) &= l(w) \quad \forall w \in X_h \end{aligned}$$

and since $X_h \subset H_0^m(\Omega)$, we can subtract them and get

$$a(u - u_h, w) = 0 \quad \forall w \in X_h \tag{3.9}$$

Now, introduce a $v \in X_h$ and set $w = v - u_h$ and using property (i) and (ii) from theorem 2.6 we get

$$\begin{aligned}
c_2\|u - u_h\|^2_{H^m(\Omega)} &\leq a(u - u_h, u - u_h) && \text{from property (ii)} \\
&\leq a(u - u_h, u - v) + a(u - u_h, v - u_h) && \text{from linearity} \\
c_2\|u - u_h\|^2_{H^m(\Omega)} &\leq c_1\|u - u_h\|_{H^m(\Omega)}\|u - v\|_{H^m(\Omega)} && \text{from (3.9) and property (i)}
\end{aligned}$$

Dividing by $\|u - u_h\|_{H^m(\Omega)}$ gives

$$\|u - u_h\|_{H^m(\Omega)} \leq \frac{c_1}{c_2}\|u - v\|_{H^m(\Omega)}$$

and since this holds for all $v \in X_h$, it will also hold for the infimum. □

*Remark.* Property (3.9) is called *Galerkin orthogonality*. The reason for this is that if we look at $a$ as an inner product on $H^m(\Omega)$, the property states that $u - u_h$, or the error of the approximation, is orthogonal to every element in $X_h$ with respect to that inner product.

*Remark.* In Céa's lemma we assumed that $a$ is coercive. As noted before, this is not true for many important cases. There are generalizations to Céa's lemma where we replace the assumption of coercivity with the assumption that $a$ satisfies a discrete inf-sup condition. These generalizations does also provide results for non-conforming methods.

### Best approximation error

The quantity $\inf_{v \in X_h} \|u - v\|_{H^m(\Omega)}$ is called the best approximation error since it is the error of the best possible approximation of $u$ in $X_h$. To get a more useful error bound, we need a bound for the best approximation error. Proving regularity and bounds for the best approximation error will require introducing several new concepts which are outside the scope of this thesis. Hence, the following results will be provided with only proof sketches.

To state a meaningful result we need one definition [4, p. 61]

**Definition 3.4** (Shape regularity)**.** Let $\mathcal{T}$ be a mesh and let $h_T = \frac{1}{2}\operatorname{diam} T$ for each $T \in \mathcal{T}$. The mesh is called *shape regular* if there exists a number $\kappa$ such that every $T \in \mathcal{T}$ contains a circle with radius $\rho_T$ where

$$\rho_T \geq \frac{h_T}{\kappa}$$

Since the best approximation error is a lower bound of the error of any approximation, it suffices to show that there exists one approximation for which we can control the error.

**Lemma 3.5.** *(Bramble-Hilbert lemma) Let $t \geq 2$, and suppose $\mathcal{T}_h$ is a shape-regular triangulation of $\Omega \in \mathbb{R}^d$. Then there exists a constant $c = c(\Omega, \kappa, t)$ such that*

$$\|u - I_h u\|_{H^m(\Omega)} \leq c h^{m-t} |u|_{H^t(\Omega)} \quad \forall u \in H^t(\Omega), \;\; 0 \leq m \leq t \tag{3.10}$$

*where $I_h$ denotes interpolation by a piecewise polynomial of degree $t-1$.*

*Proof sketch.* As noted above, the result will not be proved, but a sketch of how the result can be proved is provided here.

Now let $T \in \mathcal{T}_h$ and let $\widehat{T}$ be a scaled version of $T$ such that $\operatorname{diam} \widehat{T} = 1$. The first step is to create a bound on the form

$$\|u - Iu\|_{H^t(\widehat{T})} \leq c |u|_{H^t(\widehat{T})} \quad \forall u \in H^t(\widehat{T}) \tag{3.11}$$

where $I$ is a polynomial interpolation operator on $\widehat{T} \in \mathbb{R}^d$. This result is provided trough Deny-Lions lemma [5, p. 120]. This result requires $u \in H^2(\Omega)$ which is why we need $t \geq 2$ in the lemma. We write $h_T = \operatorname{diam} T$ and let $S$ be the isomorphism

$$S : \begin{cases} \widehat{T} \to T \\ x \mapsto h_T x \end{cases}$$

If $\alpha$ is a multi-index, let $\partial^\alpha v$ denote the weak derivative of $v$ with respect to the indices in $\alpha$. We know that the chain rule applies to weak derivatives, hence $\partial^\alpha (v \circ S) = h_T^{|\alpha|} \partial^\alpha v$. Using this we can scale the semi-norm

$$\begin{aligned} |v \circ S|^2_{H^t(\widehat{T})} &= \sum_{|\alpha|=t} \int_{\widehat{T}} (\partial^\alpha v \circ S)^2 \,\mathrm{d}x \\ &= \sum_{|\alpha|=t} \int_T h_T^{2t} (\partial^\alpha v)^2 h_T^{-d} \,\mathrm{d}(Sx) \\ &= h_T^{2t-d} |v|_{H^t(T)} \end{aligned} \tag{3.12}$$

where we have used $\mathrm{d}x = h_T^{-d} \,\mathrm{d}(Sx)$. Using this we can do something similar

to the full norm, only scaling the other way around

$$\begin{aligned}\|v\|^2_{H^m(T)} &= \sum_{l=0}^{m} |v|^2_{H^l(T)} \\ &= \sum_{l=0}^{m} \int_T h_T^{-2l+d} \, |v \circ S|^2_{H^l(\hat{T})} \\ &\leq h_T^{-2m+d} \|v \circ S\|_{H^t(\hat{T})} \end{aligned} \tag{3.13}$$

where we have assumed $h_T \leq 1$ so $h_T^{-2l+d} \leq h_T^{-2m+d}$ when $l \leq m$. Combining this scaling with (3.11), we arrive at the following

$$\begin{aligned}\|u - Iu\|_{H^m(T)} &\leq h_T^{-m+d/2} \|u \circ S - Iu \circ S\|_{H^m(\hat{T})} && \text{from (3.13)} \\ &\leq h_T^{-m+d/2} \|u \circ S - Iu \circ S\|_{H^t(\hat{T})} && \text{since } m \leq t \\ &\leq h_T^{-m+d/2} \, |u \circ S|_{H^t(\hat{T})} && \text{from (3.11)} \\ &\leq h_T^{t-m} \, |u|_{H^t(T)} && \text{from (3.12)} \end{aligned}$$

since this holds for every $T$ and the mesh is shape regular with $\max_{T \in \mathcal{T}_h} h_T \leq h$, we can sum the error without loosing any exponent of $h$, and the result follows. □

Combining these to lemmas we can state a proper error bound

**Theorem 3.6.** *Assume the bilinear form $a$ satisfies the conditions of theorem 2.6 with $H^m$-norm and assume $u \in H_0^t(\Omega)$ solves (2.6) and $u_h \in X_h \subset H_0^t(\Omega)$ solves (3.1) for some $t \geq \max\{2, m\}$. Then*

$$\|u - u_h\|_{H^m(\Omega)} \leq c h^{m-t} \, |u|_{H^t(\Omega)}$$

*Proof.* Combining lemma 3.3 with 3.5 gives the result. □

### Regularity

Note that for the convergence theorem to be applicable we need at least $u \in H_0^2(\Omega)$. Since a solution of (2.6) may only be in $H_0^1(\Omega)$, we need some way of predicting when the solution will be more regular.

**Theorem 3.7.** *Let $k$ be a nonnegative integer and $\alpha, \boldsymbol{\beta}, \gamma$ be the coefficients from (2.5) and assume*

$$\alpha \in \left[C^{m+1}(\overline{\Omega})\right]^{d \times d} \qquad \boldsymbol{\beta} \in \left[C^{m+1}(\overline{\Omega})\right]^{d} \qquad \gamma \in C^{m+1}(\overline{\Omega})$$

*and*

$$f \in H^m(\Omega)$$

*Suppose that* $u \in H_0^1(\Omega)$ *is a weak solution of* (2.6) *and finally assume*

$$\partial\Omega \text{ is of class } C^{m+2}$$

*Then*

$$u \in H^{m+2}(\Omega)$$

*Motivation.* This is proved in [9, p. 340], but the proof is too long to include here. Instead, we will look at a brief motivation for why one can expect such a result to exist. Let us look at the Poisson problem

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

and assume $u \in C_0^\infty(\Omega)$. We then square the equation and integrate over $\Omega$

$$\begin{aligned} \int_\Omega f^2 \,\mathrm{d}x = \int_\Omega (\Delta u)^2 \,\mathrm{d}x &= \sum_{i,j=1}^d \int_\Omega (\partial^i \partial^i u)(\partial^j \partial^j u) \,\mathrm{d}x \\ &= -\sum_{i,j=1}^d \int_\Omega (\partial^i \partial^i \partial^j u)\partial^j u \,\mathrm{d}x \\ &= \sum_{i,j=1}^d \int_\Omega (\partial^i \partial^j u)(\partial^i \partial^j u) \,\mathrm{d}x \\ &= \sum_{|\alpha|=2} \int_\Omega (\partial^\alpha u)^2 \,\mathrm{d}x \end{aligned}$$

hence

$$|u|_{H^2(\Omega)} = \|f\|_{L^2(\Omega)}$$

or, using Poincaré's inequality

$$\|u\|_{H^2(\Omega)} \leq C\|f\|_{L^2(\Omega)}$$

for some $C \in \mathbb{R}$. Similarly, looking instead at

$$-\Delta \tilde{u} = \tilde{f}$$

where $\tilde{u} = \partial^\alpha u$, $\tilde{f} = \partial^\alpha f$, we conclude that we can find a $C \in \mathbb{R}$ such that

$$\|u\|_{H^{m+2}(\Omega)} \leq C\|f\|_{H^m(\Omega)}$$

□

## 3.2 Discontinuous Galerkin method

The discontinuous Galerkin method may seem quite similar to the continuous variant, formally the only difference being lifting one assumption on how to assemble the elements. This does, however, change many aspects of both implementation and analysis of the methods. It even requires us to derive a different weak formulation of the discrete problem since the normal weak formulation does not support the double-valued functions that will arise.

### 3.2.1 Finite element space

In section 3.1.2 we defined a finite element method to consist of $n$ elements such that

i $\Omega = \bigcup_{r=1}^{n} T_r$

ii $\dim(T_r \cap T_s) < \dim \Omega \quad \forall r, s \in [1, n],\ r \neq s$

iii $X_h = \{u \in C(\Omega) \,:\, r \in [1, n],\ \ u|_{T_r} \in X_{T_r}\}$

The discontinuous Galerkin method only changes property (iii) to remove the requirement that $X_h$ is continuous, and instead looking at elements in the function space $X_h$ as multiple separate functions

$$X_h = \prod_{r=1}^{n} X_{T_r}$$

This means we do not need the restriction on the dofs as introduced in 3.1.3, and it makes the dof map bijective using all the local dofs as global ones. While giving us more flexibility, it also has some drawbacks. First, not collapsing dofs means we will have more global dofs which results in a bigger linear system. Secondly, we need to derive the weak formulation differently since $X_h \not\subset H^1(\Omega)$ but $a$ only takes values on $H^1(\Omega)$. Because of this, discontinuous Galerkin is an example of a non-conforming method.

### 3.2.2 Flux formulation

There are two weak forms that appear in the study of discontinuous Galerkin method, the flux formulation and the primal formulation. We will derive the flux formulation and then discuss what is needed to transform it to a primal formulation. The primal formulation often has most terms in common with the continuous weak form, only adding some integrals over the interior edges.

The flux formulation, however, will look quite different since it is a mixed formulation between the original unknown $u$ and an auxiliary variable.

Let us look at how the Helmholtz equation looks in discontinuous Galerkin. We consider the Helmholtz equation with homogeneous Dirichlet boundary conditions

$$\begin{cases} -\Delta u - k^2 u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{3.14}$$

where $\Omega$ and all the finite element domains $T_r$ are Lipschitz domains in $\mathbb{R}^d$. We also assume $f \in L^2(\Omega)$. Since the functions to be used as test and trial functions are not smooth enough, we cannot do integration by parts over $\Omega$. Instead, we have to look at the equation on a single element. Since the functions are continuous on each domain $T_r$, we can use techniques similar to those used to derive the weak formulation for continuous functions, but first we split this second order equation into a set of two first order equations by introducing the auxiliary variable $\boldsymbol{\mu} \in V_{T_r} \subset [L^2(T_r)]^d$

$$\boldsymbol{\mu} = \frac{\operatorname{grad} u}{ik}$$

Now that we have two variables, we also need two function spaces for the element. We do this by replacing $X_T$ with $U_T$ and $V_T$. This gives us the local problem of finding $u \in U_{T_r}$ and $\boldsymbol{\mu} \in V_{T_r}$ such that

$$\begin{cases} ik\boldsymbol{\mu} = \operatorname{grad} u & \text{in } T_r \\ iku - \operatorname{div} \boldsymbol{\mu} = \frac{1}{ik} f & \text{in } T_r \end{cases} \tag{3.15}$$

Analog to what was done in section 2.2.1, we multiply by test functions $v \in U_{T_r}$ and $\boldsymbol{\tau} \in V_{T_r}$ and assume the function spaces $U_{T_r}, V_{T_r}$ are continuous enough to do the integration by parts. This gives us the weak formulation on the element $T_r$

$$\int_{T_r} ik\boldsymbol{\mu} \cdot \overline{\boldsymbol{\tau}}\, dV + \int_{T_r} u\overline{\operatorname{div} \boldsymbol{\tau}}\, \mathrm{d}x - \int_{\partial T_r} u\overline{\boldsymbol{\tau} \cdot \boldsymbol{n}}\, \mathrm{d}s = 0 \tag{3.16}$$

$$\int_{T_r} iku\overline{v}\, \mathrm{d}x + \int_{T_r} \boldsymbol{\mu} \cdot \overline{\operatorname{grad} v}\, \mathrm{d}x - \int_{\partial T_r} \boldsymbol{\mu} \cdot \boldsymbol{n}\overline{v}\, \mathrm{d}s = \frac{1}{ik} \int_{\Omega} f\overline{v}\, \mathrm{d}x \tag{3.17}$$

where $\boldsymbol{n}_{T_r}$ is the outward pointing normal vector of $\partial T_r$ and $\overline{v}$ denotes the complex conjugate of $v$. We may then drop the assumptions that $U_{T_r}, V_{T_r}$ are continuous and instead assume $U_{T_r} \subset H^2(T_r)$ and $V_{T_r} \subset [H^1(T_r)]^d$.

