TESIS DE LA UNIVERSIDAD DE ZARAGOZA

2023

141

Álvaro Pé de la Riva

Robust multigrid methods for Isogeometric discretizations applied to poroelasticity problems

Director/es

Gaspar Lorenz, Francisco José

Rodrigo Cardiel, Carmen

ISSN 2254-7606

Prensas de la Universidad Universidad Zaragoza

ISSN 2254-7606

Universidad Zaragoza
1542

Tesis Doctoral

ROBUST MULTIGRID METHODS FOR ISOGEOMETRIC DISCRETIZATIONS APPLIED TO POROELASTICITY PROBLEMS

Autor

Álvaro Pé de la Riva

Director/es

Gaspar Lorenz, Francisco José
Rodrigo Cardiel, Carmen

UNIVERSIDAD DE ZARAGOZA
Escuela de Doctorado

Programa de Doctorado en Matemáticas y Estadística

2023

UNIVERSITY OF ZARAGOZA

DOCTORAL THESIS

Robust multigrid methods for Isogeometric discretizations applied to poroelasticity problems

Author:
Álvaro PÉ DE LA RIVA

Supervisor:
Francisco J. GASPAR LORENZ,
Carmen RODRIGO CARDIEL

A thesis submitted in fulfillment of the requirements for the degree of Doctorate in Mathematics

in the

Department of Applied Mathematics, IUMA
University of Zaragoza

January 16, 2023

Álvaro Pé de la Riva

Abstract

Isogeometric analysis (IGA) eliminates the gap between finite element analysis (FEA) and computer aided design (CAD). Due to this, IGA is an innovative approach that is receiving an increasing attention in the literature and it has recently become a trending topic. Many research efforts are being devoted to the design of efficient and robust solvers for this type of discretization. Given the optimality of multigrid methods for FEA, the application of these methods to IGA discretizations has not been unnoticed. We firmly think that they are a very promising approach as efficient and robust solvers for IGA and therefore in this thesis we are concerned about their application. In order to give a theoretical support to the design of multigrid solvers, local Fourier analysis (LFA) is proposed as the main quantitative analysis. Although different scalar problems are also considered along this thesis, we make a special focus on poroelasticity problems. More concretely, we focus on the quasi-static Biot's equations for the soil consolidation process. Nowadays, it is a very challenging task to achieve robust multigrid solvers for poroelasticity problems with respect physical parameters and/or the mesh size. Thus, the main contribution of this thesis is to propose robust multigrid methods for isogeometric discretizations applied to poroelasticity problems.

The first part of this thesis is devoted to the introduction of the parametric construction of curves and surfaces since these techniques are the basis of IGA. Hence, with the definition of Bernstein polynomials and Bézier curves as a starting point, we introduce B-splines and non-uniform rational B-splines (NURBS) since these will be the basis functions considered for our numerical experiments. The second part deals with the isogeometric analysis. In this part, the isoparametric approach is explained to the reader and the isogeometric analysis of some scalar problems is presented. Hence, the strong and weak formulations by means of Galerkin's method are introduced and the isogeometric approximation spaces as well. The next point of this thesis consists of multigrid methods. The basics of multigrid methods are explained and, besides the presentation of some classical iterative methods as smoothers, block-wise smoothers such as multiplicative and additive Schwarz methods are also introduced. At this point, we introduce LFA for the design of efficient and robust multigrid methods. Furthermore, both standard and infinite subgrids local Fourier analysis are explained in detail together with the analysis for block-wise smoothers and the analysis for systems of partial differential equations.

After the introduction of isogeometric discretizations, multigrid methods as our choice of solvers and LFA as theoretical analysis, our goal is to design efficient and robust multigrid methods with respect to the spline degree for IGA discretizations of some scalar problems. Hence, we show that the use of multigrid methods based on multiplicative or additive Schwarz methods provide a good performance and robust asymptotic convergence rates. The last part of this thesis is devoted to the isogeometric analysis of poroelasticity. For this task, Biot's model and its isogeometric discretization are introduced. Moreover, we present an innovative mass stabilization of the two-field formulation of Biot's equations that eliminates all the spurious oscillations in the numerical approximation of the pressure. Then, we deal with two types

of solvers for these poroelastic equations: Decoupled and monolithic solvers. In the first group we devote special attention to the fixed-stress split method and a mass stabilized iterative scheme proposed by us that can be automatically applied from the mass stabilization formulation mentioned before. In addition, we perform a von Neumann analysis for this iterative decoupled solver applied to Terzaghi's problem and demonstrate that it is stable and convergent for pairs $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (with global smoothness $\mathcal{C}^1$). Regarding the group of monolithic solvers, we propose multigrid methods based on coupled and decoupled smoothers. Coupled additive Schwarz methods are proposed as coupled smoothers for isogeometric Taylor-Hood elements. More concretely, we propose a $51-$point additive Schwarz method for the pair $Q2-Q1$. In the last part, we also propose to use an inexact version of the fixed-stress split algorithm as decoupled smoother by applying iterations of different additive Schwarz methods for each variable. For the latter approach, we consider the pairs of elements $Q2-Q1$ and $Q3-Q2$ (with global smoothness $\mathcal{C}^1$). Finally, thanks to LFA we manage to design efficient and robust multigrid solvers for the Biot's equations and some numerical results are shown.

Resumen

El análisis isogeométrico (IGA) elimina la barrera existente entre elementos finitos (FEA) y el diseño geométrico asistido por ordenador (CAD). Debido a esto, IGA es un método novedoso que está recibiendo una creciente atención en la literatura y recientemente se ha convertido en tendencia. Muchos esfuerzos están siendo puestos en el diseño de solvers eficientes y robustos para este tipo de discretizaciones. Dada la optimalidad de los métodos multimalla para elementos finitos, la aplicación de estos métodos a discretizaciones isogeométricas no ha pasado desapercibida. Nosotros pensamos firmemente que los métodos multimalla son unos candidatos muy prometedores a ser solvers eficientes y robustos para IGA y por lo tanto en esta tesis apostamos por su aplicación. Para contar con un análisis teórico para el diseño de nuestros métodos multimalla, el análisis local de Fourier es propuesto como principal análisis cuantitativo. En esta tesis, a parte de considerar varios problemas escalares, prestamos especial atención al problema de poroelasticidad, concretamente al modelo cuasiestático de Biot para el proceso de consolidación del suelo. Actualmente, el diseño de métodos multimalla robustos para problemas poroelásticos respecto a parámetros físicos o el tamaño de la malla es un gran reto. Por ello, la principal contribución de esta tesis es la propuesta de métodos multimalla robustos para discretizaciones isogeométricas aplicadas al problema de poroelasticidad.

La primera parte de esta tesis se centra en la construcción paramétrica de curvas y superficies dado que estas técnicas son la base de IGA. Así, la definición de los polinomios de Bernstein y curvas de Bézier se presenta como punto de partida. Después, introducimos los llamados B-splines y B-splines racionales no uniformes (NURBS) puesto que éstas serán las funciones base consideradas en nuestro estudio. La segunda parte trata sobre el análisis isogeométrico propiamente dicho. En esta parte, el método isoparamétrico es explicado al lector y se presenta el análisis isogeométrico de algunos problemas. Además, introducimos la formulación fuerte y débil de los problemas anteriores mediante el método de Galerkin y los espacios de aproximación isogeométricos. El siguiente punto de esta tesis se centra en los métodos multimalla. Se tratan las bases de los métodos multimalla y, además de introducir algunos métodos iterativos clásicos como suavizadores, también se introducen suavizadores por bloques como los métodos de Schwarz multiplicativos y aditivos. Llegados a esta parte, nos centramos en el LFA para el diseño de métodos multimalla robustos y eficientes. Además, se explican en detalle el análisis estándar y el análisis basado en ventanas junto al análisis de suavizadores por bloques y el análisis para sistemas de ecuaciones en derivadas parciales.

Tras introducir las discretizaciones isogeométricas, los métodos multimalla y el LFA como análisis teórico, nuestro propósito es diseñar métodos multimalla eficientes y robustos respecto al grado polinomial de los splines para discretizaciones isogeométricas de algunos problemas escalares. Así, mostramos que el uso de métodos multimalla basados en suavizadores de tipo Schwarz multiplicativo o aditivo produce buenos resultados y factores de convergencia asintóticos robustos. La última parte de esta tesis está dedicada al análisis isogeométrico del problema de poroelasticidad. Para esta tarea, se introducen el modelo de Biot y su discretización

isogeométrica. Además, presentamos una novedosa estabilización de masa para la formulación de dos campos de las ecuaciones de Biot que elimina todas las oscilaciones no físicas en la aproximación numérica de la presión. Después, nos centramos en dos tipos de solvers para estas ecuaciones poroelásticas: Solvers desacoplados y solvers monolíticos. En el primer grupo, le dedicamos una especial atención al método *fixed-stress* y a un método iterativo propuesto por nosotros que puede ser aplicado de forma automática a partir de la estabilización de masa ya mencionada. Por otro lado, realizamos un análisis de von Neumann para este método iterativo aplicado al problema de Terzaghi y demostramos su estabilidad y convergencia para los pares de elementos $Q1 - Q1$, $Q2 - Q1$ y $Q3 - Q2$ (con suavidad global $\mathcal{C}^1$). Respecto al grupo de solvers monolíticos, nosotros proponemos métodos multimalla basados en suavizadores acoplados y desacoplados. En esta parte, métodos de Schwarz aditivos acoplados son propuestos como suavizadores acoplados para elementos isogeométricos de Taylor-Hood. En concreto, proponemos un método de Schwarz aditivo de 51 puntos para el par $Q2 - Q1$. En la parte final, también proponemos usar una versión inexacta del algoritmo de fixed-stress como suavizador desacoplado mediante la aplicación de iteraciones de diferentes métodos de Schwarz aditivos para cada variable. Para este último método, consideramos los pares de elementos $Q2 - Q1$ y $Q3 - Q2$ (con suavidad global $\mathcal{C}^1$). Finalmente, gracias al LFA conseguimos diseñar métodos multimalla eficientes y robustos para las ecuaciones de Biot y algunos resultados numéricos son mostrados.

Agradecimientos

Me gustaría empezar mostrando mi agradecimiento a mis directores el doctor don Francisco Gaspar y la doctora doña Carmen Rodrigo. Ellos han sido mi guía durante este camino, me han transmitido su pasión por la investigación, me han enseñado a perseverar ante las dificultades y, en definitiva, a trabajar con el rigor que caracteriza a un matemático. Quiero también darles las gracias por compartir sus conocimientos conmigo y por impulsarme a llegar hasta aquí. La gran dedicación y el apoyo que me han brindado son inmensurables. Para mí, ha sido un honor investigar con ellos y espero tener la oportunidad de seguir desarrollando mi investigación como fruto de sus enseñanzas.

Al doctor don Kees Oosterlee por su invitación y darme la oportunidad de realizar una estancia en el Centrum Wiskunde & Informatica, en Ámsterdam.

Al doctor don Ludmil Zikatanov y al doctor don Xiaozhe Hu por su interés y sus buenos consejos.

A nivel institucional, quiero dar las gracias a la Diputación General de Aragón por su apoyo económico que me ha permitido dedicarme a tiempo completo a la elaboración de esta tesis.

Al doctor don Héctor Orera, a quien considero un buen amigo, por toda su ayuda a lo largo de la carrera, del máster universitario y del doctorado.

A mis compañeros del departamento de matemática aplicada y del IUMA por su disposición y por su apoyo.

A mis amigos, gracias por haber hecho más amena esta etapa de mi vida.

A Irem, por todo su afecto y por confiar en mí. A pesar de la distancia siempre has conseguido animarme en los momentos más difíciles.

A mis hermanos Néstor y Víctor, a quienes admiro y respeto profundamente. A ellos quiero darles especial mención y agradecerles su apoyo y paciencia conmigo. Gracias por estar siempre que os he necesitado.

Tanto a los Pé como a los de la Riva, les doy las gracias por sus ánimos y por darle un significado auténtico a la palabra familia. Aunque no puedan ver esta tesis terminada, quiero nombrar con especial cariño a mis abuelos Ángel, Petra, Cremencio y Esperanza.

El mayor de mis agradecimientos va dirigido a mis padres, Narciso y Rosa. Los dos han velado por mí durante toda mi vida, siempre se han volcado en mis hermanos y en mí al educarnos y nunca permitir que nos faltase de nada. Por haberos entregado a nosotros, por todo el tiempo depositado y por todo el amor que nos habéis dado, quiero daros las gracias de corazón y dedicaros estas palabras que sin duda alguna merecéis. En mí han inculcado el sentido, la fuerza y el valor para demostrar de lo que soy capaz. Mi mayor deseo es que esta tesis recompense en parte todos vuestros esfuerzos y que os haga sentir orgullosos. Nada me haría más feliz. Os quiero mucho.

Contents

List of Figures

List of Tables

List of Abbreviations

CAD	**C**omputer **A**ided **D**esign
FD	**F**inite **D**ifferences
FEA	**F**inite **E**lement **A**nalysis
FEM	**F**inite **E**lement **M**ethod
FV	**F**inite **V**olumes
IGA	**I**so**G**eometric **A**nalysis
LFA	**L**ocal **F**ourier **A**nalysis

List of Symbols

k	polynomial/spline degree
ξ	parametric coordinate
η	parametric coordinate
κ	permeability
τ	time step
$\boldsymbol{u}$	displacement vector
p	pressure
η_f	viscosity of the fluid
$\boldsymbol{\varepsilon}$	strain tensor
$\boldsymbol{\sigma}$	stress tensor
w	weight/s
Ξ	knot vector in $x-$direction
$\mathcal{H}$	knot vector in $y-$direction
$\mathbb{P}^k$	set of polynomials with degree less or equal than k
E	Young's modulus
ν	Poisson ratio
h	mesh size

Chapter 1

Parametric construction of curves and surfaces

The geometry of curves and surfaces can be represented mathematically by means of explicit, implicit or parametric forms. Among them, non-parametric forms are less versatile and they are not as suitable as parametric methods for fitting, mathematical description of free-form shapes, geometric design, shape manipulation, etc. These properties of parametric forms are the main cause of their popularity in the literature and their standard usage in Computer Aided Design (CAD), see Patrikalakis and Maekawa, 2009; Piegl and Tiller, 1996; Yamaguchi, 1988; Risler, 1992.

The interest in curves and surfaces manufacturing has a huge historic background. As pointed out in Farin, 2002, shipbuilding techniques have been connected to this topic since the early Roman times. Moreover, the first steps for geometric design development were taken in the area of car industry by de Casteljau and Bézier. They introduced the use of Bernstein polynomials to represent curves and during the 20th century the B-splines appeared and were recognized as a powerful tool in approximation theory and parametric construction of curves and surfaces. All these advances can be understood as a preface to CAD that would be developed by making use of B-splines and NURBS. To sum up, CAD development has many historical influences from different areas, the interested reader may refer to Farin, 2002 for further information in this topic.

Nowadays, CAD is the standard tool for the computational construction and design of curves, surfaces and volumes. Furthermore, its importance has increased worlwide and many softwares using B-splines and NURBS machinery have been developed. The mathematical research and development of numerical techniques such as adaptive refinements are also receiving a lot of attention. For this purpose, new approaches such as hierarchical B-splines and T-splines have been proposed as an alternative to tensor-product surfaces and solids.

Let this introduction serve as a motivation for this chapter. As a starting point, Bernstein polynomials and their properties will be introduced. Secondly, we will focus on Bézier curves and surfaces that make use of these polynomials. Then, the next two sections will be devoted to the description of B-splines basis functions and their properties. With these basis functions, B-spline curves and surfaces can be defined. In a natural way, NURBS basis functions together with their properties are introduced and then the definition of NURBS curves and surfaces is included.

1.1 Bernstein polynomials

Parametrizations based on polynomials exhibit a good behavior for geometric modelling. Among the benefits of using polynomials, we find an accurate and efficient

evaluation of the parametrized curves/surfaces/solids. In addition, polynomials are mathematically simple functions that allow to capture a wide set of geometries, see Piegl and Tiller, 1996. For this purpose, Bernstein polynomials have been very useful in the computer modelling framework. Nevertheless, their use is not restricted to CAD. Initially, they were introduced by S. Bernstein for the proof of Weierstrass theorem and nowadays these polynomials are widely applied in approximation theory, numerical analysis, statistics, etc. For a deeper historical review of Bernstein polynomials, see Farouki, 2012. Now, let us introduce the definition of Bernstein polynomials of degree k:

Definition 1.1.1. Given a polynomial degree k, the Bernstein polynomials are defined as

$$B_i^k(t) := \binom{k}{i} t^i (1-t)^{k-i}, \quad t \in [0,1], \;\; i = 0, \ldots, k. \tag{1.1}$$

As as example, in Figure 1.1 the quartic Bernstein polynomials are shown.

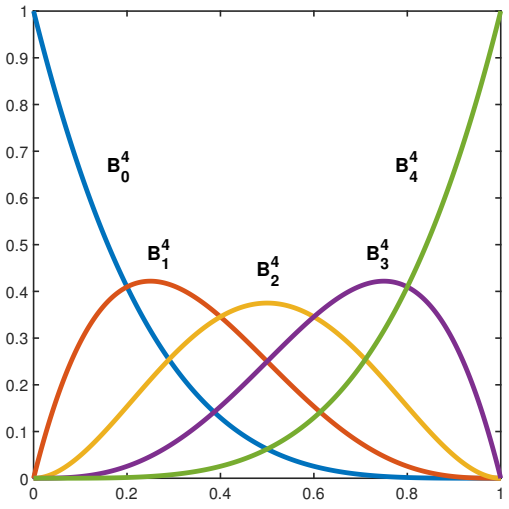


FIGURE 1.1: Quartic Bernstein polynomials.

Bernstein polynomials have many interesting properties (for instance, see Manni and Speleers, 2016; Piegl and Tiller, 1996). Some of them are the following:

1. Non-negativity: Bernstein polynomials are non-negative on the interval [0,1], that is,
$$B_i^k(t) \geq 0, \quad \forall t \in [0,1]. \tag{1.2}$$
2. Partition of unity:
$$\sum_{i=0}^{k} B_i^k(t) = 1, \quad \forall t \in [0,1]. \tag{1.3}$$
3. End-point interpolation:
$$B_0^k(0) = B_k^k(1) = 1. \tag{1.4}$$
4. Each polynomial B_i^k has its unique maximum at $t = i/k$. These values are called Greville abscissae.

5. Symmetry:
$$B_i^k(t) = B_{k-i}^k(1-t), \quad \forall t \in [0,1]. \tag{1.5}$$

6. Recursive definition:
$$B_i^k(t) = (1-t)B_i^{k-1}(t) + tB_{i-1}^{k-1}(t), \tag{1.6}$$
with $B_i^k(t) \equiv 0$ if $i < 0$ or $i > k$.

7. Derivatives recursion:
$$\frac{dB_i^k(t)}{dt} = k(B_{i-1}^{k-1}(t) - B_i^{k-1}(t)). \tag{1.7}$$

8. Integration: All the Bernstein polynomials of degree k satisfy that
$$\int_0^1 B_i^k(t)dt = \frac{1}{k+1}, \quad \forall i = 0, \ldots, k. \tag{1.8}$$

Although all these properties are pretty interesting, the most remarkable ones in the computer modeling context are the non-negativity and partition of unity properties. Moreover, their recursive definition provides an efficient way to evaluate Bernstein polynomials and in practice numerical algorithms are based on this aspect. In approximation theory, it can be proved that these polynomials $\{B_i^k\}_{i=0}^k$ form a basis for the polynomial space $\mathbb{P}_k := \text{span} < 1, t, \ldots, t^k >$ in the interval $(0,1)$, see Manni and Speleers, 2016. In addition, Bernstein used these polynomials for the constructive proof of the Weierstrass approximation theorem, see Davis, 1963. In order to achieve a uniform approximation by means of polynomials to any continuous function f, the Bernstein polynomial of a function f was introduced as
$$B^k(t; f) := \sum_{i=0}^k f(i/k)B_i^k(t). \tag{1.9}$$

Hence, Bernstein polynomials have an important role in approximation theory and their relevance is not restricted to geometric modeling. In addition, Bernstein polynomials are popular in statistics as estimators of density functions, see Babu, Canty, and Chaubey, 2002. Also, some generalizations of Bernstein polynomials have been proposed in the literature such as the q-Bernstein polynomials (Oruç and Tuncer, 2001).

1.2 Bézier curves and surfaces

Standard polynomial parametrizations are set by means of a power basis form. By using the canonical basis of the polynomial space $\mathbb{P}_k$, the $k-$th power basis form of a curve $C(t) = (x(t), y(t), z(t))$ is given by
$$\mathbf{C}(t) = \sum_{i=0}^k \mathbf{P}_i t^i, \tag{1.10}$$
where $\mathbf{P}_i = (x_i, y_i, z_i)$ are vector coefficients. However, power basis forms are not suitable for geometric modelling since they have only an algebraic essence. In addition, the vector coefficients lack a relationship with the geometry. At this point,

Bézier proposed to use Bernstein polynomials rather than the canonical basis for curves parametrization. Thus, a $k-$th degree Bézier curve is given by

$$\mathbf{C}_k(t) = \sum_{i=0}^{k} \mathbf{P}_i B_i^k(t). \tag{1.11}$$

With Bézier's method, the vector coefficients obtain an important geometric meaning for the parametrization by making a good use of Bernstein polynomials properties. For instance, the non-negativity and partition of unity properties imply the convex hull property of Bézier curves, that is, the curve is totally contained in the convex hull of the vector coefficients. Furthermore, Bernstein polynomials basis end-point interpolation property makes the curves to be interpolatory at the end-points vector coefficients. Thus, the vector coefficients for Bézier forms are called control points and the polygon formed by the segments connecting the control points is known as control polygon. As an example, we show a quartic Bézier curve in Figure 1.2 with its corresponding control points.

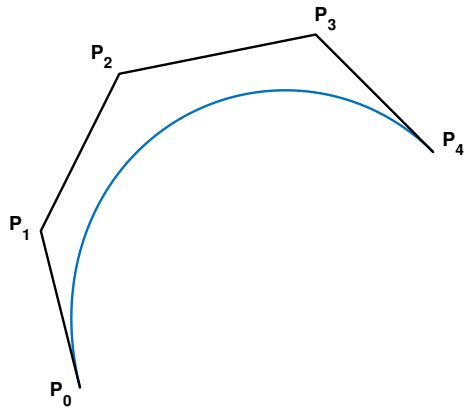


FIGURE 1.2: Bézier curve
Example of a fourth degree Bézier curve with control points $\mathbf{P}_0 = (0.5, 0)$, $\mathbf{P}_1 = (0, 2)$, $\mathbf{P}_2 = (1, 4)$, $\mathbf{P}_3 = (3.5, 4.5)$ and $\mathbf{P}_4 = (5, 3)$.

As observed in Figure 1.2, the Bézier curve shape tends to imitate the control polygon's shape. In fact, Bézier curves inherit the variation diminishing property from the non-negativity property of Bernstein polynomials. In other words, the number of intersections between a Bézier curve and a given straight line is bounded by the intersections between that line and the corresponding control polygon of the curve.

For the efficient point evaluation of Bézier curves, the first step is to realize that any given k-th degree Bézier curve C_k can be expressed as a convex combination of two $(k-1)$-th degree Bézier curves (see Piegl and Tiller, 1996) as follows

$$C_k[\mathbf{P}_0, \ldots, \mathbf{P}_k](t) = (1-t)C_{k-1}[\mathbf{P}_0, \ldots, \mathbf{P}_{k-1}] + tC_{k-1}[\mathbf{P}_1, \ldots, \mathbf{P}_k], \tag{1.12}$$

where $C_k[\mathbf{P}_0, \ldots, \mathbf{P}_k]$ denotes a k-th degree Bézier curve with $k+1$ control points $\mathbf{P}_0, \ldots, \mathbf{P}_k$. As pointed out in Piegl and Tiller, 1996, this property follows from the

recursive definition of Bernstein polynomials given in (1.6). In fact, this recursion can be transferred to the control points level such that the evaluation of C_k at any point t is obtained. Thus, convex combination of the starting control points $\{\mathbf{P}_i\}_{i=0}^{k} \equiv \{\mathbf{P}_i^0\}_{i=0}^{k}$ leads us recursively to new control polygons formed by a reduced number of control points $\{\mathbf{P}_i^j\}_{i=0}^{k-j}$. Hence, the process finishes when $C_k(t) = \mathbf{P}_0^k(t)$ is computed. This was the idea of de Casteljau, whose algorithm (see Böhm, Farin, and Kahmann, 1984; Gardan and Casteljau, 1985; Piegl and Tiller, 1996) is based on the formula

$$\mathbf{P}_i^j(t) = (1-t)\mathbf{P}_i^{j-1}(t) + t\mathbf{P}_{i+1}^{j-1}(t), \tag{1.13}$$

where $j = 1, \ldots, k$ and $i = 0, \ldots, k-j$. De Casteljau's algorithm allows us to evaluate Bézier curves and it is actually a corner cutting process since intermediate control polygons are constructed until a point of the curve is reached. Thus, in Figure 1.3 we show the control points and control polygons of the quartic Bézier curve in Figure 1.2 obtained recursively by using (1.13) in order to compute $C_4(0.6) = (2.216, 3.6336)$.

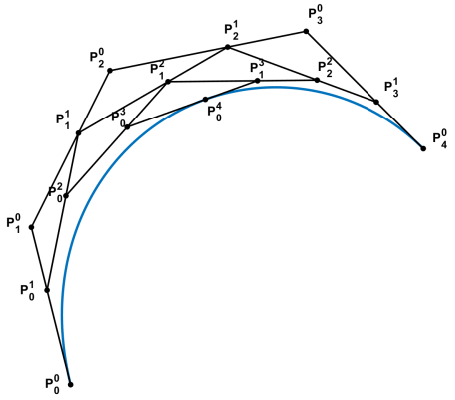


FIGURE 1.3: Corner cutting process
Control points and control polygons of the quartic Bézier curve in Figure 1.2 obtained recursively by using (1.13) in order to compute $C_4(0.6) = (2.216, 3.6336)$.

However, polynomials do not provide a parametrization to capture the exact geometry of circles, cylinders, etc. On the other hand, it is well-known that the use of rational polinomial functions is enough to parametrize conic sections. Hence, Bézier curves based on rational polinomials are capable to parametrize a wider set of geometries and they are a more powerful tool in geometric modelling. Let us introduce the definition of a k-th degree rational Bézier curve as follows

$$\mathbf{C}_k(t) = \frac{\sum_{i=0}^{k} w_i B_i^k(t)\mathbf{P}_i}{\sum_{i=0}^{k} w_i B_i^k(t)}, \tag{1.14}$$

where $\{w_i\}_{i=0}^{k}$ is a set of weights. Usually, these rational basis functions are denoted by

$$R_i^k(t) = \frac{w_i B_i^k(t)}{\sum_{j=0}^{k} w_j B_j^k(t)}, \quad i = 0, \ldots, k. \tag{1.15}$$

Weights have to be chosen carefully since they have a geometrical meaning. First, they are usually taken positive in order to hold the convex hull property of Bézier curves. Secondly, each weight w_i represents the importance of its associated control point/basis function on the curve shape. Thus, weights serve to quantify how much the curve is pulled towards the corresponding control points. The case of $w_i = 1$, $\forall i = 0, \ldots, k$ corresponds to a Bézier curve based on polynomials. Hence, rational Bézier curves are a generalization of Bézier curves. In order to visualize the weights effect on the curve, in Figure 1.4 we compare the quartic Bézier curve shown in Figure 1.2 with its rational Bézier counterpart by changing $w_2 = 10$.

FIGURE 1.4: On the right, the previous quartic Bézier curve shown in Figure 1.2 with $w_i = 1$, $i = 0 \ldots, 4$. On the left, we show the resulting quartic rational Bézier curve by only changing $w_2 = 10$.

As long as the weights $w_i > 0$, for $i = 0, \ldots, k$, rational Bézier curves hold all the desirable properties of Bézier curves. In practice, rational Bézier curves/surfaces/solids in $\mathbb{R}^d$ are algorithmically treated as geometric entities in $\mathbb{R}^{d+1}$ by means of projective transformations, see Patterson, 1985. Thus, applying this projective transformation to a $k-$th degree rational Bézier curve $C_k(t)$ in $\mathbb{R}^3$ leads to a nonrational Bézier curve $C_k^w(t)$ in $\mathbb{R}^4$ as follows

$$C_k^w(t) = (\tilde{x}(t), \tilde{y}(t), \tilde{z}(t), \tilde{w}(t)), \tag{1.16}$$

where

$$\tilde{x}(t) = \sum_{j=0}^{k} w_j x_j B_j^k(t), \tag{1.17}$$

$$\tilde{y}(t) = \sum_{j=0}^{k} w_j y_j B_j^k(t), \tag{1.18}$$

$$\tilde{z}(t) = \sum_{j=0}^{k} w_j z_j B_j^k(t), \tag{1.19}$$

$$\tilde{w}(t) = \sum_{j=0}^{k} w_j B_j^k(t). \tag{1.20}$$

Thus, in order to evaluate the rational curve, one has only to evaluate the four fields of C_k^w by applying de Casteljau's algorithm and then divide the first three fields

$(\tilde{x}(t), \tilde{y}(t), \tilde{z}(t))$ by $\tilde{w}(t)$.

Remark: Bézier curves can also be parametrized in any closed interval $[a, b] \subset \mathbb{R}$ with a variable change $\tilde{t} = a + t(b - a)$. In addition, one can split the domain $[a, b]$ into subintervals such that more Bézier polygons are obtained without changing the original shape of the curve. Thus, each polygon controls a local part of the curve leading to a more local control and design of the geometry. However, these techniques are out of the scope of this work, for a further reading see Gardan and Casteljau, 1985; Farin, 2001.

In geometric modeling of surfaces, Bézier techniques are also applied and Bézier surfaces are constructed by means of tensor product. In this framework, the basis of Bézier surfaces are bivariate Bernstein polynomials given by

$$B_{i,j}^{k,l}(s,t) := B_i^k(s) B_j^l(t), \quad \text{with } 0 \leq s, t \leq 1. \tag{1.21}$$

Note that different polynomial degrees can be considered for each parametric direction. Bézier surfaces make a good use of these polynomials and the so-called control nets $\{\mathbf{P}_{i,j}\}_{\substack{i=0,\dots,k \\ j=0,\dots,l}}$ guide their shape:

$$S(s,t) := \sum_{i=0}^{k} \sum_{j=0}^{l} \mathbf{P}_{i,j} B_{i,j}^{k,l}(s,t). \tag{1.22}$$

Rational Bézier surfaces can be constructed following the same procedure carried for curves. In this case, the rational polynomial functions are defined as follows

$$R_{i,j}^{k,l}(s,t) := \frac{w_{i,j} B_{i,j}^{k,l}(s,t)}{\sum_{i=0}^{k} \sum_{j=0}^{l} w_{i,j} B_{i,j}^{k,l}(s,t)}, \quad \text{with } 0 \leq s, t \leq 1, \tag{1.23}$$

where $\{w_{i,j}\}_{\substack{i=0,\dots,k \\ j=0,\dots,l}}$ is the corresponding net of weights. Hence, rational Bézier surfaces are given by

$$S(s,t) := \sum_{i=0}^{k} \sum_{j=0}^{l} \mathbf{P}_{i,j} R_{i,j}^{k,l}(s,t). \tag{1.24}$$

These surfaces can model many geometries as commented before. Finally, we show in Figure 1.5 a quartic rational Bézier surface of the top hemisphere centered at the origin with radius $r = 1$, see Eberly, 2016. In order to understand this parametrization, the control points and their corresponding weights are also shown in Figure 1.6. With this last example, one can fully understand the power and utility of Bézier's method for geometric modelling.

1.3 B-spline basis functions

As pointed out in Prautzsch, Boehm, and Paluszny, 2002, the word spline firstly appeared in the shipbuilding industries. In this context, splines were elastic beams intended for modelling the shape of the ships hull. These beams were fixed and weights were placed at specific points such that the spline adopted a desired shape. From the mathematical point of view, spline functions (splines) were firstly developed by I. J. Schoenberg in Schoenberg, 1946a; Schoenberg, 1946b. In those two papers, splines were introduced as piecewise polynomial curves for the interpolation

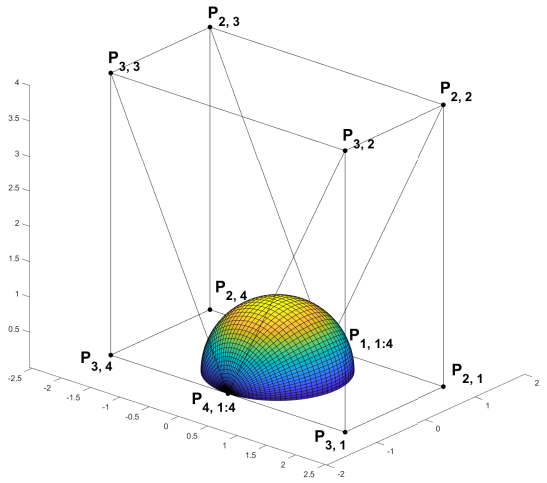


FIGURE 1.5: Hemisphere
Top hemisphere centered at the origin with radius $r = 1$: Quartic rational Bézier parametrization with its control net.

i	j	$\mathbf{P}_{i,j}$	$w_{i,j}$
1	1	$(0,1,0)$	1
1	2	$(0,1,0)$	1/3
1	3	$(0,1,0)$	1/3
1	4	$(0,1,0)$	1
2	1	$(2,1,0)$	1/3
2	2	$(2,1,4)$	1/9
2	3	$(-2,1,4)$	1/9
2	4	$(-2,1,0)$	1/3
3	1	$(2,-1,0)$	1/3
3	2	$(2,-1,4)$	1/9
3	3	$(-2,-1,4)$	1/9
3	4	$(-2,-1,0)$	1/3
4	1	$(0,-1,0)$	1
4	2	$(0,-1,0)$	1/3
4	3	$(0,-1,0)$	1/3
4	4	$(0,-1,0)$	1

FIGURE 1.6: Hemisphere:
Control points and their corresponding weights for the quartic rational Bézier parametrization shown in Figure 1.5.

of functions and some of their derivatives at some interpolation points. In the literature, many works have been devoted to the approximation and interpolation using splines, see Ahlberg, Nilson, and Walsh, 1967; Boor, 1978; Lyche and Schumaker, 1973; Schoenberg, 1973; Heinemann, 1977. Hence, let us introduce the definition of spline functions as given in Prautzsch, Boehm, and Paluszny, 2002:

Definition 1.3.1. A curve $C(\xi)$ is a spline of degree k with knots or breakpoints $\xi_1, \ldots, \xi_{m+k+1}$, where $\xi_i \leq \xi_{i+1}$ and $\xi_i < \xi_{i+k+1}$ for all possible i, if

- $C(\xi)$ is $k - r$ times differentiable at any $r-$times repeated knot.
- $C(\xi)$ is a polynomial of degree $\leq k$ over each knot interval $[\xi_i, \xi_{i+1}]$, for $i = 1, \ldots, m + k$.

In other words, a $k - th$ degree univariate spline is a mapping $C : [\xi_1, \xi_{m+k+1}] \to \mathbb{R}$ such that the image of each knot span is a $k-$th degree polynomial and the smoothness at each knot depends on its repetitions. Also, note that with this definition the global smoothness of splines can vary from continuous to $k - 1$ differentiable if we allow to repeat each internal knot k times at most. In order to represent spline curves in Bézier form, that is, as a linear combination of some control points P_i, special $k-$th degree basis functions $\hat{N}_i^k$ must be used. These basis functions were firstly called B-splines by Schoenberg in Schoenberg, 1967 and their recursive definition was found by de Boor and Mansfield (de Boor, 1972) and Cox (Cox, 1972). First, given a knot vector $\Xi = \{\xi_1, \ldots, \xi_{m+k+1}\}$ with $\xi_i \leq \xi_{i+1}$, $i = 1, \ldots, m + k$, the piecewise constant B-splines are defined as step functions

$$\hat{N}_i^0(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise,} \end{cases} \tag{1.25}$$

where $i = 1, \ldots, m$. Then, B-spline basis functions of degree j with $1 \leq j \leq k$ are defined by means of the Cox-de Boor recursion formula:

$$\hat{N}_i^j(\xi) = \frac{\xi - \xi_i}{\xi_{i+j} - \xi_i} \hat{N}_i^{j-1}(\xi) + \frac{\xi_{i+j+1} - \xi}{\xi_{i+j+1} - \xi_{i+1}} \hat{N}_{i+1}^{j-1}(\xi), \tag{1.26}$$

with $1 \leq i \leq m + k - j$. Note that in some cases a $0/0$ quotient might appear in (1.26), these terms are always considered equal to 0, see Piegl and Tiller, 1996. As mentioned before, some repeated knots might be considered in knot vector Ξ and it consequently effects on the smoothness of the spline curve at the breakpoints. In order to show the wide range of possibilies provided by knots repetitions, in Figure 1.7 we show the smoothness of cubic B-splines basis functions through internal knots obtained with a knot vector $\Xi = \{0, 0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{2}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, 1, 1, 1, 1\}$ where all the possible numbers of knots repetitions appear.

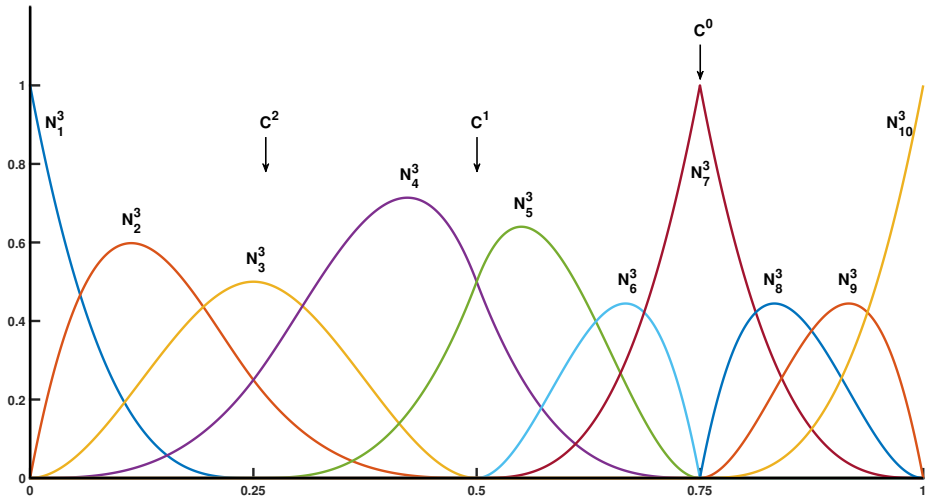


FIGURE 1.7: B-splines
Smoothnesses of cubic B-splines basis functions through internal knots obtained with a knot vector $\Xi = \{0, 0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{2}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, 1, 1, 1, 1\}$.

In practice, the end-point interpolation property is desirable for splines. Due to this, it is common to consider the so-called open knot vectors, that is, knot vectors whose first and last knots are repeated exactly $k + 1$ times. Also, the knots are usually taken equally spaced since there might not be any known reason to group more knots in a local part of the curve/surface. This type of knot vectors with equally spaced knots are called uniform knot vectors and they are often used as a first approach for the numerical approximation of PDEs.

1.4 Properties of B-splines

B-splines functions overcome many issues related to Bernstein polynomials as basis for Bézier curves. For instance, Bézier curves restrict the number of control points to $k + 1$, where k is the polynomial degree. Hence, if many control points were needed, an excessively high polynomial degree would be required. On the other hand, it is also possible to apply subdivision and deal with composite Bézier curves, but this does not seem to be very suitable for geometric modelling from a practical point of

view. In addition, control points have a global effect on Bézier curves due to the global support of Bernstein polynomials. In order to surpass these difficulties, B-splines offer a local control and spline degree is only conditioned by the complexity of the computational domain geometry. Let us introduce the following B-splines properties (see Piegl and Tiller, 1996; Prautzsch, Boehm, and Paluszny, 2002):

- Compact support:
$$\text{supp}(\hat{N}_i^k) = [\xi_i, \xi_{i+k+1}]\,. \tag{1.27}$$
- Nonzero basis functions on each knot span: In each knot interval, at most $k+1$ B-spline basis functions do not vanish.
- Nonnegativity:
$$\hat{N}_i^k(\xi) \geq 0, \quad \forall i,k \in \mathbb{N} \text{ and } \forall \xi \in \mathbb{R}. \tag{1.28}$$
In addition:
$$\hat{N}_i^k(\xi) > 0, \quad \text{for } \xi \in (\xi_i, \xi_{i+k+1}). \tag{1.29}$$
- Partition of unity:
$$\sum_{i=1}^{m} \hat{N}_i^k(\xi) = 1, \quad \xi \in [\xi_1, \xi_{m+k+1}]. \tag{1.30}$$
- Differentiability: There exist all the derivatives of basis functions in the interior of knot intervals. The derivative of a B-spline basis function is given by
$$\frac{d}{d\xi}\hat{N}_i^k(\xi) = k\left(\frac{1}{\xi_{i+k}-\xi_i}\hat{N}_i^{k-1}(\xi) - \frac{1}{\xi_{i+k+1}-\xi_{i+1}}\hat{N}_{i+1}^{k-1}(\xi)\right). \tag{1.31}$$
- Recursion formulas for $j-$th derivatives of B-splines:
$$\frac{d^j}{d\xi^j}\hat{N}_i^k(\xi) = k\left(\frac{1}{\xi_{i+k}-\xi_i}\frac{d^{j-1}}{d\xi^{j-1}}\hat{N}_i^{k-1}(\xi) - \frac{1}{\xi_{i+k+1}-\xi_{i+1}}\frac{d^{j-1}}{d\xi^{j-1}}\hat{N}_{i+1}^{k-1}(\xi)\right). \tag{1.32}$$
- Linear independency: B-splines $\{\hat{N}_i^k\}_{i=1}^m$ with a knot sequence $\xi_1, \ldots, \xi_{m+k+1}$ form a basis for all splines of degree k.

As the reader might observe, B-splines hold all the good properties of Bernstein polynomials. In fact, B-splines can be understood as an extension of Bernstein basis since open knot vectors without inner knots provide the same polynomial basis for each k. Furthermore, B-splines basis functions offer a local control of the geometry due to (1.27) and the number of basis functions can be increased by adding knots.

1.5 B-spline curves and surfaces

B-spline curves imitate the Bézier form for geometry representation by using B-splines basis functions as polynomial basis. Hence, a B-spline curve is given by

$$\mathbf{C}(\xi) = \sum_{i=1}^{m} \mathbf{P}_i \hat{N}_i^k(\xi), \tag{1.33}$$

where $\mathbf{P}_i$, $i = 1, \ldots, m$ are control points. As an example, in Figure 1.8 we show a cubic B-spline curve with knot sequence $\Xi = \{0,0,0,0,\frac{1}{4},\frac{1}{2},\frac{3}{4},1,1,1,1\}$ and control

points $\mathbf{P}_1 = (0,0)$, $\mathbf{P}_2 = (\frac{1}{8},1)$, $\mathbf{P}_3 = (\frac{3}{8},1)$, $\mathbf{P}_4 = (\frac{1}{2},0)$, $\mathbf{P}_5 = (\frac{5}{8},-1)$, $\mathbf{P}_6 = (\frac{3}{4},\frac{3}{2})$ and $\mathbf{P}_7 = (1,\frac{1}{2})$. In the case of B-spline curves, control points do not have a global effect on the geometry due to property (1.27). In fact, any control point $\mathbf{P}_i$ affects the B-spline curve only in $[\xi_i, \xi_{i+k+1}]$ and inversely the polynomial curve described in $[\xi_i, \xi_{i+1}]$ depends only on the control points $\mathbf{P}_{i-k}, \ldots, \mathbf{P}_i$. Additionally, B-splines inherit the variation diminishing property and the convex hull property such that at any knot interval $[\xi_i, \xi_{i+1}]$ the curve lies in the convex hull of the corresponding control points $\mathbf{P}_{i-k}, \ldots, \mathbf{P}_i$.

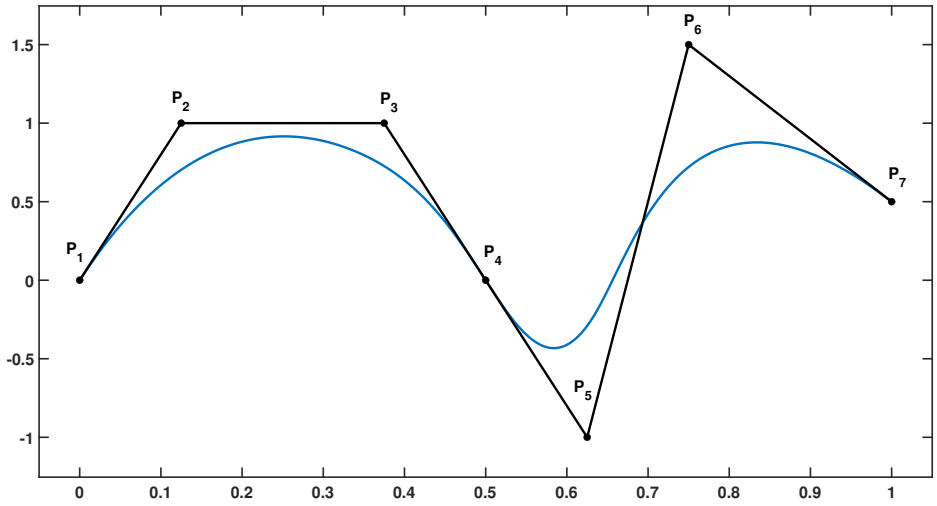


FIGURE 1.8: B-spline curve
Cubic B-spline curve with knot sequence $\Xi = \{0,0,0,0,\frac{1}{4},\frac{1}{2},\frac{3}{4},1,1,1,1\}$ and control points $\mathbf{P}_1 = (0,0)$, $\mathbf{P}_2 = (\frac{1}{8},1)$, $\mathbf{P}_3 = (\frac{3}{8},1)$, $\mathbf{P}_4 = (\frac{1}{2},0)$, $\mathbf{P}_5 = (\frac{5}{8},-1)$, $\mathbf{P}_6 = (\frac{3}{4},\frac{3}{2})$ and $\mathbf{P}_7 = (1,\frac{1}{2})$.

The same way that de Casteljau found an efficient algorithm for the point evaluation of Bézier curves, de Boor invented an efficient and numerically stable algorithm for the point evaluation of B-spline curves, see de Boor, 1972. In order to evaluate the B-spline curve at $\xi \in [\xi_1, \xi_{m+k+1}]$ his algorithm firstly finds the knot interval $[\xi_i, \xi_{i+1})$ such that $\xi \in [\xi_i, \xi_{i+1})$. At this point, let us denote $\mathbf{P}_j^0 = \mathbf{P}_j$ for $j = i-k, \ldots, i$. Thus, the recursive de Boor's evaluation algorithm reads as follows

$$\mathbf{P}_j^l(\xi) = \frac{\xi_{j+k+1-l} - \xi}{\xi_{j+k+1-l} - \xi_j}\mathbf{P}_{j-1}^{l-1}(\xi) + \frac{\xi - \xi_j}{\xi_{j+k+1-l} - \xi_j}\mathbf{P}_j^{l-1}(\xi), \tag{1.34}$$

where $l = 1, \ldots, k$ and $j = i-k+l, \ldots, i$. Finally, one obtains $\mathbf{C}(\xi) = \mathbf{P}_i^k(\xi)$. In Figure 1.9, we show the intermediate control points computed during de Boor's algorithm in order to evaluate at $\xi = \frac{1}{8}$ the previous curve given in Figure 1.8. For the same example, we show in Figure 1.10 the triangular scheme of computed points during de Boor's algorithm.

By looking more deeply into this evaluation algorithm, one realizes that de Boor's algorithm mimics the cutting process carried out by de Casteljau for the point evaluation of Bézier curves. Indeed, this fact might be even clearer if the reader compares Figure 1.9 with Figure 1.3. In the literature, more sophisticated techniques have been developed for the efficient evaluation of B-splines. For instance, we can evaluate the $k+1$ non-vanished B-splines basis functions for a given $\xi \in \Xi$ and then compute

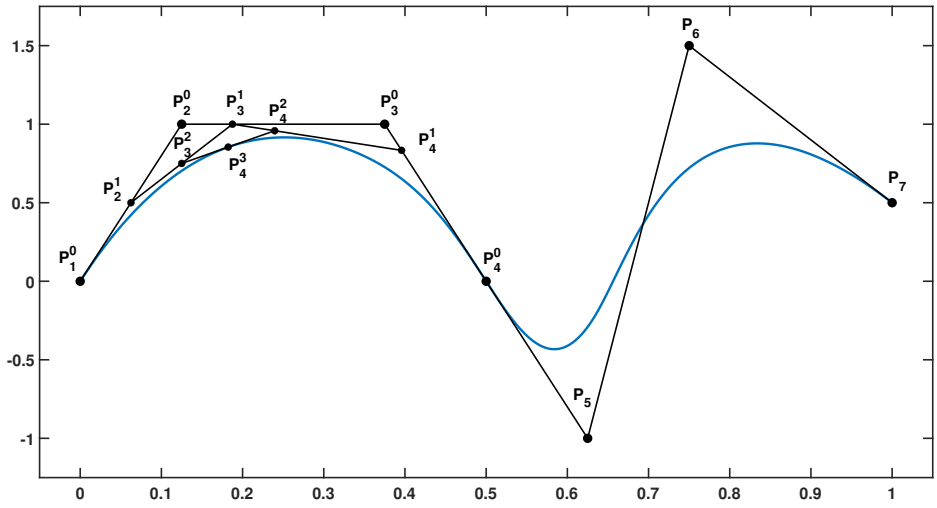


FIGURE 1.9: de Boor's algorithm
Intermediate control points computed during de Boor's algorithm in order to evaluate at $\xi = \frac{1}{8}$ the previous curve given in Figure 1.8.

(1.33). Furthermore, for this purpose we can avoid to repeat some computations by saving the recursive terms given in (1.26) when our task is to evaluate the corresponding B-spline basis functions, see Piegl and Tiller, 1996.

As we mentioned before, the number of basis functions describing a B-spline can be increased by inserting some additional knots to the knot vector Ξ. Thus, a more local and flexible control of the B-spline is achieved by considering a different basis $\{\hat{M}_i^k\}_{i=1}^{\bar{m}}$ for the B-spline representation of the curve with a refined knot vector $\bar{\Xi}$. Therefore, the new basis allows us to represent the same curve without any parametric nor geometric change since span $\{\hat{N}_1^k, \ldots, \hat{N}_m^k\} \subset$ span $\{\hat{M}_1^k, \ldots, \hat{M}_{\bar{m}}^k\}$. That is:

$$\mathbf{C}(\xi) = \sum_{i=1}^{m} \mathbf{P}_i \hat{N}_i^k(\xi) = \sum_{i=1}^{\bar{m}} \mathbf{Q}_i \hat{M}_i^k(\xi), \tag{1.35}$$

where $\bar{m}$ denote the new number of B-spline basis functions and $\mathbf{Q}_i$ are the new control points. At this point, the actual problem is to compute the new control points $\{\mathbf{Q}_i\}_{i=1}^{\bar{m}}$. This process is known as knot insertion and it is one of the most important algorithms for splines. Knot insertion was developed by CAGD community in order to render and intersect B-splines, see Goldman and Lyche, 1992. The first general knot insertion algorithm was developed by Boehm in Boehm, 1980. This algorithm allows to insert one new knot $\bar{\xi}$ into Ξ by detecting the non-degenerate knot interval such that $\bar{\xi} \in [\xi_j, \xi_{j+1})$. Hence, the new control points are calculated as follows

$$\mathbf{Q}_i = \alpha_i \mathbf{P}_i + (1 - \alpha_i)\mathbf{P}_{i-1}, \tag{1.36}$$

where

$$\alpha_i = \begin{cases} 1, & \text{if } i \leq j-k, \\ \dfrac{\bar{\xi} - \xi_i}{\xi_{i+k} - \xi_i}, & \text{if } j-k+1 \leq i \leq j, \\ 0, & \text{if } i \geq j+1. \end{cases} \tag{1.37}$$

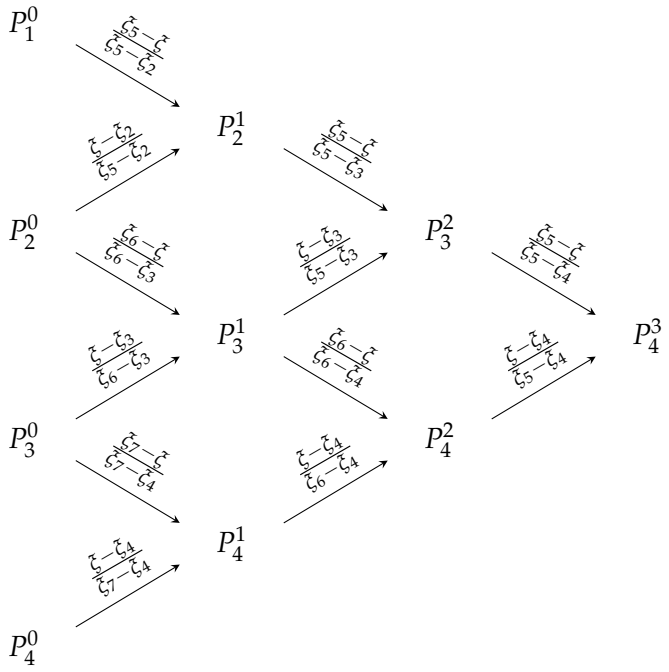


FIGURE 1.10: de Boor's algorithm
Triangular scheme of computed points during de Boor's algorithm in order to evaluate at $\bar{\xi} = \frac{1}{8}$ the cubic B-spline curve given in Figure 1.8.

Moreover, it can be very interesting to insert the same knot multiple times in one step. Thus, if the knot ξ_j has multiplicity s in the knot vector Ξ and has to be inserted r times (with $s + r \leq k$) then the formula for the new control points changes to (see Piegl and Tiller, 1996):

$$\mathbf{Q}_{i,r} = \alpha_{i,r}\mathbf{P}_{i,r-1} + (1 - \alpha_{i,r})\mathbf{P}_{i-1,r-1}, \tag{1.38}$$

where $\mathbf{Q}_{i,r}$ denotes the $i-$th control point in the $r-$th insertion of ξ_j and

$$\alpha_{i,r} = \begin{cases} 1, & \text{if } i \leq j - k + r - 1, \\ \dfrac{\bar{\xi} - \xi_i}{\xi_{i+k-r+1} - \xi_i}, & \text{if } j - k + r \leq i \leq j - s, \\ 0, & \text{if } i \geq j - s + 1. \end{cases} \tag{1.39}$$

In Figure 1.11, we show the B-spline curve given in Figure 1.8 after insertion of a knot at $\bar{\xi} = 7/8$ and its new control polygon with control points $\mathbf{Q}_1 = (0, 0)$, $\mathbf{Q}_2 = (\frac{1}{8}, 1)$, $\mathbf{Q}_3 = (\frac{3}{8}, 1)$, $\mathbf{Q}_4 = (\frac{1}{2}, 0)$, $\mathbf{Q}_5 = (\frac{29}{48}, -\frac{5}{6})$, $\mathbf{Q}_6 = (\frac{23}{32}, \frac{7}{8})$, $\mathbf{Q}_7 = (\frac{7}{8}, 1)$, $\mathbf{Q}_8 = (1, \frac{1}{2})$.

Despite we can insert several knots one by one by applying Boehm's algorithm, it might be necessary to insert many knots and it is desirable from the computational point of view to do it simuntaneously. For this purpose, Cohen, Lyche and Riesenfeld developed the so-called Oslo algorithm in Cohen, Lyche, and Riesenfeld, 1980. In addition, this algorithm was improved in Lyche and Mørken, 1986; Goldman, 1990 and knot insertion algorithms still receive a lot of attention in the literature, for further information see Goldman and Lyche, 1992. The opposite process to knot insertion is knot deletion. Knot deletion is not suitable to any B-spline since it might change the geometry of the curve.

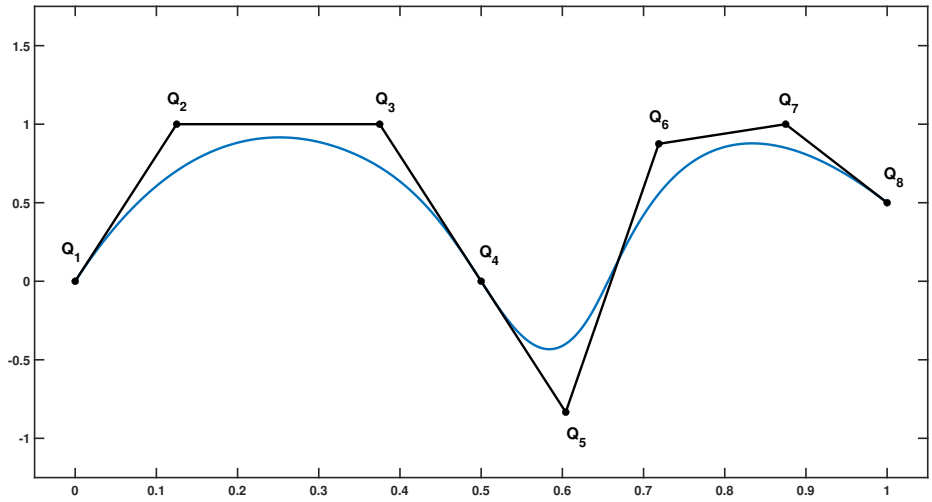


FIGURE 1.11: Knot insertion
B-spline curve given in Figure 1.8 after insertion of a knot at $\bar{\xi} = 7/8$ and its new control polygon with control points $\mathbf{Q}_1 = (0,0)$, $\mathbf{Q}_2 = (\frac{1}{8},1)$, $\mathbf{Q}_3 = (\frac{3}{8},1)$, $\mathbf{Q}_4 = (\frac{1}{2},0)$, $\mathbf{Q}_5 = (\frac{29}{48},-\frac{5}{6})$, $\mathbf{Q}_6 = (\frac{23}{32},\frac{7}{8})$, $\mathbf{Q}_7 = (\frac{7}{8},1)$, $\mathbf{Q}_8 = (1,\frac{1}{2})$.

An important derivation of knot insertion is degree elevation. This process consists in elevate the spline degree to $k+1$ in (1.33) such that the curve does not change parametrically nor geometrically. In this case, the goal of degree elevation is to find control points $\{\mathbf{Q}_i\}$ such that

$$\mathbf{C}(\xi) = \sum_{i=1}^{m} \mathbf{P}_i \hat{N}_i^k(\xi) = \sum_{i=1}^{\bar{m}} \mathbf{Q}_i \hat{M}_i^{k+1}(\xi). \tag{1.40}$$

Note that this process is always possible since the initial B-spline basis for the curve $\mathbf{C}$ generates a spline space embedded in the new one spanned by B-spline basis functions of degree $k+1$, see Cohen, Lyche, and Schumaker, 1986. The simplest way to compute the new control points is to evaluate (1.40) in adequate collocation points and then solve the corresponding linear system. However, given that this process is really expensive, many degree elevation algorithms have been developed, see Prautzsch, 1984; Cohen, Lyche, and Schumaker, 1986. A very popular degree elevation algorithm was developed by Piegl and Tiller in Piegl and Tiller, 1994 based on applying knot insertions until each internal knot is folded k times. Hence, that technique divides the B-spline curve into Bézier segments, then the degree elevation is carried out for each segment and finally the unnecesary knots are deleted. The opposite process to degree elevation is degree reduction. As it occurs with knot deletion, this algorithm is not always suitable to any curve. Thus, an artificial criteria must be set in order to determine whether a curve is degree reducible or not, see Piegl and Tiller, 1996.

B-spline surfaces are constructed by means of tensor product of two univariate B-splines. Thus, bivariate B-spline basis functions are given by

$$\hat{N}_{i,j}^{k,l}(\xi,\eta) := \hat{N}_i^k(\xi)\hat{N}_j^l(\eta), \quad \text{with } 0 \leq \xi,\eta \leq 1, \tag{1.41}$$

where k and l are the spline degrees considered for each direction. Note that for

the bivariate case, two different knot sequences Ξ and $\mathcal{H}$ can be considered for the parametric space (usually taken as $[0,1]^2$). Hence, B-spline surfaces are constructed as a linear combination of bivariate B-spline basis functions multiplied by a net of control points $\{\mathbf{P}_{i,j}\}_{\substack{i=1,\dots,m_1\\ j=1,\dots,m_2}}$ that determine the geometrical mapping:

$$S(\xi,\eta) := \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \mathbf{P}_{i,j}\hat{N}_{i,j}^{k,l}(\xi,\eta). \tag{1.42}$$

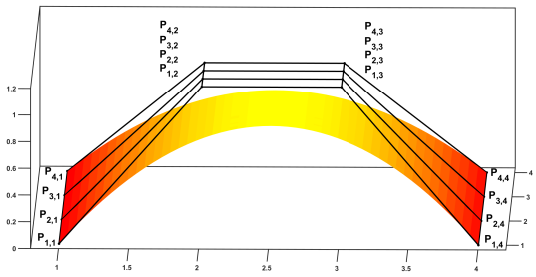
In Figure 1.12 we show a bicubic B-spline surface with knot vectors $\Xi = (0,0,0,0,1,1,1,1)$ and $\mathcal{H} = (0,0,0,0,1,1,1,1)$ and its control points. Evaluation algorithms for the univariate case such as de Boor algorithm can be extended to B-spline surfaces by applying it at both directions. However, for this purpose we can also evaluate the non-vanishing univariate B-spline basis functions and perform a vector-matrix-vector product as follows:

$$S(\xi,\eta) := \left(\hat{N}_{i-k}^{k}(\xi),\dots,\hat{N}_{i}^{k}(\xi)\right)\begin{pmatrix} \mathbf{P}_{i-k,j-l} & \dots & \mathbf{P}_{i-k,j} \\ \vdots & \vdots & \vdots \\ \mathbf{P}_{i,j-l} & \dots & \mathbf{P}_{i,j} \end{pmatrix}\left(\hat{N}_{j-l}^{l}(\eta),\dots,\hat{N}_{j}^{l}(\eta)\right), \tag{1.43}$$

with $\xi \in [\xi_i,\xi_{i+1})$ and $\eta \in [\eta_j,\eta_{j+1})$. In addition, this strategy is also useful for evaluating the partial derivatives of B-spline surfaces. Let us denote with $\dfrac{\partial^{r+s}}{\partial^r\xi\partial^s\eta}S(\xi,\eta)$ the r times partial derivative with respect to ξ and s times partial derivative with respect to η of S. Then, applying (1.41):

$$\frac{\partial^{r+s}}{\partial^r\xi\partial^s\eta}S(\xi,\eta) := \sum_{c_1=0}^{k}\sum_{c_2=0}^{l}\frac{\partial^r}{\partial^r\xi}\hat{N}_{i-k+c_1}^{k}(\xi)\mathbf{P}_{i-k+c_1,j-l+c_2}\frac{\partial^s}{\partial^s\eta}\hat{N}_{j-l+c_2}^{l}(\eta). \tag{1.44}$$

The algorithms for knots insertion, knot deletion, degree elevation and degree reduction are applied to B-spline surfaces by applying the univariate case to both directions. Thus, the implementation of these algorithms for the bivariate case is just a tedious task based on the same schemes that we do not include here, for the interested reader on the implementational details see Piegl and Tiller, 1996.



i	j	$\mathbf{P}_{i,j}$
1	1	(1,1,0)
1	2	(2,2,0)
1	3	(3,2,0)
1	4	(4,1,0)
2	1	(1,2,1)
2	2	(2,7/3,1)
2	3	(3,7/3,1)
2	4	(4,2,1)
3	1	(1,3,1)
3	2	(2,8/3,1)
3	3	(3,8/3,1)
3	4	(4,3,1)
4	1	(1,4,0)
4	2	(2,3,0)
4	3	(3,3,0)
4	4	(4,4,0)

FIGURE 1.12: B-spline surface
Bicubic B-spline surface with knot vectors $\Xi = (0,0,0,0,1,1,1,1)$ and its control points.

