Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations

FRONTIERS IN APPLIED MATHEMATICS

The SIAM series on Frontiers in Applied Mathematics publishes monographs dealing with creative work in a substantive field involving applied mathematics or scientific computation. All works focus on emerging or rapidly developing research areas that report on new techniques to solve mainstream problems in science or engineering.

The goal of the series is to promote, through short, inexpensive, expertly written monographs, cutting edge research poised to have a substantial impact on the solutions of problems that advance science and technology. The volumes encompass a broad spectrum of topics important to the applied mathematical areas of education, government, and industry.

BOOKS PUBLISHED IN FRONTIERS IN APPLIED MATHEMATICS

Rivière, Béatrice, *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations: Theory and Implementation*

Batzel, Jerry J.; Kappel, Franz; Schneditz, Daniel; and Tran, Hien T., *Cardiovascular and Respiratory Systems: Modeling, Analysis, and Control*

Li, Zhilin and Ito, Kazufumi, *The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*

Smith, Ralph C., *Smart Material Systems: Model Development*

Iannelli, M.; Martcheva, M.; and Milner, F. A., *Gender-Structured Population Modeling: Mathematical Methods, Numerics, and Simulations*

Pironneau, O. and Achdou, Y., *Computational Methods for Option Pricing*

Day, William H. E. and McMorris, F. R., *Axiomatic Consensus Theory in Group Choice and Biomathematics*

Banks, H. T. and Castillo-Chavez, Carlos, editors, *Bioterrorism: Mathematical Modeling Applications in Homeland Security*

Smith, Ralph C. and Demetriou, Michael, editors, *Research Directions in Distributed Parameter Systems*

Höllig, Klaus, *Finite Element Methods with B-Splines*

Stanley, Lisa G. and Stewart, Dawn L., *Design Sensitivity Analysis: Computational Issues of Sensitivity Equation Methods*

Lewis, F. L.; Campos, J.; and Selmic, R., *Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities*

Vogel, Curtis R., *Computational Methods for Inverse Problems*

Bao, Gang; Cowsar, Lawrence; and Masters, Wen, editors, *Mathematical Modeling in Optical Science*

Banks, H. T.; Buksas, M. W.; and Lin, T., *Electromagnetic Material Interrogation Using Conductive Interfaces and Acoustic Wavefronts*

Oostveen, Job, *Strongly Stabilizable Distributed Parameter Systems*

Griewank, Andreas, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*

Kelley, C. T., *Iterative Methods for Optimization*

Greenbaum, Anne, *Iterative Methods for Solving Linear Systems*

Kelley, C. T., *Iterative Methods for Linear and Nonlinear Equations*

Moré, Jorge J. and Wright, Stephen J., *Optimization Software Guide*

Rüde, Ulrich, *Mathematical and Computational Techniques for Multilevel Adaptive Methods*

Cook, L. Pamela, editor, *Transonic Aerodynamics: Problems in Asymptotic Theory*

Banks, H. T., editor, *Control and Estimation in Distributed Parameter Systems*

Van Loan, Charles, *Computational Frameworks for the Fast Fourier Transform*

Van Huffel, Sabine and Vandewalle, Joos, *The Total Least Squares Problem: Computational Aspects and Analysis*

Castillo, José E., editor, *Mathematical Aspects of Numerical Grid Generation*

McCormick, Stephen F., *Multilevel Adaptive Methods for Partial Differential Equations*

Grossman, Robert, editor, *Symbolic Computation: Applications to Scientific Computing*

Coleman, Thomas F. and Van Loan, Charles, *Handbook for Matrix Computations*

McCormick, Stephen F., editor, *Multigrid Methods*

Buckmaster, John D., editor, *The Mathematics of Combustion*

Ewing, Richard E., editor, *The Mathematics of Reservoir Simulation*

Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations

Theory and Implementation

Béatrice Rivière

Rice University
Houston, Texas

siam.
Society for Industrial and Applied Mathematics
Philadelphia

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data

Rivière, Béatrice.
Discontinuous Galerkin methods for solving elliptic and parabolic equations : theory and implementation / Béatrice Rivière.
p. cm. — (Frontiers in applied mathematics)
Includes bibliographical references and index.
ISBN 978-0-898716-56-6
1. Differential equations, Elliptic—Numerical solutions. 2. Differential equations, Parabolic—Numerical solutions. 3. Galerkin methods. I. Title.

QA377.R58 2008
518'.64—dc22 2008018508

To Kate, Paul, and Ingmar

Contents

List of Figures

List of Tables

List of Algorithms

Preface

This book is an introduction to a family of discontinuous Galerkin (DG) methods applied to some steady-state and time-dependent model problems. A special effort was made to have the material self-contained as much as possible. The book is well suited to numerical analysts interested in DG methods but also to applied mathematicians who study CFD or porous media flow. Practical implementation issues are discussed, which can be of interest to engineers. The material can be used in a graduate level course on the numerical solution of partial differential equations. Chapter 1 is introductory and can be used in a scientific computing class for senior undergraduate students. Prerequisites are calculus and linear algebra.

In this book, we mainly focus on the class of *primal* DG methods, namely variations of interior penalty methods. In the text, these methods are referred to as the symmetric interior penalty Galerkin (SIPG), incomplete interior penalty Galerkin (IIPG), and nonsymmetric interior penalty Galerkin (NIPG) methods. The book is divided into three parts: Part I focuses on the application of DG to second order elliptic problems in one dimension first and then in higher dimensions. In Part II, the time-dependent parabolic problems (without and with convection) are presented. Finally, Part III covers some applications of DG to solid mechanics (linear elasticity), to fluid dynamics (Stokes and Navier–Stokes), and to porous media flow (two-phase and miscible displacement).

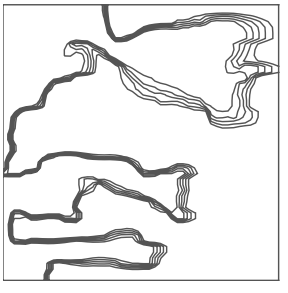
We try to discuss both theoretical and computational aspects of the DG methods. In particular, for the elliptic equations, a code written in MATLAB® for one-dimensional problems is provided in Appendix B.1. The text contains algorithms for the implementation of DG methods in two or three dimensions. Corresponding routines written in C are provided in Appendix B.2 as well.

One objective of this book is to teach the reader the basic tools for analyzing DG methods. Proofs of stability and convergence of the method are given with many details. Another objective is to teach the reader the coding issues of DG methods: data structure, construction of local matrices, and assembling of the global matrix. Several computational examples are provided. Finally, by presenting specific applications of DG to important engineering problems, we hope to convince the reader that the DG method is a competitive approach for solving his/her own scientific problem.

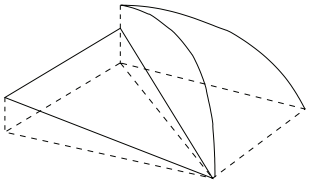
The first DG methods were introduced for hyperbolic problems, which we do not cover in this book. The treatment of DG methods for conservation laws can itself be the object of an entire book. Other important topics not discussed in this book include solvers and preconditioning.

This book stems from a collection of notes that I used in several graduate classes while I was teaching at the University of Pittsburgh. I would like to thank all the students for their feedback and comments. Some of the work presented in this book was funded by the National Science Foundation. I am grateful for their support.

B.M. Rivière



Part I
Elliptic Problems



Chapter 1
One-dimensional problem

In this chapter, we define the primal DG methods for solving a two-point boundary value problem in one dimension.

1.1 Model problem

Let us consider the following two-point boundary value problem on the unit interval:

$$\forall x \in (0,1), \quad -(K(x)p'(x))' = f(x), \tag{1.1}$$

$$p(0) = 1, \tag{1.2}$$

$$p(1) = 0, \tag{1.3}$$

where $K \in \mathcal{C}^1(0,1)$ and $f \in \mathcal{C}^0(0,1)$. We also assume that there are two constants K_0 and K_1 such that

$$\forall x \in (0,1), \quad 0 < K_0 \le K(x) \le K_1.$$

We say that p is a solution of (1.1)–(1.3) if $p \in \mathcal{C}^2(0,1)$ and p satisfies the equations (1.1)–(1.3) pointwise.

1.2 A class of DG methods

Let $0 = x_0 < x_1 < \cdots < x_N = 1$ be a partition $\mathcal{E}_h$ of $(0,1)$, denote $I_n = (x_n, x_{n+1})$, and define

$$h_n = x_{n+1} - x_n, \quad h_{n-1,n} = \max(h_{n-1}, h_n), \quad h = \max_{0 \le n \le N-1} h_n.$$

Denote by $\mathcal{D}_k(\mathcal{E}_h)$ the space of piecewise discontinuous polynomials of degree k:

$$\mathcal{D}_k(\mathcal{E}_h) = \{v : v|_{I_n} \in \mathbb{P}_k(I_n) \ \ \forall j = 0, \ldots, N-1\},$$

where $\mathbb{P}_k(I_n)$ is the space of polynomials of degree k on the interval I_n. Let us denote $v(x_n^+) = \lim_{\substack{\epsilon \to 0 \\ \epsilon > 0}} v(x_n + \epsilon)$ and $v(x_n^-) = \lim_{\substack{\epsilon \to 0 \\ \epsilon > 0}} v(x_n - \epsilon)$. Then we can define the jump and

average of v at the endpoints of I_n:

$$[v(x_n)] = v(x_n^-) - v(x_n^+), \quad \{v(x_n)\} = \frac{1}{2}(v(x_n^-) + v(x_n^+)) \quad \forall n = 1, \ldots, N-1.$$

By convention, we also extend the definition of jump and average at the endpoints of the unit interval:

$$[v(x_0)] = -v(x_0^+), \quad \{v(x_0)\} = v(x_0^+), \quad [v(x_N)] = v(x_N^-), \quad \{v(x_N)\} = v(x_N^-).$$

Next, we introduce jump terms of the solution and its derivative. Those terms are also referred to as penalty terms:

$$J_0(v, w) = \sum_{n=0}^{N} \frac{\sigma^0}{h_{n-1,n}}[v(x_n)][w(x_n)], \quad J_1(v, w) = \sum_{n=1}^{N-1} \frac{\sigma^1}{h_{n-1,n}}[v'(x_n)][w'(x_n)],$$

where σ^0 and σ^1 are two real nonnegative numbers. We note that J_0 penalizes the jump in the function v (or w), whereas J_1 penalizes the jump in the derivative of the function v (or w).

Let v be a function in $\mathcal{D}_k(\mathcal{E}_h)$. Let us multiply (1.1) by v and let us integrate by parts on each interval I_n:

$$\begin{aligned}&\int_{x_n}^{x_{n+1}} K(x)p'(x)v'(x)dx - K(x_{n+1})p'(x_{n+1})v(x_{n+1}^-) + K(x_n)p'(x_n)v(x_n^+)\\ &= \int_{x_n}^{x_{n+1}} f(x)v(x)dx, \quad n = 0, \ldots, N-1.\end{aligned}$$

By adding all N equations above, we obtain

$$\sum_{n=0}^{N-1}\int_{x_n}^{x_{n+1}} K(x)p'(x)v'(x)dx - \sum_{n=0}^{N}[K(x_n)p'(x_n)v(x_n)] = \int_0^1 f(x)v(x)dx.$$

It is easy to check that for $1 \le n \le N-1$

$$[K(x_n)p'(x_n)v(x_n)] = \{K(x_n)p'(x_n)\}[v(x_n)] + \{v(x_n)\}[K(x_n)p'(x_n)]. \tag{1.4}$$

By applying (1.4) and by noting that the exact solution p satisfies $[K(x_n)p'(x_n)] = 0$ for all $1 \le n \le N-1$, we obtain

$$\sum_{n=0}^{N-1}\int_{x_n}^{x_{n+1}} K(x)p'(x)v'(x)dx - \sum_{n=0}^{N}\{K(x_n)p'(x_n)\}[v(x_n)] = \int_0^1 f(x)v(x)dx.$$

We now note that the exact solution p is also continuous, i.e., $[p(x_n)] = 0$. Therefore, we see that if p is a solution of (1.1)–(1.3), then p satisfies

$$\begin{aligned}&\sum_{n=0}^{N-1}\int_{x_n}^{x_{n+1}} K(x)p'(x)v'(x)dx - \sum_{n=0}^{N}\{K(x_n)p'(x_n)\}[v(x_n)] + \epsilon\sum_{n=0}^{N}\{K(x_n)v'(x_n)\}[p(x_n)]\\ &= \int_0^1 f(x)v(x)dx - \epsilon K(x_0)v'(x_0)p(x_0) + \epsilon K(x_N)v'(x_N)p(x_N)\\ &= \int_0^1 f(x)v(x)dx - \epsilon K(x_0)v'(x_0).\end{aligned}$$

Here, ϵ can be any real number, as the third term in the equation above is intrinsically zero. However, we restrict ourselves to the case $\epsilon \in \{-1, 0, -1\}$.

Definition 1.1. *Given a real vector space V, the function $a : V \times V \to \mathbb{R}$ is called a bilinear form if a is linear with respect to each of its arguments. In other words, for all $\alpha \in \mathbb{R}$, $v, v_1, v_2, w, w_1, w_2 \in V$, we have*

$$\begin{aligned} a(v_1 + v_2, w) &= a(v_1, w) + a(v_2, w), \\ a(\alpha v, w) &= \alpha a(v, w), \\ a(v, w_1 + w_2) &= a(v, w_1) + a(v, w_2), \\ a(v, \alpha w) &= \alpha a(v, w). \end{aligned}$$

We now define the DG bilinear form $a_\epsilon : \mathcal{D}_k(\mathcal{E}_h) \times \mathcal{D}_k(\mathcal{E}_h) \to \mathbb{R}$:

$$\begin{aligned} a_\epsilon(w, v) = &\sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x) w'(x) v'(x) dx - \sum_{n=0}^{N} \{K(x_n) w'(x_n)\} [v(x_n)] \\ &+ \epsilon \sum_{n=0}^{N} \{K(x_n) v'(x_n)\} [w(x_n)] + J_0(w, v) + J_1(w, v). \end{aligned}$$

The DG bilinear form a_ϵ has the following properties:

- For $\epsilon = -1$, the form is symmetric, i.e.,

$$\forall v, w, \quad a_{-1}(v, w) = a_{-1}(w, v),$$

and we have

$$\begin{aligned} a_{-1}(v, v) = &\sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x) (v'(x))^2 dx - 2 \sum_{n=0}^{N} \{K(x_n) v'(x_n)\} [v(x_n)] \\ &+ J_0(v, v) + J_1(v, v). \end{aligned}$$

- For $\epsilon \in \{0, +1\}$, the form is nonsymmetric, and we have

$$a_{+1}(v, v) = \sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x) (v'(x))^2 dx + J_0(v, v) + J_1(v, v) \geq 0, \tag{1.5}$$

$$\begin{aligned} a_0(v, v) = &\sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x) (v'(x))^2 dx - \sum_{n=0}^{N} \{K(x_n) v'(x_n)\} [v(x_n)] \\ &+ J_0(v, v) + J_1(v, v). \end{aligned}$$

A class of DG methods for solving the boundary value problem (1.1)–(1.3) is as follows: Find $P^{\mathrm{DG}} \in \mathcal{D}_k(\mathcal{E}_h)$ such that

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(P^{\mathrm{DG}}, v) = L(v), \tag{1.6}$$

where $L : \mathcal{D}_k(\mathcal{E}_h) \to \mathbb{R}$ is the linear form

$$L(v) = \int_0^1 f(x)v(x)dx - \epsilon K(x_0)v'(x_0) + \frac{\sigma^0}{h_{0,1}}v(x_0).$$

Remark: Problem (1.6) is an example of a finite-dimensional variational formulation. This concept is discussed in detail in the next chapter.
Remark: Depending on the choices of the parameters ϵ, σ^0, σ^1, we obtain several variations of DG methods that have appeared in the literature at different times.

- If $\epsilon = -1$, $\sigma^1 = 0$, and σ^0 is bounded below by a large enough constant, the resulting method is called the symmetric interior penalty Galerkin (SIPG) method, introduced in the late 1970s by Wheeler [109] and Arnold [1].

- If $\epsilon = -1$ and $\sigma^0 = \sigma^1 = 0$, the resulting method is called the global element method, introduced in 1979 by Delves and Hall [43]. However, the matrix associated with the bilinear form is indefinite, as the real parts of the eigenvalues are not all positive and thus the method is not stable.

- If $\epsilon = +1$, $\sigma^1 = 0$, and $\sigma^0 = 1$, the resulting method is called the nonsymmetric interior penalty Galerkin (NIPG) method, introduced in 1999 by Rivière, Wheeler, and Girault [95].

- If $\epsilon = +1$ and $\sigma^0 = \sigma^1 = 0$, the resulting method was introduced by Oden, Babuška, and Baumann in 1998 [84]. Throughout these notes, we will refer to this method as the NIPG 0 method, since it corresponds to the particular case of NIPG with $\sigma^0 = 0$.

- If $\epsilon = 0$, we obtain the incomplete interior penalty Galerkin (IIPG) method introduced by Dawson, Sun, and Wheeler [42] in 2004.

Remark: What if $\epsilon = 0$ and $\sigma^0 = \sigma^1 = 0$? Then the method is not convergent and not stable. One cannot even prove uniqueness and existence of the discrete solution.
Remark: It could be useful in practice to allow the penalty parameters to vary with each node. For instance a large value σ_n^0 yields a numerical solution with a small jump at the node x_n.

1.3 Existence and uniqueness of the DG solution

Since the problem is finite-dimensional, existence of a solution is equivalent to uniqueness. Let us assume that P^1 and P^2 are two solutions and let us define $\theta = P^1 - P^2$. Since both P^1 and P^2 satisfy (1.6), we have

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(\theta, v) = 0.$$

Choosing in particular $v = \theta$ gives

$$a_\epsilon(\theta, \theta) = 0.$$

In the case of NIPG with $\sigma^0 > 0$, (1.5) yields

$$\forall n \geq 0, \quad \int_{x_n}^{x_{n+1}} K(x)(\theta'(x))^2 dx = 0, \quad \frac{\sigma^0}{h_{n-1,n}}[\theta(x_n)]^2 = 0.$$

Since K is strictly positive, the first equation implies that the function θ is equal to a constant k_n on each interval I_n. The second equation implies that all k_n's are equal to the same zero constant.

What should we do in the case of SIPG, IIPG, and NIPG 0? The proof is not as simple and is given in Chapter 2. In particular, some conditions on the penalty parameters need to be imposed in order to obtain uniqueness (hence existence) of the solution.

1.4 Linear system

In this section, we derive the linear system obtained from the DG scheme in the simpler case where K is the unit constant and $\sigma^1 = 0$. We also consider the case where discontinuous piecewise quadratic polynomials are used, namely $k = 2$. Let us choose for local basis functions of $\mathbb{P}_2(I_n)$ the monomial basis functions, translated from the interval $(-1, 1)$:

$$\mathbb{P}_2(I_n) = \text{span}\{\phi_0^n, \phi_1^n, \phi_2^n\}$$

with

$$\phi_0^n(x) = 1, \quad \phi_1^n(x) = 2\frac{x - x_{n+1/2}}{x_{n+1} - x_n}, \quad \phi_2^n(x) = 4\frac{(x - x_{n+1/2})^2}{(x_{n+1} - x_n)^2},$$

and $x_{n+1/2} = \frac{1}{2}(x_n + x_{n+1})$ is the midpoint of the interval I_n. To further simplify the computation, let us assume that there is a positive integer N such that

$$x_n = x_0 + nh, \quad h = \frac{1}{N}.$$

Thus, the local basis functions and their derivatives are simply

$$\phi_0^n(x) = 1, \quad \phi_1^n(x) = \frac{2}{h}(x - (n + 1/2)h), \quad \phi_2^n(x) = \frac{4}{h^2}(x - (n + 1/2)h)^2, \quad (1.7)$$

$$\phi_0^{n\prime}(x) = 0, \quad \phi_1^{n\prime}(x) = \frac{2}{h}, \quad \phi_2^{n\prime}(x) = \frac{8}{h^2}(x - (n + 1/2)h). \quad (1.8)$$

The global basis functions $\{\Phi_i^n\}$ for the space $\mathcal{D}_2(\mathcal{E}_h)$ are obtained from the local basis functions by extending them by zero:

$$\Phi_i^n(x) = \begin{cases} \phi_i^n(x), & x \in I_n, \\ 0, & x \notin I_n. \end{cases}$$

We can then expand the DG solution as

$$\forall x \in (0, 1), \quad P^{\text{DG}}(x) = \sum_{m=0}^{N-1} \sum_{j=0}^{2} \alpha_j^m \Phi_m^j(x), \quad (1.9)$$

where the coefficients α_j^m are unknown real numbers to be solved for. Plugging this form of P^{DG} into the scheme (1.6), we have

$$\forall 0 \le n \le N-1, \quad \forall 0 \le i \le 2, \quad \sum_{m=0}^{N-1} \sum_{j=0}^{2} \alpha_j^m a_\epsilon(\Phi_m^j, \Phi_n^i) = L(\Phi_n^i).$$

We obtain a linear system of the form $\boldsymbol{A}\boldsymbol{\alpha} = \boldsymbol{b}$, where $\boldsymbol{\alpha}$ is the vector with components α_j^m, $\boldsymbol{A}$ is the matrix with entries $a_\epsilon(\Phi_m^j, \Phi_n^i)$, and $\boldsymbol{b}$ is the vector with components $L(\Phi_n^i)$.

1.4.1 Computing the matrix A

Because of the local support of the global basis functions, the entries of the global matrix $\boldsymbol{A}$ can be obtained by first computing and assembling local matrices.

In what follows, we first describe how to compute the local matrices. We will regroup the terms defining a_ϵ into three groups: the terms involving integrals over I_n, the terms involving the interior nodes x_n, and those involving the boundary nodes x_0, x_N.

First, we consider the term corresponding to the integrals over the intervals I_n. On each element I_n, the DG solution P^{DG} is a quadratic polynomial, and we can write

$$\forall x \in I_n, \quad P^{\mathrm{DG}}(x) = \alpha_0^n \phi_0^n(x) + \alpha_1^n \phi_1^n(x) + \alpha_2^n \phi_2^n(x). \tag{1.10}$$

Thus, using the expansion above and choosing $v = \phi_i^n$ for $i = 0, 1, 2$, we obtain

$$\forall i = 0, 1, 2, \quad \int_{I_n} (P^{\mathrm{DG}})'(x)(\phi_i^n)'(x)dx = \sum_{j=0}^{2} \alpha_j^n \int_{I_n} (\phi_j^n)'(x)(\phi_i^n)'(x)dx.$$

This linear system can be rewritten as $\boldsymbol{A}_n \boldsymbol{\alpha}^n$, where

$$\boldsymbol{\alpha}^n = \begin{pmatrix} \alpha_0^n \\ \alpha_1^n \\ \alpha_2^n \end{pmatrix}, \quad (\boldsymbol{A}_n)_{ij} = \int_{I_n} (\phi_i^n)'(x)(\phi_j^n)'(x)dx.$$

One can easily compute $\boldsymbol{A}_n$:

$$\boldsymbol{A}_n = \frac{1}{h} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & \frac{16}{3} \end{pmatrix}.$$

Second, we consider the terms involving the interior nodes x_n. By expanding the average and jump terms, we can write

$$\begin{aligned} &-\{(P^{\mathrm{DG}})'(x_n)\}[v(x_n)] + \epsilon\{v'(x_n)\}[P^{\mathrm{DG}}(x_n)] + \frac{\sigma^0}{h}[P^{\mathrm{DG}}(x_n)][v(x_n)] \\ &= b_n + c_n + d_n + e_n, \end{aligned}$$

where the terms are defined below:

$$b_n = \frac{1}{2}(P^{\mathrm{DG}})'(x_n^+)v(x_n^+) - \frac{\epsilon}{2}P^{\mathrm{DG}}(x_n^+)v'(x_n^+) + \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_n^+)v(x_n^+),$$
$$c_n = -\frac{1}{2}(P^{\mathrm{DG}})'(x_n^-)v(x_n^-) + \frac{\epsilon}{2}P^{\mathrm{DG}}(x_n^-)v'(x_n^-) + \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_n^-)v(x_n^-),$$
$$d_n = -\frac{1}{2}(P^{\mathrm{DG}})'(x_n^+)v(x_n^-) - \frac{\epsilon}{2}P^{\mathrm{DG}}(x_n^+)v'(x_n^-) - \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_n^+)v(x_n^-),$$
$$e_n = \frac{1}{2}(P^{\mathrm{DG}})'(x_n^-)v(x_n^+) + \frac{\epsilon}{2}P^{\mathrm{DG}}(x_n^-)v'(x_n^+) - \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_n^-)v(x_n^+).$$

Again, with the expansion (1.10) and with the choice $v = \phi_i^n$, the four terms defined above will yield the local matrices $\boldsymbol{B}_n$, $\boldsymbol{C}_n$, $\boldsymbol{D}_n$, and $\boldsymbol{E}_n$, respectively. For instance, the entries of $\boldsymbol{B}_n$ and $\boldsymbol{D}_n$ are given by

$$(\boldsymbol{B}_n)_{ij} = \frac{1}{2}(\phi_j^n)'(x_n^+)\phi_i^n(x_n^+) - \frac{\epsilon}{2}\phi_j^n(x_n^+)(\phi_i^n)'(x_n^+) + \frac{\sigma^0}{h}\phi_j^n(x_n^+)\phi_i^n(x_n^+),$$
$$(\boldsymbol{D}_n)_{ij} = -\frac{1}{2}(\phi_j^n)'(x_n^+)\phi_i^{n-1}(x_n^-) - \frac{\epsilon}{2}\phi_j^n(x_n^+)(\phi_i^{n-1})'(x_n^-) - \frac{\sigma^0}{h}\phi_j^n(x_n^+)\phi_i^{n-1}(x_n^-).$$