To reason further we define the global function spaces

$$U = \prod_{r\in[1,n]} U_{T_r} \qquad V = \prod_{r\in[1,n]} V_{T_r} \qquad T(\Gamma) = \prod_{r\in[1,n]} L^2(\partial T_r)$$

We also need some notation. Let $e = T_r \cap T_s$ for $1 \leq r, s \leq n, r \neq s$ and $e \neq \emptyset$, then we can define most common operations on $T(\Gamma)$ as linear operators. First, we can define the average operators

$$\{\cdot\} : \begin{cases} T(\Gamma) & \to L^2(\Gamma) \\ [q^i]_{i\in[1,n]} & \mapsto \sum_{e\in\Gamma} \frac{1}{2}(q^r + q^s) \end{cases}$$
$$\{\cdot\} : \begin{cases} [T(\Gamma)]^d & \to [L^2(\Gamma)]^d \\ [\boldsymbol{q}^i]_{i\in[1,n]} & \mapsto \sum_{e\in\Gamma} \frac{1}{2}(\boldsymbol{q}^r + \boldsymbol{q}^s) \end{cases}$$

and the jump operators

$$[\![\cdot]\!] : \begin{cases} T(\Gamma) & \to [L^2(\Gamma)]^d \\ [q^i]_{i\in[1,n]} & \mapsto \sum_{e\in\Gamma} (q^r \boldsymbol{n}_{T_r} + q^s \boldsymbol{n}_{T_s}) \end{cases}$$
$$[\![\cdot]\!] : \begin{cases} [T(\Gamma)]^d & \to L^2(\Gamma) \\ [\boldsymbol{q}^i]_{i\in[1,n]} & \mapsto \sum_{e\in\Gamma} (\boldsymbol{q}^r \cdot \boldsymbol{n}_{T_r} + \boldsymbol{q}^s \cdot \boldsymbol{n}_{T_s}) \end{cases}$$

where $\boldsymbol{n}_{T_r}$ is the outward pointing normal from $T_r$ on $e$.

We can also define a trace operator

$$T : \begin{cases} U \to T(\Gamma) \\ [v^i]_{i\in[1,n]} \mapsto [T(v^i)]_{i\in[1,n]} \end{cases}$$

where $T : H^1(T) \to L^2(\partial T)$ denotes the standard $H^1$ trace. Similarly, we can define a trace operator on $V$, $T : V \to [T(\Gamma)]^d$. Using these traces we define $[\![v]\!] = [\![T(v)]\!]$ and $\{v\} = \{T(v)\}$ for $v \in U$ and $v \in V$.

Intuitively, the jumps should be something like $[\![q]\!] = q^r - q^s$, but this form would change sign depending on the order of $T_r$ and $T_s$. Instead, since $\boldsymbol{n}_{T_r} = -\boldsymbol{n}_{T_s}$, we have $[\![q]\!] = (q^r - q^s)\boldsymbol{n}_{T_r}$. This way they are completely symmetric in the order of $T_r$ and $T_s$.

Now we have all we need to assemble a global weak formulation of our discontinuous problem. We do, however, want one more change first. Since the discontinuities are a crucial part of how this method behaves, we want more flexibility in how they are handled. We do this by introducing the *flux functions*[6]

$$\hat{u} : (U, V) \to T(\Gamma)$$
$$\hat{\boldsymbol{\mu}} : (U, V) \to [T(\Gamma)]^d$$

and replace $u$ and $\boldsymbol{\mu}$ in the boundary terms of (3.16) and (3.17) with $\hat{u}(u, \boldsymbol{\mu})$ and $\hat{\boldsymbol{\mu}}(u, \boldsymbol{\mu})$ to get the local *flux formulation*

$$\int_{T_r} ik\boldsymbol{\mu} \cdot \overline{\boldsymbol{\tau}} \, \mathrm{d}x + \int_{T_r} u\overline{\operatorname{div} \boldsymbol{\tau}} \, \mathrm{d}x - \int_{\partial T_r} \hat{u}\overline{\boldsymbol{\tau} \cdot \boldsymbol{n}} \, \mathrm{d}s = 0 \tag{3.18}$$

$$\int_{T_r} iku\overline{v} \, \mathrm{d}x + \int_{T_r} \boldsymbol{\mu} \cdot \overline{\operatorname{grad} v} \, \mathrm{d}x - \int_{\partial T_r} \hat{\boldsymbol{\mu}} \cdot \boldsymbol{n}\overline{v} \, \mathrm{d}s = \frac{1}{ik} \int_{T_r} f\overline{v} \, \mathrm{d}x \tag{3.19}$$

where $u, v \in U_{T_r}$ and $\boldsymbol{\mu}, \boldsymbol{\tau} \in V_{T_r}$.

Using the notation we introduced, we may also construct the global flux formulation by assuming $u, v \in U$, $\boldsymbol{\mu}, \boldsymbol{\tau} \in V$ and summing these integrals for each element domain $T_r$. For $[\phi^r]_{r\in[1,n]} = \phi \in U$ and $[\boldsymbol{q}^r]_{r\in[1,n]} = \boldsymbol{q} \in V$, we have the identity

$$\sum_{r\in[1,n]} \int_{\partial T_r} \phi^r \boldsymbol{q}^r \cdot \boldsymbol{n}_{T_r} \, \mathrm{d}x = \int_{\Gamma^0} [\![\phi]\!] \cdot \{\boldsymbol{q}\} \, \mathrm{d}s + \int_{\Gamma^0} \{\phi\} [\![\boldsymbol{q}]\!] \, \mathrm{d}s \tag{3.20}$$

where we have assumed

$$\sum_{r\in[1,n]} \int_{\partial T_r \cap \partial\Omega} \phi^r \boldsymbol{q}^r \cdot \boldsymbol{n} \, \mathrm{d}s = 0$$

which is true if $\phi$ or $\boldsymbol{q}$ satisfies homogeneous Dirichlet boundary conditions. To simplify notation we will also use the notation

$$\int_\Omega \phi \, \mathrm{d}x = \sum_{r\in[1,n]} \int_{T_r} \phi^r \, \mathrm{d}x \tag{3.21}$$

for $[\phi^r]_{r\in[1,n]} = \phi \in U$. Using this we can obtain the global flux formulation of the problem

$$\int_\Omega ik\boldsymbol{\mu} \cdot \overline{\boldsymbol{\tau}} \, \mathrm{d}x + \int_\Omega u\overline{\operatorname{div} \boldsymbol{\tau}} \, \mathrm{d}x - \int_{\Gamma^0} [\![\hat{u}]\!] \cdot \{\overline{\boldsymbol{\tau}}\} \, \mathrm{d}s - \int_{\Gamma^0} \{\hat{u}\} [\![\overline{\boldsymbol{\tau}}]\!] \, \mathrm{d}s = 0 \tag{3.22}$$

$$\int_\Omega iku\overline{v} \, \mathrm{d}x + \int_\Omega \boldsymbol{\mu} \cdot \overline{\operatorname{grad} v} \, \mathrm{d}x - \int_{\Gamma^0} [\![\hat{\boldsymbol{\mu}}]\!] \{v\} \, \mathrm{d}s - \int_{\Gamma^0} \{\hat{\boldsymbol{\mu}}\} \cdot [\![v]\!] \, \mathrm{d}s = \frac{1}{ik} \int_\Omega f\overline{v} \, \mathrm{d}x \tag{3.23}$$

for $u, v \in U$ and $\boldsymbol{\mu}, \boldsymbol{\tau} \in V$.

### 3.2.3 Flux functions

The choice of flux functions greatly affects stability and accuracy of the method and also the runtime of the linear solver through sparsity and condition number of the resulting linear system [2, p. 1750]. We will not go in detail on different choices of flux functions, but there are some important properties that characterize them.

**Definition 3.8** (Locality)**.** Let $e = T_r \cap T_s$ for some $1 \leq r, s \leq n, r \neq s$, $v \in U$, $\boldsymbol{\nu} \in V$ and assume $e \neq \emptyset$. If the fluxes on $e$ only depend on the traces of its argument restricted to $e$

$$\hat{u}(v, \boldsymbol{\nu})|_e = \hat{u}_e \left(T(v^r)|_e, T(v^s)|_e, T(\boldsymbol{\nu}^r)|_e, T(\boldsymbol{\nu}^s)_e\right)$$
$$\hat{\boldsymbol{\mu}}(v, \boldsymbol{\nu})|_e = \hat{\boldsymbol{\mu}}_e \left(T(v^r)|_e, T(v^s)|_e, T(\boldsymbol{\nu}^r)|_e, T(\boldsymbol{\nu}^s)|_e\right)$$

we say the fluxes are *local*.

Local fluxes give a computational advantage in that we may only provide the function values on the edge in question to compute integrals over the flux.

**Definition 3.9** (Consistency)**.** Let $v = [v^r]_{r \in [1,n]} \in U$ such that

$$\exists \tilde{v} \in C^\infty(\Omega) : \tilde{v}|_{T_r} = v^r \ \ \forall r \in [1, n]$$

The fluxes are *consistent* if

$$\hat{u}(v, \operatorname{grad} v) = v|_\Gamma$$
$$\hat{\boldsymbol{\mu}}(v, \operatorname{grad} v) = (\operatorname{grad} v)|_\Gamma$$

Consistency ensures that if we assume smooth enough functions, the discontinuous Galerkin method will be equivalent to the corresponding continuous Galerkin method.

**Definition 3.10** (Conservation)**.** If the $\widehat{\boldsymbol{\mu}}$ flux is single valued, we say the fluxes are *conservative*. If the $\widehat{u}$ flux is also single valued, then the fluxes are *completely conservative*.

If we let $u$ represent some physical quantity in our domain and the flux $\widehat{u}$ represent some physical flux of this quantity through the boundaries of the element domains, then completely conservative fluxes will ensure that the flux out of one element is exactly the same as the flux into the other element, and the quantity will in some sense be preserved.

The most common fluxes can be written as only functions of the jump and average operators.

$$\hat{u}(v, \boldsymbol{\nu}) = \hat{u}\left(\{v\}, [\![v]\!], \{\boldsymbol{\nu}\}, [\![\boldsymbol{\nu}]\!]\right)$$
$$\hat{\boldsymbol{\mu}}(v, \boldsymbol{\nu}) = \hat{\boldsymbol{\mu}}\left(\{v\}, [\![v]\!], \{\boldsymbol{\nu}\}, [\![\boldsymbol{\nu}]\!]\right)$$

We observe from the definition of the operators that the fluxes will be local. Also, since the operators are single valued, the fluxes will be completely conservative. Finally, making them consistent is usually trivial since for $v \in C^\infty$ we have

$$[\![v]\!] = 0 \qquad\qquad \{v\} = v$$

### 3.2.4 Primal formulation

While the flux formulation provides a lot of flexibility and insight, it takes us quite far from the continuous formulation. Using a mixed formulation may complicate further analysis by having more complicated inf-sup conditions, and it may also make numerical implementations less efficient since we have to solve for $\boldsymbol{\mu}$, which we are not really interested in. Deriving the primal formulation consists of eliminating the auxiliary variable $\boldsymbol{\mu}$ [2, p. 1757].

We start by looking at the second term of (3.18). Trough integration by parts we get

$$-\int_{T_r} u\overline{\operatorname{div}\boldsymbol{\tau}}\,\mathrm{d}x = \int_{T_r} \operatorname{grad} u\cdot\overline{\boldsymbol{\tau}}\,\mathrm{d}x - \int_{\partial T_r} u\overline{\boldsymbol{\tau}}\cdot\boldsymbol{n}_{T_r}\,\mathrm{d}s$$

Summing over all the element domains and using (3.20) with $\phi = u$, $\boldsymbol{q} = \boldsymbol{\tau}$ we get

$$-\int_{\Omega} u\overline{\operatorname{div}\boldsymbol{\tau}}\,\mathrm{d}x = \int_{\Omega} \operatorname{grad} u\cdot\overline{\boldsymbol{\tau}}\,\mathrm{d}x - \int_{\Gamma^0} \{\tau\}\cdot[\![u]\!]\,\mathrm{d}s - \int_{\Gamma^0} [\![\boldsymbol{\tau}]\!]\{u\}\,\mathrm{d}s$$

which can be inserted into (3.22)

$$\int_{\Omega} ik\boldsymbol{\mu}\cdot\overline{\boldsymbol{\tau}}\,\mathrm{d}x = \int_{\Omega} \operatorname{grad} u\cdot\boldsymbol{\tau}\,\mathrm{d}x + \int_{\Gamma^0} [\![\hat{u}-u]\!]\cdot\{\boldsymbol{\tau}\}\,\mathrm{d}s + \int_{\Gamma^0} \{\hat{u}-u\}[\![\boldsymbol{\tau}]\!]\,\mathrm{d}s \quad (3.24)$$

which then holds for all $\boldsymbol{\tau} \in V$. Since $\operatorname{grad} v \in V\ \forall v \in U$, we can set $\boldsymbol{\tau} = \operatorname{grad} v$ which makes the left hand side of (3.24) match the second term of (3.23). Inserting it gives us the primal formulation

$$\begin{aligned}\int_{\Omega} \operatorname{grad} u\cdot\overline{\operatorname{grad} v}\,\mathrm{d}x - \int_{\Omega} k^2 u\overline{v}\,\mathrm{d}x + \int_{\Gamma^0} \left([\![\hat{u}-u]\!]\cdot\{\overline{\operatorname{grad} v}\} - \{\hat{\boldsymbol{\mu}}\}\cdot[\![\overline{v}]\!]\right)\mathrm{d}s \\ + \int_{\Gamma^0} \left(\{\hat{u}-u\}[\![\overline{\operatorname{grad} v}]\!] - [\![\hat{\boldsymbol{\mu}}]\!]\{\overline{v}\}\right)\mathrm{d}s = \int_{\Omega} f\overline{v}\,\mathrm{d}x\end{aligned} \quad (3.25)$$

This expression is still not completely independent of $\boldsymbol{\mu}$ since both the flux functions $\hat{u}$ and $\hat{\boldsymbol{\mu}}$ may depend on it. If we still want to allow the fluxes to depend on $\boldsymbol{\mu}$, we can use (3.24) as a way of computing $\boldsymbol{\mu}$ from $u$, but it is also possible to instead require

$$\begin{aligned}\hat{u} &= \hat{u}(u, \operatorname{grad} u)\\ \hat{\boldsymbol{\mu}} &= \hat{\boldsymbol{\mu}}(u, \operatorname{grad} u)\end{aligned}$$

which completely removes $\boldsymbol{\mu}$ from the formulation.