1.6 NURBS

Non-rational uniform B-splines(NURBS) enlarge the capabilities of B-splines by assigning weights to each basis function such that the partition of unity property holds. Univariate NURBS basis functions of spline degree k are defined as follows

$$\hat{R}_i^k(\xi) = \frac{w_i \hat{N}_i^k(\xi)}{\sum_{j=1}^{m} w_j \hat{N}_j^k(\xi)}, \tag{1.45}$$

where $\{w_j\}_{j=1}^m$ is a given set of weights. These basis functions hold all the good properties shown for B-splines if all the weights are positive and they allow us to capture the exact geometry of conic sections. In Figure 1.13 we show cubic NURBS basis functions generated with a knot vector $\Xi = (0,0,0,0,\frac{1}{4},\frac{1}{2},\frac{3}{4},1,1,1,1)$ and weights $w_1 = 1$, $w_2 = 1$,$w_3 = 2$, $w_4 = 1$, $w_5 = \frac{1}{2}$, $w_6 = 1$ and $w_7 = 1$. As a special case, when all weights $w_j = 1$ for $j = 1 \ldots, m$ this definition of NURBS basis functions coincide with the B-spline counterpart since B-spline basis functions hold partition of unity property (1.30). In addition, they do coincide with rational Bézier polynomials when there are not internal knots.

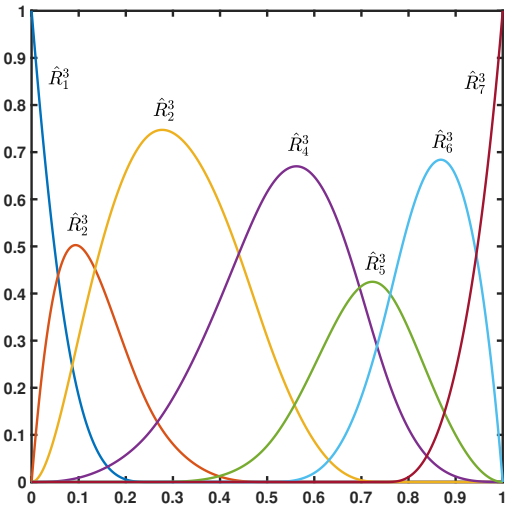


FIGURE 1.13: NURBS basis functions
Cubic NURBS basis functions generated with a knot vector $\Xi = (0,0,0,0,\frac{1}{4},\frac{1}{2},\frac{3}{4},1,1,1,1)$ and weights $w_1 = 1$, $w_2 = 1$, $w_3 = 2$, $w_4 = 1$, $w_5 = \frac{1}{2}$, $w_6 = 1$ and $w_7 = 1$.

Following the same procedure for B-splines, NURBS curves are defined by means of these basis functions and control points as follows:

$$\mathbf{C}(\xi) = \sum_{i=1}^{m} \mathbf{P}_i \hat{R}_i^k(\xi). \tag{1.46}$$

The efficient evaluation of NURBS curves is based on the combination of B-splines evaluation techniques and projective transformation as in (1.16). Hence, a k-th degree NURBS curve in $\mathbb{R}^3$ can be described as follows

$$\mathbf{C}^w(\xi) = (\tilde{x}(\xi), \tilde{y}(\xi), \tilde{z}(\xi), \tilde{w}(\xi)), \tag{1.47}$$

where

$$\tilde{x}(\xi) = \sum_{i=1}^{m} w_j x_j \hat{N}_i^k(\xi), \tag{1.48}$$

$$\tilde{y}(\xi) = \sum_{i=1}^{m} w_j y_j \hat{N}_i^k(\xi), \tag{1.49}$$

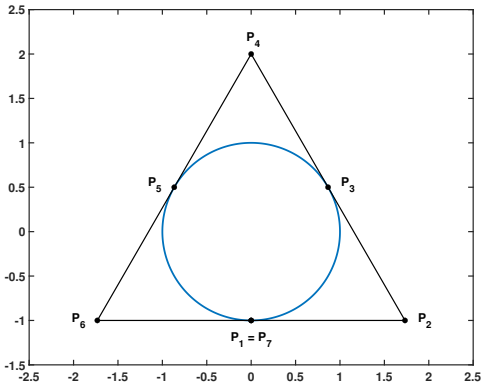
$$\tilde{z}(\xi) = \sum_{i=1}^{m} w_j z_j \hat{N}_i^k(\xi), \tag{1.50}$$

$$\tilde{w}(\xi) = \sum_{i=1}^{m} w_j \hat{N}_i^k(\xi), \tag{1.51}$$

and (x_i, y_i, z_i) are the three components of the control points $\{\mathbf{P}_{i=1}^m\}$. Thus, one is able to perform a B-spline evaluation of these fields and then obtain the actual evaluation of the curve by computing

$$\mathbf{C}(\xi) = \left(\frac{\tilde{x}(\xi)}{\tilde{w}(\xi)}, \frac{\tilde{y}(\xi)}{\tilde{w}(\xi)}, \frac{\tilde{z}(\xi)}{\tilde{w}(\xi)} \right). \tag{1.52}$$

Therefore, by applying this approach, all the algorithms presented for B-splines can be used for NURBS too. In Figure 1.14 we show a planar quadratic NURBS curve with knot vector $\Xi = (0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, 1, 1, 1)$ depicting the unit circle centered at the origin and its control points, see the details on how to obtain NURBS constructions of circles in Piegl and Tiller, 1996.



i	$\mathbf{P}_i$	w_i
1	$(0, -1)$	1
2	$(\sqrt{3}, -1)$	$\frac{1}{2}$
3	$(\frac{\sqrt{3}}{2}, \frac{1}{2})$	1
4	$(0, 2)$	$\frac{1}{2}$
5	$(-\frac{\sqrt{3}}{2}, \frac{1}{2})$	1
6	$(-\sqrt{3}, -1)$	$\frac{1}{2}$
7	$(0, -1)$	1

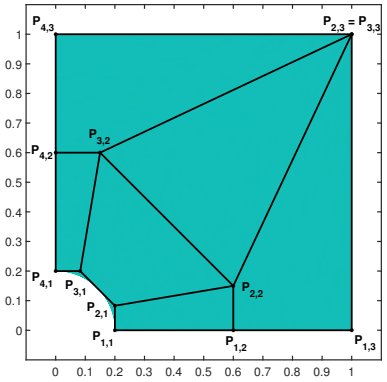
FIGURE 1.14: NURBS curve
Planar quadratic NURBS curve with knot vector $\Xi = (0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, 1, 1, 1)$ depicting the unit circle centered at the origin and its control points.

NURBS are also defined for multivariate cases by considering the corresponding multivariate NURBS basis functions and assigning weights to them. For the bivariate case, $k-$th degree NURBS basis functions are defined as follows

$$\hat{R}_{i,j}^{k,l}(\xi, \eta) = \frac{w_{i,j} \hat{N}_{i,j}^{k,l}(\xi, \eta)}{\sum_{s_1=1}^{m_1} \sum_{s_2=1}^{m_2} w_{s_1,s_2} \hat{N}_{s_1,s_2}^{k,l}(\xi, \eta)}. \tag{1.53}$$

With these basis functions, NURBS surfaces are constructed by means of

$$S(\xi,\eta) := \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \mathbf{P}_{i,j}\hat{R}_{i,j}^{k,l}(\xi,\eta). \tag{1.54}$$



i	j	$\mathbf{P}_{i,j}$	$w_{i,j}$
1	1	$(\frac{1}{5},0)$	1
1	2	$(\frac{3}{5},0)$	1
1	3	$(1,0)$	1
2	1	$(\frac{1}{5},\frac{\sqrt{(2)}-1}{5})$	$\frac{1+\sqrt{(2)}}{2\sqrt{2}}$
2	2	$(\frac{3}{5},\frac{3}{20})$	1
2	3	$(1,1)$	1
3	1	$(\frac{\sqrt{(2)}-1}{5},\frac{1}{5})$	$\frac{1+\sqrt{(2)}}{2\sqrt{2}}$
3	2	$(\frac{3}{20},\frac{3}{5})$	1
3	3	$(1,1)$	1
4	1	$(0,\frac{1}{5})$	1
4	2	$(0,\frac{3}{5})$	1
4	3	$(0,1)$	1

FIGURE 1.15: NURBS Surface
Planar biquadratic $k = l = 2$ NURBS surface with knot sequences $\Xi = (0,0,0,1,1,1)$ and $\mathcal{H} = (0,0,0,\frac{1}{2},1,1,1)$ together with the coordinates of the control points and their corresponding weights.

Note that algorithms for B-splines surfaces are extended to NURBS surfaces applying the projective transformation as done for the univariate case. In Figure 1.15, we show a planar biquadratic $k = l = 2$ NURBS surface with knot sequences $\Xi = (0,0,0,1,1,1)$ and $\mathcal{H} = (0,0,0,\frac{1}{2},1,1,1)$ together with the coordinates of the control points and their corresponding weights. Hence, the evolution from Bézier curves to B-splines and finally to NURBS proves that NURBS represent the most general of these geometric design tools and they will be used for the numerical approximation of PDEs along the next chapter.

Chapter 2

Isogeometric Analysis

2.1 Introduction

In computer engineering, the most used tool for numerical simulations has been typically finite element analysis (FEA). The reason why finite elements were so attractive to analysts is their simplicity and suitability for a wide range of problems such as fluid dynamics, structural mechanics, thermodynamics, etc. This analysis had its origins at the 1950s, being the first published paper on FEA the one by Turner et al., 1956, but the term "finite elements" was firstly coined in Clough, 1960. FEA is based on basis functions with an interpolatory behavior at the nodes. In fact, these basis functions are typically Lagrange polynomials and they are known as interpolating functions. Moreover, in FEM the use of Lagrage polynomials was not restricted to the numerical approximation of partial differential equations (PDEs) but also to representation of computational domains. However, finite element meshes allowed only to approximate the exact geometry of simple computational domains such as polygonal domains. On the other hand, the actual geometries for real models are designed with CAD and consequently any FEM mesh refinement requires communication with the corresponding CAD model. This existing gap between analysis and design working on the same models has been an important bottleneck in order to improve analysis efficiency. In this framework, one of the most concerned people on overcoming this issue was Thomas J.R. Hughes and he pursuited the integration of CAD and FEA. Together with J.A. Cottrell and Y. Bazilevs, the result of their efforts was the invention of isogeometric analysis (IGA) firstly mentioned in their paper Cottrell, Hughes, and Bazilevs, 2005. IGA is a numerical technique based on the use of spline-type basis functions for the numerical analysis and the design of the computational domain. The main purpose of IGA is to delete any gap between CAD and FEA by applying basis functions typically used for geometric modelling to the approximation of the numerical solution of simulations carried out in the corresponding models. Thus, it is possible to overcome any unefficiency of model creation problem for its analysis. Since its first appearance, isogeometric analysis has gained a huge appreciation and relevance in the numerical analysis community. Thus, the seminal work Cottrell, Hughes, and Bazilevs, 2009 on IGA was published to introduce this approach to new starters. This topic has received a lot of attention in the last years, being applied to many fields such as fluid dynamics (Tagliabue, Dedè, and Quarteroni, 2014; Nielsen et al., 2011; Akkerman et al., 2011), structural analysis (Cottrell et al., 2006; Reali, 2006; Cottrell, Hughes, and Reali, 2007), optimization problems (Dedè, Borden, and Hughes, 2012; Li and Qian, 2011; Seo, Kim, and Youn, 2010), etc.

In this chapter, the basics of isogeometric analysis based on B-splines and NURBS are explained. In addition, we apply IGA for the discretization of several classical

problems through Galerkin's approximation method. As a starting point, the unidimensional and bidimensional versions of the Poisson equation are treated in Section 2.2.1. Secondly, we consider the heat equation in Section 2.2.2 as an evolutionary problem. Then, the biharmonic equation is given in Section 2.2.3 as an example of high-order problem. In addition, we cover vector field problems with linear elasticity problem in Section 2.2.4. Finally, we solve the Stokes equations in Section 2.2.5 in order to illustrate the good performace of isogeometric discretizations for saddle point problems.

2.2 Isoparametric approach

The isoparametric concept consists to apply the same basis functions for geometry representation and numerical approximation of the solution. IGA and FEA apply this approach and show a similar procedure for the Galerkin approximation of PDEs. In addition, both of them are suitable for numerical analysis since they propose basis functions with compact support, hold partition of unity property and their discretizations yield matrices with a bandwidth related to the polynomial degree k. However, there are many differences between these analysis as we point out in Table 2.1. Firstly they apply the isoparametric approach by means of different polynomials, typically Lagrange polynomials in FEM and B-splines in IGA. In FEM, the basis interpolates the numerical solution at the nodes, but the degrees of freedom are harder to interpret in IGA since spline-type basis functions are non-interpolatory. IGA benefits from CAD technology and works on the exact geometry of every model whereas FEA can only achieve an approximation of it. In addition, IGA and FEA differ in the global smoothness of their basis. It is well known that the first one provides discretizations with a $\mathcal{C}^0$ inter-element continuity, but isogeometric discretizations hold up to $\mathcal{C}^{k-1}$ global continuity. Furthermore, although classical $h-$ and $p-$refinements can be considered for both of them, IGA introduces a new refinement strategy called $k-$refinement (see Cottrell, Hughes, and Bazilevs, 2009). This strategy consists of increasing the polynomial degree and the global smoothness of the basis at the same time. The pure $k-$refinement starts with one single element that is firstly $p-$refined and then uniform knot insertion is carried out. Thus, the number of elements is increased as in FEA but the number of basis functions is reduced and the global smoothness is maximum $\mathcal{C}^{k-1}$.

Finite element analysis	Isogeometric analysis
Lagrange polynomials (in general)	B-splines
Interpolatory basis	Non-interpolatory basis
Approximation of the geometry	Exact representation of the geometry
$\mathcal{C}^0$ inter-element continuity	Up to $\mathcal{C}^{k-1}$ global continuity
Classical $h-$ and $p-$refinements	Additional new $k-$ refinement

TABLE 2.1: FEA vs IGA
Differences between FEA and IGA.

The classical isoparametric approach considers a reference element with parametric coordinates. Then, for the construction of each element in the physical domain a different geometrical transformation is applied. Nevertheless, this procedure changes for isogeometric discretizations since there are two different notions of

element (see Cottrell, Hughes, and Bazilevs, 2009). The first notion of element corresponds to patches. Patches are usually related to the macro-element concept since they contain smaller structures called knot-spans. On the other hand, a knot-span is the parametric subdomain delimited by two consecutive knots (for each parametric direction). Given that knot-spans are the smallest structures in the parameter space, we identify the classical concept of element with knot spans. Moreover, in IGA the same B-spline or NURBS geometrical transformation is applied to each patch and that transformation is locally transferred to every knot-span. Knot vectors contain knots that serve to set the elements in the parameter domain, but they allow knots repetitions as mentioned in Section 1.3. Hence, knots repetitions yield knot-spans with zero length that are not relevant as elements. Due to this, it is convenient to introduce an index space in order to discriminate zero length knot spans. Finally, during the matrix assembly numerical integration is performed at each element. Although more efficient quadrature rules have been developed for IGA discretizations (see Hughes, Reali, and Sangalli, 2010; Auricchio et al., 2012), the standard approach is to apply Gauss quadrature rules. Thus, the parent domain is usually taken as $[-1,1]^d$ where d denotes the number of parametric spatial directions. In Figure 2.1, we show a biquadratic isoparametric construction scheme using knot vectors $\Xi = \{0,0,0,0.5,1,1,1\}$ and $\mathcal{H} = \{0,0,0,0.25,0.5,0.75,1,1,1\}$.

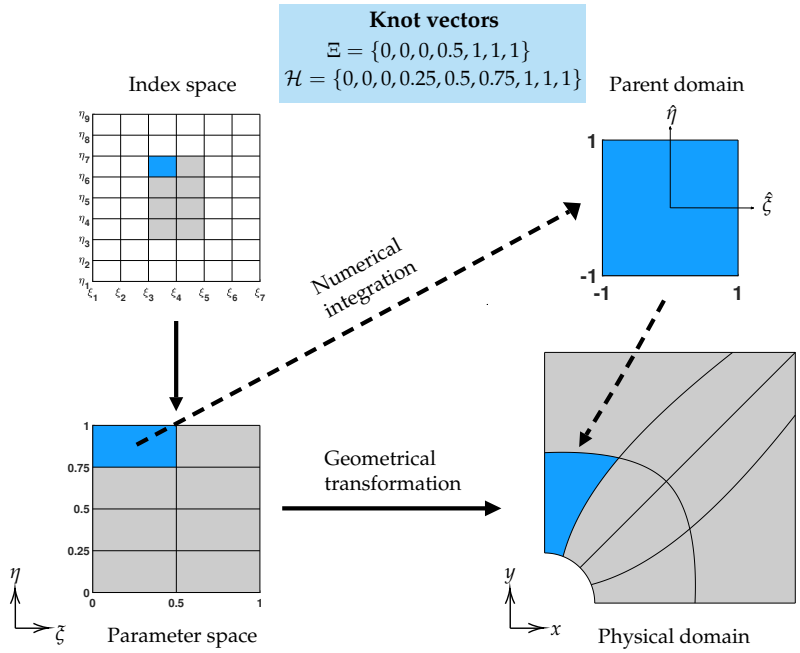


FIGURE 2.1: Biquadratic isoparametric construction scheme using knot vectors $\Xi = \{0,0,0,0.5,1,1,1\}$ and $\mathcal{H} = \{0,0,0,0.25,0.5,0.75,1,1,1\}$. First, the indices of the knots delimiting every knot span are given in the index space. Second, the parametric location of every element is represented in the parameter space. Then, a geometrical transformation is applied to the whole patch and every element is mapped to the physical domain. Finally, numerical integration is performed at the parent domain during the matrix assembly.

The NURBS basis functions defined in (1.53) act on the parameter space $\hat{\Omega}$. However, for the numerical solution in the computational domain we need a basis defined on Ω. In order to define these basis functions, we introduce the geometrical

mapping

$$F : \hat{\Omega} \longrightarrow \Omega, \tag{2.1}$$

with the form given in (1.54), assigning to each pair of parametric coordinates (ξ, η) the corresponding physical coordinates (x, y). Thus, any function $v : \Omega \to \mathbb{R}$ is constructed as the composition

$$v = \hat{v} \circ F^{-1}, \tag{2.2}$$

where $\hat{v} : \hat{\Omega} \to \mathbb{R}$ denotes any function defined on the parametric space $\hat{\Omega}$. In the isoparametric approach, the B-spline/NURBS basis employed for the transformation in (2.1) are the same basis functions with form (2.2) to approximate the solution of PDEs.

In this chapter, we consider Garlerkin's method for the numerical approximation of several model problems. Instead of working with the strong form of these problems, this method uses a weak form of it, also called variational formulation. For this formulation, trial and test spaces have to be appropriately chosen. Then, a function in the trial space is said to be a weak solution if it holds the variational equations for all the test functions. Secondly, finite dimensional approximations of these spaces are considered such that we can generate a discrete problem. Then, the last point is to obtain the discrete solution. However, in the discrete problem some integrals on Ω are involved and they have to be well approximated by means of proper quadrature rules:

$$\int_{\Omega} f d\Omega \approx \sum_{i=1}^{n_q} w_i f(q_i), \tag{2.3}$$

which are determined by quadrature points q_i, associated quadrature weights w_i, $i = 1, \ldots, n_q$ and the number of quadrature points is denoted by n_q. At this point, the usual procedure is to compute the integrals on each element and then obtain the sum of all of them. For this purpose, Gauss quadrature rules are pretty used since they have proved to be exact for polynomial functions of degree up to $2n_q - 1$ with n_q the number of quadrature points. The implementation of these rules requires to introduce linear mappings adapted to each element as follows

$$\phi_e : [-1, 1]^d \longrightarrow \hat{\Omega}_e, \tag{2.4}$$

where the elements in the parameter space (knot spans) are denoted with $\hat{\Omega}_e$, $e = 1, \ldots, n_e$ and n_e is the number of elements. Thus, these linear mappings allocate the quadrature points to the corresponding element where the basis functions are then evaluated. Finally, the discretization yields a linear system of equations where the unknowns are the coefficients that, by multiplying the corresponding basis functions of the approximation space, yield the numerical solution of the method. In IGA, the basis for approximation spaces are B-splines or NURBS basis functions and the degrees of freedom correspond to the values of the control points for the solution. Finally, the isogeometric analysis applying Galerkin's method is explained for several problems in the following subsections: The unidimensional and bidimensional versions of Poisson equation in Section 2.2.1, the heat equation in Section 2.2.2, biharmonic equation in Section 2.2.3, linear elasticity in Section 2.2.4 and the Stokes equations in Section 2.2.5.

2.2.1 Poisson equation

For the first model problem, we consider the simple unidimensional case ($d = 1$) of the Poisson equation. Thus, let us introduce the following two-point boundary problem with computational domain $\Omega = (0,1)$:

$$-u''(x) = f(x),\ x \in \Omega,\ u(0) = u(1) = 0, \tag{2.5}$$

where f is a continuous source function. Therefore, note that a strong solution of this problem has to be $u \in \mathcal{C}^2(0,1)$. Given the strong form, the next step is to set up a variational formulation and obtain a weak solution for this problem. First, let us denote with $\mathcal{V}$ the variational space of test functions. Secondly, we test both sides of the equation (2.5) with any test function $v \in \mathcal{V}$. In order to do this, both sides are multiplied by v and then integrated in the interval $(0,1)$:

$$-\int_0^1 u''(x)v(x)dx = \int_0^1 f(x)v(x)dx.$$

Applying integration by parts yields

$$\int_0^1 u'(x)v'(x)dx - \left[u'(x)v(x)\right]_0^1 = \int_0^1 f(x)v(x)dx. \tag{2.6}$$

Note that a necessary imposition of this problem is $\mathcal{V} = H^1(0,1)$, where the Sobolev space $H^1(0,1)$ is defined as follows

$$H^1(0,1) = \left\{v \in L^2(0,1) \middle| v' \in L^2(0,1)\right\}. \tag{2.7}$$

At this point, the next step is to set the trial space $\mathcal{U} = \mathcal{V}$ such that the test functions v also hold the homogeneous Dirichlet boundary conditions. Thus, the test space $\mathcal{V}$ is

$$H_0^1(0,1) = \left\{v \in H^1(0,1) \middle| v(0) = v(1) = 0\right\}. \tag{2.8}$$

Hence, the second term at the left-hand side of (2.6) vanishes. Finally, the variational formulation of (2.5) reads as follows: Find $u \in \mathcal{V}$ such that for every $v \in \mathcal{V}$,

$$a(u,v) = b(v), \tag{2.9}$$

where

$$a(u,v) = \int_0^1 u'(x)v'(x)dx, \qquad b(v) = \int_0^1 f(x)v(x)dx.$$

The existence and uniqueness of solution to this variational problem is proved via Lax-Milgram theorem using that the bilinear form a is bounded and coercive. In addition, any weak solution of problem (2.9) is also a solution of (2.5). The numerical approximation of the solution requires a discrete version of the variational formulation. For this purpose, let us consider finite dimensional approximation subspaces $\mathcal{V}_h \subset H_0^1(0,1)$. Hence, the discrete problem is given by: Find $u_h \in \mathcal{V}_h$ such that for every $v_h \in \mathcal{V}_h$:

$$a(u_h, v_h) = b(v_h). \tag{2.10}$$

In IGA the choice of $\mathcal{V}_h$ is a spline space. For this case, the parametric space and the physical space coincide ($x = \xi$) for this problem and therefore the geometrical

mapping is the identity. Thus, we will use B-spline basis functions as basis for the approximation space. A priori, the logical approach is to consider a uniform control mesh since we do not know the behavior of the solution. Thus, let us denote with n_e the number of knot spans with nonzero length, that is, the number of elements. Moreover, we denote with $\Xi^k_{s,h}$ an open and uniform knot vector with inner knots repeated $r = k - s$ times, global smoothness $\mathcal{C}^s$ of the corresponding basis functions, spline degree k and knot spans of length $h = 1/n_e$. In addition, let us denote the i-th element with $I_i = ((i-1)h, ih)$. Thus, the knot vector $\Xi^r_{k,h}$ generates a B-spline basis for the corresponding spline space

$$\mathcal{S}^k_s(\Xi^k_{s,h}) = \{u_h \in C^s(0,1) : u_h|_{I_i} \in \mathbb{P}^k, i = 1, \dots, n_e,\ u_h(0) = u_h(1) = 0\}, \tag{2.11}$$

where $\mathbb{P}^k$ denotes the set of polynomials with degree less or equal than k. Hence, we consider $\mathcal{S}^k_s(\Xi^k_{s,h})$ as approximation space since they contain univariate B-spline approximations for the discrete problem (2.10). Note that the dimension of these spaces is given by

$$dim\mathcal{S}^k_s(\Xi^k_{s,h}) = (k-s)(n_e - 1) + k - 1, \tag{2.12}$$

since the first and last B-spline basis functions have to be zero in order to satisfy the Dirichlet boundary conditions. Given this spline approximation space, the numerical solution is given by the following linear combination

$$u_h(x) = \sum_{i=1}^{m} N^k_i(x) u_i \tag{2.13}$$

where we keep using m for the total number of spline basis functions as in Chapter 1 and u_i are the so-called control variables. In order to obtain the control variables, it is enough to test the discrete problem with every B-spline basis function spanning the approximation space $\mathcal{S}^k_s(\Xi^k_{s,h})$. Thus, by using (2.13) in (2.10) we obtain the following equations:

$$\sum_{j=1}^{m} \left(\int_0^1 \frac{dN^k_j(x)}{dx} \frac{dN^k_i(x)}{dx} dx \right) u_j = \int_0^1 f(x) N^k_i(x) dx, \tag{2.14}$$

for $i = 1 \dots, m$. At this point, we only need to compute the integrals of the equation above involving B-spline basis functions, their derivatives and the source term. Once these integrals are computed, we generate a linear system

$$Ku = F, \tag{2.15}$$

where $u = (u_i)$ is the vector of control variables and

$$K_{i,j} = \int_0^1 \frac{dN^k_j(x)}{dx} \frac{dN^k_i(x)}{dx} dx, \qquad F_i = \int_0^1 f(x) N^k_i(x) dx.$$

Instead of computing these integrals directly for each B-spline basis function, they are expressed in terms of the integrals on each element I_i. However, since Gauss quadrature rules are performed on the parent domain $(-1,1)^d$, we need also to introduce the linear mappings given in (2.4). We define them for this case as follows

$$\phi_i : [-1,1] \longrightarrow I_i, \tag{2.16}$$

for $i = 1, \dots, n_e$ and ruled by

$$x = \phi_i(\hat{\xi}) = \frac{1}{2}(\xi_{i+1} - \xi_i)\hat{\xi} + \frac{1}{2}(\xi_{i+1} + \xi_i). \tag{2.17}$$

With the help of connectivity arrays, the index of non-vanishing B-spline basis functions on each element can be determined and saved into a matrix. This last ingredient completes all the requirements for assembling the so-called stiffness matrix and its right-hand side, similar to any FEA implementation. Hence, we show the pseudo-algorithm for the stiffness matrix assembly process in Algorithm 1. Firstly, a loop through the elements is carried out, the corresponding end-points are set and we save the global indices of the B-spline basis functions by recalling the connectivity matrix. Secondly, we perform a second loop over the quadrature points by applying a chosen Gaussian quadrature rule. We save the corresponding coordinate of the quadrature point in the parent domain and its weight. Then, the parametric (physical) coordinates are computed by means of ϕ_i. In addition, we compute $|\phi_i|$ and the derivative of the B-spline basis functions with support in I_i sortened. They are saved in a vector B and then the local stiffness matrix K_e is computed. Finally, the contribution of K_e is added to the stiffness matrix K.

Algorithm 1 Stiffness matrix assembly: One dimensional case with B-splines

for $i = 1 : n_e$ **do** ▷ Loop through elements
  $I_i \leftarrow (\xi_i, \xi_{i+1})$
  $conn \leftarrow connectivity(i, :)$
  for $q = 1 : n_q$ **do** ▷ Loop through the number of quadrature points
    $q_p \leftarrow q-$th quadrature point
    $q_w \leftarrow q-$th quadrature weight
    $x = \phi_i(q_p)$
    $|\phi_i| = \frac{1}{2}(\xi_{i+1} - \xi_i)$
    $B \leftarrow \frac{dN_j^k}{d\xi}(x) \cdot \phi_j^{-1}$, with $j \in conn$
    $K_e = B'B \cdot |F| \cdot |\phi_i| \cdot q_w$
    $K_e \rightarrow K$
  end
end

Remark: Dirichlet boundary conditions can be imposed by making zero the non-diagonal elements of the corresponding rows in K. Then, in order to maintain the number condition of K, the respective diagonal elements are replaced by the mean value of the $diag(K)$. In addition, the boundary conditions are imposed to the right-hand side vector by substituting the corresponding values.

The solution of the discrete problem is a B-spline curve as given in (1.33). In fact, the choice of the approximation space sets the basis with its knot-spans length h and the spline degree k whereas the solution of linear system (2.15) yields the control variables that guide the curve. In order to illustrate this, in Figure 2.2 we show the control variables obtained by solving (2.15) with $k = 2$, $s = 1$ and $h = 1/8$. We plot also the exact solution $u_{ex}(x) = \sin 5\pi x$ and the B-spline curve guided by the control variables.

The error estimates given in Bazilevs et al., 2011 for linear elasticity suggest that NURBS discretizations for Poisson's equation provide $(k+1)-$th order convergence rates in L^2 and $k-$th order convergence rates in H^1:

$$\|u_{ex} - u_h\|_{L^2} \leq Ch^{k+1}, \qquad \|u_{ex} - u_h\|_{H^1} \leq Ch^k, \tag{2.18}$$

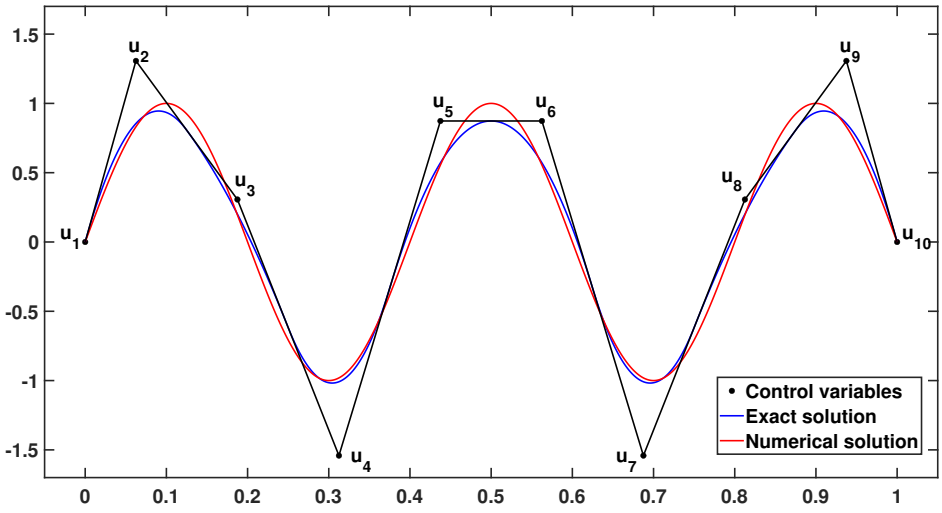


FIGURE 2.2: B-spline curve
Control variables obtained by solving (2.15) with $k = 2$, $s = 1$ and $h = 1/8$. We plot also the exact solution $u_{ex}(x) = \sin 5\pi x$ and the B-spline curve guided by the control variables.

where C is a constant that depends on the exact solution u and the shape of the computational domain. In order to check these results, in Table 2.2 we show the errors in L^2-norm and convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness C^{k-1} and several values of mesh size h for problem (2.5) with exact solution $u_{ex}(x) = \sin 5\pi x$. In addition, the corresponding errors in H^1-norm are shown in Table 2.3.

h	$k=2$		$k=3$		$k=4$	
2^{-6}	$5.1514e-05$	–	$2.3421e-06$	–	$9.4128e-08$	–
2^{-7}	$6.3254e-06$	8.1440	$1.4332e-07$	16.3415	$2.8601e-09$	32.9111
2^{-8}	$7.8712e-07$	8.0362	$8.9098e-09$	16.0858	$8.8783e-11$	32.2142
2^{-9}	$9.8278e-08$	8.0091	$5.5611e-10$	16.0215	$2.7707e-12$	32.0441

TABLE 2.2: 1D Poisson equation
Errors in L^2-norm and convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness C^{k-1} and several values of mesh size h for problem (2.5) with exact solution $u_{ex}(x) = \sin 5\pi x$.

Now, let us consider the bidimensional Poisson equation:

$$\begin{aligned} -\Delta u &= f, \text{ in } \Omega, \\ u &= 0, \text{ on } \Gamma_t, \\ \partial_n u &= g, \text{ on } \Gamma_n, \end{aligned} \tag{2.19}$$

where n denotes the normal vector to Ω, $\partial_n u = \nabla u \cdot n$ is the normal derivative of u, $\partial\Omega = \Gamma_t \cup \Gamma_n$ and $\Gamma_t \cap \Gamma_n = \emptyset$. Again, we look for a weak solution of this problem by setting up its variational formulation. Let $\mathcal{V}$ denote the variational space of weight functions v. If this problem is tested with any weight function $v \in \mathcal{V}$, it yields

h	$k=2$		$k=3$		$k=4$	
2^{-6}	$2.5117e-02$	–	$9.5917e-04$	–	$3.7156e-05$	–
2^{-7}	$6.2452e-03$	4.0218	$1.1851e-04$	8.0936	$2.2911e-06$	16.2177
2^{-8}	$1.5592e-03$	4.0055	$1.4770e-05$	8.0236	$1.4293e-07$	16.0293
2^{-9}	$3.8966e-04$	4.0014	$1.8449e-06$	8.0059	$8.9364e-09$	15.9944

TABLE 2.3: 1D Poisson equation
Errors in H^1-norm and convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.5) with exact solution $u_{ex}(x)=\sin 5\pi x$.

$$-\int_\Omega v\Delta u d\Omega=\int_\Omega f v d\Omega.$$

Now, let us note that

$$\nabla\cdot(v\nabla u)=(\nabla u)\cdot(\nabla v)+v\Delta u.$$

Using this together with Green's theorem, we obtain

$$\int_\Omega(\nabla u)\cdot(\nabla v)\,d\Omega-\int_{\partial\Omega}(v\nabla u)\cdot n dS=\int_\Omega f v d\Omega.$$

Finally, we choose $\mathcal{V}=\{v\in H^1(\Omega)\,|\,v|_{\Gamma_t}=0\}$ and then the homogeneous Dirichlet boundary conditions makes the integral on Γ_t equal to zero. Hence, the variational problem for (2.19) reads as follows: Find $u\in\mathcal{V}$ such that for every $u\in\mathcal{V}$,

$$\int_\Omega(\nabla u)\cdot(\nabla v)\,d\Omega=\int_\Omega f v d\Omega+\int_{\Gamma_n}v\cdot g dS. \tag{2.20}$$

The existence and uniqueness of solution to (2.20) are provided using again the Lax-Milgram theorem. At this point, the discrete problem is formulated by considering finite dimensional approximation spaces $\mathcal{V}_h\subset\mathcal{V}$ that actually are B-spline/NURBS spaces. For the bidimensional case, NURBS discretizations acquire importance in order to capture complex geometries while applying the isoparametric approach. Firstly, we introduce the bivariate B-spline approximation spaces defined on the parametric space, that is, $\Omega=\hat{\Omega}=(0,1)^2$. For this case, the approximation spaces are obtained by means of tensorization of the spaces given in (2.11). Note that different mesh sizes, spline degrees and global smoothnesses can be considered on each direction:

$$\mathcal{S}^{k_1,k_2}_{s_1,s_2}(\Xi^{k_1}_{s_1,h_1},\mathcal{H}^{k_2}_{s_2,h_2}):=\mathcal{S}^{k_1}_{s_1}(\Xi^{k_1}_{s_1,h_1})\times\mathcal{S}^{k_2}_{s_2}(\mathcal{H}^{k_2}_{s_2,h_2}). \tag{2.21}$$

Now, let us introduce B-spline approximation spaces when the physical space Ω differs from the parametric space $\hat{\Omega}$. In this case, the geometrical mapping $\mathbf{F}$ defined as a B-spline transformation in (1.42) takes an important role for the construction of the approximation space $\mathcal{V}_h$. In fact, given a bivariate B-spline approximation space $\mathcal{S}$ on $\hat{\Omega}$ the actual B-spline approximation space defined on Ω is given by

$$\mathcal{V}_h=\mathcal{S}\circ\mathbf{F}^{-1}. \tag{2.22}$$

Therefore, this space is spanned by basis functions with the form $N^{k,l}_{i,j}:=\hat{N}^{k,l}_{i,j}\circ\mathbf{F}^{-1}$, with $\hat{N}^{k,l}_{i,j}$ defined as in (1.41), k,l the spline degrees on each direction, $i=1,\dots,m_1$, $j=1,\dots,m_2$, and m_1,m_2 the number of basis functions on each direction,

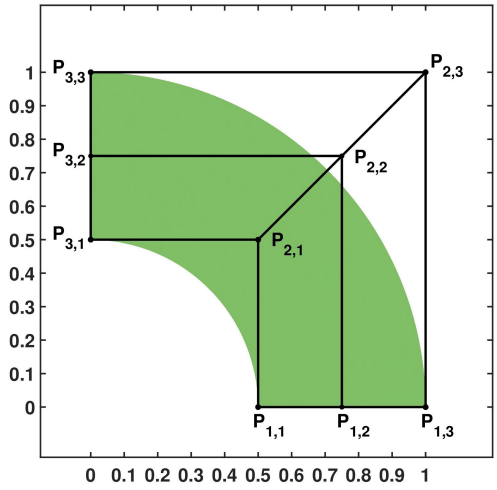
respectively. Hence, the B-spline approximation of (2.19) is given by:

$$u_h(\mathbf{x}) = \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} N_{i,j}^{k,l}(\mathbf{x})u_{i,j}. \tag{2.23}$$

For the NURBS approximation of problem (2.19), in general it is not possible to apply a tensor product of univariate NURBS spaces since different weights are assigned to every basis function. Thus, we consider bivariate NURBS basis functions defined on $\hat{\Omega}$, that is, the basis functions $\{\hat{R}_{i,j}^{k,l}\}_{\substack{i=1,\dots,m_1\\ j=1,\dots,m_2}}$ given in (1.53). Then, the inverse of the NURBS transformation $\mathbf{F}$ is applied to them in order to obtain the NURBS basis on the physical domain Ω:

$$\mathcal{V}_h = \text{span}\{R_{i,j}^{k,l} = \hat{R}_{i,j}^{k,l} \circ \mathbf{F}^{-1}, \quad i = 1,\dots,m_1;\ j = 1,\dots,m_2\}. \tag{2.24}$$

In Figure 2.3, we show the computational domain of a quarter of an annulus with inner radius $r = 0.5$ and outer radius $R = 1$ together with its control points and weights. Using this NURBS transformation, in Figure 2.4 we show the biquadratic NURBS basis functions with $h = 1$, $s = 1$ used for the approximation of problem (2.19) on the quarter of an annulus given in Figure 2.3, applying the isoparametric approach.



i	j	$\mathbf{P}_{i,j}$	$w_{i,j}$
1	1	$(\frac{1}{2}, 0)$	1
1	2	$(\frac{3}{4}, 0)$	1
1	3	$(1, 0)$	1
2	1	$(\frac{1}{2}, \frac{1}{2})$	$\frac{1}{\sqrt{2}}$
2	2	$(\frac{3}{4}, \frac{3}{4})$	$\frac{1}{\sqrt{2}}$
2	3	$(1, 1)$	$\frac{1}{\sqrt{2}}$
3	1	$(0, \frac{1}{2})$	1
3	2	$(0, \frac{3}{4})$	1
3	3	$(0, 1)$	1

FIGURE 2.3: Quarter of annulus
Computational domain of a quarter of an annulus with inner radius $r = 0.5$ and outer radius $R = 1$ together with its control points and weights.

Hence, the corresponding NURBS approximation of (2.19) is given by:

$$u_h(\mathbf{x}) = \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} R_{i,j}^{k,l}(\mathbf{x})u_{i,j}. \tag{2.25}$$

B-spline and NURBS discretizations yield a linear system with the form (2.15) and the unknowns are again the control variables that determine the approximation u_h. Thus, the entries of the stiffness matrix K using NURBS basis functions are computed as follows

$$K_{i,j} = \int_\Omega \nabla R_i(\mathbf{x}) \cdot \nabla R_j(\mathbf{x}) d\mathbf{x}, \tag{2.26}$$

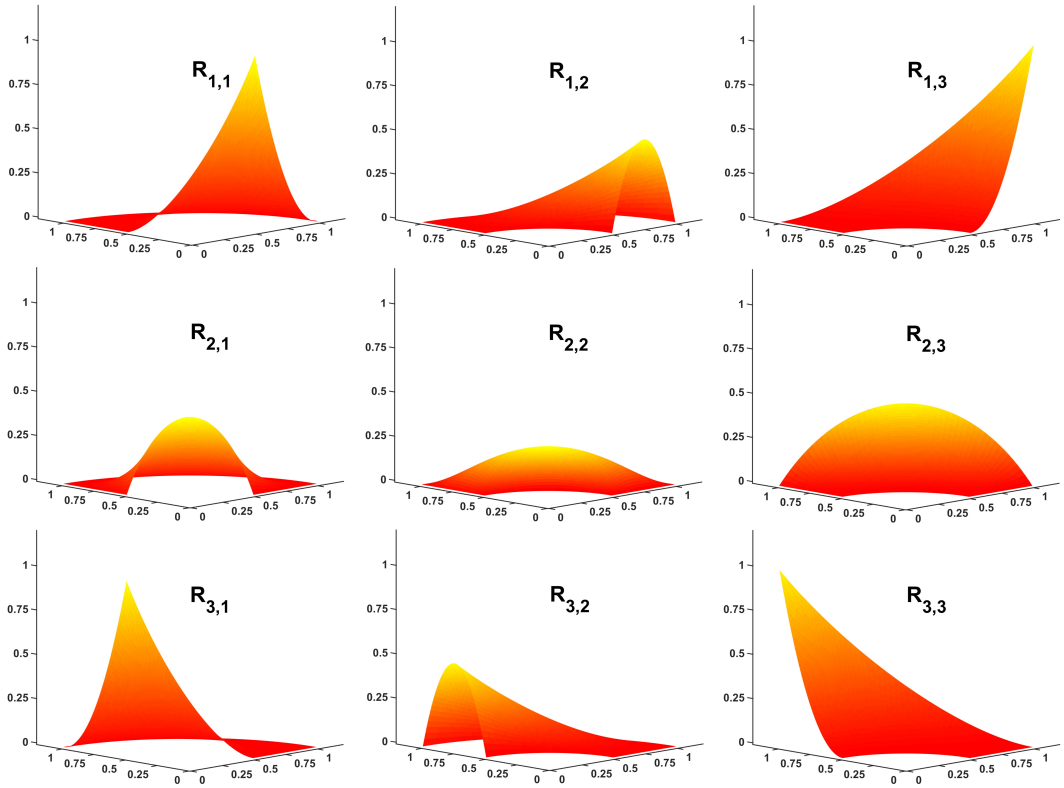


FIGURE 2.4: Biquadratic NURBS basis functions with $h = 1$, $s = 1$ used for the approximation of problem (2.19) on the quarter of an annulus given in Figure 2.3 applying the isoparametric approach.

where the indices i, j are global indices that are related to the indices of each direction by means of an "element nodes" array (IEN array), see Cottrell, Hughes, and Bazilevs, 2009. However, the assembly process is again carried out element by element. Let us denote with K_e the local stiffness matrix corresponding to element Ω_e and let $\mathbf{R_e}$ denote the vector of NURBS basis functions with support in Ω_e. Hence:

$$K_e = \int_{\Omega_e} \nabla \mathbf{R_e}(\mathbf{x}) \cdot \nabla \mathbf{R_e}(\mathbf{x}) d\mathbf{x}. \tag{2.27}$$

For the computation of these integrals, a change of variable $\boldsymbol{\xi} = (\xi, \eta) = F^{-1}(x, y)$ is applied such that

$$K_e = \int_{\hat{\Omega}_e} \left(JF^{-T} \nabla \hat{\mathbf{R}}_\mathbf{e}(\xi) \right)^T \cdot JF^{-T} \nabla \hat{\mathbf{R}}_\mathbf{e}(\xi) \, |JF| \, d\boldsymbol{\xi}, \tag{2.28}$$

where JF^{-T} denotes the transposed jacobian matrix of the NURBS transformation F^{-1}. In the bivariate case, given an element $\hat{\Omega}_e$ limited by $[\xi_i, \xi_{i+1}]$ and $[\eta_j, \eta_{j+1}]$ the linear mapping $\phi_e : [-1, 1]^2 \to \hat{\Omega}_e$ such that $\phi_e(\tilde{\xi}, \tilde{\eta}) = (\xi, \eta)$ is ruled by

$$\xi = \xi_i + \frac{\xi_{i+1} - \xi_i}{2}(\tilde{\xi} + 1),$$
$$\eta = \eta_j + \frac{\eta_{j+1} - \eta_j}{2}(\tilde{\eta} + 1).$$

The efficient assembly of mass and stiffness matrices is a key point specially for high-order spline discretizations. With tensor-product spline basis functions, one of

the top numerical techniques for this purpose is the so-called sum-factorization. This technique was firstly introduced in Orszag, 1980 for spectral methods in complex geometries and then applied to high-order FEM (see Melenk, Gerdes, and Schwab, 2001; Ainsworth, Andriamaro, and Davydov, 2011). More recently, this technique has been applied to IGA in Antolin et al., 2015 and further studied in Bressan and Takacs, 2019. In addition to this, the representation of tensor-product IGA matrices by means of kronecker products of univariate/lower-dimensional auxiliar matrices was presented in Mantzaflaris et al., 2015. A simple but useful case is the construction of bivariate mass and stiffness matrices using B-splines on $\hat{\Omega}$ by means of kronecker products of lower-dimensional mass and stiffness matrices. Let us denote with $M_i^{(1)}$ and $K_i^{(1)}$ the univariate mass and stiffness matrices, respectively, on the $i-$th spatial direction, $i = 1, 2$. Then, the corresponding bivariate B-spline mass $M^{(2)}$ and stiffness $K^{(2)}$ matrices on $(0,1)^2$ are given by

$$M^{(2)} = M_1^{(1)} \otimes M_2^{(1)}, \tag{2.29}$$

$$K^{(2)} = M_1^{(1)} \otimes K_2^{(1)} + K_1^{(1)} \otimes M_2^{(1)}. \tag{2.30}$$

This technique is not restrictive to the whole global matrix, it is also really useful for the construction of stencils and therefore helpful for matrix-free implementations. Note that these constructions can be extended to higher-dimensional cases. As pointed out in Hofreither, 2018, the multivariate mass and stiffness matrices with d spatial dimensions on the parametric domain are obtained as follows

$$M^{(d)} = M_1^{(d-1)} \otimes M_2^{(1)}, \tag{2.31}$$

$$K^{(d)} = M_1^{(d-1)} \otimes K_2^{(1)} + K_1^{(d-1)} \otimes M_2^{(1)}. \tag{2.32}$$

As expected, the convergence order of bivariate B-splines do coincide with the one-dimensional case in $\|\cdot\|_{L^2}$ and $\|\cdot\|_{H^1}$. In Table 2.4 and Table 2.5, we show the errors in L^2-norm and H^1-norm respectively next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the square domain $\Omega = (0,1)^2$ with exact solution $u_{ex}(x,y) = \sin(5\pi x)\sin(5\pi y)$ and homogeneous Dirichlet boundary conditions on $\Gamma_t = \Gamma$. In these tables, we can see that the convergence order tends to $k+1$ in L^2 and k in H^1.

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$4.4e-03$	$-$	$8.5329e-04$	$-$	$1.5032e-04$	$-$
2^{-5}	$4.3917e-04$	10.0145	$4.0609e-05$	21.0124	$3.3506e-06$	44.8641
2^{-6}	$5.1451e-05$	8.5356	$2.3421e-06$	17.3389	$9.4126e-08$	35.5963
2^{-7}	$6.3234e-06$	8.1366	$1.4332e-07$	16.3415	$2.8601e-09$	32.9105

TABLE 2.4: 2D Poisson equation: Square domain.
Errors in L^2-norm next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the square domain $\Omega = (0,1)^2$ with exact solution $u_{ex}(x,y) = \sin(5\pi x)\sin(5\pi y)$ and homogeneous Dirichlet boundary conditions on $\Gamma_t = \Gamma$.

It was shown in Cottrell, Hughes, and Bazilevs, 2009 that NURBS discretizations yield the same convergence rates obtained with FEA. In Table 2.6 and Table 2.7 we

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$4.515e-01$	–	$7.65e-02$	–	$1.30e-02$	–
2^{-5}	$1.029e-01$	4.3882	$8.1e-03$	9.5082	$6.3538e-04$	20.4488
2^{-6}	$2.51e-02$	4.0943	$9.5987e-04$	8.3874	$3.7186e-05$	17.0866
2^{-7}	$6.2e-03$	4.0234	$1.1853e-04$	8.0981	$2.2915e-06$	16.2274

TABLE 2.5: 2D Poisson equation: Square domain.
Errors in H^1-norm next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the square domain $\Omega=(0,1)^2$ with exact solution $u_{ex}(x,y)=\sin(5\pi x)\sin(5\pi y)$ and homogeneous Dirichlet boundary conditions on $\Gamma_t=\Gamma$.

show the errors in L^2-norm and H^1-norm respectively next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the quarter of an annulus described in Figure 2.3 with exact solution

$$u_{ex}(x,y)=\sin(5\pi x)\sin(5\pi y)\left(x^2+y^2-1/4\right)\left(x^2+y^2-1\right),$$

and homogeneous Dirichlet boundary conditions on $\Gamma_t=\Gamma$. Here, we observe that despite a faster convergence for coarse meshes, the error reduction with NURBS discretizations tends to imitate the behavior shown for B-splines.

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$1.8615e-03$	–	$8.2871e-04$	–	$1.5990e-04$	–
2^{-5}	$1.2770e-04$	14.5771	$2.3173e-05$	35.7615	$4.2286e-06$	95.4313
2^{-6}	$1.3055e-05$	9.7819	$1.1131e-06$	20.8186	$9.0398e-08$	46.7772
2^{-7}	$1.5423e-06$	8.4645	$6.4525e-08$	17.2508	$2.5166e-09$	35.9203

TABLE 2.6: 2D Poisson equation: Quarter of an annulus.
Errors in L^2-norm next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the quarter of an annulus described in Figure 2.3 with exact solution $u_{ex}(x,y)=\sin(5\pi x)\sin(5\pi y)\left(x^2+y^2-1/4\right)\left(x^2+y^2-1\right)$ and homogeneous Dirichlet boundary conditions on $\Gamma_t=\Gamma$.

2.2.2 Heat equation

The heat equation is a parabolic partial differential equation that models the heat diffusion through a domain $\Omega\subset\mathbb{R}^d$. The strong formulation of this problem with homogeneous Dirichlet boundary conditions is given by

$$\begin{aligned} \partial_t u-\Delta u &= f, \text{ in } \Omega\times(0,T),\\ u &= 0, \text{ on } \Gamma\times(0,T),\\ u &= u_0, \text{ on } \Omega\times\{0\}. \end{aligned} \tag{2.33}$$

In order to obtain the weak formulation of this problem, we apply the same steps given in Section 2.2.1 by introducing the following variational space $V=H_0^1(\Omega)$. With this variational space, the weak formulation of (2.33) reads as follows: For each

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$1.053e-01$	–	$4.2147e-02$	–	$1.9555e-02$	–
2^{-5}	$2.0413e-02$	5.1608	$3.0621e-03$	13.7642	$5.2368e-04$	37.3425
2^{-6}	$4.7708e-03$	4.2788	$3.2731e-04$	9.3553	$2.5021e-05$	20.9298
2^{-7}	$1.1726e-03$	4.0687	$3.9253e-05$	8.3386	$1.4496e-06$	17.2600

TABLE 2.7: 2D Poisson equation: Quarter of an annulus.
Errors in H^1-norm next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.19) on the quarter of an annulus described in Figure 2.3 with exact solution $u_{ex}(x,y)=\sin(5\pi x)\sin(5\pi y)\left(x^2+y^2-1/4\right)\left(x^2+y^2-1\right)$ and homogeneous Dirichlet boundary conditions on $\Gamma_t=\Gamma$.

$t\in(0,T)$, find $u\in\mathcal{V}$ such that the initial condition $u=u_0$ holds on $\Omega\times\{0\}$ and for every $v\in\mathcal{V}$,

$$\int_\Omega \partial_t u\cdot v d\Omega+\int_\Omega(\nabla u)\cdot(\nabla v)\,d\Omega=\int_\Omega f v d\Omega. \tag{2.34}$$

For the discretization of this parabolic evolution problem, there exist two main approaches: Space-time discretizations and time-stepping methods combined with spatial discretizations such as finite differences (FD), finite volumes (FV) or FEA. In the first group, space-time methods treat time as an extra spatial variable and PDEs are fully discretized at once. These methods generate sparse linear systems with all the unknowns in $\Omega\times(0,T)$, but the global matrix is a kronecker product of other matrices corresponding to the time derivative, mass and stiffness matrix (for a comprehensive description, see Loli et al., 2020). Thus, this structure can be used to apply time-parallel integration techniques for the efficient solution of the global system, see Gander, 2015. In addition, space-time discretizations have been recently combined with IGA in the literature (Loli et al., 2020; Langer, Moore, and Neumüller, 2016; Benedusi et al., 2021).

On the second group, we have the horizontal method (also known as Rothe's method). This method is based on approximating the time derivative by finite differences (that is, the discretization in time is applied directly to the differential equation) and then the spatial discretization is performed. We propose this type of time-stepping methods and more concretely we consider an implicit backward Euler scheme due to its numerical stability. In this approach, the time step τ and a finite approximation space $\mathcal{V}_h$ is set. Let us denote with u_h^i the numerical approximation of (2.34) at time t_i. Then, at each time t_i the following fully discretized problem has to be solved: Find $u_h^i\in\mathcal{V}_h$ such that

$$\int_\Omega\left(u_h^i-u_h^{i-1}\right)\cdot v_h d\Omega+\tau\int_\Omega\nabla u^i\cdot\nabla v_h d\Omega=\tau\int_\Omega f v_h d\Omega. \tag{2.35}$$

In matrix form, it reads,

$$(M+\tau K)\,u^i=\tau F+Mu^{i-1}, \tag{2.36}$$

where M is the mass matrix, K is the stiffness matrix, and u^i is the vector of control variables for the solution at t_i. However, this backward implicit Euler scheme yields only first-order convergence in time whereas the same convergence rates shown in Section 2.2.1 are hold in space. Hence, in Table 2.8 we show the L^2 errors next to the

error reduction for the one-dimensional case of (2.33) with exact solution $u_{ex}(x,t) = \sin(\pi x)\sin(\pi t)$, $T = 1$, using spline degrees $k = 2$ and $k = 3$ with maximum smoothness respectively and several values of h and τ such that the error reduction can be appreciated.

$k=2$	L^2		$k=3$	L^2	
$h=2^{-4}, \tau=2^{-4}$	0.1414	–	$h=2^{-4}, \tau=2^{-4}$	0.1414	–
$h=2^{-5}, \tau=2^{-7}$	0.0181	7.8100	$h=2^{-5}, \tau=2^{-8}$	0.0091	15.5990
$h=2^{-6}, \tau=2^{-10}$	0.0023	7.9814	$h=2^{-6}, \tau=2^{-12}$	$5.6728e-04$	15.9805

TABLE 2.8: 1D Heat equation
L^2 errors next to the error reduction at $T = 1$ for the one-dimensional case of (2.33) with exact solution $u_{ex}(x,t) = \sin(\pi x)\sin(\pi t)$, using spline degrees $k = 2, 3$ with maximum smoothness respectively and several values of h and τ such that the error reduction can be appreciated.

2.2.3 Biharmonic equation

The biharmonic equation is used for modelling some problems in engineering and sciences. More concretely, this equation arises in mechanics, plates theory (see Timoshenko and Woinowsky-Krieger, 1959; Ciarlet, 2002) and fluid dynamics (Girault and Raviart, 2011). The strong formulation of the biharmonic equation in a physical domain $\Omega \subset \mathbb{R}^d$ is given by

$$\begin{aligned} \Delta^2 u &= f, \text{ in } \quad \Omega, \\ u &= 0, \text{ on } \quad \partial\Omega, \\ \nabla u \cdot n &= 0, \text{ on } \quad \partial\Omega, \end{aligned} \tag{2.37}$$

where n is the outer normal vector to $\partial\Omega$. As we can observe, this is a fourth-order partial differencial equation that requires at least $\mathcal{C}^1-$continuous piecewise polynomial basis functions for the approximation of its solution. It is hard to obtain H^2-conforming discretizations in the FEM framework and their implementations are not straightforward. Some FEM approaches for the discretization of (2.37) are the so-called Argyris elements defined by means of fifth-order polynomials on triangular meshes (see Argyris, Fried, and Scharpf, 1968), the Bogner-Fox-Schmit element defined on rectangular meshes by means of cubic polynomials (Bogner, Fox, and Schmit, 1965) or some nonconforming finite elements (Brenner and Scott, 2008). However, IGA is suitable for the numerical solution of 2.37 since isogeometric discretizations offer global $\mathcal{C}^1$ smoothness with $k \geq 2$. In fact, IGA has been applied successfully to the biharmonic equation in Moore, 2018 and Pé de la Riva, Rodrigo, and Gaspar, 2020. For the variational formulation of problem (2.37) we consider the following Sobolev space:

$$H_0^2(\Omega) = \{v \in H^2(\Omega) : v = \nabla v \cdot n = 0 \quad \text{on} \quad \partial\Omega\}.$$

Thus, we test the equation given in (2.37) with trial functions $v \in H_0^2(\Omega)$ and the variational formulation is given by: Find $u \in H_0^2(\Omega)$ such that

$$\int_\Omega \Delta u \, \Delta v \, \mathrm{d}\Omega = \int_\Omega f v \, \mathrm{d}\Omega, \qquad \forall v \in H_0^2(\Omega). \tag{2.38}$$

Now, we introduce a finite dimensional space $V_h \subset H_0^2(\Omega)$ based on B-spline basis functions. Thus, the corresponding discrete problem is given by: Find $u_h \in V_h$ such that

$$\int_\Omega \Delta u_h \, \Delta v_h \, \mathrm{d}\Omega = \int_\Omega f v_h \, \mathrm{d}\Omega. \tag{2.39}$$

The discrete formulation yields a linear system $Au = b$ where the (i, j) entry of the matrix is given by

$$A_{i,j} = \int_\Omega \Delta N_j \, \Delta N_i \, \mathrm{d}\Omega,$$

where $\{N_i\}_{i=1}^{dim\, V_h}$ are the B-splines basis functions that span the discrete space $V_h \subset H_0^2(\Omega)$. Note that in (2.39) we have to compute the second derivatives of the basis functions during the matrix assembly process. In practice, matrix A is obtained by assembling the following elemental matrices:

$$A_e = \int_{\Omega_e} (\Delta N_e) \, (\Delta N_e)' \, \mathrm{d}\Omega_e,$$

where N_e denotes the vector of the B-spline basis functions with support in element Ω_e. For the fast formation of matrix A when $\Omega = \hat{\Omega}^2$, it can be used the following kronecker product of matrices that represent univariate integrals:

$$A^{(2)} = K^{(1)} \otimes M^{(1)} + M^{(1)} \otimes K^{(1)} + 2K^{(1)} \otimes K^{(1)}, \tag{2.40}$$

where $K^{(1)}$ and $M^{(1)}$ are the stiffness and mass matrices of the univariate case. In fact, this approach was used in Pé de la Riva, Rodrigo, and Gaspar, 2020 for the construction of the bivariate global matrix and its stencils. In Table 2.9 we show the errors in H^2-norm next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.37) on the square domain $(0, 1)^2$ with exact solution $u_{ex}(x, y) = \sin(\pi x) \sin(\pi y)$. Finally, as we can see in this table the convergence order for the biharmonic equation is $k - 1$.

h	$k = 2$		$k = 3$		$k = 4$	
2^{-4}	0.3959	–	0.0101	–	$2.9191e-04$	–
2^{-5}	0.1978	2.0013	0.0025	4.0154	$3.7143e-05$	7.8592
2^{-6}	0.0989	2.0003	$6.2691e-04$	4.0040	$4.6920e-06$	7.9163
2^{-7}	0.0494	2.0000	$1.5669e-04$	4.0010	$5.9021e-07$	7.9497

TABLE 2.9: 2D Biharmonic equation
Errors in H^2-norm next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of mesh size h for problem (2.37) on the square domain $(0, 1)^2$ with exact solution $u_{ex}(x, y) = \sin(\pi x) \sin(\pi y)$.