Carefully examining the terms, we see that $\boldsymbol{B}_n$ (resp., $\boldsymbol{C}_n$) corresponds to the interactions of the local basis functions of the interval I_n (resp., I_{n-1}) with themselves, whereas the matrices $\boldsymbol{D}_n$ and $\boldsymbol{E}_n$ couple the intervals I_n and I_{n-1}. One can easily compute the following four 3×3 matrices, using the definitions (1.7) and (1.8):

$$\boldsymbol{B}_n = \frac{1}{h}\begin{pmatrix} \sigma^0 & 1-\sigma^0 & -2+\sigma^0 \\ -\epsilon-\sigma^0 & -1+\epsilon+\sigma^0 & 2-\epsilon-\sigma^0 \\ 2\epsilon+\sigma^0 & 1-2\epsilon-\sigma^0 & -2+2\epsilon+\sigma^0 \end{pmatrix},$$

$$\boldsymbol{C}_n = \frac{1}{h}\begin{pmatrix} \sigma^0 & -1+\sigma^0 & -2+\sigma^0 \\ \epsilon+\sigma^0 & -1+\epsilon+\sigma^0 & -2+\epsilon+\sigma^0 \\ 2\epsilon+\sigma^0 & -1+2\epsilon+\sigma^0 & -2+2\epsilon+\sigma^0 \end{pmatrix},$$

$$\boldsymbol{D}_n = \frac{1}{h}\begin{pmatrix} -\sigma^0 & -1+\sigma^0 & 2-\sigma^0 \\ -\epsilon-\sigma^0 & -1+\epsilon+\sigma^0 & 2-\epsilon-\sigma^0 \\ -2\epsilon-\sigma^0 & -1+2\epsilon+\sigma^0 & 2-2\epsilon-\sigma^0 \end{pmatrix},$$

$$\boldsymbol{E}_n = \frac{1}{h}\begin{pmatrix} -\sigma^0 & 1-\sigma^0 & 2-\sigma^0 \\ \epsilon+\sigma^0 & -1+\epsilon+\sigma^0 & -2+\epsilon+\sigma^0 \\ -2\epsilon-\sigma^0 & 1-2\epsilon-\sigma^0 & 2-2\epsilon-\sigma^0 \end{pmatrix}.$$

Finally, we compute the local matrices arising from the boundary nodes x_0 and x_N:

$$f_0 = (P^{\mathrm{DG}})'(x_0)v(x_0) - \epsilon v'(x_0)P^{\mathrm{DG}}(x_0) + \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_0)v(x_0),$$
$$f_N = -(P^{\mathrm{DG}})'(x_N)v(x_N) + \epsilon v'(x_N)P^{\mathrm{DG}}(x_N) + \frac{\sigma^0}{h}P^{\mathrm{DG}}(x_N)v(x_N).$$

These two terms above yield the matrices $\boldsymbol{F}_0$ and $\boldsymbol{F}_N$:

$$\boldsymbol{F}_0 = \frac{1}{h}\begin{pmatrix} \sigma^0 & 2-\sigma^0 & -4+\sigma^0 \\ -2\epsilon-\sigma^0 & -2+2\epsilon+\sigma^0 & 4-2\epsilon-\sigma^0 \\ 4\epsilon+\sigma^0 & 2-4\epsilon-\sigma^0 & -4+4\epsilon+\sigma^0 \end{pmatrix},$$

$$\boldsymbol{F}_N = \frac{1}{h}\begin{pmatrix} \sigma^0 & -2+\sigma^0 & -4+\sigma^0 \\ 2\epsilon+\sigma^0 & -2+2\epsilon+\sigma^0 & -4+2\epsilon+\sigma^0 \\ 4\epsilon+\sigma^0 & -2+4\epsilon+\sigma^0 & -4+4\epsilon+\sigma^0 \end{pmatrix}.$$

These local matrices are independent of the interval I_n. In the general case where the size of the intervals varies, the local matrices vary over all intervals. Once the local matrices have been computed, they are assembled into the global matrix. The assembling depends on the order of the unknowns α_i^n. Assuming that the unknowns are listed in the following order,

$$(\alpha_0^0, \alpha_1^0, \alpha_2^0, \alpha_0^1, \alpha_1^1, \alpha_2^1, \alpha_0^2, \alpha_1^2, \alpha_2^2, \ldots, \alpha_0^{N-1}, \alpha_1^{N-1}, \alpha_2^{N-1}),$$

we obtain a global matrix that is block tridiagonal:

$$\begin{pmatrix} \boldsymbol{M}_0 & \boldsymbol{D}_1 & & & & \\ \boldsymbol{E}_1 & \boldsymbol{M} & \boldsymbol{D}_2 & & & \\ & \cdots & \cdots & \cdots & & \\ & & \cdots & \cdots & \cdots & \\ & & & \boldsymbol{E}_{N-2} & \boldsymbol{M} & \boldsymbol{D}_{N-1} \\ & & & & \boldsymbol{E}_{N-1} & \boldsymbol{M}_N \end{pmatrix},$$

where

$$\boldsymbol{M} = \boldsymbol{A}_n + \boldsymbol{B}_n + \boldsymbol{C}_{n+1}, \quad \boldsymbol{M}_0 = \boldsymbol{A}_0 + \boldsymbol{F}_0 + \boldsymbol{C}_1, \quad \boldsymbol{M}_N = \boldsymbol{A}_{N-1} + \boldsymbol{F}_N + \boldsymbol{B}_{N-1}.$$

Remark: Since the penalty parameter is constant, the local matrices are independent of the subintervals. Thus, they can be defined before assembling the global matrix. Appendix B.1 contains a MATLAB® code that computes the global matrix.

1.4.2 Computing the right-hand side *b*

Each component of $\boldsymbol{b}$ is obtained by computing

$$L(\Phi_n^i) = \int_0^1 f(x)\Phi_n^i(x)dx - \epsilon K(x_0)(\Phi_n^i)'(x_0) + \frac{\sigma^0}{h}\Phi_n^i(x_0).$$

Because of the local support of Φ_n^i, the first term is reduced to

$$\int_0^1 f(x)\Phi_i^n(x)dx = \int_{x_n}^{x_{n+1}} f(x)\phi_i^n(x)dx.$$

After a change of variable, we obtain

$$\int_0^1 f(x)\Phi_i^n(x)dx = \frac{h}{2}\int_{-1}^1 f\left(\frac{h}{2}t + (n+1/2)h\right)t^i dt.$$

Because the integral cannot be computed exactly for most functions f, we rather compute an approximation by using a quadrature rule [6]. In particular, we choose the Gauss quadrature rule defined by a set of weights $(w_j)_{1\leq j\leq Q_G}$ and a set of nodes $(s_j)_{1\leq j\leq Q_G}$:

$$\int_{-1}^{1} v(t)dt \approx \sum_{j=1}^{Q_G} w_j v(s_j). \tag{1.11}$$

One can show that if v is a polynomial of degree $2Q_G - 1$, the Gauss quadrature rule is exact; i.e., the sign $\approx$ in (1.11) becomes an equality sign. Appendix A gives the sets of weights and nodes for different values of Q_G. We therefore have

$$\int_0^1 f(x)\Phi_i^n(x)dx \approx \frac{h}{2}\sum_{j=1}^{Q_G} w_j f\left(\frac{h}{2}s_j + (n+1/2)h\right)s_j^i.$$

We write the components of the vector $\boldsymbol{b}$ in an order consistent with the ordering of the unknowns α_i^n:

$$(b_0^0, b_1^0, b_2^0, b_0^1, b_1^1, b_2^1, b_0^2, b_1^2, b_2^2, \ldots, b_0^{N-1}, b_1^{N-1}, b_2^{N-1}),$$

where the first three components are

$$\begin{aligned}
b_0^0 &= \frac{h}{2}\sum_{j=1}^{Q_G} w_j f\left(\frac{h}{2}s_j + \frac{h}{2}\right) + \frac{\sigma^0}{h},\\
b_1^0 &= \frac{h}{2}\sum_{j=1}^{Q_G} w_j f\left(\frac{h}{2}s_j + \frac{h}{2}\right)s_j - \epsilon K(x_0)\frac{2}{h} - \frac{\sigma^0}{h},\\
b_2^0 &= \frac{h}{2}\sum_{j=1}^{Q_G} w_j f\left(\frac{h}{2}s_j + \frac{h}{2}\right)s_j^2 + \epsilon K(x_0)\frac{4}{h} + \frac{\sigma^0}{h},
\end{aligned}$$

and the last $3(N-1)$ components are

$$\forall 1 \leq n \leq N-1, \quad \forall 0 \leq i \leq 2, \quad b_i^n = \frac{h}{2}\sum_{j=1}^{Q_G} w_j f\left(\frac{h}{2}s_j + (n+1/2)h\right)s_j^i.$$

We remark that the first three components of $\boldsymbol{b}$ differ from the rest because of the nonzero Dirichlet boundary condition at x_0. The MATLAB code given in Appendix B.1 shows how to build $\boldsymbol{b}$ and also how to solve for $\boldsymbol{\alpha}$.

1.4.3 Imposing boundary conditions strongly

So far, we have imposed the boundary conditions (1.2), (1.3) weakly, through the addition of the terms $-\epsilon v'(x_0)p(x_0) + \frac{\sigma^0}{h}v(x_0)p(x_0)$. One can, however, impose them strongly by restricting the approximation space to

$$\mathcal{D}_k^0(\mathcal{E}_h) = \{v \in \mathcal{D}_k(\mathcal{E}_h) : v(0) = 0, v(1) = 0\}$$

and by writing

$$P^{\mathrm{DG}} = P_0^{\mathrm{DG}} + P_1$$

with P_1 being a continuous piecewise polynomial of degree k satisfying $P_1(0) = 1$ and $P_1(1) = 0$ and with $P_0^{\mathrm{DG}} \in \mathcal{D}_k^0(\mathcal{E}_h)$ a solution of the modified DG scheme:

$$\begin{aligned}
&\forall v \in \mathcal{D}_k^0(\mathcal{E}_h), \quad \sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x)(P_0^{\mathrm{DG}})'(x)v'(x)dx - \sum_{n=1}^{N-1} \{K(x_n)(P_0^{\mathrm{DG}})'(x_n)\}[v(x_n)] \\
&+\epsilon \sum_{n=1}^{N-1} \{K(x_n)v'(x_n)\}[P_0^{\mathrm{DG}}(x_n)] + \sum_{n=1}^{N-1} \frac{\sigma^0}{h}[v(x_n)][P_0^{\mathrm{DG}}(x_n)] \\
&= \int_0^1 f(x)v(x)dx - \sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x)P_1'(x)v'(x).
\end{aligned}$$

In that case, the global matrix is still block tridiagonal, but the blocks $\boldsymbol{M}_0$ and $\boldsymbol{M}_N$ are of size 2×2, the blocks $\boldsymbol{E}_1$ and $\boldsymbol{D}_{N-1}$ of size 3×2, and the blocks $\boldsymbol{E}_{N-1}$ and $\boldsymbol{D}_1$ of size 2×3.

1.5 Convergence of the DG method

One can show that if the exact solution is smooth enough, the numerical error decreases as one increases the number of intervals, i.e., as one decreases the mesh size h. We define the numerical error obtained on the mesh $\mathcal{E}_h$ by

$$e_h = p - P^{\mathrm{DG}}.$$

Definition 1.2. *Given a space V, the function $\|\cdot\| : V \to \mathbb{R}$ is called a norm if for all $v, w \in V$ and $t \in \mathbb{R}$, we have*

$$\begin{aligned}
&\text{(i)}\ \|v\| \geq 0, \\
&\text{(ii)}\ \|v\| = 0 \Leftrightarrow v = 0, \\
&\text{(iii)}\ \|tv\| = |t|\|v\|, \\
&\text{(iv)}\ \|v + w\| \leq \|v\| + \|w\|.
\end{aligned}$$

The function $\|\cdot\|$ is called a seminorm if only properties (i)*,* (iii)*, and* (iv) *are satisfied.*

Define the energy norm of the error by

$$\|e_h\|_{\mathcal{E}} = \left(\sum_{n=0}^{N-1} \int_{x_n}^{x_{n+1}} K(x)(e_h'(x))^2 dx + J_0(e_h, e_h) \right)^{1/2} \tag{1.12}$$

and the L^2 norm of the error by

$$\|e_h\|_{L^2(0,1)} = \left(\int_0^1 (e_h(x))^2 dx \right)^{1/2}.$$

Table 1.1. *Convergence rates of primal DG method for uniform meshes in one dimension.*

Method	β_1	β_2
NIPG $\sigma^0 \geq 0$	k	$k+1$ if k odd k if k even
SIPG $\sigma^0 > \sigma_*^0$	k	$k+1$
IIPG $\sigma^0 > \sigma_*^0$	k	$k+1$ if k odd k if k even

Table 1.2. *Convergence rates of primal DG method for nonuniform meshes in one dimension.*

Method	β_1	β_2
NIPG $\sigma^0 \geq 0$	k	k
SIPG $\sigma^0 > \sigma_*^0$	k	$k+1$
IIPG $\sigma^0 > \sigma_*^0$	k	k

One can show that $\|e_h\|_{\mathcal{E}} = Ch^{\beta_1}$ and $\|e_h\|_{L^2(0,1)} = Ch^{\beta_2}$, where C is a constant independent of h (see Chapter 2). The convergence rate of the method in the energy norm (resp., L^2 norm) is then defined to be the power β_1 (resp., β_2). Assuming that the solution is smooth, the mesh is uniform ($h_n = h$ for all n), and discontinuous piecewise polynomials of degree k are used, the convergence rates are summarized in Table 1.1. These rates can be proved theoretically, and they are obtained numerically for h sufficiently small by applying the formulas

$$\beta_1 = \frac{1}{\ln(2)} \ln\left(\frac{\|e_h\|_{\mathcal{E}}}{\|e_{h/2}\|_{\mathcal{E}}}\right), \quad \beta_2 = \frac{1}{\ln(2)} \ln\left(\frac{\|e_h\|_{L^2(0,1)}}{\|e_{h/2}\|_{L^2(0,1)}}\right). \tag{1.13}$$

If the meshes are nonuniform, the rates are suboptimal in L^2 norm for the NIPG and IIPG methods (see Table 1.2). The MATLAB code given in Appendix B.1 computes both the energy norm and the L^2 norm of the error.

1.6 Numerical experiments

In the case where the solution is given by the expression

$$p(x) = (1-x)e^{-x^2},$$

convergence results are obtained for both energy norm and L^2 norm of the error. We vary the parameter ϵ in $\{-1, 0, 1\}$, the penalty values σ^0, and the polynomial approximations from linear to quartic. We first consider uniform meshes of size h. The convergence rates are given in Tables 1.3–1.6 for $k = 1$, 2, and 3, respectively, and for different penalty values. We note that for $k = 1$, the choice $\sigma^0 = 0$ yields indefinite matrices, and thus the system cannot be solved.

Table 1.3. *Numerical errors and convergence rates for piecewise linear approximation.*

Method	h	$\|e_h\|_{\mathcal{E}}$	β_1	$\|e_h\|_{L^2(0,1)}$	β_2
NIPG $\sigma^0 = 1$	1/2	2.5300×10^{-1}		7.3161×10^{-2}	
	1/4	1.1630×10^{-1}	1.1211	1.9453×10^{-2}	1.9110
	1/8	5.4024×10^{-2}	1.1067	4.9477×10^{-3}	1.9752
	1/16	2.5720×10^{-2}	1.0706	1.2416×10^{-3}	1.9945
	1/32	1.2498×10^{-2}	1.0411	3.1061×10^{-4}	1.9990
SIPG $\sigma^0 = 2$	1/2	5.8471×10^{-1}		8.9892×10^{-2}	
	1/4	2.0222×10^{-1}	1.5317	1.7327×10^{-2}	2.3751
	1/8	8.5447×10^{-2}	1.2428	3.5659×10^{-3}	2.2806
	1/16	3.5828×10^{-2}	1.2539	7.3340×10^{-4}	2.2816
	1/32	1.5448×10^{-2}	1.2136	1.5613×10^{-4}	2.2318
IIPG $\sigma^0 = 1$	1/2	3.4091×10^{-1}		9.2456×10^{-2}	
	1/4	1.2112×10^{-1}	1.4929	2.5039×10^{-2}	1.8845
	1/8	5.1662×10^{-2}	1.2292	6.5011×10^{-3}	1.9454
	1/16	2.4615×10^{-2}	1.0695	1.6553×10^{-3}	1.9735
	1/32	1.2155×10^{-2}	1.0179	4.1755×10^{-4}	1.9871

Table 1.4. *Numerical errors and convergence rates for piecewise quadratic approximation.*

Method	h	$\|e_h\|_{\mathcal{E}}$	β_1	$\|e_h\|_{L^2(0,1)}$	β_2
NIPG $\sigma^0 = 0$	1/2	9.3544×10^{-2}		2.0713×10^{-2}	
	1/4	2.5299×10^{-2}	1.8865	7.9581×10^{-3}	1.3800
	1/8	6.6182×10^{-3}	1.9345	2.4210×10^{-3}	1.7168
	1/16	1.6804×10^{-3}	1.9775	6.4211×10^{-4}	1.9147
	1/32	4.2196×10^{-4}	1.9936	1.6305×10^{-4}	1.9774
NIPG $\sigma^0 = 1$	1/2	7.0690×10^{-2}		1.5754×10^{-2}	
	1/4	1.7289×10^{-2}	2.0315	5.0566×10^{-3}	1.6395
	1/8	4.2388×10^{-3}	2.0281	1.3419×10^{-3}	1.9138
	1/16	1.0472×10^{-3}	2.0171	3.3533×10^{-4}	2.0006
	1/32	2.6026×10^{-4}	2.0085	8.3121×10^{-5}	2.0123
SIPG $\sigma^0 = 2$	1/2	1.7472×10^{-1}		1.6963×10^{-2}	
	1/4	5.7965×10^{-2}	1.5917	2.8754×10^{-3}	2.5605
	1/8	9.8399×10^{-3}	2.5584	2.5109×10^{-4}	3.5174
	1/16	2.2901×10^{-3}	2.1032	2.9131×10^{-5}	3.1075
	1/32	5.6312×10^{-4}	2.0238	3.5624×10^{-6}	3.0316
IIPG $\sigma^0 = 1$	1/2	1.3032×10^{-1}		3.9401×10^{-2}	
	1/4	2.5275×10^{-2}	2.3663	7.7062×10^{-3}	2.3541
	1/8	5.6861×10^{-3}	2.1522	1.6145×10^{-3}	2.2548
	1/16	1.3649×10^{-3}	2.0586	3.6500×10^{-4}	2.1451
	1/32	3.3563×10^{-4}	2.0238	8.6547×10^{-5}	2.0763

Table 1.5. *Numerical errors and convergence rates for piecewise cubic approximation.*

Method	h	$\|e_h\|_{\mathcal{E}}$	β_1	$\|e_h\|_{L^2(0,1)}$	β_2
NIPG $\sigma^0 = 0$	1/2	5.6180×10^{-3}		1.2627×10^{-3}	
	1/4	6.4343×10^{-4}	3.1262	6.7644×10^{-5}	4.2224
	1/8	7.5240×10^{-5}	3.0962	3.8391×10^{-6}	4.1391
	1/16	9.0807×10^{-6}	3.0506	2.2809×10^{-7}	4.0730
	1/32	1.1151×10^{-6}	3.0255	1.3892×10^{-8}	4.0373
NIPG $\sigma^0 = 1$	1/2	5.2783×10^{-3}		1.0881×10^{-3}	
	1/4	6.2018×10^{-4}	3.0893	5.8542×10^{-5}	4.2162
	1/8	7.3930×10^{-5}	3.0683	3.3513×10^{-6}	4.1266
	1/16	9.0061×10^{-6}	3.0373	1.9968×10^{-7}	4.0689
	1/32	1.1107×10^{-6}	3.0193	1.2170×10^{-8}	4.0362
SIPG $\sigma^0 = 1$	1/2	6.4024×10^{-3}		4.4484×10^{-4}	
	1/4	6.8810×10^{-4}	3.2179	2.1387×10^{-5}	4.3784
	1/8	7.7514×10^{-5}	3.1501	1.1225×10^{-6}	4.2519
	1/16	9.2066×10^{-6}	3.0737	6.3845×10^{-8}	4.1360
	1/32	1.1225×10^{-6}	3.0359	3.7981×10^{-9}	4.0712
IIPG $\sigma^0 = 1$	1/2	7.3848×10^{-3}		4.3616×10^{-3}	
	1/4	6.5496×10^{-4}	3.4950	2.2715×10^{-4}	4.2631
	1/8	7.3054×10^{-5}	3.1643	1.3096×10^{-5}	4.1164
	1/16	8.8549×10^{-6}	3.0444	7.9424×10^{-7}	4.0434
	1/32	1.0983×10^{-6}	3.0111	4.9047×10^{-8}	4.0173

We now consider a nonuniform mesh constructed as follows. The unit interval is first divided into N intervals of length $1/N$. Each subinterval is then divided into three nonuniform subintervals of length $1/(7N)$, $1/(2N)$, and $5/(14N)$, respectively. Numerical errors and convergence rates are shown in Table 1.7. The rates are suboptimal in the L^2 norm for the NIPG and IIPG methods and for polynomials of degree one or two.

1.7 Bibliographical remarks

Stability and convergence of the NIPG method with zero penalty were obtained by Babuška, Baumann, and Oden [8] in one dimension. The analysis of the NIPG method with or without penalty was proved by Rivière, Wheeler, and Girault [96, 95] in any dimensions. The analysis of the IIPG method is almost identical to the analysis of the SIPG method, which can be obtained from Wheeler's work [109]. Using a standard lift argument, one can show suboptimal error estimates in the L^2 norm for both NIPG and IIPG on general meshes. In the case of uniform meshes in one dimension, Larsson and Niklasson [77] proved optimal convergence rates for polynomial degrees of even parity. The work of Cockburn, Gunzman, and Rivière [31] shows that for some nonuniform meshes, numerical rates remain suboptimal with a loss of one power of h.

Table 1.6. *Numerical errors and convergence rates for piecewise quartic approximation.*

Method	h	$\|e_h\|_{\mathcal{E}}$	β_1	$\|e_h\|_{L^2(0,1)}$	β_2
NIPG $\sigma^0 = 0$	1/2	7.4885×10^{-4}		1.1286×10^{-4}	
	1/4	5.0944×10^{-5}	3.8776	8.7699×10^{-6}	3.6859
	1/8	3.3003×10^{-6}	3.9482	5.9422×10^{-7}	3.8834
	1/16	2.0841×10^{-7}	3.9850	3.7975×10^{-8}	3.9678
	1/32	1.3061×10^{-8}	3.9960	2.3870×10^{-9}	3.9917
NIPG $\sigma^0 = 1$	1/2	7.2760×10^{-4}		1.0604×10^{-4}	
	1/4	4.8061×10^{-5}	3.9202	7.8791×10^{-6}	3.7504
	1/8	3.0614×10^{-6}	3.9726	5.2009×10^{-7}	3.9212
	1/16	1.9200×10^{-7}	3.9949	3.2856×10^{-8}	3.9845
	1/32	1.2001×10^{-8}	3.9998	2.0550×10^{-9}	3.9989
SIPG $\sigma^0 = 1$	1/2	7.8419×10^{-4}		3.7197×10^{-5}	
	1/4	5.5204×10^{-5}	3.8283	1.3837×10^{-6}	4.7485
	1/8	3.6525×10^{-6}	3.9178	4.6745×10^{-8}	4.8875
	1/16	2.3277×10^{-7}	3.9719	1.4983×10^{-9}	4.9634
	1/32	1.4642×10^{-8}	3.9907	4.7192×10^{-11}	4.9886
IIPG $\sigma^0 = 1$	1/2	1.4898×10^{-3}		8.3173×10^{-4}	
	1/4	6.7918×10^{-5}	4.4552	4.2646×10^{-5}	4.2856
	1/8	3.6617×10^{-6}	4.2132	2.2486×10^{-6}	4.2452
	1/16	2.1688×10^{-7}	4.0775	1.2739×10^{-7}	4.1416
	1/32	1.3289×10^{-8}	4.0286	7.5627×10^{-9}	4.0742

Table 1.7. *Numerical errors and convergence rates on nonuniform meshes.*

Method	N	k	$\|e_h\|_{\mathcal{E}}$	Rate	$\|e_h\|_{L^2(0,1)}$	Rate
NIPG $\sigma^0 = 1$	256	1	6.4397×10^{-4}		4.5206×10^{-6}	
	512	1	3.2190×10^{-4}	1.000	2.1943×10^{-6}	1.043
	256	2	6.6800×10^{-7}		1.5482×10^{-7}	
	512	2	1.6700×10^{-7}	2.000	3.8701×10^{-8}	2.000
SIPG $\sigma^0 = 1$	256	1	7.0270×10^{-4}		2.9266×10^{-7}	
	512	1	3.4453×10^{-4}	1.028	7.0062×10^{-8}	2.062
	256	2	4.8454×10^{-6}		1.8525×10^{-9}	
	512	2	1.2112×10^{-6}	2.000	2.3166×10^{-10}	3.000
IIPG $\sigma^0 = 1$	256	1	6.3022×10^{-4}		6.4539×10^{-7}	
	512	1	3.1511×10^{-4}	1.000	1.6133×10^{-7}	2.000
	256	2	7.8261×10^{-7}		1.7002×10^{-7}	
	512	2	1.9567×10^{-7}	2.000	4.2490×10^{-8}	2.000

Exercises

1.1. Show that the energy norm $||\cdot||_{\mathcal{E}}$ defined by (1.12) is indeed a norm for the space X_h:

$$X_h = \{v : v|_{I_n} \in \mathcal{C}^1(I_n), \quad n = 0, \ldots, N-1\}.$$

1.2. Derive the local matrices $\boldsymbol{A}_n$, $\boldsymbol{B}_n$, $\boldsymbol{C}_n$, $\boldsymbol{D}_n$, and $\boldsymbol{E}_n$ for polynomial degree $k = 4$.

1.3. Derive all the local matrices and the global matrix in the case of nonzero σ^1 and for polynomial degree $k = 2$.

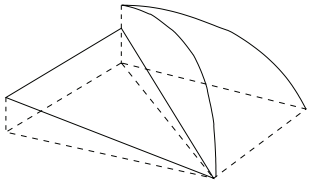
1.4. Modify the code given in Appendix B.1 so that the energy norm of the error is computed. Run the code for the following exact solutions: (a) $p(x) = (1-x)^3$, (b) $p(x) = (1-x)\cos x$. Plot the numerical solution. By varying the number of intervals, compute the numerical convergence rates for both the energy norm and the L^2 norm of the error. Choose $\sigma^0 = 1$ and $\epsilon = 1$.