We can now see the relation between the weak formulation for the continuous problem and the primal formulation. Let $a$ and $l$ be the bilinear and linear form of the standard weak formulation of (3.14)

$$a(u,v) = \int_\Omega \operatorname{grad} u \cdot \overline{\operatorname{grad} v}\,\mathrm{d}x - \int_\Omega k^2 u\overline{v}\,\mathrm{d}x$$
$$l(v) = \int_\Omega f\overline{v}\,\mathrm{d}x$$

We can then make $a_c : U \times U \to \mathbb{R}$ and $l_c : U \times U \to \mathbb{R}$ where

$$a_c(u,v) = \int_\Omega \operatorname{grad} u \cdot \operatorname{grad} v\,\mathrm{d}x - \int_\Omega k^2 u\overline{v}\,\mathrm{d}x$$
$$l_c(v) = \int_\Omega f\overline{v}\,\mathrm{d}x$$

which is exactly the same expressions, only here we use the notation trick (3.21) to allow functions from $U$ to be integrated.

The primal formulation can then be written

$$a_c(u,v) + a_d(u,v) = l(v)$$

where

$$a_d(u,v) = \int_{\Gamma^0} \left( [\![\hat{u} - u]\!] \cdot \{\overline{\operatorname{grad} v}\} - \{\hat{\boldsymbol{\mu}}\} \cdot [\![\overline{v}]\!] \right) \mathrm{d}s + \int_{\Gamma^0} \left( \{\hat{u} - u\} [\![\overline{\operatorname{grad} v}]\!] - [\![\hat{\boldsymbol{\mu}}]\!] \{\overline{v}\} \right) \mathrm{d}s$$

Let $u = [u^r]_{r\in[1,n]} \in U$ and $v = [v^r]_{r\in[1,n]} \in U$, and assume there exists $\tilde{u}, \tilde{v} \in H_0^1$ such that

$$\tilde{u}|_{T_r} = u^r \quad \forall \in [1,n]$$
$$\tilde{v}|_{T_r} = u^r \quad \forall r \in [1,n]$$

then

$$\begin{aligned}
a_c(u,v) &= \int_\Omega \operatorname{grad} u \cdot \overline{\operatorname{grad} v}\,\mathrm{d}x - \int_\Omega k^2 u\overline{v}\,\mathrm{d}x \\
&= \sum_{r\in[1,n]} \left( \int_{T_r} \operatorname{grad} u^r \cdot \overline{\operatorname{grad} v^r}\,\mathrm{d}x - \int_{T_r} k^2 u^r \overline{v^r}\,\mathrm{d}x \right) \\
&= \sum_{r\in[1,n]} \left( \int_{T_r} \operatorname{grad} \tilde{u} \cdot \overline{\operatorname{grad} \tilde{v}}\,\mathrm{d}x - \int_{T_r} k^2 \tilde{u}^r \overline{\tilde{v}^r}\,\mathrm{d}x \right) \\
&= \int_\Omega \operatorname{grad} \tilde{u} \cdot \overline{\operatorname{grad} \tilde{v}}\,\mathrm{d}x - \int_\Omega k^2 \tilde{u}^r \overline{\tilde{v}^r}\,\mathrm{d}x \\
&= a(\tilde{u}, \tilde{v})
\end{aligned}$$

and similarly for $l$ and $l_c$. Also, since the trace on any interior edge would have to be the same from any side, all the jump operators in $a_d$ will be zero, which means $a_d(u, v) = 0$. Hence, if we restrict $U$ to only contain sets of functions which can be assembled to $H_0^1$ functions, then the primal formulation and the continuous weak formulation are equivalent.

# Chapter 4

# Plane Wave Semi-Continuous Galerkin method

In this chapter we will define the Plane Wave Semi-Continuous Galerkin method, or the PWSCG-method for short. This is a finite element method which incorporates some of the structure of the expected solutions into the discrete spaces used in the elements. This is done by assuming the solution will behave like a plane wave locally and hence may be approximated by plane waves on each element.

One way of using plane waves to approximate solutions to Helmholtz equation locally was explored in [11] and [12] where they used the space of a finite number of plane waves in different directions as discrete function space on each element. One thing to consider with this method is that the number of dofs may be high since we need a lot of different wave directions on each element, while the number of elements must also be high since we only assume the solution behaves like a plane waves locally.

A different approach is assuming the wave direction is known for each element. This can be done either by deriving $\boldsymbol{k}$ from the problem being solved or it may be determined adaptively by solving the equation with simpler spaces first and then extracting the oscillatory behaviour at different points.

The PWSCG-method assumes $\boldsymbol{k}$, or at least an approximation of $\boldsymbol{k}$ is known. Using this we may rewrite Helmholtz equation resulting in an equation for which the space of solutions has a natural finite dimensional subspace. Using this space we then derive a nodal semi-continuous finite element basis which we use with the Galerkin method to find a solution.

## 4.1 The PWSCG finite element space

To derive the PWSCG element we first need to define the function space we will use locally, and then find a useful basis of that space.

### 4.1.1 Plane wave function spaces

We want to construct a function space which contains functions with some of the structure we expect from solutions of the Helmholtz equation. To achieve this we will look at the homogeneous Helmholtz equation

$$\Delta u + \boldsymbol{k}^2 u = 0 \tag{4.1}$$

where $\boldsymbol{k} \in \mathbb{R}^d$, as in section 2.1.2. In [11, p. 303] the space $PW_\omega(\mathbb{R}^2)$ is defined, and the definition expanded to $\mathbb{R}^d$ will be as follows

**Definition 4.1** (Plane wave space)**.**

$$PW_{\boldsymbol{k}}(\mathbb{R}^d) = \left\{ u \in C^2(\mathbb{R}^d) \,:\, \Delta u + \boldsymbol{k}^2 u = 0 \text{ in } \mathbb{R}^d \right\}$$

where $\boldsymbol{k} \in \mathbb{R}^d$ is constant.

While $\boldsymbol{k}$ is a vector and $\omega$ is a scalar, the spaces are equivalent as long as $|\boldsymbol{k}| = \omega$.

We will define another space $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$ which preserves some of the structure of $PW_{\boldsymbol{k}}(\mathbb{R}^d)$ while having a finite dimensional subspace of dimension $2^d$ which turns out to be useful when using it as a finite element space. To derive the space we start by assuming the solution will be on the form

$$u = e^{i\boldsymbol{k}\cdot\boldsymbol{x}} \tag{4.2}$$

We know from section 2.1.2 that (4.2) is a solution to (4.1) with constant $\boldsymbol{k}$, and we assume it will be a good approximation locally to a solution of Helmholtz equation with non-constant $\boldsymbol{k}$ in an area where $\boldsymbol{k}$ does not change rapidly.

Assuming constant $\boldsymbol{k}$ and using the identity

$$\operatorname{grad} e^{i\boldsymbol{k}\cdot\boldsymbol{x}} = i\boldsymbol{k}e^{i\boldsymbol{k}\cdot\boldsymbol{x}} \Rightarrow \operatorname{grad} u = i\boldsymbol{k}u$$

we have

$$i\boldsymbol{k} \cdot \operatorname{grad} u = -\boldsymbol{k} \cdot \boldsymbol{k}u = -\left|\boldsymbol{k}\right|^2 u$$

and hence we may rewrite (4.1) to

$$\Delta u - i\boldsymbol{k} \cdot \operatorname{grad} u = 0 \tag{4.3}$$

Then, multiplying the equation by $e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ gives

$$e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} \operatorname{div}\operatorname{grad} u + \operatorname{grad}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\right)\cdot\operatorname{grad} u = 0$$

which is equivalent to

$$\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad} u\right) = 0 \tag{4.4}$$

from the identity [14, p. 78]

$$\operatorname{div}(a\boldsymbol{b}) = a\operatorname{div}\boldsymbol{b} + \operatorname{grad} a\cdot\boldsymbol{b}$$

with $a = e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ and $\boldsymbol{b} = \operatorname{grad} u$. We the define the space

**Definition 4.2** (The $PW_{\boldsymbol{k}}^{+}(\mathbb{R}^d)$ space)**.**

$$PW_{\boldsymbol{k}}^{+}(\mathbb{R}^d) = \left\{u \in C^2(\mathbb{R}^d) \,:\, \operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad} u\right) = 0\right\}$$

The + in the space name means we chose positive exponent in (4.2). A similar space $PW_{\boldsymbol{k}}^{-}(\mathbb{R}^d)$ may be constructed by using $-i\boldsymbol{k}\cdot\boldsymbol{x}$ as exponent in (4.2).

### Functions in $PW_{\boldsymbol{k}}^{+}(\mathbb{R}^d)$

Equations (4.3) and (4.4) are equivalent, but in the transition from (4.1) to (4.3) we have introduced some new solutions and lost others. First, we show that $v = e^{i\boldsymbol{k}\cdot\boldsymbol{x}}$ is a solution of (4.4)

$$\begin{aligned}\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad} e^{i\boldsymbol{k}\cdot\boldsymbol{x}}\right) &= \operatorname{div}\left(i\boldsymbol{k}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}e^{i\boldsymbol{k}\cdot\boldsymbol{x}}\right)\\ &= \operatorname{div}(i\boldsymbol{k})\\ &= 0\end{aligned}$$

hence $v \in PW_{\boldsymbol{k}}^{+}(\mathbb{R}^d) \cap PW_{\boldsymbol{k}}(\mathbb{R}^d)$.

Now we look at the solutions we have gained. Let $\boldsymbol{k} = [k_1, \ldots, k_d]$ and $\boldsymbol{x} = [x_1, \ldots x_d]$, then

$$v = e^{ik_j x_j} \quad j \in [1, d]$$

is a solution of (4.4) since

$$\begin{aligned}\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad} e^{ik_j x_j}\right) &= \operatorname{div}\left(ik_j\boldsymbol{e}_j e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}e^{ik_j x_j}\right)\\ &= \frac{\partial}{\partial x_j}\left(ik_i e^{-i\sum_{\substack{r\in[1,d]\\ r\neq j}} k_r x_r}\right)\\ &= 0\end{aligned}$$

but if we insert it into (4.1) we get

$$\begin{aligned}\Delta e^{ik_jx_j} + \boldsymbol{k}^2 e^{ik_jx_j} &= -k_j^2 e^{ik_jx_j} + \boldsymbol{k}^2 e^{ik_jx_j} \\ &= (\boldsymbol{k}^2 - k_j^2) e^{ik_jx_j}\end{aligned}$$

which is only zero if $\boldsymbol{k} = k_j\boldsymbol{e}_j$. Hence, as long as $\boldsymbol{k} \neq k_j\boldsymbol{e}_j$ we have that $v \in PW_{\boldsymbol{k}}^+(\mathbb{R}^d) \setminus PW_{\boldsymbol{k}}(\mathbb{R}^d)$. Another function in $PW_{\boldsymbol{k}}^+(\mathbb{R}^d) \setminus PW_{\boldsymbol{k}}(\mathbb{R}^d)$ is the constant function $v = 1$ since

$$\begin{aligned}\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} \operatorname{grad} 1\right) &= 0 \\ \Delta 1 + \boldsymbol{k}^2 \times 1 &= \boldsymbol{k}^2 \neq 0\end{aligned}$$

More generally we state

**Lemma 4.3.** *If $v$ is on the form*

$$v = e^{i\boldsymbol{\gamma}\cdot\boldsymbol{x}}$$

*where*

$$\boldsymbol{\gamma} = [\alpha_1 k_1, \ldots, \alpha_d k_d], \alpha_j \in \{0, 1\}$$

*and $\boldsymbol{k} = [k_j]_{j\in[1,d]}$, then $v \in PW_{\boldsymbol{k}}^+(\mathbb{R}^d)$.*

*Proof.* We just insert $v$ into (4.4) and get

$$\begin{aligned}\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} \operatorname{grad} e^{i\boldsymbol{\gamma}\cdot\boldsymbol{x}}\right) &= \sum_{j=1}^{d} \frac{\partial}{\partial x_j}\left(i\alpha_j k_j e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} e^{i\alpha_j k_j x_j}\right) \\ &= \sum_{j=1}^{d} \alpha_j \frac{\partial}{\partial x_j}\left(ik_j e^{-i\sum_{\substack{r\in[1,d]\\ r\neq j}} k_r x_r} e^{i(\alpha_j - 1)k_j x_j}\right)\end{aligned}$$

Choose $j \in [1, d]$. If $\alpha_j = 0$ the term has a zero factor, and if $\alpha_j = 1$ then

$$e^{i(\alpha_j - 1)k_j x_j} = e^0 = 1$$

and the expression is constant with respect to $x_j$, so the derivative will be zero. □

In addition to linear combinations, we have another useful way to create more solutions from existing solutions of 4.4

**Lemma 4.4.** *Let $u, v \in PW_{\boldsymbol{k}}^+(\mathbb{R}^d)$ where $u$ depend only on $[x_j]_{j\in U}$ and $v$ depend only on $[x_j]_{j\in V}$. If $U \cap V = \emptyset$, then $uv \in PW_{\boldsymbol{k}}^+(\mathbb{R}^d)$.*

*Proof.* We insert $uv$ into (4.4) and get

$$\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad}(uv)\right) = \operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\left(u\operatorname{grad}v + v\operatorname{grad}u\right)\right)$$

If we look at the term $(u \operatorname{grad} v)$ we observe that $\operatorname{grad} v$ will be zero in all the components representing variables where $u$ is non-constant. Hence, $u$ can be regarded as a constant with respect to the divergence

$$\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}u\operatorname{grad}v\right) = u\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad}v\right) = 0$$

since $v$ is a solution. The same argument can be done for $u$ and the result follows. □

We have also lost solutions since we assumed $u = e^{i\boldsymbol{k}\cdot\boldsymbol{x}}$ while $u = e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ is also a solution of (4.1). Inserting $v = e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ into (4.4) gives

$$\begin{aligned}\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\right) &= \operatorname{div}\left(i\boldsymbol{k}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\right)\\ &= \operatorname{div}\left(i\boldsymbol{k}e^{-2i\boldsymbol{k}\cdot\boldsymbol{x}}\right)\\ &\neq 0\end{aligned}$$

hence $v \in PW_{\boldsymbol{k}}(\mathbb{R}^d) \setminus PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$.