2.2.4 Linear elasticity

Elasticity theory studies the deformations of a body under external forces acting on it. In this context, the external forces applied on the elastic material are called stresses whereas the quantity of deformation is known as strain. First, the strain tensor ε is

defined as follows:

$$\varepsilon(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right), \tag{2.41}$$

where u is the displacement vector. The previous governing equation is known as kinematic compatibility or geometric equation. Stress and strain tensors are related by the generalized Hooke's law:

$$\boldsymbol{\sigma} = \mathbb{C}\varepsilon, \tag{2.42}$$

where $\boldsymbol{\sigma}$ denotes the stress tensor, and $\mathbb{C}$ is a fourth-order elasticity tensor whose elements $c_{i,j,k,l}$ are called elastic coefficients. For our linear elastic problems, we will consider the body to be homogeneous, that is, the elasticity tensor $\mathbb{C}$ shows the following symmetry properties:

$$c_{i,j,k,l} = c_{i,j,l,k}, \tag{2.43}$$

$$c_{i,j,k,l} = c_{j,i,k,l}, \tag{2.44}$$

$$c_{i,j,k,l} = c_{k,l,i,j}. \tag{2.45}$$

In addition, we consider that the body is an isotropic elastic material (in other words, the elastic properties are the same in all directions) and we focus our study on the plane strain framework. With this approach, the stress tensor has only 3 nonzero components that are related to the strain tensor by means of the following equation

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{pmatrix} \begin{pmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{xy} \end{pmatrix}, \tag{2.46}$$

where E is the Young's modulus that measures the stiffness of the body and ν is the Poisson ratio that measures the strain in the elastic material at a perpendicular direction to the stress. Usually, it is interesting to rewrite Hooke's law in terms of the so-called Lamé parameters λ and μ. These parameters are related to E and ν as follows

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \tag{2.47}$$

$$\mu = \frac{E}{2(1+\nu)}. \tag{2.48}$$

Thus, the isotropic linear elastic version of Hooke's law in terms of Lamé parameters is given by

$$\boldsymbol{\sigma} = \lambda \operatorname{tr}(\varepsilon) I + 2\mu\varepsilon, \tag{2.49}$$

where $\operatorname{tr}(\varepsilon)$ is the trace of the strain tensor. Thus, the strong formulation of the linear elastostatic problem is given by:

$$\begin{aligned} \nabla \boldsymbol{\sigma} &= \boldsymbol{f}, \quad \text{in } \Omega, \\ \boldsymbol{u} &= \boldsymbol{0}, \quad \text{on } \Gamma_t, \\ \boldsymbol{\sigma} \cdot \boldsymbol{n} &= \boldsymbol{0}, \quad \text{on } \Gamma_n, \end{aligned} \tag{2.50}$$

where $\Gamma = \Gamma_t \cup \Gamma_n$ is the boundary of the physical domain Ω with $\Gamma_t \cap \Gamma_n = \emptyset$. In this formulation, the stress can be expressed in terms of $\boldsymbol{u}$ by combining (2.49) and

(2.41) giving rise to the displacement formulation:

$$\begin{aligned} (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u})+\mu\nabla^2\boldsymbol{u}+\boldsymbol{f} &= \boldsymbol{0}, \quad \text{in } \Omega, \\ \boldsymbol{u} &= \boldsymbol{0}, \quad \text{on } \Gamma_t, \\ \boldsymbol{\sigma}\cdot\boldsymbol{n} &= \boldsymbol{0}, \quad \text{on } \Gamma_n. \end{aligned} \tag{2.51}$$

Since we are interested in the formulation (2.51), the unknowns are the displacements $\boldsymbol{u}$ and for its weak formulation we introduce the following variational space

$$\mathcal{V} = \{\boldsymbol{u} \in H^1(\Omega)^d \,|\, \boldsymbol{u}|_{\Gamma_t} = 0\}.$$

Thus, the variational formulation of (2.51) reads as follows: Find $\boldsymbol{u} \in \mathcal{V}$ such that

$$2\mu\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v})\, d\Omega + \lambda\int_\Omega (\nabla\cdot\boldsymbol{u})(\nabla\cdot\boldsymbol{v})\, d\Omega + \int_\Omega \boldsymbol{f}\boldsymbol{v}\, d\Omega = \boldsymbol{0}, \quad \forall \boldsymbol{v} \in \mathcal{V}. \tag{2.52}$$

Following the steps given for the previous problems, we introduce a finite dimensional approximation space $V_h \subset \mathcal{V}$ and hence the discrete problem reads: Find $\boldsymbol{u}_h \in V_h$ such that

$$2\mu\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}_h) : \boldsymbol{\varepsilon}(\boldsymbol{v}_h)\, d\Omega + \lambda\int_\Omega (\nabla\cdot\boldsymbol{u}_h)(\nabla\cdot\boldsymbol{v}_h)\, d\Omega + \int_\Omega \boldsymbol{f}\boldsymbol{v}_h\, d\Omega = \boldsymbol{0}, \quad \forall \boldsymbol{v}_h \in V_h. \tag{2.53}$$

The discrete problem results in a linear system with the form:

$$E_l\boldsymbol{u} = F_u,$$

where we denote with E_l the global elasticity matrix, $\boldsymbol{u}$ is the vector with the control variables for the displacements field and F_u is the corresponding right-hand side. In order to obtain this matrix, elemental elasticity matrices are computed and then assembled on the global one. Thus, the following integrals over each element Ω_e are computed:

$$A_e = \int_{\Omega_e} \begin{pmatrix} \partial_x R_e & 0 & \partial_y R_e \\ 0 & \partial_y R_e & \partial_x R_e \end{pmatrix} C \begin{pmatrix} \partial_x R_e & 0 \\ 0 & \partial_y R_e \\ \partial_y R_e & \partial_x R_e \end{pmatrix} d\Omega_e, \tag{2.54}$$

where R_e denotes the vector of B-spline/NURBS basis functions with support on the element Ω_e and the material matrix C with the elastic coefficients of the body given in terms of the Lamé parameters is

$$C = \begin{pmatrix} 2\mu+\lambda & \lambda & 0 \\ \lambda & 2\mu+\lambda & 0 \\ 0 & 0 & \mu \end{pmatrix}.$$

By using IGA discretizations, the error reduction in plane linear elastostatic problems shows the same behavior seen for the two dimensional Poisson equation. As a first example, let us consider problem (2.51) in the square domain $(0,1)^2$ with the following conditions:

$$\begin{aligned} \boldsymbol{\sigma}\cdot\boldsymbol{n} &= \boldsymbol{g}, \quad \text{on } \Gamma_n = (0,1)\times\{1\}, \\ \boldsymbol{u} &= \boldsymbol{0}, \quad \text{on } \Gamma_t = \Gamma \setminus \Gamma_n, \\ E &= 3\cdot 10^4, \quad \nu = 0.2, \end{aligned}$$

and $\boldsymbol{g}$ and the right-hand side $\boldsymbol{f}$ are taken such that the exact solution is

$$\boldsymbol{u} = \begin{pmatrix} \sin(\pi x)\sin(\pi y) \\ \sin(\pi x)\sin(\pi y) \end{pmatrix}. \tag{2.55}$$

For this problem, in Table 2.10 and Table 2.11, we show the errors in $\boldsymbol{L}^2$ and $\boldsymbol{H}^1$ norms respectively next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h. In the

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$7.9381e-05$	–	$1.3672e-06$	–	$5.9067e-08$	–
2^{-5}	$9.8893e-06$	8.0269	$8.3192e-08$	16.4347	$1.8454e-09$	32.0072
2^{-6}	$1.2364e-06$	7.9986	$5.1638e-09$	16.1106	$5.7766e-11$	31.9468
2^{-7}	$1.5463e-07$	7.9955	$3.2218e-10$	16.0277	$2.0765e-12$	27.8193

TABLE 2.10: 2D Linear Elasticity: Square domain.
Errors in $\boldsymbol{L}^2$ norm next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h.

h	$k=2$		$k=3$		$k=4$	
2^{-4}	0.0045	–	$1.3816e-04$	–	$4.0923e-06$	–
2^{-5}	0.0011	4.0165	$1.7139e-05$	8.0610	$2.5955e-07$	15.7668
2^{-6}	$2.8221e-04$	4.0042	$2.1382e-06$	8.0156	$1.6387e-08$	15.8392
2^{-7}	$7.0534e-05$	4.0010	$2.6715e-07$	8.0039	$1.0301e-09$	15.9080

TABLE 2.11: 2D Linear Elasticity: Square domain.
Errors in $\boldsymbol{H}^1$ norm next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h.

previous tables, we can observe that B-spline discretizations applied to linear elasticity problems hold $(k+1)-$th order of convergence in $\boldsymbol{L}^2$ norm and $k-$th order of convergence in $\boldsymbol{H}^1$ norm. In order to show the convergence rates with NURBS, let us consider the quarter of an annulus given in Figure 2.3 as computational domain with

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{u}_D, \quad \text{on } \Gamma_t = \Gamma \\ E &= 3 \cdot 10^4, \quad \nu = 0.2, \end{aligned}$$

and $\boldsymbol{f}$ is taken such that the exact solution is the one given in (2.55). Thus, in Table 2.12 and Table 2.13, we show the errors in $\boldsymbol{L}^2$ and $\boldsymbol{H}^1$ norms respectively next to the convergence rates using spline degrees $k = 2, 3, 4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h with the computational domain given in Figure 2.3. Hence, we can observe that NURBS discretizations provide the same convergence rates for elastostatic problems.

2.2.5 Stokes equations

The Stokes equations model the flow of an incompressible, viscous Newtonian fluid when the inertia forces are negligible compared to the viscous forces. The strong

h	$k=2$		$k=3$		$k=4$	
2^{-4}	$8.4425e-05$	–	$9.2368e-06$	–	$1.0359e-06$	–
2^{-5}	$9.8545e-06$	8.5671	$5.0746e-07$	18.2019	$2.6638e-08$	38.8875
2^{-6}	$1.2142e-06$	8.1162	$3.0644e-08$	16.5602	$7.8726e-10$	33.8361
2^{-7}	$1.5148e-07$	8.0157	$1.8985e-09$	16.1408	$2.4231e-11$	32.4903

TABLE 2.12: 2D Linear Elasticity: Quarter of an annulus.
Errors in $\boldsymbol{L}^2$ norm next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h.

h	$k=2$		$k=3$		$k=4$	
2^{-4}	0.0069	–	$6.1846e-04$	–	$6.4746e-05$	–
2^{-5}	0.0017	4.1259	$7.1803e-05$	8.6133	$3.5489e-06$	18.2436
2^{-6}	$4.1508e-04$	4.0318	$8.8081e-06$	8.1519	$2.1464e-07$	16.5344
2^{-7}	$1.0356e-04$	4.0082	$1.0958e-06$	8.0378	$1.3316e-08$	16.1184

TABLE 2.13: 2D Linear Elasticity: Quarter of an annulus.
Errors in $\boldsymbol{H}^1$ norm next to the convergence rates using spline degrees $k=2,3,4$, maximum global smoothness $\mathcal{C}^{k-1}$ and several values of the mesh size h.

formulation of Stokes equations with non-slip boundary conditions reads as follows:

$$\begin{cases} -\nabla\cdot(2\nu\boldsymbol{\varepsilon}(\mathbf{u}))+\nabla p=\mathbf{f}, & \text{in } \Omega,\\ \nabla\cdot\mathbf{u}=\mathbf{0}, & \text{in } \Omega,\\ \mathbf{u}=\mathbf{g}, & \text{on } \partial\Omega, \end{cases} \tag{2.56}$$

where we consider Ω to be an open bounded domain in $\mathbb{R}^2$, $\mathbf{u}$ is the velocity of the incompressible fluid, ν is the kinematic viscosity of the fluid, p is the pressure and $\mathbf{f}$ is the density of the external forces. In this system, we assume that the fluid is homogeneous (constant density) and the viscosity ν remains constant. Both equations are derived from conservation laws. Thus, the first of these equations is known as momentum equation whereas the second one is named mass continuity equation. For a detailed explanation on the theoretical aspects and applications of Stokes equations, we refer the reader to Thorpe, 1982; Galdi, 1994; Charru, 2011. In order to introduce the variational formulation of (2.56), we consider the following variational spaces $\mathcal{V}$ for the velocities and $\mathcal{Q}$ for the pressure:

$$\mathcal{V}=\boldsymbol{H}_0^1(\Omega)^2, \qquad \mathcal{Q}=L_0^2(\Omega), \tag{2.57}$$

where

$$L_0^2(\Omega)=\left\{q\in L^2(\Omega)\,\middle|\,\int_\Omega q=0\right\}.$$

Note that this choice of the pressure space $\mathcal{Q}$ is necessary in order to ensure the well-possedness of the variational formulation of the Stokes equations when there are not Dirichlet boundary conditions acting on p. In this case the pressure is determined up to a constant and thus we need to enforce the following condition for its uniqueness:

$$\int_\Omega p=0.$$

Thus, we proceed to test the momentum equation with trial functions $\mathbf{v} \in \mathcal{V}$ and the mass continuity equation with $q \in \mathcal{Q}$. Then, the mixed variational formulation is given by: Find $(\mathbf{u}, p) \in \mathcal{V} \times \mathcal{Q}$ such that

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = \int_\Omega \mathbf{f}\mathbf{v}\, d\Omega, & \forall \mathbf{v} \in \mathcal{V}, \\ b(\mathbf{u}, q) = 0, & \forall q \in \mathcal{Q}, \end{cases} \tag{2.58}$$

where the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ are defined as

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) = 2\nu \int_\Omega \boldsymbol{\varepsilon}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v})\, d\Omega, \\ b(\mathbf{v}, q) = \int_\Omega (\nabla \cdot \mathbf{v})\, q d\Omega. \end{cases} \tag{2.59}$$

Now, let us denote with $V_h \subset \mathcal{V}$ and $Q_h \subset \mathcal{Q}$ two finite dimensional discrete spaces. Thus, the discrete problem reads as follows: Find $(\mathbf{u}_h, p_h) \in V_h \times Q_h$ such that

$$\begin{cases} a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) = \int_\Omega \mathbf{f}\mathbf{v}_h\, d\Omega, & \forall \mathbf{v}_h \in V_h, \\ b(\mathbf{u}_h, q_h) = 0, & \forall q_h \in Q_h, \end{cases} \tag{2.60}$$

For this problem, the pair of spaces (V_h, Q_h) has to be chosen carefully in order to obtain a stable discretization. In fact, there is a stability condition that the discrete spaces must hold. This condition is known as inf-sup condition or LBB condition (see Brezzi and Fortin, 1991):

$$\inf_{q_h \in Q_h} \sup_{\mathbf{v}_h \in V_h} \frac{b(\mathbf{v}_h, p_h)}{\|p_h\|_{L^2(\Omega)} \|\mathbf{v}_h\|_{H^1_0(\Omega)}} \geq c, \tag{2.61}$$

where c is called the inf-sup constant. There are several inf-sup stable pairs of finite element spaces in the literature. Some of these pairs such as Raviart-Thomas (RT) elements (Raviart and Thomas, 1977), Nédélec (N) elements of the second family (Nedelec, 1986) and Taylor-Hood (TH) elements (Taylor and Hood, 1973; Brezzi and Falk, 1991) have been successfully extended to IGA in $2D$. The isogeometric generalization of these spaces defined on the parametric domain $\hat{\Omega} = (0,1)^2$ is given by

$$\hat{V}_h^{RT}(k,s) = S_{s+1,s}^{k+1,k}(\mathcal{G}_h) \times S_{s,s+1}^{k,k+1}(\mathcal{G}_h), \qquad \hat{Q}_h^{RT}(k,s) = S_{s,s}^{k,k}(\mathcal{G}_h), \tag{2.62}$$

$$\hat{V}_h^{N}(k,s) = S_{s+1,s}^{k+1,k+1}(\mathcal{G}_h) \times S_{s,s+1}^{k+1,k+1}(\mathcal{G}_h), \qquad \hat{Q}_h^{N}(k,s) = S_{s,s}^{k,k}(\mathcal{G}_h), \tag{2.63}$$

$$\hat{V}_h^{TH}(k,s) = S_{s,s}^{k+1,k+1} \times S_{s,s}^{k+1,k+1}(\mathcal{G}_h), \qquad \hat{Q}_h^{TH}(k,s) = S_{s,s}^{k,k}(\mathcal{G}_h), \tag{2.64}$$

where in each case $\mathcal{G}_h$ is a mesh determined by two univariate uniform open knot vectors with the corresponding spline degrees and global smoothnesses of the space and the knot intervals length is h. These pairs of isogeometric spaces were formally introduced in Buffa, Falco, and Sangalli, 2010. In that paper, the stability of these spaces was shown by means of inf-sup numerical tests. In addition, the stability of isogeometric Taylor-Hood elements was proved in Bressan and Sangalli, 2012 by using the macro-element technique. The definition of these spaces in the physical domain Ω takes into account the geometrical transformation $\mathbf{F}$. In case of discrete spaces for the pressure, their definition is given by

$$Q_h = \left\{ q_h = \hat{q}_h \circ \mathbf{F}^{-1} \,|\, \hat{q}_h \in \hat{Q}_h \right\}. \tag{2.65}$$

This construction is the same for the pressure spaces in (2.62),(2.63),(2.64). However, there are two different approaches for velocities since the component wise composition with $\mathbf{F}^{-1}$ is not divergence-preserving in the case of Raviart-Thomas and Nédélec spaces, as pointed out in Buffa, Falco, and Sangalli, 2010. Thus, whereas the definition of Taylor-Hood spaces on the physical domain Ω is given by

$$V_h^{TH}(k,s) = \left\{ v_h = \hat{v}_h \circ \mathbf{F}^{-1} \middle| \hat{v}_h \in \hat{V}_h^{TH}(k,s) \right\}, \tag{2.66}$$

Raviart-Thomas and Nédélec spaces for the velocities defined in Ω need of the so-called Piola mapping:

$$V_h^{RT}(k,s) = \left\{ v_h = \frac{J\mathbf{F}}{|J\mathbf{F}|} \hat{v}_h \circ \mathbf{F}^{-1} \middle| \hat{v}_h \in \hat{V}_h^{RT}(k,s) \right\}, \tag{2.67}$$

$$V_h^{N}(k,s) = \left\{ v_h = \frac{J\mathbf{F}}{|J\mathbf{F}|} \hat{v}_h \circ \mathbf{F}^{-1} \middle| \hat{v}_h \in \hat{V}_h^{N}(k,s) \right\}, \tag{2.68}$$

where $J\mathbf{F}$ denotes the jacobian matrix of the geometrical transformation $\mathbf{F}$. For our numerical experiments, we consider the isogeometric Taylor-Hood elements $Q2-Q1$ (globally continuous) and $Q3-Q2$ holding C^1 global smoothness. Taylor-Hood finite elements show the following convergence rates:

$$\|\mathbf{u}_h - \mathbf{u}\|_{\mathbf{H}^1} + \|p_h - p\|_{L^2} \leq Ch^k$$

Now, let us consider problem (2.56) with $\nu = \frac{1}{2}$ and exact solution

$$\mathbf{u}(x,y) = \begin{pmatrix} \sin(\pi x)\cos(\pi y) \\ -\cos(\pi x)\sin(\pi y) \end{pmatrix}, \qquad p(x,y) = \sin(\pi x)\sin(\pi y).$$

Thus, in Table 2.14 we show these errors $\|\mathbf{u}_h - \mathbf{u}\|_{\mathbf{H}^1} + \|p_h - p\|_{L^2}$ next to the corresponding convergence rates using the $Q2-Q1$ and $Q3-Q2$ (with global smoothness $\mathcal{C}^1$) Taylor-Hood elements for several values of the mesh size h. Hence, we can observe that the convergence rates for these isogeometric elements are also optimal.

h	$Q2-Q1$		$Q3-Q2$	
2^{-4}	0.0048	–	$1.5643e-04$	–
2^{-5}	0.0012	4.1025	$1.9936e-05$	7.8465
2^{-6}	$2.8632e-04$	4.0541	$2.5158e-06$	7.9244
2^{-7}	$7.1077e-05$	4.0284	$3.1595e-07$	7.9628

TABLE 2.14: 2D Stokes equations: Unit square domain.
Errors $\|\mathbf{u}_h - \mathbf{u}\|_{\mathbf{H}^1} + \|p_h - p\|_{L^2}$ next to the convergence rates using the $Q2-Q1$ and $Q3-Q2$ (with global smoothness $\mathcal{C}^1$) Taylor-Hood elements for several values of the mesh size h.

Remark: For domains with a non-trivial geometry, it is hard to find an exact solution with non-homogeneous essential boundary conditions. As it was pointed out in Cottrell, Hughes, and Bazilevs, 2005; Shojaee, Izadpenah, and Haeri, 2012, the imposition of this type of boundary conditions requires the use of some additional techniques in order to avoid any loss of accuracy. Thus, the convergence rates for NURBS are not shown here although both discretizations yield the same convergence order, see Cottrell, Hughes, and Bazilevs, 2009.

Chapter 3

Multigrid Methods

Multigrid methods are high efficient algorithms for the numerical solution of linear systems arised from discretizations of PDEs. The first studies on multigrid took place in the 60s with the aim of accelerating the convergence of classical relaxation schemes when they are applied to elliptic boundary value problems, see Fedorenko, 1962; Fedorenko, 1964; Bakhvalov, 1966. Then, the efficiency of multigrid methods was pointed out during the 70s by Brandt in Brandt, 1973; Brandt, 1977 and also multigrid convergence theory was developed, see Hackbusch, 1977; Hackbusch, 1980; Hackbusch, 1981. Hence, multigrid methods have become popular over the years and many multigrid guides, monographies and books have been devoted to them (Trottenberg, Oosterlee, and Schüller, 2001; Briggs, Henson, and McCormick, 1987; Wesseling, 2004). Furthermore, some international conferences such as the European Multigrid Conference (EMG) and the Copper Mountain Conference on Multigrid methods have manifested the interest of the research community on multigrid methods.

It is well known that multigrid techniques are among the fastest elliptic solvers with a computational cost of $\mathcal{O}(N\log N)$ operations, where N denotes the number of unknowns. Besides their efficiency, one of the main features of multigrid methods is their generality. For instance, their application is not restricted to the type of discretization, number of spatial dimensions of the problem, shape of computational domain, etc. In addition, multigrid methods are used for the numerical solutions of problems arising in many fields such as control theory, fluid dynamics, structural analysis, statistics, etc. Robustness is another desirable feature of multigrid methods, that is, instead of designing the method for an specific problem, its components can be fixed to solve efficiently different problems where some parameters are involved.

3.1 Basics of Multigrid

The goal of multigrid methods is to solve efficiently linear systems arising from partial differential equations. In order to work in a general framework, let us denote with L a differential operator such that our problem can be written as

$$Lu = f, \quad \text{in } \Omega, \tag{3.1}$$

where u is the exact solution, f is the corresponding righ-hand side and Ω is the computational domain. Then, the discretization of a PDE by means of a grid $\mathcal{G}_h$ with mesh size h leads to a discrete problem. For this problem, the continuous functions and differential operators become grid functions and discrete operators respectively:

$$L_h u_h = f_h, \quad \text{in } \Omega_h, \tag{3.2}$$

where $\Omega_h = \mathcal{G}_h(\Omega)$, u_h and f_h are grid functions on Ω_h and the discrete differential operator is defined as a linear mapping

$$L_h : \mathcal{S}(\Omega_h) \longrightarrow \mathcal{S}(\Omega_h), \tag{3.3}$$

with $\mathcal{S}(\Omega_h)$ the space of grid functions in Ω_h. Multigrid methods make use of a hierarchy of grids where the differential problem is defined. Depending on how the grid hierarchy is constructed, there are three different types of multigrid methods:

- $h-$multigrid methods: The hierarchy is constructed by means of $h-$refinement.
- $p-$multigrid methods: Degree elevation or degree reduction on the grid functions determines the hierarchy.
- $hp-$multigrid methods: Both $h-$ and $p-$refinements performed together on the grid functions provide the grid hierarchy.

We focus now on the introduction of standard $h-$multigrid. However, in Section 5.2 a two-level method based on an aggresive $p-$multigrid will be proposed. Also, the location of grid functions can be chosen in different ways. For instance, some finite element discretizations work with vertex-centered grids whereas FV can work with cell-centered grids.

As mentioned before, the discrete operators L_h are linear mappings acting on grid functions. The standard way to represent discrete operators is the use of stencils. Each stencil is related to a given grid function such that it shows the coefficients applied to grid functions with common support in the neighbourhood. Hence, any linear discrete operator L_h can be described as follows

$$L_h u_h(x,y) = \sum_{(i,j)\in\mathcal{I}} \ell_{i,j} u_h(x+ih, y+jh),$$

where $\mathcal{I}$ is a subset of $\mathbb{Z}^2$ such that $(0,0) \in \mathcal{I}$ and $(x+ih, y+jh) \in \mathcal{G}_h$, with $(i,j) \in \mathcal{I}$. In this way, the stencil notation of the corresponding discrete operator L_h is represented as follows:

$$L_h := [\ell_{i,j}] = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \iddots \\ \cdots & \ell_{-1,1} & \ell_{0,1} & \ell_{1,1} & \cdots \\ \cdots & \ell_{-1,0} & \ell_{0,0} & \ell_{1,0} & \cdots \\ \cdots & \ell_{-1,-1} & \ell_{0,-1} & \ell_{1,-1} & \cdots \\ \iddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{3.4}$$

For example, let us denote with $A^k_{s,h}$ the two-dimensional discrete Laplace operator using spline degree k, global smoothness $\mathcal{C}^s$ of the solution and length h of the knot spans on both directions. Thus, the corresponding stencil centered on an internal point (i,j) $(3 \leq i,j \leq N-2)$ by using the spline space $\mathcal{S}^{2,2}_{1,1}(\mathcal{G}_h)$ yields the equation

$$A^2_{1,h} u_h(x,y) = f(x,y),$$

where

$$A^2_{1,h}u_h(x,y) = \begin{pmatrix} -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ -\frac{1}{12} & \frac{1}{30} & \frac{11}{10} & \frac{1}{30} & -\frac{1}{12} \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \end{pmatrix} u_h(x,y)$$
$$= \sum_{i,j=-2}^{2} A^2_{1,h}(i+3, j+3) \cdot u_h(x+jh, y+ih).$$

Naturally, the stencils of discrete operators centered on boundary points or close to them change but we cover the general case for internal points. Once we have set a grid and the corresponding discrete operator, the problem is stated at the so-called fine grid.

The next step on the understanding of multigrid is to introduce its main principles. Multigrid methods are based on the combination of smoothing error effect of relaxation schemes and the so-called coarse grid correction. These two principles are described as follows:

- **Smoothing principle**: Iterations of classical relaxation schemes cause a smoothing effect on the error.
- **Coarse grid correction**: A smooth error is well-approximated in a coarse grid where less computations have to be made.

In order to combine these principles, more than one grid has to be considered. Hence, the simplest way to explain the multigrid procedure is to introduce the two-grid cycle. As a starting point, let us consider model problem (3.2) in a fine grid $\mathcal{G}_h$ and an initial approximation u_h^m for the $m-$th iteration of this cycle. First, ν_1 iterations of a relaxation scheme with iteration matrix S_h are applied such that we obtain:

$$\bar{u}_h^m = S_h^{\nu_1} u_h^m.$$

In this context, S_h is called smoother operator and ν_1 is the number of pre-smoothing steps. If we denote the corresponding error by

$$e_h^m = u_h - \bar{u}_h^m,$$

the smoothing principle claims that e_h^m is smooth and then we can achieve a good approximation of it in a coarse grid $\mathcal{G}_H$. In order to work with the coarse grid correction, we introduce the residual or defect

$$d_h^m = f_h - L_h \bar{u}_h^m. \tag{3.5}$$

Now, the substraction of $L_h\bar{u}_h^m$ to both sides of (3.2) yields the defect equation (or residual equation)

$$L_h e_h^m = d_h^m. \tag{3.6}$$

Since the error e_h^m can be well approximated in a coarse grid, the two-grid strategy consists to transfer the defect d_h^m to $\mathcal{G}_H$ by means of a restriction operator R_h^H:

$$d_H^m = R_h^H d_h^m. \tag{3.7}$$

Thus, we solve the defect equation on Ω_H and the correction is obtained by solving a smaller linear system than the original one:

$$L_H e_H^m = d_H^m. \tag{3.8}$$

However, the correction has to be transfered to the fine grid in order to update the approximation in the fine grid. This transfer is carried out by a prolongation operator P_H^h:

$$e_h^m = P_H^h e_H^m. \tag{3.9}$$

The next step is to add the correction to the smooth approximation

$$\hat{u}_h^m = \bar{u}_h^m + e_h^m. \tag{3.10}$$

Finally, additional ν_2 iterations of the relaxation scheme (also called post-smoothing steps) can be performed and the next approximation is obtained:

$$u_h^{m+1} = S_h^{\nu_2} \hat{u}_h^m. \tag{3.11}$$

To sum up, in Figure 3.1 we show the scheme of the two-grid cycle procedure. Each iteration of the two-grid cycle multiplies the error is by the following error propagation matrix:

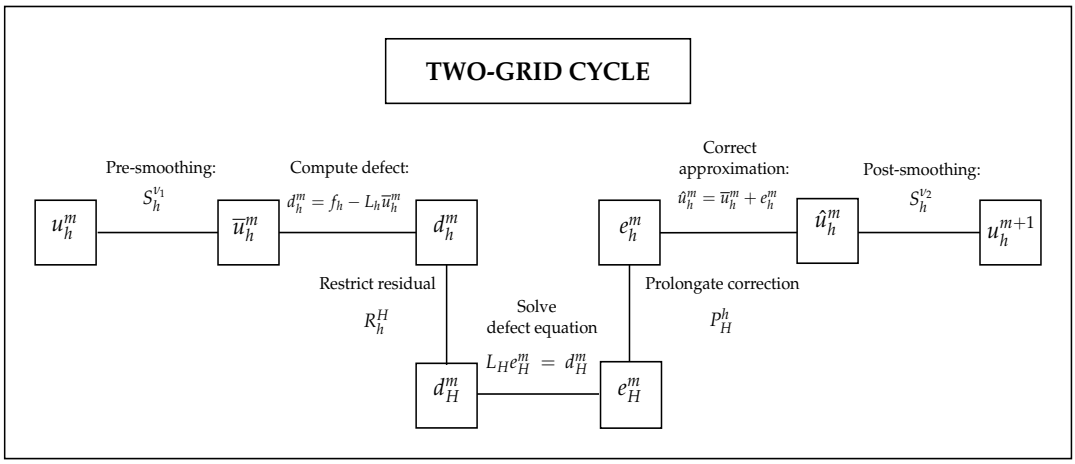


FIGURE 3.1: Two-grid cycle.
Scheme of two-grid cycle procedure.

$$M_h^{2g} := S_h^{\nu_2} \left(I_h - P_H^h L_H^{-1} R_h^H L_h \right) S_h^{\nu_1}. \tag{3.12}$$

However, two-grid cycles are not practical since solving the defect equation exactly at the coarse grid can be still expensive from the computational cost point of view. The idea of multigrid arises from a nested iteration of two-grid cycle: The convergence of a two-grid cycle is not deteriorated if we get a suitable approximation to the defect equation on Ω_H. For this approximation, we can introduce as many grids as necessary such that the coarsest grid implies to solve a linear system with

negligible computational cost. Furthermore, the solution at the coarsest grid can be approximated even by applying an iterative scheme.

The performance of multigrid methods depends strongly on the choice of their components. Hence, one component of multigrid methods is the coarsening strategy. In some cases, it might be possible to consider an aggresive coarsening if the convergence rates do not deteriorate. However, we consider as our general approach the standard coarsening $H = 2h$, that is, the grid size is doubled in each direction. Following this strategy, in Figure 3.2 we show the hierarchy of $l = 4$ grids obtained with finesh grid size $h = \frac{1}{16}$:

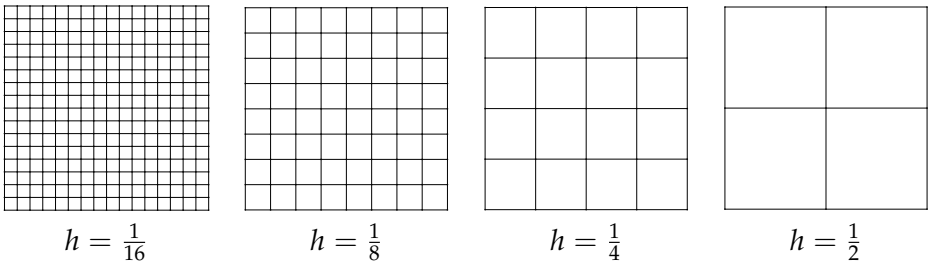


FIGURE 3.2: Grid hierarchy.
Hierarchy of $l = 4$ grids with finesh grid size $h = \frac{1}{16}$ using standard coarsening.

At this point, smoothers, transfer operators and the number of smoothing steps are clearly crucial. In addition, the number of times that the correction has to be updated at each level/grid can be changed. This number γ is referred as cycle index and has also an important role on convergence speed and computational cost. The two most common types of cycles are $V-$cycles ($\gamma = 1$) and $W-$cycles ($\gamma = 2$). In order to illustrate the effect of γ, in Figure 3.3, we show the structure of a $V-$cycle and a $W-$cycle by considering $l = 4$ grids. Moreover, in Algorithm 2 we introduce the general procedure of a multigrid cycle with cycle index γ with l levels or grids. For this purpose, we substitute index h with index j in order to denote the corresponding $j-$th grid.

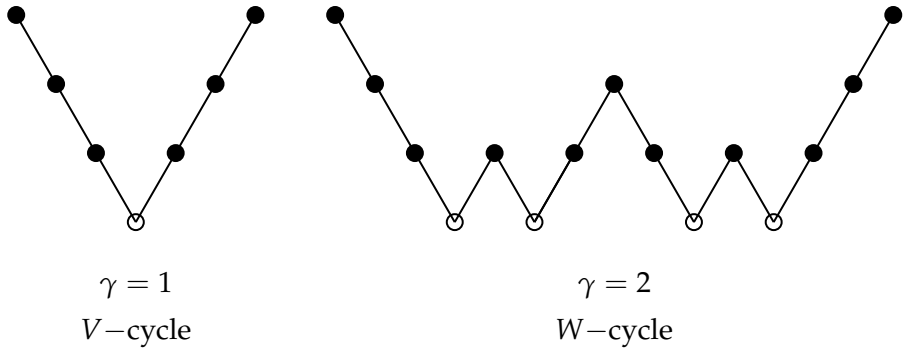


FIGURE 3.3: Types of cycle.
Structure of a $V-$cycle ($\gamma = 1$) and a $W-$cycle ($\gamma = 2$) with $l = 4$ grids.

In order to study the convergence of multigrid, it is necessary to introduce the error propagation matrix, see Trottenberg, Oosterlee, and Schüller, 2001:

$$M_j = S_j^{\nu_2}\left(I_j - P_{j-1}^{j}\left(I_{j-1} - M_{j-1}^{\gamma}\right)L_{j-1}^{-1}R_j^{j-1}L_j\right)S_j^{\nu_1}, \qquad \text{for } j = 1,\ldots,l, \tag{3.13}$$

Algorithm 2 Multigrid.

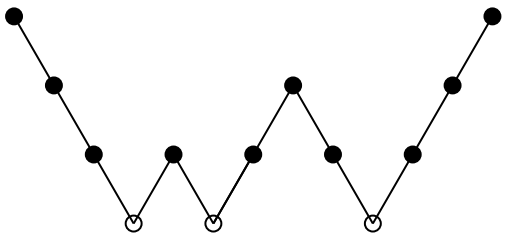
$u_j^{m+1} = \mathrm{MG}(L_j, R_j^{j-1}, P_{j-1}^j, u_j^m, f_j, \nu_1, \nu_2, \gamma, j)$

1: **if** j = l **then**
2: Solve the defect equation: $L_j e_j^m = d_j^m$.
3: **return** e_j^m
4: **end**
5: Pre-smoothing: $\bar{u}_j^m = S_j^{\nu_1} u_j^m$.
6: Computation of the defect: $d_j^m = f_j - L_j \bar{u}_j^m$.
7: Restrict the residual: $d_{j-1}^m = R_j^{j-1} d_j^m$.
8: Approximate the correction by nested iteration:

$$e_{j-1}^m = \mathrm{MG}(L_{j-1}, R_{j-1}^{j-2}, P_{j-2}^{j-1}, d_{j-1}^m, 0, \nu_1, \nu_2, \gamma, j-1).$$

9: Prolongate the correction: $e_j^m = P_{j-1}^j e_{j-1}^m$.
10: Update approximation: $\hat{u}_j^m = \bar{u}_j^m + e_j^m$.
11: Post-smoothing: $u_j^{m+1} = S_j^{\nu_2} \hat{u}_j^m$.

where I_j denotes the identity operator in the $j-$th grid and $M_0 = 0$. Since the error is multiplied by this iteration matrix, the asymptotic convergence factor is determined by the spectral radius of the previous matrix M_j and it allows us to differentiate the performances of V-cycles and W-cycles by means of rigurous analysis or local Fourier analysis (LFA). However, the cycle index γ can also depend on j giving place to new types of cycles. One of these new cycles is the so-called F-cycle, shown in Figure 3.4. The computational cost of F-cycles is in between of that of V-cycles and W-cycles but in general they provide convergence rates very similar to those from $W-$cycles. The idea of F-cycles is to perform the smoothing and restriction process until the coarsest grid. Then, during the prolongation process a $V-$cycle is performed before the correction is interpolated to a finer grid for the first time.



$F-$cycle

FIGURE 3.4: F-cycle
Structure of an $F-$cycle with $l = 4$ grids.

Transfer operators are constructed taking into account that $h-$refinement of the approximation spaces yields also a hierarchy of approximation spaces with a proper subset relation. That is, every coarse grid function ψ_i^H, $i = 1, \ldots, d_1$ can be expanded exactly as a linear combination of fine grid functions φ_j^h, $j = 1, \ldots, d_2$. Hence, there exist some coefficients $c_{i,j} \in \mathbb{R}$, with $i = 1, \ldots, d_1$ and $j = 1, \ldots, d_2$ such that

$$\psi_i^H(x) = \sum_{j=1}^{d_2} c_{i,j}\varphi_j^h(x), \quad \forall x \in \mathcal{G}_H, i = 1, \dots, d_1. \tag{3.14}$$

In order to compute the coefficients of the previous linear combination, one can take some collocation points where the basis functions are evaluated and solve a linear system per basis function ψ_i^H of the coarse space. The standard approach is to consider d_2 equally spaced collocation points x_c, $c = 1, \dots, d_2$. Hence, one has to solve the following linear systems:

$$\begin{pmatrix} \varphi_1^h(x_1) & \varphi_2^h(x_1) & \cdots & \varphi_{d_2}^h(x_1) \\ \varphi_1^h(x_2) & \varphi_2^h(x_2) & \cdots & \varphi_{d_2}^h(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1^h(x_{d_2}) & \varphi_2^h(x_{d_2}) & \cdots & \varphi_{d_2}^h(x_{d_2}) \end{pmatrix} \begin{pmatrix} c_{i,1} \\ c_{i,2} \\ \vdots \\ c_{i,d_2} \end{pmatrix} = \begin{pmatrix} \psi_i^H(x_1) \\ \psi_i^H(x_2) \\ \vdots \\ \psi_i^H(x_{d_2}) \end{pmatrix}, \quad \text{for } i = 1, \dots, d_1. \tag{3.15}$$

Thus, the coefficients obtained with the solution of the previous $i-$th linear system provide the entries of the $i-$th row of the prolongation operator P_H^h. Then, the restriction can be chosen as the transpose of the prolongation operator. In the bivariate case, we can make a good use of the tensor product structure of isogeometric discretizations and take the bivariate transfer operators as the kronecker product of two univariate transfer operators. Furthermore, in the case of NURBS discretizations we must multiply B-spline transfer operators by the inverse of a diagonal matrix with the corresponding weights given in (1.45).

3.2 Classical iterative methods

Classical iterative methods solve linear systems by means of a numerical scheme. However, our main interest lies on their smoothing properties as pointed out in Section 3.1. Let us consider the following linear system:

$$Au = b. \tag{3.16}$$

Iterative methods are based on a splitting of matrix $A = M - N$ such that M is easily invertible. With this splitting, the new approximation for (3.16) at the $m+1$ iteration is computed by

$$Mu^{m+1} = Nu^m + b. \tag{3.17}$$

It is interesting to rewrite this iterative scheme as

$$u^{m+1} = Su^m + M^{-1}b, \tag{3.18}$$

where $S = M^{-1}N$ is the iteration matrix of the corresponding method. In addition, the previous matrix is also the error propagation matrix when the iterative method is considered as a solver. Let us denote with $e^m = u - u^m$ the error in the $m-$th iteration. Then, we have

$$e^{m+1} := u - u^{m+1} = Su + M^{-1}b - \left(Su^m + M^{-1}b\right) = S(u - u^m) = Se^m. \tag{3.19}$$

Instead of computing the next approximation u^{m+1} directly, a practical approach is to obtain the correction $\delta u := u^{m+1} - u^m$ and then get $u^{m+1} = u^m + \delta u$. For this purpose, equation (3.17) is rewritten is terms of the correction as

$$M\delta u = r^m, \tag{3.20}$$

where $r^m = b - Au^m$ is the residual at the $m-$th iteration. This technique based on the correction δu is known as iterative refinement (see Higham, 1997) and is essential for the understanding of multiplicative Schwarz methods introduced in Section 3.3.1. In order to improve the efficiency of the iterative scheme in (3.18), we can add a relaxation parameter ω such that the new approximation is given by

$$u^{m+1} = \omega u^* + (1-\omega)u^m. \tag{3.21}$$

where

$$u^* = Su^m + M^{-1}b. \tag{3.22}$$

Thus, the damped relaxation scheme given in (3.21)-(3.22) can be written as

$$u^{m+1} = S_\omega u^m + \omega M^{-1}b, \qquad S_\omega = \omega S + (1-\omega)I. \tag{3.23}$$

Moreover, the corresponding relaxation scheme for correction δu by adding the damped parameter ω is given by

$$M\delta u = \omega r^m. \tag{3.24}$$

At this point, let us introduce some of the most classical iterative methods. These methods consider the following matrix splitting:

$$A = D - L - U, \tag{3.25}$$

where D is the diagonal of A, $-L$ is the lower strict part and $-U$ is the upper strict part of A. For instance, the choice of $M = D$ and $N = L + U$ in (3.25) yields Jacobi iterative method:

$$\textbf{Jacobi:} \quad Du^{m+1} = (L+U)u^m + b. \tag{3.26}$$

If we isolate the new approximation, then one gets

$$u^{m+1} = S_J u^m + D^{-1}b, \tag{3.27}$$

with iteration matrix

$$S_J = D^{-1}(L+U). \tag{3.28}$$

Furthermore, the efficiency of Jacobi scheme can be improved with the introduction of a damping coefficient ω. Thus, we obtain the damped Jacobi relaxation scheme by applying (3.23):

$$u^{m+1} = S_{J_\omega} u^m + \omega D^{-1}b, \tag{3.29}$$

where the iteration matrix S_{J_ω} is given by

$$S_{J_\omega} = \omega D^{-1}(L+U) + (1-\omega)I. \tag{3.30}$$

The main advantage of damped Jacobi relaxation scheme is that the next approximation of every unknown does not require updated values of the rest of unknowns. Hence, this scheme is fully parallelizable and all the unknowns can be updated simuntaneously. However, it has been proved that this iterative scheme has a poor performance as smoother.

If we recall matrix splitting (3.25) and take $M = D - L$, $N = U$, then we obtain the Gauss-Seidel relaxation scheme:

$$\textbf{Gauss-Seidel:} \quad (D - L)\, u^{m+1} = U u^m + b, \tag{3.31}$$

that can be rewritten as

$$u^{m+1} = S_G u^m + (D - L)^{-1} b, \tag{3.32}$$

where the Gauss-Seidel iteration matrix is given by

$$S_G = (D - L)^{-1} U. \tag{3.33}$$

As done with Jacobi, a damping parameter ω can be considered for Gauss-Seidel relaxation scheme:

$$u^{m+1} = S_{G_\omega} u^m + \omega M^{-1} b, \tag{3.34}$$

where the iteration matrix S_{G_ω} is given by

$$S_{G_\omega} = \omega (D - L)^{-1} U + (1 - \omega) I. \tag{3.35}$$

When the damping parameter $\omega > 1$, the previous scheme is known as successive over-relaxation Gauss-Seidel (SOR). On the other hand, the case $\omega < 1$ is called under-relaxed Gauss-Seidel. Due to the matrix splitting, Gauss-Seidel iterative method takes into account updated values of other grid points for the next approximation of each unknown. Furthermore, this dependence on updated values makes Gauss-Seidel not fully parallelizable and the order in which the unknowns are updated becomes important. In fact, the performance of Gauss-Seidel depends strongly on the grid points ordering. The standard grid points ordering is called forward or lexicographic ordering and it gives place to lexicographic Gauss-Seidel scheme. In this approach, a grid point (i_1, j_1) is updated before another grid point (i_2, j_2) if $i_1 < i_2$ or $i_1 = i_2$ and $j_1 < j_2$.

On the other hand, an odd-even ordering of the grid points can be considered. The odd-even or red-black ordering yields the red-black Gauss-Seidel (RB-GS) iterative method. With this grid points ordering, a grid point (i, j) is said to be odd/even if $i + j$ is odd/even. The odd-even or red-black ordering of grid-points is illustrated in Figure 3.5. RB-GS updates first the odd points and then the even points sequentially. In general, RB-GS outperforms lexicographic Gauss-Seidel since it yields better convergence rates and both have the same computational cost.

The splitting of matrix $A = M - N$ can be understood as a splitting of the discrete operator into $L_h = L_h^+ + L_h^-$, where L_h^- denotes the discrete operator with the coefficients of unknowns that have been relaxed before and L_h^+ corresponds to the discrete operator with coefficients of unknowns that are going to be relaxed. For this purpose, it is really useful to describe the operators by means of the stencil notation given in (3.4). Then, the relaxation scheme given in terms of stencils is

$$L_h^+ u_h^{m+1} = -L_h^- u_h^m + f_h, \tag{3.36}$$

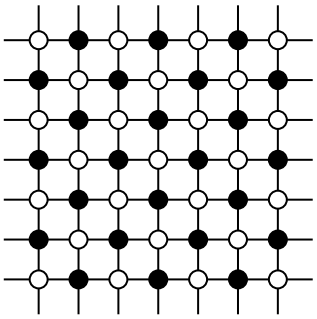


FIGURE 3.5: Red-black ordering of grid-points

and it is described for each unknown separately. For instance, let us give an example of how to compute the new approximation of each unknown by using the Jacobi relaxation scheme. If we consider the isogeometric discretization of the Poisson equation in a unit square by means of $S^{2,2}_{1,1}(\mathcal{G}_h)$, then the discrete operator is split into the two following stencils:

$$L_h^+ = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{11}{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad L_h^- = \begin{pmatrix} -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ -\frac{1}{12} & \frac{1}{30} & 0 & \frac{1}{30} & -\frac{1}{12} \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \end{pmatrix}.$$

Thus, the new approximation applying Jacobi relaxation scheme is given by

$$u_h^{m+1}(x,y) = \frac{10}{11}\left(f_h(x,y) - \sum_{i=1}^{5}\sum_{j=1}^{5} L_h^-(i,j)\cdot u_h^m(x+(j-3)h, y+(i-3)h)\right).$$

Moreover, the damped Jacobi relaxation scheme for this case yields:

$$\begin{aligned} u_h^{m+1}(x,y) =& \omega\frac{10}{11}\left(f_h(x,y) - \sum_{i,j=-2}^{2} L_h^-(i+3,j+3)\cdot u_h^m(x+jh, y+ih)\right) \\ &+ (1-\omega)u_h^m(x,y). \end{aligned}$$

In addition, the lexicographic Gauss-Seidel splitting of the same discrete operator in terms of stencils yields:

$$L_h^+ = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{12} & \frac{1}{30} & \frac{11}{10} & 0 & 0 \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \end{pmatrix},$$

$$L_h^- = \begin{pmatrix} -\frac{1}{360} & -\frac{7}{180} & -\frac{1}{12} & -\frac{7}{180} & -\frac{1}{360} \\ -\frac{7}{180} & -\frac{13}{90} & \frac{1}{30} & -\frac{13}{90} & -\frac{7}{180} \\ 0 & 0 & 0 & \frac{1}{30} & -\frac{1}{12} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

3.3 Block-wise smoothers

In the previous Section 3.2, some iterative schemes were proposed as point-wise smoothers. Within a multigrid framework, a natural extension of point-wise smoothers are patch-wise smoothers. In order to apply such a relaxation, the computational domain is divided into small (overlapping or non-overlapping) patches, and then, one smoothing step consists of solving local problems on each patch one-by-one either in a Jacobi-type or Gauss-Seidel-type manner. This results in an additive or multiplicative Schwarz-type smoother, respectively. One of the best-known multigrid methods based on this type of relaxations was proposed by Vanka in Vanka, 1986 for solving the steady state incompressible Navier-Stokes equations in primitive variables, discretized by a finite volume scheme on a staggered grid. The computational domain is divided into cells with pressure nodes at the cell centers and velocity nodes at the cell faces. The smoothing procedure is a so-called symmetric coupled Gauss-Seidel technique (SCGS), which consists of solving local problems for each cell involving all the unknowns located at the cell. This is done cell by cell in a Gauss-Seidel-type manner and, therefore, can be viewed as a multiplicative Schwarz-type iteration.

The previous classical relaxation schemes update the new approximation at each grid point separately by solving its corresponding equation. On the other hand, box-type smoothers are based on a splitting of the grid into blocks of unknowns that can be constructed in many ways. In fact, one can consider any possible block size and overlapping among the blocks of unknowns. For the one-dimensional case, one possibility is to split the grid into symmetric blocks of odd size, centered at each grid point u_i and considering maximum overlapping. For instance a block of three points centered at u_i is given by $B = (u_{i-1}, u_i, u_{i+1})$ and the corresponding blocks of five and seven points are $B = (u_{i-2}, u_{i-1}, u_i, u_{i+1}, u_{i+2})$ and $B = (u_{i-3}, u_{i-2}, u_{i-1}, u_i, u_{i+1}, u_{i+2}, u_{i+3})$. In Figure 3.6, we show the splittings of a one-dimensional grid into: (a) blocks of three points, (b) blocks of five points and (c) blocks of seven points, all of them with maximum overlapping among the blocks.

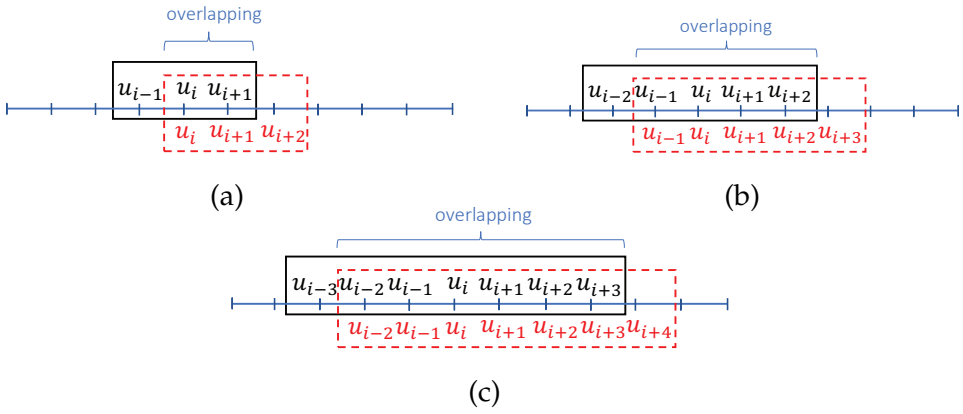


FIGURE 3.6: Splittings of a one-dimensional grid into: (a) blocks of three points, (b) blocks of five points and (c) blocks of seven points, all of them with maximum overlapping among the blocks.

The two-dimensional version of these splittings originate square blocks of unknowns around each unknown $u_{i,j}$. For instance, the square block of 9 (3×3) points centered at $u_{i,j}$, illustrated in Figure 3.7, is given by

$$B = \{u_{i-1,j-1}, u_{i,j-1}, u_{i+1,j-1}, u_{i-1,j}, u_{i,j}, u_{i+1,j}, u_{i-1,j+1}, u_{i,j+1}, u_{i+1,j+1}\}.$$

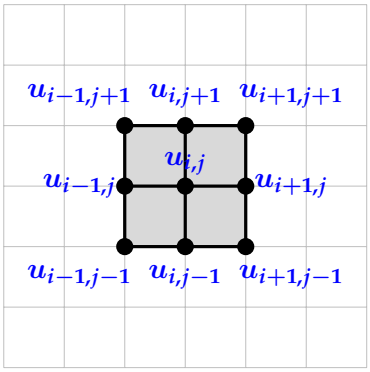


FIGURE 3.7: Square block of 9 (3×3) points centered at $u_{i,j}$.

Analogously, one can introduce the square blocks of 25 (5×5) and 49 (7×7) points. Once the blocks of unknowns are set, the equations corresponding to the grid points involved in each block are solved together. Let u_B be the vector with unknowns involved in block B, δu_B the local corrections and d_B the local defect. Then, every local system of (3.16) is given by

$$A^B \delta u_B = d_B, \tag{3.37}$$

where A^B is the local matrix corresponding to block B. The construction of local matrices A^B is carried out by means of a projection operator V_B from the vector of unkowns u to the vector of unknowns involved in the block B:

$$A^B = V_B A V_B^T. \tag{3.38}$$

Thus, one iteration of these smoothers consists of a loop over all the blocks such that the corresponding local systems (3.37) are solved. Finally, it is important to combine the local corrections δu_B appropriately. For a correct combination of the corrections, the overlapping among the blocks has to be taken into account. In addition, corrections can be combined via two different ways: Multiplicative Schwarz methods or additive Schwarz methods.

3.3.1 Multiplicative Schwarz methods

Multiplicative Schwarz methods are derived from the alternating Schwarz method introduced by H. A. Schwarz in Schwarz, 1870 as a domain decomposition algorithm for the strong formulation of the Poisson equation. Then, the discretization of the alternating Schwarz method gave rise to the computational tools that are known as multiplicative Schwarz methods, for more details see Gander, 2008. Moreover, these methods have received many attention as domain decomposition methods and smoothers in the literature, see Chan and Mathew, 1994; Smith, Bjørstad, and Gropp, 1996; Smith, 1997; Kinnewig, Roth, and Wick, 2021; Kanschat and Mao, 2016. In a multigrid framework, multiplicative Schwarz methods are block-wise smoothers that always make use of the most recent updates of u. That is, for every local system (3.38) the defect is obtained after a global update of u with the previous local corrections. Since we need to recompute the defect constantly, the local systems have to be solved sequentially. Hence, we show the algorithm of a multiplicative Schwarz method in Algorithm 3. In this algorithm, A is the global matrix with size $n \times n$, s_B is the size of all the blocks, N_B denotes the number of blocks and M_B is a matrix whose $i-th$ row contains the indices of the unknowns involved in block B_i.

Algorithm 3 Multiplicative Schwarz method

$u^{m+1} =$ Multiplicative$(A, u_j^m, b, M_B, N_B, s_B)$

for $i = 1 : N_B$ **do** ▷ Loop through blocks.
  $B_i \leftarrow B(i,:)$ ▷ Indices of the unknowns involved in block B_i.
  $V_i = 0_{s_B \times n}$
  $V_i(:, B_i) = I_{s_B}$ ▷ Construction of projection operator V_i.
  $A^{B_i} \leftarrow V_i A V_i^T$ ▷ Construction of local matrix.
  $r_{B_i} \leftarrow V_i(b - Au^m)$ ▷ Construction of local defect.
  $A^{B_i} \delta u_{B_i} = r_{B_i}$ ▷ Solve the local system.
  $u^{m+1} = u^m + V_i^T \delta u_{B_i}$ ▷ Local correction of u.
end

This structure suggests that multiplicative Schwarz methods can be understood as block Gauss-Seidel relaxation schemes. In addition, the global iteration matrix of the multiplicative Schwarz method is given by the product of iteration matrices corresponding to the local corrections of each block:

$$S_M = \prod_{i=1}^{N_B} \left(I - V_{B_i}^T (A^{B_i})^{-1} V_{B_i} A \right). \tag{3.39}$$

As we mentioned for Gauss-Seidel, the ordering in which the blocks are visited matters, and it can improve the efficiency of the method. Furthermore, we can consider different ordering strategies given the block structure of Schwarz methods. Instead of a lexicographic ordering, we can select several sets of grid points and update the blocks of each set sequentially. These sets of grid points are commonly known as colors. Thus, a colored multiplicative Schwarz method makes a loop over the colors. Then for each color, the method proceeds sequentially with the blocks centered at the grid points of the current color. In order to illustrate a multicolored grid, in Figure 3.8 we show a grid with 9 (3×3) colours numbered from 1 to 9.

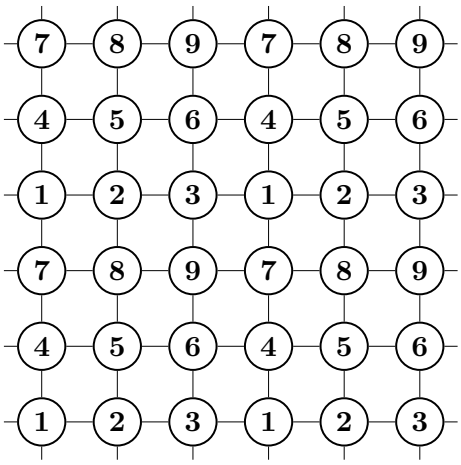


FIGURE 3.8: Grid with 9 (3×3) colours numbered from 1 to 9.

Furthermore, in Figure 3.9 we show for the coloured version of the 9-point (3×3) multiplicative Schwarz method the blocks of unknowns that are updated together with (a) the first color and (b) the fifth color.

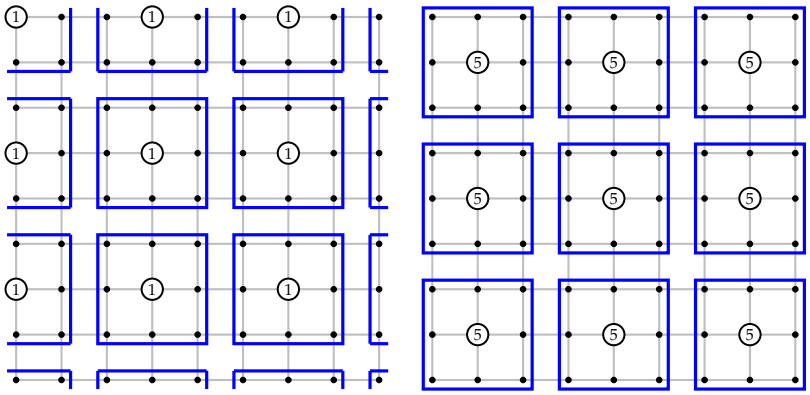


FIGURE 3.9: Coloured version of the 9-point (3×3) multiplicative Schwarz method: Blocks of unknowns that are updated together with (a) the first color and (b) the fifth color.

3.3.2 Additive Schwarz methods

Additive Schwarz methods arise from the discretization of the parallel Schwarz method introduced in Lions, 1988, but their name was firstly used in Dryja and Widlund, 1989. These methods have been studied in Schoeberl and Zulehner, 2003 as smoothers in a multigrid method for saddle point problems. It is shown that, under suitable conditions, the iteration can be interpreted as a symmetric inexact Uzawa method. Restrictive additive Schwarz methods (RAS) were introduced in Cai and Sarkis, 1999 as an efficient alternative to the classical additive Schwarz preconditioners. Convergence of RAS methods was proven in Frommer and Szyld, 2001. It was shown that this method reduces communication time while maintaining the most desirable properties of the classical Schwarz methods. RAS preconditioners are widely used in practice and are implemented in several software packages. In Saberi, Meschke, and Vogel, 2022, a restrictive Schwarz method was proposed as smoother for solving the Stokes equations. It was observed that this smoother achieves comparable convergence rates to the multiplicative version, while being computationally less expensive per iteration.

Instead of updating the defect as soon as possible, additive Schwarz methods solve all the local systems by using the same global defect and then the approximation is corrected at once by addition of all the local corrections. Since there is no need of intermediate updates, all the local systems can be solved in parallel. Hence, additive Schwarz methods can be understood as block Jacobi smoothers. Thus, the ordering in which the blocks of unknowns are visited does not affect the convergence rates of these methods. However, the efficiency of additive Schwarz methods can be improved by means of a weights vector ω during the correction process. Taking this into account, in Algorithm 4 we show the pseudo-code of one iteration of an additive Schwarz method.

Moreover, the iteration matrix of additive Schwarz methods is determined by the addition of all the local corrections as follows:

$$S_A = I - \left(\sum_{i=1}^{N_B} \omega V_{B_i}^T \left(A^{B_i} \right)^{-1} V_{B_i} \right) A. \tag{3.40}$$

Algorithm 4 Additive Schwarz method

$u^{m+1} = \text{Additive}(A, u_j^m, b, M_B, N_B, s_B, \omega)$

$r \leftarrow (b - Au^m)$
for $i = 1 : N_B$ **do** ▷ Loop through blocks.
  $B_i \leftarrow B(i,:)$ ▷ Indices of the unknowns involved in block B_i.
  $V_i = 0_{s_B \times n}$
  $V_i(:, B_i) = I_{s_B}$ ▷ Construction of projection operator V_i.
  $A^{B_i} \leftarrow V_i A V_i^T$ ▷ Construction of local matrix.
  $r_{B_i} \leftarrow V_i r$ ▷ Construction of local defect.
  $A^{B_i} \delta u_{B_i} = r_{B_i}$ ▷ Solve the local system.
end
$u^{m+1} = u^m + \sum_{i=1}^{N_B} V_i^T \omega \delta u_{B_i}$ ▷ Global correction of u.

The choice of ω depends strongly on the overlapping among the blocks given that the unknowns are updated several times. Furthermore, one might consider different values for the weights in ω such that some grid points are relaxed differently. This is specially useful if the discretization provides more than one stencil or there is an intermediate overlapping. Thus, the smoothing effect of additive Schwarz methods is highly improved.

Algorithm 5 Restrictive additive Schwarz method

$u^{m+1} = \text{Additive}(A, u_j^m, b, M_B, N_B, s_B, \omega, ov)$

$r \leftarrow (b - Au^m)$
for $i = 1 : N_B$ **do** ▷ Loop through blocks.
  $B_i \leftarrow B(i,:)$ ▷ Indices of the unknowns involved in block B_i.
  $V_i = 0_{s_B \times n}$
  $\tilde{V}_i = 0_{s_B \times n}$ ▷ Auxiliar projection operator $\tilde{V}_i$.
  $V_i(:, B_i) = I_{s_B}$ ▷ Construction of projection operator V_i.
  $\tilde{V}_i(:, B_i(1 : s_B - ov)) = I_{s_B - ov}$ ▷ ov is the overlapping among the blocks.
  $P_i = \tilde{V}_i'$ ▷ Construction of restrictive prolongation operator P_i.
  $A^{B_i} \leftarrow V_i A V_i^T$ ▷ Construction of local matrix.
  $r_{B_i} \leftarrow V_i r$ ▷ Construction of local defect.
  $A^{B_i} \delta u_{B_i} = r_{B_i}$ ▷ Solve the local system.
end
$u^{m+1} = u^m + \sum_{i=1}^{N_B} P_i \omega \delta u_{B_i}$ ▷ Global correction of u.

Although overlapping additive Schwarz methods are parallelizable, the overlapping among blocks requires communication between processors during the smoothing process. In order to avoid it, we can keep solving all the previous local systems but omit multiple local corrections of the same grid point. There are two possibles ways to achieve this effect: To modify the weights associated to each block or to construct a different prolongation operator P_B instead of using directly V_B^T. The first approach requires to change to zero the weights multiplying local corrections such that every point is updated only once. On the other hand, one can take P_B as done for the non-restrictive case but making zero the columns whose contribution overlap a previous correction. Using this approach, in Algorithm 5 we show the pseudo-code of one iteration of such an additive Schwarz method. However, both approaches are known as restrictive additive Schwarz methods.

Chapter 4

Local Fourier Analysis

4.1 Introduction

Local Fourier analysis is a very useful tool for the quantitative analysis and design of multigrid algorithms. This analysis was firstly introduced by Achi Brandt in Brandt, 1977 and then developed in Brandt, 1981; Brandt and Livne, 2011; Brandt, 1994. In those papers, Brandt proposed that the operations involved in multigrid algorithms can be analyzed as local processes by Fourier expansion of the error function. However, it is necessary to make some assumptions for the analysis. In fact, the discrete operator is assumed to be linear with constant coefficients. This assumption does not affect the generality of LFA since any general discrete operator can be locally linearized and local constant coefficients can be considered. Hence, LFA can be applied to a wide range of different problems. In addition, the analysis deals with the problem in an infinite grid

$$\mathcal{G}_h = \{x = ch \big| c \in \mathbb{Z}^d, h = (h_1, \ldots, h_d)^T\},$$

and the effect of boundary conditions is neglected. Moreover, LFA is based on a Fourier decomposition of the error in eigenvectors of the discrete operator. Formally, the error is considered as a grid function that can be expanded as a linear combination of complex exponential functions. This expansion comes from the Fourier transform of any given $u \in L^2(\mathbb{R}^d)$ defined as follows

$$\hat{u}(\theta) := \frac{1}{2\pi} \int_{R^d} e^{-\imath\theta\cdot x/h} u(x) dx, \tag{4.1}$$

where the frequency $\theta = (\theta_1, \ldots, \theta_d)$. Nevertheless, it is well known that the function u can be pulled back by using the Fourier inversion formula:

$$u(x) := \frac{1}{2\pi} \int_{R^d} e^{\imath\theta\cdot x/h} \hat{u}(\theta) d\theta. \tag{4.2}$$

Thus, the contribution of every $e^{\imath\theta\cdot x/h}$ in $u(x)$ is quantified as its corresponding $\hat{u}(\theta)$. This interpretation suggests that any grid function can be expanded as a linear combination of the so-called Fourier modes:

$$\varphi_h(\theta, x) = e^{\imath\theta\cdot x/h} = e^{\imath\theta_1 x_1/h_1} \cdots e^{\imath\theta_d x_d/h_d}. \tag{4.3}$$

Indeed, the error function is assumed to be a linear combination of these Fourier modes. Thus, LFA is also known as local mode analysis. Note that the Fourier modes are 2π-periodic in the frequency domain

$$\varphi_h(\theta, x) = \varphi_h(\theta', x) \Leftrightarrow \begin{cases} \theta_1 = \theta_1 \,(\text{mod } 2\pi)\,, \\ \vdots \\ \theta_d = \theta_d \,(\text{mod } 2\pi)\,. \end{cases} \tag{4.4}$$

Thus, the Fourier modes with $\theta \in \Theta_h := (-\pi, \pi]^d$ define a basis on $\mathcal{G}_h$ and they span the so-called Fourier space:

$$\mathcal{F}(\mathcal{G}_h) := span\{\varphi_h(\theta, x) \big| \theta \in \Theta_h\}. \tag{4.5}$$

Moreover, LFA is able to predict the asymptotic convergence behavior of multigrid methods by studying the effect of each component of the algorithm on the Fourier modes. The first operator involved in any multigrid method is the discrete operator. By taking into account the assumptions mentioned previously, the Fourier modes are eigenfunctions of discrete operators:

$$L_h \varphi_h(\theta, x) = \widetilde{L}_h(\theta) \varphi_h(\theta, x), \tag{4.6}$$

where the formal eigenvalue $\widetilde{L}_h(\theta)$ is called the Fourier symbol of L_h. In order to illustrate Fourier symbols, let us show the stencil notation of the one-dimensional quadratic IGA discrete Laplace operator with global smoothness $\mathcal{C}^1$:

$$L_h = \frac{1}{h}\left(-\frac{1}{6}, -\frac{1}{3}, 1, -\frac{1}{3}, -\frac{1}{6}\right). \tag{4.7}$$

Thus, its Fourier symbol is given by

$$\widetilde{L}_h(\theta) = \frac{1}{h}\left(1 - \frac{2\cos(\theta)}{3} - \frac{2\cos(2\theta)}{6}\right) = \frac{2}{3h}(2 - \cos\theta(1 + \cos\theta)).$$

The study of the Fourier symbol of discrete operators gives us some insight about the difficulties to annihilate the high-frequency components of the error during the smoothing process. In fact, it was pointed out in Donatelli et al., 2017 that the appearance of small eigenvalues associated with high frequencies foretell the need of an intense smoothing procedure to achieve a good convergence rate. In Figure 4.1, we show the Fourier symbol of the IGA discrete operator for the one-dimensional Poisson problem (2.5), with $k = 2, 5, 8$ and holding maximum global smoothness $\mathcal{C}^{k-1}$.