1.5. Define the DG method for solving (1.1) with $K = 1$ and with the following boundary conditions:

$$p(0) = 1,$$
$$p'(1) = 0.$$

Modify the code given in Appendix B.1 and run it for the exact solution: $p(x) = (1-x)^2 e^x$. Compute numerical errors obtained for the number of intervals $N = 4, 8, 16, 32$. Choose $\epsilon = -1$ and vary $\sigma^0 = 0.1, 1, 10$.

1.6. Implement the DG method in the case where the boundary condition $p(1) = 0$ is imposed strongly, as discussed in Section 1.4.3. Compute numerical convergence rates for the L^2 norm of the error for the exact solution $p(x) = (1-x)e^{-x}$. Compare the results obtained with NIPG, SIPG, and IIPG for $\sigma^0 = 1$, $k = 2$.



Chapter 2
Higher dimensional problem

This chapter deals with the formulation and analysis of the primal DG methods NIPG, SIPG, and IIPG in two and three dimensions for a general elliptic equation. The chapter also includes a brief description of the local discontinuous Galerkin (LDG) method that is based on a mixed formulation of the elliptic equation.

2.1 Preliminaries

2.1.1 Vector notation

The gradient of a scalar function $v : \mathbb{R}^d \to \mathbb{R}$ is a vector and the divergence of a vector function $\boldsymbol{w} : \mathbb{R}^d \to \mathbb{R}^d$ is a scalar:

$$\nabla v = \left(\frac{\partial v}{\partial x_i} \right)_{1 \le i \le d}, \quad \nabla \cdot \boldsymbol{w} = \sum_{i=1}^{d} \frac{\partial w_i}{\partial x_i}.$$

The dot product between two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is

$$\boldsymbol{u} \cdot \boldsymbol{v} = \sum_{i=1}^{d} u_i v_i.$$

2.1.2 Sobolev spaces

Throughout the book, Ω denotes a bounded polygonal domain in $\mathbb{R}^d$. The vector space $L^2(\Omega)$ is the space of square-integrable functions:

$$L^2(\Omega) = \left\{ v \text{ measurable } : \int_\Omega v^2 < \infty \right\}.$$

Without going into too many details, we can say that the measure considered here is the Lebesgue measure and that the elements of $L^2(\Omega)$ are actually classes of functions: two functions v_1 and v_2 belong to the same class if and only if they differ on a set of measure

zero. We say that $v_1 = v_2$ almost everywhere (a.e. for short). The reader can refer to [97] for an introduction to Lebesgue measure.

Definition 2.1. *Let V be a vector space. A symmetric bilinear form $a : V \times V \to \mathbb{R}$ is an inner product if $a(v, v) \geq 0$ for all $v \in V$ and $a(v, v) = 0$ if and only if $v = 0$. The space V is a normed space for the norm $\|\cdot\|_V = (a(\cdot, \cdot))^{1/2}$. Furthermore, the space V equipped with an inner product is a Hilbert space if it is complete, i.e., if every Cauchy sequence is convergent. A sequence $(v_n)_n$ is said to be a Cauchy sequence if for all $\delta > 0$ there is a natural integer n_0 such that for all $n, m > n_0$, we have $\|v_n - v_m\|_V \leq \delta$. The dual space of V, denoted by V', is the space of continuous linear mappings from V to $\mathbb{R}$.*

The space $L^2(\Omega)$ is a Hilbert space with respect to the following inner product and norm:

$$(u, v)_\Omega = \int_\Omega uv, \quad \|v\|_{L^2(\Omega)} = \left(\int_\Omega v^2\right)^{1/2}.$$

We extend naturally these definitions to vector functions $\boldsymbol{u} = (u_i)_{1\leq i\leq d}$ and $\boldsymbol{v} = (v_i)_{1\leq i\leq d}$:

$$(\boldsymbol{u}, \boldsymbol{v})_\Omega = \int_\Omega \boldsymbol{u} \cdot \boldsymbol{v}, \quad \|\boldsymbol{v}\|_{L^2(\Omega)} = \left(\sum_{i=1}^{d} \|v_i\|^2_{L^2(\Omega)}\right)^{1/2}.$$

The space $L^\infty(\Omega)$ is the space of bounded functions:

$$L^\infty(\Omega) = \{v : \|v\|_{L^\infty(\Omega)} < \infty\}$$

with

$$\|v\|_{L^\infty(\Omega)} = \text{ess sup}\{|v(\boldsymbol{x})| : \boldsymbol{x} \in \Omega\}.$$

Definition 2.2. *The support of a continuous function v defined on $\mathbb{R}^d$ is the closure of the set of points at which the function is not equal to zero. If it is bounded and included in the interior of the domain Ω, then v is said to have compact support in Ω.*

Let $\mathcal{D}(\Omega)$ denote the space of $\mathcal{C}^\infty$ functions with compact support in Ω. The dual space $\mathcal{D}'(\Omega)$ is called the space of distributions. For any multi-index $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d$ and $|\alpha| = \sum_{i=1}^d \alpha_i$, the distributional derivative $D^\alpha v \in \mathcal{D}'(\Omega)$ is defined by

$$\forall \phi \in \mathcal{D}(\Omega), \quad D^\alpha v(\phi) = (-1)^{|\alpha|} \int_\Omega v(x) \frac{\partial^{|\alpha|}\phi}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}}.$$

For instance, we have

$$\forall \phi \in \mathcal{D}(\Omega), \quad \frac{\partial v}{\partial x_1}(\phi) = -\int_\Omega v \frac{\partial \phi}{\partial x_1}.$$

We introduce the Sobolev space

$$H^1(\Omega) = \left\{ v \in L^2(\Omega) : \frac{\partial v}{\partial x_i} \in L^2(\Omega), i = 1, \ldots, d \right\}.$$

It can be shown that if v belongs to $L^2(\Omega)$, then v can be identified with a distribution, still denoted by v, in the following sense:

$$\forall \phi \in \mathcal{D}(\Omega), \quad v(\phi) = \int_\Omega v\phi.$$

Therefore, for $v \in H^1(\Omega)$, we can write

$$\forall \phi \in \mathcal{D}(\Omega), \quad \frac{\partial v}{\partial x_i}(\phi) = \int_\Omega \frac{\partial v}{\partial x_i}\phi = -\int_\Omega v\frac{\partial \phi}{\partial x_i}.$$

We will write for short

$$H^1(\Omega) = \{v \in L^2(\Omega) : \nabla v \in (L^2(\Omega))^d\}.$$

Similarly, we introduce $H^s(\Omega)$ for integer s:

$$H^s(\Omega) = \{v \in L^2(\Omega) : \forall 0 \le |\alpha| \le s, D^\alpha v \in L^2(\Omega)\}.$$

In particular, in two dimensions, we have

$$H^2(\Omega) = \left\{ v \in H^1(\Omega) : \frac{\partial^2 v}{\partial x_1^2}, \frac{\partial^2 v}{\partial x_1 \partial x_2}, \frac{\partial^2 v}{\partial x_2^2} \in L^2(\Omega) \right\},$$

and we write for short

$$H^2(\Omega) = \{v \in L^2(\Omega) : \nabla^2 v \in (L^2(\Omega))^{d\times d}\}.$$

For $v \in H^s(\Omega)$, we can write for $|\alpha| \le s$:

$$\forall \phi \in \mathcal{D}(\Omega), \quad D^\alpha v(\phi) = \int_\Omega D^\alpha v\, \phi = (-1)^{|\alpha|}\int_\Omega v\frac{\partial^{|\alpha|}\phi}{\partial x_1^{\alpha_1}\cdots\partial x_d^{\alpha_d}}.$$

If v is smooth enough, we recover the usual derivatives:

$$D^\alpha v = \frac{\partial^{|\alpha|} v}{\partial x_1^{\alpha_1}\cdots\partial x_d^{\alpha_d}}.$$

The Sobolev norm associated with $H^s(\Omega)$ is

$$\|v\|_{H^s(\Omega)} = \left(\sum_{0\le|\alpha|\le s} \|D^\alpha v\|^2_{L^2(\Omega)} \right)^{1/2}.$$

The Sobolev seminorm associated with $H^s(\Omega)$ is

$$|v|_{H^s(\Omega)} = \|\nabla^s v\|_{L^2(\Omega)} = \left(\sum_{|\alpha|=s} \|D^\alpha v\|^2_{L^2(\Omega)} \right)^{1/2}.$$

Let us now define the Sobolev spaces with fractional indices. The space $H^{s+1/2}(\Omega)$ with s integer is obtained by interpolating between the spaces $H^s(\Omega)$ and $H^{s+1}(\Omega)$. The

K-interpolation method [14] is used: Given $v \in H^s(\Omega)$, we define the following splitting:

$$v = v_1 + v_2,$$

where $v_1 \in H^s(\Omega)$ and $v_2 \in H^{s+1}(\Omega)$. Then, for a given real number t, we define the kernel

$$K(v,t) = \left(\inf_{v_1+v_2=v} (\|v_1\|^2_{H^s(\Omega)} + t^2 \|v_2\|^2_{H^{s+1}(\Omega)}) \right)^{1/2}.$$

Definition 2.3. *A space V equipped with the norm $||\cdot||_V$ is said to be the completion of a subset W if, for any element $v \in V$ and any $\delta > 0$, there exists $w \in W$ such that*

$$||v - w||_V \leq \delta.$$

The space $H^{s+1/2}(\Omega)$ is then defined as the completion of all functions in $H^{s+1}(\Omega)$ with respect to the following norm:

$$\|v\|_{H^{s+1/2}(\Omega)} = \left(\int_0^\infty t^{-2} K^2(v,t) dt \right)^{1/2}.$$

Then, we have the properties

$$H^{s+1}(\Omega) \subset H^{s+1/2}(\Omega) \subset H^s(\Omega),$$
$$\forall v \in H^{s+1}(\Omega), \quad \|v\|_{H^{s+1/2}(\Omega)} \leq C(\Omega) \|v\|^{1/2}_{H^s(\Omega)} \|v\|^{1/2}_{H^{s+1}(\Omega)},$$

where $C(\Omega)$ is a positive constant that depends on the domain Ω.

An important result is the imbedding theorem that relates the Sobolev spaces to the standard spaces of $\mathcal{C}^r(\Omega)$ functions.

Theorem 2.4. *For $\Omega \subset \mathbb{R}^d$, we have*

$$H^s(\Omega) \subset \mathcal{C}^r(\Omega) \quad \text{if} \quad \frac{1}{2} < \frac{s-r}{d}.$$

To be more precise, the theorem says that under certain conditions depending on s and d, if $v \in H^s(\Omega)$, then there is a continuous representative in the equivalence class of v. The conditions are given below:

$$H^s(\Omega) \subset \mathcal{C}^0(\Omega) \quad \text{if} \quad \begin{cases} s > \frac{1}{2} & \text{for} \quad d = 1, \\ s > 1 & \text{for} \quad d = 2, \\ s > \frac{3}{2} & \text{for} \quad d = 3. \end{cases}$$

2.1.3 Trace theorems

Using distributional derivatives, we can formulate partial differential equations in the distributional sense. The notion of traces [81] is used to define the restriction of a Sobolev function along the boundary of the domain. This is important for properly defining boundary conditions.

Theorem 2.5. *Let Ω be a bounded domain with polygonal boundary $\partial\Omega$ and outward normal vector $\boldsymbol{n}$. There exist trace operators $\gamma_0 : H^s(\Omega) \to H^{s-1/2}(\partial\Omega)$ for $s > 1/2$ and*

$\gamma_1 : H^s(\Omega) \to H^{s-3/2}(\partial\Omega)$ for $s > 3/2$ that are extensions of the boundary values and boundary normal derivatives, respectively. The operators γ_j are surjective. Furthermore, if $v \in \mathcal{C}^1(\bar{\Omega})$, then

$$\gamma_0 v = v|_{\partial\Omega}, \quad \gamma_1 v = \nabla v \cdot \boldsymbol{n}|_{\partial\Omega}.$$

As a consequence, if $v \in H^1(\Omega)$, then its trace $\gamma_0 v$ belongs to $H^{1/2}(\partial\Omega)$, the interpolated space between $L^2(\partial\Omega)$ and $H^1(\partial\Omega)$. In that case, $\gamma_1 v$ may not be defined.

The subspace of $H^s(\Omega)$, $s > 1/2$, consisting of functions whose traces vanish on the boundary is denoted by

$$H_0^s(\Omega) = \{v \in H^s(\Omega) : \gamma_0 v = 0 \text{ on } \partial\Omega\}.$$

We recall some important trace inequalities that are frequently used in the analysis of the DG methods. Let E be a bounded polygonal domain with diameter h_E:

$$h_E = \sup_{\boldsymbol{x},\boldsymbol{y}\in E} \|\boldsymbol{x} - \boldsymbol{y}\|,$$

where $\|\boldsymbol{x}\|$ is the Euclidean norm ($\|\boldsymbol{x}\| = (\boldsymbol{x} \cdot \boldsymbol{x})^{1/2}$). Let $|E|$ denote the length of E in one dimension (1D), the area of E in two dimensions (2D), and the volume of E in three dimensions (3D). Similarly, we will use the length or area $|e|$ for an edge or a face of E. Then, there is a constant C independent of h_E and v such that for any $v \in H^s(E)$

$$s \geq 1 \,\forall e \subset \partial E,\ \|\gamma_0 v\|_{L^2(e)} \leq C|e|^{1/2}|E|^{-1/2}(\|v\|_{L^2(E)} + h_E\|\nabla v\|_{L^2(E)}), \tag{2.1}$$

$$s \geq 2 \,\forall e \subset \partial E,\ \|\gamma_1 v\|_{L^2(e)} \leq C|e|^{1/2}|E|^{-1/2}(\|\nabla v\|_{L^2(E)} + h_E\|\nabla^2 v\|_{L^2(E)}). \tag{2.2}$$

In the rest of the text, we will abuse the notation and replace the traces $\gamma_0 v$ and $\gamma_1 v$ by v and $\nabla v \cdot \boldsymbol{n}$, respectively.

Note that if v is a polynomial, we can take advantage of equivalence of norms in finite-dimensional spaces. Denote by $\mathbb{P}_k(E)$ the space of polynomials of degree less than or equal to k:

$$\mathbb{P}_k(E) = \text{span}\{x_1^{i_1} x_2^{i_2} \cdots x_d^{i_d} : i_1 + i_2 + \cdots + i_d \leq k, \boldsymbol{x} \in E\}.$$

The trace inequalities now become

$$\forall v \in \mathbb{P}_k(E),\ \forall e \subset \partial E,\ \|v\|_{L^2(e)} \leq \tilde{C}_t|e|^{1/2}|E|^{-1/2}\|v\|_{L^2(E)}, \tag{2.3}$$

$$\forall v \in \mathbb{P}_k(E),\ \forall e \subset \partial E,\ \|v\|_{L^2(e)} \leq C_t h_E^{-1/2}\|v\|_{L^2(E)}, \tag{2.4}$$

$$\forall v \in \mathbb{P}_k(E),\ \forall e \subset \partial E,\ \|\nabla v \cdot \boldsymbol{n}\|_{L^2(e)} \leq \tilde{C}_t|e|^{1/2}|E|^{-1/2}\|\nabla v\|_{L^2(E)}, \tag{2.5}$$

$$\forall v \in \mathbb{P}_k(E),\ \forall e \subset \partial E,\ \|\nabla v \cdot \boldsymbol{n}\|_{L^2(e)} \leq C_t h_E^{-1/2}\|\nabla v\|_{L^2(E)}. \tag{2.6}$$

Here, the constants $\tilde{C}_t, C_t$ are independent of h_E, v but depend on the polynomial degree k. In the case where E is an interval, a triangle, or a tetrahedron, one can obtain an exact expression for the constant C_t as a function of the polynomial degree [108]:

$$d = 1 \quad \forall v \in \mathbb{P}_k(E), \quad \forall t \in \partial E, \quad |v(t)| \leq \frac{k+1}{\sqrt{|E|}}\|v\|_{L^2(E)}, \tag{2.7}$$

$$d = 2 \quad \forall v \in \mathbb{P}_k(E), \quad \|v\|_{L^2(e)} \leq \sqrt{\frac{(k+1)(k+2)}{2}\frac{|e|}{|E|}}\|v\|_{L^2(E)}, \tag{2.8}$$

$$d = 3 \quad \forall v \in \mathbb{P}_k(E), \quad \|v\|_{L^2(e)} \leq \sqrt{\frac{(k+1)(k+3)}{3}\frac{|e|}{|E|}}\|v\|_{L^2(E)}. \tag{2.9}$$

2.1.4 Approximation properties

In this section, we state approximation results in the space of polynomials of degree k (see [9, 96]).

Theorem 2.6. *Let E be a triangle or parallelogram in 2D or a tetrahedron or hexahedron in 3D. Let $v \in H^s(E)$ for $s \geq 1$. Let $k \geq 0$ be an integer. There exist a constant C independent of v and h_E and a function $\tilde{v} \in \mathbb{P}_k(E)$ such that*

$$\forall 0 \leq q \leq s, \quad \|v - \tilde{v}\|_{H^q(E)} \leq C h_E^{\min(k+1,s)-q}|v|_{H^s(E)}. \tag{2.10}$$

As a consequence, if Ω is subdivided into triangles or tetrahedra, one can construct a global approximation $\tilde{v}$ that is continuous over the domain Ω and satisfies the same approximation result (2.10). If Ω is subdivided into parallelograms or hexahedra, the same result holds if the space $\mathbb{P}_k(E)$ is replaced by the space $\mathbb{Q}_k(E)$, namely the space of polynomials of degree less than or equal to k in each space direction.

The next result yields an approximation that conserves the average of the normal flux on each edge.

Theorem 2.7. *Let E be a triangle or parallelogram in 2D or a tetrahedron in 3D. Denote by $\boldsymbol{n}_E$ the outward normal to E. Let $v \in H^s(E)$ for $s \geq 2$. Let $\boldsymbol{K}$ be a symmetric positive definite matrix with constant entries. There exists an approximation $\tilde{v} \in \mathbb{P}_k(E)$ of v satisfying*

$$\int_e \boldsymbol{K}\nabla(\tilde{v} - v) \cdot \boldsymbol{n}_E = 0 \quad \forall e \in \partial E$$

and the optimal error bounds

$$\forall i = 0, 1, 2, \quad \|\nabla^i(\tilde{v} - v)\|_{L^2(E)} \leq C h_E^{\min(k+1,s)-i}|v|_{H^s(E)}, \tag{2.11}$$

where C is independent of h_E.

If the matrix $\boldsymbol{K}$ is a function of space, the previous result is still valid for small enough h_E. The proof of this theorem for a triangle or a tetrahedron is given in Appendix C.

2.1.5 Green's theorem

Given E a bounded domain and $\boldsymbol{n}_E$ the outward normal vector to ∂E, we have for all $v \in H^2(E)$ and $w \in H^1(E)$

$$-\int_E w\Delta v = \int_E \nabla v \cdot \nabla w - \int_{\partial E} \nabla v \cdot \boldsymbol{n}_E w, \tag{2.12}$$

where $\Delta w = \nabla \cdot \nabla w = \sum_{i=1}^{d} \frac{\partial^2 w}{\partial x_i^2}$. A more generalized Green's theorem is

$$-\int_E w \nabla \cdot \boldsymbol{F} \nabla v = \int_E \boldsymbol{F} \nabla v \cdot \nabla w - \int_{\partial E} \boldsymbol{F} \nabla v \cdot \boldsymbol{n}_E w, \tag{2.13}$$

where $\boldsymbol{F}$ is a matrix-valued function.

2.1.6 Cauchy–Schwarz's and Young's inequalities

The following two inequalities are used at several places in this text.

Cauchy–Schwarz's inequality:

$$\forall f, g \in L^2(\Omega), \quad |(f, g)_\Omega| \leq \|f\|_{L^2(\Omega)} \|g\|_{L^2(\Omega)}. \tag{2.14}$$

Young's inequality:

$$\forall \epsilon > 0, \quad \forall a, b \in \mathbb{R}, \quad ab \leq \frac{\epsilon}{2} a^2 + \frac{1}{2\epsilon} b^2. \tag{2.15}$$

2.2 Model problem

Let Ω be a polygonal domain in $\mathbb{R}^d$, $d = 2$ or 3. The sides of the boundary $\partial\Omega$ of the domain are grouped into two disjoint sets Γ_D and Γ_N. Let $\boldsymbol{n}$ be the unit normal vector to the boundary exterior to Ω. For f given in $L^2(\Omega)$, g_D given in $H^{\frac{1}{2}}(\Gamma_D)$, and g_N given in $L^2(\Gamma_N)$, we consider the following elliptic problem:

$$-\nabla \cdot (\boldsymbol{K} \nabla p) + \alpha p = f \quad \text{in} \quad \Omega, \tag{2.16}$$

$$p = g_D \quad \text{on} \quad \Gamma_D, \tag{2.17}$$

$$\boldsymbol{K} \nabla p \cdot \boldsymbol{n} = g_N \quad \text{on} \quad \Gamma_N. \tag{2.18}$$

The coefficient $\boldsymbol{K}$ is a matrix-valued function $\boldsymbol{K} = (k_{ij})_{1 \leq i,j \leq d}$ that is symmetric ($k_{ij} = k_{ji}$) positive definite and bounded below and above uniformly; i.e., there exist two positive constants K_0 and K_1 such that

$$\forall \boldsymbol{x} \in \mathbb{R}^d, \quad K_0 \boldsymbol{x} \cdot \boldsymbol{x} \leq \boldsymbol{K} \boldsymbol{x} \cdot \boldsymbol{x} \leq K_1 \boldsymbol{x} \cdot \boldsymbol{x}. \tag{2.19}$$

The other coefficient α is a nonnegative scalar function. The second equation (2.17) is called a Dirichlet boundary condition. The value of the solution is prescribed on Γ_D. The third equation (2.18) is called a Neumann boundary condition. The normal derivative or flux is prescribed on Γ_N.

The problem (2.16)–(2.18) has a solution $p \in \mathcal{C}^2(\bar{\Omega})$, called strong solution under additional smoothness on the data f, g_D, g_N, $\boldsymbol{K}$, and α. The equations are then satisfied pointwisely. With the definition of weak derivatives, we can rewrite the partial differential equation into a weak form and define a weak solution.

2.2.1 Weak solution

For simplicity, assume that $\partial\Omega = \Gamma_D$. From the trace theorem, there is an extension of $g_D \in H^{1/2}(\partial\Omega)$ in Ω. Let $p_D \in H^1(\Omega)$ be the extension:

$$p_D = g_D \quad \text{on} \quad \partial\Omega.$$

The variational formulation (or weak formulation) of problem (2.16)–(2.18) is as follows: Find $p = p_D + w$ with $w \in H_0^1(\Omega)$ such that

$$\forall v \in H_0^1(\Omega), \quad \int_\Omega (\boldsymbol{K}\nabla w \cdot \nabla v + \alpha w v) = \int_\Omega f v - \int_\Omega (\boldsymbol{K}\nabla p_D \cdot \nabla v + \alpha p_D v). \quad (2.20)$$

The solution p is called the weak solution to problem (2.16)–(2.18). Existence and uniqueness of w is a consequence of the Lax–Milgram theorem given below [79].

Theorem 2.8. *Let V be a real Hilbert space. Let $a : V \times V \to \mathbb{R}$ be a bilinear form that is*
(i) *continuous: $|a(u, v)| \le C_1\|u\|_V\|v\|_V$,*
(ii) *coercive: $C_2\|u\|_V^2 \le a(u, u)$, with positive constants C_1 and C_2.*
Let $L : V \to \mathbb{R}$ be a continuous linear functional. Then, there exists a unique $u \in V$ satisfying

$$\forall v \in V, \quad a(u, v) = L(v).$$

Moreover, the solution u is bounded by the data

$$\|u\|_V \le \frac{1}{C_2}\|L\|.$$

If $\partial\Omega = \Gamma_N$ and $\alpha = 0$, the weak solution is unique up to an additive constant, provided the compatibility condition $\int_\Omega f + \int_{\partial\Omega} g_N = 0$ is satisfied. Indeed, this condition is obtained by integrating (2.16) over Ω and by using Green's theorem.

2.2.2 Numerical solution

There are several methods available for solving problem (2.16)–(2.18). We mention here two basic ones: finite difference method and finite element method.

The finite difference method approximates the partial derivatives by finite differences. Let the domain be subdivided into uniform squares with vertices $A_{ij}(x_i, y_j)$ for $1 \le i, j \le M$. This grid is characterized by the length of the side of a square denoted by h (see Fig. 2.1). We have

$$\frac{\partial^2 p}{\partial x^2}(A_{ij}) \approx \frac{p(x_{i-1}, y_j) - 2p(x_{i,j}) + p(x_{i+1}, y_j)}{h^2},$$
$$\frac{\partial^2 p}{\partial y^2}(A_{ij}) \approx \frac{p(x_i, y_{j-1}) - 2p(x_{i,j}) + p(x_i, y_{j+1})}{h^2}.$$

The finite difference solution is a set of values P_{ij} approximating $p(x_i, y_j)$. For instance, the finite difference method applied to the Poisson equation $-\Delta p = f$ is

$$\forall i, j, \quad -\frac{P_{i-1,j} - 2P_{ij} + P_{i+1,j}}{h^2} - \frac{P_{i,j-1} - 2P_{i,j} + P_{i,j+1}}{h^2} = f(x_i, y_j).$$

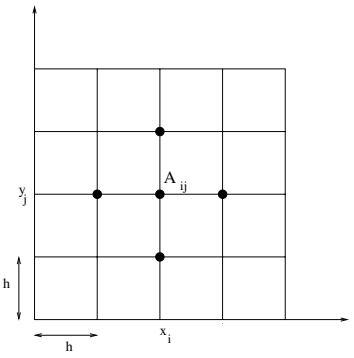


Figure 2.1. *Finite difference grid.*

After taking account of the boundary conditions, we obtain a linear system with unknowns P_{ij}. This method is easy to implement. However, the accuracy is limited, as the method is of low order and the method is not well suited to complicated geometries.