**Linear functions** Linear functions are generally not a part of $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$ since

$$\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad}(\boldsymbol{a}\cdot\boldsymbol{x})\right) = \operatorname{div}\left(\boldsymbol{a}e^{-i\boldsymbol{k}\cdot\boldsymbol{x}})\right) \not\equiv 0$$

There are, however, special cases where linear functions do become a part of the space. If we assume $k_i = 0$ for some $i \in [1, d]$ we have that

$$\operatorname{div}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\operatorname{grad}x_i)\right) = \frac{\partial}{\partial x_i}\left(e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}\right)$$

and since $k_i = 0$, we have that $\boldsymbol{k}\cdot\boldsymbol{x}$ is constant with respect to $x_i$, this means the derivate is zero. Hence, if there is an axis along which the plane wave would always be constant, we instead get a linear function in this direction.

### 4.1.2 Finite plane wave space

We will now construct the finite subspace $\widehat{PW}^+_{\boldsymbol{k}}(T) \subset PW^+_{\boldsymbol{k}}(T)$ which we will use in the finite element method. To define the space, we will assume $T$ is a $d$-parallelotope, and we want any function in $\widehat{PW}^+_{\boldsymbol{k}}(T)$ to be uniquely defined by the values in the vertices of $T$.

We start by introducing the desired basis, and then we will show that the function space spanned by this basis has the desired properties. Since we know a $d$-parallelotope has $2^d$ vertices, we know we need $2^d$ basis functions.

### Basis for $\widehat{PW}^+_{\boldsymbol{k}}(T)$

To simplify notation, we will in this section assume the parallelotope is rectangular and aligned with the axes of $\mathbb{R}^d$. To extend this to any $d$-parallelotope one just have to do a coordinate transformation using the vectors spanning the parallelotope, taking care not to change the global direction of $\boldsymbol{k}$.

From lemma 4.3 we have $2^d$ functions which are in $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$. The problem with these are that if $k_i = 0$ for some $i \in [1, d]$, then all these functions will not be linearly independent. We do, however, have a solution to this. If $k_i = 0$ then, we know that a linear function in $x_i$ will be included in $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$. By combining these two properties we can define

$$\phi_r = \begin{cases} e^{ik_r x_r} & k_r \neq 0 \\ x_r & k_r = 0 \end{cases} \quad \forall r \in [1, d] \tag{4.5}$$

where $\boldsymbol{k} = [k_r]_{r\in[1,d]}$ and $\boldsymbol{x} = [x_r]_{r\in[1,d]}$.

Now, using lemma 4.4 we know that products of $\phi_r$ are in $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$ and we can define the basis functions

$$\psi \in \left\{ \prod_{r=1}^{d} (\phi_r)^{\alpha_r} \ : \ \alpha_r \in \{0, 1\} \right\} \tag{4.6}$$

and since there are $2^d$ unique ways to choose the values of $\{\alpha_r\}_{r\in[1,d]}$, we have $2^d$ basis functions, which is what we need.

**Definition 4.5.** Let $T$ be a rectangular $d$-parallelotope aligned with the axes of $\mathbb{R}^d$. Then $\widehat{PW}^+_{\boldsymbol{k}}(T)$ is

$$\widehat{PW}^+_{\boldsymbol{k}}(T) = \operatorname{span}\{\psi_s\}_{s\in[1,2^d]}$$

where $\phi_r$ are defined in (4.5),

$$\psi_s \in \left\{ \prod_{r=1}^{d} (\phi_r)^{\alpha_r} \ : \ \alpha_r \in \{0, 1\} \right\}$$

and all $\psi_s$ are distinct. We call a function in $\widehat{PW}^+_{\boldsymbol{k}}(T)$ a *finite plane wave*.

An example is the basis for $\widehat{PW}^+_{\boldsymbol{k}}(T)$ when T is a cube in 3D and all components of $\boldsymbol{k}$ are non-zero. It will consist of the following functions

$$\begin{aligned} \psi_1 &= 1 & \psi_2 &= e^{ik_1x_1} \\ \psi_3 &= e^{ik_2x_2} & \psi_4 &= e^{ik_3x_3} \\ \psi_5 &= e^{i(k_1x_1+k_2x_2)} & \psi_6 &= e^{i(k_1x_1+k_3x_3)} \\ \psi_7 &= e^{i(k_2x_2+k_3x_3)} & \psi_8 &= e^{i(k_1x_1+k_2x_2+k_3x_3)} \end{aligned}$$

### Properties of $\widehat{PW}_{\boldsymbol{k}}^{+}(T)$

A property we want from $\widehat{PW}_{\boldsymbol{k}}^{+}(T)$ is that the restriction of a function on $T$ to one of the faces $F$ of $T$ should be in the finite plane wave space on $F$.

**Lemma 4.6.** *Let $T$ be a rectangular $d$-parallelotope aligned with the axes of $\mathbb{R}^d$. Let $F$ be a face of $T$, and assume $u \in \widehat{PW}_{\boldsymbol{k}}^{+}(T)$. Then*

$$\left\{ v|_F \,:\, v \in \widehat{PW}_{\boldsymbol{k}}^{+}(T) \right\} = \widehat{PW}_{\boldsymbol{k}}^{+}(F)$$

*Proof.* Let $F$ be a face on $T$ with codimension 1. Since $T$ is aligned to $\mathbb{R}^d$, there exists a $s \in [1, d]$ such that $x_s = c_1$ on $F$ for some constant $c_1$. From (4.5) it follows that $\phi_s = c$ for some $c \in \mathbb{R}$.

Let $\{\psi_r\}_{r\in[1,2^d]}$ be the basis functions of $\widehat{PW}_{\boldsymbol{k}}^{+}(T)$ as defined in definition 4.5, and similarly let $\{\psi_r^F\}_{r\in[1,2^{d-1}]}$ be the basis functions of $\widehat{PW}_{\boldsymbol{k}}^{+}(F)$. It follows that

$$\psi_r^F \in \left\{ \prod_{\substack{r\in[1,d]\\ r\neq s}} (\phi_r)^{\alpha_r} \,:\, \alpha_r \in \{0,1\} \right\} \quad \psi_r|_F \in \left\{ c^{\alpha_s} \prod_{\substack{r\in[1,d]\\ r\neq s}} (\phi_r)^{\alpha_r} \,:\, \alpha_r \in \{0,1\} \right\}$$

hence, $\{\psi_r|_F\}_{r\in[1,2^d]}$ contains only differently scaled versions of the same functions as in $\{\psi_i|_F\}_{r\in[1,2^d]}$. This implies that

$$\operatorname{span}\{\psi_r|_F\}_{r\in[1,2^d]} = \operatorname{span}\left\{\psi_r^F\right\}_{r\in[1,2^{d-1}]}$$

and the result follows for faces of codimension 1. Since every face with codimension $> 1$ will be a face of a face with codimension 1, the argument can be applied recursively, and the result follows. □

The last property of $\widehat{PW}_{\boldsymbol{k}}^{+}(T)$ we need is that for any set of values in the vertices of $T$, there is a single $u \in \widehat{PW}_{\boldsymbol{k}}^{+}(T)$ such that the function values in the vertices are exactly these values. For this to be true we cannot allow the length of the parallelotope in direction $r$ to be $\frac{2\pi}{k_r}$ when $k_r \neq 0$. The reason for this requirement is that $\phi_r(x + \frac{2\pi}{k_r}) = \phi_r(x)$, forcing the two endpoints of this line to have the same value.

**Lemma 4.7.** *Let $T$ be a rectangular $d$-parallelotope aligned with the axes of $\mathbb{R}^d$ and where the length along the $r$-th axis is not $\frac{2\pi n}{k_r}$ for $n \in \mathbb{N}$ when $k_r \neq 0$. Then, for any set of values in the vertices of $T$ there exists an unique $u \in \widehat{PW}_{\boldsymbol{k}}^{+}(T)$ which has these values as function values in the respective vertices.*

*Proof.* We will prove this by induction. Let $T$ be a 1-parallelotope along the $r$-axis. The vertices of this line are the endpoints, $x_a$ and $x_b$. Then we have only one function $\phi_r$ on the form of 4.5, making two basis functions, $\psi_1 = 1$ and $\psi_2 = \phi_r$. Since we have assumed that the length of the line cannot be $2\pi n/k_r$ for $n \in \mathbb{N}$, we know that $\phi_r(x_a) \neq \phi_r(x_b)$. This means there is exactly one $u \in \widehat{PW}_{\boldsymbol{k}}^+(T)$ such that $u(x_a) = a$ and $u(x_b) = b$ for any given values $a, b \in \mathbb{C}$.

Assume the statement holds for $d < k$ and let $T$ be a $k$-parallelotope. From this assumption we know that on every face of $T$ there is a finite plane wave uniquely determined by the values in the vertices of $T$. We know from lemma 4.6 that for any face $F$ of $T$, any function in $\widehat{PW}_{\boldsymbol{k}}^+(F)$ is the restriction of some function in $\widehat{PW}_{\boldsymbol{k}}^+(T)$. This means that any set of values in the vertices corresponds to a function in $\widehat{PW}_{\boldsymbol{k}}^+(T)$. Since there are $2^k$ vertices and $\dim \widehat{PW}_{\boldsymbol{k}}^+(T) = 2^k$ we know, that the function is uniquely determined. □

One easy way to ensure that the size of the parallelotope does not interfere with the choice of basis functions is to require that $\operatorname{diam} T \leq h < \frac{2\pi}{|\boldsymbol{k}|}$, which is not a very strict requirement. This is not an optimal bound, but it shows how making the grid fine enough removes this concern. If the method is to be applied on very coarse grids, then other strategies to avoid the problem may be developed, as it is only the very specific values that have to be avoided.

### 4.1.3 The PWCSG element

We will now define an element $(T, X_T, \Sigma_T)$ for the plane wave method. First we set $X_T = \widehat{PW}_k^+(T)$. Since we already have shown that any function in $X_T$ is uniquely defined by the values in the vertices of a $d$-parallelotope, it is natural to use these values as degrees of freedom, which is what we do.

This provides us with the structure of $T$, the function space $X_T$ and the set of dofs $\Sigma_T$. What remains is to find a basis $\phi_i$ such that

$$u = \sum_{i=1}^{m_T} \phi_i \sigma_T^i(u) \quad \forall u \in X_T$$

which would be a nodal basis in the vertices of the parallelotope. Since we already have a basis $\{\psi_j\}_{j\in[1,m_T]}$ for $X_T$, we can write

$$\phi_i(x) = \sum_{j=1}^{m^T} d_i^j \psi_j(x)$$

and if we let $x_T^1, \dots x_T^{m_T}$ be the vertices, we can then solve

$$\sum_{j=1}^{m^T} d_i^j \psi_j(x_T^k) = \delta_{ik} \quad \forall i, k \in [1, m_T] \tag{4.7}$$

for $d_i^j$. This will generate the desired basis. While (4.7) can be solved as $m_T$ linear $m_T \times m_T$ systems of equations, solving it symbolically for a general parallelotope results in very long expressions. Even tough, we now have a simple implicit definition and also, if needed, an explicit definition of the basis functions.

We can now define the Plane Wave Semi-Continuous Galerkin element

**Definition 4.8** (The PWSCG element)**.** The Plane Wave Semi-Continuous Galerkin element in $\mathbb{R}^d$ is defined by the triple $(T, X_T, \Sigma_T)$ where

(i) $T$ is a $d$-parallelotope

(ii) $X_T = \widehat{PW}_{\boldsymbol{k}}^+(\mathbb{R}^d)$ for some $\boldsymbol{k}$

(iii) $\Sigma_T = \{\sigma_i(v) : X_T \to \mathbb{C} \,:\, \sigma_i(v) = v(x_i)\}$ where $\{x_i\}$ is the vertices of $T$

### 4.1.4 Discontinuity

We will now look at whether the method conforming. Since we have a nodal basis and we know that the value on the edges are uniquely identified by the nodes on that edge, we only have to satisfy condition (3.7) stating that the function space on the edge have to be unique, and (3.8) which states that any node has a equivalent node on every neighbour element. Since the dofs are located in the corners, (3.8) will be satisfied as long as we have no hanging nodes. The second requirement will be satisfied if and only if $\boldsymbol{k}$ is the same on all elements. Since we want to support different $\boldsymbol{k}$ on different elements, this is not in general satisfied, leading to discontinuities in the method.

To justify why the method still works, we look at it in the framework of discontinuous Galerkin. Since every node has a corresponding node on every neighbour element, we may maintain the requirement that the values in these points are the same. This effectively reduces the number of dofs to the number used by the continuous Galerkin method, eliminating one of the disadvantages of using discontinuous Galerkin. Using this assumption we now know that the function is continuous in each node. Since we require the functions to be continuous in each node, we call this method semi-continuous.