We can observe in Figure 4.1 that the values of the eigenvalues associated with high frequencies get reduced as the polynomial degree is increased. Thus, the same smoother will deteriorate its performance for higher spline degrees and the search of a suitable smoother for each k is not straightforward. For this task, we utilize the power of LFA to design efficient and robust multigrid methods for IGA discretizations with respect to the polynomial degree k.

We organize the remaining part of this chapter as follows: In Section 4.2, we introduce the standard analysis of LFA by presenting the smoothing analysis 4.2.1, the two-grid analysis 4.2.2) and three-grid analysis 4.2.3. In Section 4.3, we show the LFA for block-wise smoothers and a recent LFA based on infinite subgrids is presented in Section 4.4. Finally, we introduce the analysis for systems of PDEs in Section 4.5.

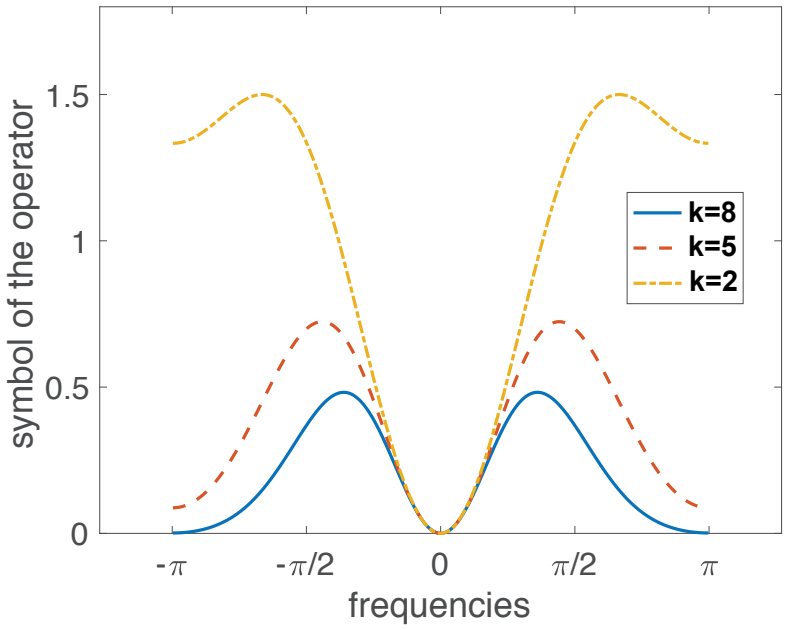


FIGURE 4.1: Fourier symbol of the IGA discrete operator for the one-dimensional Poisson problem (2.5), with $k = 2, 5, 8$ and holding maximum global smoothness $\mathcal{C}^{k-1}$.

4.2 Standard analysis: h-multigrid methods

The standard local Fourier analysis is based on the Fourier modes presented in 4.1. In addition, this analysis is often introduced for multigrid methods whose hierarchy is based on an $h-$refinement of the mesh. It is also common to introduce the analysis with a standard coarsening strategy, that is, the grid size h is doubled $H = 2h$ on the following coarse grid of the hierarchy. Furthermore, we will consider the same grid size on each spatial direction for the sake of simplicity. As pointed out before, the goal of LFA is to predict the asymptotic convergence behavior of multigrid methods. This is achieved by studying the effect of each operator on the Fourier modes and considering that the operations are local processes. The idea within LFA is to simplify the computation of the spectral radius of the iteration matrices and the use of local Fourier modes allows to do this task accurately. Hence, the common approaches of standard LFA are the smoothing analysis (or one-grid analysis), two-grid analysis and three-grid analysis. Although $k-$grid analysis can also be considered, those analysis use to provide enough information about the asymptotic convergence behavior of multigrid methods. Now, we proceed to introduce them one by one.

4.2.1 Smoothing analysis

Given the importance of smoothers for the performance of multigrid methods, a special analysis is devoted to this operator in LFA. As it happened with discrete operators, Fourier modes are eigenfunctions of the smoother:

$$S_h \varphi_h(\theta, x) = \widetilde{S}_h(\theta) \varphi_h(\theta, x), \tag{4.8}$$

where $\widetilde{S}_h(\theta)$ denotes the Fourier symbol of the smoother operator S_h. For instance, let us compute the Fourier symbol of Gauss-Seidel for a B-spline discretization of the $1D$ Poisson equation with $k = 2$ and $\mathcal{C}^1$ global smoothness. Then, the corresponding discrete operator L_h given in 4.7 is split into

$$L_h^+ = \left(0, 0, 1, -\frac{1}{3}, -\frac{1}{6}\right), \qquad L_h^- = \left(-\frac{1}{6}, -\frac{1}{3}, 0, 0, 0\right),$$

and therefore the Fourier symbol of Gauss-Seidel is

$$\widetilde{S}(\theta) = \left(\widetilde{L}_h^+(\theta)\right)^{-1} \widetilde{L}_h^-(\theta) = \frac{-\frac{e^{-\imath\theta}}{3} - \frac{e^{-2\imath\theta}}{6}}{1 - \frac{e^{\imath\theta}}{3} - \frac{e^{2\imath\theta}}{6}}.$$

As pointed out in Chapter 3, the purpose of the smoother is to annihilate the high-frequency components of the error. With a standard coarsening $H = 2h$, the set of low frequencies is usually denoted by Θ_{2h} and it is given by

$$\Theta_{2h} := (-\frac{\pi}{2}, \frac{\pi}{2}]^d. \tag{4.9}$$

On the other hand, the set of high frequencies is given by $\Theta_h \setminus \Theta_{2h}$. Accordingly to this, the Fourier modes are called low/high-frequency components when the associated θ is a low/high frequency. At this point, the smoothing factor of a smoother S_h is defined as follows:

$$\mu = \sup_{\theta \in \Theta_h \setminus \Theta_{2h}} |\widetilde{S}_h(\theta)|. \tag{4.10}$$

Hence, the smoothing factor measures how much the high components of the error are reduced after one iteration of the smoother. In Figure 4.2, we show the Fourier symbol of Gauss-Seidel smoother for the $1D$ Poisson equation (2.5), discretized with B-splines of degree $k = 2, 5, 8$ and holding maximum global smoothness.

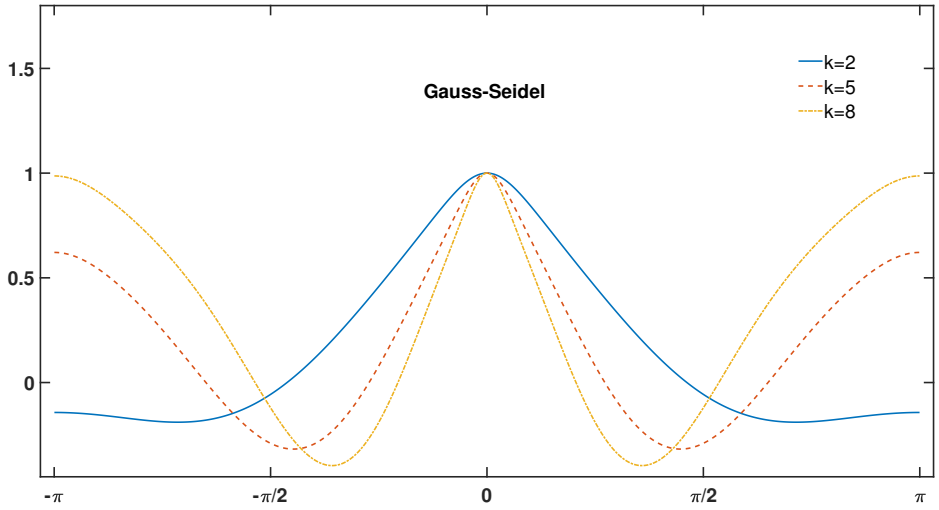


FIGURE 4.2: Fourier symbol of Gauss-Seidel smoother for the $1D$ Poisson equation (2.5), discretized with B-splines of degree $k = 2, 5, 8$ and holding maximum global smoothness.

As we can see, the eigenvalues of Gauss-Seidel smoother for these discretizations associated to high frequencies get closer to 1 when the spline degree is increased. Moreover, the same situation can be observed in the two-dimensional case. In fact, we show in Figure 4.3 the Fourier symbol of Gauss-Seidel smoother for the 2D Poisson equation discretized with B-splines holding maximum global smoothness and spline degrees (a) $k = 2$ and (b) $k = 5$. We conclude with the smoothing analysis that the performance of point-wise smoothers such as Gauss-Seidel deteriorates for high-order IGA discretizations.

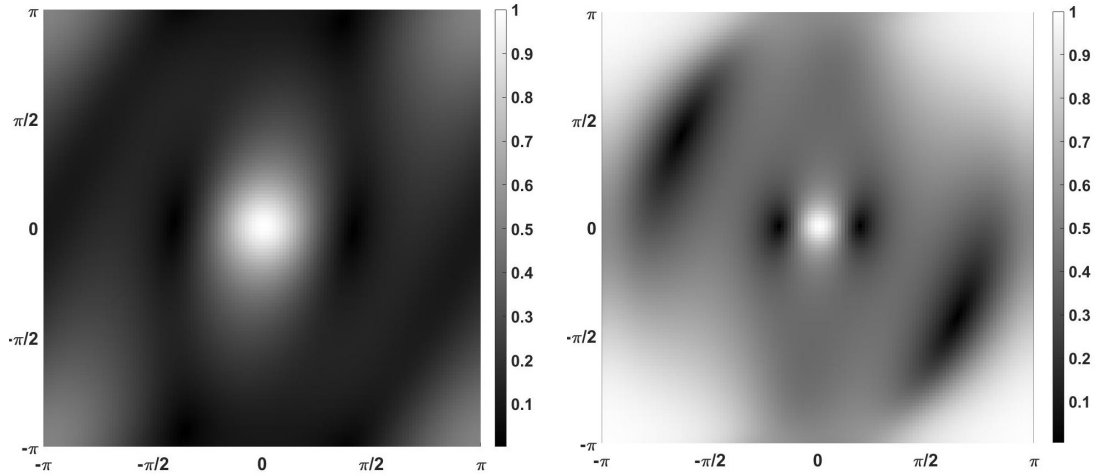


FIGURE 4.3: Fourier symbol of Gauss-Seidel smoother for the 2D Poisson equation discretized with B-splines holding maximum global smoothness and spline degrees (a) $k = 2$ and (b) $k = 5$.

4.2.2 Two-grid analysis

In order to analyze the interplay between the smoother and the coarse-grid correction, it is necessary at least a two-grid analysis that takes into account the coarse-grid principle for the acceleration of multigrid convergence. The aim of this analysis is to compute the spectral radius of the two-grid operator. Hence, the error propagation matrix of a two-grid method with a standard coarsening is given by

$$M_h^{2h} = S_h^{\nu_2} \left(I_h - P_{2h}^h (L_{2h})^{-1} R_h^{2h} L_h \right) S_h^{\nu_1}, \tag{4.11}$$

where L_h and L_{2h} represent the discrete operators in the fine and coarse grids. Moreover, the Fourier symbol of this two-grid operator is

$$\widetilde{M}_h^{2h} = \widetilde{S}_h^{\nu_2} \left(\widetilde{I}_h - \widetilde{P}_{2h}^h (\widetilde{L}_{2h})^{-1} \widetilde{R}_h^{2h} \widetilde{L}_h \right) \widetilde{S}_h^{\nu_1}. \tag{4.12}$$

We have already introduced the Fourier symbol of the discrete operator and smoother. However, the Fourier symbol of the inter-grid transfer operators I_h^{2h} and I_{2h}^h requires a special attention since they produce an aliasing effect on the frequencies θ: In the transition between fine and coarse grids, each low frequency $\theta_0 \in \Theta_{2h}$ is coupled with $2^d - 1$ high frequencies. In other words, those high frequencies are not visible in the coarse grid and the high-frequency components are coupled. In the one-dimensional case each low frequency θ_0 is coupled with one high frequency

$\theta^1 = \theta^0 - \text{sign}(\theta^0)\pi$ and the components associated to these frequencies span the so-called spaces of $2h-$harmonics:

$$\mathcal{F}^2(\theta^0) = \text{span}\left\{\varphi_h(\theta^0,\cdot), \varphi_h(\theta^1,\cdot)\right\}. \tag{4.13}$$

One can observe that

$$\varphi_h(\theta^0, x) = \varphi_h(\theta^1, x) = \varphi_{2h}(2\theta^0, x), \quad x \in \mathcal{G}_{2h}.$$

Then, we obtain as a direct consequence that the spaces of $2h-$harmonics are invariant under the restriction and prolongation operators and therefore the two-grid operator as well:

$$\begin{array}{rccc} R_h^{2h}: & \mathcal{F}^2(\theta^0) & \longrightarrow & \text{span}\left\{\varphi_{2h}(2\theta^0,\cdot)\right\}, \\ P_{2h}^h: & \text{span}\left\{\varphi_{2h}(2\theta^0,\cdot)\right\} & \longrightarrow & \mathcal{F}^2(\theta^0). \end{array}$$

Hence, we denote the matrix representation of the restriction with respect the $2h-$harmonics with $(R_h^{2h})^{2g} = (\widetilde{R}_h^{2h}(\theta^0), \widetilde{R}_h^{2h}(\theta^1)) \in \mathbb{C}^{1\times 2}$ whereas for the prolongation we have $(P_{2h}^h)^{2g} = (\widetilde{P}_{2h}^h(\theta^0), \widetilde{P}_{2h}^h(\theta^1))' \in \mathbb{C}^{2\times 1}$. As an example, let us compute the Fourier symbol of restriction operator for quadratic B-spline basis functions with global smoothness $\mathcal{C}^1$. Note that

$$\begin{aligned} R_h^{2h}\varphi_h(\theta,x) &= \frac{1}{4}\varphi_{2h}(\theta, x-\frac{3}{2}h) + \frac{3}{4}\varphi_{2h}(\theta, x-\frac{1}{2}h) + \frac{3}{4}\varphi_{2h}(\theta, x+\frac{1}{2}h) + \frac{1}{4}\varphi_{2h}(\theta, x+\frac{3}{2}h) \\ &= \left(\frac{1}{4}e^{-\frac{3}{2}\imath\theta} + \frac{3}{4}e^{-\frac{1}{2}\imath\theta} + \frac{3}{4}e^{\frac{1}{2}\imath\theta} + \frac{1}{4}e^{\frac{3}{2}\imath\theta}\right)\varphi_{2h}(2\theta, x). \end{aligned}$$

Then, the Fourier symbol $\widetilde{R}_h^{2h}(\theta)$ is given by

$$\widetilde{R}_h^{2h}(\theta) = \frac{1}{4}e^{-\frac{3}{2}\imath\theta} + \frac{3}{4}e^{-\frac{1}{2}\imath\theta} + \frac{3}{4}e^{\frac{1}{2}\imath\theta} + \frac{1}{4}e^{\frac{3}{2}\imath\theta}.$$

In addition, the prolongation operator is taken as its transpose operator and we have that

$$\begin{aligned} P_{2h}^h\varphi_{2h}(2\theta,x) &= \frac{1}{4}\varphi_h(2\theta, x-3h) + \frac{3}{4}\varphi_h(2\theta, x-h) + \frac{3}{4}\varphi_h(2\theta, x+h) + \frac{1}{4}\varphi_h(2\theta, x+3h) \\ &= \left(\frac{1}{4}e^{-3\imath\theta} + \frac{3}{4}e^{-\imath\theta} + \frac{3}{4}e^{\imath\theta} + \frac{1}{4}e^{3\imath\theta}\right)\varphi_h(\theta, x). \end{aligned}$$

Therefore, the Fourier symbol $\widetilde{P}_{2h}^h(\theta)$ is

$$\widetilde{P}_{2h}^h(\theta) = \frac{1}{4}e^{-3\imath\theta} + \frac{3}{4}e^{-\imath\theta} + \frac{3}{4}e^{\imath\theta} + \frac{1}{4}e^{3\imath\theta}.$$

Given that the rest of operators do not couple frequencies, it is clear that the rest of operators hold

$$\begin{array}{rccc} S_h: & \mathcal{F}^2(\theta^0) & \longrightarrow & \mathcal{F}^2(\theta^0), \\ L_h: & \mathcal{F}^2(\theta^0) & \longrightarrow & \mathcal{F}^2(\theta^0), \\ L_{2h}: & \text{span}\left\{\varphi_{2h}(2\theta^0,\cdot)\right\} & \longrightarrow & \text{span}\left\{\varphi_{2h}(2\theta^0,\cdot)\right\}. \end{array}$$

Note that the matrix representations of S_h and L_h are (2×2) diagonal matrices whose diagonal elements are their corresponding Fourier symbols associated to the frequencies $\{\theta^0, \theta^1\}$. Hence, it results that

$$M_h^{2h} : \mathcal{F}^2(\theta^0) \longrightarrow \mathcal{F}^2(\theta^0).$$

Then, the matrix representation of the two-grid operator $(M_h^{2h})^{2g}$ with respect to $\mathcal{F}^2(\theta^0)$ is in this case a 2×2-matrix. This matrix is obtained by means of

$$(M_h^{2h})^{2g} = (S_h^{2g})^{\nu_2} \left(I_h^{2g} - (P_{2h}^h)^{2g} (L_{2h}^{2g})^{-1} (R_h^{2h})^{2g} L_h^{2g} \right) (S_h^{2g})^{\nu_1}. \tag{4.14}$$

In addition, given the linear independence of the Fourier modes the spaces of $2h-$harmonics form a decomposition of the Fourier space, it holds that:

$$\mathcal{F}(\mathcal{G}_h) = \bigoplus_{\theta^0 \in \Theta_{2h}} \mathcal{F}^2(\theta^0).$$

At this point, we conclude that the spectral radius of M_h^{2h} can be computed by calculating the spectral radius of the smaller matrices $(M_h^{2h})^{2g}$ as follows

$$\rho_{2g} = \rho(M_h^{2h}) = \sup_{\theta^0 \in \widetilde{\Theta}_{2h}} \rho((M_h^{2h})^{2g}(\theta^0)), \tag{4.15}$$

where the frequency space $\widetilde{\Theta}_{2h}$ is given by

$$\widetilde{\Theta}_{2h} = \Theta_{2h} \setminus \left\{ \theta^0 \in \Theta_{2h} \middle| \widetilde{L}_{2h}(2\theta^0) = 0 \text{ or } \widetilde{L}_h(\theta^0) = 0 \right\}. \tag{4.16}$$

Note that it is necessary to introduce the previous frequency space since the error propagation matrix includes the inverse of L_{2h}. On the other hand, in the two-dimensional case we denote with $\theta^{00} = (\theta_1^{00}, \theta_2^{00}) \in \Theta_{2h} = (-\pi/2, \pi/2]^2$ any low frequency that is coupled with the three following frequencies:

$$\begin{aligned} \theta^{11} &:= \theta^{00} - (\text{sign}(\theta_1^{00})\pi, \text{sign}(\theta_2^{00})\pi), \\ \theta^{10} &:= \theta^{00} - (\text{sign}(\theta_1^{00})\pi, 0), \\ \theta^{01} &:= \theta^{00} - (0, \text{sign}(\theta_2^{00})\pi). \end{aligned}$$

In order to illustrate this, in Figure 4.4 we show a low frequency θ^{00} in the low-frequency domain with its coupling three high frequencies on the high-frequency region (gray).

Therefore, the $2h-$harmonics space for the $d = 2$ case is given by the following four-dimensional space

$$\mathcal{E}^4(\theta^{00}) = \text{span} \left\{ \varphi_h(\theta^{00}, \cdot), \varphi_h(\theta^{10}, \cdot), \varphi_h(\theta^{01}, \cdot), \varphi_h(\theta^{11}, \cdot) \right\}. \tag{4.17}$$

Using these $2h-$harmonics, we obtain that

$$\mathcal{E}(\mathcal{G}_h) = \bigoplus_{\theta^{00} \in \Theta_{2h}} \mathcal{E}^4(\theta^{00}), \tag{4.18}$$

where we denote with $\mathcal{E}(\mathcal{G}_h)$ the Fourier space on an infinite grid $\mathcal{G}_h$ with two spatial directions. In this case, the matrix representation of the two-grid operator $(M_h^{2h})^{2g}$ is a (4×4)-matrix. In fact, this matrix is obtained by means of the matrix representation of (4.14) but using the corresponding operators with respect to subspaces of $2h-$harmonics for $d = 2$:

$$S_h^{2g}(\theta^{00}) = diag \left\{ \widetilde{S}_h^{2g}(\theta^{00}), \widetilde{S}_h^{2g}(\theta^{10}), \widetilde{S}_h^{2g}(\theta^{01}), \widetilde{S}_h^{2g}(\theta^{11}) \right\} \in \mathbb{C}^{4\times 4}, \tag{4.19}$$

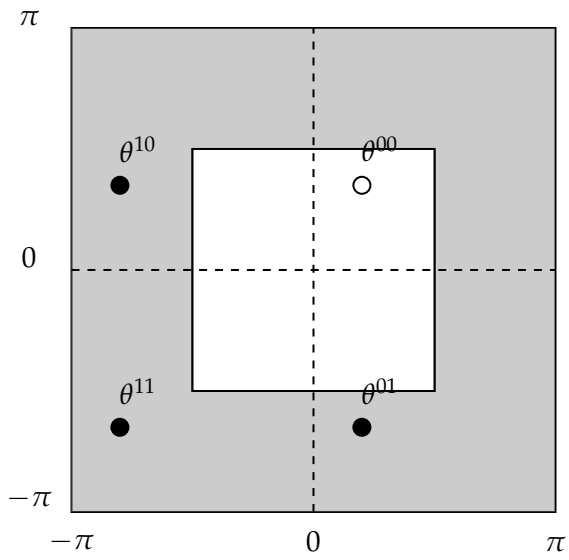


FIGURE 4.4: Low frequency θ^{00} in the low-frequency domain with its coupling three high frequencies on the high-frequency region (gray)

$$L_h^{2g}(\theta^{00}) = diag\left\{\widetilde{L}_h^{2g}(\theta^{00}), \widetilde{L}_h^{2g}(\theta^{10}), \widetilde{L}_h^{2g}(\theta^{01}), \widetilde{L}_h^{2g}(\theta^{11})\right\} \in \mathbb{C}^{4\times 4}, \tag{4.20}$$

$$(R_h^{2h})^{2g}(\theta^{00}) = \left((\widetilde{R}_h^{2h})^{2g}(\theta^{00}), (\widetilde{R}_h^{2h})^{2g}(\theta^{10}), (\widetilde{R}_h^{2h})^{2g}(\theta^{01}), (\widetilde{R}_h^{2h})^{2g}(\theta^{11})\right) \in \mathbb{C}^{1\times 4}, \tag{4.21}$$

$$(P_{2h}^{h})^{2g}(\theta^{00}) = \left((\widetilde{P}_{2h}^{h})^{2g}(\theta^{00}), (\widetilde{P}_{2h}^{h})^{2g}(\theta^{10}), (\widetilde{P}_{2h}^{h})^{2g}(\theta^{01}), (\widetilde{P}_{2h}^{h})^{2g}(\theta^{11})\right)' \in \mathbb{C}^{4\times 1}. \tag{4.22}$$

Hence, the spectral radius of M_h^{2h} in the two-dimensional case is computed as follows

$$\rho_{2g} = \rho(M_h^{2h}) = \sup_{\theta^{00}\in\widetilde{\Theta}_{2h}} \rho((M_h^{2h})^{2g}(\theta^{00})), \tag{4.23}$$

where $\widetilde{\Theta}_{2h}$ is given by

$$\widetilde{\Theta}_{2h} = \Theta_{2h} \setminus \left\{\theta^{00} \in \Theta_{2h} \middle| \widetilde{L}_{2h}(2\theta^{00}) = 0 \text{ or } \widetilde{L}_h(\theta^{00}) = 0\right\}. \tag{4.24}$$

4.2.3 Three-grid analysis

Although the two-grid analysis is capable to capture the interplay between smoother and coarse-grid correction, that analysis does not distinguish between $V-$cycles and $W-$cycles. In order to notice the effect of the index cycle γ, at least a three-grid analysis is required. In addition, this analysis allows us to study the difference between the choice of pre- and post-smoothing steps. Considering a standard coarsening, the mesh sizes are taken as $h, 2h$ and $4h$ for the corresponding grids. Following this strategy, the goal of three-grid analysis is to compute the spectral radius of the following error propagation matrix.

$$M_h^{4h} = S_h^{\nu_2}(I_h - P_{2h}^h(I_{2h} - (M_{2h}^{4h})^\gamma)L_{2h}^{-1}R_h^{2h}L_h)S_h^{\nu_1}, \tag{4.25}$$

where M_{2h}^{4h} is the two-grid operator between the two coarse grids

$$M_{2h}^{4h} = S_{2h}^{\nu_2}(I_{2h} - P_{4h}^{2h}L_{4h}^{-1}R_{2h}^{4h}L_{2h})S_{2h}^{\nu_1}. \tag{4.26}$$

As it happened with the two-grid analysis, the three-grid analysis requires to take into account the aliasing effect of the inter-grid transfer operators on Fourier modes. However, this coupling takes place twice: In the transition from the finest to the second grid and in the transition from the second to the coarsest grid. Thus, in the one-dimensional case the low frequency space is given by

$$\Theta_{4h} = (-\frac{\pi}{4}, \frac{\pi}{4}].$$

Hence, in the transition from the fine grid to the coarsest grid any low frequency $\theta_0^0 \in \Theta_{4h}$ is coupled with the following three frequencies

$$\begin{aligned}
\theta_0^1 &:= \theta^0 - \operatorname{sign}(\theta^0)\pi/2, \\
\theta_1^0 &:= \theta^0 - \operatorname{sign}(\theta^0)\pi \\
\theta_1^1 &:= \theta^0 - \operatorname{sign}(\theta^0)\pi/2 + \operatorname{sign}(\theta^0)\pi.
\end{aligned}$$

In addition, an usual way to introduce these four frequencies is by using the following formula:

$$\theta_\beta^\alpha = \theta^0 - \alpha \operatorname{sign}(\theta^0)\pi/2 + (-1)^{\alpha+\beta}\beta \operatorname{sign}(\theta^0)\pi, \quad \alpha, \beta \in \{0, 1\}. \tag{4.27}$$

More specifically, in the transition from the finest grid to the second grid the low frequency $\theta_0^0 \in \Theta_{4h} \subset \Theta_{2h}$ is coupled with the high frequency $\theta_1^0 \in \Theta \setminus \Theta_{2h}$ whereas $\theta_0^1 \in \Theta_{2h}$ is coupled with $\theta_1^1 \in \Theta \setminus \Theta_{2h}$. Then, $\theta_0^0 \in \Theta_{4h}$ is coupled with the high frequency $\theta_0^1 \in \Theta_{2h} \setminus \Theta_{4h}$ in the transition from the second grid to the coarsest grid. Thus, in Figure 4.5 we show the scheme on how frequencies θ_β^α are coupled in the transition from fine grid to second grid and in the transition from second grid to third grid.

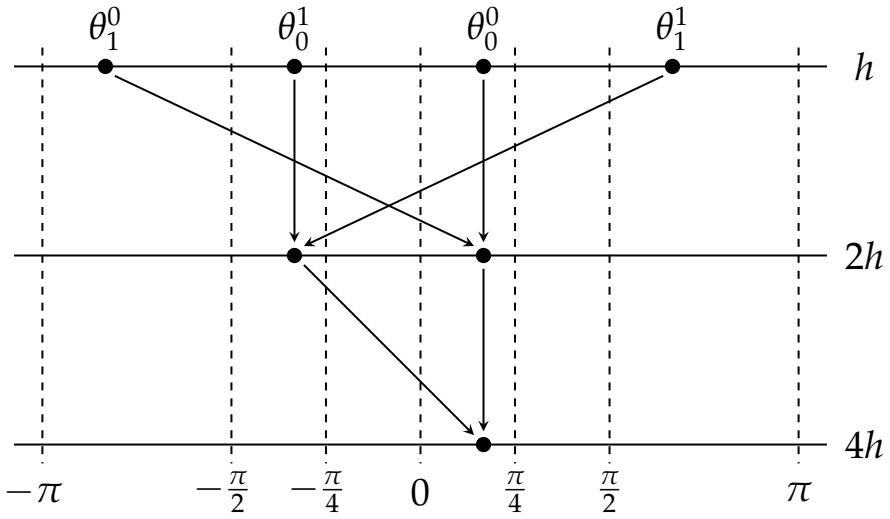


FIGURE 4.5: Scheme on how frequencies θ_β^α are coupled in the transition from fine grid to second grid and in the transition from second grid to third grid.

These four frequencies θ_β^α define the so-called spaces of $4h-$harmonics as follows:

$$\mathcal{F}^4(\theta_0^0) = \operatorname{span}\left\{\varphi_h(\theta_\beta^\alpha, \cdot) \,|\, \alpha,\, \beta \in \{0, 1\}\right\}, \tag{4.28}$$

with $\theta_0^0 \in \Theta_{4h}$. In addition, the Fourier space can be decomposed as a direct sum of spaces of $4h-$harmonics:

$$\mathcal{F}(\mathcal{G}_h) = \bigoplus_{\theta^0 \in \Theta_{4h}} \mathcal{F}^4(\theta_0^0).$$

This decomposition allows us to compute the spectral radius of the three-grid error propagation matrix by studying smaller $(4 \times 4)-$matrices. In order to understand the formation of this process, let us analyze each one carefully. As we mentioned, discrete operators and smoothers do not couple frequencies. In fact, they leave invariant the spaces of $4h-$harmonics

$$\begin{array}{rccc} L_h: & \mathcal{F}^4(\theta_0^0) & \longrightarrow & \mathcal{F}^4(\theta_0^0), \\ S_h: & \mathcal{F}^4(\theta_0^0) & \longrightarrow & \mathcal{F}^4(\theta_0^0). \end{array}$$

Thus, the matrix representation of L_h and S_h are diagonal matrices (4×4) given by

$$\begin{aligned} L_h^{3g} &= bdiag(L_h^{2g}(\theta_0^0), L_h^{2g}(\theta_0^1)) \in \mathbb{C}^{4\times 4}, \\ S_h^{3g} &= bdiag(S_h^{2g}(\theta_0^0), S_h^{2g}(\theta_0^1)) \in \mathbb{C}^{4\times 4}. \end{aligned}$$

Since the error propagation matrix includes the inverse of discrete operators L_{2h} and L_{4h}, it will be necessary to apply the analysis on the following frequency space

$$\widetilde{\Theta}_{4h} = \{\theta_0^0 \in \Theta_{4h} \big| \widetilde{L}_{4h}(4\theta_0^0) \neq 0, \widetilde{L}_{2h}(2\theta_0^\alpha) \neq 0, \widetilde{L}_{2h}(\theta_\beta^\alpha) \neq 0, \text{ with } \alpha, \beta \in \{0,1\}\}. \tag{4.29}$$

In order to get more insight on the mappings of inter-grid transfer operators, let us note that

$$\begin{aligned} \varphi_h(\theta_0^0, x) = \varphi_h(\theta_1^0, x) = \varphi_{2h}(2\theta_0^0, x), \quad x \in \mathcal{G}_{2h}, \\ \varphi_h(\theta_0^1, x) = \varphi_h(\theta_1^1, x) = \varphi_{2h}(2\theta_0^1, x), \quad x \in \mathcal{G}_{2h}. \end{aligned}$$

Moreover, it is clear that each space of $4h-$harmonics can be decomposed into the following direct sum of two spaces of $2h-$harmonics:

$$\mathcal{F}^4(\theta_0^0) = \mathcal{F}^2(\theta_0^0) \oplus \mathcal{F}^2(\theta_0^1). \tag{4.30}$$

At this point, we can define the inter-grid transfer operators as mappings between the Fourier space $\mathcal{F}_4$ and some spaces spanned by Fourier modes whose frequencies are high frequencies with respect to $\mathcal{G}_{4h}$ but low frequencies with respect to $\mathcal{G}_{2h}$. These are the so-called subspaces of $(2h, 4h)-$harmonics

$$\mathcal{F}_2^4(\theta_0^0) = span\{\varphi_h(\theta^\alpha, \cdot) \big| \alpha \in \{0,1\}\}, \tag{4.31}$$

where $\theta_0^0 \in \Theta_{4h}$. Then, the inter-grid transfer operators:

$$\begin{array}{rccc} R_h^{2h}: & \mathcal{F}^4(\theta_0^0) & \longrightarrow & \mathcal{F}_2^4(\theta_0^0)\}, \\ P_{2h}^h: & \mathcal{F}_2^4(\theta_0^0) & \longrightarrow & \mathcal{F}^4(\theta_0^0). \end{array}$$

Given that the restriction and prolongation operators couple frequencies by pairs, the matrix representation of these operators (with $d = 1$) is given by

$$(R_h^{2h})^{3g}(\theta_0^0) = \begin{pmatrix} R_h^{2h}(\theta_0^0) & 0 \\ R_h^{2h}(\theta_0^1) & 0 \\ 0 & R_h^{2h}(\theta_1^0) \\ 0 & R_h^{2h}(\theta_1^1) \end{pmatrix}, \tag{4.32}$$

$$P_{2h}^h(\theta_0^0) = \begin{pmatrix} (P_{2h}^h)^{3g}(\theta_0^1) & P_{2h}^h(\theta_0^0) & 0 & 0 \\ 0 & 0 & P_{2h}^h(\theta_1^0) & P_{2h}^h(\theta_1^1) \end{pmatrix}. \tag{4.33}$$

In addition to this, the operators L_{2h} and S_{2h} maintain invariant the subspaces of $(2h, 4h)-$harmonics:

$$L_{2h}, S_{2h}: \quad \mathcal{F}_2^4(\theta_0^0) \quad \longrightarrow \mathcal{F}_2^4(\theta_0^0).$$

Then, their matrix representation is

$$L_{2h}^{3g} = diag(\widetilde{L}_h(\theta_0^0), \widetilde{L}_h(\theta_0^1)) \in \mathbb{C}^{2\times 2}, \tag{4.34}$$

$$S_{2h}^{3g} = diag(\widetilde{S}_h(\theta_0^0), \widetilde{S}_h(\theta_0^1)) \in \mathbb{C}^{2\times 2}, \tag{4.35}$$

Let us also note that the coupling among frequencies in the transition from the second grid to the coarsest grid yields

$$\varphi_h(\theta_0^0, x) = \varphi_h(\theta_0^1, x) = \varphi_{4h}(4\theta_0^0, x), \quad x \in \mathcal{G}_{4h}.$$

Consequently, we obtain that

$$\begin{array}{rccc} R_{2h}^{4h}: & \mathcal{F}_2^4(\theta_0^0) & \longrightarrow & \text{span}\left\{\varphi_{4h}(4\theta_0^0, \cdot)\right\}, \\ P_{4h}^{2h}: & \text{span}\left\{\varphi_{4h}(4\theta_0^0, \cdot)\right\} & \longrightarrow & \mathcal{F}_2^4(\theta_0^0), \end{array}$$

and their matrix representation is given by

$$(R_{2h}^{4h})^{3g}(\theta_0^0) = \begin{pmatrix} \widetilde{R}_{2h}^{4h}(\theta_0^0) \\ \widetilde{R}_{2h}^{4h}(\theta_0^1) \end{pmatrix}, \qquad (P_{4h}^{2h})^{3g}(\theta_0^0) = \begin{pmatrix} (\widetilde{P}_{4h}^{2h})^{2g}(\theta_0^0) & , \widetilde{P}_{4h}^{2h}(\theta_0^1) \end{pmatrix}.$$

Moreover, the discrete operator L_{4h} holds the invariance on span $\left\{\varphi_{4h}(4\theta_0^0, \cdot)\right\}$:

$$L_{4h}: \quad \text{span}\left\{\varphi_{4h}(4\theta_0^0, \cdot)\right\} \quad \longrightarrow \text{span}\left\{\varphi_{4h}(4\theta_0^0, \cdot)\right\}.$$

Hence, we can claim that the matrix representation of the three-grid operator $(M_h^{4h})^{3g}$ on the Fourier space is given by a (4×4) matrix by composition of the previous operators:

$$(M_h^{4h})^{3g} = (S_h^{3g})^{\nu_2}\left(I_h^{3g} - (P_{2h}^h)^{3g}(I_{2h} - \left((M_{2h}^{4h})^{3g}\right)^{\gamma})(L_{2h}^{3g})^{-1}(R_h^{2h})^{3g}L_h^{3g}\right)(S_h^{3g})^{\nu_1}, \tag{4.36}$$

where

$$(M_{2h}^{4h})^{3g} = (S_{2h}^{3g})^{\nu_2}(I_{2h}^{3g} - (P_{4h}^{2h})^{3g}\widetilde{L}_{4h}^{-1}(R_{2h}^{4h})^{3g}L_{2h}^{3g})(S_{2h}^{3g})^{\nu_1}. \tag{4.37}$$

Furthermore, the spectral radius of M_h^{4h} can be computed by means of

$$\rho_{3g} = \rho(M_h^{4h}) = \sup_{\theta_0^0 \in \widetilde{\Theta}_{4h}} \rho((M_h^{4h})^{3g}(\theta_0^0)). \tag{4.38}$$

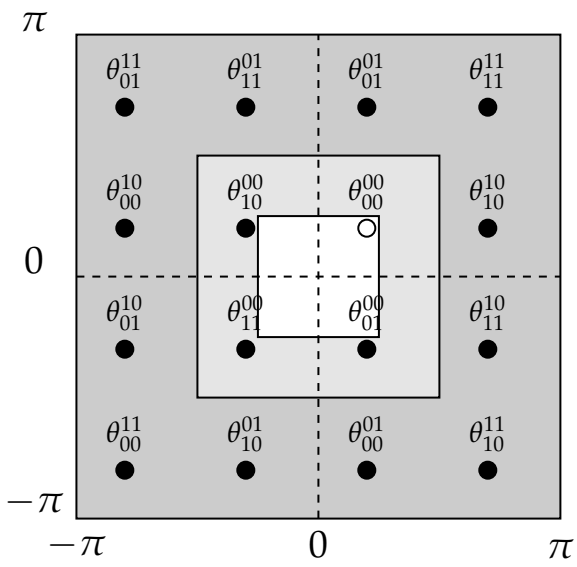


FIGURE 4.6: Coupled frequencies by the three-grid error propagation matrix spanning the subspaces of $4h-$harmonics in the bidimensional case.

The same proceeding is carried out in the two-dimensional case. For this case, we define the low frequency space as $\Theta_{4h} = (-\pi/4, \pi/4]^2$. During the transition between second and third grid, each low frequency $\theta_{00}^{00} \in \Theta_{4h}$ is coupled with the following three frequencies

$$\theta_{ij}^{00} = \theta_{00}^{00} - (i\text{sign}_1(\theta_{00}^{00})\pi/2, j\text{sign}_2(\theta_{00}^{00})\pi/2), \quad i,j \in \{0,1\}. \tag{4.39}$$

The corresponding Fourier modes associated to these frequencies generate the $(2h,4h)-$harmonics as follows

$$\mathcal{E}_4^{16}(\theta_{00}^{00}) = span\{\varphi_h(\theta_{ij}^{00},\cdot)\big|i,j \in \{0,1\}\}. \tag{4.40}$$

Moreover, during the transition from the finest grid to the second grid each low frequency $\theta_{ij}^{00} \in \Theta_{2h} = (-\pi/2, \pi/2]^2$ is coupled with the following three high frequencies in $\mathcal{G}_{2h}$:

$$\theta_{ij}^{nm} = \theta_{ij}^{00} - (n\text{sign}_1(\theta_{ij}^{00})\pi, m\text{sign}_2(\theta_{ij}^{00})\pi), \quad n,m \in \{0,1\}. \tag{4.41}$$

The corresponding Fourier modes to these sixteen frequencies span the subspaces of $4h-$harmonics in the bidimensional case. Thus, we depict the coupled frequencies by the three-grid error propagation matrix in Figure 4.6. In addition, the subspaces of $4h-$harmonics are given by

$$\mathcal{E}^{16}(\theta_{00}^{00}) = \text{span}\left\{\varphi_h(\theta_{ij}^{nm},\cdot)\,|\,i,\,j,\,n,\,m \in \{0,1\}\right\}.$$

As it happened in the unidimensional case, the subspaces of $4h-$harmonics can be expressed in terms of direct sums of the $2h-$harmonics:

$$\mathcal{E}^{16}(\theta_{00}^{00}) = \mathcal{E}^4(\theta_{00}^{00}) \oplus \mathcal{E}^4(\theta_{10}^{00}) \oplus \mathcal{E}^4(\theta_{01}^{00}) \oplus \mathcal{E}^4(\theta_{11}^{00}). \tag{4.42}$$

Besides it, the Fourier space can be decomposed into direct sum of the subspaces of $4h-$harmonics

$$\mathcal{E}(\mathcal{G}_h) = \bigoplus_{\theta_{00}^{00}\in\Theta_{4h}} \mathcal{E}^{16}(\theta_{00}^{00}). \tag{4.43}$$

With respect to the matrix representation of the operators involved in (4.36), their matrix representations have a structure based on the subspaces of $4h-$harmonics and they are given by

$$S_h^{3g}(\theta_{00}^{00}) = bdiag\left\{S_h^{2g}(\theta_{00}^{00}), S_h^{2g}(\theta_{10}^{00}), S_h^{2g}(\theta_{01}^{00}), S_h^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{16\times 16}, \tag{4.44}$$

$$L_h^{3g}(\theta_{00}^{00}) = bdiag\left\{L_h^{2g}(\theta_{00}^{00}), L_h^{2g}(\theta_{10}^{00}), L_h^{2g}(\theta_{01}^{00}), L_h^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{16\times 16}, \tag{4.45}$$

$$(R_h^{2h})^{3g}(\theta_{00}^{00}) = bdiag\left\{(R_h^{2h})^{2g}(\theta_{00}^{00}), (R_h^{2h})^{2g}(\theta_{10}^{00}), (R_h^{2h})^{2g}(\theta_{01}^{00}), (R_h^{2h})^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{4\times 16}, \tag{4.46}$$

$$(P_{2h}^{h})^{3g}(\theta_{00}^{00}) = bdiag\left\{(P_{2h}^{h})^{2g}(\theta_{00}^{00}), (P_{2h}^{h})^{2g}(\theta_{10}^{00}), (P_{2h}^{h})^{2g}(\theta_{01}^{00}), (P_{2h}^{h})^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{16\times 4}, \tag{4.47}$$

$$S_{2h}^{3g}(\theta_{00}^{00}) = diag\left\{S_{2h}^{2g}(\theta_{00}^{00}), S_{2h}^{2g}(\theta_{10}^{00}), S_{2h}^{2g}(\theta_{01}^{00}), S_{2h}^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{4\times 4}, \tag{4.48}$$

$$L_{2h}^{3g}(\theta_{00}^{00}) = diag\left\{L_{2h}^{2g}(\theta_{00}^{00}), L_{2h}^{2g}(\theta_{10}^{00}), L_{2h}^{2g}(\theta_{01}^{00}), L_{2h}^{2g}(\theta_{11}^{00})\right\} \in \mathbb{C}^{4\times 4}, \tag{4.49}$$

$$(R_{2h}^{4h})^{3g}(\theta_{00}^{00}) = ((R_{2h}^{4h})^{2g}(\theta_{00}^{00}), (R_{2h}^{4h})^{2g}(\theta_{10}^{00}), (R_{2h}^{4h})^{2g}(\theta_{01}^{00}), (R_{2h}^{4h})^{2g}(\theta_{11}^{00})) \in \mathbb{C}^{1\times 4}, \tag{4.50}$$

$$(P_{4h}^{2h})^{3g}(\theta_{00}^{00}) = ((P_{4h}^{2h})^{2g}(\theta_{00}^{00}), (P_{4h}^{2h})^{2g}(\theta_{10}^{00}), (P_{4h}^{2h})^{2g}(\theta_{01}^{00}), (P_{4h}^{2h})^{2g}(\theta_{11}^{00})) \in \mathbb{C}^{4\times 1}. \tag{4.51}$$

By using the Fourier representation of these operators with respect the $4h-$harmonics, the matrix representation of M_h^{4h} is obtained as in the one-dimensional case (4.36). Thus, the spectral radius of M_h^{4h} is computed by means of

$$\rho_{3g} = \rho(M_h^{4h}) = \sup_{\theta_0^0\in\widetilde{\Theta}_{4h}} \rho((M_h^{4h})^{3g}(\theta_{00}^{00})), \tag{4.52}$$

where

$$\widetilde{\Theta}_{4h} = \{\theta_{00}^{00} \in \Theta_{4h} | \widetilde{L}_{4h}(4\theta_{00}^{00}) \neq 0, \widetilde{L}_{2h}(2\theta_{ij}^{00}) \neq 0, \widetilde{L}_{2h}(\theta_{ij}^{nm}) \neq 0, \text{ with } i,j,n,m \in \{0,1\}\}.$$

4.3 Analysis for block-wise smoothers

This section is devoted to the analysis of block-wise smoothers such as the multiplicative Schwarz methods. These methods hold the invariance property, that is, the Fourier modes are eigenvectors of this type of smoother, for a rigorous proof see MacLachlan and Oosterlee, 2011. The complexity of the analysis for these methods lies on the overlapping among the blocks of unknowns. Due to the overlapping, the unknowns are relaxed more than once and therefore some intermediate errors appear beside the initial and final errors. Hence, the appearance of these intermediate errors ask for a special strategy for the analysis.

For the sake of simplicity, we consider the $1D$ Poisson equation discretized with B-splines holding maximum global smoothness $\mathcal{C}^{k-1}$. However, the analysis extends straightforwardly to the $2D$ case and it is suitable to other problems by adapting the corresponding stencils. Thus, let us denote with e_h^m the initial error function at $k-$iteration and let $e_h^{m+j/b}$ the intermediate error after the corresponding unknown

has been updated j times out of an odd block size $b = 2c + 1$, with $c \in \mathbb{N}$. Also, we denote with u_i the unknown with parametric location ξ_i. Furthermore, we denote with B_i the block of unknowns centered at u_i. Then, associated to the block B_i we have to solve the local system described in (3.37) where the local corrections to compute are:

$$\delta u_{B_i} = \begin{pmatrix} \delta u_{i-c} \\ \vdots \\ \delta u_i \\ \vdots \\ \delta u_{i+c} \end{pmatrix} = \begin{pmatrix} e_h^{m+1}(\xi_{i-c}) - e_h^{m+(b-1)/b}(\xi_{i-c}) \\ \vdots \\ e_h^{m+(b-c)/b}(\xi_i) - e_h^{m+c/b}(\xi_i) \\ \vdots \\ e_h^{m+1/b}(\xi_{i+c}) - e_h^{m}(\xi_{i+c}) \end{pmatrix}. \tag{4.53}$$

Moreover, the local matrix $L_h^{B_i}$ has size $b \times b$ and it is constructed by means of the projection operators as described in (3.38). For this case, let us note that the entries of $L_h^{B_i}$ are based on the stencil $L_k = [\ell_k, \ell_{k-1}, \ldots, \ell_1, \ell_0, \ell_1, \ldots, \ell_{k-1}, \ell_k]$. For example, the local system associated to the block B_i using a quadratic discretization with global smoothness $\mathcal{C}^1$ is given by

$$\frac{1}{h}\begin{pmatrix} 1 & -1/3 & -1/6 \\ -1/3 & 1 & -1/3 \\ -1/6 & -1/3 & 1 \end{pmatrix}\begin{pmatrix} \delta u_{i-1} \\ \delta u_i \\ \delta u_{i+1} \end{pmatrix} = \begin{pmatrix} r_{i-1} \\ r_i \\ r_{i+1} \end{pmatrix}, \tag{4.54}$$

where r_i denotes the $i-$th component of the residual. In addition, we can generalize for every k the description of local matrix whose associated block has size $b = 3$. For this purpose, let us denote with L_h^B the local matrix associated with a block of three grid points. Then, this matrix is given in terms of L_k as follows:

$$L_h^B = \frac{1}{h}\begin{pmatrix} \ell_0 & \ell_1 & \ell_2 \\ \ell_1 & \ell_0 & \ell_1 \\ \ell_2 & \ell_1 & \ell_0 \end{pmatrix}. \tag{4.55}$$

Unfortunately, we can not give a general description of every local matrix since its structure depends on the relationship between the block size b and the length of the stencil L_k. Without loss of generality, we can consider that the error is given as a single Fourier mode multiplied by a coefficient $\alpha_\theta^{(j)}$, where $j = 0, \ldots, b$ is the number of times that the corresponding unknown has been updated in the current iteration. In this way, we can rewrite the local systems such that these coefficients are the unknowns to solve. For instance, we can rewrite (4.54) by expressing the local corrections and the residual in terms of the error. Then, we have that

$$\begin{pmatrix} e_h^m(\xi) \\ e_h^{m+1/3}(\xi) \\ e_h^{m+2/3}(\xi) \\ e_h^{m+1}(\xi) \end{pmatrix} = \begin{pmatrix} \alpha_\theta^{(0)} e^{\imath\theta\xi} \\ \alpha_\theta^{(1)} e^{\imath\theta\xi} \\ \alpha_\theta^{(2)} e^{\imath\theta\xi} \\ \alpha_\theta^{(3)} e^{\imath\theta\xi} \end{pmatrix}.$$

Thus, the local corrections are given in terms of the error as follows

$$\begin{pmatrix} \delta u_{i-1} \\ \delta u_i \\ \delta u_{i+1} \end{pmatrix} = \begin{pmatrix} \left(\alpha_\theta^{(3)} - \alpha_\theta^{(2)}\right) e^{-\imath\theta} \\ \left(\alpha_\theta^{(2)} - \alpha_\theta^{(1)}\right) \\ \left(\alpha_\theta^{(1)} - \alpha_\theta^{(0)}\right) e^{\imath\theta} \end{pmatrix} e^{\imath\theta\xi}.$$

In addition to this, we have for the residuals that

$$\begin{pmatrix} r_{i-1} \\ r_i \\ r_{i+1} \end{pmatrix} = -\frac{1}{h} \begin{pmatrix} -\frac{1}{6}\alpha_\theta^{(3)} e^{-\imath 3\theta} - \frac{1}{3}\alpha_\theta^{(3)} e^{-\imath 2\theta} + \alpha_\theta^{(2)} e^{-\imath\theta} - \frac{1}{3}\alpha_\theta^{(1)} - \frac{1}{6}\alpha_\theta^{(0)} e^{\imath\theta} \\ -\frac{1}{6}\alpha_\theta^{(3)} e^{-\imath 2\theta} - \frac{1}{3}\alpha_\theta^{(2)} e^{-\imath\theta} + \alpha_\theta^{(1)} - \frac{1}{3}\alpha_\theta^{(0)} e^{\imath\theta} - \frac{1}{6}\alpha_\theta^{(0)} e^{\imath 2\theta} \\ -\frac{1}{6}\alpha_\theta^{(2)} e^{-\imath\theta} - \frac{1}{3}\alpha_\theta^{(1)} + \alpha_\theta^{(0)} e^{\imath\theta} - \frac{1}{3}\alpha_\theta^{(0)} e^{\imath 2\theta} - \frac{1}{6}\alpha_\theta^{(0)} e^{\imath 3\theta} \end{pmatrix} e^{\imath\theta\xi}.$$

Then, we can rearrange the previous system 4.54 into a system of equations for the updated coefficients:

$$\underbrace{\begin{pmatrix} -\frac{1}{6}e^{\imath\theta} & -\frac{1}{3} & e^{-\imath\theta} - \frac{1}{3}e^{-\imath 2\theta} - \frac{1}{6}e^{-\imath 3\theta} \\ -\frac{1}{3}e^{\imath\theta} & 1 & -\frac{1}{3}e^{-\imath\theta} - \frac{1}{6}e^{-\imath 2\theta} \\ e^{\imath\theta} & -\frac{1}{3} & -\frac{1}{6}e^{-\imath\theta} \end{pmatrix}}_{P} \begin{pmatrix} \alpha_\theta^{(1)} \\ \alpha_\theta^{(2)} \\ \alpha_\theta^{(3)} \end{pmatrix} e^{\imath\theta\xi} = \underbrace{\begin{pmatrix} 0 \\ \frac{1}{6}e^{\imath 2\theta} \\ \frac{1}{3}e^{\imath 2\theta} + \frac{1}{6}e^{\imath 3\theta} \end{pmatrix}}_{Q} \alpha_\theta^{(0)} e^{\imath\theta\xi}.$$

Consequently, after canceling the factors $e^{\imath\theta\xi}$ we obtain that

$$\begin{pmatrix} \alpha_\theta^{(1)} \\ \alpha_\theta^{(2)} \\ \alpha_\theta^{(3)} \end{pmatrix} = P^{-1} Q \alpha_\theta^{(0)}.$$

At this point, the amplification factor for the error is given by the last component of $P^{-1}Q$, since this one represents how the Fourier coefficient of the fully corrected error $\alpha_\theta^{(3)}$ is related with that of the initial error $\alpha_\theta^{(0)}$. The previous rearragement can be generalized for any IGA discretization with spline degree k and maximum global smoothness $\mathcal{C}^{k-1}$. Hence, the matrices P and Q for the three-point multiplicative Schwarz smoother are given as follows:

$$P = \begin{pmatrix} \ell_2 e^{\imath\theta} & \ell_1 & \sum_{j=0}^{k} \ell_j e^{-\imath(j+1)\theta} \\ \ell_1 e^{\imath\theta} & \ell_0 & \sum_{j=1}^{k} \ell_j e^{-\imath j\theta} \\ \ell_0 e^{\imath\theta} & \ell_1 & \sum_{j=2}^{k} \ell_j e^{-\imath(j-1)\theta} \end{pmatrix}, \quad Q = \begin{pmatrix} -\sum_{j=3}^{k} \ell_j e^{\imath(j-1)\theta} \\ -\sum_{j=2}^{k} \ell_j e^{\imath j\theta} \\ -\sum_{j=1}^{k} \ell_j e^{\imath(j+1)\theta} \end{pmatrix}.$$

Local Fourier analysis can be performed in the same way for bigger blocks of unknowns, but with heavier computations. Since we consider blocks centered around a grid-point, the number of unknowns within the block, b, is odd. Thus, matrices P and Q for an arbitrary $b-$point multiplicative Schwarz are $(b \times b)-$ and $(b \times 1)-$matrices respectively given by

$$P = \overline{P} D_P, \quad \text{and} \quad Q = \begin{pmatrix} -\sum_{j=b}^{k} \ell_j e^{\imath\left(j-\frac{b-1}{2}\right)\theta} \\ -\sum_{j=b-1}^{k} \ell_j e^{\imath\left(j-\frac{b-1}{2}+1\right)\theta} \\ \vdots \\ -\sum_{j=\frac{b-1}{2}+1}^{k} \ell_j e^{\imath j\theta} \\ -\sum_{j=\frac{b-1}{2}}^{k} \ell_j e^{\imath(j+1)\theta} \\ \vdots \\ -\sum_{j=1}^{k} \ell_j e^{\imath\left(j+\frac{b-1}{2}\right)\theta} \end{pmatrix},$$

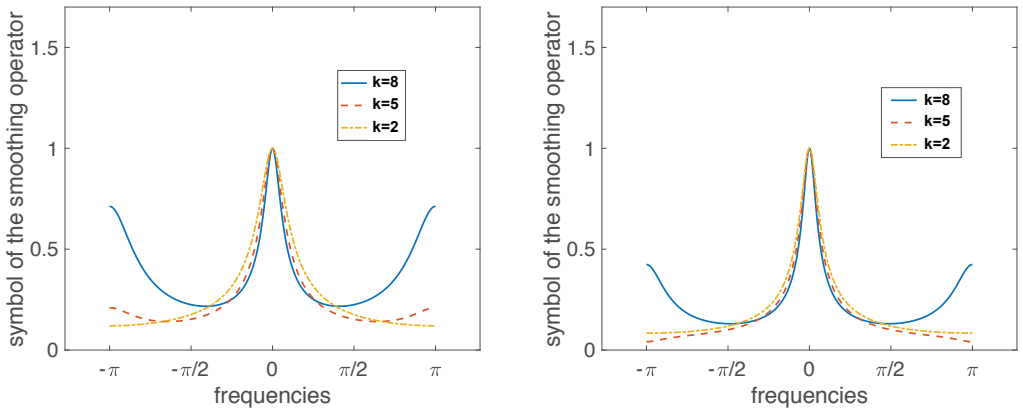
where D_P is the following diagonal matrix

$$D_P = diag\left\{ e^{\imath\frac{b-1}{2}\theta},\ e^{\imath\left(\frac{b-1}{2}-1\right)\theta},\ \cdots,\ e^{\imath\theta},\ 1,\ e^{-\imath\theta},\ \cdots,\ e^{-\imath\left(\frac{b-1}{2}-1\right)\theta},\ 1 \right\},$$

and the matrix $\overline{P}$ is given by

$$\begin{pmatrix}
\ell_{b-1} & \ell_{b-2} & \ell_{b-3} & \cdots & \ell_{\frac{b-1}{2}} & \cdots & \ell_2 & \ell_1 & \sum_{j=0}^{k} \ell_j e^{-\imath\left(j+\frac{b-1}{2}\right)\theta} \\
\ell_{b-2} & \ell_{b-3} & \cdots & & \vdots & \cdots & \ell_1 & \ell_0 & \sum_{j=1}^{k} \ell_j e^{-\imath\left(j+\frac{b-1}{2}-1\right)\theta} \\
\ell_{b-3} & \cdots & & & \vdots & \cdots & \ell_0 & \ell_1 & \sum_{j=2}^{k} \ell_j e^{-\imath\left(j+\frac{b-1}{2}-2\right)\theta} \\
\vdots & & & & \vdots & & & & \vdots \\
\vdots & & & & \ell_0 & & & & \sum_{j=\frac{b-1}{2}}^{k} \ell_j e^{-\imath j\theta} \\
\vdots & & & & \vdots & & . & & \vdots \\
\ell_2 & \ell_1 & \ell_0 & \cdots & \cdots & & \cdots & \ell_{b-4} & \sum_{j=b-3}^{k} \ell_j e^{-\imath\left(j-\frac{b-1}{2}+2\right)\theta} \\
\ell_1 & \ell_0 & \ell_1 & \cdots & \cdots & \cdots & \ell_{b-4} & \ell_{b-3} & \sum_{j=b-2}^{k} \ell_j e^{-\imath\left(j-\frac{b-1}{2}+1\right)\theta} \\
\ell_0 & \ell_1 & \cdots & \cdots & \ell_{\frac{b-1}{2}} & \cdots & \ell_{b-3} & \ell_{b-2} & \sum_{j=b-1}^{k} \ell_j e^{-\imath\left(j-\frac{b-1}{2}\right)\theta}
\end{pmatrix},$$

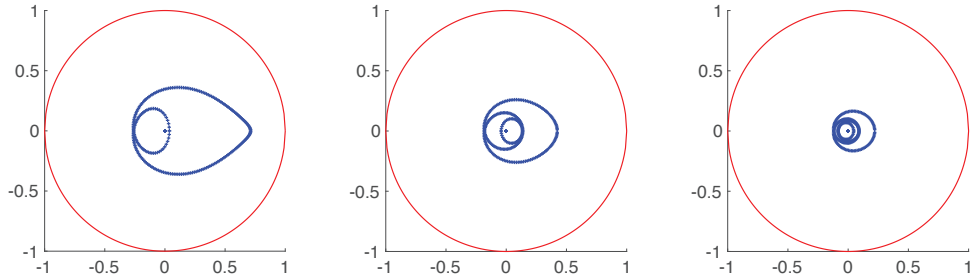
where $\ell_i = 0$ for $i > k$. Using this analysis, we can obtain the Fourier symbol of the multiplicative Schwarz methods for isogeometric discretizations. For instance, in Figure 4.7 we show the Fourier symbol of the smoothing operator corresponding to (a) the three-point and (b) the five-point multiplicative Schwarz smoothers, for three different spline degrees $k = 2, 5, 8$.



(a) Three-point Schwarz (b) Five-point Schwarz

FIGURE 4.7: Symbol of the smoothing operator corresponding to (a) the three-point and (b) the five-point multiplicative Schwarz smoothers, for three different spline degrees $k = 2, 5, 8$.

Furthermore, we can perform this analysis for the matrix representation of the smoother with respect the subspaces of $2h-$harmonics such that the eigenvalues of the two-grid operator based on these smoothers are obtained. Hence, in order to find out the performance on high-order discretizations, we show in Figure 4.8 the distribution of the eigenvalues of the two-grid method based on the three-, five- and seven-point multiplicative Schwarz smoothers with maximum overlapping applied to $k = 8$.



(a) Three-point Schwarz (b) Five-point Schwarz (c) Seven-point Schwarz

FIGURE 4.8: Distribution of the eigenvalues of the two-grid method based on the three-, five- and seven-point multiplicative Schwarz smoothers with maximum overlapping applied to $k = 8$.

4.4 Infinite subgrids analysis

As mentioned in Section 4.2, the standard local Fourier analysis makes some assumptions such as the discrete operator is linear with constant coefficients. This limitation to the standard LFA prevents a direct analysis for some problems. This fact was pointed out in Kumar et al., 2019, where the authors introduced a LFA based on infinite subgrids such that the analysis is suitable for PDEs with jumping and random coefficients. Instead of considering an infinite grid, the idea of this LFA lies on the periodic extension of several infinite subgrids. Thus, the main difference among the standard analysis and the infinite subgrids analysis is the choice of the basis functions considered for the error decomposition. This specific basis for the Fourier spaces allows us to generalize the standard analysis. For this purpose, let us consider a splitting of the grid $\mathcal{G}_h$ into n_g^d infinite subgrids. Hence, we proceed by taking a fixed window with size $\underbrace{n_g \times \ldots \times n_g}_{d}$ and then considering its periodic extension. The size of the window is chosen such that the different stencils provided by the corresponding discretization are taken into account. Note that we are considering the same size n_g at each direction for the sake of simplicity, but this is not restrictive for this analysis. With a fixed size of the window, for every $\boldsymbol{j} = (j_1, \ldots, j_d)$, with $j_1, \ldots, j_d = 0, \ldots, n_g - 1$, the infinite subgrids are defined as follows:

$$G_h^{\boldsymbol{j}} = \{\boldsymbol{j}h + (l_1, \ldots, l_d)n_g h \mid l_1, \ldots, l_d \in \mathbb{Z}\}. \tag{4.56}$$

For instance, in Figure 4.9 we show the periodic extension of a window with size 2×2 on an infinite grid with 4 infinite subgrids ($\square$, $\bullet$, $\circ$ and $\times$).

In the infinite subgrids LFA, the low-frequency space is given by

$$\Theta_{n_g h} := (-\frac{\pi}{n_g h}, \frac{\pi}{n_g h}]^d. \tag{4.57}$$

For each low frequency $\theta^0 \in \Theta_{n_g h}$, we introduce the following grid functions:

$$\psi_h^{\boldsymbol{j}}(\theta^0, x) = \varphi_h(\theta^0, x)\chi_{G_h^{\boldsymbol{j}}}(x), \tag{4.58}$$

where

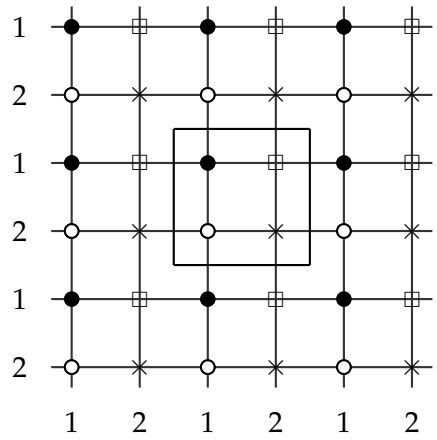


FIGURE 4.9: Periodic extension of a window with size 2×2 on an infinite grid with infinite subgrids ($\square$, $\bullet$, $\circ$ and $\times$).

$$\chi_{G_h^{\boldsymbol{j}}}(x) = \begin{cases} 1, & \text{if } x \in G_h^{\boldsymbol{j}} \\ 0, & \text{otherwise.} \end{cases} \tag{4.59}$$

As it was seen in Kumar et al., 2019, the Fourier space spanned by these functions

$$\mathcal{F}_h^{n_g^d}(\theta^0) = span\{\psi_h^{\boldsymbol{j}}(\theta^0, \cdot),\ j_1, \ldots, j_d = 0, \ldots, n_g - 1\}, \tag{4.60}$$

is the same Fourier space generated by the standard Fourier modes

$$\text{span}\{\varphi_h(\theta_j^0, \cdot), \theta_j^0 = \theta^0 + 2\pi \boldsymbol{j}/(n_g h),\ j_1, \ldots, j_d = 0, \ldots, n_g - 1\}. \tag{4.61}$$

Whereas the Fourier representation of the discrete operators by means of the standard Fourier modes is a diagonal matrix, with this specific basis the symbol of the discrete operator is a dense matrix. Hence, let us consider the univariate discrete Poisson operator using quadratic FEM with a fixed $n_g = 4$. Then, the Fourier representation of L_h with respect to $\{\psi_h^j(\theta^0, \cdot)\}_{j=1}^4$ is given by

$$\widehat{L}_h(\theta^0) = \begin{pmatrix} \frac{8}{3} & -\frac{2}{3}e^{\imath\theta^0} & -\frac{2}{3}(e^{2\imath\theta^0} + e^{-2\imath\theta^0}) & -\frac{2}{3}e^{-\imath\theta^0} \\ -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} & -\frac{2}{3}e^{\imath\theta^0} & 0 \\ -\frac{2}{3}(e^{2\imath\theta^0} + e^{-2\imath\theta^0}) & -\frac{2}{3}e^{-\imath\theta^0} & \frac{8}{3} & -\frac{2}{3}e^{\imath\theta^0} \\ -\frac{2}{3}e^{\imath\theta^0} & 0 & -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} \end{pmatrix}, \tag{4.62}$$

with $\theta^0 \in \Theta_{4h}$. For this example, the points contained in the window are vertex, mid-point, vertex, mid-point as depicted in Figure 4.10.