The finite element method uses the variational formulation of the partial differential equation. Let Ω be partitioned into elements (for instance triangles or rectangles in 2D) that form a mesh. Let X_h be the finite-dimensional subspace of $H_0^1(\Omega)$, consisting of continuous piecewise polynomials of degree k on each element. Based on (2.20), the finite element method is to find $P_h = \tilde{p}_{\mathrm{D}} + W_h$ with $W_h \in X_h$ such that

$$\forall v \in X_h, \quad \int_\Omega (\boldsymbol{K} \nabla W_h \cdot \nabla v + \alpha W_h v) = \int_\Omega f v - \int_\Omega (\boldsymbol{K} \nabla \tilde{p}_{\mathrm{D}} \cdot \nabla v + \alpha \widetilde{p_{\mathrm{D}}} v). \tag{2.21}$$

The function $\tilde{p}_{\mathrm{D}} \in X_h$ is an interpolant of the extension p_{D}. Finite element methods were first introduced by engineers in the 1950s. The mathematical theory was developed in the late 1960s for steady-state problems. We refer the reader to [28, 17] for a general treatment of the theory. Compared to the finite difference methods, finite element methods offer several attractive features: their accuracy depends on the polynomial degree k; they can handle complicated geometries by the use of unstructured grids. However, these methods are not locally mass conservative (see Section 2.7.3), which means that in nonlinear reactive transport problems, finite difference methods still prevailed. Another issue is the rather complicated use of local mesh refinement.

DG methods also use a variational formulation of the problem. In that sense, DG and finite element methods share many properties, and we can abuse the terminology by saying that the DG method is a particular type of finite element method. In addition to the high order of accuracy and the use of unstructured meshes, DG methods are locally mass conservative, and they easily handle local mesh refinement. A more detailed comparison of the finite element method with DG is given in Section 2.12.

2.3 Broken Sobolev spaces

Broken Sobolev spaces are natural spaces to work with the DG methods. These spaces depend strongly on the partition of the domain. Let Ω be a polygonal domain subdivided into

elements E, where E is a triangle or a quadrilateral in 2D, or a tetrahedron or hexahedron in 3D. For simplicity, we assume that the intersection of two elements is either empty, a vertex, an edge, or a face. Such a mesh is called a conforming mesh. The resulting subdivision (or mesh) is denoted by $\mathcal{E}_h$, and h is the maximum element diameter. We also assume that the subdivision is regular [28]. This means that if h_E denotes the diameter of E and ρ_E denotes the maximum diameter of a ball inscribed in E, there is a constant $\rho > 0$ such that

$$\forall E \in \mathcal{E}_h, \quad \frac{h_E}{\rho_E} \leq \rho.$$

We introduce the broken Sobolev space for any real number s,

$$H^s(\mathcal{E}_h) = \{v \in L^2(\Omega) : \forall E \in \mathcal{E}_h, v|_E \in H^s(E)\},$$

equipped with the broken Sobolev norm:

$$|\!|\!| v |\!|\!|_{H^s(\mathcal{E}_h)} = \left(\sum_{E \in \mathcal{E}_h} \|v\|^2_{H^s(E)} \right)^{1/2}.$$

In particular, we will use the broken gradient seminorm:

$$|\!|\!| \nabla v |\!|\!|_{H^0(\mathcal{E}_h)} = \left(\sum_{E \in \mathcal{E}_h} \|\nabla v\|^2_{L^2(E)} \right)^{1/2}.$$

Clearly, we have

$$H^s(\Omega) \subset H^s(\mathcal{E}_h) \quad \text{and} \quad H^{s+1}(\mathcal{E}_h) \subset H^s(\mathcal{E}_h).$$

In Sections 3.1.4, 5.1.2, and 7.1.1, the classical Poincaré inequality, Korn's inequality, and a Sobolev imbedding are generalized for the broken Sobolev space.

2.3.1 Jumps and averages

We denote by Γ_h the set of interior edges (or faces) of the subdivision $\mathcal{E}_h$. With each edge (or face) e, we associate a unit normal vector $\boldsymbol{n}_e$. If e is on the boundary $\partial\Omega$, then $\boldsymbol{n}_e$ is taken to be the unit outward vector normal to $\partial\Omega$.

If v belongs to $H^1(\mathcal{E}_h)$, the trace of v along any side of one element E is well defined. If two elements E_1^e and E_2^e are neighbors and share one common side e, there are two traces of v along e. We can add or subtract those values, and we obtain an average and a jump for v. We assume that the normal vector $\boldsymbol{n}_e$ is oriented from E_1^e to E_2^e:

$$\{v\} = \frac{1}{2}(v|_{E_1^e}) + \frac{1}{2}(v|_{E_2^e}), \quad [v] = (v|_{E_1^e}) - (w|_{E_2^e}) \quad \forall e = \partial E_1^e \cap \partial E_2^e.$$

As in the one-dimensional case, by convention, we extend the definition of jump and average to sides that belong to the boundary $\partial\Omega$:

$$\{v\} = [v] = (v|_{E_1^e}) \quad \forall e = \partial E_1^e \cap \partial\Omega.$$

2.4 Variational formulation

In what follows, we assume that $s > 3/2$. We introduce two bilinear forms $J_0^{\sigma_0,\beta_0}$, $J_1^{\sigma_1,\beta_1} : H^s(\mathcal{E}_h) \times H^s(\mathcal{E}_h) \to \mathbb{R}$ that penalize the jump of the function values and the jump of the normal derivatives values:

$$J_0^{\sigma_0,\beta_0}(v, w) = \sum_{e\in\Gamma_h\cup\Gamma_D} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e [v][w],$$

$$J_1^{\sigma_1,\beta_1}(v, w) = \sum_{e\in\Gamma_h} \frac{\sigma_e^1}{|e|^{\beta_1}} \int_e [\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e][\boldsymbol{K}\nabla w \cdot \boldsymbol{n}_e].$$

The parameters σ_e^0 and σ_e^1 are called penalty parameters. They are nonnegative real numbers. The powers β_0 and β_1 are positive numbers that depend on the dimension d. All parameters will be specified later. We recall that the notation $|e|$ simply means the length of e in 2D and the area of e in 3D. We clearly have

$$\forall e \subset \partial E, \quad |e| \le h_E^{d-1} \le h^{d-1}. \tag{2.22}$$

We now define the DG bilinear forms $a_\epsilon : H^s(\mathcal{E}_h) \times H^s(\mathcal{E}_h) \to \mathbb{R}$:

$$\begin{aligned} a_\epsilon(v, w) = &\sum_{E\in\mathcal{E}_h} \int_E \boldsymbol{K}\nabla v \cdot \nabla w + \int_\Omega \alpha v w \\ &- \sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\,[w] + \epsilon \sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \{\boldsymbol{K}\nabla w \cdot \boldsymbol{n}_e\}\,[v] \\ &+ J_0^{\sigma_0,\beta_0}(v, w) + J_1^{\sigma_1,\beta_1}(v, w). \end{aligned} \tag{2.23}$$

The bilinear form a_ϵ contains another parameter ϵ that may take the value -1, 0, or 1. As in Section 1.2, we have the following symmetry property: a_ϵ is symmetric if $\epsilon = -1$ and it is nonsymmetric otherwise.

We also define the following linear form:

$$L(v) = \int_\Omega f v + \epsilon \sum_{e\in\Gamma_D} \int_e \left(\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e + \frac{\sigma_e^0}{|e|^{\beta_0}} v\right) g_D + \sum_{e\in\Gamma_N} \int_e v g_N.$$

Cauchy–Schwarz's inequality and trace inequalities imply that all integral terms in the forms defined above make sense if the functions belong to $H^s(\mathcal{E}_h)$ for any $s > 3/2$.

The general DG variational formulation of problem (2.16)–(2.18) is as follows: Find p in $H^s(\mathcal{E}_h)$, $s > 3/2$, such that

$$\forall v \in H^s(\mathcal{E}_h), \quad a_\epsilon(p, v) = L(v). \tag{2.24}$$

Remark: We note that the problem (2.24) is independent of the choice of the normal $\boldsymbol{n}_e$. Indeed, let e be one edge (or face) shared by two elements E_i and E_j. Let $\boldsymbol{n}_{ij}$ be the unit normal vector pointing from E_i to E_j. If $\boldsymbol{n}_e$ coincides with $\boldsymbol{n}_{ij}$, we have

$$\{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}[w] = \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_{ij}\}(w|_{E_i} - w|_{E_j}).$$

If $\boldsymbol{n}_e$ has the opposite direction to $\boldsymbol{n}_{ij}$, the jump $[w]$ has a different sign and

$$\{\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\}[w]=\{\boldsymbol{K}\nabla v\cdot(-\boldsymbol{n}_{ij})\}(w|_{E_j}-w|_{E_i}),$$

which gives the same expression as above.

2.4.1 Consistency

The next proposition establishes the equivalence between the model problem and the variational formulation.

Proposition 2.9. *Let $s>3/2$. Assume that the weak solution p of problem* (2.16)–(2.18) *belongs to $H^s(\mathcal{E}_h)$; then p satisfies the variational problem* (2.24)*. Conversely, if $p\in H^1(\Omega)\cap H^s(\mathcal{E}_h)$ satisfies* (2.24)*, then p is the solution of problem* (2.16)–(2.18)*.*

Proof. First, we prove that if the solution p of (2.16)–(2.18) belongs to $H^s(\Omega)$, then it also solves (2.24). For this, let v be an element in $H^s(\mathcal{E}_h)$. We multiply (2.16) by v, integrate on one element E, and use Green's theorem (2.13):

$$\int_E(\boldsymbol{K}\nabla p\cdot\nabla v+\alpha pv)-\int_{\partial E}\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_E v=\int_E fv.$$

We recall that $\boldsymbol{n}_E$ is the outward normal to E. We sum over all elements, switch to the normal vectors $\boldsymbol{n}_e$, and observe that

$$\sum_{E\in\mathcal{E}_h}\int_{\partial E}\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_E v=\sum_{e\in\Gamma_h}\int_e[\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e v]+\sum_{e\in\partial\Omega}\int_e\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e v. \tag{2.25}$$

By regularity of the solution p, we have

$$\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e=\{\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e\}\quad\text{a.e.}$$

Therefore, we obtain the resulting equation

$$\sum_{E\in\mathcal{E}_h}\int_E(\boldsymbol{K}\nabla p\cdot\nabla v+\alpha pv)-\sum_{e\in\Gamma_h}\int_e\{\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e\}[v]-\int_{\partial\Omega}(\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e)v=\int_\Omega fv.$$

Using the Neumann boundary condition (2.18), we get

$$\begin{aligned}\sum_{E\in\mathcal{E}_h}\int_E(\boldsymbol{K}\nabla p\cdot\nabla v+\alpha pv)-\sum_{e\in\Gamma_h}\int_e\{\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e\}[v]\\ -\sum_{e\in\Gamma_{\mathrm{D}}}\int_e(\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e)v=\int_\Omega fv+\sum_{e\in\Gamma_{\mathrm{N}}}\int_e g_{\mathrm{N}}v.\end{aligned}$$

We add $\epsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e(\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e)p$ and $\sum_{e\in\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|^{\beta_0}}\int_e pv$ to both sides and use the Dirichlet boundary condition (2.17):

$$\sum_{E\in\mathcal{E}_h}\int_E(\boldsymbol{K}\nabla p\cdot\nabla v+\alpha pv)-\sum_{e\in\Gamma_h}\int_e\{\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e\}[v]-\sum_{e\in\Gamma_{\mathrm{D}}}\int_e(\boldsymbol{K}\nabla p\cdot\boldsymbol{n}_e)v$$

$$+\epsilon \sum_{e\in\Gamma_{\mathrm{D}}} \int_e (\boldsymbol{K}\nabla v\cdot \boldsymbol{n}_e)p + \sum_{e\in\Gamma_{\mathrm{D}}} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e pv = \int_\Omega fv + \epsilon \sum_{e\in\Gamma_{\mathrm{D}}} \int_e (\boldsymbol{K}\nabla v\cdot \boldsymbol{n}_e) g_{\mathrm{D}} + \sum_{e\in\Gamma_{\mathrm{N}}} \int_e g_{\mathrm{N}} v + \sum_{e\in\Gamma_{\mathrm{D}}} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e g_{\mathrm{D}} v.$$

Finally, we note that the jumps $[p] = [\boldsymbol{K}\nabla p\cdot \boldsymbol{n}_e]$ are zero a.e. on the interior edges (or faces). Then, we clearly have (2.24).

Conversely, take $v \in \mathcal{D}(E)$. Then (2.24) reduces to

$$\sum_{E\in\mathcal{E}_h} \int_E \boldsymbol{K}\nabla p\cdot\nabla v + \int_\Omega \alpha pv = \int_\Omega fv,$$

which immediately yields in the distributional sense, for all $E \in \mathcal{E}_h$,

$$-\nabla\cdot \boldsymbol{K}\nabla p + \alpha p = f \quad \text{in} \quad E. \tag{2.26}$$

Next, let e be an interior edge (or face) and let E_e^1 and E_e^2 be the two elements adjacent to e. Take $v \in H_0^2(E_e^1 \cup E_e^2)$ and extend it by zero over the rest of the domain. On one hand, if we multiply (2.26) by v and use Green's theorem (2.13), we have

$$\int_{E_e^1\cup E_e^2} \boldsymbol{K}\nabla p\cdot\nabla v + \int_{E_e^1\cup E_e^2} \alpha pv - \int_e [\boldsymbol{K}\nabla p\cdot \boldsymbol{n}_e]v = \int_{E_e^1\cup E_e^2} fv. \tag{2.27}$$

On the other hand, since $[v] = 0$, (2.24) reduces to

$$\int_{E_e^1\cup E_e^2} \boldsymbol{K}\nabla p\nabla v + \int_{E_e^1\cup E_e^2} \alpha pv = \int_{E_e^1\cup E_e^2} fv.$$

Hence, we have

$$\forall v \in H_0^2(E_e^1 \cup E_e^2), \quad \int_e [\boldsymbol{K}\nabla p\cdot \boldsymbol{n}_e]v = 0.$$

This implies that $[\boldsymbol{K}\nabla p\cdot \boldsymbol{n}_e]|_e = 0$ in $L^2(e)$. Since this holds for all e, it implies that $\nabla\cdot \boldsymbol{K}\nabla p \in L^2(\Omega)$, and hence we have globally

$$-\nabla\cdot \boldsymbol{K}\nabla p + \alpha p = f \quad \text{in} \quad \Omega. \tag{2.28}$$

To recover the Dirichlet boundary conditions, we multiply (2.28) by a function v in $H^2(\Omega)\cap H_0^1(\Omega)$, apply Green's theorem (2.13), and compare with (2.24):

$$-\sum_{e\in\Gamma_{\mathrm{D}}} \int_e (\boldsymbol{K}\nabla v\cdot \boldsymbol{n}_e)(p - g_{\mathrm{D}}) = 0.$$

This being true for all $v \in H^2(\Omega) \cap H_0^1(\Omega)$, we have $p = g_{\mathrm{D}}$ on Γ_{D}. Finally, choosing $v \in H^2(\Omega)$, $v|_{\Gamma_{\mathrm{D}}} = 0$, we find

$$-\sum_{e\in\Gamma_{\mathrm{N}}} \int_e (\boldsymbol{K}\nabla p\cdot \boldsymbol{n}_e)v = -\sum_{e\in\Gamma_{\mathrm{N}}} \int_e gv,$$

and this gives the other boundary condition. We clearly have (2.18). □

2.5 Finite element spaces

We will consider finite-dimensional subspaces of the broken Sobolev space $H^s(\mathcal{E}_h)$ for $s > 3/2$. Let k be a positive integer. The finite element subspace is taken to be

$$\mathcal{D}_k(\mathcal{E}_h) = \{v \in L^2(\Omega) : \forall E \in \mathcal{E}_h, v|_E \in \mathbb{P}_k(E)\}, \tag{2.29}$$

where $\mathbb{P}_k(E)$ denotes the space of polynomials of total degree less than or equal to k. We will refer to the functions in $\mathcal{D}_k(\mathcal{E}_h)$ as test functions. We note that the test functions are discontinuous along the edges (or faces) of the mesh.

As is done in the classical finite element method, each mesh element E (also called physical element) is mapped to a reference element $\hat{E}$, and all computations are done on the reference element. The following section introduces triangular, quadrilateral, and tetrahedral reference elements.

2.5.1 Reference elements versus physical elements

When implementing the DG method, one has to compute integrals over volumes (such as triangles or quadrilaterals in 2D, tetrahedra or hexahedra in 3D) and faces (such as edges in 2D, triangles or quadrilaterals in 3D). It would be too costly to compute the integrals over each physical element in the mesh. A more economical and effective approach is to use a change of variables to obtain an integral on a fixed element, called the reference element [28, 101].

Reference triangular element: It consists of a triangle $\hat{E}$ with vertices $\hat{A}_1(0, 0)$, $\hat{A}_2(1, 0)$, and $\hat{A}_3(0, 1)$ (see Fig. 2.2). For a given physical element E, there is an affine map F_E from the reference element onto E. If E has vertices $A_i(x_i, y_i)$ for $i = 1, 2, 3$, then the map F_E is defined by

$$F_E\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}, \quad x = \sum_{i=1}^{3} x_i \hat{\phi}_i(\hat{x}, \hat{y}), \quad y = \sum_{i=1}^{3} y_i \hat{\phi}_i(\hat{x}, \hat{y}),$$

where

$$\begin{aligned} \hat{\phi}_1(\hat{x}, \hat{y}) &= 1 - \hat{x} - \hat{y}, \\ \hat{\phi}_2(\hat{x}, \hat{y}) &= \hat{x}, \\ \hat{\phi}_3(\hat{x}, \hat{y}) &= \hat{y}. \end{aligned}$$

We can rewrite the mapping

$$\begin{pmatrix} x \\ y \end{pmatrix} = F_E\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \boldsymbol{B}_E \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} + \mathbf{b}_E, \tag{2.30}$$

where $\boldsymbol{B}_E$ is a 2×2 matrix and $\mathbf{b}_E$ a vector. It is easy to show that

$$\boldsymbol{B}_E = \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}, \quad \mathbf{b}_E = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}.$$

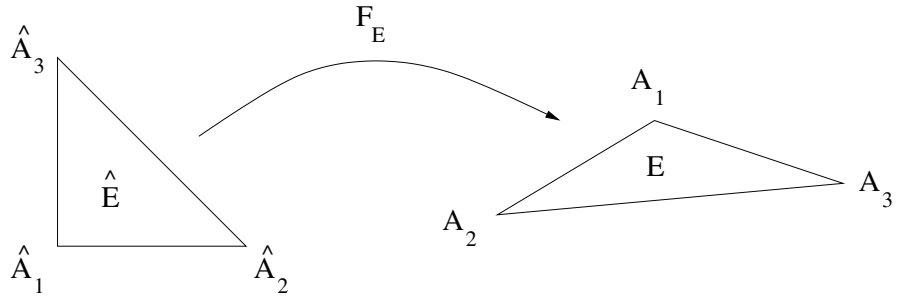


Figure 2.2. *Reference triangular element $\hat{E}$ and physical element E.*

The determinant of $\boldsymbol{B}_E$ appears in the computation of the integrals. If $|E|$ denotes the area of E, then we have

$$\det(\boldsymbol{B}_E) = 2|E|. \tag{2.31}$$

Thus $\boldsymbol{B}_E$ is invertible and the matrix norm (induced by the Euclidean norm) of $\boldsymbol{B}_E$ and $\boldsymbol{B}_E^{-1}$ is bounded as follows:

$$\|\boldsymbol{B}_E\| \equiv \sup_{(\hat{x},\hat{y})\in\hat{E}} \frac{\|\boldsymbol{B}_E\binom{\hat{x}}{\hat{y}}\|}{\|\binom{\hat{x}}{\hat{y}}\|} \leq \frac{h_E}{\hat{\rho}}, \quad \|\boldsymbol{B}_E^{-1}\| \leq \frac{\hat{h}}{\rho_E}.$$

Here, $\hat{h}$ denotes the diameter of $\hat{E}$ and $\hat{\rho}$ denotes the diameter of the largest circle inscribed in $\hat{E}$. Similarly, ρ_E denotes the diameter of the largest circle inscribed in E.

The mapping F_E corresponds to a change of variable. We denote

$$\hat{v} = v \circ F_E.$$

In other words, $\hat{v}(\hat{x}, \hat{y}) = v(x, y)$. We also denote by $\hat{\nabla}\hat{v}$ the gradient of $\hat{v}$ with respect to $\hat{x}$ and $\hat{y}$:

$$\hat{\nabla}\hat{v} = \begin{pmatrix} \frac{\partial \hat{v}}{\partial \hat{x}} \\ \frac{\partial \hat{v}}{\partial \hat{y}} \end{pmatrix}.$$

We can prove that

$$\hat{\nabla}\hat{v} = \boldsymbol{B}_E^T \nabla v \circ F_E, \tag{2.32}$$

where $\boldsymbol{B}_E^T$ is the transpose of the matrix $\boldsymbol{B}_E$ (i.e., $(\boldsymbol{B}_E^T)_{ij} = (\boldsymbol{B}_E)_{ji}$).

Reference quadrilateral element: It consists of the square $\hat{E}$ with vertices $\hat{A}_1(-1, -1)$, $\hat{A}_2(1, -1)$, $\hat{A}_3(1, 1)$, and $\hat{A}_4(-1, 1)$ (see Fig. 2.3). If E has vertices $A_i(x_i, y_i)$ for $i = 1, \ldots, 4$, the transformation map $F_E : \hat{E} \to E$ is defined by

$$F_E\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}, \quad x = \sum_{i=1}^{4} x_i \hat{\phi}_i(\hat{x}, \hat{y}), \quad y = \sum_{i=1}^{4} y_i \hat{\phi}_i(\hat{x}, \hat{y}), \tag{2.33}$$

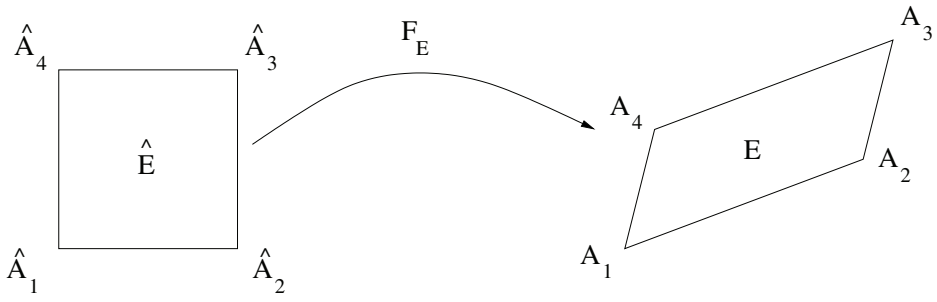


Figure 2.3. *Reference quadrilateral element $\hat{E}$ and physical element E.*

where

$$\hat{\phi}_1(\hat{x}, \hat{y}) = \frac{1}{4}(1 - \hat{x})(1 - \hat{y}),$$
$$\hat{\phi}_2(\hat{x}, \hat{y}) = \frac{1}{4}(1 + \hat{x})(1 - \hat{y}),$$
$$\hat{\phi}_3(\hat{x}, \hat{y}) = \frac{1}{4}(1 + \hat{x})(1 + \hat{y}),$$
$$\hat{\phi}_4(\hat{x}, \hat{y}) = \frac{1}{4}(1 - \hat{x})(1 + \hat{y}).$$

The mapping F_E is affine if the physical element E is a parallelogram. In the general case, we define $\boldsymbol{B}_E$ to be the Jacobian matrix of F_E:

$$\boldsymbol{B}_E = \begin{pmatrix} \frac{\partial x}{\partial \hat{x}} & \frac{\partial x}{\partial \hat{y}} \\ \frac{\partial y}{\partial \hat{x}} & \frac{\partial y}{\partial \hat{y}} \end{pmatrix}.$$

It is sufficient to have the determinant of $\boldsymbol{B}_E$ nonvanishing in order to have an invertible map F_E. This condition is satisfied if E is convex.

Reference tetrahedral element: It consists of the tetrahedron $\hat{E}$ with vertices $\hat{A}_1(0, 0, 0)$, $\hat{A}_2(1, 0, 0)$, $\hat{A}_3(0, 1, 0)$, and $\hat{A}_4(0, 0, 1)$. There is an affine map $F_E : \hat{E} \to E$, defined from the coordinates of the vertices $A_i(x_i, y_i)$:

$$F_E \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix},$$

$$x = \sum_{i=1}^{4} x_i \hat{\phi}_i(\hat{x}, \hat{y}, \hat{z}), \quad y = \sum_{i=1}^{4} y_i \hat{\phi}_i(\hat{x}, \hat{y}, \hat{z}), \quad z = \sum_{i=1}^{4} z_i \hat{\phi}_i(\hat{x}, \hat{y}, \hat{z}),$$

where

$$\hat{\phi}_1(\hat{x}, \hat{y}, \hat{z}) = 1 - \hat{x} - \hat{y} - \hat{z},$$
$$\hat{\phi}_2(\hat{x}, \hat{y}, \hat{z}) = \hat{x},$$

$$\hat{\phi}_3(\hat{x}, \hat{y}, \hat{z}) = \hat{y},$$
$$\hat{\phi}_4(\hat{x}, \hat{y}, \hat{z}) = \hat{z}.$$

All properties for the reference triangle are valid for the reference tetrahedron.

Remark on choice of finite element spaces: We recall that the DG finite element space $\mathcal{D}_k(\mathcal{E}_h)$ is the space of discontinuous polynomials defined on the physical elements and not on the reference element. In practice, in the case of triangles, parallelograms in 2D, and tetrahedra or parallelepipeds, we could and *should* choose instead

$$\tilde{\mathcal{D}}_k(\mathcal{E}_h) = \{v \in L^2(\Omega) : \forall E \in \mathcal{E}_h,\ v \circ F_E \in \mathbb{P}_k(\hat{E})\}.$$

On such elements, the approximation results for $\mathcal{D}_k(\mathcal{E}_h)$ and $\tilde{\mathcal{D}}_k(\mathcal{E}_h)$ are the same (see Section 2.1.4). However, in the case of general quadrilaterals, the space $\mathbb{P}_k(\hat{E})$ does not have optimal approximation properties (see [2]), whereas the space $\mathbb{P}_k(E)$ has optimal approximation properties (see [58]).

Therefore, for general quadrilateral meshes, we can either choose $\mathbb{P}_k(E)$ and do the computations on the physical elements, or we can choose to increase the discrete space and use the space $\mathbb{Q}_k(\hat{E})$, where $\mathbb{Q}_k$ denotes the space of polynomials of degree less than k in each space direction. The space $\mathbb{Q}_k$ is a tensor product space, and its dimension is strictly greater than the dimension of $\mathbb{P}_k$ for $k \geq 1$. Therefore, the computational costs increase.