In section (3.2.4) we derived the primal formulation for the Helmholtz equation, splitting it into two parts, $a_c(u, v)$ which has the same form as the

continuous problem, and $a_d(u,v)$ containing the extra boundary integrals. We defined $a_d$ as follows

$$a_d(u,v) = \int_{\Gamma^0} \left( [\![ \hat{u} - u ]\!] \cdot \{ \overline{\operatorname{grad} v} \} - \{ \hat{\boldsymbol{\mu}} \} \cdot [\![ \overline{v} ]\!] \right) \mathrm{d}s + \int_{\Gamma^0} \left( \{ \hat{u} - u \} [\![ \overline{\operatorname{grad} v} ]\!] - [\![ \hat{\boldsymbol{\mu}} ]\!] \{ \overline{v} \} \right) \mathrm{d}s$$

When introducing the flux functions we replaced $u$ by $\widehat{u}$, and $\boldsymbol{\mu}$ which is an approximation of $\operatorname{grad} u$, with $\widehat{\boldsymbol{\mu}}$. If we choose the fluxes to just be this substitution in reverse, we get

$$\begin{aligned} a_d(u,v) &= \int_{\Gamma^0} \left( [\![ u - u ]\!] \cdot \{ \overline{\operatorname{grad} v} \} - \{ \operatorname{grad} u \} \cdot [\![ \overline{v} ]\!] \right) \mathrm{d}s \\ &\quad + \int_{\Gamma^0} \left( \{ u - u \} [\![ \overline{\operatorname{grad} v} ]\!] - [\![ \operatorname{grad} u ]\!] \{ \overline{v} \} \right) \mathrm{d}s \\ &= - \int_{\Gamma^0} \left( \{ \operatorname{grad} u \} \cdot [\![ \overline{v} ]\!] + [\![ \operatorname{grad} u ]\!] \{ \overline{v} \} \right) \mathrm{d}s \end{aligned}$$

Let $T_r, T_s \in \mathcal{T}$ such that $e = T_r \cap T_s$, $e \neq \emptyset$, and assume $u^r, v^r \in U_{T_r}$ and $u^s, v^s \in U_{T_s}$ then we rewrite the part of the integral on $e$ as

$$- \int_e \left( \frac{1}{2} (\operatorname{grad} u^r + \operatorname{grad} u^s) \cdot (\overline{v^r \boldsymbol{n}_{T_r} + v^s \boldsymbol{n}_{T_s}}) + \frac{1}{2} (\operatorname{grad} u^r \cdot \boldsymbol{n}_{T_r} + \operatorname{grad} u^s \cdot \boldsymbol{n}_{T_s}) (\overline{v^r + v^s}) \right) \mathrm{d}s$$

If we compute this integral, we will get that for any $e \in \Gamma^0$ and for $U_{T_r} = \widehat{PW}^+_{\boldsymbol{k}_1}(T_r)$ and $U_{T_s} = \widehat{PW}^+_{\boldsymbol{k}_2}(T_s)$ where $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are any vectors, this integral will be zero. This is also true when one of the functions is an affine function on the edge which happens when one of the $\boldsymbol{k}$s has value zero in the component tangential to $e$. Also, if both function spaces contain functions which are linear on $e$ then the function will be continuous and all the jumps will be zero, thus making the integral zero. Since this holds for all $e$ we conclude that

$$a_d(u,v) = 0 \quad \forall u, v \in U_h$$

and this semi-continuous Galerkin method will be exactly the same as the corresponding continuous method assuming the integrals are computed per-element and not globally.

It is important to note that this does not automatically ensure convergence. Convergence analysis for discontinuous Galerkin is a complex subject,

and while there exists some results that cover a wide range of methods [2], I have not found any results covering the case where we assume continuity in the nodes. How to derive such a result is not obvious and outside the scope of this thesis.

While applying discontinuous Galerkin methods to this problem is not in any way a proof of convergence, it does give us a mathematical backing of the method which the continuous analysis does not cover. It may also be a starting point to actually proving that the method will converge despite the discontinuities. It also gives us some direct insight trough what flux functions we used.

With the fluxes $\widehat{u} = u$ and $\widehat{\boldsymbol{\mu}} = \operatorname{grad} u$ we obviously have that the fluxes are local and consistent. They are not, however, conservative, which indicates that when using the method to simulate physical phenomenas, we cannot expect the it to preserve the total amount of energy in the system.

### 4.1.5 Real-valued solutions

One shortcoming of this method worth noting is it does not handle real-valued solutions of Helmholtz equation as well as the complex solutions. If we look at the function

$$u = \sin(\boldsymbol{k} \cdot \boldsymbol{x})$$

then, obviously

$$\Delta \sin(\boldsymbol{k} \cdot \boldsymbol{x}) + \boldsymbol{k}^2 \sin(\boldsymbol{k} \cdot \boldsymbol{x}) = -\boldsymbol{k}^2 \sin(\boldsymbol{k} \cdot \boldsymbol{x}) + \boldsymbol{k}^2 \sin(\boldsymbol{k} \cdot \boldsymbol{x}) = 0$$

hence $u \in PW_{\boldsymbol{k}}(\mathbb{R}^d)$, but it is not in $u \in PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$. This is because we can write

$$u = \sin(\boldsymbol{k} \cdot \boldsymbol{x}) = \frac{e^{i\boldsymbol{k}\cdot\boldsymbol{x}} - e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}}{2i}$$

and, as we noted in section 4.1.1, $e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$ is not included in $PW^+_{\boldsymbol{k}}(\mathbb{R}^d)$.

## 4.2 Implementation

To test the method an implementation had to be made. Since this method has many aspects that differ from the most common methods there was no high-level, straight forward way of implementing it. I decided the best way to approach the problem was to find a FEM-framework which was modularized in a way that allows the user to utilize the modules that are compatible with the problem and reimplement the unique parts needed for this implementation. I found most FEM-frameworks are implemented in a compiled language and providing language bindings in higher level languages

like Python or Matlab. Because of the way this method differs from normal methods, it needs access to the internals of the framework, and hence the language in which to make the implementation would need to be the main implementation language of the framework.

### 4.2.1 Framework

There are many FEM-frameworks to choose from, but for this project the C++-based framework GetFEM++ [10] seemed most suitable, mainly because of the way it handles the finite element spaces. While other frameworks like FeniCS[3] handle basis functions by an offline compilation process [13] or other sophisticated processes making it hard to specify general parameters of the bases on each individual element, GetFEM++ has a much simpler implementation allowing the user of the library to implement functions returning function values of the basis functions at given points. This makes it much easier to implement basis functions where coefficients of the functions vary from element to element depending on the problem to be solved.

### 4.2.2 Complex basis functions

GetFEM++ does have some limitations that had to be circumvented. Firstly, it does not support complex basis functions. Since the method requires them, I started investigating what it would take to change GetFEM++ to allow them, but since the basis values are used throughout the code and their type always hardcoded to be real numbers, this was not an option. The other option was to make the assembly expressions manually handle the complex inner products.

Let $V$ be a complex finite element space with

$$\operatorname{span}_{\mathbb{C}} \{\phi_1, \phi_2, \ldots, \phi_n\} = V$$

Then introduce

$$V_{\text{re}} = \{\operatorname{Re} v \,:\, v \in V\}$$
$$V_{\text{im}} = \{\operatorname{Im} v \,:\, v \in V\}$$

then, obviously

$$V_{\text{re}} = \operatorname{span}_{\mathbb{R}} \{\operatorname{Re} \phi_1, \operatorname{Re} \phi_2, \ldots, \operatorname{Re} \phi_n\}$$
$$V_{\text{im}} = \operatorname{span}_{\mathbb{R}} \{\operatorname{Im} \phi_1, \operatorname{Im} \phi_2, \ldots, \operatorname{Im} \phi_n\}$$

are also finite element spaces. Now we can decompose every $v \in V$ into $v = v^{\text{re}} + iv^{\text{im}}$ for some $v^{\text{re}} \in V_{\text{re}}$ and $v^{im} \in V_{\text{im}}$. We also see by looking at the definition of $a$ that

$$a(iu, v) = ia(u, v) \qquad\qquad a(u, iv) = -ia(u, v)$$

when $u, v$ are real valued, similar to the behaviour of an inner product. Using this we can expand the complex assembly expressions into two real ones

$$\begin{aligned} a(\phi_i, \phi_j) &= a(\phi_i^{\text{re}} + i\phi_i^{\text{im}}, \phi_j^{\text{re}} + i\phi_j^{\text{im}}) \\ &= a(\phi_i^{\text{re}}, \phi_j^{\text{re}}) + a(\phi_i^{\text{im}}, \phi_j^{\text{im}}) + i\left(a(\phi_i^{\text{im}}, \phi_j^{\text{re}}) - a(\phi_i^{\text{re}}, \phi_j^{\text{im}})\right) \end{aligned}$$

and when this is assembled into a matrix, two matrices are used, one for the real part and one for the complex part

$$\begin{aligned} M_{i,j}^{\text{re}} &= \operatorname{Re}\left(a(\phi_i, \phi_j)\right) = a(\phi_i^{\text{re}}, \phi_j^{\text{re}}) + a(\phi_i^{\text{im}}, \phi_j^{\text{im}}) \\ M_{i,j}^{\text{im}} &= \operatorname{Im}\left(a(\phi_i, \phi_j)\right) = a(\phi_i^{\text{im}}, \phi_j^{\text{re}}) - a(\phi_i^{\text{re}}, \phi_j^{\text{im}}) \end{aligned}$$

This can then be solved either by using a solver that handles the complex matrix $M = M^{\text{re}} + iM^{\text{im}}$ directly, or by observing that a system

$$\left(M^{\text{re}} + iM^{\text{im}}\right)\left(c^{\text{re}} + ic^{\text{im}}\right) = \left(b^{\text{re}} + ib^{\text{im}}\right)$$

can be expanded to two systems

$$\begin{aligned} M^{\text{re}}c^{\text{re}} - M^{\text{im}}c^{\text{im}} &= b^{\text{re}} \\ M^{\text{im}}c^{\text{re}} + M^{\text{re}}c^{\text{im}} &= b^{\text{im}} \end{aligned}$$

which can be reconnected

$$\begin{bmatrix} M^{\text{re}} & -M^{\text{im}} \\ M^{\text{im}} & M^{\text{re}} \end{bmatrix} \begin{bmatrix} c^{\text{re}} \\ c^{\text{im}} \end{bmatrix} = \begin{bmatrix} b^{\text{re}} \\ b^{\text{im}} \end{bmatrix}$$

and solved with a normal real solver.

### 4.2.3 Implementing function spaces

Implementing the $\widehat{PW}_{\boldsymbol{k}}^{+}(\mathbb{R}^d)$ function space posed several problems. First, most finite element frameworks use the notion of a reference element $\widehat{T}$ such that $\operatorname{diam}\widehat{T} = 1$. Then they define the basis $\{\widehat{\phi}_j\}_{j\in[1,m_{\widehat{T}}]}$ on the reference element and then assume that for an element $T \in \mathcal{T}$ with basis $\{\phi_j\}_{j\in[1,m_T]}$ we have

$$\phi_j(\boldsymbol{x}) = \widehat{\phi}_j(G\boldsymbol{x})$$

where $G : T \to \widehat{T}$ is an affine transform. The framework can then compute the integrals only on the reference element, doing the appropriate scaling according to the affine transformation to derive the values of the integrals on each element.

This method assumes the function spaces are invariant to affine transforms. This is true for polynomials since if $p \in \mathcal{P}_k$ and $G\boldsymbol{x} = G_l\boldsymbol{x} + \boldsymbol{x}_0$ then

$$p(G\boldsymbol{x}) = \sum_{j=0}^{k} c_j (G\boldsymbol{x})^j = \sum_{j=0}^{k} c_j (G_l\boldsymbol{x} + \boldsymbol{x}_0)^j \in \mathcal{P}_k$$

hence, it still element a polynomial in $\boldsymbol{x}$ of degree $\leq k$.

This is, however, not the case for $\widehat{PW}^+_{\boldsymbol{k}}(\mathbb{R}^d)$. Let $e^{i\boldsymbol{k}\cdot\boldsymbol{x}} \in \widehat{PW}^+_{\boldsymbol{k}}(\mathbb{R}^d)$ and $G\boldsymbol{x} = G_l\boldsymbol{x} + \boldsymbol{x}_0$ as before. Then

$$e^{i\boldsymbol{k}\cdot G\boldsymbol{x}} = e^{i\boldsymbol{k}\cdot(G_l\boldsymbol{x}+\boldsymbol{x}_0)} = e^{i\boldsymbol{k}\cdot G_l\boldsymbol{x}+i\boldsymbol{k}\cdot\boldsymbol{x}_0)} = e^{i\boldsymbol{k}\cdot\boldsymbol{x}_0} e^{iG_l\boldsymbol{k}\cdot\boldsymbol{x}} \in \widehat{PW}^+_{G_l\boldsymbol{k}}(\mathbb{R}^d)$$

which is not the same space. This can be solved either by making special assembly routines taking this effect into account, or we can avoid using reference element altogether. Since deriving an integral between two $\widehat{PW}^+_{\boldsymbol{k}}(\mathbb{R}^d)$ functions with different $\boldsymbol{k}$ from the integral on a reference element is not straight forward, the implementation uses the second method, not using a reference element at all.

To evaluate the basis functions the coefficients $d_i^j$ from (4.7) need to be determined. Since these depend on $\boldsymbol{k}$ and the location of the corners of the given element, they have to be computed for every element. In the current implementation this is done by solving the linear system directly for each basis function.

### 4.2.4 Dirichlet condition

The Dirichlet condition is handled in the continuous case by limiting the test and trial spaces to functions with zero trace, then using a boundary function to handle non-homogeneous boundary conditions. This may be done in the discrete case by making sure no basis function has a non-zero value on the boundary, but GetFEM++, nor any other finite element framework I have worked with, facilitate this approach.

Instead there are two main strategies for implementing Dirichlet boundary conditions. One is direct manipulation of the linear system, the other is to transform a weak formulation of the condition into an underdetermined linear system and solving the internal nodes in the kernel of the boundary condition system.

Since finding a boundary function $b$ with trace $g$ is not trivial, we will here assume the inhomogeneous problem

$$u = g \quad \text{on } \partial\Omega$$

or, the weak form

$$\int_{\partial\Omega} u\overline{v}\,\mathrm{d}s = \int_{\partial\Omega} g\overline{v}\,\mathrm{d}s \quad \forall v \in H^1(\Omega)$$

and solve this directly, not trough the use of a boundary function.

### Direct manipulation

Let us first examine the direct manipulation approach. We assume the finite element space $X_h$ has a nodal basis $\{\phi_i\}$. Also let

$$u(\boldsymbol{x}) = \sum_{i=1}^{m} c_i \phi_i(\boldsymbol{x})$$

and since $\{\phi_i\}$ is nodal, we have points $\{\boldsymbol{x}_i\}$ such that $u(\boldsymbol{x}_i) = c_i$. Let $k$ be the index of a dof on the boundary $\boldsymbol{x}_k \in \partial\Omega$, and let $g_k = g(\boldsymbol{x}_k)$. This means the $k$-th linear equation in the system is the following

$$\sum_{i=1}^{m} a(\phi_i, \phi_k) c_i = l(\phi_k)$$

but since $\phi_k(\boldsymbol{x}_k) = 1$ it should not be included in our test space to begin with, we can repurpose this equation for our boundary condition. From the boundary condition we have

$$\begin{aligned} u(\boldsymbol{x}_k) &= g(\boldsymbol{x}_k) \\ c_k &= g_k \\ \sum_i \delta_{ik} c_i &= g_k \end{aligned}$$

Now let $Ac = b$ be the linear system resulting from the Galerkin method without taking boundary condition into account. Replacing the $k$-th equation translates to replacing the $k$-th row of $A$ with

$$a_{k,i} = \delta_{ik} \quad \forall i \in [1, m]$$

and doing this for every $k$ for which $\boldsymbol{x}_k \in \partial\Omega$ will remove all the test functions which should not be included, combined with enforcing the boundary condition for every point on the boundary.