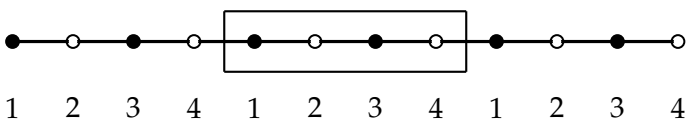


FIGURE 4.10: Periodic extension of the considered window of 4 points for the Fourier representation (4.62). The two infinite subgrids consisting on vertices ($\bullet$) and midpoints ($\circ$) are depicted and they are distributed as ($\bullet, \circ, \bullet, \circ$) in the window.

Note that, with this analysis, the symbols of the discrete and smoother operators are $n_g^d \times n_g^d$ matrices. In fact, these operators hold invariance property on $\mathcal{F}_h^{n_g^d}(\theta)$ since it is actually the Fourier space with a change of basis.

$$\begin{aligned} L_h: \quad \mathcal{F}_h^{n_g^d}(\theta) &\longrightarrow \mathcal{F}_h^{n_g^d}(\theta), \\ S_h: \quad \mathcal{F}_h^{n_g^d}(\theta) &\longrightarrow \mathcal{F}_h^{n_g^d}(\theta). \end{aligned}$$

The structure of the matrix representation of the discrete operator and smoother with respect the basis functions $\psi_h^j(\theta^0, \cdot)$, with $j_1, \ldots, j_d = 0, \ldots, n_g - 1$ depends on the number of different subgrids, i.e. the number of stencils provided by the discretization. Thus, there is not a general expression for the matrix construction. For example, the matrix representation of Gauss-Seidel smoother with respect the new basis using the previous quadratic FEM discretization with $n_g = 4$ is:

$$\widehat{S}_{GS}(\theta^0) = -\left(\widehat{L}_h^+(\theta^0)\right)^{-1} \widehat{L}_h^-(\theta^0),$$

where

$$\widehat{L}_h^+(\theta^0) = \begin{pmatrix} \frac{8}{3} & 0 & -\frac{2}{3}e^{-2\imath\theta^0} & -\frac{2}{3}e^{-\imath\theta^0} \\ -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} & 0 & 0 \\ -\frac{2}{3}e^{-2\imath\theta^0} & -\frac{2}{3}e^{-\imath\theta^0} & \frac{8}{3} & 0 \\ 0 & 0 & -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} \end{pmatrix}, \quad \widehat{L}_h^-(\theta^0) = \widehat{L}_h(\theta^0) - \widehat{L}_h^+(\theta^0).$$

In addition, the Fourier representation of the smoother using the new basis has additional issues when a block-wise smoother is considered. Since we already introduced the LFA for multiplicative Schwarz methods, in this section we describe the matrix representation of additive Schwarz smoothers using the infinite subgrids approach. It is very important to set an appropiate window size n_g suitable to the block size b, the overlapping o, the number of stencils $k - s$ and the type of analysis (smoothing analysis, two-grid analysis, etc.). First, it is clear that necessarily $n_g > b$ since otherwise the analysis can not capture the effect of the smoother on the window. However, the analysis based on infinite subgrids for these smoothers is very specific to each case. Secondly, we should take into account that it is necessary to hold $n_g > b$. Moreover, the window size n_g must be multiple of $b - o$ in order to apply uniformly the smoothing effect on the window points taking into account its periodic extension. Furthermore, n_g must be multiple of the number of stencils $k - s$ since they periodic extension of the window has to represent the infinite grid $\mathcal{G}_h$. In fact, the window size must be multiple of $2^{l-1}(k - s)$, where l denotes here the number of grids considered for the analysis. Finally, the minimum value of the window size on each direction is given by

$$n_g^{min} = \text{l.c.m.} \left\{ 2^{l-1}(k-s), \min_{j \in \mathbb{N}} \{(b-o)j \big| (b-o)j > b\} \right\}, \tag{4.63}$$

where l.c.m. denotes the least common multiple. Note that for the sake of simplicity, we consider the same window size on each direction. However, this is not restrictive for the analysis carried out here. As we can see, to determine the minimum (even any appropiate) value of the window size is not a straightforward task and this is why the matrix representation of the smoother can not be generalized easily. Once we have set n_g, we need to construct the local matrices associated to the blocks of

points that are updated on the window during the smoothing process. For this task, we can apply local projection operators to the Fourier representation $\widehat{L}_h$ as done in (3.38). The number of blocks of unknowns that take place on the window is given by

$$n_b = \left(\frac{n_g}{b-o}\right)^d. \tag{4.64}$$

Hence, let us denote the local projection operators with V_i, $i = 1, \ldots, n_b$. Then, the matrix representation of the local matrix associated to the block of unknowns B_i is given by

$$\widehat{L}_h^{B_i}(\theta^0) = V_i \widehat{L}_h(\theta^0) V_i^t. \tag{4.65}$$

As an example, let us consider again the previous univariate quadratic FEM discretization with $n_g = 4$ starting with a vertex point at the left side (see Figure (4.10)). Then, the local projection operators are given by

$$V_1 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \qquad V_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

$$V_3 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad V_4 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Therefore we have that the Fourier representation of the local matrices is

$$\widehat{L}_h^{B_1}(\theta^0) = \widehat{L}_h^{B_3}(\theta^0) = \begin{pmatrix} \frac{4}{3} & -\frac{2}{3}e^{\imath\theta^0} & 0 \\ -\frac{2}{3}e^{-\imath\theta^0} & \frac{8}{3} & -\frac{2}{3}e^{\imath\theta^0} \\ 0 & -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} \end{pmatrix},$$

$$\widehat{L}_h^{B_2}(\theta^0) = \widehat{L}_h^{B_4}(\theta^0) = \begin{pmatrix} \frac{8}{3} & -\frac{2}{3}e^{\imath\theta^0} & -\frac{2}{3}e^{2\imath\theta^0} \\ -\frac{2}{3}e^{-\imath\theta^0} & \frac{4}{3} & -\frac{2}{3}e^{\imath\theta^0} \\ -\frac{2}{3}e^{-2\imath\theta^0} & -\frac{2}{3}e^{-\imath\theta^0} & \frac{8}{3} \end{pmatrix}.$$

At this point, the contribution obtained by solving the local system associated to a block B_i is

$$\widehat{c}_i(\theta^0) = diag\{\omega\} V_i^t (\widehat{L}_h^{B_i}(\theta^0))^{-1} V_i \widehat{L}_h(\theta^0), \tag{4.66}$$

where ω is the vector of weights introduced in (3.40). Thus, the Fourier representation of additive Schwarz smoothers by means of infinite subgrids analysis is given by

$$\widehat{S}_{ad}(\theta^0) = \widehat{I}(\theta^0) - \sum_{i=1}^{n_b^d} \widehat{c}_i(\theta^0), \tag{4.67}$$

which is a $(n_g^d \times n_g^d)$-matrix. As mentioned in Section 3.3.2, the weights contained in ω have a crucial role for the performance of the smoother. In fact, the analysis described here allows us to find the optimal values of ω for every case. In addition, when we have more than one stencil ($k - s > 1$) and there is not maximum overlapping among the blocks ($b - o > 1$) there is more than one way to select the blocks of unknowns uniformly located on the periodic extension of the window. Thus, each

of these ways provide a different smoothing effect since their corresponding local systems are also different. Thanks to LFA, we can find out which way is the best to place the blocks.

Based on their invariance property for the $2h-$harmonics, the inter-grid transfer operators (considering a standard coarsening $H = 2h$) also hold

$$\begin{aligned} R_h^{2h}: \quad & \mathcal{F}_h^{n_g^d}(\theta^0) \quad \longrightarrow \quad \mathcal{F}_{2h}^{n_g^d/2^d}(2\theta^0), \\ P_{2h}^{h}: \quad & \mathcal{F}_{2h}^{n_g^d/2^d}(2\theta^0) \quad \longrightarrow \quad \mathcal{F}_h^{n_g^d}(\theta^0), \end{aligned}$$

with $\theta^0 \in \Theta_{2n_g h}$. However, there is not a general expression for matrix representation of the restriction and prolongation operators using this analysis. The length of the stencils appearing in the inter-grid tranfer operators changes with respect to k and s. In addition, the structure of these operators varies depending on the location of the window within the infinite grid. For instance, in Figure 4.10 the first point of the window is a vertex point but it could be chosen as a midpoint. In order to illustrate the construction of the restriction operator for the case depicted in Figure 4.10, we show in Figure 4.11 a scheme with the existing contributions of the grid-points on the fine grid to the ones of the coarse grid.

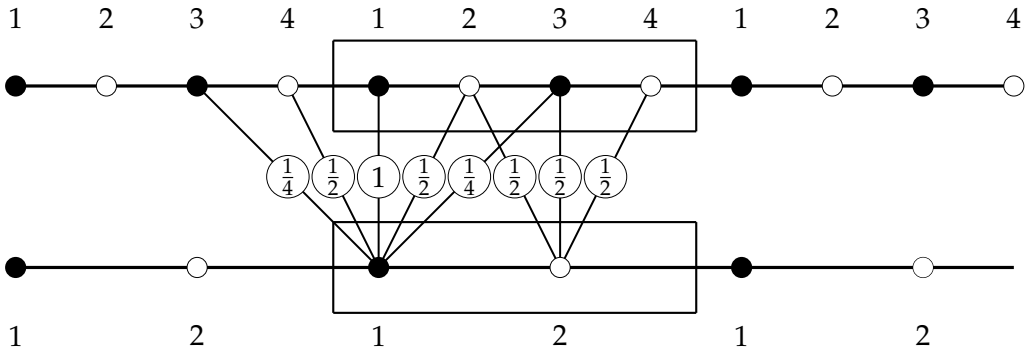


FIGURE 4.11: Scheme with contributions of the grid-points on the fine grid to the ones of the coarse grid.

Hence, the Fourier representation of the restriction operator for this case is given by

$$\widehat{R}_h^{2h}(\theta^0) = \begin{pmatrix} 1 & \frac{1}{2}e^{\imath\theta^0} & \frac{1}{4}(e^{-2\imath\theta^0} + e^{2\imath\theta^0}) & \frac{1}{2}e^{-\imath\theta^0} \\ 0 & \frac{1}{2}e^{-\imath\theta^0} & \frac{1}{2} & \frac{1}{2}e^{\imath\theta^0} \end{pmatrix}.$$

About the prolongation operator, its matrix representation can be taken as the transposed conjugate matrix of $\widehat{R}_h^{2h}$:

$$\widehat{P}_{2h}^{h}(\theta^0) = \begin{pmatrix} 1 & 0 \\ \frac{1}{2}e^{\imath\theta^0} & \frac{1}{2}e^{-\imath\theta^0} \\ \frac{1}{4}(e^{-2\imath\theta^0} + e^{2\imath\theta^0}) & \frac{1}{2} \\ \frac{1}{2}e^{-\imath\theta^0} & \frac{1}{2}e^{\imath\theta^0} \end{pmatrix}.$$

Fortunately, opposite to the rest of operators the matrix representation of inter-grid transfer operators for a $d-$dimensional problem can be obtained by means of d times the kronecker product of the univariate counterpart. Now, for a two-grid analysis only the discrete operator L_{2h} is left. Thus, let us note that this discrete operator holds

$$L_{2h} : \mathcal{F}_{2h}^{n_g^d/2^d}(2\theta^0) \longrightarrow \mathcal{F}_{2h}^{n_g^d/2^d}(2\theta^0).$$

Note that its matrix representation can be obtained exactly by the same way we did with $\widehat{L}_h$ or we can approximate it by means of Galerkin approximation as follows:

$$\widehat{L}_{2h}(2\theta^0) = \widehat{R}_h^{2h}(\theta^0)\widehat{L}_h(\theta^0)\widehat{P}_{2h}^h(2\theta^0).$$

Hence, due to the relation between the grid-functions of the given new Fourier basis and the standard Fourier modes we can claim that the two grid operator satisfies for any frequency $\theta^0 \in \Theta_{n_g h}$ the following invariance property:

$$M_h^{2h} : \mathcal{F}_h^{n_g^d}(\theta^0) \longrightarrow \mathcal{F}_h^{n_g^d}(\theta^0). \tag{4.68}$$

Finally, the goal of this analysis is to compute the spectral radius of the two-grid operator M_h^{2h} described in (4.11) by means of

$$\rho_{2g} = \sup_{\theta^0 \in \widetilde{\Theta}_{2n_g h}} \rho(\widehat{M}_h^{2h}(\theta^0)). \tag{4.69}$$

where

$$\widehat{M}_h^{2h}(\theta^0) = \widehat{S}_h^{\nu_2}(\theta^0)\left(\widehat{I}_h(\theta^0) - \widehat{P}_{2h}^h(\theta^0)(\widehat{L}_{2h}(\theta^0))^{-1}\widehat{R}_h^{2h}(\theta^0)\widehat{L}_h(\theta^0)\right)\widehat{S}_h^{\nu_1}(\theta^0), \tag{4.70}$$

and the frequency space $\widetilde{\Theta}_{2n_g h}$ is given by

$$\widetilde{\Theta}_{2n_g h} = \Theta_{2n_g h} \setminus \left\{\theta_0 \in \Theta_{2n_g h} \middle| \det\left(\widehat{L}_{2h}(2\theta^0)\right) = 0 \text{ or } \det\left(\widehat{L}_h(\theta^0)\right) = 0\right\}.$$

4.5 Analysis for systems of PDEs

Local Fourier analysis is also useful for the design of multigrid methods applied to systems of PDEs. In fact, it is specially difficult to find appropiate methods for this type of problems. Thus, the analysis previously introduced for scalar problems is extended to systems of PDEs by means of vector forms. In this framework, let q be the number of variables involved in the PDE system. Based on the Fourier transform for vector functions, the analysis assumes that the error function $\boldsymbol{e}_h(\theta, x)$ can be decomposed into the following Fourier modes:

$$\boldsymbol{\varphi_h}(\boldsymbol{\theta}, \boldsymbol{x}) := \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} e^{\imath\boldsymbol{\theta}\cdot\boldsymbol{x}/\boldsymbol{h}}, \tag{4.71}$$

where $\boldsymbol{h} = (h_1, \ldots, h_q)$, $\boldsymbol{x} = (x_1, \ldots, x_q)$, $x_i \in \mathcal{G}_h$, $i = 1, \ldots, q$, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_q)$, $\theta_j \in \Theta_h$ with $j = 1, \ldots, q$. For the sake of simplicity, we consider the same grid size $h = h_1 = \ldots = h_q$ for all the variables. However, it is also possible for the analysis to consider a different grid size for every variable of the PDE system. LFA for systems of PDEs neglects the effect of boundary conditions and considers linear discrete operators with constant coefficients. Hence, the discrete problem $\boldsymbol{L_h u_h} = \boldsymbol{f_h}$ can be written as follows:

$$\begin{pmatrix} L_h^{1,1} & \dots & L_h^{1,q} \\ \vdots & \ddots & \vdots \\ L_h^{q,1} & \dots & L_h^{q,q} \end{pmatrix} \begin{pmatrix} u_h^1 \\ \vdots \\ u_h^q \end{pmatrix} = \begin{pmatrix} f_h^1 \\ \vdots \\ f_h^q \end{pmatrix}, \tag{4.72}$$

where every $L_h^{i,j}$, $i,j = 1,\dots,q$ are scalar discrete operators defined as in (3.2). At this point, the vector Fourier modes are eigenfunctions of the discrete operators $\boldsymbol{L_h}$, that is,

$$\boldsymbol{L_h}\boldsymbol{\varphi_h}(\boldsymbol{\theta},\boldsymbol{x}) = \widetilde{\boldsymbol{L}}_{\boldsymbol{h}}(\boldsymbol{\theta})\boldsymbol{\varphi_h}(\boldsymbol{\theta},\boldsymbol{x}) = \begin{pmatrix} \widetilde{L}_h^{1,1} & \dots & \widetilde{L}_h^{1,q} \\ \vdots & \ddots & \vdots \\ \widetilde{L}_h^{q,1} & \dots & \widetilde{L}_h^{q,q} \end{pmatrix} \boldsymbol{\varphi_h}(\boldsymbol{\theta},\boldsymbol{x}), \tag{4.73}$$

where the Fourier symbol $\widetilde{\boldsymbol{L}}_{\boldsymbol{h}}$ is composed of the scalar Fourier symbols $\widetilde{L}_h^{i,j}$ corresponding to the scalar discrete operators $L_h^{i,j}$, with $i,j = 1,\dots,q$. The smoothing analysis for PDE systems is carried out applying this vector version of the discrete operator. In fact, classical iterative methods such as Jacobi or Gauss-Seidel split the vector discrete operator as follows

$$\boldsymbol{L_h} = \boldsymbol{L_h^+} + \boldsymbol{L_h^-},$$

with $\boldsymbol{L_h^+}$ and $\boldsymbol{L_h^-}$ denoting the parts of $\boldsymbol{L_h}$ corresponding to the unknowns that are going to be updated and the ones that have been already updated, respectively. This partition gives place to the numerical iterative scheme

$$\boldsymbol{L_h^+}\boldsymbol{u_h^{m+1}} = \boldsymbol{f_h} - \boldsymbol{L_h^-}\boldsymbol{u_h^m}.$$

With this scheme, the smoother operator is given by

$$\boldsymbol{S_h} = -\boldsymbol{L_h^+}\boldsymbol{L_h^-}. \tag{4.74}$$

and its Fourier symbol is

$$\widetilde{\boldsymbol{S}}_{\boldsymbol{h}}(\boldsymbol{\theta}) = -\widetilde{\boldsymbol{L}}_{\boldsymbol{h}}^{+}(\boldsymbol{\theta})\widetilde{\boldsymbol{L}}_{\boldsymbol{h}}^{-}(\boldsymbol{\theta}).$$

Remark: It might not possible to apply standard smoothers for some coupled systems of PDEs. In fact, sometimes it is necessary to apply a coupled smoother with a special structure. This is the case for example of the Stokes equations. A suitable smoothing strategy for these equations is to apply a coupled Schwarz method, that is, the grid is split into blocks such that each block B_i is associated to one pressure variable p_i. Furthermore, each block is constructed such that the velocity field unknowns involved in the corresponding pressure equation are also part of the block. Thus, the coupled smoother has the structure given in (3.39) or (3.40) but with particular projection operators acting on the whole set of variables unknowns.

At this point, the smoothing factor for PDE systems is defined as

$$\mu = \sup_{\boldsymbol{\theta}\in\boldsymbol{\Theta_h}\setminus\boldsymbol{\Theta_{2h}}} |\rho(\widetilde{\boldsymbol{S}}_{\boldsymbol{h}}(\boldsymbol{\theta})|. \tag{4.75}$$

In addition, the two-grid analysis is extended to PDE systems by means of the Fourier representation of the operators with respect to the subspaces of $2h-$harmonics.

Here, the goal of the two-grid analysis is to compute the spectral radius of the two-grid operator given by

$$\boldsymbol{M}_h^{2h} = \boldsymbol{S}_h^{\nu_2}\left(\boldsymbol{I}_h - \boldsymbol{P}_{2h}^h(\boldsymbol{L}_{2h})^{-1}\boldsymbol{R}_h^{2h}\boldsymbol{L}_h\right)\boldsymbol{S}_h^{\nu_1}, \tag{4.76}$$

where ν_1, ν_2 are the pre- and post- smoothing steps, $\boldsymbol{R}_h^{2h}$, $\boldsymbol{P}_{2h}^h$ are the restriction and prolongation operators and $\boldsymbol{L}_{2h}$ is the discrete operator at the coarse grid considering a standard coarsening $H = 2h$. The subspaces of $2h-$harmonics are defined as in the scalar case and all the operators hold the corresponding invariance property. Thus, the matrix representation of the error propagation matrix $\boldsymbol{M}_h^{2h}$ with respect the subspaces of the $2h-$harmonics is a $(4q \times 4q)$ matrix given by

$$(\boldsymbol{M}_h^{2h})^{2g}(\boldsymbol{\theta}) = (\boldsymbol{S}_h^{2g}(\boldsymbol{\theta}))^{\nu_2}\left(\boldsymbol{I}_h^{2g}(\boldsymbol{\theta}) - (\boldsymbol{P}_{2h}^h)^{2g}(\boldsymbol{\theta})(\boldsymbol{L}_{2h}^{2g}(\boldsymbol{\theta}))^{-1}(\boldsymbol{R}_h^{2h})^{2g}(\boldsymbol{\theta})\boldsymbol{L}_h^{2g}(\boldsymbol{\theta})\right)(\boldsymbol{S}_h^{2g}(\boldsymbol{\theta}))^{\nu_1}, \tag{4.77}$$

where $\boldsymbol{S}_h^{2g}, (\boldsymbol{P}_{2h}^h)^{2g}, (\boldsymbol{R}_h^{2h})^{2g}, \boldsymbol{L}_h^{2g}$ and $\boldsymbol{L}_{2h}^{2g}$ are the matrix representations of the corresponding operators with respect the subspaces of $2h-$harmonics. Given that the spaces of $2h-$harmonics form a decomposition of the Fourier space, the spectral radius of $\boldsymbol{M}_h^{2h}$ is computed with

$$\rho_{2g} = \rho(\boldsymbol{M}_h^{2h}) = \sup_{\boldsymbol{\theta^0}\in\widetilde{\boldsymbol{\Theta}}_{\boldsymbol{2h}}} \rho((\boldsymbol{M}_h^{2h})^{2g}(\boldsymbol{\theta^0})), \tag{4.78}$$

where

$$\widetilde{\boldsymbol{\Theta}}_{2h} = \boldsymbol{\Theta}_{2h} \setminus \left\{\boldsymbol{\theta} \in \boldsymbol{\Theta}_{2h} \middle| \det\left(\widetilde{\boldsymbol{L}}_h(\boldsymbol{\theta})\right) = 0 \text{ or } \det\left(\widetilde{\boldsymbol{L}}_{2h}(\boldsymbol{\theta})\right) = 0\right\}.$$

Analogously, the three-grid analysis can be extended to PDE systems. For this analysis, the matrix representation of the error propagation operator with respect the subspaces of $4h-$harmonics is given by

$$\begin{aligned}(\boldsymbol{M}_h^{4h})^{3g}(\boldsymbol{\theta}) = \\ (\boldsymbol{S}_h^{3g})^{\nu_2}(\boldsymbol{\theta})(\boldsymbol{I}_h^{3g}(\boldsymbol{\theta}) - (\boldsymbol{P}_{2h}^h)^{3g}(\boldsymbol{\theta})(\boldsymbol{I}_{2h}(\boldsymbol{\theta}) - ((\boldsymbol{M}_{2h}^{4h})^{3g}(\boldsymbol{\theta}))^{\gamma})(\boldsymbol{L}_{2h}^{3g}(\boldsymbol{\theta}))^{-1}(\boldsymbol{R}_h^{2h})^{3g}(\boldsymbol{\theta})\boldsymbol{L}_h^{3g}(\boldsymbol{\theta}))(\boldsymbol{S}_h^{3g})^{\nu_1}(\boldsymbol{\theta}),\end{aligned} \tag{4.79}$$

where

$$(\boldsymbol{M}_{2h}^{4h})^{3g}(\boldsymbol{\theta}) = (\boldsymbol{S}_{2h}^{3g}(\boldsymbol{\theta}))^{\nu_2}(\boldsymbol{I}_{2h}^{3g}(\boldsymbol{\theta}) - (\boldsymbol{P}_{4h}^{2h})^{3g}(\boldsymbol{\theta})(\widetilde{\boldsymbol{L}}_{4h}(\boldsymbol{\theta}))^{-1}(\boldsymbol{R}_{2h}^{4h})^{3g}(\boldsymbol{\theta})\boldsymbol{L}_{2h}^{3g}(\boldsymbol{\theta}))(\boldsymbol{S}_{2h}^{3g}(\boldsymbol{\theta}))^{\nu_1}. \tag{4.80}$$

Note that the subspaces of $4h-$harmonics and $(2h, 4h)-$harmonics are the same for the vector case, but now the matrix representation of the three-grid operator is a $(16q, 16q)$ matrix. By applying the same invariance properties given in 4.2.3 for the scalar case, the spectral radius of the three-grid operator is computed with

$$\rho_{3g} = \rho(\boldsymbol{M}_h^{4h}) = \sup_{\boldsymbol{\theta}\in\widetilde{\boldsymbol{\Theta}}_{4h}} \rho((\boldsymbol{M}_h^{4h})^{3g}(\boldsymbol{\theta})), \tag{4.81}$$

where

$$\widetilde{\boldsymbol{\Theta}}_{4h} = \left\{\boldsymbol{\theta^0} \in \boldsymbol{\Theta}_{4h} \middle| \det\left(\boldsymbol{L}_{4h}(4\boldsymbol{\theta^0})\right) \neq 0, \det\left(\boldsymbol{L}_{2h}(2\boldsymbol{\theta_i^0})\right) \neq 0, \det\left(\boldsymbol{L}_{2h}(2\boldsymbol{\theta_i^n})\right) \neq 0\right\}, \tag{4.82}$$

and $\boldsymbol{\theta_i^0}$, $\boldsymbol{\theta_i^n}$ denote the frequencies that generate the corresponding subspaces of $(2h, 4h)-$ and $4h-$harmonics.

Chapter 5

Multigrid methods for IGA discretizations for scalar problems

Given their optimality, the aim of many recent works has been the application of multigrid algorithms to isogeometric discretizations, see Gahalaut, Kraus, and Tomar, 2013; Hofreither and Zulehner, 2014; Hofreither and Zulehner, 2016; Donatelli et al., 2016; Hofreither and Zulehner, 2015; Chemin, Elguedj, and Gravouil, 2015; Hofreither and Takacs, 2017; Hofreither, Takacs, and Zulehner, 2017. In some of these works (Donatelli et al., 2016; Hofreither and Zulehner, 2015; Hofreither and Takacs, 2017; Hofreither, Takacs, and Zulehner, 2017), it was pointed out the fact that the convergence rates of multigrid methods with classical relaxation schemes deteriorate when the spline degree k is increased. In order to overcome this issue, a multigrid method based on preconditioned CG or GMRES smoothers was proposed in Donatelli et al., 2016. In addition, mass smoothers were proposed based on stable splittings of spline spaces (Hofreither and Takacs, 2017) or using boundary correction (Hofreither, Takacs, and Zulehner, 2017). Although these works showed robustness with respect to k, the proposed multigrid methods (for $d \geq 2$) required many iterations to converge (Hofreither and Takacs, 2017). Thus, our attention is devoted to the design of robust multigrid methods with respect to the spline degree k. With this purpose, Schwarz methods are proposed as smoothers within the multigrid algorithm (see Pé de la Riva, Rodrigo, and Gaspar, 2019; Pé de la Riva, Rodrigo, and Gaspar, 2020; Pé de la Riva, Rodrigo, and Gaspar, 2021).

In this chapter, our goal is to propose robust multigrid methods for IGA discretizations applied to some scalar problems. We will show that the use of pointwise smoothers such as Gauss-Seidel is ineffective for the design of robust multigrid methods with respect to the spline degree k. In order to achieve robustness, it is necessary to find smoothers with a powerful smoothing effect which are suitable for different values of the polynomial degree. In IGA, the number of basis functions sharing a common support increase linearly with k and therefore the size of the stencils as well. Block-wise smoothers seem to be ideal for our purpose. However, it is necessary to set the block size and the overlapping among the blocks for each spline degree k. For this task, we make use of the LFA introduced in Chapter 4. As we will see, our LFA and numerical results show an excellent match for the considered problems. Thus, the analysis proves to be useful also for the design of multigrid methods when applied to IGA.

This chapter is structured as follows: First, we propose $h-$multigrid methods in Section 5.1. The Poisson equation with $d = 1$ and $d = 2$ is proposed for IGA discretizations of two model problems in Section 5.1.1 as our first model problem. Thus, the Schwarz methods mentioned in Section 3.3 are proposed as smoothers in order to achieve robustness with respect the polynomial degree k, those are, the multiplicative Schwarz methods, additive Schwarz methods and Restrictive additive Schwarz

methods. Then, in Section 5.1.2 multigrid methods based on overlapping multiplicative Schwarz iterations are applied to the biharmonic equation in order to show their robustness for high-order scalar problems. Finally, we improve the $h-$multigrid method proposed for Poisson problem by introducing a two-level algorithm in Section 5.2. For this two-level method, again iterations of the multiplicative Schwarz smoothers are applied in the fine level.

5.1 h-multigrid methods

In this section, we propose robust $h-$multigrid methods for isogeometric discretizations of two model problems. In order to analyze their convergence, numerical results provided by LFA are shown together with the numerical results of our multigrid codes. As pointed out before, we consider the Schwarz methods introduced in Section 3.3 for the Poisson equation. Then, multigrid methods based on multiplicative Schwarz iterations are proposed for the biharmonic equation. For each problem, the corresponding LFA is carried out in order to design robust methods.

In order to perform the analysis of IGA discretizations, it is important to note that the location of each grid function is placed where the corresponding basis functions gets its highest value rather than guided by the knot lines. However, for IGA discretizations with maximum global smoothness C^{k-1} derived from uniform knot vectors these locations are equally spaced and the distance between grid functions coincides with the knot spans length. In order to illustrate this, in Figure 5.1 we show the values of unique knot lines and the locations of each grid function(•) considering the unit square domain and a spline space $\mathcal{S}^{2,2}_{1,1}(\mathcal{G}_h)$ with $h = 2^{-3}$. In this figure, we can observe that the location of grid functions coincide with the center of inner cells but not close to the boundary. However, special cases arise when intermediate global smoothnesses are considered, that is, the number of knot repetitions r satisfies $1 < r < k$.

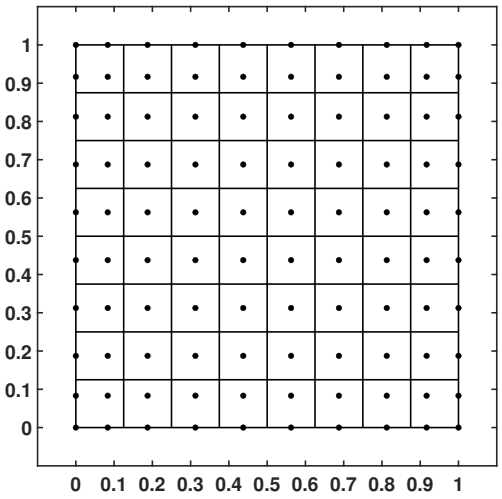


FIGURE 5.1: Location of B-spline grid functions
Unique values of knot lines and the locations of each grid function(•) considering the unit square domain and a spline space $\mathcal{S}^{2,2}_{1,1}(\mathcal{G}_h)$ with $h = 2^{-3}$.

5.1.1 Poisson equation

Here, we consider the two-point boundary value problem given in (2.5) as a model problem for the one-dimensional case. In addition, the Poisson equation described in (2.19) serves us as model problem with $d = 2$. For these problems, we propose a geometric $h-$multigrid based on multiplicative Schwarz smoothers. The choice of the block size depends on the spline degree k in order to maintain a robust performance of our solver. Thus, we can obtain a good performance by applying $V(1,0)-$cycles of the most appropriate multiplicative Schwarz method. In addition, we consider the canonical spline embedding operator as the prolongation operators and their adjoint for the restriction operators. Numerical experiments with our in-house codes are shown in the search of robust multigrid methods. First, the LFA results are shown providing a correct choice of the smoother for each k, that is, the number of unknowns involved in the blocks. Furthermore, in order to improve the performance of the methods, we decided to apply a coloured version of the Schwarz methods giving place to high efficient solvers.

Multiplicative Schwarz methods

As a starting point, we consider the two-point boundary value problem given in (2.5) with a right-hand side $f(x) = \pi^2 \sin(\pi x)$. We discretize this problem with B-splines holding maximum global smoothness $\mathcal{C}^{k-1}$. First, in Table (5.1) we provide the smoothing (ρ_{1g}), two-grid (ρ_{2g}) and three-grid (ρ_{3g}^V) convergence factors obtained from standard LFA (see Section (4.2)), considering one smoothing step of a Gauss-Seidel relaxation, together with the asymptotic convergence factors provided by the $W(1,0)-$ and $V(1,0)-$cycle multigrid codes (ρ_h^W and ρ_h^V, respectively). We provide these results for different values of k ranging from $k = 2$ to $k = 8$. We can observe a perfect match between the experimental factors and those predicted by standard LFA for both W- and V-cycles. Also, we can see that the performance of the multigrid method based on point-wise smoothers, as the Gauss-Seidel relaxation is, deteriorates when k gets bigger.

	Gauss-Seidel				
	ρ_{1g}	ρ_{2g}	ρ_h^W	ρ_{3g}^V	ρ_h^V
$k = 2$	0.31	0.19	0.19	0.19	0.19
$k = 3$	0.26	0.22	0.22	0.22	0.22
$k = 4$	0.38	0.38	0.38	0.38	0.38
$k = 5$	0.62	0.62	0.62	0.62	0.62
$k = 6$	0.79	0.79	0.80	0.79	0.80
$k = 7$	0.89	0.89	0.90	0.89	0.90
$k = 8$	0.99	0.99	0.96	0.99	0.96

TABLE 5.1: One-dimensional case: smoothing (ρ_{1g}), two-grid (ρ_{2g}) and three-grid (ρ_{3g}^V) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the W(1,0) and V(1,0) cycle multigrid codes (ρ_h^W and ρ_h^V, respectively), for different spline degrees k.

Given that the Gauss-Seidel smoother seems to be useless for the design of robust multigrid methods, we propose the use of three-, five- and seven-point multiplicative Schwarz methods as smoothers. Hence, in Table 5.2, we show the smoothing and three-grid convergence factors predicted by the analysis and the experimentally

obtained asymptotic convergence factors of a $V-$cycle multigrid with one pre- and no post-smoothing steps. These results are shown for the three considered multiplicative Schwarz smoothers and for different spline degrees from $k = 2$ to $k = 8$. We consider $V-$cycles because we have seen that their convergence rates are as those provided by $W-$cycles but with a lower computational cost. We observe in the table that the smoothing ability of the proposed three-point multiplicative Schwarz relaxation deteriorates when k becomes larger. This affects the three-grid convergence factor that also gets worse, showing the necessity of considering multiplicative Schwarz smoothers coupling more than three points.

	3p Schwarz			5p Schwarz			7p Schwarz		
	ρ_{1g}	ρ_{3g}^V	ρ_h^V	ρ_{1g}	ρ_{3g}^V	ρ_h^V	ρ_{1g}	ρ_{3g}^V	ρ_h^V
$k=2$	0.176	0.127	0.127	0.119	0.088	0.087	0.089	0.065	0.065
$k=3$	0.156	0.114	0.113	0.112	0.086	0.086	0.086	0.066	0.066
$k=4$	0.146	0.127	0.127	0.104	0.084	0.084	0.082	0.067	0.067
$k=5$	0.209	0.209	0.211	0.101	0.095	0.095	0.078	0.069	0.069
$k=6$	0.389	0.389	0.389	0.147	0.147	0.147	0.077	0.077	0.077
$k=7$	0.564	0.564	0.564	0.279	0.279	0.276	0.119	0.119	0.121
$k=8$	0.712	0.712	0.712	0.424	0.424	0.426	0.221	0.221	0.224

TABLE 5.2: One-dimensional case: smoothing (ρ_{1g}) and three-grid (ρ_{3g}^V) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the V(1,0)-cycle multigrid code (ρ_h^V), for different spline degrees k.

Notice that the different blocks within the multiplicative Schwarz smoothers can also be visited in different orderings, for instance, they can also be treated with some patterning scheme, yielding a multicolored version of these relaxation schemes. This type of approaches usually provides better convergence. In our case, we have done a comparison of the convergence rates of the considered multiplicative Schwarz smoothers and their coloured counterparts. In particular, we have considered a three-colour version of these relaxation procedures. In Figure 5.2, we compare the convergence factors of the multigrid based on the lexicographic and coloured multiplicative Schwarz smoothers for the one-dimensional case. By using dotted lines, we show the convergence rates provided by the proposed multigrid method based on the lexicographic three-, five-, and seven-point multiplicative Schwarz smoothers, for $k = 2, \ldots, 8$. By using continuous lines, we display the asymptotic convergence factors provided by a multigrid based on $V(1,0)-$cycles and the coloured versions of the considered three-, five- and seven-point multiplicative Schwarz smoothers. We can observe that the qualitative behavior of the method is as in the case for the non-coloured smoothers, but the convergence rates are much better. We will take this into account in order to set our smoothing strategy and therefore we will consider the coloured counterparts of the selected relaxations. Notice that, from Figure 5.2, it is observed that the convergence rates provided with the coloured multiplicative Schwarz smoothers are five times smaller than those provided with the lexicographic version of the smoothers, for the spline degrees for which they are chosen.

Now, we would like to compare the performance of the multigrid method based on different smoothers: a standard Gauss-Seidel and the considered multiplicative Schwarz relaxations with both lexicographic and coloured orderings. With this purpose, we fix the grid-size to $h = 2^{-18}$. The comparison is done in Table 5.3, where the number of V(1,0)-multigrid iterations and the CPU times necessary to reduce

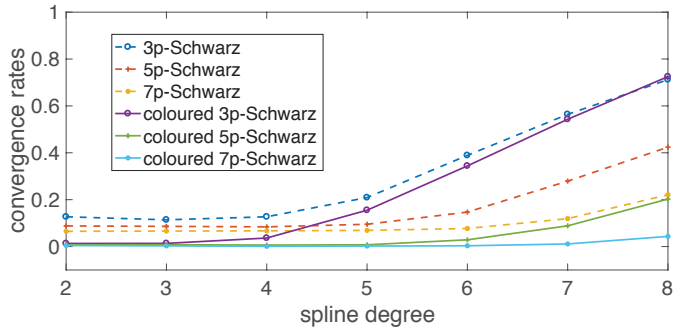


FIGURE 5.2: Convergence rates of the multigrid method based on overlapping multiplicative Schwarz smoothers for the three considered Schwarz relaxations and their coloured counterparts, and for spline degrees $k = 2, \ldots, 8$.

the initial residual in a factor of 10^{-8} are shown for different spline degrees for the considered approaches. All the methods have been implemented in our in-house Fortran code, and the numerical computations were carried out on a MacBook Pro with a Core i5 2.7 GHz and 8 GB RAM, running OS X 10.10 (Yosemite). Thus, we consider coloured multiplicative Schwarz smoothers with blocks composed of three points for the cases $k = 2, 3, 4$, five points for $k = 5, 6$ and seven points for $k = 7, 8$.

	GS		3p-Sch		c-3p-Sch		5p-Sch		c-5p-Sch		7p-Sch		c-7p-Sch	
k	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
2	10	0.28	9	0.36	5	0.22	7	0.43	5	0.32	7	0.62	4	0.36
3	10	0.33	7	0.33	5	0.25	7	0.49	4	0.28	6	0.58	4	0.40
4	16	0.56	8	0.41	5	0.27	7	0.52	4	0.32	6	0.64	4	0.44
5	29	1.24	10	0.55	8	0.48	7	0.57	4	0.35	6	0.71	4	0.47
6	55	2.29	14	0.84	12	0.71	7	0.63	5	0.45	6	0.83	4	0.52
7	106	4.98	20	1.24	19	1.20	10	0.96	6	0.59	6	0.79	4	0.53
8	201	10.44	31	2.05	38	2.52	13	1.26	9	0.88	8	1.15	5	0.68

TABLE 5.3: One-dimensional case. Number of $V(1,0)$ multigrid iterations (it) and computational time (cpu) necessary to reduce the initial residual in a factor of 10^{-8}, for different values of the spline degree k, using Gauss-Seidel (GS) smoother or the multiplicative Schwarz relaxations considered here: three-point (3p-Sch), five-point (5p-Sch), seven-point (7p-Sch), coloured three-point (c-3p-Sch), coloured five-point (c-5p-Sch), and coloured seven-point (c-7p-Sch) multiplicative Schwarz smoothers.

The next step is to see the scalability of the chosen strategy with respect to h and k. We apply the proposed $V(1,0)$ multigrid cycle based on the coloured multiplicative Schwarz smoother for solving the linear system. Table 5.4 shows the number of iterations needed to reduce the initial residual by a factor of 10^{-8} for several mesh sizes $h = 2^{-\ell}$ with $\ell = 16, \ldots, 20$, and different spline degrees $k = 2, \ldots, 8$. We also report the computational time (in seconds) needed for solving the linear system by the proposed multigrid method. We observe that our approach shows an excellent convergence with respect to the mesh refinement, for all the polynomial orders. But overall, we would like to remark the robustness of the method with respect to the spline degree. In all cases, only four or five $V(1,0)-$cycles are needed to reach the stopping criterium, independently of the mesh size and the polynomial order.

Regarding the computational cost, we observe a very good scaling, not only with respect to h but also with respect to k.

	Color 3p Schwarz						Color 5p Schwarz				Color 7p Schwarz			
	$k=2$		$k=3$		$k=4$		$k=5$		$k=6$		$k=7$		$k=8$	
ℓ	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
16	5	0.07	5	0.07	5	0.08	4	0.11	5	0.13	4	0.15	5	0.19
17	5	0.13	5	0.15	5	0.15	4	0.18	5	0.24	4	0.28	5	0.36
18	5	0.22	5	0.25	5	0.27	4	0.35	5	0.45	4	0.53	5	0.68
19	5	0.43	5	0.47	5	0.51	4	0.66	5	0.88	4	1.07	5	1.35
20	5	0.82	5	0.91	5	0.99	4	1.28	5	1.70	4	2.08	5	2.69

TABLE 5.4: One-dimensional test problem. Number of $V(1,0)$ multigrid iterations (it) and computational time (cpu) necessary to reduce the initial residual in a factor of 10^{-8}, for different grid-sizes h and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k.

Now, let us focus on the bidimensional Poisson problem defined in (2.19) with $\Omega = (0,1)^2$ and dirichlet boundary conditions in $\partial\Omega$. Since the problem is solved in the parametric domain, we choose the basis functions as B-splines. For this problem, we consider square blocks centered on each unknown by extending the previous one-dimensional approaches to each spatial direction. More concretely, $9(3\times 3)-$ $25(5\times 5)-$ and $49(7\times 7)-$point multiplicative Schwarz smoothers are considered. Due to the tensor-product structure of IGA discretizations, transfer operators can be obtained by means of kronecker product of their corresponding one-dimensional version. For the LFA of the two-dimensional problem, the analysis results are compared to the ones provided by our multigrid codes on a square domain. Thus, in Table 5.5, the three-grid convergence factors predicted by LFA are shown together with the corresponding asymptotic convergence factors calculated by using the proposed $V(1,0)$ multigrid method. A lexicografic Gauss-Seidel smoother and the multiplicative Schwarz relaxations with blocks of $9(3\times 3)$, $25(5\times 5)$ and $49(7\times 7)$ unknowns (9p Schwarz, 25p Schwarz and 49p Schwarz, respectively) are considered. The results are presented again for different spline degrees ranging from $k=2$ to $k=8$. The good match between the estimates predicted by LFA and the real asymptotic convergence factors is shown in the table. It is also observed that the bad behavior of the multigrid based on Gauss-Seidel smoother, which was seen in the one-dimensional results, is even more remarkable in this two-dimensional case. Finally, it can be seen that, given a spline degree k, a multiplicative Schwarz smoother which provides a multigrid method with an efficient performance can be found.

At this point, we have demonstrated that the local Fourier analysis is a very useful tool to obtain information about the performance of multigrid for IGA since it predicts very accurately the convergence of the method. This is very interesting since for a fixed spline degree k, we can choose the appropriate number of points in the blocks to construct an efficient multiplicative Schwarz smoother. As we did in the one-dimensional case, we propose to use the colored version of the Schwarz methods. Given that we considered 3 colors in the one-dimensional case, 9 different colors are considered for the two-dimensional case (3 colors per spatial dimension). Now, we test the performance of the geometric multigrid solver based on overlapping multiplicative Schwarz smoothers with a right-hand side

$$f(x) = 2\pi^2 \sin(\pi x)\sin(\pi y). \tag{5.1}$$

	Gauss-Seidel		9p Schwarz		25p Schwarz		49p Schwarz	
	ρ_{3g}^V	ρ_h^V	ρ_{3g}^V	ρ_h^V	ρ_{3g}^V	ρ_h^V	ρ_{3g}^V	ρ_h^V
$k=2$	0.510	0.510	0.099	0.099	0.067	0.066	0.050	0.050
$k=3$	0.827	0.830	0.212	0.214	0.065	0.067	0.052	0.051
$k=4$	0.954	0.955	0.452	0.455	0.127	0.145	0.051	0.053
$k=5$	0.990	0.990	0.701	0.703	0.260	0.262	0.098	0.119
$k=6$	0.999	0.999	0.870	0.872	0.447	0.440	0.190	0.165
$k=7$	0.999	0.999	0.947	0.955	0.650	0.657	0.317	0.325
$k=8$	0.999	0.999	0.985	0.982	0.816	0.807	0.473	0.440

TABLE 5.5: Two-dimensional case: Three-grid (ρ_{3g}^V) convergence factors predicted by LFA together with the asymptotic convergence factors provided by the V(1,0)-cycle multigrid code (ρ_h^V), for different spline degrees k.

Hence, in Table 5.6, we show the behavior of the multigrid method based on different smoothers: Gauss-Seidel and lexicographic/coloured multiplicative Schwarz relaxations. For a mesh of size 512×512, we display the number of V(1,0)-multigrid iterations together with the CPU times necessary to reduce the initial residual in a factor of 10^{-8}. Again, the bad behavior of the Gauss-Seidel based multigrid method due to the presence of many small eigenvalues associated with oscillatory components of the error, is clearly observed. There are some cases, indicated with the symbol $-$, for which more than 500 iterations would be needed for convergence. For the optimal smoothing strategy, we choose the coloured multiplicative Schwarz relaxations with blocks of size 3×3 for the cases $k = 2, 3, 4$, blocks of size 5×5 for the cases $k = 5, 6$ and blocks of size 7×7 for splines degree $k = 7, 8$. We now apply this proposed strategy for different grids with mesh-sizes from 128×128 to 1024×1024 and various spline degrees ranging from $k = 2$ to $k = 8$ in order to show the good scalability of the solver. Again, we use $V-$cycles with one pre-smoothing step and no post-smoothing steps. The number of iterations needed to reduce the initial residual by a factor of 10^{-8} together with the computational time (in seconds) are given in Table 5.7. We observe that the iteration numbers are robust with respect to both the size of the grid h and the spline degree k. Moreover, we see that the number of iterations is similar to those reported in the one-dimensional case. We emphasize that a small number of $V(1,0)-$cycles are needed to reach the stopping criterium, independently of h and k. Just like in the one-dimensional case, we can conclude that the multigrid method based on an appropriate multiplicative Schwarz smoother provides an efficient and robust solver for B-spline isogeometric discretizations.

The next experiment demonstrates the efficiency and robustness of the proposed multigrid method based on multiplicative Schwarz iterations when dealing with a nontrivial geometry. Hence, we take as domain the quarter of an annulus,

$$\Omega = \{(x, y) \in \mathbb{R}^2 \,|\, r^2 \leq x^2 + y^2 \leq R^2, x, y \geq 0\},$$

with $r = 0.3$ and $R = 0.5$. We consider the 2D Poisson problem (2.19) in such domain with homogeneous Dirichlet boundary conditions and $f(x, y)$ is such that the exact solution is

$$u(x, y) = \sin(\pi x)\sin(\pi y)(x^2 + y^2 - r^2)(x^2 + y^2 - R^2).$$

The geometry of the computational domain is described exactly by quadratic C^1 NURBS and depicted in Figure 5.3.

	GS		9p-Sch		c-9p-Sch		25p-Sch		c-25p-Sch		49p-Sch		c-49p-Sch	
k	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
2	20	1.10	8	1.29	4	0.68	7	4.89	4	2.88	6	14.21	3	7.46
3	57	4.90	8	2.13	4	0.99	6	5.60	4	4.02	6	17.80	3	9.30
4	166	21.04	14	4.66	7	2.40	7	8.38	4	5.08	6	22.40	4	15.56
5	492	96.03	19	17.26	17	10.40	9	17.83	4	6.09	6	39.96	3	14.69
6	–		31	21.53	40	28.40	11	20.01	5	9.33	7	39.90	3	16.06
7	–		60	48.04	112	98.70	15	34.32	9	21.80	9	60.44	3	17.15
8	–		88	105.62	254	310.55	22	71.19	15	48.87	11	88.17	4	26.01

TABLE 5.6: Two-dimensional case. Number of $V(1,0)$ multigrid iterations (it) and computational time (cpu) necessary to reduce the initial residual in a factor of 10^{-8}, for different values of the spline degree k, using Gauss-Seidel (GS) smoother or the multiplicative Schwarz relaxations considered here: three-point (9p-Sch), five-point (25p-Sch), seven-point (49p-Sch), coloured three-point (c-9p-Sch), coloured five-point (c-25p-Sch), and coloured seven-point (c-49p-Sch) multiplicative Schwarz smoothers.

	Color 9p Schwarz						Color 25p Schwarz				Color 49p Schwarz			
	$k=2$		$k=3$		$k=4$		$k=5$		$k=6$		$k=7$		$k=8$	
Grid	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
128^2	4	0.06	4	0.10	7	0.22	4	0.62	5	0.99	3	2.22	4	3.35
256^2	4	0.21	4	0.29	7	0.69	4	1.40	5	2.86	3	6.03	4	9.24
512^2	4	0.68	4	0.99	7	2.40	4	6.09	5	9.33	3	17.15	4	26.01
1024^2	4	2.60	4	3.80	7	9.13	3	16.49	5	33.05	3	53.69	4	80.74

TABLE 5.7: Two-dimensional case. Number of $V(1,0)$ multigrid iterations (it) and computational time (cpu) necessary to reduce the initial residual in a factor of 10^{-8}, for different grid-sizes and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k.

To discretize this problem, we use NURBS of degree $k = 2, \ldots, 8$ with maximal smoothness $\mathcal{C}^{k-1}$. We solve the corresponding linear systems using $V(1,0)-$ cycles. If Gauss-Seidel smoother is considered, the bad behavior of the multigrid method reported in the previous experiments is also observed here. More concretely, for $k = 2$ around 50 iterations are required to achieve the desired convergence, about 70 iterations are needed when $k = 3$ and for $k \geq 4$ the resulting number of iterations is bigger than 200. We consider then the coloured multiplicative Schwarz smoother used for the square domain. The size of the blocks of the Schwarz smoother depends on k, and they are chosen following the same strategy than in the previous experiment. The numbers of iterations and CPU times needed to reach the stopping criterium, for various degrees k and for different mesh sizes, are reported in Table 5.8. We observe an excellent performance, obtaining similar results to those of the example with the parametric domain. We can conclude that the proposed solver is robust with respect to this geometry transformation, having a great potential for solving problems in more complicated multi-patch geometries, which is subject of future research.

Additive Schwarz methods

In this section, we focus on the use of additive Schwarz methods as smoothers for multigrid methods. However, a standard LFA is not suitable for the analysis of

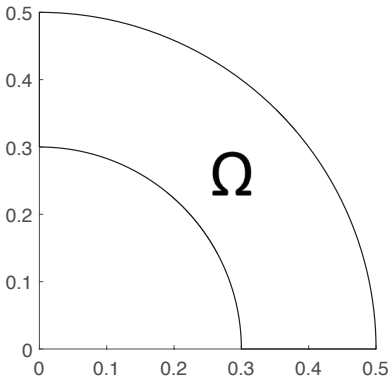


FIGURE 5.3: Computational domain for the Poisson problem on a quarter of an annulus with $r = 0.3$ and $R = 0.5$.

	Color 9p Schwarz						Color 25p Schwarz				Color 49p Schwarz			
	$k=2$		$k=3$		$k=4$		$k=5$		$k=6$		$k=7$		$k=8$	
Grid	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu	it	cpu
32^2	4	0.02	4	0.02	8	0.05	4	0.13	5	0.20	3	0.65	4	0.72
64^2	4	0.03	4	0.04	7	0.08	4	0.30	5	0.46	3	1.28	5	2.15
128^2	4	0.07	4	0.10	8	0.25	4	0.96	6	1.42	3	3.70	5	7.01
256^2	4	0.20	4	0.30	7	0.74	4	3.44	6	5.01	3	13.13	5	23.75

TABLE 5.8: Quarter of an annulus problem. Number of $V(1,0)$ multigrid iterations (it) and computational time (cpu) necessary to reduce the initial residual in a factor of 10^{-8}, for different grid-sizes and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k.

these smoothers. Thus, we carry out a LFA based on the infinite subgrids approach in this section. It is not surprising that multiplicative Schwarz smoothers provide better convergence rates than their additive counterparts. However, the efficiency of additive Schwarz methods lies on its parallelizability while they still provide good asymptotic convergence factors.

Thus, let us consider the Poisson problem (2.5) discretized with B-splines holding maximal global smoothness. As a starting point, let us consider the three-, five- and seven-point additive Schwarz methods as smoothers with maximum overlapping among the blocks. Therefore, it looks natural to apply weights $\omega = 1/b$ since each unknown is updated b times (b is the size of the blocks). Hence, we show in Table 5.9 the two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA applied to IGA discretizations holding maximum global continuity $\mathcal{C}^{k-1}$ together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on one smoothing step of additive Schwarz iterations, with a mesh size $h = 2^{-11}$ and different spline degrees k.

We would like to point out the good match among the asymptotic convergence factors provided by the analysis and the ones obtained by means of our numerical experiments. However, these asymptotic convergence factors suggest that by using additive Schwarz smoothers the block size should be more enlarged in order to achieve robustness with respect the polynomial degree. The convergence rates of these methods is similar if we consider FEM discretizations applied to the same

	3p ad Schwarz $\omega = 1/3$		5p ad Schwarz $\omega = 1/5$		7p ad Schwarz $\omega = 1/7$	
	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
$k = 1$	0.33	0.33	0.20	0.20	0.14	0.14
$k = 2$	0.37	0.37	0.19	0.19	0.16	0.16
$k = 3$	0.41	0.41	0.22	0.22	0.18	0.18
$k = 4$	0.45	0.45	0.30	0.30	0.22	0.22
$k = 5$	0.62	0.62	0.47	0.47	0.37	0.37
$k = 6$	0.76	0.77	0.62	0.62	0.50	0.50
$k = 7$	0.86	0.86	0.74	0.74	0.62	0.63
$k = 8$	0.92	0.93	0.83	0.83	0.72	0.73

TABLE 5.9: 1D Poisson problem: Two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA applied to IGA discretizations holding maximum global continuity $\mathcal{C}^{k-1}$ together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on one smoothing step of additive Schwarz iterations, with a mesh size $h = 2^{-11}$ and different spline degrees k.

problem. In fact, we show in Table 5.10 the two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA applied to FEM discretizations together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on one smoothing step of additive Schwarz iterations, with a mesh size $h = 2^{-11}$ and different spline degrees k.

	3p ad Schwarz $\omega = 1/3$		5p ad Schwarz $\omega = 1/5$		7p ad Schwarz $\omega = 1/7$	
	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
$k = 1$	0.33	0.33	0.20	0.20	0.14	0.14
$k = 2$	0.24	0.23	0.13	0.13	0.11	0.10
$k = 3$	0.39	0.38	0.25	0.25	0.18	0.18
$k = 4$	0.33	0.36	0.22	0.23	0.16	0.17
$k = 5$	0.48	0.48	0.31	0.30	0.23	0.23
$k = 6$	0.69	0.69	0.44	0.44	0.31	0.31
$k = 7$	0.83	0.83	0.58	0.58	0.41	0.42
$k = 8$	0.92	0.92	0.72	0.72	0.52	0.52

TABLE 5.10: 1D Poisson problem: Two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA applied to FEM discretizations together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on one smoothing step of additive Schwarz iterations, with a mesh size $h = 2^{-11}$ and different spline degrees k.

Now, let us show the performance of additive Schwarz methods in the case $d = 2$ and let us check the utility of LFA in order to predict the asymptotic convergence factors too. Hence, let us consider the 2D Poisson equation:

$$\begin{array}{rcl} -\Delta u & = & f, \text{ in } \Omega, \\ u & = & 0, \text{ on } \partial\Omega, \end{array} \tag{5.2}$$

on the square domain $\Omega = (0,1)^2$. For this problem, we use again B-spline discretizations holding maximum global continuity and FEM discretizations. Thus, we suggest to apply a multigrid method based on V(1,1) cycles using additive Schwarz

methods as smoothers with maximum overlapping. In this problem, we consider $9(3 \times 3)$-point, $25(5 \times 5)$-point and $49(7 \times 7)$-point additive Schwarz smoothers. Moreover, the number of points considered in our LFA based on the infinite subgrids approach is $n_g = 16, 36, 64$ points respectively, see (4.63). The solution obtained at each local problem is multiplied by $\omega = 1/b$, where b the block-size of the corresponding smoother. Hence, in Table 5.11 we show the two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA (based on the infinite subgrids approach) applied to IGA discretizations holding maximum global continuity $\mathcal{C}^{k-1}$ together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on two smoothing steps of additive Schwarz iterations, for different spline degrees k.

	9p ad Schwarz $\omega = 1/9$		25p ad Schwarz $\omega = 1/25$		49p ad Schwarz $\omega = 1/49$	
	ρ_{2g}^V	ρ_h^V	ρ_{2g}^V	ρ_h^V	ρ_{2g}^V	ρ_h^V
$k = 1$	0.09	0.10	0.05	0.09	0.03	0.06
$k = 2$	0.24	0.23	0.12	0.12	0.07	0.07
$k = 3$	0.57	0.57	0.35	0.34	0.21	0.21
$k = 4$	0.80	0.81	0.60	0.59	0.41	0.41
$k = 5$	0.93	0.93	0.79	0.78	0.61	0.61
$k = 6$	0.97	0.98	0.90	0.89	0.76	0.77
$k = 7$	-	-	0.96	0.95	0.87	0.87
$k = 8$	-	-	0.98	0.98	0.93	0.93

TABLE 5.11: 2D Poisson problem: Two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA applied to IGA discretizations holding maximum global continuity $\mathcal{C}^{k-1}$ together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on two smoothing steps of additive Schwarz iterations, for different spline degrees k.

This match between the experimental factors and those predicted by the analysis proves again that LFA is a useful tool for IGA in the two-dimensional case. The asymptotic convergence factors show again that the size (and/or overlapping) of the blocks shall be enlarged since the convergence rates per iteration of additive Schwarz smoothers decay. Something similar happens with the case of FEM discretizations applied to the same problem. Hence, in Table 5.12 we show the two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA (based on the infinite subgrids approach) applied to FEM discretizations together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on two smoothing steps of additive Schwarz iterations, for different spline degrees k.

Hence, we have already seen that the LFA can predict the asymptotic convergence behavior of multigrid algorithms based on additive Schwarz methods. Now, we would like to improve the performance of additive Schwarz smoothers by using optimal weights ω. For this task, we start performing a LFA based on infinite subgrids but starting with linear discretizations of the 1D Poisson equation considering fixed weights. Our first choice of these weights is the natural weight $1/b$, with b denoting the number of blocks where each unknown appears. Thus, in Table 5.13 we show the two-grid asymptotic convergence factors applying a linear finite element discretization $p = 1$ for the 1D Poisson problem with $\nu = 1$ smoothing steps of additive Schwarz methods, block-size (b) ranging from 2 to 7, overlapping ov ranging from 1 to $b - 1$ and considering natural weights:

	9p ad Schwarz $\omega = 1/9$		25p ad Schwarz $\omega = 1/25$		49p ad Schwarz $\omega = 1/49$	
	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
$k = 1$	0.09	0.10	0.05	0.09	0.03	0.06
$k = 2$	0.21	0.22	0.10	0.11	0.06	0.06
$k = 3$	0.49	0.48	0.28	0.27	0.17	0.17
$k = 4$	0.71	0.72	0.45	0.45	0.29	0.29
$k = 5$	–	–	0.62	0.62	0.43	0.43
$k = 6$	–	–	0.78	0.76	0.56	0.56
$k = 7$	–	–	0.88	0.85	0.68	0.67
$k = 8$	–	–	0.96	0.96	0.79	0.78

TABLE 5.12: 2D Poisson problem: Two-grid (ρ_{2g}) asymptotic convergence factors predicted by LFA (based on the infinite subgrids approach) applied to FEM discretizations together with the asymptotic convergence factors provided by our multigrid code (ρ_h) based on two smoothing steps of additive Schwarz iterations, for different spline degrees k.

	Block-size (b)					
ov	2	3	4	5	6	7
1	0.33	0.99	0.40	0.50	0.43	0.50
2	-	0.33	0.20	0.50	0.29	0.25
3	-	-	0.20	0.50	0.21	0.99
4	-	-	-	0.20	0.14	0.25
5	-	-	-	-	0.14	0.33
6	-	-	-	-	-	0.14

TABLE 5.13: Linear finite element discretization $p = 1$ for the 1D Poisson problem. Two-grid asymptotic convergence factors applying $\nu = 1$ smoothing steps of an additive Schwarz method. Block-size (b) ranges from 2 to 7 and the size of the overlapping ov from 1 to $b - 1$. Natural weights are taken in the smoother.

In the previous table, we observe that a bigger block-size or overlapping among blocks do not always ensure us a better asymptotic convergence factor. However, this fact might take place because we are not taking the optimal weight for each case. At this point, our LFA based on infinite subgrids allows us to perform an analysis to find the optimal weights. Depending on the block-sizes and overlappings, we found out that our search lies on two parameters. In Table 5.14 we show the optimal convergence factors for the previous cases together with the corresponding optimal weights between brackets:

The results obtained in Table 5.14 shows that even using the optimal weights, the best choice of block-size and overlapping for linear discretization of Poisson problem is not straightforward. Again, a bigger block-size or overlapping do not guarantee a better asymptotic convergence factor. Among the previous cases, the most efficient additive Schwarz method for $k = 1$ seems to be block-size 2 with overlapping 1 between the blocks. Furthermore, with a deeper search of the optimal weight, an asymptotic factor of $\rho_{2g} = 8e - 09$ is obtained with weights $(0.375, 0.375)$.

Restrictive additive Schwarz methods can be considered as a particular case of additive Schwarz methods that avoid communication between processors. Thus, we are interested on comparing them with their non-restrictive counterparts. Hence,

	Block-size					
ov	2	3	4	5	6	7
1	0.01 (0.38)	0.15 (0.57)	0.10 (0.55,0.77)	0.07 (0.7,0.71)	0.18 (0.49,0.82)	0.20 (0.8)
2	-	0.08 (0.27)	0.01 (0.42)	0.08 (0.47,0.61)	0.04 (0.51,0.78)	0.12 (0.45,0.89)
3	-	-	0.01 (0.21)	0.07 (0.35)	0.10 (0.45)	0.07 (0.53)
4	-	-	-	0.04 (0.17)	0.01 (0.29)	0.06 (0.35,0.42)
5	-	-	-	-	0.03 (0.15)	0.04 (0.26)
6	-	-	-	-	-	0.04 (0.13)

TABLE 5.14: Linear finite element discretization $k = 1$ for the 1D Poisson problem. Asymptotic convergence factors and optimal weights, between brackets, provided by our two-grid analysis by using one iteration of additive Schwarz methods with optimal weights. Block-size (b) ranges from 2 to 7 and the size of the overlapping (ov) from 1 to $b - 1$.

we consider restrictive additive Schwarz methods for the 1D Poisson equation discretized with linear basis functions and natural weights $\omega = 1$. In addition, we performed a LFA based on infinite subgrids for this case. In Table 5.15 we show the two-grid asymptotic convergence factors applying $\nu = 1$ smoothing steps of the restrictive additive Schwarz method with natural weights, block-sizes ranging from 2 to 7 and all the possible overlappings among the blocks.