2.5.2 Basis functions

Because of the lack of continuity constraints between mesh elements for the test functions, the basis functions of $\mathcal{D}_k(\mathcal{E}_h)$ have a support contained in one element. We write

$$\mathcal{D}_k(\mathcal{E}_h) = \text{span}\{\phi_i^E : 1 \leq i \leq N_{\text{loc}}, E \in \mathcal{E}_h\}$$

with

$$\phi_i^E(\boldsymbol{x}) = \begin{cases} \hat{\phi}_i \circ F_E(\boldsymbol{x}), & \boldsymbol{x} \in E, \\ 0, & \boldsymbol{x} \notin E. \end{cases} \tag{2.34}$$

The local basis functions $(\hat{\phi}_i)_{1 \leq i \leq N_{\text{loc}}}$ are defined on the reference element. We propose to simply use the monomial functions. For instance, in 2D, we have

$$\hat{\phi}_i(\hat{x}, \hat{y}) = \hat{x}^I \hat{y}^J, \quad I + J = i, \quad 0 \leq i \leq k.$$

This yields the local dimension

$$N_{\text{loc}} = \frac{(k+1)(k+2)}{2}.$$

For instance, we have the following:

- Piecewise linears:

$$\hat{\phi}_0(\hat{x}, \hat{y}) = 1, \quad \hat{\phi}_1(\hat{x}, \hat{y}) = \hat{x}, \quad \hat{\phi}_2(\hat{x}, \hat{y}) = \hat{y}.$$

- Piecewise quadratics:

$$\hat{\phi}_0(\hat{x},\hat{y}) = 1, \quad \hat{\phi}_1(\hat{x},\hat{y}) = \hat{x}, \quad \hat{\phi}_2(\hat{x},\hat{y}) = \hat{y},$$
$$\hat{\phi}_3(\hat{x},\hat{y}) = \hat{x}^2, \quad \hat{\phi}_4(\hat{x},\hat{y}) = \hat{x}\hat{y}, \quad \hat{\phi}_5(\hat{x},\hat{y}) = \hat{y}^2.$$

Similarly, in 3D, we define

$$\hat{\phi}_i(\hat{x},\hat{y},\hat{z}) = \hat{x}^I \hat{y}^J \hat{z}^K, \quad I + J + K = i, \quad 0 \le i \le k.$$

This yields the local dimension

$$N_{\text{loc}} = \frac{(k+1)(k+2)(k+3)}{6}.$$

The flexibility of DG methods allows us to easily change basis functions. For instance, we could use Legendre polynomials or some other polynomials satisfying a desired orthogonality property.

2.5.3 Numerical quadrature

One-dimensional case: An integral over a segment is computed by first mapping the physical edge to the segment $(-1, 1)$, which is the reference element in 1D. Then, the integral is approximated by using a numerical quadrature rule on the interval $(-1, 1)$ such as the Gauss quadrature rule (1.11) defined in Section 1.4.2 and in Appendix A.

Two-dimensional case: The integral of a function $\hat{v}$ defined on the reference element $\hat{E}$ can be computed by using a quadrature rule [44]:

$$\int_{\hat{E}} \hat{v} \approx \sum_{j=1}^{Q_{\text{D}}} w_j \hat{v}(s_{x,j}, s_{y,j}).$$

Appendix A contains the sets of weights w_j and nodes $(s_{x,j}, s_{y,j}) \in \hat{E}$ for different values of Q_{D}. For instance, Table 2.1 gives a rule with 6 quadrature points that is exact for polynomials of total degree less than 4. Since DG methods easily allow for high order approximation, it is important to have high order quadrature rules.

Let E be a triangle or a tetrahedron. The mapping $F_E : \hat{E} \to E$ is affine, and we have

$$\int_E v = \int_{\hat{E}} v \circ F_E \det(\boldsymbol{B}_E) = 2|E| \int_{\hat{E}} \hat{v}.$$

This integral is then approximated by

$$\int_E v \approx 2|E| \sum_{j=1}^{Q_{\text{D}}} w_j \hat{v}(s_{x,j}, s_{y,j}).$$

Table 2.1. *Weights and points for quadrature rule on reference triangle.*

w_j	$s_{x,j}$	$s_{y,j}$
0.11169079483901	0.445948490915965	0.445948490915965
0.11169079483901	0.108103018168070	0.445948490915965
0.11169079483901	0.445948490915965	0.108103018168070
0.05497587182766	0.091576213509771	0.091576213509771
0.05497587182766	0.816847572980459	0.091576213509771
0.05497587182766	0.091576213509771	0.816847572980459

If the integrand involves a vector function $\boldsymbol{w}$ and the gradient of v, we have

$$\begin{aligned}\int_E \nabla v \cdot \boldsymbol{w} &= 2|E| \int_{\hat{E}} (\boldsymbol{B}_E^T)^{-1} \hat{\nabla} \hat{v} \cdot \hat{\boldsymbol{w}} \\ &\approx 2|E| \sum_{j=1}^{Q_D} w_j (\boldsymbol{B}_E^T)^{-1} \hat{\nabla} \hat{v}(s_{x,j}, s_{y,j}) \cdot \hat{\boldsymbol{w}}(s_{x,j}, s_{y,j}).\end{aligned}$$

Similarly, if the integrand involves the gradient of both v and w, we have

$$\int_E \nabla v \cdot \nabla w \approx 2|E| \sum_{j=1}^{Q_D} w_j (\boldsymbol{B}_E^T)^{-1} \hat{\nabla} \hat{v}(s_{x,j}, s_{y,j}) \cdot (\boldsymbol{B}_E^T)^{-1} \hat{\nabla} \hat{w}(s_{x,j}, s_{y,j}).$$

2.6 DG scheme

The general DG finite element method is as follows: Find P_h in $\mathcal{D}_k(\mathcal{E}_h)$ such that

$$\forall\, v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(P_h, v) = L(v). \tag{2.35}$$

The same terminology defined for the one-dimensional case (see Section 1.2) applies here.

- If $\epsilon = -1$, the method is called symmetric interior penalty Galerkin (SIPG). We will see that this method converges if the penalty parameter σ_e^0 is large enough.
- If $\epsilon = +1$, the method is called nonsymmetric interior penalty Galerkin (NIPG). We will see that this method converges for any nonnegative values of the penalty parameter σ_e^0. This class of methods also encompasses the case where $\sigma_e^0 = 0$, which has appeared in the literature as the OBB method [84].
- If $\epsilon = 0$, the method is called incomplete interior penalty Galerkin (IIPG). We will see that this method converges under the same condition as for the SIPG; namely the penalty parameter σ_e^0 should be large enough.
- The $J_1^{\sigma_1,\beta_1}$ term is an extra stabilization term. The analysis of the method is independent of this term, and, from now on, we will assume for simplicity that $\sigma_e^1 = 0$ for all e.

2.7 Properties

2.7.1 Coercivity of bilinear forms

Definition 2.10. *A bilinear form a defined on a normed linear space V with norm $\|\cdot\|_V$ is coercive if there is a positive constant κ such that*

$$\forall v \in V, \quad \kappa \|v\|_V^2 \leq a(v, v).$$

For the DG bilinear form, we have

$$\begin{aligned} a_\epsilon(v, v) = &\sum_{E\in\mathcal{E}_h} \int_E \boldsymbol{K}(\nabla v)^2 + \int_\Omega \alpha v^2 \\ &+ (\epsilon - 1) \sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\,[v] + J_0^{\sigma_0,\beta_0}(v, v). \end{aligned}$$

Define the *energy* norm on $\mathcal{D}_k(\mathcal{E}_h)$:

$$\|v\|_{\mathcal{E}} = \left(\sum_{E\in\mathcal{E}_h} \int_E \boldsymbol{K}\nabla v \cdot \nabla v + \int_\Omega \alpha v^2 + J_0^{\sigma_0,\beta_0}(v, v) \right)^{1/2}. \tag{2.36}$$

It is easy to check that it is indeed a norm if $\sigma_0^e > 0$ for all e. We remark that we immediately have the coercivity property satisfied for $\epsilon = 1$. The coercivity constant is $\kappa = 1$. Indeed,

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \|v\|_{\mathcal{E}}^2 = a_\epsilon(v, v).$$

In the case where $\epsilon = -1$ or $\epsilon = 0$, we obtain using Cauchy–Schwarz's inequality an upper bound of the term $\sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\,[v]$:

$$\begin{aligned} \sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}[v] &\leq \sum_{e\in\Gamma_h\cup\Gamma_D} \|\{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\|_{L^2(e)} \|[v]\|_{L^2(e)} \\ &\leq \sum_{e\in\Gamma_h\cup\Gamma_D} \|\{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\|_{L^2(e)} \left(\frac{1}{|e|^{\beta_0}}\right)^{1/2-1/2} \|[v]\|_{L^2(e)}. \end{aligned}$$

Next, we consider the average of the fluxes for an interior edge e shared by the elements E_1^e and E_2^e:

$$\|\{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\|_{L^2(e)} \leq \frac{1}{2}\|(\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e)|_{E_1^e}\|_{L^2(e)} + \frac{1}{2}\|(\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e)|_{E_2^e}\|_{L^2(e)}.$$

Using the property (2.19) of $\boldsymbol{K}$ and the trace inequality (2.6), we have

$$\begin{aligned} \|\{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}\|_{L^2(e)} &\leq \frac{K_1}{2}\|(\nabla v \cdot \boldsymbol{n}_e)|_{E_1^e}\|_{L^2(e)} + \frac{K_1}{2}\|(\nabla v \cdot \boldsymbol{n}_e)|_{E_2^e}\|_{L^2(e)} \\ &\leq \frac{C_t K_1}{2} h_{E_1^e}^{-1/2} \|\nabla v\|_{L^2(E_1^e)} + \frac{C_t K_1}{2} h_{E_2^e}^{-1/2} \|\nabla v\|_{L^2(E_2^e)}. \end{aligned}$$

So we have using (2.22)

$$\begin{aligned}\int_e \{\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\}[v] &\le \frac{C_t K_1}{2}|e|^{\beta_0/2}\Big(h_{E_1^e}^{-1/2}\|\nabla v\|_{L^2(E_1^e)}\\ &\quad + h_{E_2^e}^{-1/2}\|\nabla v\|_{L^2(E_2^e)}\Big)\left(\frac{1}{|e|^{\beta_0}}\right)^{1/2}\|[v]\|_{L^2(e)}\\ &\le \frac{C_t K_1}{2}\left(h_{E_1^e}^{\frac{\beta_0}{2}(d-1)-\frac{1}{2}} + h_{E_2^e}^{\frac{\beta_0}{2}(d-1)-\frac{1}{2}}\right)\Big(\|\nabla v\|^2_{L^2(E_1^e)}\\ &\quad + \|\nabla v\|^2_{L^2(E_2^e)}\Big)^{1/2}\left(\frac{1}{|e|^{\beta_0}}\right)^{1/2}\|[v]\|_{L^2(e)}\\ &\le C_t K_1\Big(\|\nabla v\|^2_{L^2(E_1^e)} + \|\nabla v\|^2_{L^2(E_2^e)}\Big)^{1/2}\left(\frac{1}{|e|^{\beta_0}}\right)^{1/2}\|[v]\|_{L^2(e)}\end{aligned}$$

if β_0 satisfies the condition $\beta_0(d-1)\ge 1$ and if we assume, without loss of generality, that $h\le 1$. A similar bound is obtained if e is a boundary edge. Let n_0 denote the maximum number of neighbors an element can have, i.e., for a conforming mesh, $n_0 = 3$ for a triangle and $n_0 = 4$ for a quadrilateral or tetrahedron:

$$\begin{aligned}\sum_{e\in\Gamma_h\cup\Gamma_D}\int_e \{\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\}[v] &\le C_t K_1\left(\sum_{e\in\Gamma_h\cup\Gamma_D}\frac{1}{|e|^{\beta_0}}\|[v]\|^2_{L^2(e)}\right)^{1/2}\\ &\quad\times\left(\sum_{e\in\Gamma_h}\|\nabla v\|^2_{L^2(E_1^e)} + \|\nabla v\|^2_{L^2(E_2^e)} + \sum_{e\in\Gamma_D}\|\nabla v\|^2_{0,E_1^e}\right)\\ &\le C_t K_1\sqrt{n_0}\left(\sum_{e\in\Gamma_h\cup\Gamma_D}\frac{1}{|e|^{\beta_0}}\|[v]\|^2_{L^2(e)}\right)^{1/2}\left(\sum_{E\in\mathcal{E}_h}\|\nabla v\|^2_{L^2(E)}\right)^{1/2}.\end{aligned}$$

Using Young's inequality, we have for $\delta > 0$

$$\sum_{e\in\Gamma_h\cup\Gamma_D}\int_e \{\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\}[v] \le \frac{\delta}{2}\sum_{E\in\mathcal{E}_h}\|\boldsymbol{K}^{1/2}\nabla v\|^2_{L^2(E)} + \frac{C_t^2K_1^2n_0}{2\delta K_0}\sum_{e\in\Gamma_h\cup\Gamma_D}\frac{1}{|e|^{\beta_0}}\|[v]\|^2_{L^2(e)}.$$

Thus, we obtain a lower bound for $a_\epsilon(v,v)$:

$$a_\epsilon(v,v) \ge \left(1-\frac{\delta}{2}|1-\epsilon|\right)\sum_{E\in\mathcal{E}_h}\|\boldsymbol{K}^{1/2}\nabla v\|^2_{L^2(E)} + \sum_{e\in\Gamma_h\cup\Gamma_D}\frac{\sigma_e^0 - \frac{C_t^2K_1^2n_0}{2\delta K_0}|1-\epsilon|}{|e|^{\beta_0}}\|[v]\|^2_{L^2(e)}.$$

Choosing, for instance, $\delta = 1$ if $\epsilon = 0$ and $\delta = 1/2$ if $\epsilon = -1$ and choosing σ_e^0 large enough (for example, $\sigma_e^0 \ge (C_t^2K_1^2n_0/K_0)$ if $\epsilon = 0$ and $\sigma_e^0 \ge (2C_t^2K_1^2n_0/K_0)$ if $\epsilon = -1$), then we have the coercivity result with $\kappa = 1/2$:

$$a_\epsilon(v,v) \ge \kappa\|v\|^2_{\mathcal{E}}. \tag{2.37}$$

Summarizing the results above, we have

- a_{+1} is coercive;
- a_{-1} and a_0 are coercive if $\beta_0(d-1) \geq 1$ and if σ_e^0 is bounded below by a constant σ_e^* that depends only on K_0, K_1, and the constant in the trace inequality (2.6).

Remark: As expected, the threshold value for the penalty parameter is twice as large for the SIPG method as for the IIPG method. A more precise value of σ_e^* can be obtained if one uses the trace inequalities (2.7)–(2.9) rather than (2.6). For instance, on a triangular mesh, for a given triangle E, if θ^E denotes the smallest angle in E, if K_0^E, K_1^E denote the lower and upper bound of $\boldsymbol{K}$ on E, and if k^E denotes the polynomial degree of the approximation on E, the limiting value of the penalty depends on the local quantities θ^E, K_0^E, K_1^E, and k^E as follows:

$$\begin{aligned}\forall e \in \Gamma_h, \quad \sigma_e^* = &\frac{3(K_1^{E_e^1})^2}{2K_0^{E_e^1}}(k^{E_e^1})(k^{E_e^1}+1)|e|^{\beta_0-1}\cot\theta^{E_e^1} \\ &+\frac{3(K_1^{(E_e^2)})^2}{2K_0^{E_e^2}}(k^{E_e^2})(k^{E_e^2}+1)|e|^{\beta_0-1}\cot\theta^{E_e^2}, \end{aligned} \tag{2.38}$$

$$\forall e \in \Gamma_D, \quad \sigma_e^* = \frac{6(K_1^{E_e^1})^2}{K_0^{E_e^1}}(k^{E_e^1})(k^{E_e^1}+1)\cot\theta^{E_e^1}|e|^{\beta_0-1}. \tag{2.39}$$

Similarly, in the three-dimensional case, with a tetrahedral mesh, the limiting value depends also on local quantities such as the dihedral angle θ^E in the tetrahedron E that yields the smallest value for $\sin\theta$ over all dihedral angles θ of E:

$$\begin{aligned}\forall e \in \Gamma_h, \quad \sigma_e^* = &\frac{3}{2}\frac{(K_1^{E_e^1})^2}{K_0^{E_e^1}}k^{E_e^1}(k^{E_e^1}+2)h\,|e|^{\beta_0-1}\cot\theta_{E_e^1} \\ &+\frac{3}{2}\frac{(K_1^{E_e^2})^2}{K_0^{E_e^2}}k^{E_e^2}(k^{E_e^2}+2)h\,|e|^{\beta_0-1}\cot\theta_{E_e^2}, \end{aligned} \tag{2.40}$$

$$\forall e \in \Gamma_D, \quad \sigma_e^* = 6\frac{(K_1^{E_e^1})^2}{K_0^{E_e^1}}k^{E_e^1}(k^{E_e^1}+2)h\,|e|^{\beta_0-1}\cot\theta_{E_e^1}. \tag{2.41}$$

If $\sigma_e \geq \sigma_e^*$, then the SIPG and IIPG methods are stable and convergent. The proof of these results can be found in [50].

2.7.2 Continuity of bilinear form

Definition 2.11. *A bilinear form a defined on a linear space V equipped with norm $\|\cdot\|_V$ is continuous if there is a positive constant M such that*

$$\forall v, w \in V, \quad a(v,w) \leq M\|v\|_V\|w\|_V.$$

If $\sigma_e^0 > 0$ for all e, then one can show that the bilinear form a_ϵ is continuous on $\mathcal{D}_k(\mathcal{E}_h)$ equipped with the energy norm $\|\cdot\|_\mathcal{E}$:

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(v, w) \leq M\|v\|_\mathcal{E}\|w\|_\mathcal{E}.$$

However, the bilinear form is not continuous in general on the broken space $H^2(\mathcal{E}_h)$ with respect to the energy norm.

2.7.3 Local mass conservation

One interesting property that naturally comes with the primal DG methods is the conservation of mass on each mesh element. Because of the lack of continuity constraints between the elements, we can choose a test function $v \in \mathcal{D}_k(\mathcal{E}_h)$ that is equal to a different constant on each element. If we fix an element E that belongs to the interior of the domain and if we choose v equal to the constant 1 on E and the constant 0 elsewhere, the method (2.35) reduces to

$$\int_E \alpha P_h - \sum_{e\in\partial E}\int_e \{\boldsymbol{K}\nabla P_h \cdot \boldsymbol{n}_e\}[v] + \sum_{e\in\partial E}\frac{\sigma_e^0}{|e|^{\beta_0}}\int_e [P_h][v] = \int_E f.$$

This is equivalent to

$$\int_E (\alpha P_h - f) + \sum_{e\in\partial E}\frac{\sigma_e^0}{|e|^{\beta_0}}\int_e (P_h|_E - P_h|_{\mathcal{N}(e;E)}) = \int_{\partial E}\{\boldsymbol{K}\nabla P_h \cdot \boldsymbol{n}_E\},$$

where $\mathcal{N}(e; E)$ denotes the element in $\mathcal{E}_h$ that is a neighbor of E through the edge e. Thus, we have obtained a balance equation valid on the element E. A similar equation can be derived if the element E shares at least one face with the boundary of the domain. If we assume that the quantity P_h represents a mass density, then the term $\int_E(\alpha P_h - f)$ corresponds to the mass that is created or destroyed inside E and the term $\int_{\partial E}\{\boldsymbol{K}\nabla P_h \cdot \boldsymbol{n}_E\}$ corresponds to the flux of mass passing through the boundary ∂E. The additional term involving the penalty parameter is a pure numerical mass that is zero if the penalty value is zero. In general, this artificial mass can be exactly computed and can be subtracted if needed.

Local mass conservation is important in particular in coupled flow and transport problems arising in porous media. For instance, Darcy flow can be characterized with the elliptic problem (2.16) with $\alpha = 0$, and the flow velocity $\boldsymbol{u} = -\boldsymbol{K}\nabla p$ is approximated by $\mathbf{U}_h = -\boldsymbol{K}\nabla P_h$. Then, the reactive transport of a chemical species of concentration c can be modeled by the following partial differential equation:

$$\frac{\partial c}{\partial t} - \nabla\cdot(D\nabla c - \boldsymbol{u}c) = r(c).$$

In this case, if the penalty is zero, local mass conservation means

$$\int_{\partial E}\{\mathbf{U}_h\}\cdot\boldsymbol{n}_E = \int_E f.$$

If the numerical approximation of the velocity is not locally conservative, the numerical solution of the transport equation becomes unstable after a few time steps. Chapter 4 describes the transport problem in more detail.

2.7.4 Existence and uniqueness of DG solution

Lemma 2.12. *Assume that* (i), (ii), *or* (iii) *holds true:*

(i) *in the NIPG case,* $k \geq 1$ *and either* $\alpha > 0$ *or* $\sigma_e^0 > 0$ *for all* e*;*

(ii) *in the SIPG or IIPG case,* $k \geq 1$ *and* σ_e^0 *is bounded below by a large constant for all* e*;*

(iii) *in the NIPG case,* $k \geq 2$ *and* $\sigma_e^0 = 0$ *for all* e *and* $\alpha = 0$.

Then, the DG solution P_h *exists and is unique.*

Proof. Since (2.35) is a linear problem in finite dimension, existence is equivalent to uniqueness. We assume that there are two solutions P_h^1 and P_h^2. The difference $w_h = P_h^1 - P_h^2$ satisfies

$$a_\epsilon(w_h, w_h) = 0.$$

By the coercivity result (2.37), we have

$$\|w_h\|_{\mathcal{E}} = 0.$$

Clearly in both cases (i) and (ii), this implies that $w_h = 0$ since $\|\cdot\|_{\mathcal{E}}$ is a norm. The case (iii) is not as easy. Indeed, we can conclude only that w_h is piecewise constant on each element $E \in \mathcal{E}_h$. In order to prove that w_h is globally constant in Ω, we need to construct a test function v on a given element E such that the quantity $\int_e \boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e$ is given on one edge (or face) of E and vanishes on the other edges (or faces). If $\boldsymbol{K}$ is constant in each E, one can construct such a test function on a triangle, parallelogram, or tetrahedron (see Lemma C.1). If $\boldsymbol{K}$ is not constant in each E, one needs to assume in addition that h is small enough. □

2.8 Error analysis

In this section, we assume that the exact solution p belongs to $H^s(\mathcal{E}_h)$ for some $s > 3/2$, and we prove that the DG solution converges to the exact solution. We will first derive a priori error estimates in the energy norm.