There is one common extension to the above method to allow the resulting system to remain Hermitian. This consists of using row reduction techniques to eliminate the non-zero elements of column $k$ such that

$$a_{i,k} = \delta_{ik} \quad \forall i \in [1, m]$$

Now that both row $k$ and column $k$ are replaced with the $k$-th unit vector, the matrix remains Hermitian if the original matrix was Hermitian, which allows the use of faster linear solvers.

A weakness of this method is that the boundary values $g$ will be interpolated, rather than approximated by Galerkin orthogonality. When dealing with low-order polynomial elements this is usually not be a problem, but it may introduce an error when dealing with more complex elements.

### Weak Dirichlet condition

Plane Wave Continuous Galerkin methods may handle functions where the value oscillate multiple times between each node. Combining this with the fact that the values of $\boldsymbol{k}$ are approximations may lead to more inaccurate results when using interpolation, as in the example given in figure 4.1.

To approximate the Dirichlet condition by Galerkin method instead of interpolation, we need to use the weak formulation of the Dirichlet condition and substitute the continuous space $H^1(\Omega)$ by our finite space $X_h$, resulting in the linear system

$$Hc = R$$

where $H = \{h_{i,j}\}$, $R = \{r_j\}$

$$h_{i,j} = \int_{\partial\Omega} \phi_i \overline{\phi}_j \, \mathrm{d}s \tag{4.8}$$

and

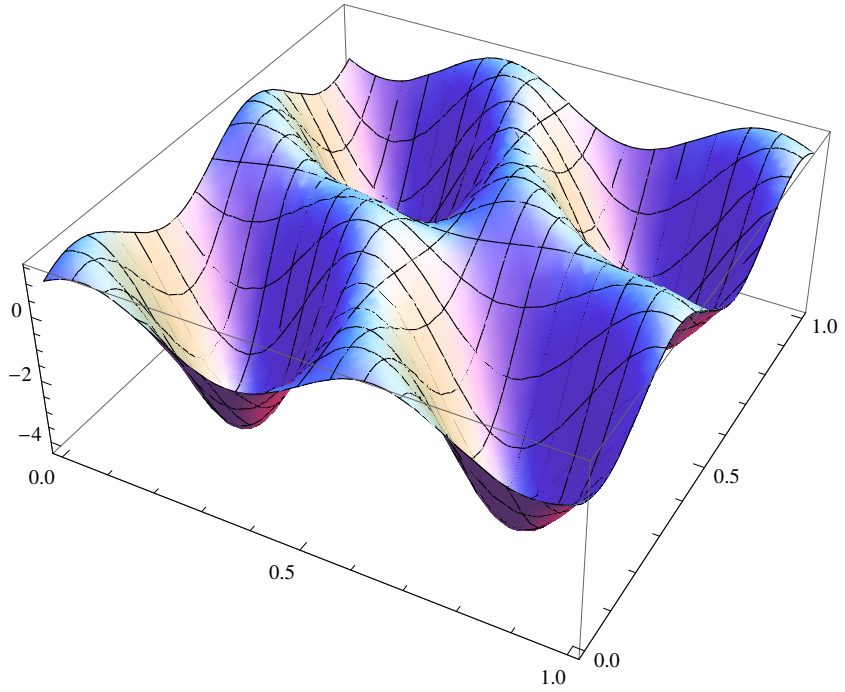
$$r_j = \int_{\partial\Omega} g \overline{\phi}_j \, \mathrm{d}s$$

Now let $Ac = b$ be the linear system generated by the Galerkin method when not taking Dirichlet conditions into account. To prevent conflicts we have to assume that a well posed problem has the property

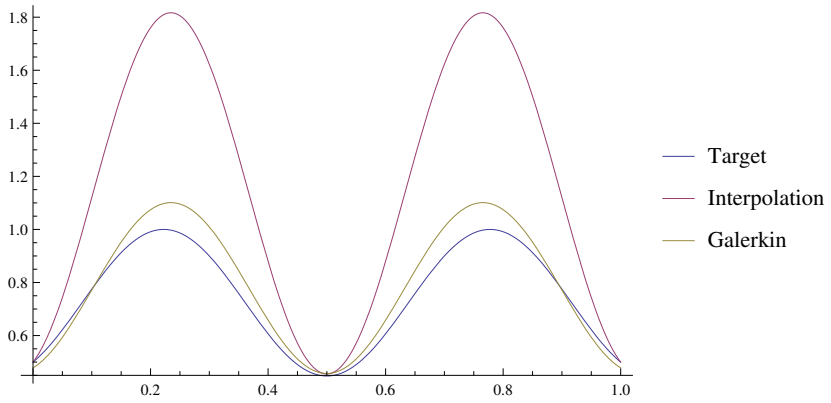
$$\mathrm{im}A = (\mathrm{im}H)^{\perp} \tag{4.9}$$

which implies

$$\mathrm{rank}\, A + \mathrm{rank}\, H = m \tag{4.10}$$




(a) The target function on the unit square.




(b) Intersection at $y = \frac{1}{2}$ of the target, the interpolated approximation and the Galerkin approximation. The Galerkin approximation is clearly closer to the target. Let $u_t$ be the target, $u_i$ be interpolated approximation and $u_g$ be Galerkin approximation. Then $\|u_i - u_t\|_{L^2(T)} = 0.731882$ and $\|u_a - u_t\|_{L^2(T)} = 0.411986$.

Figure 4.1: This figure shows an example of how interpolation compares to Galerkin approximation when using plane waves. Both approximations use a $\boldsymbol{k}$ which is rotated about 2.7° compared to the $\boldsymbol{k}$ of the target. Interpolation points are taken in the four corners of the square.

To ensure uniqueness we will also assume

$$\ker A = (\ker H)^{\perp} \tag{4.11}$$

These requirements are natural since the Dirichlet condition is meant to determine the dofs on the boundary that the equation would leave undetermined.

Now choose any $c^d = [c_i^d]_{i\in[1,m]}$ such that $Hc^d = R$, then

$$u_d(x) = \sum_{i=1}^{m} c_i^d \phi_i(x) \tag{4.12}$$

satisfies

$$\int_{\partial\Omega} u_d \overline{v}\,\mathrm{d}s = \int_{\partial\Omega} g\overline{v}\,\mathrm{d}s \quad \forall v \in X_h$$

Then let the columns of $N$ be an orthonormal basis for the kernel of $H$ and let $r = \dim\ker H$. We may then use the $u_d$ from (4.12) to split $u$

$$u = u_s + u_d$$

and since $u$ should satisfy the Dirichlet condition we deduce

$$R = Hu = Hu_s + Hu_d = Hu_s + R \Rightarrow u_s \in \ker H$$

and there exists an unique $u_n$ such that

$$Nu_n = u_s$$

Since we assume $u_d$ is known, we now only have to find $u_n$ to determine $u$. Since $u_n \in \mathbb{R}^r$ and we know from (4.10) that $\operatorname{rank} A = r$, we want to reduce the original system $Ac = b$ to a $r \times r$ full rank system of equations in $u_n$, and then we will have the solution given by $u = Nu_n + u_d$. We start by rewriting the original system

$$\begin{aligned} Au &= b \\ ANu_n + Au_d &= b \\ ANu_n &= b - Au_d \end{aligned} \tag{4.13}$$

Finding $\operatorname{rank}(AN)$ is easy since

$$\mathrm{im}N = \ker H = (\ker A)^{\perp}$$

so $\operatorname{rank}(AN) = r$. We can use assumption (4.9) to eliminate the redundant equations and reduce the system to a square system with $u_n$ as unknown. From (4.8) it is trivial to see that $H = H^*$. From theory of linear functionals [15, Lemma 6.11] we have the lemma

**Lemma 4.9.** *Let $H$ and $K$ be Hilbert spaces and let $T : H \to K$ be a linear functional. Then*

*(i)* $\ker T = (imT^*)^\perp$

*(ii)* $\ker T^* = (imT)^\perp$

Using this we have

$$\begin{aligned} \mathrm{im}A &= (\mathrm{im}H)^\perp && \text{from assumption (4.9)} \\ &= (\mathrm{im}H^*)^\perp && \text{since } H \text{ is Hermitian} \\ &= \ker H && \text{from lemma 4.9} \\ &= \mathrm{im}N && \text{from the definition of } N \\ &= (\ker N^*)^\perp && \text{from lemma 4.9} \end{aligned}$$

which implies that $N^* : \mathrm{im}A \to \mathbb{R}^r$ is injective which again implies

$$\mathrm{rank}\, N^*AN = r$$

and hence we have arrived at the $r \times r$ linear system with full rank

$$N^*ANu_n = N^*(b - Au_d)$$

Given the assumptions, we will always have a unique solution $u = Nu_n + u_d$. This method is more versatile than direct manipulation as it works for any elements, not just for nodal elements. Also, since

$$(N^*AN)^* = N^*A^*N$$

we have that $N^*AN$ is Hermitian if $A$ is.

### Simplification for nodal elements

Finding the kernel of $H$ can be computationally expensive, usually in the order of $\mathcal{O}(m^3)$ operations. This is considerably worse than direct manipulation which requires only $\mathcal{O}(m)$ operations. However, this is not fair comparison since direct manipulation only works with nodal elements. In the case of these simple elements we can assume that the kernel of $H$ is spanned by a subset of the canonical basis of $\mathbb{R}^m$ corresponding to the dof numbers on the boundary, and it can be computed in $\mathcal{O}(m)$ operations as well.

# Chapter 5

# Numerical Results

Because of the discontinuities in the discrete space, we cannot use continuous Galerkin theory to ensure convergence of the PWSCG method, and because of the way we assure continuity in all the nodes, we cannot use normal discontinuous Galerkin error bounds either. Hence, we have no analytical proof of how the method will behave, only assumptions based on similarities with the two mentioned methods.

In this chapter we will look at a posteriori error estimates based on numerical results which will give an indication of how well we can expect the method to perform. There are many ways to perform a posteriori error estimates, but since we will only look at cases where the analytical solution is known, the $L^2$-norm of the difference between the exact and the numerical solution will be used.

The equation we solve will always be Helmholtz equation (2.1) on the unit cube with the values of $k$ and $f$ given for each test case. The values used for $\boldsymbol{k}$ in the PWSCG elements will also be given. We will also compare the PWSCG-method to standard $Q_1$ elements since these methods have the same number of dofs and can be applied to the same mesh as the PWSCG-method, thus giving the most fair comparison. $Q_1$ elements are also what we get when choosing $\boldsymbol{k} = 0$ in the PWSCG-method since (4.4) then reduces to $\Delta u = 0$.

Since we will compare the numerical approximations to a known solution, we will set the Dirichlet boundary condition to enforce the correct values on the boundary.

| Mesh Size | PWSCG elements $L^2$-error | $Q_1$ elements $L^2$-error |
|---|---|---|
| $2 \times 2 \times 2$ | 0.00000000 | 0.00000000 |
| $4 \times 4 \times 4$ | 0.00000005 | 0.00000005 |
| $8 \times 8 \times 8$ | 0.00000004 | 0.00000040 |
| $16 \times 16 \times 16$ | 0.00000056 | 0.00000058 |
| $32 \times 32 \times 32$ | 0.00000067 | 0.00000067 |

Table 5.1: Error of PWSCG and $Q_1$ methods when approximating $u = 1$ which is included in the discrete space for both methods.

## 5.1 Exact approximation

From Céa's lemma 3.3, we have that the error of a continuous method is bounded by the best approximation error, and similar results also exist for many discontinuous Galerkin methods [2, p. 1767]. One consequence of this is that when the exact solution is included in our discrete space, then the method will return that solution. Since the PWSCG-method is not covered by any of these results, we will numerically verify that this is the case for some test cases to conclude that it is plausible that it will be true for any solution included in the discrete space.

First, we look at a simple case. Let $k = 3$, $f = 9$. Then obviously $u = 1$ will be a solution. Since constants have no wave-like behaviour the value of $\boldsymbol{k}$ should not matter, and we choose $\boldsymbol{k} = [1, 1, 1]$ for simplicity. Also, since constants are included in the discrete space for both PWCSG and $Q_1$ we may expect similar results.

For $Q_1$ we know the solution must be the exact solution up to some round-off error arising from the finite precision of floating point numbers. Looking at the results in table 5.1 we see that the PWCSG-method handles this case just as well as $Q_1$ elements does. This is no surprise since when $\boldsymbol{k}$ is constant throughout the domain, then the PWSCG-method is also continuous and covered by Céas lemma and its generalizations.

Constants are a particularly simple solution, even when only looking at problems where the exact solution is in the discrete space. In the next case we let the solution be a linear combination of functions we know are in $PW^+_{\boldsymbol{k}}$. We will use $\boldsymbol{k} = [5, 3, 1]$, which means $k = \sqrt{35}$ and $f = 0$. When setting the value of $\boldsymbol{k}$ used in the PWSCG-method equal to the value in the exact solution, then the exact solution is in the discrete space of PWSCG, but not for $Q_1$.