	Block-size					
ov	2	3	4	5	6	7
1	0.75	0.99	0.40	0.50	0.43	0.50
2	-	0.99	0.60	0.50	0.57	0.44
3	-	-	0.87	0.87	0.29	0.99
4	-	-	-	1.00	0.71	0.44
5	-	-	-	-	0.93	0.99
6	-	-	-	-	-	0.99

TABLE 5.15: Linear finite element discretization $k = 1$ for the 1D Poisson problem. Two-grid asymptotic convergence factors applying $\nu = 1$ smoothing steps of the restrictive additive Schwarz method with natural weights. Block-size (b) ranges from 2 to 7 and the size of the overlapping ov from 1 to $b - 1$.

We observe that LFA predicts a very poor performance of restrictive additive Schwarz methods as smoothers in some cases. Fortunately, our LFA based on infinite subgrids do not fall behind regarding the search of optimal weights for restrictive cases. Hence, in Table 5.16 we show the two-grid asymptotic convergence factors applying $\nu = 1$ smoothing steps of the restrictive additive Schwarz method with optimal weights, block-sizes ranging from 2 to 7 and all the possible overlappings among the blocks.

We can observe a significant improvement of the asymptotic convergence factors when appropiate, non-trivial weights are applied. Once again, better asymptotic convergence factors are not directly related to bigger block-sizes or overlappings among blocks. The potential of our analysis allows us to extend LFA to other polynomial degrees. In practice, for FEM discretizations, additive Schwarz methods are often applied with overlapping 1 among the blocks and an usual strategy is to make a splitting of the grid such that each block contains all the basis functions with support in the corresponding element. That is, the size of the blocks is chosen as $k + 1$ for FEM. Hence, we consider this approach for additive Schwarz

	Block-size					
ov	2	3	4	5	6	7
1	0.45 (0.6)	0.34 (0.67)	0.17 (0.83)	0.20 (0.8)	0.18 (0.82)	0.20 (0.8)
2	-	0.37 (0.68)	0.15 (0.71)	0.20 (0.8)	0.23 (0.78)	0.18 (0.82)
3	-	-	0.43 (0.71)	0.34 (0.66)	0.16 (0.84)	0.34 (0.67)
4	-	-	-	0.36 (0.66)	0.20 (0.7)	0.18 (0.82)
5	-	-	-	-	0.40 (0.7)	0.34 (0.66)
6	-	-	-	-	-	0.35 (0.66)

TABLE 5.16: Linear finite element discretization $k = 1$ for the 1D Poisson problem. Asymptotic convergence factors and optimal weights, between brackets, provided by our two-grid analysis by using one iteration of the restrictive additive Schwarz method with optimal weights. Block-size (b) ranges from 2 to 7 and the size of the overlapping (ov) from 1 to $b-1$.

methods and their restrictive counterparts with natural and optimal ones provided by our LFA. Thus, in Table 5.17 we show the asymptotic convergence factors predicted by our two-grid analysis by using one smoothing step of the element-based additive Schwarz method (AS), restrictive additive Schwarz method (RAS) and their versions with optimal weights (AS opt and RAS opt), for polynomial degrees ranging from $k = 2, \ldots, 8$. Note that in the case of additive Schwarz methods with optimal weights, these are shown between parenthesis. In this table, we observe that, aside from the quadratic case with a poor performance, the AS and RAS methods provide satisfying and robust convergence rates with respect the spline degree. Besides it, the asymptotic convergence factors are improved with AS opt and RAS opt and their performance is even more robust than by using natural weights.

	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$
AS	0.50	0.10	0.17	0.11	0.17	0.12	0.17
RAS	0.50	0.12	0.15	0.11	0.17	0.12	0.16
AS opt	0.13 (0.62,0.75)	0.07 (0.94,0.01)	0.08 (0.44,0.92)	0.05 (0.46,0.95)	0.08 (0.46,0.92)	0.06 (0.47,0.94)	0.08 (0.46,0.92)
RAS opt	0.16(1.17,0.77)	0.06(0.92,0.95)	0.07(0.94,0.93)	0.05(0.91,0.95)	0.08 (0.92,0.92)	0.06 (0.94,0.94)	0.08(0.91,0.92)

TABLE 5.17: High order FEM discretizations for the 1D Poisson problem: Asymptotic convergence factors predicted by our two-grid analysis by using one smoothing step of element-based the additive Schwarz method (AS), restrictive additive Schwarz method (RAS) and their versions with optimal weights (AS opt and RAS opt), for polynomial degrees ranging from $k = 2, \ldots, 8$.

For the two-dimensional case, we consider the same element-based smoothing strategy using additive Schwarz methods obtained by tensorization. Thus, the size of local linear systems will be $(k+1)^2$ and the overlapping among the blocks in both directions is still one. Hence, in Table 5.18 we show the asymptotic convergence factors predicted by the two-grid analysis by using two smoothing steps of the two-dimensional version of the element-based additive Schwarz methods considered for the one-dimensional case and polynomial degrees ranging from $k = 1, \ldots, 8$.

We can observe that, although there is a slight deterioration of the asymptotic convergence factors with respect to the spline degree k, these approaches provide satisfying convergence rates. Element-based additive Schwarz methods show a similar performance to their restrictive counterparts. Hence, the restrictive additive Schwarz methods are more likely to be used since they avoid communication among

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$
AS	0.14	0.15	0.21	0.25	0.31	0.36	0.40	0.43
RAS	0.19	0.15	0.21	0.25	0.31	0.36	0.40	0.43
AS opt	0.08 (0.46)	0.10 (0.52,1.06)	0.15 (0.55,1.05)	0.19 (0.57,1.05)	0.25 (0.57,1.06)	0.30 (0.62,1.04)	0.34 (0.63,1.05)	0.38 (0.64,1.04)
RAS opt	0.10 (0.94)	0.11 (1.01,1.06)	0.15 (1.10,1.05)	0.19 (1.15,1.05)	0.25 (1.2,1.05)	0.30 (1.25,1.04)	0.34 (1.18,1.05)	0.38 (1.27,1.04)

TABLE 5.18: High order FEM discretizations for the 2D Poisson problem: Asymptotic convergence factors predicted by our two-grid analysis by using one smoothing step of element-based the additive Schwarz method (AS), restrictive additive Schwarz method (RAS) and their versions with optimal weights (AS opt and RAS opt), for polynomial degrees ranging from $k = 2, \ldots, 8$.

processors. In addition, the use of optimal weights only improves slightly the performance with natural weights. Then, optimal weights might not be worth to use in practice. Thus, we propose the restrictive additive Schwarz methods based on elements with minimum overlapping among the blocks and natural weights. Furthermore, one is interested in the number of smoothing steps in order to achieve a determined performance for the solver. Hence, in Table 5.19 we show the asymptotic convergence factors provided by LFA together with the asymptotic convergence factors obtained with our multigrid codes using $W(1,0)$, $W(1,1)$, $W(2,1)$ and $W(2,2)$ cycles of the element-based restrictive additive Schwarz methods with natural weights, for polynomial degrees ranging from $k = 1, \ldots, 8$:

k	$W(1,0)$		$W(1,1)$		$W(2,1)$		$W(2,2)$	
	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.41	0.41	0.19	0.18	0.10	0.09	0.03	0.03
2	0.39	0.40	0.15	0.15	0.06	0.06	0.02	0.02
3	0.46	0.45	0.21	0.21	0.09	0.09	0.04	0.04
4	0.50	0.50	0.25	0.25	0.13	0.13	0.06	0.06
5	0.56	0.56	0.31	0.31	0.18	0.18	0.10	0.10
6	0.60	0.60	0.36	0.36	0.21	0.21	0.13	0.13
7	0.63	0.63	0.40	0.40	0.26	0.25	0.16	0.16
8	0.66	0.66	0.43	0.43	0.28	0.28	0.19	0.19

TABLE 5.19: FEM discretizations of 2D Poisson equation: Asymptotic convergence factors provided by LFA together with the asymptotic convergence factors obtained with our multigrid codes using $W(1,0)$,$W(1,1)$,$W(2,1)$,$W(2,2)$ cycles of the element-based restrictive additive Schwarz methods with natural weights, for polynomial degrees ranging from $k = 1, \ldots, 8$.

Note that there is a very good match between the asymptotic convergence factors provided by LFA and the ones obtained with our multigrid codes. Also, the use of $W-$cycles provides asymptotic convergence factors that are more robust with respect to the spline degree k. We can reduce the computational cost of our solver by performing $V-$cycles instead of $W-$cycles. In Table 5.20, we show the asymptotic convergence factors provided by our multigrid codes applying $V-$cycles of the proposed smoothers for polynomial degrees ranging from $k = 1, \ldots, 8$.

As we can see, the asymptotic convergence factors are substantially similar to the ones obtained by applying $W-$cycles. Thus, we propose to use $V-$cycles of the element-based restrictive additive Schwarz methods with natural weights as optimal solver for FEM discretizations of the 2D Poisson equation. A further analysis of these smoothers would determine whether restrictive or non-restrictive additive

	$V(1,0)$	$V(1,1)$	$V(2,1)$	$V(2,2)$
k	ρ_h	ρ_h	ρ_h	ρ_h
1	0.40	0.19	0.09	0.03
2	0.40	0.15	0.06	0.02
3	0.45	0.20	0.09	0.04
4	0.50	0.25	0.13	0.06
5	0.56	0.31	0.18	0.10
6	0.60	0.36	0.21	0.13
7	0.63	0.40	0.25	0.16
8	0.66	0.43	0.28	0.19

TABLE 5.20: FEM discretizations of 2D Poisson equation: Asymptotic convergence factors provided by LFA together with the asymptotic convergence factors obtained with our multigrid codes using $V(1,0)$, $V(1,1)$, $V(2,1)$, $V(2,2)$ cycles of the element-based restrictive additive Schwarz methods with natural weights, for polynomial degrees ranging from $k = 1,\ldots,8$.

Schwarz methods should be applied to IGA discretizations holding maximal continuity. However, as commented for the additive Schwarz methods, this analysis is still an ongoing work.

5.1.2 Biharmonic equation

In this section, we propose the use of multigrid methods based on multiplicative Schwarz iterations as smoothers for the biharmonic equation. In order to set the most suitable size of the blocks as we did for the Poisson problem in the previous section, we carry out a LFA for this problem. Thus, we present some local Fourier analysis results starting with the one-dimensional biharmonic equation. In Table 5.21 we show the smoothing factor (ρ_{1g}) and three-grid convergence factors predicted by LFA for $W-$cycle (ρ_{3g}^W) and for $V-$cycle (ρ_{3g}^V), for different spline degrees varying from $k = 2$ to $k = 6$, when using the three-point multiplicative Schwarz smoother. Together with these results, we also display the asymptotic convergence factors provided by the $W(1,0)-$ and the $V(1,0)-$multigrid cycles (ρ_h^W and ρ_h^V, respectively). A perfect match between the theoretical and experimental factors is observed from the table, indicating the utility of this analysis for the design of multigrid methods for the isogeometric discretization of the one-dimenstional biharmonic equation. Moreover, we can see from the table that, opposite to the case of the Laplacian operator for which the multigrid based on the three-point multiplicative Schwarz smoother deteriorates when k gets bigger (see Pé de la Riva, Rodrigo, and Gaspar, 2019), in this case such a multigrid method provides a solver for the biharmonic equation which is robust with respect to the polynomial degree k. Finally, we want to remark that the three-grid convergence factors obtained for $V-$cycles are very close to those provided by using a three-grid analysis for $W-$cycles, and therefore, it is preferred the use of $V-$cycles.

Now, we extend the LFA to the two-dimensional case of the biharmonic equation. Hence, we consider the two-dimensional bilaplacian equation and analyze the behavior of multigrid for this problem. With this purpose, in Figure 5.4, we display the convergence factors predicted by LFA for different spline degrees k when multigrid based on Gauss-Seidel or $9(3\times 3)$- or $25(5\times 5)$-point multiplicative Schwarz smoothers is considered. It can be clearly seen that in the two-dimensional case, a

	ρ_{1g}	ρ_{3g}^W	ρ_h^W	ρ_{3g}^V	ρ_h^V
$k=2$	0.33	0.33	0.33	0.33	0.33
$k=3$	0.30	0.27	0.27	0.28	0.28
$k=4$	0.28	0.22	0.22	0.23	0.23
$k=5$	0.26	0.20	0.20	0.21	0.21
$k=6$	0.25	0.20	0.20	0.20	0.20

TABLE 5.21: One-dimensional biharmonic equation. Smoothing factor (ρ_{1g}) and three-grid convergence factors predicted by LFA for $W-$ and $V-$cycles (ρ_{3g}^W and ρ_{3g}^V, respectively) together with the asymptotic convergence factors provided by the $W(1,0)-$ and $V(1,0)-$multigrid cycles (ρ_h^W and ρ_h^V, respectively) by using the three-point multiplicative Schwarz relaxation, for different spline degrees k.

deterioration of the multigrid convergence appears when k becomes bigger and a Gauss-Seidel or a nine-point multiplicative Schwarz smoothers are considered. This behavior is similar to that observed for the laplacian equation in Pé de la Riva, Rodrigo, and Gaspar, 2019, and makes necessary the use of a bigger block of unknowns within the Schwarz relaxation iteration for large values of k. Thus, we also consider the twenty-five-point multiplicative Schwarz smoother which provides satisfactory results for the values of k considered here, as seen also in the picture. For bigger values of the spline degree, the presented LFA can be used for the choice of an adequate block-size within the multiplicative Schwarz smoother.

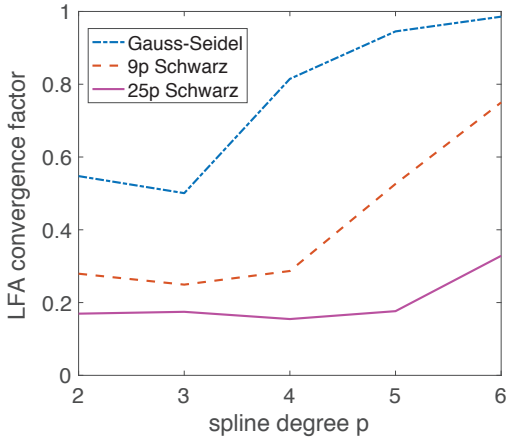


FIGURE 5.4: Two-dimensional biharmonic equation. LFA convergence factors for multigrid based on Gauss-Seidel, nine-point Schwarz and twenty-five-point Schwarz smoothers for different values of the spline degree k.

From the convergence factors provided by Figure 5.4, and taking into account the computational complexity of the proposed multiplicative Schwarz smoothers, we can define a strategy to obtain a multigrid method for the isogeometric discretization of the two-dimensional biharmonic problem which is robust with respect to the spline degree k. This strategy is fixed as considering the nine-point multiplicative Schwarz smoother within the multigrid method when $k \leq 4$, and the twenty-five-point counterpart of the Schwarz relaxation for $k = 5, 6$.

Once again, we propose an improvement of the strategy by applying the three-colour ordering per spatial direction as applied for the Poisson equation. By applying this smoothing strategy instead of the lexicographic order of the blocks, our multigrid algorithm provides better convergence factors.

At this point, we demonstrate the robustness and efficiency of the proposed multigrid solver based on multiplicative Schwarz smoothers. We consider two numerical experiments such that the one-dimensional and two-dimensional cases are included. For these numerical experiments we apply our solver based on the three-coloured version of the proposed smoothers since they yield better results. As it was observed in the local Fourier analysis, the good performance of V-cycles suggest their use in order to obtain a more efficient multigrid method. In addition, only one pre-smoothing and no post-smoothing steps are considered. The initial guess is taken as a random vector and the stoping criterion for the presented examples consists of reducing the initial residual by a factor of 10^{-8}. In the first numerical experiment we consider the following one-dimensional biharmonic problem:

$$\begin{cases} \dfrac{d^4u}{dx^4}(x) = \pi^4 \sin(\pi x), & x \in \Omega = (0,1), \\ u(0) = u(1) = \dfrac{du}{dx}(0) = \dfrac{du}{dx}(1) = 0. \end{cases}$$

Here, we consider the uniform and open knot vector $\Xi^k_{k-1,h}$. Thus, the problem is discretized with maximum continuity B-spline basis functions for different degrees ranging from $k = 2$ until $k = 6$. In Table 5.22, we show the number of iterations needed to reduce the initial residual by a factor of 10^{-8} for several mesh sizes $h = 2^{-\ell}$ with $\ell = 16, \ldots, 20$, and different spline degrees $k = 2, \ldots, 6$.

ℓ	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
16	8	5	4	5	5
17	7	5	5	5	5
18	8	5	4	5	5
19	7	5	5	5	5
20	8	5	4	5	5

TABLE 5.22: One-dimensional biharmonic equation. Number of $V(1,0)$ multigrid iterations necessary to reduce the initial residual in a factor of 10^{-8}, for different grid-sizes $h = 2^{-\ell}$ and for different values of the spline degree k, using the coloured three-point multiplicative Schwarz smoother.

It is clearly seen that the proposed multigrid method based on the coloured version of the three-point multiplicative Schwarz smoother is robust with respect to the grid size, as expected, and also with respect to the polynomial degree k. In addition, we observe that only few iterations are needed to reach the stopping criterion. In fact, in most of the cases only four or five $V(1,0)-$cycles are required for convergence.

For the second numerical experiment, let us consider the following biharmonic problem on a unit square domain:

$$\begin{cases} \Delta^2 u = 4\pi^4 \sin(\pi x)\sin(\pi y), & (x,y) \in \Omega = (0,1)^2, \\ u(x,y) = 0, & (x,y) \text{ on } \partial\Omega, \\ \nabla u \cdot n = 0, & (x,y) \text{ on } \partial\Omega. \end{cases}$$

We discretize this problem with B-spline basis functions and spline degrees ranging from $k = 2$ to $k = 6$. Moreover, the parameter space is constructed by means of the tensor product of two univariate, uniform and open knot vectors. Thus, the considered basis functions hold maximum continuity for this problem.

	Color 9p Schwarz			Color 25p Schwarz	
Grid	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$
32^2	5	3	3	3	4
64^2	6	3	5	3	4
128^2	7	4	5	3	5

TABLE 5.23: Two-dimensional biharmonic equation. Number of $V(1,0)$ multigrid iterations (it) necessary to reduce the initial residual in a factor of 10^{-8}, for different grid-sizes and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k.

In Table 5.23, we show the number of iterations required by our multigrid solver based on the three-color version of the nine- and twenty-five-point multiplicative Schwarz smoothers. It is observed again that a small number of $V(1,0)$ cycles is needed in order to reduce the initial residual by a factor of 10^{-8}. We also want to remark that the proposed multigrid solver is robust with respect to the spline degree k.

To sum up, geometric multigrid methods based on the proposed multiplicative Schwarz smoothers provide efficient and robust solvers with respect to k for solving the biharmonic equation discretized by using IGA. In the one dimensional case, this is achieved by using a three-colour three-point multiplicative Schwarz iteration as the smoothing operator within the multigrid method, which gives robust convergence factors for any value of k. For the two-dimensional case, however, the behavior of such a solver deteriorates when k becomes larger, and therefore, the size of the blocks of unknowns within the multiplicative Schwarz smoother has to be chosen appropriately for each spline degree. This choice is supported with the LFA introduced in Chapter 4 and we have proved the accuracy of this analysis at predicting the behavior of the multigrid algorithms applied to the biharmonic equation. Furthermore, the proposed coloured version of the multiplicative Schwarz smoothers improves the performance of our multigrid solvers and the use of $V-$cycles with only one pre-smoothing step and no post-smoothing steps is enough to obtain an excellent and robust convergence.

5.2 Two-level method for Poisson equation

In this section, we propose a two-level method for solving isogeometric discretizations of the Poisson problem with an arbitrary polynomial degree in an efficient and robust way. This two-level method considers the target polynomial degree on the fine level whereas the order of the approximation at the coarse level is as low as possible, dictated by the parametrization of the physical domain. More concretely, we will consider a linear/quadratic discretization on the coarse level. At this point, let us denote with k and k_{low} the polynomial orders of the discretization at the fine and coarse level respectively. Thus, the goal of our two-level algorithm is to solve the linear system:

$$A_k u_k = b_k, \tag{5.3}$$

where A_k denotes the isogeometric discretization of spline degree k. One iteration of our proposed two-level method consists of the following steps:

1. Apply ν_1 pre-smoothing steps of a suitable smoother S_k to the initial approximation u_k^0 for the fine level:
$$u_k^m = u_k^{m-1} + S_k(b_k - A_k u_k^{m-1}), \quad m = 1, \dots, \nu_1.$$

2. Compute the defect on the fine level $d_k = b_k - A_k u_k^{\nu_1}$ and restrict it to the coarse level by using the fine-to-coarse transfer operator
$$d_{k_{low}} = R_k^{k_{low}} d_k.$$

3. Compute the correction $e_{k_{low}}$ in the coarse level by solving the defect equation
$$A_{k_{low}} e_{k_{low}} = d_{k_{low}},$$
where $A_{k_{low}}$ denotes the isogeometric discretization of spline degree k_{low}.

4. Prolongate and update the correction to the fine level by means of the coarse-to-fine transfer operator
$$u_k^{\nu_1} = u_k^{\nu_1} + P_{k_{low}}^k e_{k_{low}}.$$

5. Apply ν_2 post-smoothing steps of the same smoother S_k to the current approximation:
$$u_k^{\nu_1+m} = u_k^{\nu_1+m-1} + S_k(b_k - A_k u_k^{\nu_1+m-1}), \quad m = 1, \dots, \nu_2.$$

As mentioned before, the performance of multigrid methods depend strongly on the choice of their components. Thus, we focus now on the choice of the smoother and transfer operators. In Section 5.1, we observed that h-multigrid methods based on multiplicative Schwarz methods provide robust convergence with respect to the spline degree k. Hence, we consider these smoothers for our two-level algorithm too. We suggest that only one pre-smoothing step of the adequate multiplicative Schwarz method is enough for this task. With respect to the transfer operators, one can follow the strategy shown for geometric multigrid methods in Chapter 3. However, the application of this strategy based on collocation points often leads to singular systems when spline spaces of different spline degrees are considered. This might be caused by the common support of their basis functions that makes impossible or very complicated to find suitable collocation points. Another possible cause is the fact that B-spline spaces of different spline degrees are not embedded. Thus, we propose to construct the transfer operators by means of a L^2 projection among spline spaces with polynomial degrees k and k_{low}, that is, V_h^k and $V_h^{k_{low}}$. Let us suppose that the solution on the fine level with spline degree k and mesh size h is given by

$$u_h^k = \sum_{j=1}^{n_h^k} u_j^k \varphi_j^k, \tag{5.4}$$

where $n_h^k = \dim V_h^k$, $\{\varphi_j^k\}_{j=1}^{n_h^k}$ are the basis functions of the approximation space V_h^k and $\mathbf{u}^k = (u_j^k)_{j=1}^{n_h^k}$ is the vector with the corresponding control variables. Then, the

restriction operator $R_k^{k_{low}} : V_h^k \longrightarrow V_h^{k_{low}}$ applied to the solution u_h^k can be expanded as a linear combination of the spline basis functions of $V_h^{k_{low}}$ as follows

$$R_k^{k_{low}} u_h^k = \sum_{j=1}^{n_h^{k_{low}}} u_j^{k_{low}} \varphi_j^{k_{low}}. \tag{5.5}$$

In order to obtain the influence between the coefficients $\{u_j^k\}_{j=1}^{n_h^k}$ and $\{u_j^{k_{low}}\}_{j=1}^{n_h^{k_{low}}}$, we test both the approximation on the fine level and its restricted term with every basis function spanning $V_h^{k_{low}}$. Hence, we obtain

$$\sum_{l=1}^{n_h^{k_{low}}} u_l^{k_{low}} \left(\varphi_l^{k_{low}}, \varphi_i^{k_{low}}\right) = \sum_{j=1}^{n_h^k} u_j^k \left(\varphi_j^k, \varphi_i^{k_{low}}\right), \quad \forall i = 1, \ldots, n_h^{p_{low}}. \tag{5.6}$$

This system can also be described as follows,

$$\mathbf{M}_{k_{low}}^{k_{low}} \mathbf{u}^{k_{low}} = \mathbf{M}_k^{k_{low}} \mathbf{u}^k, \tag{5.7}$$

where $\mathbf{u}^{k_{low}} = (u_j^{k_{low}})_{j=1}^{n_h^{k_{low}}}$ is the vector with the control variables for the approximation on $V_h^{k_{low}}$ and

$$\begin{aligned} \left(\mathbf{M}_{k_{low}}^{k_{low}}\right)_{i,j} &= \int_\Omega \varphi_i^{k_{low}} \varphi_j^{k_{low}} \mathrm{d}x, \\ \left(\mathbf{M}_k^{k_{low}}\right)_{i,j} &= \int_\Omega \varphi_i^{k_{low}} \varphi_j^k \mathrm{d}x. \end{aligned} \tag{5.8}$$

Therefore, the restriction operator is given by

$$R_k^{k_{low}} = \left(\mathbf{M}_{k_{low}}^{k_{low}}\right)^{-1} \mathbf{M}_k^{k_{low}}. \tag{5.9}$$

Moreover, the prolongation operator is taken as its adjoint, that is

$$P_{k_{low}}^k = \left(\mathbf{M}_k^{k_{low}}\right)^T \left(\mathbf{M}_{k_{low}}^{k_{low}}\right)^{-T}. \tag{5.10}$$

In both cases, it is desirable to approximate $\left(\mathbf{M}_{k_{low}}^{k_{low}}\right)^{-1}$ by row-sum lumping in order to avoid the computation of this inverse matrix exactly.

At the coarse level, the stiffness matrix is conveniently obtained by rediscretization. In the case of a Galerkin aproximation, the resulting matrix is obtained by

$$A_{k_{low}} = R_k^{k_{low}} A_k P_{k_{low}}^k. \tag{5.11}$$

However, the Galerkin approximation of $A_{k_{low}}$ may contain extra non-zero entries and it yields a similar bandwidth to the matrix of the fine level. To sum up, in Algorithm 6 we describe one iteration of our two-level algorithm. In this algorithm, we denote with S_k an appropiate multiplicative Schwarz method as smoother for isogeometric discretizations with spline degree k.

Although there is an open choice for the solver at the coarse level, instead of solving exactly the coarse problem, it can also be approximated by using a suitable iterative method. The goal of this approach is to reduce the computational

Algorithm 6 Two-level algorithm

$u_k^{m+1} =$ Two-level$(A_k, A_{k_{low}}, b_k, u_k^m, S_k, R_k^{k_{low}}, P_{k_{low}}^k)$

1: $u_k^{m+1} = u_k^m + S_k(b_k - A_k u_k^m)$ ▷ Apply one step of the multiplicative Schwarz method on the fine level.
2: $d_k = b_k - A_k u_k^m$ ▷ Compute the defect on the fine level.
3: $d_{k_{low}} = R_k^{k_{low}} d_k$ ▷ Restrict the defect on the coarse level
4: $A_{k_{low}} e_{k_{low}} = d_{k_{low}}$ ▷ Compute the correction $e_{k_{low}}$ in the coarse level by solving the defect equation.
5: $u^{m+1} = u^{m+1} + P_{k_{low}}^k e_{k_{low}}$ ▷ Prolongate and update the correction to the fine level.

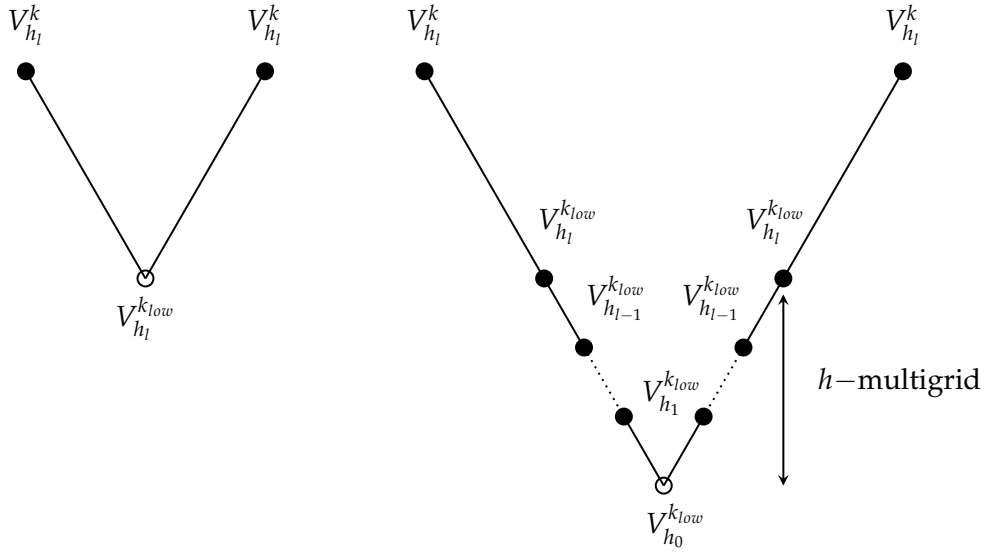


FIGURE 5.5: Schemes of the two-level method (left) and the two-level with an $h-$multigrid method for the approximation of the coarse level (right) pointing out the approximation spaces at each level.

cost at the coarse level. In addition, there are many efficient iterative solvers for linear/quadratic discretizations. For instance, an $h-$multigrid method can be performed to approximate the solution of the defect equation in the coarse level. Hence, we can define a grids sequence with mesh sizes $h_l = h, h_{l-1}, \ldots, h_1$, and h_0 for the coarsest grid. In Figure 5.5 we show the schemes of the two-level method (left) and the two-level with an $h-$multigrid method for the approximation of the coarse level (right) pointing out the approximation spaces at each level.

A further improvement of the algorithm can be achieved by using a more aggressive coarsening strategy. More concretely, we can take a discretization with k_{low} and double mesh size $2h$ as the coarse level. Thus, the computational cost can be reduced.

5.2.1 LFA for the two-level method

Now, we carry out a local Fourier analysis in order to design efficiently the components of the two-level algorithm presented above. As mentioned in there, an IGA discretization with spline degree k is considered in the fine level whereas the spline degree k_{low} for the discretization at the coarse level is as low as possible. We will denote the discrete operator in the fine and coarse level with A_k and $A_{k_{low}}$ respectively.

Moreover, let us denote with $R_k^{k_{low}}$, $P_{k_{low}}^k$ the restriction and prolongation operators between the fine and coarse levels. Thus, the error propagation operator of the two-level method is

$$\mathcal{L}_k^{k_{low}} = (I - P_{k_{low}}^k A_{k_{low}}^{-1} R_k^{k_{low}} A_k) S_k. \tag{5.12}$$

In the previous expression, S_k represents the smoothing operator which is applied within the two level method one time in the pre-smoothing step. Let us note that the Fourier modes are eigenfunctions of all the operators involved in the two level method. In this case, the transfer operators $R_k^{k_{low}}$, $P_{k_{low}}^k$ between levels do not couple Fourier modes unlike the inter-grid transfer operators within the standard $h-$multigrid method. Thus, the Fourier symbol of the error transfer operator for $\theta \in \Theta = (-\pi, \pi]^d$ is given by

$$\widetilde{\mathcal{L}}_k^{k_{low}}(\theta) = (\widetilde{I}(\theta) - \widetilde{P}_{k_{low}}^k(\theta)\widetilde{A}_{k_{low}}^{-1}(\theta)\widetilde{R}_k^{k_{low}}(\theta)\widetilde{A}_k(\theta))\widetilde{S}_k(\theta). \tag{5.13}$$

By using this Fourier symbol, the asymptotic convergence factor of the two-level method is estimated by the following expression:

$$\rho_{2g} = \sup_{\theta \in \Theta} |\widetilde{\mathcal{L}}_k^{k_{low}}(\theta)|. \tag{5.14}$$

Thus, we start our analysis by considering a linear discretization as the second level, that is, $k_{low} = 1$. In Table 5.24, the two-level convergence factors predicted by LFA, ρ_{2g}, are shown together with the asymptotic convergence factors, ρ_h, obtained numerically for different values of the spline degree k varying from $k = 2$ to $k = 8$. The asymptotic converge factors are obtained numerically by solving problem (2.19) with $\Omega = (0,1)^2$, a zero right-hand side, Dirichlet boundary conditions in $\partial\Omega$ and a random initial guess. We consider the $9-$point, $25-$point and $49-$point multiplicative Schwarz iterations at the first level. It can be seen from Table 5.24 that the factors predicted by LFA match very accurately the asymptotic convergence factors numerically obtained, and therefore the LFA results in a very useful tool to analyze the performance of the method.

	9p Schwarz		25p Schwarz		49p Schwarz	
	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
$k=2$	0.12	0.12	0.08	0.08	0.06	0.07
$k=3$	0.22	0.21	0.09	0.09	0.06	0.07
$k=4$	0.46	0.46	0.13	0.15	0.07	0.09
$k=5$	0.71	0.71	0.27	0.28	0.10	0.12
$k=6$	0.88	0.88	0.45	0.46	0.19	0.21
$k=7$	0.96	0.96	0.66	0.66	0.33	0.33
$k=8$	0.99	0.99	0.83	0.81	0.49	0.48

TABLE 5.24: Two-level (ρ_{2g}) convergence factors predicted by LFA together with the asymptotic convergence factors obtained numerically (ρ_h), for different values of the spline degree k. In this case, the second level is a linear discretization with the same mesh size h considered for the first level.

It is also observed from Table 5.24 that by choosing an appropriate multiplicative Schwarz smoother for each polynomial degree k, we obtain a robust solver with respect to k. This choice of the size of the blocks in the relaxation depending on the

spline degree is done by taking into account the two-grid convergence factors provided by the LFA, as well as the computational cost of the algorithm. In particular, we choose blocks of size 3×3 ($9-$point Schwarz smoother) for the cases $k = 2, 3, 4$, blocks of size 5×5 ($25-$point Schwarz smoother) for the cases $k = 5, 6$ and blocks of size 7×7 ($49-$point Schwarz smoother) for spline degree $k = 7, 8$.

As previously mentioned, we can also consider to approximate the solution of the defect equation at the coarse level by applying an $h-$multigrid. In order to analyze this improvement of the method, a three-grid analysis is required and at this time some frequencies are coupled in the transition from the second level with parameters (k_{low}, h) to the third level/grid with $(k_{low}, 2h)$. For this three-grid analysis, a different smoothing operator $S_{k_{low}}$ can be considered for the low-order IGA discretization. Hence, the error propagation matrix of the two-level algorithm improved with the $h-$multigrid approximation on the coarse grid is

$$M_{k,h}^{k_{low},2h} = (I - P_{k_{low}}^{k}(I - (M_{k_{low},h}^{k_{low},2h}))A_{k_{low}}^{-1} R_{k}^{k_{low}} A_k)S_k, \tag{5.15}$$

where $M_{k_{low},h}^{k_{low},2h}$ is the two-grid operator between the second and third levels, that is,

$$M_{k_{low},h}^{k_{low},2h} = S_{k_{low}}^{\nu_{2,2}}(I - P_{2h}^{h} A_{k_{low},2h}^{-1} R_{h}^{2h} A_{k_{low},h}) S_{k_{low}}^{\nu_{1,2}}, \tag{5.16}$$

with the inter-grid transfer operators R_{2h}^{h} and P_{h}^{2h} taken as the canonical spline embedding operator. In addition, $\nu_{1,2}$ and $\nu_{2,2}$ denote the number of pre- and post-smoothing steps of the smoother $S_{k_{low}}$ on the second level. As mentioned previously, in the transition from the second to the third level some Fourier modes are coupled. Hence, the Fourier components are split into high- and low-frequency components on $\mathcal{G}_h$ as performed during the two-grid analysis in Section 4.2.2. Based on the decomposition of the Fourier space in terms of the subspaces of $2h-$harmonics, the spectral radius of the three-grid operator can be computed as follows:

$$\rho_{3g} = \rho(M_{k,h}^{k_{low},2h}) = \sup_{\boldsymbol{\theta} \in \Theta_{2h}} \rho((M_{k,h}^{k_{low},2h})^{2g}(\theta)), \tag{5.17}$$

where $(M_{k,h}^{k_{low},2h})^{2g}$ denotes the matrix representation of $M_{k,h}^{k_{low},2h}$ with respect to the subspaces of $2h-$harmonics. Note that the construction of this matrix is straightforward by applying the same process performed during the two-grid analysis since there is not aliasing effect of frequencies in the transition from the first to the second level. Next, we present some LFA results in order to support the approach based on approximating the defect equation on the second level by means of one $h-$multigrid iteration. In this case, one single iteration of a $V(1,1)-$cycle using red-black Gauss-Seidel as smoother is considered to approximate the problem on the coarse level. Thus, in order to analyze such approximation, we need to use the three-grid local Fourier analysis introduced above. In Table 5.25, we show the three-grid convergence factors (ρ_{3g}) provided by LFA. One can observe that the predictions provided by the three-grid LFA match very well with the two-grid convergence factors predicted by the analysis for the two-level algorithm (with exact solve on the coarse level) shown in Table 5.24. This means that by doing such an approximation on the coarse grid we are not deteriorating the convergence obtained with the two-level method with exact solve on the coarse grid.

The previous version of the two-grid algorithm can be further improved by considering a more aggresive coarsening strategy. For instance, we propose to consider directly a grid size $2h$ in the second level. This strategy can be also understood as omitting any smoothing effect on the level with polynomial degree k_{low} and grid size

	9p Schwarz	25p Schwarz	49p Schwarz
$k = 2$	0.13	0.08	0.06
$k = 3$	0.21	0.09	0.07
$k = 4$	0.46	0.13	0.07
$k = 5$	0.71	0.27	0.10
$k = 6$	0.88	0.45	0.19
$k = 7$	0.96	0.66	0.32
$k = 8$	0.99	0.83	0.47

TABLE 5.25: Three-level (ρ_{3g}) convergence factors predicted by LFA, for different values of the spline degree k.

h. Thus, it is enough a two-grid analysis for this version of our two-level method. The corresponding error transfer operator is given by

$$\mathcal{L}_{k,h}^{k_{low},2h} = (I - P_{k_{low},2h}^{k,h} A_{k_{low},2h}^{-1} R_{k,h}^{k_{low},2h} A_k) S_k, \tag{5.18}$$

where the transfer operators $P_{k_{low},2h}^{k,h}$ and $R_{k,h}^{k_{low},2h}$ between the first and second level are obtained by composition of $P_{k_{low}}^{k}$, $R_{k}^{k_{low}}$ and the transfer operators P_{2h}^{h}, R_{h}^{2h}. From this expression, the asymptotic convergence factor of the improved two level method can be estimated by the following expression:

$$\rho_{2g}^{ag} = \rho(\mathcal{L}_{k,h}^{k_{low},2h}) = \sup_{\boldsymbol{\theta} \in \Theta_{2h}} \rho((\mathcal{L}_{k,h}^{k_{low},2h})^{2g}(\theta)), \tag{5.19}$$

where $(\mathcal{L}_{k,h}^{k_{low},2h})^{2g}$ is the matrix representation of $\mathcal{L}_{k,h}^{k_{low},2h}$ with respect the subspaces of $2h-$harmonics. Thus, we want to analyze the improvement of the two-level considering a more aggresive coarsening, that is, on the second level we keep considering a polynomial degree k_{low} but also the grid size h is doubled. Again, LFA is able to support this approach by using a two-grid analysis. In Table 5.26, the two-level convergence factors provided by this analysis are shown.

	9p Schwarz	25p Schwarz	49p Schwarz
$k = 2$	0.17	0.11	0.08
$k = 3$	0.21	0.12	0.09
$k = 4$	0.46	0.13	0.09
$k = 5$	0.71	0.27	0.10
$k = 6$	0.88	0.45	0.19
$k = 7$	0.96	0.66	0.32
$k = 8$	0.99	0.83	0.47

TABLE 5.26: Two-grid (ρ_{2g}^{ag}) convergence factors predicted by LFA for different values of the spline degree k, for the improved version of the algorithm.

Given that this last approach is more efficient and does not deteriorate the performance of the two-level method introduced before, this will be the strategy used for the numerical experiments.

5.2.2 Numerical results

Here, we show some numerical results in order to see the efficiency of the proposed solver by considering IGA discretizations with maximum global smoothness $\mathcal{C}^{k-1}$. For this purpose, we consider only one step of the coloured version of the multiplicative Schwarz method at the fine level. Instead of solving exactly at the coarse level, we follow the approximation strategy based on applying one iteration of an $h-$multigrid method. In addition, we use the aggresive coarsening approach and therefore a mesh size $H = 2h$ is considered on the second level with polynomial degree k_{low}. In these numerical experiments, the initial guess is taken as a random vector and the stopping criterion for our two-level solver is set to reduce the initial residual by a factor of 10^{-8}. All the methods have been implemented in our in-house Fortran codes, and the numerical computations have been carried out on an hp pavilion laptop 15-cs0008ns with a Core i7-8550U with 1,80 GHz and 16 GB RAM, running Windows 10.

In order to support the robustness and efficiency of the proposed two-level method, we have considered two different numerical experiments. In the first one, we apply our two-level method based on overlapping multiplicative Schwarz iterations to the two-dimensional problem (2.19) defined on a square domain $\Omega = (0,1)^2$ with Dirichlet boundary conditions on $\partial\Omega$ and the right-hand side given in (5.1). Hence, we discretize this problem with B-splines of different polynomial degrees ranging from $k = 2$ until $k = 8$ on the fine level whereas linear B-splines ($k_{low} = 1$) are considered for the second level. Thanks to the three-grid LFA, we are able to choose the size of the blocks for each k such that a robust two-level method is achieved. Hence, we choose blocks of size 3×3 for the cases $k = 2,3,4$, blocks of size 5×5 for the cases $k = 5,6$ and blocks of size 7×7 for spline degree $k = 7,8$.

	Color 9p Schwarz						Color 25p Schwarz				Color 49p Schwarz			
	$k=2$		$k=3$		$k=4$		$k=5$		$k=6$		$k=7$		$k=8$	
Grid	*it*	*cpu*	*it*	*cpu*	*it*	*cpu*	*it*	*cpu*	*it*	*cpu*	*it*	*cpu*	*it*	*cpu*
64^2	6	0.05	6	0.06	7	0.08	5	0.19	5	0.25	4	0.89	4	1.04
128^2	6	0.14	6	0.17	7	0.23	5	0.50	5	0.62	4	2.14	4	2.50
256^2	6	0.46	6	0.55	7	0.79	5	1.45	5	1.78	5	6.68	5	7.63
512^2	6	1.69	6	2.06	7	2.86	5	4.87	5	5.83	5	18.14	5	20.71
1024^2	6	6.83	6	8.34	7	11.21	6	21.22	5	21.12	5	55.76	5	62.86

TABLE 5.27: Square domain problem. Number of the proposed two-level method iterations (*it*) and computational time (*cpu*) necessary to reduce the initial residual in a factor of 10^{-8}, for different mesh-sizes h and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k.

In Table 5.27, we show the number of iterations (*it*) and the cpu time (*cpu*) in seconds needed to reach the stopping criterion for several mesh sizes and different spline degrees $k = 2,\ldots,8$. We observe that the iteration numbers are robust with respect to the size of the grid h and the spline degree k. With these results, we can conclude that our two-level method provides an efficient and robust solver for B-spline isogeometric discretizations.

For the second experiment, our goal is to apply the two-level method to a two-dimensional problem defined in a nontrivial geometry. Thus, we set as physical domain the same quarter of an annulus used for the numerical experiments shown in 5.1.1. Note that in order to construct this computational domain, the use of

quadratic NURBS basis funcions is required. Thus, we consider discretizations of degree $k = 3, \ldots, 8$ with maximal smoothness for the fine level whereas the quadratic discretization is used at the coarse level. In this case, we compare the performance of the proposed multigrid method (MG) with a two-level based on a direct solver (DS) at the second level. For this purpose, in Table 5.28 we show the number of iterations needed to reach the stopping criterion for several mesh sizes and different spline degrees $k = 3, \ldots, 8$. We observe that the use of the mentioned MG at the coarse level slightly increases the number of iterations for some cases. Finally, we conclude that our two-level method provides an efficient and robust solver also for NURBS discretizations.

	Color 9p Schwarz				Color 25p Schwarz				Color 49p Schwarz			
	$k=3$		$k=4$		$k=5$		$k=6$		$k=7$		$k=8$	
Grid	DS	MG	DS	MG	DS	MG	DS	MG	DS	MG	DS	MG
32^2	5	5	8	8	4	4	6	6	3	3	4	4
64^2	5	7	8	8	4	5	6	6	4	4	5	5
128^2	6	8	7	8	4	6	6	6	4	5	5	5
256^2	6	9	8	8	4	6	6	6	4	6	5	6

TABLE 5.28: Quarter annulus problem. Number of the proposed two-level method iterations (it) necessary to reduce the initial residual in a factor of 10^{-8}, for different mesh-sizes h and for different values of the spline degree k, using the most appropriate coloured multiplicative Schwarz smoother for each k, and considering both exact solver on the coarse grid (DS) or an $h-$multigrid approach on the coarse level (MG).

Thus, our purely algebraic two-level method solves the two-dimensional Poisson discretized with isogeometric discretizations of an arbitrary polynomial degree in an efficient and robust way. We would like to remark that our method acts as a black-box in which only one iteration of an appropriate multiplicative Schwarz method is applied on the fine level, and the coarse level can be exactly solved by using well-known techniques for solving linear and quadratic discretizations. The user can choose the preferred approach on the coarse level, but here we propose to approximate the coarse problem by using one single iteration of a suitable $h-$multigrid. In particular, we have applied one $V(1,1)-$cycle based on a red-black Gauss-Seidel smoother. An enhancement of the performance of the solver is obtained if we apply a standard coarsening strategy from the first to the second level by considering a grid of size h on the fine level and a coarse grid-size of $2h$. In addition, the good convergence results of the proposed method have been theoretically supported by two- and three-grid local Fourier analysis and also they are demonstrated by means of two numerical experiments.

Chapter 6

Isogeometric analysis of Poroelasticity

The theory of poroelasticity studies the interaction between the fluid flow and the deformation of a elastic porous material. In this context, a porous medium is composed of a skeleton or solid matrix and some pores that are fully or partially saturated by a fluid. When the solid is subjected to any stress, the matrix experiments some deformations which cause an increment on the pore fluid pressure. Consequently, the fluid flows from high-pressure to low-pressure locations. In order to describe these processes, poroelastic models must couple the mechanics of the solid matrix and fluid dynamics and the result is a coupled system of PDEs. The theory of poroelasticity was firstly developed by Terzaghi for a one-dimensional case with an incompressible fluid and pore fluid, see Terzaghi, 1923. This theory was extended and generalized to the three-dimensional case by Biot in Biot, 1941; Biot, 1955; Biot, 1956. Nowadays, Biot's model is applied in practice and it has received the attention of many authors in the literature, see Ehlers and Bluhm, 2002; Verruijt, 2018; Wang, 2001; Zienkiewicz, Chang, and Bettess, 1980. Nowadays, poroelasticity theory is essential in many scientific and engineering areas such as hydrology, geomechanics, reservoir engineering, biomechanics, etc.

In this chapter, we consider the quasi-static Biot's model for soil consolidation process. Our goal is to discretize the corresponding PDE system by using IGA. In order to solve the discrete problem, both decoupled and monolithic (multigrid) solvers are proposed. To design these latter, we make use of LFA in order to analyze different types of smoothers. This chapter is structured as follows: In Section 6.1, we introduce the governing equations for Biot's poroelastic model and the two-field formulation is proposed. Secondly, the variational formulation and isogeometric discretization of Biot's model is given in Section 6.2, including the proposal of a new stabilization scheme in order to eliminate the spurious numerical oscillations frequently appearing in the pressure approximation. Then, we comment on the different types of solvers for the poroelastic equations. Decoupled solvers are described in Section 6.3 with special focus on the fixed-stress split method and a decoupled solution method proposed by us. In order to support the stability and convergence of our iterative decoupled method with IGA discretizations, a Von Neumann analysis is presented in 6.3.3. In Section 6.4, monolithic solvers are described with special attention devoted to monolithic multigrid methods. Finally, we give a description of coupled smoothers in 6.4.1 and decoupled smoothers in 6.4.2 together with their corresponding LFA results and some numerical experiments.

6.1 Biot's model

In this section, we describe the constitutive equations for poroelastic Biot's model. Among the basic principles of poroelasticity, we find the equilibrium, deformation and motion equations of the porous medium. In addition, mass conservation laws of the solid matrix and the pore fluid and momentum conservation equations for the fluid flow must be considered. Hence, we proceed to formulate these equations one by one in order to construct the Biot's model.

6.1.1 Mass conservation

Given a reference elementary volume (such as an infinitesimal cube), the mass conservation states that the difference between the inflow and outflow of mass must be equal to the total change of mass. Thus, let us consider an infinitesimal cube with dimensions Δx, Δy and Δz. We denote the mass flux of the fluid with $\boldsymbol{q} = (q_1, q_2, q_3)$ which represents the rate of mass flow per unit area per unit time. Hence, the total inflow on each direction during a time interval Δt is given by

$$\text{Total inflow} \begin{cases} x-direction: & q_1(x)\Delta y\Delta z\Delta t, \\ y-direction: & q_2(y)\Delta x\Delta z\Delta t, \\ z-direction: & q_3(z)\Delta x\Delta y\Delta t, \end{cases}$$

whereas the outflows on each direction during the same time are

$$\text{Total outflow} \begin{cases} x-direction: & q_1(x+\Delta x)\Delta y\Delta z\Delta t, \\ y-direction: & q_2(y+\Delta y)\Delta x\Delta z\Delta t, \\ z-direction: & q_3(z+\Delta z)\Delta x\Delta y\Delta t. \end{cases}$$

In Figure 6.1, we depict an infinitesimal cube with the inflows and outflows on each spatial direction. Moreover, the change of fluid mass in the cube during the time interval Δt is:

$$\left(\rho_f(t+\Delta t) - \rho_f(t)\right)\Delta x\Delta y\Delta z, \tag{6.1}$$

where $\rho_f(t)$ denotes the fluid density at time t. In addition, if we consider any external source density f, the mass flow rate due to it during the time interval Δt is given by

$$\rho_f f\Delta x\Delta y\Delta z\Delta t. \tag{6.2}$$

At this point, the mass conservation states that

$$\begin{aligned} \left(\rho_f(t+\Delta t) - \rho_f(t)\right)\Delta x\Delta y\Delta z = &-\left(q_1(x+\Delta x) - q_1(x)\right)\Delta y\Delta z\Delta t \\ &-\left(q_2(y+\Delta y) - q_2(y)\right)\Delta x\Delta z\Delta t \\ &-\left(q_3(z+\Delta z) - q_3(z)\right)\Delta x\Delta y\Delta t \\ &+\rho_f f\Delta x\Delta y\Delta z\Delta t. \end{aligned} \tag{6.3}$$

If the previous equation is divided by $\Delta x\Delta y\Delta z$ and we take the limits as the dimensions of the reference volume tend to zero and the limit as $\Delta t \to 0$, then we obtain the mass conservation equation

$$\frac{\partial \rho_f}{\partial t} + \nabla \cdot \boldsymbol{q} = \rho_f f. \tag{6.4}$$

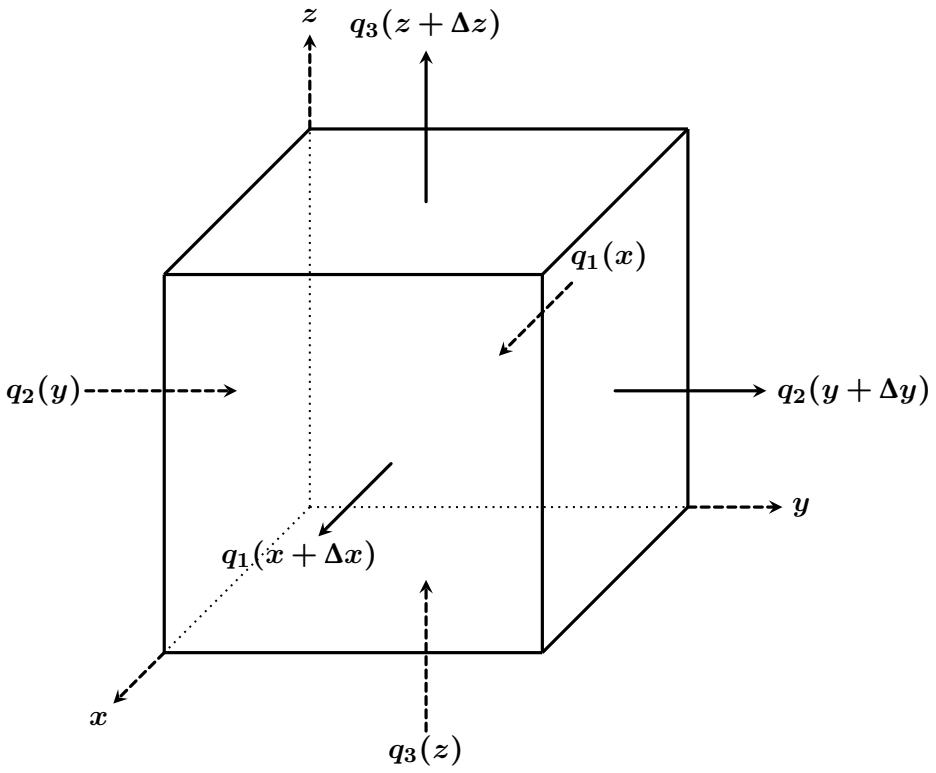


FIGURE 6.1: Infinitesimal cube: Inflows and outflows on each spatial direction.

In addition, the mass flux can be expressed as

$$\boldsymbol{q} = \rho_f \boldsymbol{v}_f, \tag{6.5}$$

with $\boldsymbol{v}_f$ denoting the fluid velocity. In a porous medium, the porosity is defined as the fraction of the pores available for the fluid

$$\phi = \frac{V_v}{V_t}, \tag{6.6}$$

where V_v denotes the volume of void space and V_t denotes the total volume or bulk volume including both solid and void components. Finally, by substituting the mass flux in (6.4) with the expression given in (6.5) and taking into account the porosity ϕ, the continuity equation for mass convservation in a porous medium is given by

$$\frac{\partial \rho_f \phi}{\partial t} + \nabla \cdot (\boldsymbol{\rho_f \phi v_f}) = \rho_f f. \tag{6.7}$$

6.1.2 Darcy's law

Darcy's law describes the conservation of momentum for fluid flow in a porous media. Darcy formulated his law in 1856 and stated that the percolation velocity of the fluid relative to the soil is proportional to the gradient of the pore pressure:

$$\textbf{Darcy's law:} \qquad \boldsymbol{v} = -\frac{\boldsymbol{\kappa}}{\eta_f}(\nabla p - \rho_f \boldsymbol{g}), \tag{6.8}$$

where $\boldsymbol{\kappa}$ is the permeability of the soil, η_f is the viscosity of the fluid and p is the pore pressure. Note that the product $\rho_f \boldsymbol{g}$ is the volumetric weight of the fluid and the relationship between Darcy's velocity and fluid velocity $\boldsymbol{v}_f$ is given by

$$\boldsymbol{v} = \phi \boldsymbol{v}_f. \tag{6.9}$$

In general, the permeability $\boldsymbol{\kappa}$ is a property of the soil that quantifies the ability of a porous medium to transmit fluids through it. It is often represented by means of a second order tensor as follows:

$$\kappa = \begin{pmatrix} \kappa_{11} & \kappa_{12} & \kappa_{13} \\ \kappa_{21} & \kappa_{22} & \kappa_{23} \\ \kappa_{31} & \kappa_{32} & \kappa_{33} \end{pmatrix}. \tag{6.10}$$

However, if the porous medium is isotropic, the permeability tensor is a diagonal matrix with equal coefficients on the diagonal. Due to this, we will treat the permeability as an scalar value κ for our experiments since isotropic porous medium will be considered for our model.

At this point, if we substitute Darcy's law (6.8) in the continuity equation (6.7) and take into account (6.9), we obtain the governing equation for a single phase flow:

$$\frac{\partial \rho_f \phi}{\partial t} - \nabla \cdot \left(\rho_f \frac{\boldsymbol{\kappa}}{\eta_f} (\nabla p - \rho_f \boldsymbol{g}) \right) = \rho_f f. \tag{6.11}$$

Since there might be changes in porosity and fluid density, the time differentiation of the previous equation yields

$$\phi \frac{\partial \rho_f}{\partial t} + \rho_f \frac{\partial \phi}{\partial t} - \nabla \cdot \left(\rho_f \frac{\boldsymbol{\kappa}}{\eta_f} (\nabla p - \rho_f \boldsymbol{g}) \right) = \rho_f f. \tag{6.12}$$

Now, the fluid density under isothermal conditions can be expressed as

$$\rho_f = \rho_f^0 e^{c_f (p - p^0)}, \tag{6.13}$$

where ρ_f^0 is the fluid density at the reference pressure p^0 and c_f denotes the fluid compressibility given by:

$$c_f = \frac{1}{\rho_f} \frac{d\rho_f}{dp}. \tag{6.14}$$

Differentiation with respect to t of both sides of equation (6.13) yields

$$\frac{\partial \rho_f}{\partial t} = c_f \rho_f \frac{\partial p}{\partial t}. \tag{6.15}$$

Moreover, we have that the time differentiation of the porosity gives:

$$\frac{\partial \phi}{\partial t} = \frac{d\phi}{dp} \frac{\partial p}{\partial t} = c_s \phi \frac{\partial p}{\partial t}, \tag{6.16}$$

where the solid compressibility is defined as

$$c_s = \frac{1}{\phi} \frac{d\phi}{dp}. \tag{6.17}$$

Then, substituting expressions (6.15) and (6.16) into (6.12), we obtain that

$$\rho_f c_t \frac{\partial p}{\partial t} - \nabla \cdot \left(\rho_f \frac{\boldsymbol{\kappa}}{\eta_f} (\nabla p - \rho_f \boldsymbol{g}) \right) = \rho_f f, \tag{6.18}$$

where $c_t = \phi(c_s + c_f)$ is known as the total compressibility coefficient. Finally, if we assume that the spatial changes in ρ_f and $\boldsymbol{g}$ can be neglected, after cancellation of ρ_f on both sides we have that the governing equation for single phase flow is given by the following parabolic equation

$$c_t \frac{\partial p}{\partial t} - \nabla \cdot \left(\frac{\boldsymbol{\kappa}}{\eta_f} \nabla p \right) = f. \tag{6.19}$$

6.1.3 Equilibrium equation

The fundamentals of solid mechanics are a crucial part of any poroelastic model. The principles of solid mechanics are the equilibrium equation and the compatibility condition. At this point, it is important to introduce the concept of stress tensor. As briefly mentioned in 2.2.4, the stress is defined as the external force applied on a body. In order to represent the stress vector effectively in a three-dimensional framework, it is very useful to decompose the stress into the stress components as follows:

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{pmatrix}. \tag{6.20}$$

Note that in the stress components the first index defines the direction of the plane where the stress acts whereas the second index defines the direction of the stress component. Hence, the stresses $\sigma_{xx}, \sigma_{yy}, \sigma_{zz}$ are called normal stresses whereas $\sigma_{xy}, \sigma_{xz}, \sigma_{yx}, \sigma_{yz}, \sigma_{zx}$ and σ_{zy} are named shear stresses. Moreover, the stress tensor is symmetric, that is

$$\sigma_{xy} = \sigma_{yx}, \quad \sigma_{xz} = \sigma_{zx}, \quad \sigma_{yz} = \sigma_{zy}.$$

The equilibrium equations describe the cancellation of all the external forces acting on a body at rest. Let us consider again the infinitesimal cube shown in Figure 6.1. The stress forces are split into normal forces and shear forces corresponding to the respective normal and shear stress components. Thus, we depict in Figure 6.2 the stress components acting on the infinitesimal cube. If we consider an additional external body force $\boldsymbol{f} = (f_x, f_y, f_z)$ acting on the stressed infinitesimal cube, the total sum of forces acting on each direction is given by

$$\begin{aligned} x - direction: \quad & \left(\sigma_{xx}(x+\Delta x) + \Delta x f_x - \sigma_{xx}(x)\right) \Delta y \Delta z + \left(\sigma_{yx}(y+\Delta y) - \sigma_{yx}(y)\right) \Delta x \Delta z \\ & + \left(\sigma_{zx}(z+\Delta z) - \sigma_{zx}(z)\right) \Delta x \Delta y. \end{aligned}$$

$$\begin{aligned} y - direction: \quad & \left(\sigma_{yy}(y+\Delta y) + \Delta y f_y - \sigma_{yy}(y)\right) \Delta x \Delta z + \left(\sigma_{xy}(x+\Delta x) - \sigma_{xy}(x)\right) \Delta y \Delta z \\ & + \left(\sigma_{zy}(z+\Delta z) - \sigma_{zy}(z)\right) \Delta x \Delta y. \end{aligned}$$

$$\begin{aligned} z - direction: \quad & \left(\sigma_{zz}(z+\Delta z) + \Delta z f_z - \sigma_{zz}(z)\right) \Delta x \Delta y + \left(\sigma_{yz}(y+\Delta y) - \sigma_{yz}(y)\right) \Delta x \Delta z \\ & + \left(\sigma_{xz}(x+\Delta x) - \sigma_{xz}(x)\right) \Delta y \Delta z. \end{aligned}$$

If we make equal to 0 these forces divided by $\Delta x \Delta y \Delta z$ and take the limits $\Delta x \to 0$, $\Delta y \to 0$ and $\Delta z \to 0$, then we obtain

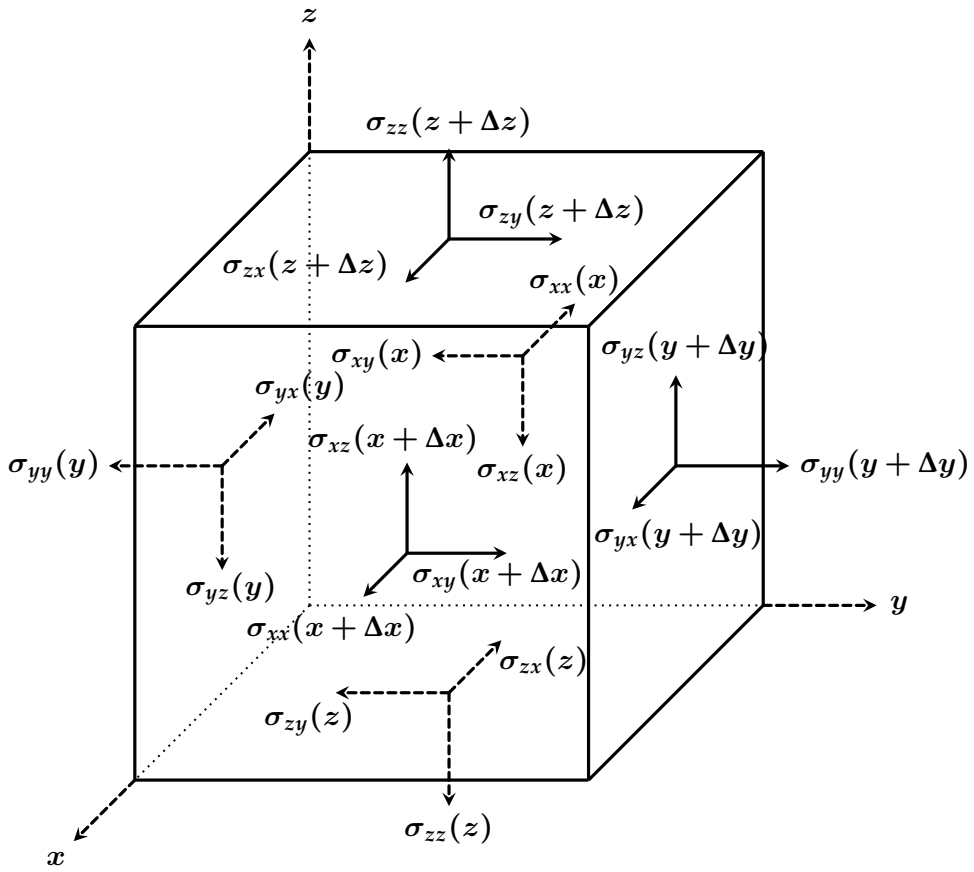


FIGURE 6.2: Infinitesimal cube: Stress components.

$$\begin{cases} \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{yx}}{\partial y} + \frac{\partial \sigma_{zx}}{\partial z} + f_x = 0, \\ \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{zy}}{\partial z} + f_y = 0, \\ \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + f_z = 0. \end{cases}$$

Hence, the equilibrium equation in a tensor form is given by

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{f} = 0. \tag{6.21}$$

When the solid matrix is subjected to stress, its deformation is quantified by the second-order strain tensor introduced also in the compatibility equation (2.41). Note that in the compatibility equation, the strain tensor is related to the displacement field $\boldsymbol{u}$ of the body due to external forces. Furthermore, the connection between stress and strain tensors is given by means of the generalized Hooke's law (2.42), where the elasticity or stiffness tensor $\mathbb{C}$ is involved. In the three-dimensional case, the number of independent elastic moduli appearing in $\mathbb{C}$ is reduced to 36. However, since we are considering homogeneous and isotropic materials, we can rewrite Hooke's law in terms of the two Lamé parameters λ and μ (see (2.49)).