2.8.1 Error estimates in the energy norm

By the triangle inequality, we have

$$\|p - P_h\|_{\mathcal{E}} \leq \|p - \tilde{p}\|_{\mathcal{E}} + \|P_h - \tilde{p}\|_{\mathcal{E}}$$

for a function $\tilde{p} \in \mathcal{D}_k(\mathcal{E}_h)$ that approximates the exact solution p as in Theorem 2.6. Then, it suffices to bound $\|P_h - \tilde{p}\|_{\mathcal{E}}$. By consistency (see Section 2.4.1), the error satisfies the *orthogonality* equation

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(P_h - p, v) = 0. \tag{2.42}$$

Denoting $\chi = P_h - \tilde{p}$ and adding and subtracting $\tilde{p}$ in each term yields

$$a_\epsilon(\chi, v) = a_\epsilon(p - \tilde{p}, v).$$

Choosing the test function $v = \chi$ and using the coercivity result (2.37) gives

$$\kappa\|\chi\|_{\mathcal{E}}^2 \le \Big| \sum_{E\in\mathcal{E}_h}\int_E (\boldsymbol{K}\nabla(p-\tilde{p})\nabla\chi + \alpha(p-\tilde{p})\chi) - \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\int_e \{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}[\chi] + \epsilon\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\int_e \{\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\}[p-\tilde{p}] + J_0^{\sigma_0,\beta_0}(p-\tilde{p},\chi)\Big| \le |T_1+\cdots+T_4|.$$

Using the bound (2.19), Cauchy–Schwarz's inequality, and Young's inequality, we have

$$|T_1| \le K_1^{1/2}\left(\sum_E \|\boldsymbol{K}^{1/2}\nabla\chi\|_{L^2(E)}^2\right)^{1/2}\left(\sum_E \|\nabla(p-\tilde{p})\|_{L^2(E)}^2\right)^{1/2} + \|\alpha\|_{L^\infty(\Omega)}^{1/2}\|\alpha^{\frac{1}{2}}\chi\|_{L^2(\Omega)}\|p-\tilde{p}\|_{L^2(\Omega)} \le \frac{3}{2\kappa}(K_1+\|\alpha\|_{L^\infty(\Omega)})|||p-\tilde{p}|||_{H^1(\mathcal{E}_h)}^2 + \frac{\kappa}{6}\|\chi\|_{\mathcal{E}}^2.$$

Let C denote a generic constant independent of h that takes different values at different places. From the approximation result (2.10), we obtain

$$T_1 \le Ch^{2\min(k+1,s)-2}|||p|||_{H^s(\mathcal{E}_h)}^2 + \frac{\kappa}{6}\|\chi\|_{\mathcal{E}}^2.$$

Let us now bound T_3: this term disappears if the method is IIPG ($\epsilon = 0$) or if $\tilde{p}$ is chosen to be a continuous interpolant (such as the classical Lagrange interpolant) and either $|\Gamma_{\mathrm{D}}| = 0$ or g_{D} is a polynomial of degree k (hence, one can choose $\tilde{p} = g_{\mathrm{D}}$ on Γ_{D}). However, in the general case (for example, if $\tilde{p}$ is not continuous), we can still control this term by using trace inequalities and approximation results. First, we have by Cauchy–Schwarz's inequality

$$|T_3| \le \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}} \|\{\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\|[p-\tilde{p}]\|_{L^2(e)}.$$

Now if the edge (or face) is interior, $e = \partial E_e^1 \cap \partial E_e^2$, we can apply the trace inequality (2.1) for each neighboring element:

$$\|[p-\tilde{p}]\|_{L^2(e)} \le \|(p-\tilde{p})|_{E_e^1}\|_{L^2(e)} + \|(p-\tilde{p})|_{E_e^2}\|_{L^2(e)} \le C|e|^{1/2}|E_e^1|^{-1/2}(\|p-\tilde{p}\|_{L^2(E_e^1)} + h_{E_e^1}\|\nabla(p-\tilde{p})\|_{L^2(E_e^1)}) + C|e|^{1/2}|E_e^2|^{-1/2}(\|p-\tilde{p}\|_{L^2(E_e^2)} + h_{E_e^2}\|\nabla(p-\tilde{p})\|_{L^2(E_e^2)}).$$

Using the trace inequality (2.5) in finite-dimensional spaces, we have

$$\|\{\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\}\|_{L^2(e)} \le \frac{1}{2}\|(\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e)|_{E_e^1}\|_{L^2(e)} + \frac{1}{2}\|(\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e)|_{E_e^2}\|_{L^2(e)} \le \frac{K_1}{2}\tilde{C}_t|e|^{1/2}|E_e^1|^{-1/2}||\nabla\chi\|_{L^2(E_e^1)} + \frac{K_1}{2}\tilde{C}_t|e|^{1/2}|E_e^2|^{-1/2}||\nabla\chi\|_{L^2(E_e^2)}.$$

Combining the two bounds above, we obtain

$$\begin{aligned}
&\forall e \in \Gamma_h, \quad \|\{\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\|[p-\tilde{p}]\|_{L^2(e)}\\
&\le C|e||E_e^1|^{-1}(\|p-\tilde{p}\|_{L^2(E_e^1)}+h_{E_e^1}\|\nabla(p-\tilde{p})\|_{L^2(E_e^1)})||\nabla\chi||_{L^2(E_e^1)}\\
&+C(|e||E_e^2|^{-1/2})|E_e^1|^{-1/2}(\|p-\tilde{p}\|_{L^2(E_e^1)}+h_{E_e^1}\|\nabla(p-\tilde{p})\|_{L^2(E_e^1)})||\nabla\chi||_{L^2(E_e^2)}\\
&+C|e||E_e^2|^{-1}(\|p-\tilde{p}\|_{L^2(E_e^2)}+h_{E_e^2}\|\nabla(p-\tilde{p})\|_{L^2(E_e^2)})||\nabla\chi||_{L^2(E_e^2)}\\
&+C(|e||E_e^1|^{-1/2})|E_e^2|^{-1/2}(\|p-\tilde{p}\|_{L^2(E_e^2)}+h_{E_e^2}\|\nabla(p-\tilde{p})\|_{L^2(E_e^2)})||\nabla\chi||_{L^2(E_e^1)}.
\end{aligned}$$

Using the approximation results (2.10) and the fact that for $i=1,2$, the product $|e||E_e^i|^{-1/2}$ is bounded by a constant C in 2D and bounded by $Ch_{E_e^i}^{1/2}$ in 3D, we have

$$\begin{aligned}
&\forall e \in \Gamma_h, \quad \|\{\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\|[p-\tilde{p}]\|_{L^2(e)}\\
&\le Ch^{\min(k+1,s)-1}(|p|_{H^s(E_e^1)}+|p|_{H^s(E_e^2)})(\|\nabla\chi\|_{L^2(E_e^1)}+\|\nabla\chi\|_{L^2(E_e^2)}).
\end{aligned}$$

Assume now that the edge (or face) e is on the Dirichlet boundary Γ_{D} and belongs to the element E_e^1. Following a similar argument as above, we have

$$\|\boldsymbol{K}\nabla\chi\cdot\boldsymbol{n}_e\|_{L^2(e)}\|p-\tilde{p}\|_{L^2(e)} \le Ch^{\min(k+1,s)-1}|p|_{H^s(E_e^1)}\|\nabla\chi\|_{L^2(E_e^1)}.$$

Therefore, the term T_3 is bounded by

$$T_3 \le Ch^{2\min(k+1,s)-2}|\!|\!|p|\!|\!|^2_{H^s(\mathcal{E}_h)}+\frac{\kappa}{6}\|\chi\|^2_{\mathcal{E}}.$$

The term T_4 is zero if $\sigma_e^0=0$ for all e or if $\tilde{p}$ is continuous and either $|\Gamma_{\mathrm{D}}|=0$ or g_{D} is a continuous piecewise polynomial of degree k. Otherwise, using the fact that $|e|\le h^{d-1}$, the term T_4 is simply bounded using Cauchy–Schwarz's and Young's inequalities:

$$\begin{aligned}
|T_4| &\le \frac{3}{2\kappa}J_0^{\sigma_0,\beta_0}(p-\tilde{p},p-\tilde{p})+\frac{\kappa}{6}J_0^{\sigma_0,\beta_0}(\chi,\chi)\\
&\le \frac{\kappa}{6}\|\chi\|^2_{\mathcal{E}}+Ch^{2\min(k+1,s)-1-\beta_0(d-1)}|\!|\!|p|\!|\!|^2_{H^s(\mathcal{E}_h)}.
\end{aligned}$$

Thus, T_4 is optimal if the condition $\beta_0(d-1)\le 1$ is satisfied. Under the assumptions given above, we obtain

$$\frac{\kappa}{2}\|\chi\|^2_{\mathcal{E}} \le Ch^{2\min(k+1,s)-2}|\!|\!|p|\!|\!|^2_{H^s(\mathcal{E}_h)}+|T_2|.$$

In order to conclude, it remains to bound the term T_2. On one hand, this term is relatively easy to bound if all penalty values are nonzero. On the other hand, if some penalty values are zero, the bound of T_2 requires an additional property on the approximation $\tilde{p}$ and a restriction of the polynomial degree $k\ge 2$. Thus, we distinguish two cases. First, let us assume that $\sigma_e^0>0$ for all e; then we can write

$$\left|\int_e\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}[\chi]\right| \le \left(\frac{|e|^{\beta_0}}{\sigma_e^0}\right)^{\frac{1}{2}}\|\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\left(\frac{\sigma_e^0}{|e|^{\beta_0}}\right)^{\frac{1}{2}}\|[\chi]\|_{L^2(e)},$$

$$\left|\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}[\chi]\right| \le \frac{\kappa}{6}J_0^{\sigma_0,\beta_0}(\chi,\chi)+C\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\frac{|e|^{\beta_0}}{\sigma_e^0}\|\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}\|^2_{L^2(e)}.$$

Using the trace inequality (2.2) and the approximation result (2.10), we have

$$|T_2| \leq \frac{\kappa}{6}\|\chi\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-3+\beta_0(d-1)}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}^2.$$

Thus, if $\beta_0(d-1) \geq 1$, we have

$$|T_2| \leq \frac{\kappa}{6}\|\chi\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-2}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}^2, \tag{2.43}$$

and the final error estimate is

$$\frac{\kappa}{3}\|\chi\|_{\mathcal{E}}^2 \leq Ch^{2\min(k+1,s)-2}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}^2. \tag{2.44}$$

In the second case, let us assume that $\sigma_e^0 = 0$ for some e. Then, for each element E, we use the approximation $\tilde{p} \in \mathbb{P}_k(E)$ defined in Theorem 2.7. Since this approximation is defined locally on each E, we have

$$\forall E \in \mathcal{E}_h, \quad \forall e \in \partial E, \int_e \{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\} = 0.$$

We then rewrite the term T_2 for any real number c_e:

$$\begin{aligned} T_2 &= \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}} \int_e \{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}([\chi]-c_e) \\ &\leq \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}} \|\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\|[\chi]-c_e\|_{L^2(e)}. \end{aligned}$$

If e is an interior edge (or face) and is shared by E_e^1 and E_e^2, we choose

$$c_e = c_1 - c_2, \quad c_i = \frac{1}{|E_e^i|}\int_{E_e^i}\chi, \quad i = 1,2,$$

and we observe that

$$[\chi] - c_e = \chi|_{E_e^1} - \chi|_{E_e^2} - (c_1 - c_2) = (\chi|_{E_e^1} - c_1) - (\chi|_{E_e^2} - c_2).$$

Thus, we have by the trace inequality (2.1):

$$\begin{aligned} \|[\chi]-c_e\|_{L^2(e)} &\leq \|\chi|_{E_1} - c_1\|_{L^2(e)} + \|\chi|_{E_2} - c_2\|_{L^2(e)} \\ &\leq Ch_{E_e^1}^{-1/2}(\|\chi - c_1\|_{L^2(E_e^1)} + h_{E_e^1}\|\nabla\chi\|_{L^2(E_e^1)}) \\ &\quad + Ch_{E_e^2}^{-1/2}(\|\chi - c_2\|_{L^2(E_e^2)} + h_{E_e^2}^{1/2}\|\nabla\chi\|_{L^2(E_e^2)}). \end{aligned}$$

Next, by definition of the constant c_i, we have

$$\int_{E_e^i}(\chi|_{E_e^i} - c_i) = 0, \quad i = 1,2.$$

Thus, we have

$$\|[\chi] - c_e\|_{L^2(e)} \le Ch_{E_e^1}^{1/2}\|\nabla\chi\|_{L^2(E_e^1)} + Ch_{E_e^1}^{1/2}\|\nabla\chi\|_{L^2(E_e^2)}.$$

Indeed, we have used the following result: If a function ϕ belongs to $H^1(E)$ such that $\int_E \phi = 0$, then there is a constant C independent of h_E such that

$$\|\phi\|_{L^2(E)} \le Ch_E\|\nabla\phi\|_{L^2(E)}.$$

Note that if the face e belongs to the boundary $\Gamma_h \cap \partial E_e^1$, then we choose $c_e = \frac{1}{|E_e^1|}\int_{E_e^1}\chi$, and we obtain similarly

$$\|\chi - c_e\|_{L^2(e)} \le Ch_{E_e^1}^{1/2}\|\nabla\chi\|_{L^2(E_e^1)}.$$

The other factor in the term T_2 is bounded using trace inequality (2.2) and approximation result (2.10):

$$\forall e \in \Gamma_h, \quad \|\{\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\}\|_{L^2(e)} \le Ch^{\min(k+1,s)-3/2}(|p|_{H^s(E_e^1)} + |p|_{H^s(E_e^2)}),$$
$$\forall e \in \Gamma_D, \quad \|\boldsymbol{K}\nabla(p-\tilde{p})\cdot\boldsymbol{n}_e\|_{L^2(e)} \le Ch^{\min(k+1,s)-3/2}|p|_{H^s(E_e^1)}.$$

Combining the bounds above gives an inequality identical to (2.43), and thus the bound (2.44) is obtained. We saw that the derivation of the error estimates requires a constraint on the power β_0, under a certain condition. Before summarizing the results, we state that condition.

Condition A: The approximation $\tilde{p}$ of the exact solution p can be chosen to be continuous. In addition, either the Dirichlet data g_D is a continuous piecewise polynomial of degree k, or the whole boundary is a Neumann boundary ($\partial\Omega = \Gamma_N$).

Theorem 2.13. *Assume that the exact solution to* (2.16)–(2.18) *belongs to* $H^s(\mathcal{E}_h)$ *for* $s > 3/2$. *Assume also that the penalty parameter* σ_e^0 *is large enough for the SIPG and IIPG methods and that* $k \ge 2$ *for the NIPG method with zero penalty. Then, there is a constant* C *independent of* h *such that the following optimal a priori error estimate holds:*

$$\|p - P_h\|_{\mathcal{E}} \le Ch^{\min(k+1,s)-1}|||p|||_{H^s(\mathcal{E}_h)}.$$

This estimate is valid if Condition A holds true and if $\beta_0 \ge (d-1)^{-1}$. *Otherwise, if Condition A fails, this estimate is valid if* $\beta_0 = (d-1)^{-1}$.

2.8.2 Error estimates in the L^2 norm

Next, we prove an error estimate in the L^2 norm. We will apply the Aubin–Nitsche lift technique used in the analysis of the classical finite element method to the DG method. This technique works well if the scheme is symmetric. This is the case for the SIPG method. We will see below that optimal estimates cannot be derived for IIPG and NIPG. For simplicity, we assume that the entire boundary is a Dirichlet boundary, i.e., $\partial\Omega = \Gamma_D$. We assume that the domain is convex and that the solution to the dual problem

$$-\nabla\cdot(\boldsymbol{K}\nabla\phi) + \alpha\phi = p - P_h \quad \text{in} \quad \Omega,$$
$$\phi = 0 \quad \text{on} \quad \partial\Omega$$

belongs to $H^2(\Omega)$ with continuous dependence on $p - P_h$:

$$\|\phi\|_{H^2(\Omega)} \leq C\|p - P_h\|_{L^2(\Omega)}. \tag{2.45}$$

Then, we have

$$\|P_h - p\|^2_{L^2(\Omega)} = \int_\Omega (P_h - p)^2 = \int_\Omega (-\nabla \cdot (\boldsymbol{K}\nabla\phi) + \alpha\phi)(P_h - p).$$

Denoting $\theta = P_h - p$ and integrating by parts on each element yields

$$\|\theta\|^2_{L^2(\Omega)} = \sum_{E\in\mathcal{E}_h} \int_E (\boldsymbol{K}\nabla\phi \cdot \nabla\theta + \alpha\phi\theta) - \sum_{E\in\mathcal{E}_h} \int_{\partial E} (\boldsymbol{K}\nabla\phi \cdot \boldsymbol{n}_E)\theta.$$

The last term can be rewritten as in (2.25). Since $\phi \in H^2(\Omega)$, we have

$$\|\theta\|^2_{L^2(\Omega)} = \sum_{E\in\mathcal{E}_h} \int_E (\boldsymbol{K}\nabla\phi\nabla\theta + \alpha\phi\theta) - \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\boldsymbol{K}\nabla\phi \cdot \boldsymbol{n}_e\}[\theta].$$

We now subtract the orthogonality equation (2.42) from the equation above:

$$\begin{aligned}
\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \|\theta\|^2_{L^2(\Omega)} &= \sum_{E\in\mathcal{E}_h} \int_E (\boldsymbol{K}\nabla(\phi - v)\nabla\theta + \alpha(\phi - v)\theta) \\
&\quad -\epsilon \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}[\theta] - \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\boldsymbol{K}\nabla\phi \cdot \boldsymbol{n}_e\}[\theta] \\
&\quad + \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\boldsymbol{K}\nabla\theta \cdot \boldsymbol{n}_e\}[v] - J_0^{\sigma_0,\beta_0}(\theta, v) \\
&= A_1 + \cdots + A_5.
\end{aligned} \tag{2.46}$$

We choose $v = \tilde{\phi}$, a continuous interpolant of ϕ of degree k. We assume that such an interpolant exists. In that case, we note that $\tilde{\phi} = \phi = 0$ on the boundary $\partial\Omega$. The last two terms on the right-hand side of (2.46), namely A_4 and A_5, vanish. The first term is easily bounded using Cauchy–Schwarz's inequality and the approximation result (2.10):

$$A_1 \leq Ch\|\phi\|_{H^2(\Omega)}\|\theta\|_{\mathcal{E}}.$$

Therefore, we obtain

$$\|\theta\|^2_{L^2(\Omega)} \leq Ch\|\phi\|_{H^2(\Omega)}\|\theta\|_{\mathcal{E}} + S,$$

where

$$S = \left| \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\boldsymbol{K}\nabla(\phi + \epsilon\tilde{\phi}) \cdot \boldsymbol{n}_e\}[\theta] \right|.$$

If the method employed is the SIPG method, the term $(\phi + \epsilon\tilde{\phi}) = (\phi - \tilde{\phi})$ is the approximation error. A bound of S can be derived by taking advantage of the penalty parameter:

$$\begin{aligned} S &\leq \sum_{e\in\Gamma_h\cup\Gamma_D} \left(\frac{|e|^{\beta_0}}{\sigma_e^0}\right)^{\frac{1}{2}-\frac{1}{2}} \|\{\boldsymbol{K}\nabla(\phi-\tilde{\phi})\cdot\boldsymbol{n}_e\}\|_{L^2(e)}\|[\theta]\|_{L^2(e)} \\ &\leq J_0^{\sigma_0,\beta_0}(\theta,\theta)^{\frac{1}{2}} \left(\sum_{e\in\Gamma_h\cup\Gamma_D} \frac{|e|^{\beta_0}}{\sigma_e^0}\|\{\boldsymbol{K}\nabla(\phi-\tilde{\phi})\cdot\boldsymbol{n}_e\}\|^2_{L^2(e)}\right)^{\frac{1}{2}} \\ &\leq Ch^{\frac{\beta_0}{2}(d-1)+\frac{1}{2}}\|\phi\|_{H^2(\Omega)}\|\theta\|_{\mathcal{E}}. \end{aligned}$$

Therefore, using the bound (2.45), we obtain

$$\|\theta\|^2_{L^2(\Omega)} \leq C(h + h^{\frac{\beta_0}{2}(d-1)+\frac{1}{2}})\|\theta\|_{L^2(\Omega)}\|\theta\|_{\mathcal{E}}.$$

With Theorem 2.13 and under the condition $\beta_0(d-1) \geq 1$, this implies

$$\|\theta\|_{L^2(\Omega)} \leq Ch^{\min(k+1,s)}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}. \tag{2.47}$$

If the method employed is the IIPG method or the NIPG method with nonzero penalty, one can recover an additional power of h with the term S if a stricter constraint is imposed on the parameter β_0. Indeed, using Cauchy–Schwarz's inequality and the trace inequality (2.2), we obtain if $\epsilon = 0$ or $\epsilon = 1$

$$\begin{aligned} S &\leq 2J_0^{\sigma_0,\beta_0}(\theta,\theta)^{\frac{1}{2}} \left(\sum_{e\in\Gamma_h\cup\partial\Omega} \frac{|e|^{\beta_0}}{\sigma_e^0}\|\{\boldsymbol{K}\nabla\phi\cdot\boldsymbol{n}_e\}\|^2_{L^2(e)}\right)^{\frac{1}{2}} \\ &\leq Ch^{\frac{\beta_0}{2}(d-1)-\frac{1}{2}}\|\phi\|_{H^2(\Omega)}\|\theta\|_{\mathcal{E}}. \end{aligned}$$

Therefore, under the assumptions of Theorem 2.13 and if $\beta_0(d-1) \geq 3$, we obtain the optimal error estimate (2.47). One can check that a similar result holds true in the NIPG case with nonzero penalty. We say that the DG method is *superpenalized* if $\beta_0 > (d-1)^{-1}$.

The only case that we did not consider is the case of the NIPG method with $\sigma_e^0 = 0$ for all e. One can prove a suboptimal error estimate, namely

$$\|\theta\|_{L^2(\Omega)} = \mathcal{O}(h^{\min(k+1,s)-1}).$$

We summarize the results below.

Theorem 2.14. *Assume that Theorem* 2.13 *holds. There is a constant C independent of h such that*

$$\|p - P_h\|_{L^2(\Omega)} \leq Ch^{\min(k+1,s)}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}.$$

This estimate is valid for the SIPG method unconditionally and for the NIPG and IIPG methods under Condition A and the superpenalization $\beta_0 \geq 3(d-1)^{-1}$. If Condition A is not satisfied, then the numerical error for both the NIPG and IIPG methods satisfies the following suboptimal error estimate:

$$\|p - P_h\|_{L^2(\Omega)} \leq Ch^{\min(k+1,s)-1}|\!|\!|p|\!|\!|_{H^s(\mathcal{E}_h)}.$$

Remark: In the standard penalization $\beta_0 = (d-1)^{-1}$, we know how to prove suboptimal error estimates for both the IIPG and NIPG methods. It has been observed numerically on uniform meshes that convergence rates are optimal if the polynomial degree is odd and suboptimal if the polynomial degree is even (see Section 2.10). This is an interesting question that remains to be theoretically solved. For general meshes one can construct an example for which the numerical rates are suboptimal even if the polynomial degree is odd.

Remark: In this section, we have considered convergence of the h-version of the DG method. The polynomial degree is kept fixed, and the mesh is successively refined. In the hp-version, both mesh size and polynomial degrees can be changed and error estimates can be derived (see, for instance, [96, 72]). They are suboptimal with respect to the polynomial degree.

Remark: In the case of meshes containing quadrilaterals in 2D and hexahedra in 3D, Condition A can be satisfied if we use the space of piecewise polynomials of degree k in each direction given by

$$\tilde{\mathcal{D}}_k(\mathcal{E}_h) = \{v \in L^2(\Omega) : \forall E \in \mathcal{E}_h, v|_E \in \mathbb{Q}_k(E)\}.$$

Indeed, one can construct a continuous interpolant of p in the space $\tilde{\mathcal{D}}_k(\mathcal{E}_h)$. The benefit of using $\tilde{\mathcal{D}}_k(\mathcal{E}_h)$ rather than $\mathcal{D}_k(\mathcal{E}_h)$ is that optimal L^2 error estimates are obtained if superpenalization is used. The drawback is that the method is more expensive as the total number of degrees of freedom increases. Therefore, if one does not want superpenalization (it is known that increasing the power β_0 worsens the condition number of the global matrix), then one should use $\mathcal{D}_k(\mathcal{E}_h)$ for all meshes.

2.9 Implementing the DG method

There is more than one way to write a DG code. Our preferred choice is to use a parent-child data structure. This allows for an easy implementation of local mesh refinement and derefinement. In this section, we first present the data structure and then discuss the construction of the local and global matrices. For simplicity, we will assume that $\boldsymbol{K}$ is piecewise constant.

2.9.1 Data structure

A parent-child data structure uses a list of elements, faces, and vertices. It is understood that for two-dimensional problems, an edge is called a face. We assume that a given element has M_F faces and that each interior face belongs to two elements. If an element is refined, it has at most M_C children. We also denote by M_V the number of vertices of one face. Attributes of the elements and faces are given in Table 2.2. Those attributes contain the information that is being stored. One can choose to either store more information or recompute information when needed. There is a delicate balance between the amount of storage and the amount of computation that will yield a minimum simulation time. In the programming language C, we can take advantage of the structure data type to store the attributes. For instance, for a triangular mesh, we give below the definition of the structures `element`, `face`, and `vertex` and arrays of those particular data types.

Table 2.2. *Attributes of elements and faces for the data structure.*

Object	Attributes	Definition
element	face	array of M_F components: global number of faces
	parent	integer: global number of parents
	child	array of M_C components: global number of children
	degree	integer: polynomial degree
	reftype	integer: -1 for inactive (not refined) element 0 for active (refined) element
	soldofs	array of N_{loc} components: local degrees of freedom
face	vertex	array of M_V components: global number of vertices
	neighbor	array of 2 components: global number of elements sharing the face
	reftype	integer: -1 for inactive (not refined) face 0 for active (refined) face
	bctype	integer: 0 for interior face 1 for Dirichlet face 2 for Neumann face
vertex	coor	array: coordinates of the vertex

```
typedef struct {
    int    face[3];
    int    parent;
    int    child[4];
    int    degree;
    int    reftype;
    double *soldofs;
} element;

typedef struct {
    int    vertex[2];
    int    neighbor[2];
    int    reftype;
    int    bctype;
} face;

typedef struct {
    double coor[2];
} vertex;

element meshelt[100];
face meshface[300];
vertex meshvertex[300];
```

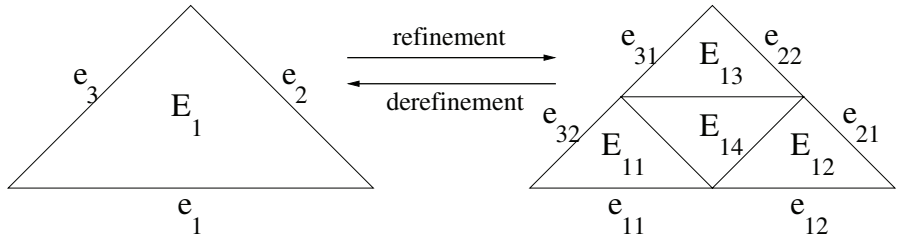


Figure 2.4. *Example of a refinement/derefinement strategy for a triangular element.*

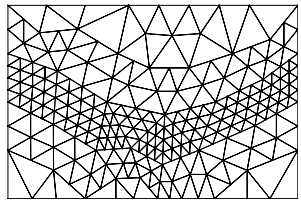


Figure 2.5. *Example of a nonconforming mesh.*

If the mesh uses quadrilateral elements, it suffices to increase the size of the attribute `face` to four entries. Fig. 2.4 shows an example of refinement/derefinement of a triangle. In this case, the element has $M_F = 3$ faces and $M_C = 4$ children, and each face has $M_V = 2$ vertices. The children of the element E_1 are the elements E_{11}, E_{12}, E_{13}, and E_{14}. New faces corresponding to the refinement of the faces e_1, e_2, e_3 are created. For example, the children of face e_1 are the faces e_{11} and e_{12}. Once the element and faces are refined, they become "inactive." The inverse process, also called derefinement, changes the "inactive" state of the parents to "active" and vice versa for the children. Note that the parents of the elements in the coarsest mesh do not exist and by default can be set to zero. With the DG method, it is possible to refine a few elements in the mesh as many times as possible without worrying about refining their neighbors. The resulting mesh is called nonconforming. An example of a nonconforming mesh is given in Fig. 2.5.