For completeness we will also look at a case where $\boldsymbol{k}$ is the same, but

| Mesh Size | PWSCG elements | $Q_1$ elements | |
|---|---|---|---|
| | $L^2$-error | $L^2$-error | Order |
| $2 \times 2 \times 2$ | 0.00000001 | 1.07994324 | |
| $4 \times 4 \times 4$ | 0.00000137 | 0.26100418 | 2.05 |
| $8 \times 8 \times 8$ | 0.00000071 | 0.05684609 | 2.20 |
| $16 \times 16 \times 16$ | 0.00000341 | 0.01409821 | 2.01 |
| $32 \times 32 \times 32$ | 0.00000848 | 0.00354906 | 1.99 |

Table 5.2: Error of PWSCG and $Q_1$ methods when approximating $u = e^{i[5,3,1]\cdot \boldsymbol{x}}$ which is a plane wave in the discrete space for PWSCG-method, but has no special relation to the $Q_1$ function spaces.

| Mesh Size | PWSCG elements | $Q_1$ elements | |
|---|---|---|---|
| | $L^2$-error | $L^2$-error | Order |
| $2 \times 2 \times 2$ | 0.00000000 | 0.00950955 | |
| $4 \times 4 \times 4$ | 0.00000081 | 0.00234677 | 2.02 |
| $8 \times 8 \times 8$ | 0.00000534 | 0.00058417 | 2.01 |
| $16 \times 16 \times 16$ | 0.00000304 | 0.00014590 | 2.00 |
| $32 \times 32 \times 32$ | 0.00001198 | 0.00003811 | 1.94 |

Table 5.3: Error of PWSCG and $Q_1$ methods when approximating $u = e^{ix_3} + 7 + 11i$ which is a linear combination of the basis functions introduced in $PW^+_{\boldsymbol{k}}(T)$ witch are not plane waves along $\boldsymbol{k}$.

where we set $k = 1$ and the exact solution to be $u = e^{ix_3} + 7 + 11i$. In this case we know that $u \in PW^+_{\boldsymbol{k}}$ even tough $|\boldsymbol{k}| \neq k$.

As we see in tables 5.2 and 5.3, the results are as expected. For the PWSCG-method, where the exact solution is in the discrete space, we get errors with the same behaviour as in the constant case, only with somewhat larger round-off errors. For $Q_1$ elements there is nothing special about this solution, hence we get second order convergence which is what we can expect from normal convergence analysis for continuous Galerkin.

One thing we can derive from these results is what kind of round-off errors we should expect with this method. The finite element method computes integrals on each element of the domain. This means that the intermediate values of the integrals are smaller for smaller elements. Hence, we have to expect more significant round-off errors on smaller elements. This seems to be reflected in the results, both for PWSCG-method and the normal $Q_1$ method. As we see from the results, we have to expect round-off error in the

| Mesh size | PWSCG elements $L^2$-error | $Q_1$ elements $L^2$-error |
|---|---|---|
| $2 \times 2 \times 2$ | 0.00000002 | 0.00000001 |
| $4 \times 4 \times 4$ | 0.00000078 | 0.00000018 |
| $8 \times 8 \times 8$ | 0.00000097 | 0.00000085 |
| $16 \times 16 \times 16$ | 0.00000256 | 0.00000279 |
| $32 \times 32 \times 32$ | 0.00000368 | 0.00000381 |

Table 5.4: Error of PWSCG and $Q_1$ methods when approximating $u = 3$ for non-constant $\boldsymbol{k}$ resulting in a truly discontinuous PWSCG-method.

order of $10^{-5}$ for $32 \times 32 \times 32$ grid. It can be useful to keep that in mind when looking at more complex results.

Finally, we will test two cases where $\boldsymbol{k}$ varies throughout the domain. In these cases the method will actually be discontinuous thus invalidating any analytical results we have used so far. One difficulty with testing such cases is that the discrete space will mostly contain discontinuous functions which cannot be analytical solutions since the analytical problem is not well posed for discontinuous functions. An exception is constant functions which will be included in all the element spaces even if $\boldsymbol{k}$ is different. Let $\boldsymbol{k} = [y + 1, x + y + 1, z + 3]$ and $u = 3$. This gives

$$k = \sqrt{(y+1)^2 + (x+y+1)^2 + (z+3)^2}$$
$$f = 3\left((y+1)^2 + (x+y+1)^2 + (z+3)^2\right)$$

The results of this test can be found in table 5.4.

The other case we can test is when $\boldsymbol{k}$ is constant along one axis, making continuous plane waves in only that direction part of the discrete space. We choose $\boldsymbol{k}(x_1, x_2, x_3) = [2, x_2x_3 + 1, x_2 + 3]$, $k = 2$ and $f = 0$ making the solution $u(x_1, x_2, x_3) = e^{i2x_1}$ both a solution of the equation and part of the discrete space. The errors of this approximation is found in table 5.5.

If we look at table 5.4 and 5.5 we see that the behaviour and magnitude of the errors are still the same as for the simpler cases. This can be regarded as numerical evidence that the semi-continuous method will find the exact solution if it exists in the discrete space, and it may also hint that the method satisfies some error bound which includes the best approximation error and have similar behaviour to both continuous and most discontinuous Galerkin methods in this regard.

| Mesh Size | PWSCG elements | $Q_1$ elements | |
|---|---|---|---|
| | $L^2$-error | $L^2$-error | Order |
| $2 \times 2 \times 2$ | 0.00000000 | 0.04004808 | |
| $4 \times 4 \times 4$ | 0.00000003 | 0.00969514 | 2.05 |
| $8 \times 8 \times 8$ | 0.00000004 | 0.00239743 | 2.02 |
| $16 \times 16 \times 16$ | 0.00000063 | 0.00059819 | 2.00 |
| $32 \times 32 \times 32$ | 0.00000056 | 0.00014957 | 2.00 |

Table 5.5: Error of PWSCG and $Q_1$ methods when approximating $u = e^{i2x_1}$ where the PWSCG plane waves vary along the two other axes.

## 5.2 Manufactured solution

Now we will consider a case where the exact solution is not in the discrete space and there is no correlation between the plane wave spaces and the solution. This is to establish a baseline expectation for the convergence of the PWSCG-method for general problems.

The method used for constructing this test case is the method of manufactured solutions. This method consists of choosing any function $u \in H^2(\Omega)$ and setting $f$ equal to the residual of the homogeneous equation, thus eliminating the residual and making $u$ an exact solution of the problem.

The function used in this test case is

$$\begin{aligned} u(x_1, x_2, x_3) &= x_3^2 \log\left(\sqrt{1 + x_1}\right) + x_2 \\ k(x_1, x_2, x_3) &= x_3 \sin\left(1 + x_1^2 + x_2^3\right) \end{aligned}$$

and inserting this into Helmholtz equation gives

$$\begin{aligned} f(x_1, x_2, x_3) = \frac{-x_3^2}{2(1 + x_1)^2} + \log(1 + x_1) + \\ x_3^2 \left( x_2 + \frac{x_3^2 \log(1 + x_1)}{2} \right) \sin\left(1 + x_1^2 + x_2^3\right) \end{aligned}$$

Again, the direction of $\boldsymbol{k}$ should not matter since the solution does not have any wave-like properties. Hence, we choose $\boldsymbol{k} = [x_1 + 1, x_3 + 1, x_2 + 1]$ which means the PWSCG-method is discontinuous.

Since this method should work for any choice of $u$, there is no particular reason for choosing exactly this function. We just want a function which is not trivial and which has no particular structure favouring any of the methods we test.

| Mesh size | PWSCG elements | | $Q_1$ elements | | Factor |
|---|---|---|---|---|---|
| | $L^2$-error | Order | $L^2$-error | Order | |
| $2 \times 2 \times 2$ | 0.01621736 | | 0.00421469 | | 3.85 |
| $4 \times 4 \times 4$ | 0.00392179 | 2.05 | 0.00105771 | 1.99 | 3.71 |
| $8 \times 8 \times 8$ | 0.00096933 | 2.02 | 0.00026467 | 2.00 | 3.66 |
| $16 \times 16 \times 16$ | 0.00024155 | 2.00 | 0.00006620 | 2.00 | 3.65 |
| $32 \times 32 \times 32$ | 0.00006034 | 2.00 | 0.00001656 | 2.00 | 3.64 |
| $64 \times 64 \times 64$ | 0.00001518 | 1.99 | 0.00000437 | 1.92 | 3.47 |

Table 5.6: Error of PWSCG and $Q_1$ methods when approximating a manufactured solution with no apparent structure favoring any of the methods.

As we see from table 5.6, both methods have second order convergence, the PWSCG-method having larger errors by a factor of between 3 and 4. It is not surprising that the PWSCG-method is worse than $Q_1$ at approximating general solutions as it was designed to approximate solutions with a particular behaviour. The fact that it has second order convergence can be considered a good thing, since this means that if we approximate solutions which are not wave-like in some areas, or if we fail to find a good approximation of $\boldsymbol{k}$ in some areas, the results will not be catastrophic. However, if we can detect such areas beforehand, reducing the elements in these areas to a $Q_1$ elements by setting $\boldsymbol{k} = 0$ may be favorable.

## 5.3 Radial wave

Let us now consider a problem where the solution is a wave-like function. In section 2.1.2, we saw that radial waves were solutions to the Helmholtz equation everywhere except in the point of origin of the waves. We also saw how the radial waves could be written as a plane wave with varying $\boldsymbol{k}$. Hence, this is a case where the PWSCG-method should be useful while not being trivial.

The particular problem we will look at has a solution

$$u = \frac{e^{ik|\boldsymbol{x}-\boldsymbol{x}_0|}}{|\boldsymbol{x}-\boldsymbol{x}_0|}$$

where $x_0 = [-1, -1, -1]$ and we will test for $k \in \{2, 4, 8, 16\}$. Since radial plane waves solve the homogeneous Helmholtz equation in every point except $x_0$, which is outside our domain, we set $f = 0$. Figure 5.1 shows the real value of the solution in a cut trough the cube.

| Mesh size | PWSCG elements | | $Q_1$ elements | | Factor |
|---|---|---|---|---|---|
| | $L^2$-error | Order | $L^2$-error | Order | |
| $2 \times 2 \times 2$ | 0.00652124 | | 0.03936990 | | 6.04 |
| $4 \times 4 \times 4$ | 0.00128754 | 2.34 | 0.01135903 | 1.79 | 8.82 |
| $8 \times 8 \times 8$ | 0.00029661 | 2.12 | 0.00308185 | 1.88 | 10.39 |
| $16 \times 16 \times 16$ | 0.00007291 | 2.02 | 0.00079951 | 1.95 | 10.97 |
| $32 \times 32 \times 32$ | 0.00001827 | 2.00 | 0.00020333 | 1.98 | 11.13 |
| $64 \times 64 \times 64$ | 0.00000462 | 1.98 | 0.00005128 | 1.99 | 11.10 |

Table 5.7: Error of PWSCG and $Q_1$ methods when approximating a radial wave originating in in $\boldsymbol{x}_0 = [-1, -1, -1]$ and $k = 4$.

| Mesh size | PWSCG elements | | $Q_1$ elements | | Factor |
|---|---|---|---|---|---|
| | $L^2$-error | Order | $L^2$-error | Order | |
| $2 \times 2 \times 2$ | 0.14860910 | | 0.37975217 | | 2.56 |
| $4 \times 4 \times 4$ | 0.01803060 | 3.04 | 0.38437972 | -0.02 | 21.32 |
| $8 \times 8 \times 8$ | 0.00660201 | 1.45 | 0.28188089 | 0.45 | 42.70 |
| $16 \times 16 \times 16$ | 0.00097020 | 2.77 | 0.08197577 | 1.78 | 84.49 |
| $32 \times 32 \times 32$ | 0.00016290 | 2.57 | 0.03122489 | 1.39 | 191.68 |
| $64 \times 64 \times 64$ | 0.00007626 | 1.10 | 0.01462363 | 1.09 | 191.76 |

Table 5.8: Error of PWSCG and $Q_1$ methods when approximating a radial wave originating in $\boldsymbol{x}_0 = [-1, -1, -1]$ and $k = 16$.

We start out by looking at the results for $k = 4$, shown in table 5.7. Both methods seems to have asymptotic convergence rate of 2, but the PWSCG-method is better that the $Q_1$ method by a factor that seems to converge towards some number close to 11, or about one order of magnitude better than $Q_1$.

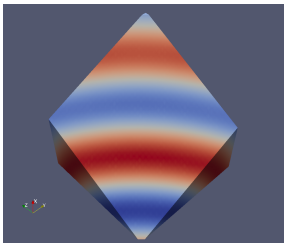


Figure 5.1: A cut trough the real part of the solution of a radial wave problem. We see how the top of the wave curves.

As a comparison, we look at the case where $k = 16$, shown in table 5.8. Here, the difference between the methods is larger, PWSCG converging towards being better by a factor of almost 200. Hence, it seems the difference between the two methods increase when the exact solution oscillates more.

One last thing worth noting about these tables is that the convergence rate in many of these test cases seems to dip somewhat

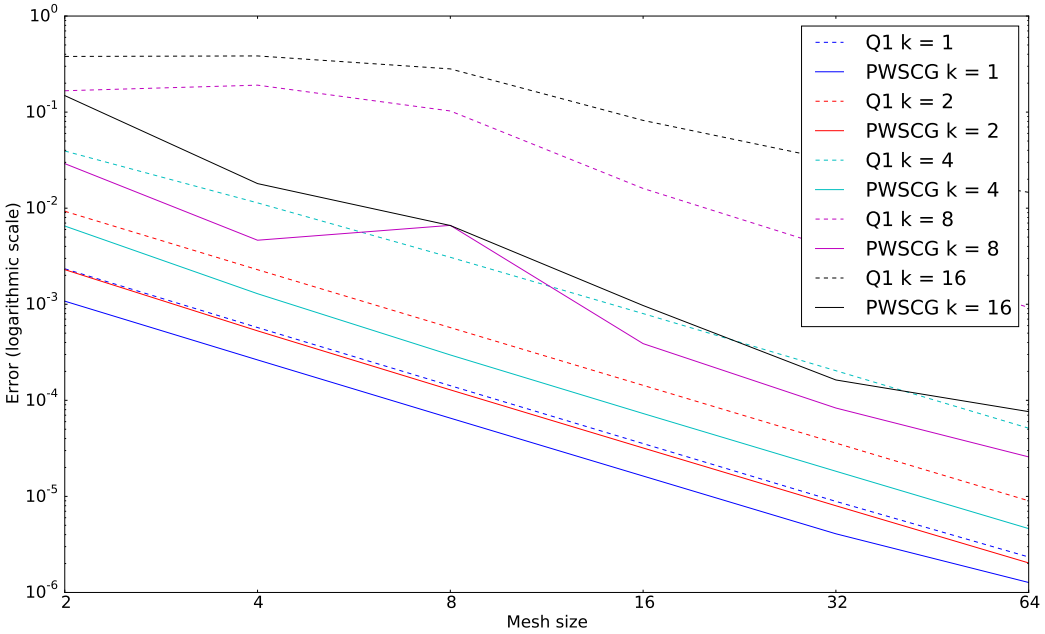



Figure 5.2: Error of PWSCG and $Q_1$ methods when approximating a radial wave originating in $\boldsymbol{x}_0 = [-1, -1, -1]$ for different values of $k$ between 1 and 16. Dotted lines represent methods using $Q_1$ elements and solid lines represent PWSCG-methods. Lines having the same color represent the two methods using the same value of $k$.

for the $64 \times 64 \times 64$ mesh. This can be explained by the fact that the approximation error at this level approaches the expected round-off error. When round-off errors start becoming a significant part of the total error, the convergence will start to flat out and the error may even increase as the round-off errors outgrow the approximation error.