6.1.4 Poroelasticity equations

Here, we introduce the concept of effective stress and combine the previous equations for fluid flow and the elastic deformation of the solid matrix in order to formulate the poroelastic Biot's model. The concept of effective stress was firstly introduced by Terzaghi and it claims that part of the total stress $\boldsymbol{\sigma}$ is carried by the soil structure and the other part is carried by the fluid. Hence, the part of the stress applied to the solid is called effective stress and it causes the rearrangement of the soil grains. Terzaghi's principle claims that the total stress is split into the effective stress $\boldsymbol{\sigma'}$ and the pore pressure p:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma'} - \alpha \boldsymbol{I} p, \tag{6.22}$$

where α is the so-called Biot-Willis constant. This constant is determined by the compressibilities of the solid c_s and the porous medium c_b:

$$\alpha = 1 - \frac{c_s}{c_b}. \tag{6.23}$$

In addition, it is common to express the Biot-Willis constant in terms of the bulk modulus of the solid K_s and drained bulk modulus of the porous medium K_b. For this purpose, let us note that:

$$c_s = \frac{1}{K_s}, \qquad c_b = \frac{1}{K_b}.$$

Thus, the effective stress coefficient can be rewritten as follows

$$\alpha = 1 - \frac{K_b}{K_s}. \tag{6.24}$$

Also, note that $0 \leq \alpha \leq 1$ since the compressibilities hold $c_s \leq c_b$.

Another important ingredient in poroelastic Biot's theory is the so-called variation in fluid content ς. In fact, the mass conservation equation (6.4) can be rewritten in terms of ς (see Detournay and Cheng, 1993; Wang, 2001) by ignoring the effect of fluid density ρ_f:

$$\frac{\partial \varsigma}{\partial t} + \nabla \cdot \boldsymbol{v} = f. \tag{6.25}$$

The variation of fluid content depends on the pore pressure p and the pore volume measured by the volumetric strain $\varepsilon_v = \nabla \cdot \boldsymbol{u}$. More specifically,

$$\varsigma = \frac{p}{M} + \alpha \nabla \cdot \boldsymbol{u}, \tag{6.26}$$

where M is known as Biot modulus. Moreover, the inverse of M is called storage coefficient and it is related to the compressibilities c_s, c_f and the porosity ϕ as follows:

$$\frac{1}{M} = c_s(\alpha - \phi) + c_f \phi = \frac{\alpha - \phi}{K_s} + \frac{\phi}{K_f}, \tag{6.27}$$

where $K_f = 1/c_f$ is the bulk modulus of the fluid. At this point, if we substitute (6.26) in the fluid mass conservation equation (6.25), then we obtain:

$$\frac{1}{M}\frac{\partial p}{\partial t} + \alpha \frac{\partial \nabla \cdot \boldsymbol{u}}{\partial t} + \nabla \cdot \boldsymbol{v} = f. \tag{6.28}$$

Thus, the previous equation together with the combination of the equilibrium equation (6.21) and Terzaghi's principle given in (6.22) yields the following system:

$$-\nabla \cdot \boldsymbol{\sigma}'(\boldsymbol{u}) + \alpha \nabla p = \boldsymbol{f}, \tag{6.29}$$

$$\frac{1}{M}\frac{\partial p}{\partial t} + \alpha \frac{\partial}{\partial t}(\nabla \cdot \boldsymbol{u}) + \nabla \cdot \boldsymbol{v} = f. \tag{6.30}$$

Finally, if we apply generalized Hooke's law for linear isotropic and homogeneous materials to (6.29) and we use Darcy's law (6.8) to substitute the percolation velocity $\boldsymbol{v}$ in (6.30), the following poroelastic system is obtained:

$$-\mu \boldsymbol{\Delta u} - (\lambda + \mu)\nabla(\nabla \cdot \boldsymbol{u}) + \alpha \nabla p = \boldsymbol{f}, \tag{6.31}$$

$$\frac{1}{M}\frac{\partial p}{\partial t} + \alpha \frac{\partial}{\partial t}(\nabla \cdot \boldsymbol{u}) - \nabla \cdot \left(\frac{\boldsymbol{\kappa}}{\eta_f}\nabla p\right) = f. \tag{6.32}$$

The previous PDE system is known as the two-field or displacement-pressure formulation of the quasi-static Biot's model for soil consolidation process. Note that in (6.32), the changes in the volumetric weight of the fluid have been neglected. In order to complete the formulation of this problem, some boundary and initial conditions must be added. Thus, let us denote with $\Omega \subset \mathbb{R}^d$ an open and bounded domain. In addition let us consider the splits of the boundary $\partial\Omega = \Gamma_{\boldsymbol{u}}^D \cup \Gamma_{\boldsymbol{u}}^N$ and $\partial\Omega = \Gamma_p^D \cup \Gamma_p^N$, with $\Gamma_{\boldsymbol{u}}^D \cap \Gamma_{\boldsymbol{u}}^N = \emptyset$ and $\Gamma_p^D \cap \Gamma_p^N = \emptyset$. Thus, Dirichlet type boundary conditions set the value of displacement field $\boldsymbol{u}$ and pressure p on $\Gamma_{\boldsymbol{u}}^D$ and Γ_p^D respectively as follows:

$$\text{Dirichlet b.c.:} \begin{cases} \boldsymbol{u}(\boldsymbol{x}) = u_D \text{ with } \boldsymbol{x} \in \Gamma_{\boldsymbol{u}}^D, \\ p(\boldsymbol{x}) = p_D \text{ with } \boldsymbol{x} \in \Gamma_p^D, \end{cases} \tag{6.33}$$

whereas the Neumann type boundary conditions impose the traction $\boldsymbol{t}_N$ on $\Gamma_{\boldsymbol{u}}^N$ and the normal flux $\boldsymbol{v}_N$ on Γ_p^N

$$\text{Neumann b.c.:} \begin{cases} \boldsymbol{\sigma}'\boldsymbol{n} = \boldsymbol{t} \text{ with } \boldsymbol{x} \in \Gamma_{\boldsymbol{u}}^N, \\ \boldsymbol{v}\boldsymbol{n} = \boldsymbol{v}_N \text{ with } \boldsymbol{x} \in \Gamma_p^N. \end{cases} \tag{6.34}$$

Finally, the initial conditions for the quasi-stationary Biot's model can be set either for (6.31) with a initial displacement $\boldsymbol{u}$ or stress $\boldsymbol{\sigma}$

$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}_0 \quad \text{or} \quad \boldsymbol{\sigma}(\boldsymbol{x}, 0) = \boldsymbol{\sigma}_0, \quad \boldsymbol{x} \in \Omega, \tag{6.35}$$

and initial pressure or flux initial conditions for equation (6.32) as follows

$$p(\boldsymbol{x}, 0) = p_0 \quad \text{or} \quad \boldsymbol{v}(\boldsymbol{x}, 0) = \boldsymbol{v}_0, \quad \boldsymbol{x} \in \Omega. \tag{6.36}$$

6.2 Isogeometric discretization

In this section we introduce the strong formulation of the quasi-static Biot's model for soil consolidation and discretize this problem by using isogeometric discretizations. For our numerical experiments, we consider the following displacement-pressure strong formulation:

$$
\begin{aligned}
-\mu\boldsymbol{\Delta u} - (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u}) + \alpha\nabla p &= \boldsymbol{f}, && \text{in } \Omega,\\
\frac{1}{M}\frac{\partial p}{\partial t} + \alpha\frac{\partial}{\partial t}(\nabla\cdot\boldsymbol{u}) - \nabla\cdot\left(\frac{\boldsymbol{\kappa}}{\eta_f}\nabla p\right) &= f, && \text{in } \Omega,\\
\boldsymbol{\sigma'n} = \boldsymbol{t}, \quad p &= 0, && \text{on } \Gamma_t,\\
\boldsymbol{u} = \boldsymbol{0}, \quad \nabla p\cdot\boldsymbol{n} &= 0, && \text{on } \Gamma_c,\\
\left(\frac{p}{M} + \alpha\nabla\cdot\boldsymbol{u}\right) &= 0, && \text{in } \Omega\times\{0\},
\end{aligned}
\tag{6.37}
$$

where $\Omega\subset\mathbb{R}^d$ is an open, bounded domain and time $t\in(0,T)$. Moreover, in terms of the notation given in (6.33) and (6.34), we have set $\Gamma_t = \Gamma_{\boldsymbol{u}}^N = \Gamma_p^D$, $\Gamma_c = \Gamma_{\boldsymbol{u}}^D = \Gamma_p^N$. For the variational formulation of this problem, let us consider the variational spaces

$$
\begin{aligned}
\boldsymbol{\mathcal{V}} &= \{\boldsymbol{u}\in H^1(\Omega)^d \;|\; \boldsymbol{u}|_{\Gamma_c} = \boldsymbol{0}\},\\
\mathcal{Q} &= \{p\in H^1(\Omega) \;|\; p|_{\Gamma_t} = 0\}.
\end{aligned}
$$

Hence, if we test equations (6.31) and (6.32) with trial functions $\boldsymbol{v}\in\boldsymbol{\mathcal{V}}$ and $q\in\mathcal{Q}$ respectively, the variational formulation reads as follows: For each $t\in(0,T)$, find $(\boldsymbol{u}(t),p(t))\in\boldsymbol{\mathcal{V}}\times Q$ such that:

$$
\begin{cases}
a(\boldsymbol{u}(t),\boldsymbol{v}) + \alpha(\nabla p,\boldsymbol{v}) = (\boldsymbol{f},\boldsymbol{v}), & \forall\boldsymbol{v}\in\boldsymbol{\mathcal{V}},\\
\alpha(\nabla\cdot\dot{\boldsymbol{u}}(t),q) + \frac{1}{M}(\dot{p}(t),q) + b(p,q) = (f,q), & \forall q\in\mathcal{Q},
\end{cases}
\tag{6.38}
$$

where the bilinear forms $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ are given by

$$
\begin{aligned}
a(\boldsymbol{u},\boldsymbol{v}) &= 2\int_\Omega \mu\boldsymbol{\epsilon}(\boldsymbol{u}):\boldsymbol{\epsilon}(\boldsymbol{v})d\Omega + \int_\Omega \lambda\nabla\cdot\boldsymbol{u}\,\nabla\cdot\boldsymbol{v}d\Omega,\\
b(p,q) &= \int_\Omega \frac{\boldsymbol{\kappa}}{\eta_f}\nabla p\cdot\nabla q\,d\Omega.
\end{aligned}
$$

As mentioned before, our goal is to discretize this problem with isogeometric discretizations. Although there are not many published works about IGA discretizations of poroelasticity problems, discretizations of equal polynomial order for pressure and displacements, that is, $k_p = k_u$ were proposed in Irzal et al., 2013; Bekele et al., 2022. In addition, this approach was compared with discretizations using the polynomial degree relation $k_u = k_p + 1$ in Bekele et al., 2022 showing a better performance of mixed degrees. Moreover, an isogeometric space-time method for two-field Biot's model was proposed in Arf and Simeon, 2022 and it was shown that the use of mixed degrees reduce pressure oscillations and the locking effect. Due to this, we consider the isogeometric generalization of Taylor-Hood finite dimensional spaces $(V_h,Q_h) = (V_h^{TH}(k,s),Q_h^{TH}(k,s))$ introduced in (2.66). With regard to the time discretization, we propose an implicit Euler scheme such that the fully discretized scheme at time t_m is written as: Find $(\boldsymbol{u}_h^m,p_h^m)\in V_h\times Q_h$ such that

$$
\begin{cases}
a(\boldsymbol{u}_h^m,\boldsymbol{v}_h) + \alpha(\nabla p_h^m,\boldsymbol{v}_h) = (\boldsymbol{f}_h^m,\boldsymbol{v}_h), & \forall\boldsymbol{v}_h\in V_h,\\
\alpha(\nabla\cdot\bar{\boldsymbol{u}}_h^m,q_h) + \frac{1}{M}(\bar{p}_h^m,q_h) + b(p_h^m,q_h) = (f_h^m,q_h), & \forall q_h\in\mathcal{Q}_h,
\end{cases}
\tag{6.39}
$$

where $\bar{\boldsymbol{u}}_h^m := (\boldsymbol{u}_h^m - \boldsymbol{u}_h^{m-1})/\tau$, $\bar{p}_h^m := (p_h^m - p_h^{m-1})/\tau$ and τ is the time step. This numerical scheme leads to a large sparse linear system that has to be solved at each

time step t_m:

$$\mathcal{A}\begin{pmatrix} \boldsymbol{U}^m \\ P^m \end{pmatrix} = \begin{pmatrix} F_u^m \\ \tau F_p^m \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -B & \frac{1}{M}M_p \end{pmatrix}\begin{pmatrix} \boldsymbol{U}^{m-1} \\ P^{m-1} \end{pmatrix}, \tag{6.40}$$

where $\boldsymbol{U}^m$ and P^m denote vectors with the control variables for $\boldsymbol{u}_h^m$ and p_h^m respectively, $-B$ is related to the divergence operator multiplied by α and M_p denotes the mass matrix associated to the pressure. In addition, the matrix $\mathcal{A}$ is structured as follows

$$\mathcal{A} = \begin{pmatrix} E_l & B^T \\ -B & \tau C + \frac{1}{M}M_p \end{pmatrix}, \tag{6.41}$$

where E_l is the elasticity matrix corresponding to the IGA discretization of $a(\boldsymbol{u}_h^m, \boldsymbol{v}_h)$ and C corresponds to the contribution of p_h^m in the discretized continuity equation. In practice, the value $1/M$ is very small and we will consider $1/M = 0$ and $\alpha = 1$ for our experiments. Thus, matrix C corresponds to the stiffness matrix of a Poisson problem multiplied by κ/η_f.

6.2.1 Mass stabilization

Standard discretizations of Biot's model might give non-physical oscillations in the numerical approximation of the pressure. Because of this, an stabilization term is required for some cases such that the spurious oscillations are eliminated. This is the case for example when very small time steps τ are taken or low permeabilities $\boldsymbol{\kappa}$ close to zero are considered. For instance, in Aguilar et al., 2008; Rodrigo et al., 2016 the authors proposed an stabilization of the flow equation based on the stiffness matrix of p multiplied by a parameter that depends on h, the type of discretization and Lamé moduli λ and μ.

Here, we propose a discretization of the Biot's equations with a mass stabilization term that suits perfectly to the application of the fixed-stress split method. This stabilization has been introduced in Pé de la Riva et al., 2022 for finite element discretizations and it consists of adding two terms to the flow equation such that the following discrete variational formulation is obtained:

$$a(\boldsymbol{u}_h^m, \boldsymbol{v}_h) - \alpha(p_h^m, \nabla\cdot\boldsymbol{v}_h) = (\boldsymbol{f}_h^m, \boldsymbol{v}_h), \quad \forall\, \boldsymbol{v}_h \in \boldsymbol{V}_h, \tag{6.42}$$

$$\alpha(\nabla\cdot\bar{\boldsymbol{u}}_h^m, q_h) + b(p_h^m, q_h) + L(\bar{p}_h^m, q_h)_0 - L(\bar{p}_h^m, q_h) = (f_h^m, q_h), \ \forall\, q_h \in Q_h, \tag{6.43}$$

where $(\cdot,\cdot)_0$ is an approximation of the $L_2(\Omega)$ inner product defined by mass lumping. Thus, in a matrix form the discrete problem is given by

$$\mathcal{A}_{stab}\begin{pmatrix} \mathbf{U}^m \\ P^m \end{pmatrix} = \begin{pmatrix} F_u^m \\ \tau F_p^m \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -B & L(M_l - M_p) \end{pmatrix}\begin{pmatrix} \mathbf{U}^{m-1} \\ P^{m-1} \end{pmatrix}, \tag{6.44}$$

where M_l is the row-sum lumped mass matrix and the matrix $\mathcal{A}_{stab}$ is given by

$$\mathcal{A}_{stab} = \begin{pmatrix} E_l & B^T \\ -B & \tau C + L(M_l - M_p) \end{pmatrix}. \tag{6.45}$$

The motivation of this stabilization and the choice of parameter L comes from isolating $\boldsymbol{u}$ on the mechanics equation at the discrete scheme in (6.39) without external forces. Then, the expression obtained for $\boldsymbol{u}$ is substituted on the flow equation. If we isolate the displacement vector at the mechanics part of (6.44) we obtain

$\mathbf{U}^m = -E_l^{-1}B^T P^m$. Thus, by substituting on the flow equation (and considering that τC is negligible) one gets

$$(-BE_l^{-1}B^T + L(M_l - M_p))P^m = (-BE_l^{-1}B^T + L(M_l - M_p))P^{m-1}, \tag{6.46}$$

and then if we choose L such that $(-BE_l^{-1}B^T + L(M_l - M_p))$ is a M-matrix, an oscillation-free solution is ensured. Note that the goal of discretizing both added terms in two different ways lies on the spectral equivalence between $-BE_l^{-1}B^T$ and M_p. Applying mass-lumping provides a diagonal matrix that does not affect on the off-diagonal entries. Thus, it is always possible to find a constant L big enough such that $-BE_l^{-1}B^T - LM_p$ has non-positive off-diagonal entries. Moreover, the optimal choice of L is the minimum value that satisfies that condition. At this point, let us consider Terzaghi's problem:

$$\begin{aligned} -(2\mu+\lambda)\frac{\partial^2 u}{\partial x^2} + \frac{\partial p}{\partial x} &= 0, \quad x \in (0,1), \quad t \in (0,1), \\ \frac{\partial}{\partial t}\left(\frac{\partial u}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\kappa}{\eta_f}\frac{\partial p}{\partial x}\right) &= 0, \quad x \in (0,1), \quad t \in (0,1), \end{aligned} \tag{6.47}$$

with the following initial and boundary conditions

$$\begin{aligned} -(2\mu+\lambda)\frac{\partial u}{\partial x}(0,t) &= 1, \quad p(0,t) = 0, \quad t \in (0,1), \\ u(1,t) &= 0, \quad \frac{\kappa}{\eta_f}\frac{\partial p}{\partial x}(1,t) = 0, \quad t \in (0,1), \\ \frac{\partial u}{\partial x}(x,0) &= 0, \quad x \in (0,1). \end{aligned} \tag{6.48}$$

For this one-dimensional problem, the optimal choice of L using a linear $Q1-Q1$ discretization is

$$L_{Q1-Q1} = \frac{3}{2(2\mu+\lambda)}. \tag{6.49}$$

Also for Terzaghi's problem, we consider the isogeometric Taylor-Hood elements. More concretely we apply the $Q2-Q1$ elements and the global $\mathcal{C}^1$ smooth numerical solution provided by the isogeometric $Q3-Q2$ elements. For these isogeometric Taylor-Hood elements, the optimal choice of the parameter L is

$$L_{TH} = \frac{1}{(2\mu+\lambda)}. \tag{6.50}$$

By adding the mass stabilization term multiplied by a suitable value of L, all the spurious oscillations in the numerical approximation of pressure are eliminated. In order to visualize this, in Figure 6.3, we show the exact solution and numerical approximation plots of pressure for problem (6.47) with $\tau = 10^{-6}$, $\frac{\kappa}{\eta_f} = 1$, Lamé parameters such that $2\mu + \lambda = 1$, $h = 2^{-6}$ by using those isogeometric elements $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding global $\mathcal{C}^1$ smoothness) (a) without stabilization and (b) with the corresponding optimal mass stabilization term. As we can observe, this stabilization based on the mass matrix annihilates all the non-physical oscillations.

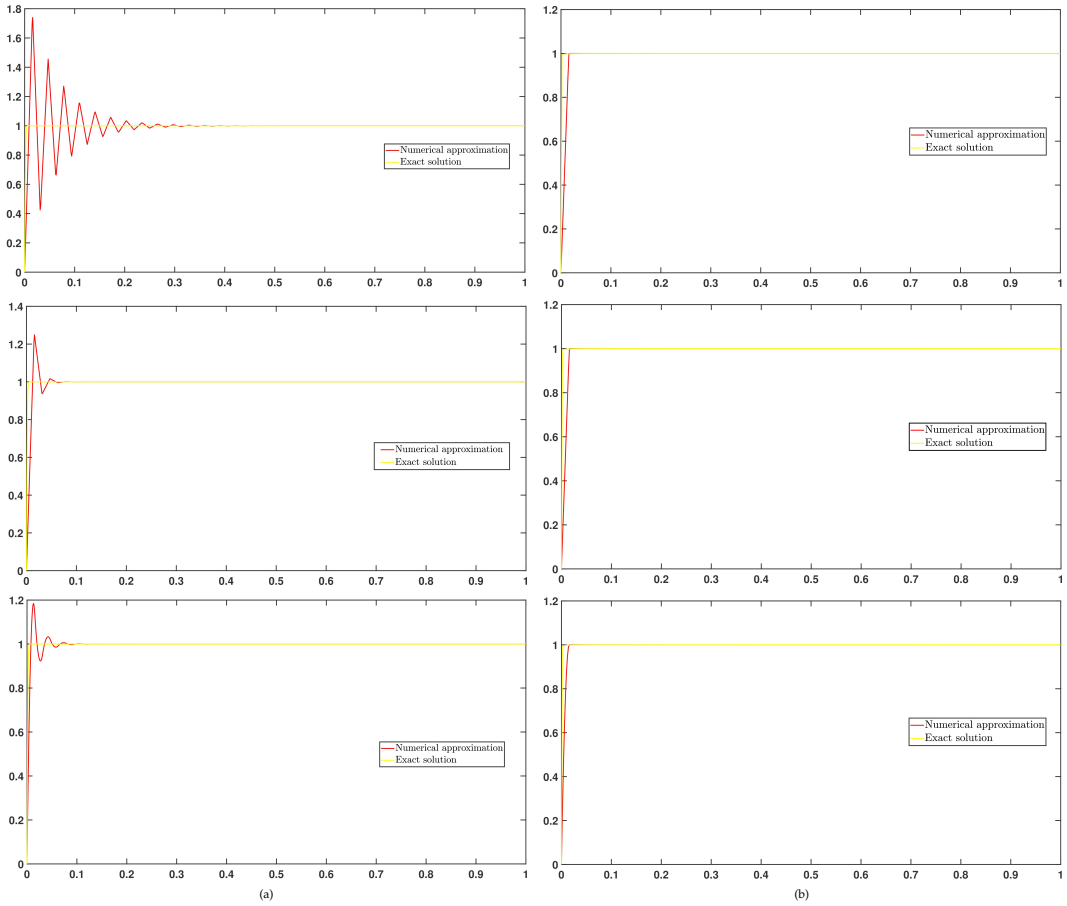


FIGURE 6.3: Terzaghi's problem: Exact solution and numerical approximation plots of pressure for problem (6.47) with $\tau = 10^{-6}$, $\frac{\kappa}{\eta_f} = 1$, Lamé parameters such that $2\mu + \lambda = 1$, $h = 2^{-6}$ by using those isogeometric elements $Q1 - Q1$, $Q2 - Q1$ and $Q3 - Q2$ (holding global $\mathcal{C}^1$ smoothness) (a) without stabilization and (b) with the corresponding optimal mass-stabilization term.

6.3 Decoupled solvers

Decoupled solvers are one of the main approaches for solving Biot's equations. The iterative coupling approach consists of solving sequentially the flow and mechanics parts using the latest solution available. At each time step, this process is repeated until convergence is achieved. The main advantage of decoupled solvers is their flexibility since they allow to use separate codes for each sub-problem of the system. In addition, we can make a good use of already existing solvers for each problem and only the interface between them has to be carefully treated.

Classification of decoupled solvers depends on which part is solved first, either the mechanical or flow problem. Hence, there are two main methods that solve first the mechanical problem: The undrained and drained split methods. The undrained split imposes that $\frac{\partial \varsigma}{\partial t}$ remains constant during the mechanical step and then the pressure is allowed to change, see Kim, Tchelepi, and Juanes, 2011a. Regarding the drained split, this approach solves the mechanical problem first with a fixed pressure and then the flow problem is solved with a fixed displacement field.

The two main approaches based on solving first the fluid flow sub-problem are

the fixed-strain and fixed-stress split methods. The fixed-strain split method freezes the volumetric strain $\varepsilon_v = (\varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz})$ in the flow equation. Then, the mechanical problem is solved using the new approximation for the pressure p. Finally, the fixed-stress split method fixes the volumetric mean stress for the solution of the flow problem. Next, we are going to give a deeper insight on this latter.

6.3.1 Fixed-stress split method

Here, we describe in detail the fixed-stress split method whose stability and convergence was firstly demonstrated in Mikelić and Wheeler, 2013. In physical terms, the fixed-stress split scheme is an iterative method that assumes the volumetric mean stress $\sigma_v = K_b\varepsilon_v - \alpha p$ to be constant, see for instance Kim, Tchelepi, and Juanes, 2011b; Both et al., 2017. Hence, let us rewrite the flow equation (6.32) in terms of σ_v instead of the volumetric strain:

$$\left(\frac{1}{M} + \frac{\alpha^2}{K_b}\right)\frac{\partial p}{\partial t} + \frac{\alpha}{K_b}\frac{\partial \sigma_v}{\partial t} - \nabla\cdot\left(\frac{\boldsymbol{\kappa}}{\eta_f}\nabla p\right) = f. \tag{6.51}$$

Thus, the fixed-stress split scheme solves equation (6.51) with $\frac{\alpha}{K_b}\frac{\partial \sigma_v}{\partial t}$ already known and it results in the following splitting of the matrix $\mathcal{A} = \mathcal{M}_A - \mathcal{N}_A$ with

$$\mathcal{M}_{\mathcal{A}} = \begin{pmatrix} E_l & B^T \\ 0 & \tau C + \dfrac{\alpha^2}{K_b}M_p \end{pmatrix}, \qquad \mathcal{N}_{\mathcal{A}} = \begin{pmatrix} 0 & 0 \\ B & \dfrac{\alpha^2}{K_b}M_p \end{pmatrix}. \tag{6.52}$$

This splitting leads to fixed-stress as an iterative coupling method and a sequence of approximations at each time t_m are obtained. Given this matrix splitting, the fixed-stress splitting algorithm can be understood as an iterative scheme based on adding an stabilization term $L\frac{\partial p}{\partial t}$ on both sides of (6.32):

$$\left(\frac{1}{M} + L\right)\frac{\partial p}{\partial t} - \nabla\cdot\left(\frac{\boldsymbol{\kappa}}{\eta_f}\nabla p\right) = f - \alpha\frac{\partial}{\partial t}(\nabla\cdot\mathbf{u}) + L\frac{\partial p}{\partial t}. \tag{6.53}$$

Thus, parameter L has to be fixed carefully in order to achieve convergence for fixed-stress splitting method. For this task, a rigorous convergence analysis of this method was presented in Borregales et al., 2019 and it was proved that the fixed-stress split method for the two-field formulation of Biot's model converges for any value of L such that

$$L \geq \frac{\alpha^2}{2\,(\lambda + 2\mu/d)}. \tag{6.54}$$

In addition, it was proved in Both et al., 2017 that optimal convergence rates are obtained in case of equality in (6.54). Thus, the fixed-stress split algorithm (considering $1/M \approx 0$) is performed in two steps:

- **Step 1.** Given $(\boldsymbol{u}_h^{m,i-1}, p_h^{m,i-1}) \in V_h \times Q_h$, find $p_h^{m,i} \in Q_h$ such that

$$\begin{aligned}(Lp_h^{m,i}, q_h) + \tau b(p_h^{m,i}, q_h) = \tau(f_h, q_h) + (\nabla\cdot\boldsymbol{u}_h^{m-1}, q_h) + (Lp_h^{m,i-1}, q_h) \\ - (\nabla\cdot\boldsymbol{u}_h^{m,i-1}, q_h),\ \forall q_h \in Q_h.\end{aligned} \tag{6.55}$$

- **Step 2.** Given $p_h^{m,i} \in Q_h$, find $\mathbf{u}_h^{m,i} \in V_h$ such that

$$a(\boldsymbol{u}_h^{m,i}, \boldsymbol{v}_h) = (\boldsymbol{f}, \boldsymbol{v}_h) - (\nabla p_h^{m,i}, \boldsymbol{v}_h), \ \forall \boldsymbol{v}_h \in V_h. \tag{6.56}$$

In a matrix form, the fixed-stress iterative coupling method can be written as follows: Given the approximations $(\boldsymbol{U}^{m,i-1}, P^{m,i-1})$, solve sequentially the two following linear systems until convergence is achieved:

$$\begin{cases} (\tau C + LM_p)P^{m,i} &= \tau F_p + B(\boldsymbol{U}^{m,i-1} - \boldsymbol{U}^{m-1}) + LM_p P^{m,i-1}, \\ E_l \boldsymbol{U}^{m,i} &= F_u - B^T P^{m,i}, \end{cases} \tag{6.57}$$

where $\boldsymbol{U}^{m,i}$ and $P^{m,i}$ denote the vector with the control variables for the displacements and pressure approximation respectively at the $i-$th iteration of the fixed-stress split method and time t_m.

6.3.2 Mass stabilized iterative scheme

The mass stabilization introduced in Section 6.2.1 suits perfectly for the automatic application of a decoupled algorithm of the type of the fixed-stress split method, giving rise to a mass stabilized iterative scheme for the solution of (6.42)-(6.43). This method is introduced in Algorithm 7: First, we solve the flow equation with the two stabilization terms and then the updated approximation for the pressure is used in the equilibrium equation for the next approximation of the displacements.

Algorithm 7 An iterative method for the solution of discrete problem (6.42)-(6.43)

for $i = 1, 2, \ldots$ **do**

Step 1: Given $(\boldsymbol{u}_h^{m,i-1}, p_h^{m,i-1}) \in \boldsymbol{V}_h \times Q_h$, find $p_h^{m,i} \in Q_h$ such that:

$$L\left(\frac{p_h^{m,i} - p_h^{m-1}}{\tau}, q_h\right)_0 + b(p_h^{m,i}, q_h) = -\alpha\left(\nabla \cdot \frac{\boldsymbol{u}_h^{m,i-1} - \boldsymbol{u}_h^{m-1}}{\tau}, q_h\right) + L\left(\frac{p_h^{m,i-1} - p_h^{m-1}}{\tau}, q_h\right) + (f_h^m, q_h), \quad \forall\, q_h \in Q_h,$$

Step 2: Given $p_h^{m,i} \in Q_h$, find $\boldsymbol{u}_h^{m,i} \in \boldsymbol{V}_h$ such that

$$a(\boldsymbol{u}_h^{m,i}, \boldsymbol{v}_h) = \alpha(p_h^{m,i}, \nabla \cdot \boldsymbol{v}_h) + (\boldsymbol{f}_h^m, \boldsymbol{v}_h), \quad \forall\, \boldsymbol{v}_h \in \boldsymbol{V}_h.$$

In matrix form, this iterative scheme is based on the splitting of $\mathcal{A}_{stab}$ as follows

$$\mathcal{A}_{stab} = \mathcal{M}_{\mathcal{A}_{stab}} - \mathcal{N}_{\mathcal{A}_{stab}}, \tag{6.58}$$

where

$$\mathcal{M}_{\mathcal{A}_{stab}} = \begin{pmatrix} E_l & B^T \\ 0 & \tau C + LM_l \end{pmatrix}, \qquad \mathcal{N}_{\mathcal{A}_{stab}} = \begin{pmatrix} 0 & 0 \\ B & LM_p \end{pmatrix}.$$

Note that this approach is similar to the fixed-stress method with the advantage of using the mass term for two different purposes: The stabilization of the numerical solution and the application of the iterative method.

6.3.3 Von Neumann analysis

Now, we focus our attention on the convergence and stability analysis of the previously proposed decoupled method for solving isogeometric discretizations for poroelastic problems. In order to prove that the optimal choice of L provides an stable and convergent scheme, we perform the so-called von Neumann analysis. The von Neumann analysis is an useful tool for the study of stability and convergence of numerical schemes and it has been widely applied in the literature, see for instance Strikwerda, 2004; Kim, Tchelepi, and Juanes, 2011a; Kim, Tchelepi, and Juanes, 2011b. In this section we carry out the von Neumann analysis for stability and convergence of the proposed decoupled scheme (6.58). We perform this analysis for Terzaghi's problem discretized with the pair of $Q1-Q1$ elements and isogeometric Taylor-Hood elements $Q2-Q1$ and $Q3-Q2$ holding a $\mathcal{C}^1$ global smoothness of the solution. Then, we extend the results obtained for the one-dimensional case to the Biot's equations making a good use of the tensor-product structure of B-splines discretizations. However, the authors are aware of the fact that the general extension to $d-$dimensional cases is semi-heuristic.

The Von Neumann analysis is based on the Fourier transform as LFA and analyzes the amplification factor γ of the solution in advancing one time step for stability. As it is described in Strikwerda, 2004, a shortcut to the transform theory consists to consider that the solutions at t_m can be expressed as

$$U_j^m = \gamma^m e^{\imath j\theta}\hat{U}, \qquad P_j^m = \gamma^m e^{\imath j\theta}\hat{P}, \tag{6.59}$$

where $\hat{U}, \hat{P}$ are the initial solutions and $\theta \in [-\pi, \pi]$. As a starting point, let us consider the pair of elements $Q1-Q1$. Hence, by substituting (6.59) into our decoupled scheme applied to Terzaghi's problem we obtain the following linear system:

$$\widetilde{\mathcal{A}}^{Q1-Q1}\begin{pmatrix}\hat{U}\\ \hat{P}\end{pmatrix} = \begin{pmatrix}0\\0\end{pmatrix}, \tag{6.60}$$

where matrix $\widetilde{\mathcal{A}}^{Q1-Q1}$ is given by

$$\widetilde{\mathcal{A}}^{Q1-Q1} = \begin{pmatrix} \frac{2(2\mu+\lambda)}{h}\gamma(1-\cos\theta) & \gamma\imath\sin\theta \\ (\gamma-1)\imath\sin\theta & \gamma\left(\frac{2\kappa\tau}{\eta_f h}(1-\cos\theta)\right) + (\gamma-1)\frac{h}{2(2\mu+\lambda)}(1-\cos\theta) \end{pmatrix}.$$

In order to obtain the possible values of the amplification factor γ, we impose to the previous matrix to be singular since the regular case yields directly $\gamma = 0$. Hence, we obtain that the two values for the amplification factor γ are

$$\gamma_1(\theta) = 0, \qquad \gamma_2(\theta) = \left(1 + 2\kappa\tau(2\mu+\lambda)(1-\cos\theta)/(\eta_f h^2)\right)^{-1}.$$

Then, since $|\gamma_1| \leq 1$ and $|\gamma_2| \leq 1$, we conclude that this scheme is stable. We study now the stability of our proposed iterative scheme by using the pair of Taylor-Hood elements $Q2-Q1$. In this case, there are two different stencils for the displacements u. Hence, we consider a linear system with vertex U_v and mid-point U_m displacements unknowns for the analysis. Therefore, the corresponding numerical scheme is given by

$$\widetilde{\mathcal{A}}^{Q2-Q1}\begin{pmatrix}\hat{U}_v\\ \hat{U}_m\\ \hat{P}\end{pmatrix} = \begin{pmatrix}0\\0\\0\end{pmatrix}, \tag{6.61}$$

where the matrix $\widetilde{\mathcal{A}}^{Q2-Q1}$ is given by

$$\widetilde{\mathcal{A}}^{Q2-Q1} = \begin{pmatrix} \gamma\frac{4(2\mu+\lambda)}{3h}(2-\cos\,\theta) & -\gamma\frac{4(2\mu+\lambda)}{3h}\cos\,\theta & \gamma\frac{2\imath}{3}\sin\,\theta \\ -\gamma\frac{4(2\mu+\lambda)}{3h}\cos\,\theta & \gamma\frac{4(2\mu+\lambda)}{3h} & \gamma\frac{2\imath}{3}\sin\,\frac{\theta}{2} \\ (\gamma-1)\frac{2\imath}{3}\sin\,\theta & (\gamma-1)\frac{2\imath}{3}\sin\,\frac{\theta}{2} & \gamma\frac{2\tau\kappa(1-\cos\,\theta)}{\mu_f h}+(\gamma-1)\frac{h(1-\cos\,\theta)}{3(2\mu+\lambda)} \end{pmatrix}.$$

Thus, if we make the determinant of $\widetilde{\mathcal{A}}^{Q2-Q1}$ equal to zero, we obtain that the values for the amplification factor are

$$\gamma_1(\theta)=\gamma_2(\theta)=0, \qquad \gamma_3(\theta)=\left(1+2\kappa\tau(2\mu+\lambda)(1-\cos\,\theta)/(\mu_f h^2)\right)^{-1}.$$

Again, all the values of the amplification factor hold $|\gamma_i|\leq 1$, $i=1,2,3$. Therefore, our proposed iterative scheme for the pair $Q2-Q1$ is stable.

At this point, we perform a stability analysis for the case in which we consider the isogeometric Taylor-Hood pair of elements $Q3-Q2$ with global $\mathcal{C}^1$ smoothness. In this case, the discretization for the displacements yields two different stencils and therefore two different unknowns U_v, U_m are considered. The numerical scheme for this discretization is given by

$$\widetilde{\mathcal{A}}^{Q3-Q2}\begin{pmatrix}\hat{U}_v\\ \hat{U}_m\\ \hat{P}\end{pmatrix}=\begin{pmatrix}0\\0\\0\end{pmatrix}, \tag{6.62}$$

where the elements $\{\widetilde{a}_{i,j}^{Q3-Q2}\}$, $i,j=1,2,3$ of the matrix $\widetilde{\mathcal{A}}^{Q3-Q2}$ are

$$\begin{aligned}
\widetilde{a}_{1,1}^{Q3-Q2} =& \gamma\frac{6(2\mu+\lambda)}{h}\left(\frac{1}{5}-\frac{1}{8}\cos\,\theta\right),\\
\widetilde{a}_{1,2}^{Q3-Q2} =& -\gamma\frac{3(2\mu+\lambda)}{40h}e^{\frac{2\imath\theta}{5}}\left(5e^{-\imath\theta}+e^{\imath\theta}\right),\\
\widetilde{a}_{1,3}^{Q3-Q2} =& \gamma\frac{e^{\frac{\imath\theta}{5}}}{40}\left(-5e^{\frac{-3\imath\theta}{2}}-9e^{\frac{-\imath\theta}{2}}+13e^{\frac{\imath\theta}{2}}+e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{a}_{2,1}^{Q3-Q2} =& -\gamma\frac{3(2\mu+\lambda)}{40h}e^{\frac{-2\imath\theta}{5}}\left(e^{-\imath\theta}+5e^{\imath\theta}\right),\\
\widetilde{a}_{2,2}^{Q3-Q2} =& \gamma\frac{6(2\mu+\lambda)}{h}\left(\frac{1}{5}-\frac{1}{8}\cos\,\theta\right),\\
\widetilde{a}_{2,3}^{Q3-Q2} =& \gamma\frac{e^{\frac{-\imath\theta}{5}}}{40}\left(-e^{\frac{-3\imath\theta}{2}}-13e^{\frac{-\imath\theta}{2}}+9e^{\frac{\imath\theta}{2}}+5e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{a}_{3,1}^{Q3-Q2} =& (\gamma-1)\frac{e^{\frac{-\imath\theta}{5}}}{40}\left(-e^{\frac{-3\imath\theta}{2}}-13e^{\frac{-\imath\theta}{2}}+9e^{\frac{\imath\theta}{2}}+5e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{a}_{3,2}^{Q3-Q2} =& (\gamma-1)\frac{e^{\frac{\imath\theta}{5}}}{40}\left(-5e^{\frac{-3\imath\theta}{2}}-9e^{\frac{-\imath\theta}{2}}+13e^{\frac{\imath\theta}{2}}+e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{a}_{3,3}^{Q3-Q2} =& \gamma\frac{2\kappa\tau(1-\frac{2}{3}\cos\,\theta-\frac{1}{3}\cos\,2\theta)}{\eta_f h}+(\gamma-1)\frac{h}{60(2\mu+\lambda)}\left(27-26\cos\,\theta-\cos\,2\theta\right).
\end{aligned}$$

For this matrix, if we make $\det\left(\widetilde{\mathcal{A}}_{Q3-Q2}\right)=0$, the two first obtained values for the amplification factor are $\gamma_1=\gamma_2=0$ since the first and second rows of $\widetilde{\mathcal{A}}^{Q3-Q2}$ are multiplied by γ. However, the third value $\gamma_3=0$ requires tough computations that

we can shortcut by means of an auxiliar matrix $\mathcal{J}$ defined as follows

$$\mathcal{J} = \begin{pmatrix} \frac{1}{\gamma}\widetilde{a}_{1,1}^{Q3-Q2} & \frac{1}{\gamma}\widetilde{a}_{1,2}^{Q3-Q2} & \frac{1}{\gamma}\widetilde{a}_{1,3}^{Q3-Q2} \\ \frac{1}{\gamma}\widetilde{a}_{2,1}^{Q3-Q2} & \frac{1}{\gamma}\widetilde{a}_{2,2}^{Q3-Q2} & \frac{1}{\gamma}\widetilde{a}_{2,3}^{Q3-Q2} \\ \frac{1}{(\gamma-1)}\widetilde{a}_{3,1}^{Q3-Q2} & \frac{1}{(\gamma-1)}\widetilde{a}_{3,2}^{Q3-Q2} & \frac{h}{60(2\mu+\lambda)}\left(27-26\cos\theta-\cos 2\theta\right) \end{pmatrix}.$$

Then, we have that

$$\gamma_3 = \frac{\det\left(J\right)}{\det\left(J\right) + 18\left(\frac{2\mu+\lambda}{h}\right)^2\left(2\left(\frac{8-5\cos\theta}{40}\right)^2 - \frac{13+5\cos(2\theta)}{1600}\right)\left(\frac{2\kappa\tau(1-\frac{2}{3}\cos\theta-\frac{1}{3}\cos 2\theta)}{\eta_f h}\right)}.$$

In the previous expression, $\det(J)$ and the second term in the denominator are non-negative $\forall\theta \in [-\pi,\pi]$. Consequently, we can claim that $|\gamma_3(\theta)| \leq 1, \forall\theta \in [-\pi,\pi]$ and therefore our decoupled method applied to Terzaghi's problem discretized with the Taylor-Hood pair $Q3-Q2$ (with global smoothness $\mathcal{C}^1$) is also stable. Thus, we can conclude that our decoupled method is stable for all the proposed discretizations.

At this point, we make use of the von Neumann analysis in order to prove that the previous schemes are also convergent. Hence, let us denote with $e^i_{u_j}$ the error computed at the $i-$th iteration of the decoupled solver for the $j-$th unknown of the displacements and $e^i_{p_j}$ is the respective error for the $j-$th pressure unknown at a given time t_m. Here, we assume that the errors at the $i-$th iteration of the decoupled scheme have the form

$$e^i_{u_j} = \gamma^i e^{\imath j\theta}\hat{e}_{u_j}, \qquad e^i_{p_j} = \gamma^i e^{\imath j\theta}\hat{e}_{p_j},$$

where γ is the error amplification factor. Hence, the error equations can be obtained for each discretization by following the procedure given in Kim, Tchelepi, and Juanes, 2011b. For the pair $Q1-Q1$, our stabilized iterative method yields the following error equations:

$$\mathcal{B}^{Q1-Q1}\begin{pmatrix}\hat{e}_u \\ \hat{e}_p\end{pmatrix} = \begin{pmatrix}0\\0\end{pmatrix}, \tag{6.63}$$

where

$$\mathcal{B}^{Q1-Q1} = \begin{pmatrix} \frac{2(2\mu+\lambda)}{h}\gamma(1-\cos\theta) & \imath\sin\theta \\ \imath\sin\theta & \gamma\left(\frac{2\kappa\tau}{\mu_f h}(1-\cos\theta)+\frac{3h}{2(\mu+2\lambda)}\right)-\frac{h}{2(2\mu+\lambda)}(2+\cos\theta) \end{pmatrix}.$$

If we make equal to zero the determinant of the previous matrix, we obtain that the first value for the error amplification factor is $\gamma_1 = 0$ and the second one is $\gamma_2 = 1/(3+4\kappa\tau(2\mu+\lambda)(1-\cos\theta)/h^2)$. In fact, $\lim_{\tau\to 0}\gamma_2 = 1/3 < 1$. Then, the previous scheme is convergent and the analysis claims that the error is reduced at least in a factor of $\gamma_2 = 1/3$ per iteration.

Now, we proceed to make the convergence analysis for our stabilized iterative method by using the Taylor-Hood elements $Q2-Q1$. In this case, the error equation is given by

$$\mathcal{B}^{Q2-Q1}\begin{pmatrix}\hat{e}_{u_v}\\ \hat{e}_{u_m}\\ \hat{e}_p\end{pmatrix} = \begin{pmatrix}0\\0\\0\end{pmatrix}, \tag{6.64}$$

where the matrix $\mathcal{B}^{Q2-Q1}$ of this system is given by

$$\begin{pmatrix} \gamma\frac{4(2\mu+\lambda)}{3h}(2-\cos\theta) & -\gamma\frac{4(2\mu+\lambda)}{3h}\cos\theta & \gamma\frac{2\imath}{3}\sin\theta \\ -\gamma\frac{4(2\mu+\lambda)}{3h}\cos\theta & \gamma\frac{4(2\mu+\lambda)}{3h} & \gamma\frac{2\imath}{3}\sin\frac{\theta}{2} \\ \frac{2\imath}{3}\sin\theta & \frac{2\imath}{3}\sin\frac{\theta}{2} & \gamma\left(\frac{2\tau\kappa(1-\cos\theta)}{\mu_f h}+\frac{h}{2\mu+\lambda}\right)-\frac{h(2+\cos\theta)}{3(2\mu+\lambda)} \end{pmatrix}.$$

In this error equation, the three values for the error amplification γ are equal to zero. That means that the iterative method is convergent and furthermore it needs theoretically only one iteration to converge.

Finally, the error equations for the isogeometric Taylor-Hood elements $Q3-Q2$ with global smoothness $\mathcal{C}^1$ are given by

$$\mathcal{B}^{Q3-Q2}\begin{pmatrix}\hat{e}_{u_v}\\ \hat{e}_{u_m}\\ \hat{e}_p\end{pmatrix}=\begin{pmatrix}0\\0\\0\end{pmatrix}, \tag{6.65}$$

where the elements $\{\widetilde{b}_{i,j}^{Q3-Q2}\}$, $i,j=1,2,3$ of the matrix $\mathcal{B}^{Q3-Q2}$ are

$$\begin{aligned}
\widetilde{b}_{1,1}^{Q3-Q2} &= \widetilde{a}_{1,1}^{Q3-Q2},\\
\widetilde{b}_{1,2}^{Q3-Q2} &= \widetilde{a}_{1,2}^{Q3-Q2},\\
\widetilde{b}_{1,3}^{Q3-Q2} &= \widetilde{a}_{1,3}^{Q3-Q2},\\
\widetilde{b}_{2,1}^{Q3-Q2} &= \widetilde{a}_{2,1}^{Q3-Q2},\\
\widetilde{b}_{2,2}^{Q3-Q2} &= \widetilde{a}_{2,2}^{Q3-Q2},\\
\widetilde{b}_{2,3}^{Q3-Q2} &= \widetilde{a}_{2,3}^{Q3-Q2},\\
\widetilde{b}_{3,1}^{Q3-Q2} &= \frac{e^{\frac{-\imath\theta}{5}}}{40}\left(-e^{\frac{-3\imath\theta}{2}}-13e^{\frac{-\imath\theta}{2}}+9e^{\frac{\imath\theta}{2}}+5e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{b}_{3,2}^{Q3-Q2} &= \frac{e^{\frac{\imath\theta}{5}}}{40}\left(-5e^{\frac{-3\imath\theta}{2}}-9e^{\frac{-\imath\theta}{2}}+13e^{\frac{\imath\theta}{2}}+e^{\frac{3\imath\theta}{2}}\right),\\
\widetilde{b}_{3,3}^{Q3-Q2} &= \gamma\left(\frac{2\kappa\tau(1-\frac{2}{3}\cos\theta-\frac{1}{3}\cos 2\theta)}{\mu_f h}+\frac{h}{2\mu+\lambda}\right)\\
&\qquad -\frac{h}{60(2\mu+\lambda)}(33+26\cos\theta+\cos 2\theta).
\end{aligned}$$

By imposing $\det(\mathcal{B}^{Q3-Q2})=0$, we obtain that the three values of the error amplification factor are $\gamma_1=\gamma_2=\gamma_3=0$. Hence, the von Neumann analysis for this pair predicts that the iterative method is convergent and, as it happened for the Taylor-Hood elements $Q2-Q1$, only one iteration is required to vanish the error.

We can conclude that the von Neumann analysis ensures us that the proposed iterative method with the pairs of elements $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ is stable and convergent. Moreover, the error amplification factor of this method for the pair $Q1-Q1$ is $1/3$ and the same iterative method needs only one iteration to converge for the pairs $Q2-Q1$ and $Q3-Q2$.

6.3.4 Numerical experiments

Now, we present some numerical experiments applying the iterative method with a mass stabilization described in Algorithm 7. By means of these experiments, we

show that the von Neumann analysis predicts pretty accurately the performance of our method with $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding global $\mathcal{C}^1$ smoothness) discretizations of Terzaghi's problem. Moreover, we apply the stabilized iterative method for the Biot's equations in a quarter of annulus in order to prove its efficiency for non-trivial geometries.

Terzaghi's problem:

First, we consider the Terzaghi's one-dimensional problem given in 6.47 in Section 6.2.1 with the initial and boundary conditions given in 6.48. We include the mass stabilization presented in Section 6.2.1 and we discretize this problem by using the isogeometric pairs of elements $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding global $\mathcal{C}^1$ smoothness). In order to show the efficiency of our proposed iterative solution method, in Table 6.1 we show the number of iterations of our mass stabilized iterative algorithm such that the residual res holds $\|\text{res}^i\|_\infty \leq 10^{-8}$, where res^i denotes the residual at the $i-$th iteration of the method. These results are shown for several values of the permeability κ and mesh size h, and considering the pairs $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding a global smoothness $\mathcal{C}^1$).

	$h=1/16$			$h=1/32$			$h=1/64$			$h=1/128$		
κ	$Q1Q1$	$Q2Q1$	$Q3Q2$	$Q1Q1$	$Q2Q1$	$Q3Q2$	$Q1Q1$	$Q2Q1$	$Q3Q2$	$Q1Q1$	$Q2Q1$	$Q3Q2$
10^{-6}	17	2	2	17	2	2	17	2	2	17	2	2
10^{-8}	17	2	2	17	2	2	17	2	2	17	2	2
10^{-10}	17	2	2	17	2	2	17	2	2	17	2	2
10^{-12}	17	2	2	17	2	2	17	2	2	17	2	2
10^{-14}	17	2	2	17	2	2	17	2	2	17	2	2

TABLE 6.1: Terzaghi's problem: Number of iterations of our mass stabilized iterative method such that the residual res holds $\|\text{res}^i\|_\infty \leq 10^{-8}$, where res^i denotes the residual at the $i-$th iteration of method, for several values of the permeability κ and mesh size h and by using the pairs $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding global $\mathcal{C}^1$ smoothness).

Note that the von Neumann analysis predicted that the amplification factor of the error was $1/3$ for the Q1-Q1 pair of elements whereas it was exact for $Q2-Q1$ and $Q3-Q2$ with global $\mathcal{C}^1$ smoothness. Hence, the theoretical number of required iterations to reduce the error in a factor of 10^{-8} coincide with the 17 iterations obtained for $Q1-Q1$. Also, the pairs $Q2-Q1$ and $Q3-Q2$ need theoretically 1 iteration only but in practice an additional iteration seems to be needed.

Quarter of an annulus:

For our second numerical experiment, we consider the two-field formulation of the Biot's equations given in (6.31)-(6.32) on the quarter of an annulus described in Figure 2.3, that is, $\Omega = \{(x,y) \in \mathbb{R}^2 \,|\, r^2 \leq x^2+y^2 \leq R^2, x, y \geq 0\}$, with $r = 0.5$ and $R = 1$. For discretization of this problem, the isogeometric pair of $Q3-Q2$ with global smoothness $\mathcal{C}^1$ is required for parametrization of the physical domain Ω. Let us consider that our domain is split in three subdomains $\Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3$ such that $\Omega_1 = \{(x,y) \in \mathbb{R}^2 \,|\, (1/2)^2 \leq x^2+y^2 \leq (2/3)^2, x, y \geq 0\}$ and $\Omega_3 = \{(x,y) \in \mathbb{R}^2 \,|\, (5/6)^2 \leq x^2+y^2 \leq 1, x, y \geq 0\}$ are highly permeable with $\kappa = 10^{-2}$ whereas $\Omega_2 = \Omega \setminus \Omega_1 \setminus \Omega_3$ is low-permeable with $\kappa = 10^{-7}$. In addition, we consider the parameter values $E = 3 \cdot 10^4$, $\nu = 0.2$, $\eta_f = 10^{-3}$ and the following initial and

boundary conditions:

$$\begin{aligned} \boldsymbol{u} = \mathbf{0}, \quad (\nabla p)\cdot \boldsymbol{n} = 0, \quad &\text{in } \Gamma \setminus \Gamma_2, \\ \sigma \cdot \boldsymbol{n} = 10^4(\cos\theta, \sin\theta), \quad p = 0, \quad &\text{in } \Gamma_2, \end{aligned} \tag{6.66}$$

where $\Gamma = \partial\Omega$ and $\Gamma_2 = \{(x,y) \in \mathbb{R}^2 \,|\, x^2 + y^2 = (1/2)^2, x, y \geq 0\}$. In Figure 6.4, we show these boundary conditions and regions Ω_1, Ω_2 and Ω_3.

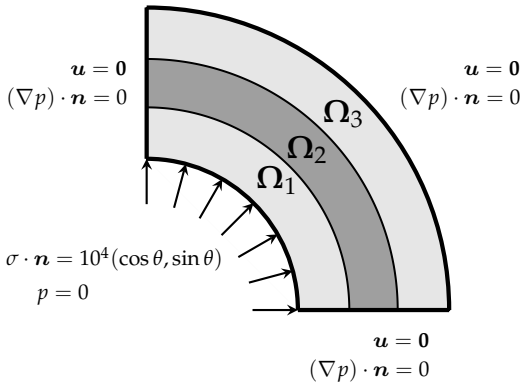


FIGURE 6.4: Quarter of an annulus: Boundary conditions and regions Ω_1, Ω_2 and Ω_3.

In order to illustrate the stabilization effect, in Figure 6.5 we show the numerical approximation of pressure p (a) without and (b) with mass-stabilization at $t = 10^{-6}$. We can observe that the extension of the stabilization term to the 2D case also seems to remove the oscillations very efficiently. Finally, in Table 6.2 we show the number of iterations of the proposed decoupled solution method by using the pair $Q3 - Q2$ stabilized such that the residual res^i in the $i-$th iteration holds $\|\text{res}^i\|_\infty \leq 10^{-8}$, for several mesh sizes h with time step $\tau = 10^{-2}$.

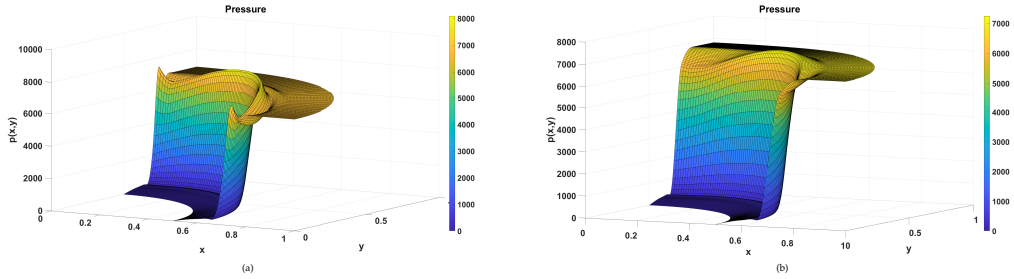


FIGURE 6.5: Quarter of an annulus: Numerical approximation of pressure (a) without and (b) with mass-stabilization at $t = 10^{-6}$.

τ	$h = 1/16$	$h = 1/32$	$h = 1/64$	$h = 1/128$
10^{-2}	9	9	8	7

TABLE 6.2: Quarter of an annulus: Number of iterations of the proposed decoupled solution method by using the pair $Q3 - Q2$ stabilized such that $\|\text{res}^i\|_\infty \leq 10^{-8}$, for several mesh sizes h and time step $\tau = 10^{-2}$.

6.4 Monolithic solvers: Multigrid methods

Fully coupled methods solve all the unknowns of the PDE system simuntaneously. With this approach, multigrid methods applied to PDE systems are called monolithic multigrid methods. Given that multigrid methods are fast and highly efficient solvers, they have been widely applied to poroelasticity problems in the literature, see Alpak and Wheeler, 2012; Luo et al., 2015; Rodrigo, 2016; Gaspar and Rodrigo, 2017. As mentioned in Chapter 3, the performance of multigrid methods depends strongly on their components. For transfer operators we consider the canonical restriction and prolongation operators. On the other hand, the choice of suitable smoothers is not direct and requires special attention. In fact, the proposed smoothers for scalar problems can not be applied straightforwardly for saddle point problems. There are two different types of smoothing schemes for PDE systems: Coupled or Schwarz-type smoothers and decoupled smoothers.

At this point, we introduce both types of smoothers for monolithic multigrid methods applied to the quasi-static consolidation Biot's model and show some numerical results in order to prove their efficiency and robustness with respect the mesh size h and physical parameters.

6.4.1 Coupled smoothers

This type of smoothers is based on coupled relaxation of the unknowns. Coupled or Schwarz-type smoothers are specially important for PDE systems whose discretization yields saddle point problems. Moreover, coupled smoothers must be adapted to the algebraic equations of the problem and therefore the relaxation scheme depends on the type of discretization. The classical Schwarz smoother for poroelastic saddle-point problems is typically defined as a set of blocks that consist of one pressure unknown and all the displacement unknowns that are connected to it. That is, the degrees of freedom corresponding to the nonzero entries in the $i-$th row of B plus the i-th pressure degree of freedom are simuntaneously relaxed. In the case of the $Q2-Q1$ Taylor-Hood elements, this yields to solve local systems of size 51×51 for two-dimensional problems.

Due to the high computational cost of this type of smoothers, the paralellization of the relaxation method becomes crucial for real applications. The parallelization of the multiplicative version of this smoother is a non-trivial task, and therefore the additive and the restricted additive Schwarz smoothers become a natural alternative to the former. However, the analysis of additive Schwarz methods applied to saddle point problems has not received a lot of attention in the literature opposite to that of their multiplicative counterpart, see Schöberl and Zulehner, 2003; Farrell, He, and MacLachlan, 2020.

For the design of monolithic multigrid methods for Biot's equations, we consider an LFA based on infinite subgrids (see Section 4.4). We focus on the isogeometric pair of $Q2-Q1$ Taylor-Hood elements. For this case, each basis function for the pressure shares common support with 5 displacement unknowns on each direction and therefore the minimum window size required for the analysis will be 8×8. In Figure 6.6, we show the periodic extension of this window with a selected block of unknowns in blue corresponding to the up-left pressure unknown.

We consider the two-field formulation given in (6.31)-(6.32) in the square domain $\Omega = (0,1)^2$ with $\Gamma_t = (0,1) \times \{1\}$, $\Gamma_c = \Gamma \setminus \Gamma_t$, $\boldsymbol{t} = -10^4$ and $1/M \approx 0$. For our experiments, we fix the Lamé parameters in such a way that the Young's modulus is $E = 3 \cdot 10^4$, and the Poisson's ratio is $\nu = 0.2$ and we consider different values

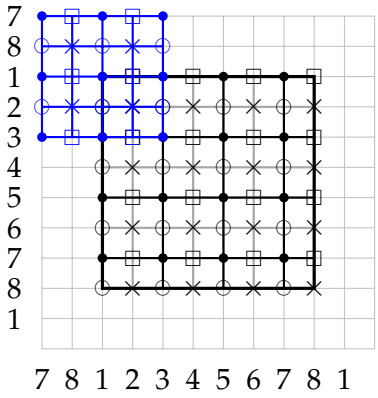


FIGURE 6.6: Two-grid analysis for the consolidation Biot's model: Considered window with minimum size. In blue, the block centered in the up-left pressure unknown is shown.

for constant permeability κ, ranging from $\kappa = 1$ to $\kappa = 1e-15$. First, we test the multigrid algorithm based on additive Schwarz smoothers considering the natural weights, i.e., $w = 1$ for the pressure and $w = 1/9, 1/6$ or $1/4$ depending if the degree of freedom of the displacement corresponds to a vertex, to an edge or a cell node, respectively. In Table 6.3, we show the asymptotic convergence factors provided by our analysis (ρ_{2g}) together with the ones provided by our multigrid codes by using W-cycles (ρ_h), considering different numbers of smoothing steps and different values of permeability for the $Q2-Q1$ discretization on the unit square domain.

	$W(1,0)$		$W(1,1)$		$W(2,1)$		$W(2,2)$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.49	0.49	0.25	0.22	0.12	0.11	0.06	0.06
10^{-3}	0.49	0.49	0.25	0.21	0.12	0.11	0.06	0.06
10^{-6}	0.49	0.49	0.25	0.21	0.12	0.11	0.06	0.06
10^{-9}	0.65	0.65	0.42	0.42	0.27	0.27	0.18	0.18
10^{-12}	0.72	0.72	0.52	0.52	0.52	0.56	0.28	0.28
10^{-15}	0.72	0.72	0.52	0.52	0.52	0.56	0.28	0.28

TABLE 6.3: Quasi-static Biot's model: Asymptotic convergence factors provided by our analysis (ρ_{2g}) together with the ones provided by our multigrid codes (ρ_h) considering several permeability values and different numbers of smoothing steps of our 51−point additive Schwarz relaxation with natural weights for the $Q2-Q1$ discretization on the unit square domain.

In the previous table, we can observe that there is a good match between the factors provided by LFA and the ones experimentally obtained with the multigrid codes. Besides, we can see that the performance of the multigrid method deteriorates when κ tends to zero, yielding a non-robust algorithm. An additional advantage of our analysis is that we can find the optimal weights for the additive Schwarz smoother for a fixed permeability in order to obtain a robust algorithm. Since in real applications the permeability is heterogeneous, we have found the following weights in order to have a robust method:

$$w_u^{opt} = \begin{pmatrix} 0.0484 & 0.0924 & 0.0484 & 0.0924 & 0.0484 \\ 0.0924 & 0.1764 & 0.0924 & 0.1764 & 0.0924 \\ 0.0484 & 0.0924 & 0.0484 & 0.0924 & 0.0484 \\ 0.0924 & 0.1764 & 0.0924 & 0.1764 & 0.0924 \\ 0.0484 & 0.0924 & 0.0484 & 0.0924 & 0.0484 \end{pmatrix},$$

for the 25 unknowns of each displacement component and $w_p^{opt} = 1.02$ for the pressure considered on each block. In Table 6.4, we show the asymptotic convergence factors obtained by applying the additive Schwarz smoother with these optimal weights provided by our analysis, considering different number of smoothing steps and several permeability values.

κ	$W(1,0)$	$W(1,1)$	$W(2,1)$	$W(2,2)$
1	0.58	0.34	0.19	0.11
10^{-3}	0.58	0.34	0.19	0.11
10^{-6}	0.58	0.34	0.19	0.11
10^{-9}	0.58	0.34	0.19	0.11
10^{-12}	0.60	0.36	0.21	0.13
10^{-15}	0.60	0.36	0.21	0.13

TABLE 6.4: Quasi-static Biot's model: Asymptotic convergence factors (ρ_{2_g}), considering several permeability values and different number of smoothing steps of the $51-$point additive Schwarz smoother with optimal weights w_u^{opt}, w_p^{opt}.