2.9.2 Local matrices and right-hand sides

There are two types of local matrices depending on the domain of integration. First, we compute the matrix $\boldsymbol{A}_E$ resulting from the volume integral over a fixed element E. We recall (see Section 2.5.2) that the local basis functions $\phi_{i,E}$ are obtained from mapping monomial functions $\hat{\phi}_i$ from the reference element $\hat{E}$ onto the element E:

$$\phi_{i,E} = \hat{\phi}_i \circ F_E^{-1}.$$

Then, we have

$$\forall 1 \le i, j \le N_{\text{loc}}, \quad (A_E)_{i,j} = \int_E (\boldsymbol{K}\nabla\phi_{i,E} \cdot \nabla\phi_{j,E} + \alpha\phi_{i,E}\phi_{j,E}).$$

Applying a change of variable with the mapping F_E (see (2.30) and (2.32)), we can compute the integral on the reference element:

$$(A_E)_{i,j} = 2|E| \int_{\hat{E}} (\boldsymbol{K}(\boldsymbol{B}_E^T)^{-1}\hat{\nabla}\hat{\phi}_i \cdot (\boldsymbol{B}_E^T)^{-1}\hat{\nabla}\hat{\phi}_j + (\alpha \circ F_E)\hat{\phi}_i\hat{\phi}_j).$$

The volume contributions to the local right-hand side $\boldsymbol{b}_E$ are

$$(\boldsymbol{b}_E)_i = \int_E f\phi_{i,E}.$$

We now compute the local matrices corresponding to the integrals over a fixed face e. If e is an interior face, let us denote by E_e^1 and E_e^2 the elements that share the face such that the normal vector $\boldsymbol{n}_e$ points from E_e^1 to E_e^2. The terms involving integrals on e in the bilinear form a_ϵ are recalled below:

$$T = -\int_e \{\boldsymbol{K}\nabla P_h \cdot \boldsymbol{n}_e\}[v] + \epsilon \int_e \{\boldsymbol{K}\nabla v \cdot \boldsymbol{n}_e\}[P_h] + \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e [P_h][v].$$

Denoting by $P_{h,i}$ and v_i the restrictions of P_h and v to the element E_i and expanding the averages and jumps, we obtain

$$T = m_e^{11} + m_e^{22} + m_e^{12} + m_e^{21},$$

where the term m_e^{11} (resp., m_e^{22}) corresponds to the interactions of the local basis of the neighboring element E_e^1 (resp., E_e^2) with itself and the term m_e^{12} (resp., m_e^{21}) corresponds to the interactions of the local basis of the neighboring element E_e^1 (resp., E_e^2) with the element E_e^2 (resp., E_e^1). More precisely, we have the expressions

$$\begin{aligned}
m_e^{11} &= -\frac{1}{2}\int_e \boldsymbol{K}\nabla P_{h,1} \cdot \boldsymbol{n}_e v_1 + \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla v_1 \cdot \boldsymbol{n}_e P_{h,1} + \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e P_{h,1}v_1,\\
m_e^{22} &= \frac{1}{2}\int_e \boldsymbol{K}\nabla P_{h,2} \cdot \boldsymbol{n}_e v_2 - \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla v_2 \cdot \boldsymbol{n}_e P_{h,2} + \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e P_{h,2}v_2,\\
m_e^{12} &= -\frac{1}{2}\int_e \boldsymbol{K}\nabla P_{h,2} \cdot \boldsymbol{n}_e v_1 - \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla v_1 \cdot \boldsymbol{n}_e P_{h,2} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e P_{h,2}v_1,\\
m_e^{21} &= -\frac{1}{2}\int_e \boldsymbol{K}\nabla P_{h,1} \cdot \boldsymbol{n}_e v_2 + \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla v_2 \cdot \boldsymbol{n}_e P_{h,1} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e P_{h,1}v_2.
\end{aligned}$$

These four terms will yield four matrices of size $N_{\text{loc}} \times N_{\text{loc}}$, namely $\boldsymbol{M}_e^{11}$, $\boldsymbol{M}_e^{22}$, $\boldsymbol{M}_e^{12}$, $\boldsymbol{M}_e^{21}$, whose entries are defined below:

$$\begin{aligned}
(\boldsymbol{M}_e^{11})_{ij} &= -\frac{1}{2}\int_e \boldsymbol{K}\nabla\phi_{j,E_e^1} \cdot \boldsymbol{n}_e\phi_{i,E_e^1} + \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla\phi_{i,E_e^1} \cdot \boldsymbol{n}_e\phi_{j,E_e^1} + \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^1}\phi_{i,E_e^1},\\
(\boldsymbol{M}_e^{22})_{ij} &= \frac{1}{2}\int_e \boldsymbol{K}\nabla\phi_{j,E_e^2} \cdot \boldsymbol{n}_e\phi_{i,E_e^2} - \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla\phi_{i,E_e^2} \cdot \boldsymbol{n}_e\phi_{j,E_e^2} + \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^2}\phi_{i,E_e^2},
\end{aligned}$$

$$(\boldsymbol{M}_e^{12})_{ij} = -\frac{1}{2}\int_e \boldsymbol{K}\nabla\phi_{j,E_e^2}\cdot\boldsymbol{n}_e\phi_{i,E_e^1} - \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla\phi_{i,E_e^1}\cdot\boldsymbol{n}_e\phi_{j,E_e^2} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^2}\phi_{i,E_e^1},$$
$$(\boldsymbol{M}_e^{21})_{ij} = -\frac{1}{2}\int_e \boldsymbol{K}\nabla\phi_{j,E_e^1}\cdot\boldsymbol{n}_e\phi_{i,E_e^2} + \frac{\epsilon}{2}\int_e \boldsymbol{K}\nabla\phi_{i,E_e^2}\cdot\boldsymbol{n}_e\phi_{j,E_e^1} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^1}\phi_{i,E_e^2}.$$

Next, if e is a boundary face, let us also denote by E_e^1 the element to which it belongs. If a Dirichlet boundary condition is applied on e, the following local matrix $\boldsymbol{M}_e^{11}$ is created:

$$(\boldsymbol{M}_e^{11})_{ij} = -\int_e \boldsymbol{K}\nabla\phi_{j,E_e^1}\cdot\boldsymbol{n}_e\phi_{i,E_e^1} + \epsilon\int_e \boldsymbol{K}\nabla\phi_{i,E_e^1}\cdot\boldsymbol{n}_e\phi_{j,E_e^1} + \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^1}\phi_{i,E_e^1},$$

and the local right-hand side $\boldsymbol{b}_e$ is

$$(\boldsymbol{b}_e)_i = \epsilon\int_e \left(\boldsymbol{K}\nabla\phi_{i,E_e^1}\cdot\boldsymbol{n}_e + \frac{\sigma_e^0}{|e|^{\beta_0}}\phi_{i,E_e^1}\right) g_{\mathrm{D}}.$$

If the edge e is a Neumann boundary edge, no local matrix is created, but the following local right-hand side is defined:

$$(\boldsymbol{b}_e)_i = \int_e \phi_{i,E_e^1} g_{\mathrm{N}}.$$

As usual, all integrals on the physical face are transformed into integrals on the reference element in the space $\mathbb{R}^{d-1}$.

We now present the algorithm for computing the local matrices and the local right-hand sides.

ALGORITHM 2.1.
Computing local contributions from element E
initialize $\boldsymbol{A}_E = \boldsymbol{0}$
initialize the quadrature weights $\boldsymbol{w}$ and points $\boldsymbol{s}$
loop over quadrature points: for $k = 1$ to N_{G} do
  compute Jacobian matrix $\boldsymbol{B}_E$
  for $i = 1$ to N_{loc} do
    compute values of basis function $\phi_{i,E}(\boldsymbol{s}(k))$
    compute derivatives of basis functions $\nabla\phi_{i,E}(\boldsymbol{s}(k))$
  end
  compute global coordinates $\boldsymbol{x}$ of quadrature point $\boldsymbol{s}(k)$
  compute source function $f(\boldsymbol{x})$
  for $i = 1$ to N_{loc} do
    for $j = 1$ to N_{loc} do
      $\boldsymbol{A}_E(i,j) = \boldsymbol{A}_E(i,j) + \boldsymbol{w}(k)\det(\boldsymbol{B}_E)\nabla\phi_{i,E}(\boldsymbol{s}(k))\cdot\nabla\phi_{j,E}(\boldsymbol{s}(k))$
    end
    $\boldsymbol{b}_E(i) = \boldsymbol{b}_E(i) + \boldsymbol{w}(k)\det(\boldsymbol{b}_E)f(\boldsymbol{x})\phi_{i,E}(\boldsymbol{s}(k))$
  end
end

The next algorithm computes the local stiffness matrices obtained by the integration over one interior edge shared by two elements. We recall that the choice of the method is defined by the parameters ϵ and σ_e^0.

Algorithm 2.2.
Computing local contributions from edge e
initialize $\boldsymbol{M}_e^{11} = \boldsymbol{M}_e^{22} = \boldsymbol{M}_e^{12} = \boldsymbol{M}_e^{21} = \mathbf{0}$
initialize parameters ϵ and σ_e^0
initialize the quadrature weights $\boldsymbol{w}$ and points $\boldsymbol{s}$
compute edge length $|e|$
compute normal vector $\boldsymbol{n}_e$
get face neighbors E_e^1 and E_e^2
loop over quadrature points: for $k = 1$ to N_{G} do
  compute Jacobian matrices $\boldsymbol{M}_{E_e^1}$ and $\boldsymbol{M}_{E_e^2}$
  for $i = 1$ to N_{loc} do
    compute values of basis functions $\phi_{i,E_e^1}(\boldsymbol{s}(k))$ and $\phi_{i,E_e^2}(\boldsymbol{s}(k))$
    compute derivatives of basis functions $\nabla\phi_{i,E_e^1}(\boldsymbol{s}(k))$ and $\nabla\phi_{i,E_e^2}(\boldsymbol{s}(k))$
  end
  compute $\boldsymbol{M}_k^{11}$ contributions:
    for $i = 1$ to N_{loc} do
      for $j = 1$ to N_{loc} do
        $\boldsymbol{M}_e^{11}(i,j) = \boldsymbol{M}_e^{11}(i,j) - 0.5\boldsymbol{w}(k)|e|\phi_{i,E_e^1}(\boldsymbol{s}(k))(\nabla\phi_{j,E_e^1}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{11}(i,j) = \boldsymbol{M}_e^{11}(i,j) + 0.5\epsilon\boldsymbol{w}(k)|e|\phi_{j,E_e^1}(\boldsymbol{s}(k))(\nabla\phi_{i,E_e^1}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{11}(i,j) = \boldsymbol{M}_e^{11}(i,j) + \sigma_e^0\boldsymbol{w}(k)\phi_{i,E_e^1}(\boldsymbol{s}(k))\phi_{j,E_e^1}(\boldsymbol{s}(k))$
      end
    end
  compute $\boldsymbol{M}_k^{22}$ contributions:
    for $i = 1$ to N_{loc} do
      for $j = 1$ to N_{loc} do
        $\boldsymbol{M}_e^{22}(i,j) = \boldsymbol{M}_e^{22}(i,j) + 0.5\boldsymbol{w}(k)|e|\phi_{i,E_e^2}(\boldsymbol{s}(k))(\nabla\phi_{j,E_e^2}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{22}(i,j) = \boldsymbol{M}_e^{22}(i,j) - 0.5\epsilon\boldsymbol{w}(k)|e|\phi_{j,E_e^2}(\boldsymbol{s}(k))(\nabla\phi_{i,E_e^2}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{22}(i,j) = \boldsymbol{M}_e^{22}(i,j) + \sigma_e^0\boldsymbol{w}(k)\phi_{i,E_e^2}(\boldsymbol{s}(k))\phi_{j,E_e^2}(\boldsymbol{s}(k))$
      end
    end
  compute $\boldsymbol{M}_k^{12}$ contributions:
    for $i = 1$ to N_{loc} do
      for $j = 1$ to N_{loc} do
        $\boldsymbol{M}_e^{12}(i,j) = \boldsymbol{M}_e^{12}(i,j) - 0.5\boldsymbol{w}(k)|e|\phi_{i,E_e^1}(\boldsymbol{s}(k))(\nabla\phi_{j,E_e^2}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{12}(i,j) = \boldsymbol{M}_e^{12}(i,j) - 0.5\epsilon\boldsymbol{w}(k)|e|\phi_{j,E_e^2}(\boldsymbol{s}(k))(\nabla\phi_{i,E_e^1}(\boldsymbol{s}(k))\cdot\boldsymbol{n}_e)$
        $\boldsymbol{M}_e^{12}(i,j) = \boldsymbol{M}_e^{12}(i,j) - \sigma_e^0\boldsymbol{w}(k)\phi_{i,E_e^1}(\boldsymbol{s}(k))\phi_{j,E_e^2}(\boldsymbol{s}(k))$
      end
    end
  compute $\boldsymbol{M}_k^{21}$ contributions:
    for $i = 1$ to N_{loc} do

```
        for j = 1 to N_loc do
            M_e^21(i, j) = M_e^21(i, j) + 0.5 w(k)|e| φ_{i,E_e^2}(s(k))(∇φ_{j,E_e^1}(s(k)) · n_e)
            M_e^21(i, j) = M_e^21(i, j) + 0.5 ε w(k)|e| φ_{j,E_e^1}(s(k))(∇φ_{i,E_e^2}(s(k)) · n_e)
            M_e^21(i, j) = M_e^21(i, j) − σ_e^0 w(k) φ_{i,E_e^1}(s(k)) φ_{j,E_e^1}(s(k))
        end
    end
end
```

The corresponding C routines are given in Appendix B.2.

2.9.3 Global matrix and right-hand side

Assembling of the global matrix $\boldsymbol{A}_{\text{global}}$ is done in two steps. First, the local matrices $\boldsymbol{A}_E$ are added to the block diagonal entries of $\boldsymbol{A}_{\text{global}}$. We can assume that the mesh elements are numbered from 1 to N_{el}. We denote the global right-hand side by $\boldsymbol{b}_{\text{global}}$. The local contributions $\boldsymbol{b}_E$ can be added to $\boldsymbol{b}_{\text{global}}$ in the same algorithm.

ALGORITHM 2.3.
Volume contributions

```
initialize k = 0
loop over the elements: for k = 1 to N_el do
    compute local volume matrix A_{E_k}
    compute local right-hand side b_{E_k}
    for i = 1 to N_el do
        ie = i + k
        b_global(ie) = b_global(ie) + b_{E_k}(i)
        for j = 1 to N_loc do
            je = j + k
            A_global(ie, je) = A_global(ie, je) + A_{E_k}(i, j)
        end
        k = k + N_loc
    end
end
```

Second, we assemble the local matrices $\boldsymbol{M}_e^{ij}$ for $1 \le i, j \le 2$. We can assume that the edges are numbered from 1 to N_{face}. The numbers of the neighboring elements of the face k are E_1^k and E_2^k.

ALGORITHM 2.4.
Face contributions

```
loop over the edges: for k = 1 to N_face do
    get face neighbors E_k^1 and E_k^2
    if face is an interior face do
    compute local matrices M_k^11, M_k^22, M_k^12, M_k^21
```

```
    assemble M_k^11 contributions:
        for i = 1 to N_loc do
            ie = i + (E_k^1 − 1)N_loc
            for j = 1 to N_loc do
                je = j + (E_k^1 − 1)N_loc
                A_global(ie, je) = A_global(ie, je) + M_k^11(i, j)
            end
        end
    assemble M_k^22 contributions:
        for i = 1 to N_loc do
            ie = i + (E_k^2 − 1)N_loc
            for j = 1 to N_loc do
                je = j + (E_k^2 − 1)N_loc
                A_global(ie, je) = A_global(ie, je) + M_k^22(i, j)
            end
        end
    assemble M_k^12 contributions:
        for i = 1 to N_loc do
            ie = i + (E_k^1 − 1)N_loc
            for j = 1 to N_loc do
                je = j + (E_k^2 − 1)N_loc
                A_global(ie, je) = A_global(ie, je) + M_k^12(i, j)
            end
        end
    assemble M_k^21 contributions:
        for i = 1 to N_loc do
            ie = i + (E_k^2 − 1)N_loc
            for j = 1 to N_loc do
                je = j + (E_k^1 − 1)N_loc
                A_global(ie, je) = A_global(ie, je) + M_k^11(i, j)
            end
        end
    end
    else if face is a boundary face do
    compute local matrix M_k^11
    assemble M_k^11 contributions:
        for i = 1 to N_loc do
            ie = i + (E_k^1 − 1)N_loc
            for j = 1 to N_loc do
                je = j + (E_k^1 − 1)N_loc
                A_global(ie, je) = A_global(ie, je) + M_k^11(i, j)
            end
        end
    end
end
```

The corresponding C routines are given in Appendix B.2.

2.10 Numerical experiments

We solve on the unit square the model problem (2.16)–(2.18) with $\boldsymbol{K} = \mathbf{I}$, $\alpha = 0$, and Dirichlet boundary condition ($\Gamma_{\mathrm{D}} = \partial\Omega$). We present numerical convergence rates for both smooth and unsmooth exact solutions and for all primal DG methods. We vary the polynomial degree from $k = 1, 2, 3$. We denote the numerical error by

$$e_h = p - P_h.$$

We compute the seminorm $|||\nabla e_h|||_{H^0(\mathcal{E}_h)}$, which is bounded above by the energy norm, and the L^2 norm $\|e_h\|_{L^2(\Omega)}$. The penalty parameter σ_e^0 is equal to a constant σ for all interior edges. For the boundary edges, the penalty parameter is equal to 2σ for both IIPG and SIPG and equal to σ for NIPG.

2.10.1 Smooth solution

Let the exact solution be

$$\forall (x, y) \in (0, 1)^2, \quad p(x, y) = e^{-x-y^2}.$$

Table 2.3 contains the numerical errors $|||\nabla e_h|||_{H^0(\mathcal{E}_h)}$ and $\|e_h\|_{L^2(\Omega)}$ obtained on a fine triangular mesh. Convergence rates are computed as in (1.13). We choose $\beta_0 = 1$. The rates correspond to the theoretical rates: they are all optimal in the gradient broken norm $|||\nabla e_h|||_{H^0(\mathcal{E}_h)} = \mathcal{O}(h^k)$. The L^2 rates are optimal for the SIPG method: $\|e_h\|_{L^2(\Omega)} = \mathcal{O}(h^{k+1})$. For NIPG or IIPG, they are suboptimal if the polynomial degree is even.

Next, we use superpenalization and choose $\beta_0 = 3$. We consider a different smooth solution such that its Dirichlet value is a polynomial of degree k. The exact solution is given by

$$\forall (x, y) \in (0, 1)^2, \quad p(x, y) = x(x-1)y(y-1)e^{-x^2-y^2}.$$

Table 2.3. *Numerical errors and convergence rates for smooth function without superpenalization.*

Method	k	σ	$\|\|\|\nabla e_h\|\|\|_{H^0(\mathcal{E}_h)}$	Rate	$\|e_h\|_{L^2(\Omega)}$	Rate
NIPG	1	1	8.4846×10^{-3}	1.0123	8.9099×10^{-5}	2.0083
	2	1	7.6614×10^{-5}	2.0011	1.8632×10^{-6}	2.0186
	3	1	4.1740×10^{-7}	3.0157	3.3112×10^{-9}	4.0153
NIPG	2	0	8.3851×10^{-5}	2.0035	1.7316×10^{-6}	2.0307
	3	0	4.9857×10^{-7}	3.0103	3.8794×10^{-9}	4.0036
SIPG	1	6	8.9986×10^{-3}	1.0007	3.9981×10^{-5}	1.9717
	2	18	7.3139×10^{-5}	2.0009	1.5827×10^{-7}	2.9942
	3	36	3.8845×10^{-7}	3.0044	1.4746×10^{-9}	3.9879
IIPG	1	6	8.9885×10^{-3}	0.9996	3.2571×10^{-5}	1.9994
	2	18	7.1979×10^{-5}	2.0014	2.7825×10^{-7}	2.4695
	3	36	3.8427×10^{-7}	3.0023	1.5009×10^{-9}	3.9921

Table 2.4. *Numerical errors and convergence rates for smooth function with superpenalization.*

Method	k	σ	$\|\|\|\nabla e_h\|\|\|_{H^0(\mathcal{E}_h)}$	Rate	$\|e_h\|_{L^2(\Omega)}$	Rate
NIPG	1	1	5.1010×10^{-3}	0.9872	6.1576×10^{-5}	1.9537
	2	1	9.8300×10^{-5}	1.9707	3.7058×10^{-7}	3.1578
	3	1	8.5460×10^{-7}	2.9787	4.6797×10^{-9}	4.0106
IIPG	1	6	5.1107×10^{-3}	0.9959	6.2081×10^{-5}	1.9893
	2	18	9.8839×10^{-5}	1.9951	3.5405×10^{-7}	3.0000
	3	36	8.6042×10^{-7}	3.0135	4.6953×10^{-9}	4.0230

Therefore, Condition A is satisfied. Table 2.4 shows the numerical errors and convergence rates for NIPG and IIPG. The rates are optimal for the L^2 norm, as predicted by the theory.

2.10.2 Singular solution

We consider a solution $p \in H^{1+\delta}(\Omega)$ with $0 < \delta < 1$. Consider a domain $\Omega = (-1, 1)^2$ subdivided into four subdomains Ω_i such that $\Omega_1 = (0, 1)^2$, $\Omega_2 = (-1, 0) \times (0, 1)$, $\Omega_3 = (-1, 0)^2$, and $\Omega_4 = (0, 1) \times (-1, 0)$. We solve (2.16)–(2.18) with $\alpha = 0$, $f = 0$, and $\Gamma_D = \partial\Omega$. The coefficient matrix $\boldsymbol{K}$ is equal to $K_i \boldsymbol{I}$ on each subdomain Ω_i. We assume that $K_1 = K_3 = 5$ and $K_2 = K_4 = 1$. The exact solution in polar coordinates is

$$p(r, \theta) = r^\delta (a_i \sin(\delta\theta) + b_i \cos(\delta\theta)) \quad \text{in} \quad \Omega_i$$

with coefficients given up to nine accurate digits:

$$\begin{aligned}
a_1 &= 0.4472135955, \\
a_2 &= -0.7453559925, \\
a_3 &= -0.9441175905, \\
a_4 &= -2.401702643, \\
b_1 &= 1, \\
b_2 &= 2.333333333, \\
b_3 &= 0.55555555555, \\
b_4 &= -0.4814814814, \\
\delta &= 0.5354409456.
\end{aligned}$$

The exact solution is singular at the origin in the sense that its gradient is not defined at the point $(0, 0)$. We compute the DG solution on a sequence of uniformly refined rectangular meshes. The relative error in the L^2 norm, defined as $\frac{\|p - P_h\|_{L^2(\Omega)}}{\|p\|_{L^2(\Omega)}}$, is plotted

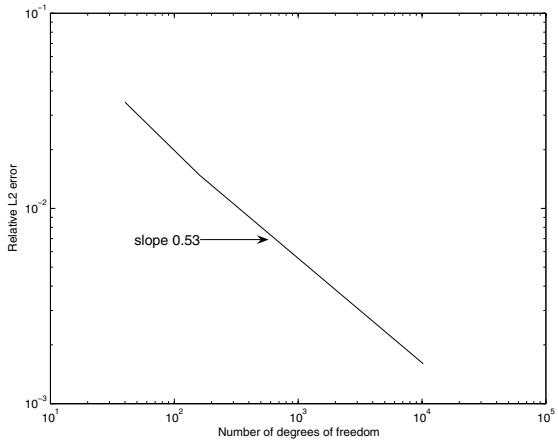


Figure 2.6. *Relative error in the L^2 norm versus the number of degrees of freedom.*

against the number of degrees of freedom in Fig. 2.6. We use the NIPG method without penalty and polynomials of degree three. We see that the convergence rate is independent of the polynomial order. This is expected, as the solution has poor regularity. Indeed, since p belongs to $H^{1+\alpha}(\Omega)$, the convergence rate in the L^2 norm is $\mathcal{O}(h^{2\alpha})$, or equivalently $\mathcal{O}(N^{\alpha})$, where N is the total number of degrees of freedom. In order to recover the rate obtained with the polynomial degrees, we need to locally refine the mesh around the origin.

2.10.3 Condition number

We fix the polynomial degree $k = 2$ and compute the condition number $\|\boldsymbol{A}\| \, \|\boldsymbol{A}^{-1}\|$ of the global matrix for the DG method with and without superpenalization. Fig. 2.7 shows that if no superpenalization is used, $\beta = 1$, then the condition number is $\mathcal{O}(h^2)$, whereas it is $\mathcal{O}(h^4)$ if $\beta = 3$. The method used here is NIPG with $\sigma_e^0 = 1$. Similar results are observed with SIPG and IIPG methods.

2.11 The local discontinuous Galerkin method

The local discontinuous Galerkin (LDG) method was introduced by Cockburn and Shu [36] and is based on the work by Bassi and Rebay [12]. We present the method for the model problem (2.16)–(2.18) with $\boldsymbol{K} = \mathbf{I}$ and $\alpha = 0$. Because this method solves for two unknowns, namely the solution and its gradient, it can be called a *dual* DG method or a *mixed* DG method.