Figure 5.2 shows the results of all the test cases using radial waves. Since the $y$-axis is logarithmic, straight lines represent constant convergence rate, where the inclination is proportional to the convergence rate. As we can see, all the lines are more or less parallel with some irregularities for the cases with highest values of $k$. The fact that the lines are parallel mean they have about the same convergence rate, which we know to be around 2. The irregularities for high $k$ are also not surprising since highly oscillatory functions are harder to sample by the finite element method.

The distance between lines representing the $Q_1$ method and the corresponding PWSCG-method increases for larger $k$, supporting what we found comparing $k = 4$ to $k = 16$.

| Mesh size | PWSCG elements | | $Q_1$ elements | |
|---|---|---|---|---|
| | $L^2$-error | Order | $L^2$-error | Order |
| $2 \times 2 \times 2$ | 0.04241647 | | 0.02892805 | |
| $4 \times 4 \times 4$ | 0.01489936 | 1.51 | 0.00737201 | 1.97 |
| $8 \times 8 \times 8$ | 0.00437660 | 1.77 | 0.00193444 | 1.93 |
| $16 \times 16 \times 16$ | 0.00115968 | 1.92 | 0.00049675 | 1.96 |
| $32 \times 32 \times 32$ | 0.00029649 | 1.97 | 0.00012583 | 1.98 |
| $64 \times 64 \times 64$ | 0.00007482 | 1.99 | 0.00003167 | 1.99 |

Table 5.9: Error of PWSCG and $Q_1$ methods when approximating a radial sine wave originating in $\boldsymbol{x}_0 = [-1, -1, -1]$ and $k = 4$.

We also note that the $Q_1$ method starts almost flat for high values of $k$. Figure 5.3 demonstrates the problem with using too coarse mesh of linear functions to approximate an oscillating function. From the results we see that the PWSCG-method also has irregularities when using too coarse grids for very oscillatory solutions, but it seems to handle the case better, both by having much lower initial error, and by having somewhat more consistent convergence rate on coarse grids.

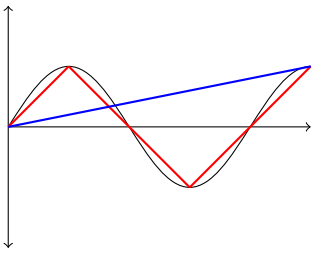


Figure 5.3: A linear (blue) and a piecewise linear (red) function interpolating an oscillating sine. The blue line has no way of capturing the behaviour of the sine, however, the red lines capture most of the oscillating nature of the sine function. This is related to the Nyquist sampling rate from signal processing [1, p. 26].

We will look at one final test case. In section 4.1.5 we looked at how real-valued plane waves are not part of $PW^+_{\boldsymbol{k}}$, and hence we assumed the PWSCG-method would not be as well suited for these waves as it is for the complex plain-waves. In table 5.9 are the results of running the same case as in 5.7, only now the exact solution is a sine wave

$$u = \frac{\sin\left(k\left|\boldsymbol{x} - \boldsymbol{x}_0\right|\right)}{\left|\boldsymbol{x} - \boldsymbol{x}_0\right|}$$

As we see from the results, the PWSCG-method is now worse than $Q_1$. The convergence rate is basically the same, however, and the difference between the methods are still somewhat smaller than it was in the manufactured solutions test case in section 5.2.

| | PWSCG elements | $Q_1$ elements |
|---|---|---|
| | 36.37s | 3.41s |
| | 36.80s | 3.60s |
| | 38.19s | 3.57s |
| | 38.21s | 3.52s |
| | 37.84s | 3.52s |
| Average | 37.48s | 3.52s |

Table 5.10: Execution times of the PWSCG-method and the $Q_1$ method for a radial plane wave on the same $8 \times 8 \times 8$ grid.

| | PWSCG elements | $Q_1$ elements |
|---|---|---|
| | 8.36s | 19.31s |
| | 8.00s | 19.46s |
| | 8.33s | 21.92s |
| | 8.44s | 21.78s |
| | 8.42s | 20.45s |
| Average | 8.31s | 20.58s |

Table 5.11: Run times of the PWSCG-method and the $Q_1$ method for a radial plane wave on the coarsest grid which produces errors less than $10^{-3}$ for each method.

## 5.4 Execution time

Since the main focus of this project was to determine the characteristics of the PWSCG-method, the implementation has not been optimized for execution time. In the present sate, the implementation of the PWSCG-method is a lot slower than than the $Q_1$ method on the same grid, as we see in table 5.10.

However, a more interesting comparison is how long it takes to execute the two methods on grids such that the error will be equivalent. To test this, we will look at the coarsest grid which produces an error less than $10^{-3}$ for both methods. The problem we use for this test case is the radial wave from section 5.3 with $k = 4$ which has a moderately oscillating solution.

The coarsest mesh for which the PWSCG-method gives an error less than $10^{-3}$ is $4 \times 5 \times 5$, while the coarsest mesh for which the $Q_1$ method provides a similar error is $14 \times 14 \times 15$. The results of this comparison is presented in table 5.11. Here the PWSCG-method is clearly more efficient.

As noted, the focus of the implementation was not the speed, but rather accuracy of the convergence characteristics produced by the program. There are many things that can be done to drastically reduce the execution time, including making an assembly routine which can correctly scale the integration results from a reference element, implementing a special numerical integration scheme that handles the integrals encountered in a more efficient manner, or just eliminating redundant calculations in general. The fact that the PWSCG-method outperforms $Q_1$ in some tests is very promising when considering how much faster we can expect an optimized version would be.

# Chapter 6

# Conclusion

In this thesis, we introduced the Plane Wave Semi-Continuous Galerkin method. Using a detailed description of the degrees of freedom on elements and on the intersections between elements, we saw how the method could be considered semi-continuous as it satisfies one of the two conditions that a method needs to fit into the continuous Galerkin framework.

We also showed how the remaining discontinuities in the method could be reasoned about using the theory of discontinuous Galerkin methods, and how not considering the surface integrals over the interior edges corresponded to a particular choice of flux functions.

In chapter 5, we saw how the PWSCG-method had second order convergence for all the cases, excluding the exact ones. This is similar to the $Q_1$ method, which has the same number of dofs. For solutions without any plane wave-like behaviour the PWSCG-method produced, not surprisingly, higher errors than the corresponding $Q_1$ method on the same mesh. However, when approximating solutions which did have plane wave-like behaviour locally, the PWSCG-method produced significantly smaller errors than the $Q_1$ method, the difference increasing for larger values of $k$.

From this we conclude that the PWSCG-method shows some promise as a method for approximating solutions with plane wave-like behaviour locally, even on relatively coarse grids. The methods used to derive the function spaces and the properties of these, including how the semi-continuous method related to the continuous and discontinuous Galerkin methods, may also be used to derive other semi-continuous methods with different properties.

## 6.1 Future work

There are multiple aspects of the PWSCG-method that would be interesting to investigate further. In light of the numerical results of chapter 5, we may expect there to exist error bounds for the PWSCG-method similar to those for continuous and discontinuous Galerkin methods. Deriving these error bounds analytically would not only give proof that the method will behave in all cases, but also give more insight into which factors are important for the performance of the method.

Another interesting question is how to best compute approximations of $\boldsymbol{k}$. There are many possible ways of doing this, non of which immediately stand out as the best. One method would be to use $P_0$ elements to solve the equation initially, and then extracting wave-like behaviour from that solution and solve again using PWSCG-elements, this time expecting much more accurate results.

When deriving $PW_{\boldsymbol{k}}^+(T)$ we noted that we could derive a similar space $PW_{\boldsymbol{k}}^-(T)$. The combined space $\{u+v \,:\, u \in PW_{\boldsymbol{k}}^+(T),\, v \in PW_{\boldsymbol{k}}^-(T)\}$ would also contain the real-valued waves $\cos(\boldsymbol{k}\cdot\boldsymbol{x})$ and $\cos(\boldsymbol{k}\cdot\boldsymbol{x})$ which would be useful. The problem is that the combination of the finite spaces $\widehat{PW}_{\boldsymbol{k}}^+(T)$ and $\widehat{PW}_{\boldsymbol{k}}^-(T)$ will have dimension 7, since both includes the constant function. The linear functions which appear with $\boldsymbol{k}$ has some zero-components, will also overlap. This means it is not obvious what degrees for freedom should be used to ensure semi-continuity. It could be interesting to explore this possibility further.

The last section of chapter 5 presented some result including execution times of the program using the different elements. It was noted that these results could be expected to improve a lot by optimizing the implementation. Doing this and then run further tests on execution time would give more insight into what practical cases this method could be useful for. Also, making an implementation of the PWDG-method from [11] and comparing these two methods could be an interesting case. The PWDG-method is a truly discontinuous method with both the advantages and disadvantages that brings, making for an interesting comparison.

# Bibliography


[1] A. Ambardar. *Intl Stdt Ed - Digital Signal Processing*. Thomson Learning EMEA, Limited, 2006. ISBN: 9780495082385. URL: https://books.google.no/books?id=JiqNAAAACAAJ.

[2] Douglas N. Arnold et al. "Unified analysis of discontinuous Galerkin methods for elliptic problems". In: *SIAM J. Numer. Anal.* 39.5 (2001/02), pp. 1749–1779. ISSN: 0036-1429. DOI: 10.1137/S0036142901384162. URL: http://dx.doi.org/10.1137/S0036142901384162.

[3] *Automated Solution of Differential Equations by the Finite Element Method — FEniCS Project*. May 12, 2015. URL: http://fenicsproject.org/.

[4] Dietrich Braess. *Finite elements*. Third. Theory, fast solvers, and applications in elasticity theory, Translated from the German by Larry L. Schumaker. Cambridge University Press, Cambridge, 2007, pp. xviii+365. ISBN: 978-0-521-70518-9. DOI: 10.1017/CBO9780511618635. URL: http://dx.doi.org/10.1017/CBO9780511618635.

[5] P. G. Ciarlet and J.-L. Lions, eds. *Handbook of numerical analysis. Vol. II*. Handbook of Numerical Analysis, II. Finite element methods. Part 1. North-Holland, Amsterdam, 1991, pp. x+928. ISBN: 0-444-70365-9.

[6] Bernardo Cockburn and Chi-Wang Shu. "The local discontinuous Galerkin method for time-dependent convection-diffusion systems". In: *SIAM J. Numer. Anal.* 35.6 (1998), 2440–2463 (electronic). ISSN: 0036-1429. DOI: 10.1137/S0036142997316712. URL: http://dx.doi.org/10.1137/S0036142997316712.

[7] Germund Dahlquist and Åke Björck. *Numerical methods in scientific computing. Vol. I*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008, pp. xxviii+717. ISBN: 978-0-898716-44-3. DOI: 10.1137/1.9780898717785. URL: http://dx.doi.org/10.1137/1.9780898717785.

[8] Françoise Demengel and Gilbert Demengel. *Functional spaces for the theory of elliptic partial differential equations*. Universitext. Translated from the 2007 French original by Reinie Erné. Springer, London; EDP Sciences, Les Ulis, 2012, pp. xviii+465. ISBN: 978-1-4471-2806-9; 978-2-7598-0698-0. DOI: 10.1007/978-1-4471-2807-6. URL: http://dx.doi.org/10.1007/978-1-4471-2807-6.

[9] Lawrence C. Evans. *Partial differential equations*. Second. Vol. 19. Graduate Studies in Mathematics. American Mathematical Society, Providence, RI, 2010, pp. xxii+749. ISBN: 978-0-8218-4974-3. DOI: 10.1090/gsm/019. URL: http://dx.doi.org/10.1090/gsm/019.

[10] *GetFEM++ Homepage — GetFEM++*. May 12, 2015. URL: http://download.gna.org/getfem/html/homepage/.

[11] Claude J. Gittelson, Ralf Hiptmair, and Ilaria Perugia. "Plane wave discontinuous Galerkin methods: analysis of the $h$-version". In: *M2AN Math. Model. Numer. Anal.* 43.2 (2009), pp. 297–331. ISSN: 0764-583X. DOI: 10.1051/m2an/2009002. URL: http://dx.doi.org/10.1051/m2an/2009002.

[12] Ralf Hiptmair and Ilaria Perugia. "Mixed plane wave discontinuous Galerkin methods". In: *Domain decomposition methods in science and engineering XVIII*. Vol. 70. Lect. Notes Comput. Sci. Eng. Springer, Berlin, 2009, pp. 51–62. DOI: 10.1007/978-3-642-02677-5_5. URL: http://dx.doi.org/10.1007/978-3-642-02677-5_5.

[13] Robert C. Kirby and Anders Logg. "A compiler for variational forms". In: *ACM Trans. Math. Software* 32.3 (2006), pp. 417–444. ISSN: 0098-3500. DOI: 10.1145/1163641.1163644. URL: http://dx.doi.org/10.1145/1163641.1163644.

[14] P. C. Matthews. *Vector calculus*. Springer Undergraduate Mathematics Series. Springer-Verlag London, Ltd., London, 1998, pp. x+182. ISBN: 3-540-76180-2. DOI: 10.1007/978-1-4471-0597-8. URL: http://dx.doi.org/10.1007/978-1-4471-0597-8.

[15] Bryan P. Rynne and Martin A. Youngson. *Linear functional analysis*. Second. Springer Undergraduate Mathematics Series. Springer-Verlag London, Ltd., London, 2008, pp. x+324. ISBN: 978-1-84800-004-9. DOI: 10.1007/978-1-84800-005-6. URL: http://dx.doi.org/10.1007/978-1-84800-005-6.