Given that $V-$cycles are less expensive than $W-$cycles, we would like to compare their performance in number of iterations. Hence, in Table 6.5, we show the number of iterations of our multigrid codes required to reduce the initial residual in a factor of 10^{-10}, using the optimal weights with $V-$ and $W-$cycles for several smoothing steps and permeabilities. We conclude that $V-$cycles are also robust with respect to κ when applying $V(2,1)$ or $V(2,2)-$cycles yielding a similar number of iterations than $W-$cycles in order to reach the stopping criterion. Thus, the use of $V(2,2)-$cycles with the $51-$point additive Schwarz smoother applying the optimal weights w_u^{opt}, w_p^{opt} is a good alternative in order to solve the two-field formulation of Biot's equations.

	Smoothing steps (ν_1, ν_2)							
	$(1,0)$		$(1,1)$		$(2,1)$		$(2,2)$	
κ	V	W	V	W	V	W	V	W
1	39	40	20	20	13	14	10	10
10^{-3}	40	40	20	20	14	13	10	10
10^{-6}	39	40	20	20	13	14	10	10
10^{-9}	40	40	23	20	14	13	10	10
10^{-12}	50	43	40	22	18	15	12	12
10^{-15}	50	45	40	23	18	15	12	12

TABLE 6.5: Quasi-static Biot's model: Number of iterations of our multigrid codes required to reduce the initial residual in a factor of 10^{-10}, using the optimal weights w_u^{opt}, w_p^{opt} with $V-$ and $W-$cycles for several smoothing steps and permeabilities.

6.4.2 Decoupled smoothers

As a second type of smoothers suitable for saddle point problems we are going to propose some decoupled smoothers. These smoothers relax separately the variables involved in the PDE system. For instance, decoupled smoothers can be based on decoupled solvers mentioned in Section 6.3 for the Biot's equations. Hence, we propose the use of decoupled smoothers based on the fixed-stress split algorithm given its unconditional stability. By applying directly the fixed-stress splitting method as smoother, the new approximation (u_m, p_m) is obtained with the following relaxation procedure:

$$\begin{bmatrix} u_m \\ p_m \end{bmatrix} = \begin{bmatrix} u_{m-1} \\ p_{m-1} \end{bmatrix} + \mathcal{M}_A^{-1} \left(\begin{bmatrix} \mathbf{f} \\ f \end{bmatrix} - \mathcal{N}_A \begin{bmatrix} u_{m-1} \\ p_{m-1} \end{bmatrix} \right), \tag{6.67}$$

where

$$\mathcal{M}_A = \begin{pmatrix} E_l & B^T \\ 0 & \tau C + LM_p \end{pmatrix}, \qquad \mathcal{N}_A = \begin{pmatrix} 0 & 0 \\ B & LM_p \end{pmatrix}.$$

However, solving exactly these linear systems could be very costly, specially on fine grids. Thus, instead of computing $\mathcal{M}_A^{-1}$, an approximation can be obtained by applying separately appropriate smoothers for displacements $\boldsymbol{u}$ and pressure p. In other terms, we propose the following inexact version of fixed-stress splitting method as smoother:

$$\widetilde{\mathcal{M}} = \begin{bmatrix} M_{E_l} & B^T \\ 0 & M_S \end{bmatrix}, \tag{6.68}$$

where M_{E_l} and M_S are suitable smoothers for operators E_l and $S = \tau C + LM_p$. Applying this scheme, we obtain a decoupled smoother based on an inexact version of the fixed-stress split algorithm that gives rise to the following iterative method:

$$\begin{cases} p_m = p_{m-1} + M_S^{-1} \left(f - Bu_{m-1} + Cp_{m-1} \right), \\ u_m = u_{m-1} + M_{E_l}^{-1} \left(\boldsymbol{f} - E_l u_{m-1} - B^T p_m \right). \end{cases} \tag{6.69}$$

At this point, the choice of the smoothers M_{E_l} and M_S and the number of smoothing iterations for each operator is a crucial point. As observed in Chapter 5, pointwise smoothers do not yield robust neither efficient multigrid solvers when high-order spline discretizations are considered for discretization of Poisson's problem. Thus, we also propose the use of block-wise smoothers for quasi-static Biot's model equations. More precisely, additive Schwarz methods seem to be adequate candidates for this task given their efficiency and parallelizability. As a starting point, we consider a $50-$point additive Schwarz method for the displacements and a $9-$point additive Schwarz method for the pressure. In order to provide a theoretical analysis for this method, we carried out an LFA based on infinite subgrids for the problem (6.37) considered in the numerical experiments of Section 6.4.1. Thus, the Fourier symbol of the fixed-stress split algorithm as smoother is given by:

$$\widetilde{S}_{fs}(\theta) = - \left(\widetilde{M}_{\mathcal{A}}(\theta) \right)^{-1} \widetilde{N}_{\mathcal{A}}(\theta), \tag{6.70}$$

where $\widetilde{M}_{\mathcal{A}}$ and $\widetilde{N}_{\mathcal{A}}$ are the Fourier symbols of the matrices $M_{\mathcal{A}}$ and $N_{\mathcal{A}}$ respectively. The Fourier symbol of these matrices is

$$\widetilde{M}_{\mathcal{A}}(\theta) = \begin{pmatrix} \widetilde{E}_l(\theta) & \widetilde{B^T}(\theta) \\ 0 & \tau\widetilde{C}(\theta) + L\widetilde{M}_p(\theta) \end{pmatrix}, \qquad \widetilde{N}_{\mathcal{A}} = \begin{pmatrix} 0 & 0 \\ \widetilde{B}(\theta) & L\widetilde{M}_p(\theta) \end{pmatrix},$$

where $\widetilde{E}_l(\theta), \widetilde{B^T}(\theta), -\widetilde{B}(\theta), \widetilde{C}(\theta)$ and $\widetilde{M}$ are the Fourier symbols of the elasticity, gradient, divergence, laplacian (multiplied by κ/η_f) and mass operators. However, when we substitute the fixed-stress split scheme by an inexact version where suitable smoothers M_{E_l} for the displacements and M_S for the pressure are employed, the Fourier symbol of the resulting smoother is given by

$$\widetilde{S}_{ifs}(\theta) = \widetilde{M}_{approx}(\theta)\left[\begin{pmatrix} \widetilde{E}_l(\theta) & \widetilde{B^T}(\theta) \\ -\widetilde{B}(\theta) & \tau\widetilde{C}(\theta) \end{pmatrix} - \widetilde{M}_{approx}(\theta)\right], \tag{6.71}$$

where the matrix $\widetilde{M}_{approx}$ is given by

$$\widetilde{M}_{approx}(\theta) = \begin{pmatrix} \widetilde{E}_l(\theta)\left(\widetilde{I}(\theta) - \widetilde{M}_{E_l}^{\nu_u}(\theta)\right)^{-1} & \left(\widetilde{B^T}(\theta)\right)^{-1} \\ 0 & \left(\tau\widetilde{C}(\theta) + L\widetilde{M}_p(\theta)\right)\left(\widetilde{I}(\theta) - \widetilde{M}_S^{\nu_p}(\theta)\right)^{-1} \end{pmatrix},$$

and ν_u, ν_p denote the number of smoothing steps of the smoothers M_{E_l}, M_S for the displacements and pressure in order to construct the inexact version of the fixed-stress split algorithm as smoother. Note that the rest of the operators for the $2-$grid analysis of the Biot's equations by using the inexact version of the fixed-stress split method as smoother are obtained as explained in 4.56. We consider the two-field formulation of the Biot's model given in (6.31)-(6.32) discretized with the pairs of elements $Q2-Q1$ and $Q3-Q2$ (holding global smoothness $\mathcal{C}^1$). Hence, in Table 6.6 we show the corresponding asymptotic convergence factors of our multigrid code based on $W(1,0)$-cycles with the inexact version of fixed-stress as smoother, with one iteration of $9-$point additive Schwarz for the pressure and one iteration of $50-$point additive Schwarz for displacements $\boldsymbol{u}$ using natural weights, for different values of the permeability κ.

	$Q2-Q1$		$Q3-Q2$ $\mathcal{C}^1$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.3136	0.3110	0.5937	0.5895
$1e-3$	0.3136	0.3110	0.5937	0.5895
$1e-6$	0.3136	0.3109	0.5937	0.5895
$1e-9$	0.3894	0.3853	0.7094	0.7152
$1e-12$	0.4996	0.4948	0.8442	0.8486
$1e-15$	0.5003	0.5010	0.8444	0.8516

TABLE 6.6: Two-level (ρ_{2g}) convergence factors predicted by LFA together with the asymptotic convergence factors obtained numerically (ρ_h) using $W(1,0)$-cycles with the inexact version of fixed-stress as smoother, with one iteration of $9-$point additive Schwarz for the pressure and one iteration of $50-$point additive Schwarz for displacements $\boldsymbol{u}$ using natural weights. These results are shown for different values of the permeability κ applying the $Q2-Q1$ and $Q3-Q2$ (holding global smoothness $\mathcal{C}^1$) pairs of elements.

In the previous table, we observe that there is a good match between the asymptotic convergence factors provided by the analysis and the ones obtained with our multigrid codes. However, there is a deterioration of the asymptotic convergence factors when we deal with low permeabilities and the convergence factors are not very satisfactory with this approach. In order to improve the performance of our monolithic multigrid solver based on an inexact version of the fixed-stress split method as smoother, we propose to raise the number of iterations of the $50-$point additive Schwarz smoother for the displacements to two iterations. Thus, in Table 6.7 we show the corresponding asymptotic convergence factors of our multigrid code based on $W(1,0)$-cycles with the inexact version of fixed-stress as smoother, with one iteration of $9-$point additive Schwarz for the pressure p and one iteration of $50-$point additive Schwarz for displacements $\boldsymbol{u}$.

	$Q2-Q1$		$Q3-Q2\ \mathcal{C}^1$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.2736	0.2746	0.4857	0.4849
$1e-3$	0.2736	0.2781	0.4857	0.4848
$1e-6$	0.2703	0.2777	0.4859	0.4852
$1e-9$	0.1914	0.1904	0.5692	0.5691
$1e-12$	0.4191	0.4153	0.7013	0.7026
$1e-15$	0.4192	0.4158	0.7016	0.7030

TABLE 6.7: Two-level (ρ_{2g}) convergence factors predicted by LFA together with the asymptotic convergence factors obtained numerically (ρ_h) using $W(1,0)$-cycles with the inexact version of fixed-stress as smoother, with one iteration of $9-$point additive Schwarz for the pressure p and two iterations of $50-$point additive Schwarz for displacements $\boldsymbol{u}$ using natural weights. These results are shown for different values of the permeability κ applying the $Q2-Q1$ and $Q3-Q2$ (holding global smoothness $\mathcal{C}^1$) pairs of elements.

Again, a good match between the asymptotic factors provided by our multigrid codes an the analysis is observed. Although the asymptotic convergence factors have been improved, we consider that the performance of this solver based on $W(1,0)-$cycles with the inexact version of fixed-stress as smoother is not good enough. At this point, one way to get better results is to add a post-smoothing step of the smoother. Hence, we show the corresponding asympotic convergence factors in Table 6.8:

Table 6.8 provides more satisfying results for both discretizations of the Biot's equations by using iterations of the $W(1,1)$-cycle based on the inexact version of the fixed-stress split algorithm as smoother. However, there is still some deterioration of the asymptotic convergence factor and it would be desirable to obtain a method with better convergence rates. For this purpose, it seems natural to apply different smoothing strategies for each discretization and to increase the number of smoothing steps. Fortunately, LFA helps us again to design our multigrid methods based on this type of smoother. Thus, we proceed to analyze some smoothing strategies for the pair $Q2-Q1$ and then do the same for the pair $Q3-Q2$ holding global smoothness $\mathcal{C}^1$. Hence, we start analyzing the performance of different smoothing strategies for the $Q2-Q1$ Taylor-Hood elements. First, we propose to apply the inexact version of the fixed-stress split algorithm as smoother by using the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p respectively. In Table 6.9 we show the asymptotic convergence factors provided by a two-grid analysis (ρ_{2g})

	$Q2-Q1$		$Q3-Q2$ $\mathcal{C}^1$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.0904	0.0912	0.2359	0.2350
$1e-3$	0.0904	0.0907	0.2359	0.2350
$1e-6$	0.0888	0.0901	0.2361	0.2340
$1e-9$	0.0365	0.0434	0.3236	0.3197
$1e-12$	0.1749	0.1727	0.4911	0.4954
$1e-15$	0.1755	0.1732	0.4916	0.4956

TABLE 6.8: Two-level (ρ_{2g}) convergence factors predicted by LFA together with the asymptotic convergence factors obtained numerically (ρ_h) using $W(1,1)$-cycles with the inexact version of fixed-stress as smoother, with one iteration of $9-$point additive Schwarz for the pressure p and two iterations of $50-$point additive Schwarz for displacements $\boldsymbol{u}$ using natural weights. These results are shown for different values of the permeability κ applying the $Q2-Q1$ and $Q3-Q2$ (holding global smoothness $\mathcal{C}^1$) pairs of elements.

and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,1)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with different numbers of iterations ν_u, ν_p of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p.

	$W(1,1)-$cycles with $18-$ and $9-$point additive Schwarz.					
	$\nu_u=1,\nu_p=1$		$\nu_u=2,\nu_p=1$		$\nu_u=2,\nu_p=2$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.2116	0.2502	0.0904	0.0657	0.0457	0.0657
$1e-12$	0.5203	0.5147	0.1391	0.1270	0.1122	0.1207

TABLE 6.9: $Q2-Q1$: Asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,1)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with different numbers of iterations ν_u, ν_p of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p.

In Table 6.9, we can observe that using $\nu_u = 1, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods for the displacements and pressure respectively on the inexact version of the fixed-stress stress split algorithm as smoother is not enough to achieve robustness and satisfying results. On the other hand, the approaches with $\nu_u = 2, \nu_p = 1$ and $\nu_u = 2, \nu_p = 2$ provide a good performance and they do seem to be robust with respect the permeability κ. However, the performance of these two approaches is very similar. Due to this, although the results provided by using $\nu_u = 2, \nu_p = 2$ are slightly better, it does not look to be worth the use of two smoothing steps for the pressure. Therefore, among the proposed inexact versions of the fixed-stress split algorithm as smoother for the pair $Q2-Q1$ we propose to use $\nu_u = 2, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p respectively. Moreover, we are interested on how the performance of this approach is by applying different numbers of smoothing steps. Thus, in Table 6.10 we show asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,0)-$, $W(1,1)-$, $W(2,1)-$, and $W(2,2)-$cycles with the inexact version of the fixed-stress split algorithm as

smoother with $\nu_u = 2, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p.

	$Q2-Q1, \nu_u=2, \nu_p=1.$							
	$W(1,0)$		$W(1,1)$		$W(2,1)$		$W(2,2)$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.2736	0.2516	0.0904	0.0657	0.0205	0.0228	0.0256	0.0262
$1e-12$	0.3745	0.3708	0.1391	0.1270	0.0516	0.0400	0.0191	0.0253

TABLE 6.10: $Q2-Q1$: Asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,0)-$, $W(1,1)-$, $W(2,1)-$, and $W(2,2)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with $\nu_u = 2, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p.

At this point, we propose as optimal choice the use of $W(2,1)-$cycles with $\nu_u = 2, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods within the inexact fixed-stress split smoother since the performance of $W(2,2)-$cycles is only slightly better than the one provided by using $W(2,1)-$cycles. In addition, we base our optimal choice of $W(2,1)-$cycles on the fact that this approach provide a robust factor with respect to the permeability κ more clearly than the use of $W(1,1)-$cycles.

Now, we focus on the search of the optimal smoothing strategy for the $Q3-Q2$ isogeometric Taylor-Hood elements by using additive Schwarz smoothers on the inexact version of the fixed-stress split algorithm as smoother. For this purpose, we started by considering $\nu_u = 1, \nu_p = 1$ iterations of the $18-$ and $9-$point additive Schwarz methods for the displacements and the pressure. As we can observe in Table 6.11 by applying $W(1,1)-$cycles with different numbers of iterations ν_u and ν_p, the performance of the fixed-stress split algorithm as smoother with the $18-$ and $9-$point additive Schwarz methods for the displacements and the pressure seems to be too poor.

	$W(1,1)-$cycles of $18-$ and $9-$point additive Schwarz.					
	$\nu_u=1, \nu_p=1$		$\nu_u=2, \nu_p=1$		$\nu_u=2, \nu_p=2$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.5905	0.5768	0.3488	0.3333	0.3488	0.3333
$1e-12$	–	–	0.5350	0.5653	0.5333	0.5378

TABLE 6.11: $Q3-Q2$: Asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,1)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with several numbers of iterations ν_u, ν_p of the $18-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p, respectively

Given that the $18-$point additive Schwarz smoother has not enough smoothing effect for the displacements, we consider the inexact version of the fixed-stress split algorithm as smoother with the $50-$point additive Schwarz method for $\boldsymbol{u}$ and the $9-$point additive Schwarz method for p. Thus, in Table 6.12 we show asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,1)-$cycles with the inexact version of the fixed-stress

split algorithm as smoother with several numbers of iterations ν_u, ν_p of the $50-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p, respectively.

	$W(1,1)-$cycles of $50-$ and $9-$point additive Schwarz.							
	$\nu_u = 1, \nu_p = 1$		$\nu_u = 2, \nu_p = 1$		$\nu_u = 3, \nu_p = 1$		$\nu_u = 2, \nu_p = 2$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.3530	0.3365	0.2359	0.2183	0.2359	0.2333	0.1247	0.1203
$1e-12$	0.7195	–	0.4911	0.4835	0.5206	0.4907	0.1773	0.1749

TABLE 6.12: $Q3-Q2$: Asymptotic convergence factors provided by a two-grid analysis (ρ_{2g}) and the ones obtained experimentally with our multigrid codes (ρ_h) for a high and low value of the permeability κ by using $W(1,1)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with several numbers of iterations ν_u, ν_p of the $50-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p, respectively.

In the previous table, we observe that the performance of our solver is satisfactory when we consider $W(1,1)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with $\nu_u = \nu_p = 2$ iterations of the $50-$ and $9-$point additive Schwarz methods for $\boldsymbol{u}$ and p, respectively. Note that besides it seems to be very efficient, it shows robustness with respect the permeability κ. Now, we would like to know how the performance of this solver is if we change the number of smoothing steps. Thus, in Table 6.13 we show the asymptotic convergence factors provided by the analysis and our multigrid codes by applying $W(1,0)-$, $W(1,1)-$, $W(2,1)-$ and $W(2,2)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with $\nu_u = 2, \nu_p = 2$ iterations of the $50-$ and $9-$point additive Schwarz methosd for $\boldsymbol{u}$ and p:

	$50-$ and $9-$point additive Schwarz methods: $\nu_u = 2, \nu_p = 2$							
	$W(1,0)$		$W(1,1)$		$W(2,1)$		$W(2,2)$	
κ	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h	ρ_{2g}	ρ_h
1	0.3530	0.3377	0.1247	0.1203	0.0440	0.0426	0.0156	0.0142
$1e-12$	0.4191	0.5035	0.1773	0.1749	0.0752	0.0815	0.0465	0.0320

TABLE 6.13: $Q3-Q2$: Asymptotic convergence factors provided by the analysis and our multigrid codes by applying $W(1,0)-$, $W(1,1)-$, $W(2,1)-$ and $W(2,2)-$cycles with the inexact version of the fixed-stress split algorithm as smoother with $\nu_u = 2, \nu_p = 2$ iterations of the $50-$ and $9-$point additive Schwarz methosd for $\boldsymbol{u}$ and p.

At this point, we observe that a good performance of the inexact version of the fixed-stress split algorithm as smoother for the pair $Q3-Q2$ is achieved when we apply $W(1,1)-$, $W(2,1)-$ or $W(2,2)-$cycles with $\nu_u = 2, \nu_p = 2$ iterations of the $50-$ and $9-$point additive Schwarz smoothers to approximate the solution of $\boldsymbol{u}$ and p respectively. Besides a very good asymptotic convergence factor, we obtain a robust method with respect the permeability κ with these approaches.

Chapter 7

Conclusions and future work

In this work, efficient and robust multigrid methods have been proposed for IGA discretizations of several problems that arise in physics and engineering. In order to do this, the basics of the parametric construction of curves and surfaces have been introduced and then the isogeometric analysis of some problems by means of B-splines and NURBS as well. After this, the basics of multigrid methods and local Fourier analysis were stablished with the purpose of giving a useful tool for the design of multigrid methods. Then, we devoted our attention to the design of efficient and robust multigrid methods for some scalar problems. Finally, the main contribution of this thesis is to propose efficient and robust multigrid methods with respect the spline degree k for isogeometric discretizations of the poroelastic Biot's equations.

The first important contribution of this work is to propose a robust and efficient multigrid method for isogeometric discretizations of the Poisson equation holding maximum global smoothness $\mathcal{C}^{k-1}$ based on multiplicative Schwarz methods as smoothers with maximum overlapping among the blocks. For this purpose, we carry out an LFA for block-wise smoothers for IGA discretizations. In fact, thanks to this analysis we are able to set the most appropriate size of the blocks for every polynomial degree until $k = 8$. All these contributions are published in Pé de la Riva, Rodrigo, and Gaspar, 2019 and this research on the Poisson equation constitutes our initial step towards the study of the Biot's equations.

In order to analyze the efficiency and robustness with respect the polynomial degree k of the previous approach for high-order PDEs, we have designed a multigrid method based on multiplicative Schwarz smoothers for IGA discretizations of the bilaplacian equation. Again, LFA has helped us with the design of this method and we have achieved a robust method for the biharmonic equation, see Pé de la Riva, Rodrigo, and Gaspar, 2020.

Another contribution derived of this work is an improvement of the multigrid method previously mentioned for Poisson equation, based on a two-level method that reduces the computational cost of the solution method. As described in Pé de la Riva, Rodrigo, and Gaspar, 2021, this method considers a fine level with spline degree k and mesh size h whereas a low polynomial degree k_{low} is considered at the second level with the same mesh size. This method has been improved by considering an approximation for the solution of the defect equation at the second level. Our proposed approximation consists of applying one iteration of one $h-$multigrid method and the efficiency of this approach has been improved by considering a more aggresive coarsening (spline degree k_{low} and mesh size $H = 2h$) in the second level. Moreover, we would like to remark that the design of this method is supported by LFA.

The next contribution of this thesis deals with the study of additive Schwarz smoothers instead of considering the previously mentioned multiplicative Schwarz

iterations. Our idea here was to make a good use of the parallelizability of this type of smoother in order to design even more efficient solvers. Since a standard LFA is not suitable to additive Schwarz methods, we have carried out a recent LFA based on infinite subgrids for this case. This approach has been considered for scalar problems and systems of saddle point type. We are currently working on the publication of them in a joint work with professors Ludmil Zikatanov, Xiaozhe Hu, James H. Adler, Carmen Rodrigo and Francisco J. Gaspar.

We have introduced the Biot's model and explained in detail its constitutive equations. In this work, we have considered the two-field or displacements-pressure formulation of the Biot's equations. Furthermore, we have proposed an innovative mass stabilization that eliminates all the spurious oscillations of the pressure. In fact, this stabilization was firstly introduced in Pé de la Riva et al., 2022 for FEM. As pointed out in that paper, this mass stabilization enables the automatic application of an innovative iterative decoupled method and we have applied it successfully to the isogeometric pairs of elements $Q1-Q1$, $Q2-Q1$ and $Q3-Q2$ (holding global smoothness $\mathcal{C}^1$). In order to analyze the stability and convergence of the proposed method, we performed a von Neumann analysis for the previous pairs of elements. In addition, some numerical experiments for Terzaghi's problem and Biot's equations demonstrate the good performance of this approach.

The main goal of this thesis is to design efficient and robust multigrid methods for isogeometric discretizations applied to poroelasticity problems. We have considered monolithic multigrid methods for Biot's equations with coupled and decoupled smoothers. More concretely, we have proposed coupled Schwarz additive smoothers and an inexact version of the fixed-stress split algorithm as decoupled smoother. The design of multigrid methods based on these smoothers has been guided by the predictions of an LFA based on infinite subgrids. This analysis for poroelasticity equations has allowed us to find robust and efficient monolithic multigrid methods based on these smoothers. In fact, a $51-$point coupled additive Schwarz smoother has been proposed for the isogeometric pair $Q2-Q1$. Regarding the inexact version of the fixed-stress split algorithm as smoother, we proposed to use $18-$ and $9-$ point additive Schwarz smoothers for the approximation of displacements and pressure, respectively, by using elements $Q2-Q1$. In the case of $Q3-Q2$ with global smoothness $\mathcal{C}^1$, $50-$ and $9-$ point additive Schwarz smoothers were considered for the approximation of displacements and pressure. Finally, by means of some numerical experiments, we demonstrated the efficiency and robustness with respect the permeability κ of the proposed multigrid solvers.

The field of multigrid methods for isogeometric discretizations applied to poroelasticity problems is still very unexplored, this opens a wide range of topics for future research. Although we considered the two-field formulation in this thesis, IGA has not been applied to other formulations of Biot's equations such as the classical three-field formulation, for example. Modelling of cracks and multiphase problems are also object of future research. Furthermore, non-tensor product discretizations are becoming popular among the IGA community in order to allow adaptive refinement of the mesh. This is the case, for example, of T-splines and hierarchical B-splines. Nowadays, some studies have been devoted to the implementation of multigrid methods for this type of discretizations but there is still a lot of work to do. In addition, triangular meshes can be considered for isogeometric discretizations. For instance, triangular Bézier splines or Powell-Sabin B-splines could be used and multigrid methods could be designed for their solution. Moreover, LFA could be applied to design new efficient and robust multigrid methods for the problems and discretizations mentioned above.

Bibliography

Aguilar, G. et al. (Sept. 2008). "Numerical stabilization of Biot's consolidation model by a perturbation on the flow equation". In: *International Journal for Numerical Methods in Engineering* 75, pp. 1282 –1300. DOI: 10.1002/nme.2295.

Ahlberg, J.H., E.N. Nilson, and Walsh (1967). *The Theory of Splines and Their Applications*. Acad. Press.

Ainsworth, Mark, Gaelle Andriamaro, and Oleg Davydov (2011). "Bernstein–Bézier Finite Elements of Arbitrary Order and Optimal Assembly Procedures". In: *SIAM Journal on Scientific Computing* 33.6, pp. 3087–3109. DOI: 10.1137/11082539X. eprint: https://doi.org/10.1137/11082539X. URL: https://doi.org/10.1137/11082539X.

Akkerman, I. et al. (2011). "Isogeometric analysis of free-surface flow". In: *Journal of Computational Physics* 230.11. Special issue High Order Methods for CFD Problems, pp. 4137–4152. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2010.11.044.

Alpak, Faruk and Mary Wheeler (Sept. 2012). "A supercoarsening multigrid method for poroelasticity in 3D coupled flow and geomechanics modeling". In: *Computational Geosciences* 16. DOI: 10.1007/s10596-012-9297-z.

Antolin, P. et al. (2015). "Efficient matrix computation for tensor-product isogeometric analysis: The use of sum factorization". In: *Computer Methods in Applied Mechanics and Engineering* 285, pp. 817–828. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2014.12.013. URL: https://www.sciencedirect.com/science/article/pii/S0045782514004927.

Arf, Jeremias and Bernd Simeon (Feb. 2022). "A space-time isogeometric method for the partial differential-algebraic system of Biot's poroelasticity model". In: *Electronic Transactions on Numerical Analysis* 55, pp. 310–340. DOI: doi:10.1553/etna_vol55s310.

Argyris, J. H., I. Fried, and D. W. Scharpf (1968). "The TUBA Family of Plate Elements for the Matrix Displacement Method". In: *The Aeronautical Journal (1968)* 72.692, 701–709. DOI: 10.1017/S000192400008489X.

Auricchio, F. et al. (2012). "A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis". In: *Computer Methods in Applied Mechanics and Engineering* 249, pp. 15–27. DOI: https://doi.org/10.1016/j.cma.2012.04.014.

Babu, Gutti Jogesh, Angelo J. Canty, and Yogendra P. Chaubey (2002). "Application of Bernstein Polynomials for smooth estimation of a distribution and density function". In: *Journal of Statistical Planning and Inference* 105, pp. 377–392. DOI: https://doi.org/10.1016/S0378-3758(01)00265-8.

Bakhvalov, N.S. (1966). "On the convergence of a relaxation method with natural constraints on the elliptic operator". In: *USSR Computational Mathematics and Mathematical Physics* 6.5, pp. 101–135. ISSN: 0041-5553. DOI: https://doi.org/10.1016/0041-5553(66)90118-2. URL: https://www.sciencedirect.com/science/article/pii/0041555366901182.

Bazilevs, Yuri et al. (Nov. 2011). "Isogeometric Analysis: Approximation, stability and error estimates for h-refined meshes". In: *Mathematical Models and Methods in Applied Sciences* 16. DOI: 10.1142/S0218202506001455.

Bekele, Yared Worku et al. (2022). "Mixed Method for Isogeometric Analysis of Coupled Flow and Deformation in Poroelastic Media". In: *Applied Sciences* 12.6. ISSN: 2076-3417. DOI: 10.3390/app12062915. URL: https://www.mdpi.com/2076-3417/12/6/2915.

Benedusi, Pietro et al. (2021). "Fast Parallel Solver for the Space-Time IgA-DG Discretization of the Diffusion Equation". In: *J. Sci. Comput.* 89.1. ISSN: 0885-7474. DOI: 10.1007/s10915-021-01567-z. URL: https://doi.org/10.1007/s10915-021-01567-z.

Böhm, Wolfgang, Gerald Farin, and Jürgen Kahmann (1984). "A survey of curve and surface methods in CAGD". In: *Computer Aided Geometric Design* 1.1, pp. 1–60. ISSN: 0167-8396. DOI: https://doi.org/10.1016/0167-8396(84)90003-7. URL: https://www.sciencedirect.com/science/article/pii/0167839684900037.

Biot, M.A. (1941). "General theory of three-dimensional consolidation". In: *Journal of Applied Physics* 12, 155–164. DOI: https://doi.org/10.1063/1.1712886.

— (1955). "Theory of elasticity and consolidation for a porous anisotropic solid". In: *Journal of Applied Physics* 26, 182–185. DOI: https://doi.org/10.1063/1.1721956.

— (1956). "General solutions of the equations of elasticity and consolidation for a porous material". In: *Journal of Applied Mechanics* 23, 91–96. DOI: https://doi.org/10.1115/1.4011213.

Boehm, Wolfgang (1980). "Inserting new knots into B-spline curves". In: *Computer-Aided Design* 12.4, pp. 199–201. ISSN: 0010-4485. DOI: https://doi.org/10.1016/0010-4485(80)90154-2. URL: https://www.sciencedirect.com/science/article/pii/0010448580901542.

Bogner, F. K., R. L. Fox, and L. A. Schmit (1965). "The generation of interelement compatible stiffness and mass matrices by the use of interpolation formulae". In: *Proceedings of the Conference on Matrix Methods in Structural Mechanics*, 397–444.

Boor, Carl de (Jan. 1978). *A Practical Guide to Spline*. Vol. Volume 27. DOI: 10.2307/2006241.

Borregales, Manuel et al. (2019). "A partially parallel-in-time fixed-stress splitting method for Biot's consolidation model". In: *Computers Mathematics with Applications* 77.6. 7th International Conference on Advanced Computational Methods in Engineering (ACOMEN 2017), pp. 1466–1478. ISSN: 0898-1221. DOI: https://doi.org/10.1016/j.camwa.2018.09.005.

Both, Jakub Wiktor et al. (2017). "Robust fixed stress splitting for Biot's equations in heterogeneous media". In: *Applied Mathematics Letters* 68, pp. 101–108. ISSN: 0893-9659. DOI: https://doi.org/10.1016/j.aml.2016.12.019. URL: https://www.sciencedirect.com/science/article/pii/S0893965917300034.

Brandt, Achi (1973). "Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems". In: *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*. Ed. by Henri Cabannes and Roger Temam. Berlin, Heidelberg: Springer, pp. 82–89. ISBN: 978-3-540-06170-0. DOI: 10.1007/BFb0118663.

— (1977). "Multi-Level Adaptive Solutions to Boundary-Value Problems". In: *Mathematics of Computation* 31.138, pp. 333–390. ISSN: 00255718, 10886842. DOI: https://doi.org/10.2307/2006422. URL: http://www.jstor.org/stable/2006422.

— (Dec. 1981). *Multigrid solvers for non-elliptic and singular-perturbation steady-state problems*. Tech. rep. Weizmann Institute of Science, Rehovot, Israel.

— (1994). "Rigorous Quantitative Analysis of Multigrid, I. Constant Coefficients Two-Level Cycle with L2-Norm". In: *SIAM Journal on Numerical Analysis* 31.6, pp. 1695–1730. DOI: 10.1137/0731087.

Brandt, Achi and Oren E. Livne (2011). *Multigrid Techniques*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611970753. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611970753.

Brenner, S. and R. Scott (2008). *The Mathematical Theory of Finite Element Methods*. Springer, New York.

Bressan, Andrea and Giancarlo Sangalli (Sept. 2012). "Isogeometric discretizations of the Stokes problem: stability analysis by the macroelement technique". In: *IMA Journal of Numerical Analysis* 33.2, pp. 629–651. ISSN: 0272-4979. DOI: 10.1093/imanum/drr056. eprint: https://academic.oup.com/imajna/article-pdf/33/2/629/1845094/drr056.pdf. URL: https://doi.org/10.1093/imanum/drr056.

Bressan, Andrea and Stefan Takacs (2019). "Sum factorization techniques in Isogeometric Analysis". In: *Computer Methods in Applied Mechanics and Engineering* 352, pp. 437–460. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2019.04.031. URL: https://www.sciencedirect.com/science/article/pii/S0045782519302348.

Brezzi, Franco and Richard S. Falk (1991). "Stability of Higher-Order Hood–Taylor Methods". In: *SIAM Journal on Numerical Analysis* 28.3, pp. 581–590. DOI: 10.1137/0728032. eprint: https://doi.org/10.1137/0728032. URL: https://doi.org/10.1137/0728032.

Brezzi, Franco and Michel Fortin (1991). "Mixed and Hybrid Finite Element Methods". In: *Springer Series in Computational Mathematics*.

Briggs, W.L., V.E. Henson, and S.F. McCormick (1987). *A Multigrid Tutorial*. Miscellaneous Bks. Society for Industrial and Applied Mathematics. ISBN: 9780898712216. URL: https://books.google.es/books?id=UDjvAAAAMAAJ.

Buffa, A., C. de Falco, and G. Sangalli (2010). "IsoGeometric Analysis: Stable elements for the 2D Stokes equation". In: *International Journal for Numerical Methods in Fluids* 65.11-12, pp. 1407–1422. DOI: 10.1002/fld.2337. URL: https://doi.org/10.1002%2Ffld.2337.

Cai, Xiao-Chuan and Marcus Sarkis (1999). "A Restricted Additive Schwarz Preconditioner for General Sparse Linear Systems". In: *SIAM Journal on Scientific Computing* 21.2, pp. 792–797. DOI: 10.1137/S106482759732678X. eprint: https://doi.org/10.1137/S106482759732678X. URL: https://doi.org/10.1137/S106482759732678X.

Chan, Tony F and Tarek P Mathew (1994). "Domain decomposition algorithms". In: *Acta numerica* 3, pp. 61–143.

Charru, François (2011). *Hydrodynamic Instabilities*. Ed. by PatriciaTranslator de Forcrand-Millard. Cambridge Texts in Applied Mathematics. Cambridge University Press. DOI: 10.1017/CBO9780511975172.

Chemin, Alexandre, Thomas Elguedj, and Anthony Gravouil (2015). "Isogeometric local h-refinement strategy based on multigrids". In: *Finite Elements in Analysis and Design* 100, pp. 77–90. ISSN: 0168-874X. DOI: https://doi.org/10.1016/j.finel.2015.02.007. URL: https://www.sciencedirect.com/science/article/pii/S0168874X15000384.

Ciarlet, P. (2002). *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9780898719208.

Clough, Ray William (1960). "The Finite Element Method in Plane Stress Analysis". In: Proceedings of American Society of Civil Engineers, pp. 345–378.

Cohen, Elaine, Tom Lyche, and Richard Riesenfeld (1980). "Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics". In: *Computer Graphics and Image Processing* 14.2, pp. 87–111. ISSN: 0146-664X. DOI: https://doi.org/10.1016/0146-664X(80)90040-4. URL: https://www.sciencedirect.com/science/article/pii/0146664X80900404.

Cohen, Elaine, Tom Lyche, and Larry L Schumaker (1986). "Degree raising for splines". In: *Journal of Approximation Theory* 46.2, pp. 170–181. ISSN: 0021-9045. DOI: https://doi.org/10.1016/0021-9045(86)90059-6. URL: https://www.sciencedirect.com/science/article/pii/0021904586900596.

Cottrell, J. Austin, Thomas J.R. Hughes, and Yuri Bazilevs (2005). "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement". In: *Computer Methods in Applied Mechanics and Engineering* 194.39, pp. 4135–4195. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2004.10.008. URL: https://www.sciencedirect.com/science/article/pii/S0045782504005171.

— (2009). *Isogeometric Analysis: Toward Integration of CAD and FEA*. 1st. Wiley Publishing. ISBN: 0470748737.

Cottrell, J. Austin, Thomas J.R. Hughes, and A. Reali (2007). "Studies of refinement and continuity in isogeometric structural analysis". In: *Computer methods in applied mechanics and engineering* 196.41-44, pp. 4160–4183. DOI: https://doi.org/10.1016/j.cma.2007.04.007.

Cottrell, J. Austin et al. (2006). "Isogeometric analysis of structural vibrations". In: *Computer methods in applied mechanics and engineering* 195.41-43, pp. 5257–5296. DOI: https://doi.org/10.1016/j.cma.2005.09.027.

Cox, M. G. (Oct. 1972). "The Numerical Evaluation of B-Splines*". In: *IMA Journal of Applied Mathematics* 10.2, pp. 134–149. ISSN: 0272-4960. DOI: 10.1093/imamat/10.2.134. eprint: https://academic.oup.com/imamat/article-pdf/10/2/134/2008789/10-2-134.pdf. URL: https://doi.org/10.1093/imamat/10.2.134.

Davis, P. J. (1963). *Interpolation and approximation*. English. A Blaisdell Book in the Pure and Applied Sciences. New York-Toronto- London: Blaisdell Publishing Company, a division of Ginn and Company, XIV, 393 p. (1963).

de Boor, Carl (1972). "On calculating with B-splines". In: *Journal of Approximation Theory* 6.1, pp. 50–62. ISSN: 0021-9045. DOI: https://doi.org/10.1016/0021-9045(72)90080-9. URL: https://www.sciencedirect.com/science/article/pii/0021904572900809.

Dedè, Luca, Micheal J. Borden, and Thomas J.R. Hughes (2012). "Isogeometric analysis for topology optimization with a phase field model". In: *Archives of Computational Methods in Engineering* 19.3, pp. 427–465. DOI: 10.1007/s11831-012-9075-z.

Detournay, E. and A. H.D. Cheng (1993). "Fundamentals of poroelasticity". In: *Journal of Fluid Mechanics*, pp. 113–171. ISSN: 0022-1120. DOI: 10.1016/b978-0-08-040615-2.50011-3.

Donatelli, Marco et al. (Dec. 2016). "Symbol-based multigrid methods for Galerkin B-spline isogeometric analysis". In: *SIAM Journal on Numerical Analysis*.

— (2017). "Symbol-Based Multigrid Methods for Galerkin B-Spline Isogeometric Analysis". In: *SIAM Journal on Numerical Analysis* 55.1, pp. 31–62. DOI: 10.1137/140988590. eprint: https://doi.org/10.1137/140988590. URL: https://doi.org/10.1137/140988590.

Dryja, Maksymilian and Olof B Widlund (1989). "Some domain decomposition algorithms for elliptic problems". In: *Iterative methods for large linear systems*. Ed. by L. Hayes and D. Kincaid. Academic Press, pp. 273–291.

Eberly, David H. (2016). *Representing a Circle or a Sphere with NURBS*. Geometric Tools.

Ehlers, Wolfgang and Joachim Bluhm (Jan. 2002). *Porous media. Theory, experiments and numerical applications*. DOI: 10.1007/978-3-662-04999-0.

Farin, Gerald (2001). *Curves and Surfaces for CAGD: A Practical Guide*. 5th. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN: 1558607374.

— (Dec. 2002). "A History of Curves and Surfaces in CAGD". In: *Computer Aided Geometric Design - CAGD*. DOI: 10.1016/B978-044451104-1/50002-2.

Farouki, Rida T. (2012). "The Bernstein polynomial basis: A centennial retrospective". In: *Computer Aided Geometric Design* 29.6, pp. 379–419. ISSN: 0167-8396. DOI: https://doi.org/10.1016/j.cagd.2012.03.001. URL: https://www.sciencedirect.com/science/article/pii/S0167839612000192.

Farrell, Patrick, Yunhui He, and Scott MacLachlan (June 2020). "A local Fourier analysis of additive Vanka relaxation for the Stokes equations". In: *Numerical Linear Algebra with Applications* 28. DOI: 10.1002/nla.2306.

Fedorenko, R.P. (1962). "A relaxation method for solving elliptic difference equations". In: *USSR Computational Mathematics and Mathematical Physics* 1.4, pp. 1092–1096. ISSN: 0041-5553. DOI: https://doi.org/10.1016/0041-5553(62)90031-9. URL: https://www.sciencedirect.com/science/article/pii/0041555362900319.

— (1964). "The speed of convergence of one iterative process". In: *USSR Computational Mathematics and Mathematical Physics* 4.3, pp. 227–235. ISSN: 0041-5553. DOI: https://doi.org/10.1016/0041-5553(64)90253-8. URL: https://www.sciencedirect.com/science/article/pii/0041555364902538.

Frommer, Andreas and Daniel B. Szyld (2001). "An Algebraic Convergence Theory for Restricted Additive Schwarz Methods Using Weighted Max Norms". In: *SIAM Journal on Numerical Analysis* 39.2, pp. 463–479. DOI: 10.1137/S0036142900370824. eprint: https://doi.org/10.1137/S003614290037082. URL: https://doi.org/10.1137/S0036142900370824.

Gahalaut, K.P.S., Johannes Kraus, and Satyendra Tomar (Jan. 2013). "Multigrid methods for isogeometric discretization". In: *Computer Methods in Applied Mechanics and Engineering* 253, 413–425. DOI: 10.1016/j.cma.2012.08.015.

Galdi, Giovanni P. (1994). *An introduction to the mathematical theory of the Navier-Stokes equations*. Springer. ISBN: 978-1-4757-3866-7. DOI: https://doi.org/10.1007/978-1-4757-3866-7.

Gander, Martin J. (2008). "Schwarz Methods over the Course of Time". In: *Electronic Transactions on Numerical Analysis* 31, pp. 228–255.

— (2015). "50 Years of Time Parallel Time Integration". In: *Multiple Shooting and Time Domain Decomposition Methods*. Ed. by Thomas Carraro et al. Cham: Springer International Publishing, pp. 69–113. ISBN: 978-3-319-23321-5.

Gardan, Y. and P. de Faget de Casteljau (1985). *Mathematics and CAD.: Shape mathematics and CAD*. Mathematics and CAD. Kogan Page. ISBN: 9781850910213. URL: https://books.google.es/books?id=frI-AQAAIAAJ.

Gaspar, Francisco J. and Carmen Rodrigo (2017). "On the fixed-stress split scheme as smoother in multigrid methods for coupling flow and geomechanics". In: *Computer Methods in Applied Mechanics and Engineering* 326, pp. 526–540. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2017.08.025. URL: https://www.sciencedirect.com/science/article/pii/S0045782517306047.

Girault, Vivette and Pierre-Arnaud Raviart (2011). *Finite Element Methods for Navier-Stokes Equations: Theory and Algorithms*. 1st. Springer Publishing Company, Incorporated. ISBN: 3642648886, 9783642648885.

Goldman, Ronald (June 1990). "Blossoming and knot insertion algorithms for B-spline curves". In: *Computer Aided Geometric Design* 7, pp. 69–81. DOI: 10.1016/0167-8396(90)90022-J.

Goldman, Ronald N. and Tom Lyche (1992). *Knot Insertion and Deletion Algorithms for B-Spline Curves and Surfaces*. Ed. by Ronald N. Goldman and Tom Lyche. Philadelphia, PA: Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611971583. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611971583.

Hackbusch, Wolfgang (1977). "On the convergence of a multigrid iteration applied to finite element equations". In: *University Koln Report*, pp. 77–78.

— (1980). "Convergence of Multi-Grid Iterations Applied to Difference Equations". In: *Mathematics of Computation* 34.150, pp. 425–440. ISSN: 00255718, 10886842. URL: http://www.jstor.org/stable/2006094.

— (Jan. 1981). "On the convergence of multi-grid iterations". In: *Beiträge zur Numerischen Mathematik* 9, pp. 213–239. DOI: 10.1007/978-3-662-02427-0_7.

Heinemann, B. (1977). "Prenter, P. M., Splines and Variational Methods, A Wiley-Interscience Publication, New York-London-Sydney-Toronto. John Wiley & Sons. 1975. 323 S., £ 10.75 ." In: *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik* 57.7, pp. 421–421. DOI: https://doi.org/10.1002/zamm.19770570718. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/zamm.19770570718. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/zamm.19770570718.

Higham, N. J. (Oct. 1997). "Iterative refinement for linear systems and LAPACK". In: *IMA Journal of Numerical Analysis* 17.4, pp. 495–509. ISSN: 0272-4979. DOI: 10.1093/imanum/17.4.495. eprint: https://academic.oup.com/imajna/article-pdf/17/4/495/2013423/17-4-495.pdf. URL: https://doi.org/10.1093/imanum/17.4.495.

Hofreither, Clemens (2018). "A black-box low-rank approximation algorithm for fast matrix assembly in Isogeometric Analysis". In: *Computer Methods in Applied Mechanics and Engineering* 333, pp. 311–330. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2018.01.014. URL: https://www.sciencedirect.com/science/article/pii/S0045782518300173.

Hofreither, Clemens and Stefan Takacs (Jan. 2017). "Robust Multigrid for Isogeometric Analysis Based on Stable Splittings of Spline Spaces". In: *SIAM Journal on Numerical Analysis* 55, pp. 2004–2024. DOI: 10.1137/16M1085425.

Hofreither, Clemens, Stefan Takacs, and Walter Zulehner (2017). "A robust multigrid method for Isogeometric Analysis in two dimensions using boundary correction". In: *Computer Methods in Applied Mechanics and Engineering* 316. Special Issue on Isogeometric Analysis: Progress and Challenges, pp. 22–42. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2016.04.003. URL: https://www.sciencedirect.com/science/article/pii/S0045782516301414.

Hofreither, Clemens and Walter Zulehner (June 2014). "Mass Smoothers in Geometric Multigrid for Isogeometric Analysis". In: vol. 9213. DOI: 10.1007/978-3-319-22804-4_20.

— (Feb. 2015). "Spectral Analysis of Geometric Multigrid Methods for Isogeometric Analysis". In: pp. 123–129. ISBN: 978-3-319-15584-5. DOI: 10.1007/978-3-319-15585-2_14.

— (2016). "On full multigrid schemes for isogeometric analysis". In: *Domain Decomposition Methods in Science and Engineering XXII*. Springer, pp. 267–274.

Hughes, Thomas J.R., Alessandro Reali, and Giancarlo Sangalli (2010). "Efficient quadrature for NURBS-based isogeometric analysis". In: *Computer methods in applied mechanics and engineering* 199.5-8, pp. 301–313. DOI: https://doi.org/10.1016/j.cma.2008.12.004.

Irzal, Faisal et al. (2013). "Isogeometric finite element analysis of poroelasticity". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 37.12, pp. 1891–1907. DOI: https://doi.org/10.1002/nag.2195. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nag.2195.

Kanschat, Guido and Youli Mao (2016). "Multiplicative Overlapping Schwarz Smoothers for H div-Conforming Discontinuous Galerkin Methods for the Stokes Problem". In: *Domain Decomposition Methods in Science and Engineering XXII*. Springer, pp. 285–292.

Kim, Jihoon, Hamdi Tchelepi, and R. Juanes (June 2011a). "Stability and convergence of sequential methods for coupled flow and geomechanics: Drained and undrained splits". In: *Computer Methods in Applied Mechanics and Engineering* 200, pp. 2094–2116. DOI: 10.1016/j.cma.2011.02.011.

— (Mar. 2011b). "Stability and convergence of sequential methods for coupled flow and geomechanics: Fixed-stress and fixed-strain splits". In: *Computer Methods in Applied Mechanics and Engineering* 200, pp. 1591–1606. DOI: 10.1016/j.cma.2010.12.022.

Kinnewig, Sebastian, Julian Roth, and Thomas Wick (2021). "Geometric multigrid with multiplicative Schwarz smoothers for eddy-current and Maxwell's equations in deal.II". In: *Examples and Counterexamples* 1, p. 100027. ISSN: 2666-657X. DOI: https://doi.org/10.1016/j.exco.2021.100027. URL: https://www.sciencedirect.com/science/article/pii/S2666657X2100015X.

Kumar, Prashant et al. (2019). "On Local Fourier Analysis of Multigrid Methods for PDEs with Jumping and Random Coefficients". In: *SIAM Journal on Scientific Computing* 41.3, A1385–A1413. DOI: 10.1137/18M1173769. eprint: https://doi.org/10.1137/18M1173769. URL: https://doi.org/10.1137/18M1173769.

Langer, Ulrich, Stephen E. Moore, and Martin Neumüller (2016). "Space–time isogeometric analysis of parabolic evolution problems". In: *Computer Methods in Applied Mechanics and Engineering* 306, pp. 342–363. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2016.03.042. URL: https://www.sciencedirect.com/science/article/pii/S0045782516301293.

Li, Kang and Xiaoping Qian (2011). "Isogeometric analysis and shape optimization via boundary integral". In: *Computer-Aided Design* 43.11. Solid and Physical Modeling 2011, pp. 1427–1437. ISSN: 0010-4485. DOI: https://doi.org/10.1016/j.cad.2011.08.031.

Lions, Pierre-Louis (1988). "On the Schwarz alternating method. I". In: *First international symposium on domain decomposition methods for partial differential equations*. Ed. by G. A. Meurant R. Glowinski G. H. Golub and J. Periaux. SIAM, pp. 1–42.

Loli, Gabriele et al. (2020). "An efficient solver for space–time isogeometric Galerkin methods for parabolic problems". In: *Computers Mathematics with Applications* 80.11. High-Order Finite Element and Isogeometric Methods 2019, pp. 2586–2603. ISSN: 0898-1221. DOI: https://doi.org/10.1016/j.camwa.2020.09.014. URL: https://www.sciencedirect.com/science/article/pii/S0898122120303709.

Luo, P. et al. (Oct. 2015). "Multigrid method for nonlinear poroelasticity equations". In: *Computing and Visualization in Science* 17. DOI: 10.1007/s00791-016-0260-8.

Lyche, Tom and Knut Mørken (June 1986). "Making the Oslo Algorithm More Efficient". In: *Siam Journal on Numerical Analysis* 23, pp. 663–675. DOI: 10.1137/0723042.

Lyche, Tom and Larry L. Schumaker (1973). "On the Convergence of Cubic Interpolating Splines". In: *Spline Functions and Approximation Theory: Proceedings of the Symposium held at the University of Alberta, Edmonton May 29 to June 1, 1972*. Ed. by A. Meir and A. Sharma. Basel: Birkhäuser Basel, pp. 169–189. ISBN: 978-3-0348-5979-0. DOI: 10.1007/978-3-0348-5979-0_9. URL: https://doi.org/10.1007/978-3-0348-5979-0_9.

MacLachlan, Scott and Cornelis Oosterlee (Aug. 2011). "Local Fourier analysis for multigrid with overlapping smoothers applied to systems of PDEs". In: *Numerical Linear Algebra with Applications* 18, pp. 751 –774. DOI: 10.1002/nla.762.

Manni, Carla and Hendrik Speleers (Oct. 2016). "Standard and Non-standard CAGD Tools for Isogeometric Analysis: A Tutorial". In: vol. 2161. ISBN: 978-3-319-42308-1. DOI: 10.1007/978-3-319-42309-8_1.

Mantzaflaris, Angelos et al. (2015). "Matrix Generation in Isogeometric Analysis by Low Rank Tensor Approximation". In: *Curves and Surfaces*. Ed. by Jean-Daniel Boissonnat et al. Cham: Springer International Publishing, pp. 321–340. ISBN: 978-3-319-22804-4.

Melenk, J.M., K. Gerdes, and C. Schwab (2001). "Fully discrete hp-finite elements: fast quadrature". In: *Computer Methods in Applied Mechanics and Engineering* 190.32, pp. 4339–4364. ISSN: 0045-7825. DOI: https://doi.org/10.1016/S0045-7825(00)00322-4. URL: https://www.sciencedirect.com/science/article/pii/S0045782500003224.

Mikelić, Andro and Mary Wheeler (June 2013). "Convergence of iterative coupling for coupled flow and geomechanics". In: *Computational Geosciences* 17, pp. 455–461. DOI: 10.1007/s10596-012-9318-y.

Moore, Stephen Edward (2018). "Discontinuous Galerkin Isogeometric Analysis for the biharmonic equation". In: *Computers Mathematics with Applications* 76.4, pp. 673–685. ISSN: 0898-1221. DOI: https://doi.org/10.1016/j.camwa.2018.05.001. URL: https://www.sciencedirect.com/science/article/pii/S089812211830261X.

Nedelec, J. (1986). "A new family of mixed
nite elements in R3". In: *Numerische Mathematik* 50.1, pp. 57–81.

Nielsen, Peter Nortoft et al. (2011). "Discretizations in isogeometric analysis of Navier–Stokes flow". In: *Computer Methods in Applied Mechanics and Engineering* 200.45, pp. 3242–3253. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2011.06.007. URL: https://www.sciencedirect.com/science/article/pii/S0045782511002258.

Orszag, Steven A (1980). "Spectral methods for problems in complex geometries". In: *Journal of Computational Physics* 37.1, pp. 70–92. ISSN: 0021-9991. DOI: https://doi.org/10.1016/0021-9991(80)90005-4. URL: https://www.sciencedirect.com/science/article/pii/0021999180900054.

Oruç, Halil and Necibe Tuncer (July 2001). *On the q-Bernstein Polynomials*. Tech. rep. Defense technical information center, p. 7.

Pé de la Riva, Álvaro, Carmen Rodrigo, and Francisco Gaspar (Jan. 2019). "A Robust Multigrid Solver for Isogeometric Analysis Based on Multiplicative Schwarz Smoothers". In: *SIAM Journal on Scientific Computing* 41, S321–S345. DOI: 10.1137/18M1194407.

— (2020). "On the robust solution of an isogeometric discretization of bilaplacian equation by using multigrid methods". In: *Computers Mathematics with Applications* 80.2. Numerical Methods for Scientific Computations and Advanced Applications II, pp. 386–394. ISSN: 0898-1221. DOI: https://doi.org/10.1016/j.

camwa.2019.08.011. URL: https://www.sciencedirect.com/science/article/pii/S0898122119304080.

— (Oct. 2021). "A two-level method for isogeometric discretizations based on multiplicative Schwarz iterations". In: *Computers Mathematics with Applications* 100, pp. 41–50. DOI: 10.1016/j.camwa.2021.08.020.

Pé de la Riva, Álvaro et al. (2022). *Oscillation-free numerical schemes for the Biot's model with automatic stabilization*. Applied Mathematics Letters. Submitted.

Patrikalakis, Nicholas M. and Takashi Maekawa (2009). *Shape Interrogation for Computer Aided Design and Manufacturing*. 1st. Springer Publishing Company, Incorporated. ISBN: 364204073X.

Patterson, Richard R. (Oct. 1985). "Projective Transformations of the Parameter of a Bernstein-Bézier Curve". In: *ACM Trans. Graph.* 4.4, 276–290. ISSN: 0730-0301. DOI: 10.1145/6116.6119. URL: https://doi.org/10.1145/6116.6119.

Piegl, Les and Wayne Tiller (1994). "Software-engineering approach to degree elevation of B-spline curves". In: *Computer-Aided Design* 26.1, pp. 17–28. ISSN: 0010-4485. DOI: https://doi.org/10.1016/0010-4485(94)90004-3. URL: https://www.sciencedirect.com/science/article/pii/0010448594900043.

— (1996). *The NURBS Book*. second. New York, NY, USA: Springer-Verlag.

Prautzsch, Hartmut (1984). "Degree elevation of B-spline curves". In: *Computer Aided Geometric Design* 1.2, pp. 193–198. ISSN: 0167-8396. DOI: https://doi.org/10.1016/0167-8396(84)90031-1. URL: https://www.sciencedirect.com/science/article/pii/0167839684900311.

Prautzsch, Hartmut, Wolfgang Boehm, and Marco Paluszny (Jan. 2002). *Bézier and B-Spline Techniques*. ISBN: 978-3-642-07842-2. DOI: 10.1007/978-3-662-04919-8.

Raviart, P. and J. Thomas (1977). "A mixed finite element method for 2-nd order elliptic problems". In: *Mathematical Aspects of Finite Element Methods*. Ed. by Ilio Galligani and Enrico Magenes. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 292–315. ISBN: 978-3-540-37158-8. DOI: https://doi.org/10.1007/BFb0064470.

Reali, Allessandro (2006). "An isogeometric analysis approach for the study of structural vibrations". In: *Journal of Earthquake Engineering* 10.spec01, pp. 1–30. DOI: 10.1080/13632460609350626.

Risler, J.J. (1992). *Mathematical Methods for CAD*. Recherches en mathématiques appliquées. Cambridge University Press. ISBN: 9780521436915. URL: https://books.google.es/books?id=txuYsi99QocC.

Rodrigo, C. et al. (2016). "Stability and monotonicity for some discretizations of the Biot's consolidation model". In: *Computer Methods in Applied Mechanics and Engineering* 298, pp. 183–204. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2015.09.019. URL: https://www.sciencedirect.com/science/article/pii/S0045782515003138.

Rodrigo, Carmen (2016). "Poroelasticity problem: numerical difficulties and efficient multigrid solution". In: *SeMA Journal* 73.1, pp. 31–57.

Saberi, S., G. Meschke, and A. Vogel (2022). "A Restricted Additive Vanka Smoother for Geometric Multigrid". In: *Journal of Computational Physics* 459.C. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2022.111123. URL: https://doi.org/10.1016/j.jcp.2022.111123.

Schöberl, Joachim and Walter Zulehner (2003). "On Schwarz-type Smoothers for Saddle Point Problems". In: *Numerische Mathematik* 95.2, pp. 377–399. DOI: 10.1007/s00211-002-0448-3. URL: https://doi.org/10.1007/s00211-002-0448-3.

Schoeberl, Joachim and Walter Zulehner (Aug. 2003). "On Schwarz-type Smoothers for Saddle Point Problems". In: *Numerische Mathematik* 95.2, pp. 377–399. DOI: 10.1007/s00211-002-0448-3.

Schoenberg, I. J. (1946a). "Contributions to the problem of approximation of equidistant data by analytic functions: Part A—On the problem of smoothing or graduation. A first Class of analytic approximation formulae". In: *Quarterly of Applied Mathematics* 4.1, pp. 45–99. ISSN: 0033569X, 15524485. URL: http://www.jstor.org/stable/43633538.

— (1946b). "Contributions to the problem of approximation of equidistant data by analytic functions: Part B—On the problem of osculatory interpolation. A second class of analytic approximation formulae". In: *Quarterly of Applied Mathematics* 4.2, pp. 112–141. ISSN: 0033569X, 15524485. URL: http://www.jstor.org/stable/43633544.

— (1967). "On the Convergence of Cubic Interpolating Splines". In: *Inequalities (Symposium at Wright-Patterson Air Force Base)*. New York: Academic Press, pp. 255–291.

Schoenberg, I.J. (1973). *Cardinal Spline Interpolation*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics. ISBN: 9780898710090. URL: https://books.google.es/books?id=03ipetjS_64C.

Schwarz, H. A. (1870). "Über einen Grenzübergang durch alternierendes Verfahren". In: *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich* 15, 272–286.

Seo, Yu-Deok, Hyun-Jung Kim, and Sung-Kie Youn (2010). "Shape optimization and its extension to topological design based on isogeometric analysis". In: *International Journal of Solids and Structures* 47.11-12, pp. 1618–1640.

Shojaee, Saeed, E. Izadpenah, and Azadeh Haeri (2012). "Imposition of essential boundary conditions in isogeometric analysis using the lagrange multiplier method". In: *Iran University of Science & Technology* 2, pp. 247–271.

Smith, Barry F (1997). "Domain decomposition methods for partial differential equations". In: *Parallel Numerical Algorithms*. Springer, pp. 225–243.

Smith, Barry F., Petter E. Bjørstad, and William Gropp (1996). "Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations". In.

Strikwerda, John C. (2004). *Finite Difference Schemes and Partial Differential Equations*. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics.

Tagliabue, Anna, Luca Dedè, and Alfio Quarteroni (2014). "Isogeometric Analysis and error estimates for high order partial differential equations in fluid dynamics". In: *Computers Fluids* 102, pp. 277–303. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2014.07.002. URL: https://www.sciencedirect.com/science/article/pii/S0045793014002813.

Taylor, C. and P. Hood (1973). "A numerical solution of the Navier-Stokes equations using the finite element technique". In: *Computers Fluids* 1.1, pp. 73–100. ISSN: 0045-7930. DOI: https://doi.org/10.1016/0045-7930(73)90027-3. URL: https://www.sciencedirect.com/science/article/pii/0045793073900273.

Terzaghi, K.V. (1923). "Die berechnung der durchlassigkeitsziffer des tones aus dem verlauf der hydrodynamischen spannungserscheinungen". In: *Sitzungsberichte Kais. Akad. Wiss. Wien* 132, pp. 125 –138.

Thorpe, S. A. (1982). "Hydrodynamic Stability by P. G. Drazin and W. H. Reid, Cambridge Monographs on Mechanics and Applied Mathematics, General Editors G. K. Batchelor and J. W. Miles, Cambridge University Press. Pp. 525. £35.00; Hardback". In: *Quarterly Journal of the Royal Meteorological Society* 108.455, pp. 263–263. DOI: https://doi.org/10.1002/qj.49710845516. eprint: https://rmets.

onlinelibrary.wiley.com/doi/pdf/10.1002/qj.49710845516. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.49710845516.

Timoshenko, S. and S. Woinowsky-Krieger (1959). *Theory of Plates and Shells*. in: Engineering Societies Monographs, McGraw-Hill.

Trottenberg, Ulrich, Cornelis W. Oosterlee, and Anton Schüller (2001). *Multigrid*. Vol. 33. Texts in Applied Mathematics. Bd. With contributions by A. Brandt, P. Oswald and K. Stüben. San Diego [u.a.]: Academic Press. ISBN: 0-12-701070-X.

Turner, M. J. et al. (1956). "Stiffness and Deflection Analysis of Complex Structures". In: *Journal of the Aeronautical Sciences* 23.9, pp. 805–823. DOI: 10.2514/8.3664. eprint: https://doi.org/10.2514/8.3664. URL: https://doi.org/10.2514/8.3664.

Vanka, S P (1986). "Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables". In: *Journal of Computational Physics* 65.1, 138–158. ISSN: 0021-9991. DOI: 10.1016/0021-9991(86)90008-2. URL: https://doi.org/10.1016/0021-9991(86)90008-2.

Verruijt, Arnold (Jan. 2018). *An Introduction to Soil Mechanics*. ISBN: 978-3-319-61184-6. DOI: 10.1007/978-3-319-61185-3.

Wang, Herbert (Dec. 2001). *Theory of Linear Poroelasticity with Applications to Geomechanics and Hydrogeology*. ISBN: 9781400885688. DOI: 10.1515/9781400885688.

Wesseling, P. (2004). *An Introduction to Multigrid Methods*. An Introduction to Multigrid Methods. R.T. Edwards. ISBN: 9781930217089. URL: https://books.google.es/books?id=RoRGAAAAYAAJ.

Yamaguchi, Fujio (1988). *Curves and Surfaces in Computer Aided Geometric Design*. Berlin, Heidelberg: Springer-Verlag. ISBN: 0387174494.

Zienkiewicz, O. C., C. T. Chang, and P. Bettess (1980). "Drained, undrained, consolidating and dynamic behaviour assumptions in soils". In: *Géotechnique* 30.4, pp. 385–395. DOI: 10.1680/geot.1980.30.4.385. eprint: https://doi.org/10.1680/geot.1980.30.4.385. URL: https://doi.org/10.1680/geot.1980.30.4.385.