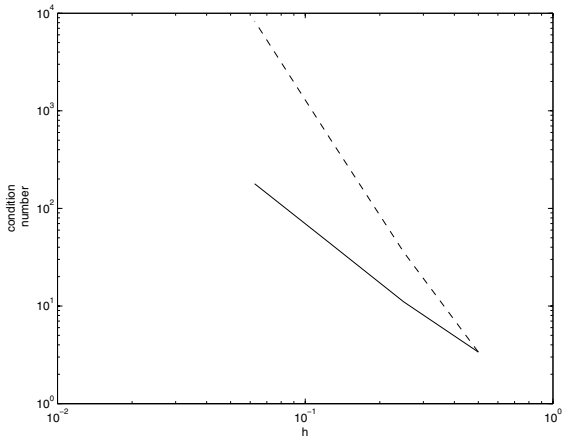


Figure 2.7. *Condition number versus mesh size for NIPG* 1: $\beta = 1$ *(solid line) and* $\beta = 3$ *(dashed line).*

2.11.1 Definition of the mixed DG method

Let us rewrite the model problem into a mixed form by introducing an auxiliary variable $\boldsymbol{u}$ for the gradient of the solution:

$$\boldsymbol{u} = \nabla p \quad \text{in} \quad \Omega, \tag{2.48}$$

$$-\nabla \cdot \boldsymbol{u} = f \quad \text{in} \quad \Omega. \tag{2.49}$$

The Dirichlet and Neumann boundary conditions are rewritten as

$$p = g_{\mathrm{D}} \quad \text{on} \quad \Gamma_{\mathrm{D}}, \tag{2.50}$$

$$\boldsymbol{u} \cdot \boldsymbol{n} = \boldsymbol{g} \cdot \boldsymbol{n} \quad \text{on} \quad \Gamma_{\mathrm{N}}. \tag{2.51}$$

Let $\mathcal{E}_h$ be a subdivision of Ω and let Γ_h be the set of interior edges (or faces). Let $\boldsymbol{v} \in H^1(\mathcal{E}_h)^d$ and let $q \in H^1(\mathcal{E}_h)$. We multiply (2.48) and (2.49) by $\boldsymbol{v}$ and q, integrate over one element $E \in \mathcal{E}_h$, and use Green's theorem (2.13):

$$\int_E \boldsymbol{u} \cdot \boldsymbol{v} = -\int_E p \nabla \cdot \boldsymbol{v} + \int_{\partial E} p \boldsymbol{v} \cdot \boldsymbol{n}_E,$$

$$\int_E \boldsymbol{u} \cdot \nabla q - \int_{\partial E} \boldsymbol{u} \cdot \boldsymbol{n}_E q = \int_E f q.$$

We look for a solution pair $(\boldsymbol{U}_h, P_h)$ that belongs to a finite-dimensional space $M_h^d \times M_h$, to be specified later, that satisfies for all $E \in \mathcal{E}_h$

$$\forall \boldsymbol{v} \in M_h^d, \quad \int_E \boldsymbol{U}_h \cdot \boldsymbol{v} + \int_E P_h \nabla \cdot \boldsymbol{v} = \int_{\partial E} \Phi(P_h) \boldsymbol{v} \cdot \boldsymbol{n}_E, \tag{2.52}$$

$$\forall q \in M_h, \quad \int_E \boldsymbol{U}_h \cdot \nabla q = \int_E f q + \int_{\partial E} \Psi(\boldsymbol{U}_h) \cdot \boldsymbol{n}_E q, \tag{2.53}$$

where $\Phi(P_h)$ and $\Psi(\boldsymbol{U}_h)$ are called numerical fluxes and they are defined below. Given two real numbers δ_1, δ_2 and a vector $\boldsymbol{\delta}_3 \in \mathbb{R}^d$, we define

$$\begin{aligned}
\forall e \in \Gamma_h, \quad & \Psi(\boldsymbol{U}_h)|_e = \{\boldsymbol{U}_h\} - (\delta_1[P_h])\boldsymbol{n}_e - ([\boldsymbol{U}_h]\cdot\boldsymbol{n}_e)\boldsymbol{\delta}_3,\\
\forall e \in \Gamma_h, \quad & \Phi(P_h)|_e = \{P_h\} + \boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[P_h] - \delta_2[\boldsymbol{U}_h]\cdot\boldsymbol{n}_e,\\
\forall e \in \Gamma_{\mathrm{D}}, \quad & \Psi(\boldsymbol{U}_h)|_e = \boldsymbol{U}_h - \delta_1(P_h - g_{\mathrm{D}})\boldsymbol{n}_e,\\
\forall e \in \Gamma_{\mathrm{D}}, \quad & \Phi(P_h)|_e = g_{\mathrm{D}},\\
\forall e \in \Gamma_{\mathrm{N}}, \quad & \Psi(\boldsymbol{U}_h)|_e = \boldsymbol{g},\\
\forall e \in \Gamma_{\mathrm{N}}, \quad & \Phi(P_h)|_e = P_h - \delta_2(\boldsymbol{U}_h - \boldsymbol{g})\cdot\boldsymbol{n}.
\end{aligned}$$

We note that the scheme is consistent because of the regularity of the exact solution; the numerical fluxes are equal to the exact fluxes. More precisely, we have

$$\forall e, \quad \Phi(p)|_e = p|_e, \quad \Psi(\boldsymbol{u})|_e = \boldsymbol{u}|_e.$$

By summing (2.52) and (2.53) over all the elements, we obtain

$$\begin{aligned}
\int_\Omega \boldsymbol{U}_h\cdot\boldsymbol{v} + \sum_{E\in\mathcal{E}_h}\int_E P_h\nabla\cdot\boldsymbol{v} &= \sum_{E\in\mathcal{E}_h}\int_{\partial E}\Phi(P_h)\boldsymbol{v}\cdot\boldsymbol{n}_E\\
= \sum_{e\in\Gamma_h}\int_e(\{P_h\} + \boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[P_h] - \delta_2[\boldsymbol{U}_h]\cdot\boldsymbol{n}_e)[\boldsymbol{v}]\cdot\boldsymbol{n}_e &+ \int_{\Gamma_{\mathrm{D}}} g_{\mathrm{D}}\boldsymbol{v}\cdot\boldsymbol{n}\\
&+ \int_{\Gamma_{\mathrm{N}}}(P_h - \delta_2(\boldsymbol{U}_h - \boldsymbol{g})\cdot\boldsymbol{n})\boldsymbol{v}\cdot\boldsymbol{n}
\end{aligned}$$

and

$$\begin{aligned}
\sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{U}_h\cdot\nabla q &= \int_\Omega fq + \sum_{E\in\mathcal{E}_h}\int_{\partial E}\Psi(\boldsymbol{U}_h)\cdot\boldsymbol{n}_E q\\
= \int_\Omega fq + \sum_{e\in\Gamma_h}\int_e(\{\boldsymbol{U}_h\} - (\delta_1[P_h])\boldsymbol{n}_e - ([\boldsymbol{U}_h]\cdot\boldsymbol{n}_e)\boldsymbol{\delta}_3)\cdot\boldsymbol{n}_e[q] &\\
+ \int_{\Gamma_{\mathrm{D}}}(\boldsymbol{U}_h - \delta_1(P_h - g_{\mathrm{D}})\boldsymbol{n}_e)\cdot\boldsymbol{n}_e q + \int_{\Gamma_{\mathrm{N}}}\boldsymbol{g}\cdot\boldsymbol{n}q &\\
= \int_\Omega fq + \sum_{e\in\Gamma_h}\int_e(\{\boldsymbol{U}_h\}\cdot\boldsymbol{n}_e - (\delta_1[P_h]) - ([\boldsymbol{U}_h]\cdot\boldsymbol{n}_e)\boldsymbol{\delta}_3\cdot\boldsymbol{n}_e)[q] &\\
+ \int_{\Gamma_{\mathrm{D}}}(\boldsymbol{U}_h\cdot\boldsymbol{n}_e - \delta_1(P_h - g_{\mathrm{D}}))q + \int_{\Gamma_{\mathrm{N}}}(\boldsymbol{g}\cdot\boldsymbol{n})q. &
\end{aligned}$$

Let us define the following bilinear forms:

$$\begin{aligned}
a_{\mathrm{ldg}}(\boldsymbol{U}_h,\boldsymbol{v}) &= \int_\Omega \boldsymbol{U}_h\cdot\boldsymbol{v} + \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{N}}}\int_e \delta_2[\boldsymbol{U}_h]\cdot\boldsymbol{n}_e[\boldsymbol{v}]\cdot\boldsymbol{n}_e,\\
b_{\mathrm{ldg}}(P_h,\boldsymbol{v}) &= \sum_{E\in\mathcal{E}_h}\int_E P_h\nabla\cdot\boldsymbol{v} - \sum_{e\in\Gamma_h}\int_e(\{P_h\} + \boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[P_h])[\boldsymbol{v}]\cdot\boldsymbol{n}_e - \int_{\Gamma_{\mathrm{N}}} P_h\boldsymbol{v}\cdot\boldsymbol{n},\\
J_{\mathrm{ldg}}(P_h,q) &= \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\int_e \delta_1[P_h][q].
\end{aligned}$$

We remark, by using Green's theorem (2.13), that the form b can be rewritten as

$$\begin{aligned} b_{\mathrm{ldg}}(P_h, \boldsymbol{v}) &= -\sum_{E\in\mathcal{E}_h}\int_E \nabla P_h\cdot\boldsymbol{v} + \sum_{e\in\Gamma_h}\int_e [P_h\boldsymbol{v}\cdot\boldsymbol{n}_e] + \int_{\Gamma_{\mathrm{D}}} P_h\boldsymbol{v}\cdot\boldsymbol{n} \\ &\quad -\sum_{e\in\Gamma_h}\int_e (\{P_h\}+\boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[P_h])[\boldsymbol{v}]\cdot\boldsymbol{n}_e \\ &= -\sum_{E\in\mathcal{E}_h}\int_E \nabla P_h\cdot\boldsymbol{v} + \sum_{e\in\Gamma_h}\int_e [P_h]\{\boldsymbol{v}\cdot\boldsymbol{n}_e\} \\ &\quad +\int_{\Gamma_{\mathrm{D}}} P_h\boldsymbol{v}\cdot\boldsymbol{n} - \sum_{e\in\Gamma_h}\int_e \boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[P_h][\boldsymbol{v}]\cdot\boldsymbol{n}_e; \end{aligned}$$

equivalently,

$$b_{\mathrm{ldg}}(P_h, \boldsymbol{v}) = -\sum_{E\in\mathcal{E}_h}\int_E \nabla P_h\cdot\boldsymbol{v} + \sum_{e\in\Gamma_h}\int_e [P_h](\{\boldsymbol{v}\cdot\boldsymbol{n}_e\} - \boldsymbol{\delta}_3\cdot\boldsymbol{n}_e[\boldsymbol{v}]\cdot\boldsymbol{n}_e) + \int_{\Gamma_{\mathrm{D}}} P_h\boldsymbol{v}\cdot\boldsymbol{n}. \quad (2.54)$$

The discrete space $M_h \subset H^1(\mathcal{E}_h)$ is chosen so that the following two conditions hold:

(i) $\{q \in L^2(\Omega) : \forall E \quad q|_E \in \mathbb{P}_k(E)\} \subset M_h$,

(ii) $\forall E \in \mathcal{E}_h, \forall q \in M_h, (\int_E \nabla q\cdot\boldsymbol{v} = 0 \quad \forall \boldsymbol{v} \in M_h^d) \Longrightarrow \nabla q = 0$.

The global formulation of the general LDG scheme is as follows: Find $\boldsymbol{U}_h \in M_h^d$ and $P_h \in M_h$ such that

$$\forall \boldsymbol{v} \in M_h^d, \quad a_{\mathrm{ldg}}(\boldsymbol{U}_h, \boldsymbol{v}) + b_{\mathrm{ldg}}(P_h, \boldsymbol{v}) = \int_{\Gamma_{\mathrm{D}}} g_{\mathrm{D}}\boldsymbol{v}\cdot\boldsymbol{n} + \int_{\Gamma_{\mathrm{N}}} \delta_2(\boldsymbol{g}\cdot\boldsymbol{n})\boldsymbol{v}\cdot\boldsymbol{n}, \quad (2.55)$$

$$\forall q \in M_h, \quad -b_{\mathrm{ldg}}(q, \boldsymbol{U}_h) + J_{\mathrm{ldg}}(P_h, q) = \int_\Omega f v + \int_{\Gamma_{\mathrm{D}}} \delta_1 g_{\mathrm{D}} q + \int_{\Gamma_{\mathrm{N}}} (\boldsymbol{g}\cdot\boldsymbol{n})q. \quad (2.56)$$

2.11.2 Existence and uniqueness of the solution

Lemma 2.15. *Assume that $\delta_1 > 0$ and $\delta_2 \geq 0$; then there exists a unique solution to the scheme* (2.55)–(2.56).

Proof. Assume that $g_{\mathrm{D}} = f = 0$ and $\boldsymbol{g} = \boldsymbol{0}$. Then, choosing $\boldsymbol{v} = \boldsymbol{U}_h$ in the first equation and $q = P_h$ in the second, we have

$$\begin{aligned} a_{\mathrm{ldg}}(\boldsymbol{U}_h, \boldsymbol{U}_h) + b_{\mathrm{ldg}}(P_h, \boldsymbol{U}_h) &= 0, \\ -b_{\mathrm{ldg}}(P_h, \boldsymbol{U}_h) + J_{\mathrm{ldg}}(P_h, P_h) &= 0. \end{aligned}$$

By adding the two equations above, we obtain

$$a_{\mathrm{ldg}}(\boldsymbol{U}_h, \boldsymbol{U}_h) + J_{\mathrm{ldg}}(P_h, P_h) = 0.$$

Equivalently,

$$\int_\Omega \boldsymbol{U}_h^2 + \sum_{e\in\Gamma_h\cup\Gamma_\mathrm{N}} \int_e \delta_2([\boldsymbol{U}_h]\cdot\boldsymbol{n}_e)^2 + \sum_{e\in\Gamma_h\cup\Gamma_\mathrm{D}} \int_e \delta_1 [P_h]^2 = 0.$$

Thus, if $\delta_1 > 0$ and $\delta_2 \geq 0$, we immediately have $\boldsymbol{U}_h = \boldsymbol{0}$, and we have

$$\forall e \in \Gamma_h \cup \Gamma_\mathrm{D}, \quad [P_h] = 0.$$

This implies that P_h is continuous across the domain. Then, (2.55) becomes

$$\forall \boldsymbol{v} \in M_h^d, \quad b_{\mathrm{ldg}}(P_h, \boldsymbol{v}) = 0.$$

Using the second form (2.54) of b_{ldg}, we have

$$\forall \boldsymbol{v} \in M_h^d, \quad \sum_{E\in\mathcal{E}_h} \int_E \nabla P_h \cdot \boldsymbol{v} = 0,$$

which implies that $\nabla P_h = 0$ because of the definition of M_h. Since P_h is continuous and zero on Γ_D, this implies that $P_h = 0$. □

2.11.3 A priori error estimates

Assume that $p \in H^{s+2}(\Omega)$ with $s \geq 0$ and that the mesh consists of elements that are affine equivalent to a particular reference element. Define the parameters

$$\delta_1 = \sigma_1 h^{\beta_1}, \quad \sigma_1 > 0,$$
$$\delta_2 = \sigma_2 h^{\beta_2}, \quad \sigma_2 \geq 0,$$

with $-1 \leq \beta_1 \leq 0 \leq \beta_2 \leq 1$. Then, for $s \geq 0$ and $k \geq 1$, we have the following error estimates:

$$\|p - P_h\|_{L^2(\Omega)} \leq C h^{\min(s+\frac{1}{2}(1+m),k+\frac{1}{2}(1-M))+\frac{1}{2}(1+m)} \|p\|_{H^{s+2}(\Omega)},$$
$$\|\boldsymbol{u} - \boldsymbol{U}_h\|_{L^2(\Omega)} \leq C h^{\min(s+\frac{1}{2}(1+m),k+\frac{1}{2}(1-M))} \|p\|_{H^{s+2}(\Omega)}.$$

If $k = 0$, we have

$$\|p - P_h\|_{L^2(\Omega)} \leq C h^{1-M} \|p\|_{H^{s+2}(\Omega)},$$
$$\|\boldsymbol{u} - \boldsymbol{U}_h\|_{L^2(\Omega)} \leq C h^{\frac{1-M}{2}} \|p\|_{H^{s+2}(\Omega)},$$

where

$$M = \max(-\beta_1, \beta_2), \quad m = \min(-\beta_1, \beta_2) \quad \text{if} \quad \sigma_2 > 0,$$
$$M = \max(-\beta_1, 1), \quad m = \min(-\beta_1, 1) \quad \text{if} \quad \sigma_2 = 0.$$

The convergence rates for $k \geq 1$ and $k = 0$ are given in Tables 2.5 and 2.6, respectively. Thus, for $k \geq 1$, we do not have optimal convergence rates for both errors. Assuming s is large enough, the optimal convergence rate for $\|p - P_h\|_{L^2(\Omega)}$ is obtained for cases where

Table 2.5. *Convergence rates of LDG method for piecewise polynomial approximation of degree greater than or equal to one.*

δ_1	δ_2	$\|\boldsymbol{u} - \boldsymbol{U}_h\|_{L^2(\Omega)}$	$\|p - P_h\|_{L^2(\Omega)}$
1	0	$\min(s + 1/2, k)$	$\min(s + 1/2, k) + 1/2$
1	h	$\min(s + 1/2, k)$	$\min(s + 1/2, k) + 1/2$
h^{-1}	0	$\min(s + 1, k)$	$\min(s + 1, k) + 1$
h^{-1}	h	$\min(s + 1, k)$	$\min(s + 1, k) + 1$
1	1	$\min(s, k) + 1/2$	$\min(s, k) + 1$
h^{-1}	1	$\min(s + 1/2, k)$	$\min(s + 1/2, k) + 1/2$

Table 2.6. *Convergence rates of LDG method for piecewise constant approximation.*

δ_1	δ_2	$\|\boldsymbol{u} - \boldsymbol{U}_h\|_{L^2(\Omega)}$	$\|p - P_h\|_{L^2(\Omega)}$
1	1	1/2	1
h^{-1}	0	0	0
h^{-1}	1	0	0
h^{-1}	h	0	0
1	0	0	0
1	h	0	0

(δ_1, δ_2) belongs to $\{(h^{-1}, 0), (h^{-1}, h), (1, 1)\}$. For the error $\|\boldsymbol{u} - \boldsymbol{U}_h\|_{L^2(\Omega)}$, the best rate is $\mathcal{O}(h^{k+1/2})$. In the case where $k = 0$, the method converges in the case where $\delta_1 = \mathcal{O}(1)$ and $\delta_2 = \mathcal{O}(1)$.

2.12 DG versus classical finite element method

In this section, we denote the finite element method by CG (continuous Galerkin), and we present a comparison of CG versus DG from a practical point of view. The CG method was briefly introduced in Section 2.2.2. We recall that the CG solution is a continuous piecewise polynomial, whereas the DG solution is a discontinuous piecewise polynomial.

(i) **Age of the method:** The CG method has been around for more than 60 years, and hundreds of books have been written on many aspects of the method. The primal DG methods have only recently gained an interest from the scientific community. In many cases, one can apply the techniques developed for CG to solve problems related to DG. Still, many questions remain unanswered.

(ii) **Size of problem:** For DG, the total number of degrees of freedom is proportional to the number of elements in the mesh. The constant of proportionality is a function of the polynomial degree. For CG, the degrees of freedom depend on the number of vertices and possibly the number of vertices and elements in the mesh. For instance, consider a structured mesh of 5×5 rectangular elements. The degrees of freedom for a DG approximation of degree 1, 2, 3, 4 are 75, 150, 250, 375, respectively, whereas the degrees of freedom for a CG approximation of degree 1, 2, 3, 4 are 36, 121, 256, 441, respectively. Thus, on such small mesh, if $k \geq 3$, the CG method is more costly than

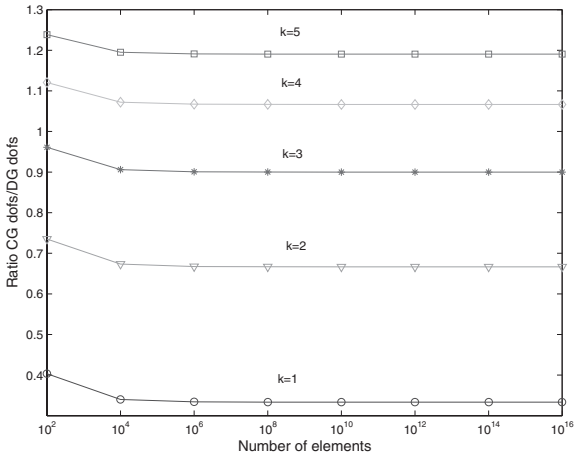


Figure 2.8. *Ratios of degrees of freedom for CG over DG with respect to the total number of degrees of freedom, computed on a uniform rectangular mesh.*

DG. The reason is that we have to use the space $\mathbb{Q}_k$ on rectangular elements for CG, but we can still use the space $\mathbb{P}_k$ on rectangular elements for DG. Fig. 2.8 gives the ratio of the total number of degrees of freedom for CG to the total number of degrees of freedom for DG on a uniform mesh of $N \times N$ rectangles. We vary N from 10 to 10^8. The CG method is less costly than DG if the polynomial degree is less than or equal to 3. The ratios tend to the limit values $1/3, 2/3, 9/10, 16/15, 15/21$ for the degrees 1, 2, 3, 4, 5, respectively. On triangular meshes, the DG method is more costly than the CG method. For example, on a uniform mesh of $N \times N \times 2$ triangular elements, the ratios of the number of degrees of freedom for CG over DG tend to $1/6, 1/3, 9/20, 14/30, 25/42$ for the degrees 1, 2, 3, 4, 5, respectively, as N tends to infinity. We see that this ratio increases as the order of polynomial increases.

(iii) **Meaning of degrees of freedom:** Many users of the finite element method compute only with piecewise polynomials of degree one. Because of the "chapeau" basis functions, the resulting CG degrees of freedom correspond to the values of the CG solution at the vertices of the mesh. This is a desirable property that can be exploited, for instance, in visualization routines. The degrees of freedom in the DG method do not have any meaning besides being coefficients in the expansion of the solution with respect to the basis functions. This means that in order to obtain the DG solution at a particular point, one has to compute the expansion, i.e., compute the basis functions and multiply them by the coefficients. At a given vertex, there are several values of the numerical solution. Note that we can also use the same local basis functions as in CG.

(iv) **Hanging nodes:** The name "hanging node" comes from the CG method for which mesh vertices correspond to degrees of freedom or nodes. We abuse the notation and call a hanging node any mesh vertex located on the interior of an edge (or face). Fig. 2.9 contains a mesh with 11 hanging nodes. This nonconforming mesh can be

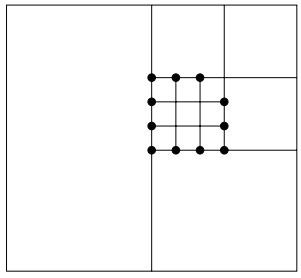


Figure 2.9. *Rectangular mesh with hanging nodes (black dots).*

used with the DG method of any order, but it cannot be used with the CG method. In general, one can have as many hanging nodes per face as one wishes for the DG method because there are no continuity constraints between the elements. In the case of the CG method, one can have at most one hanging node per edge, and special continuous basis functions have to be used.

(v) **Polynomial degree and basis functions:** It is relatively easy to change the degree of approximation of a DG solution using the same piece of software. Only the routine that computes the basis functions should be modified. The user (even beginners) can then easily perform hp-analysis of the method. Using the data structure described in Section 2.9.1, it is easy to write a DG code that uses different polynomial degrees for different mesh elements. This is an important benefit of using discontinuous approximations. For the CG method, things are less simple. In general, CG codes are first written for the piecewise linear approximations. The user then writes different codes for other polynomial degrees. As the degrees increase, the basis functions become more complicated and one has to keep track of the degrees of freedom. Some care and thought are required to obtain an hp software. In the CG method, basis functions are obtained by "pasting together" local basis functions whose support lie in one mesh element. These local basis functions can also be used to form the basis for the DG method. It suffices to extend those local basis functions by zero outside the mesh element. In practice, a simple choice of local basis functions for DG is the set of monomials.

(vi) **Accuracy:** Both methods converge as the mesh size decreases or as the polynomial degree increases. Error estimates in the energy norm are optimal. However, error estimates in the L^2 norm are optimal for the CG method, whereas they are optimal only in the symmetric version (SIPG) if no superpenalization is used. For a fixed mesh, it is irrelevant to compare the accuracy of DG with CG, as it is easy to come up with a problem that yields a better DG solution than CG and vice versa.

(vii) **Boundary condition:** Dirichlet boundary conditions are usually imposed weakly with the DG method, whereas they are imposed strongly with the CG method. But this is a matter of taste, and we can also impose the boundary conditions strongly with the DG method.

(viii) **Mass conservation:** As discussed in Section 2.7.3, the DG method satisfies a local mass balance. The CG method satisfies only a global mass balance over the whole computational domain. The property of mass conservation is crucial in flow and

transport problems, such as the ones arising in porous media. For other applications, the importance of the local mass conservation is questionable.

2.13 Bibliographical remarks

The introduction of penalty terms originates from Nitsche's work [83] in which Dirichlet boundary conditions are imposed weakly by means of the addition of a penalty term in the variational formulation rather than strongly in the space of test functions. Babuška [7] proposes another penalty method that enforces the Dirichlet boundary condition weakly. The idea of using discontinuous approximations and penalty parameters as a way to enforce interelement continuity was first introduced and analyzed by Wheeler [109] and Percell and Wheeler [85]. The method was generalized to nonlinear elliptic and parabolic problems by Arnold [1]. Similar ideas appear in the work of Baker for biharmonic problems [10]. More recently, the NIPG methods with zero penalty have been analyzed for one-dimensional problems by Babuška, Baumann, and Oden [8] and for two- and three-dimensional problems by Rivière, Wheeler, and Girault [96, 95]. The NIPG methods with nonzero penalty have been introduced by Rivière, Wheeler, and Girault [96, 95] and by Houston, Schwab, and Süli [72]: error estimates are obtained with respect to both the mesh size h and the polynomial degree k. The analysis of LDG methods can be found in the work of Castillo et al. [24], Perugia and Schötzau [86], and Dawson [40]. A unified framework for both primal and LDG methods is proposed by Arnold et al. [3, 4]. Other relevant works include [19, 50, 13, 23, 35].

Exercises

2.1. Define the set of locally integrable functions

$$L^1_{\mathrm{loc}}(\Omega) = \{v : \forall K \text{ compact} \subset \text{interior } \Omega : v|_K \in L^1(K)\}.$$

Show that if v is locally integrable, the mapping defined below is a distribution:

$$T_v(\phi) = \int_\Omega v\phi.$$

2.2. Show that if a function ϕ belongs to $H^1(E)$ such that $\int_E \phi = 0$, then there is a constant C independent of h_E such that

$$\|\phi\|_{L^2(E)} \leq Ch_E \|\nabla\phi\|_{L^2(E)}.$$

(Hint: use approximation results.)

2.3. Modify the assembling algorithm in the case of different polynomial degrees for different elements. (Hint: it might be useful to introduce an array containing the cumulative local degrees of freedom.)

2.4. Show that the mapping F_E defined by (2.33) is affine if E is a parallelogram.

2.5. Let Ω be the L-shaped domain given in Fig. 2.10. The domain is subdivided into 12 triangles. Element numbers and edge numbers are given in the left and middle

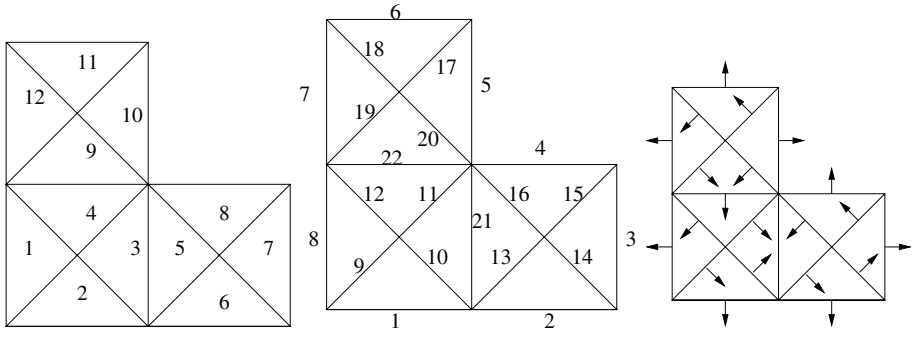


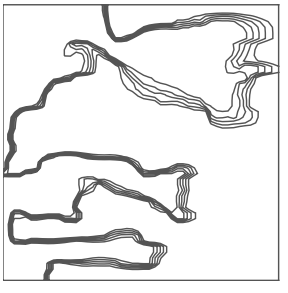
Figure 2.10. *Element numbers (left), edge numbers (middle), and normal directions (right).*

figures. The orientation of the unit normal vector $\boldsymbol{n}_e$ for each edge e is given in the right figure. Write the global matrix obtained in that case: the entries should be functions of the local matrices.

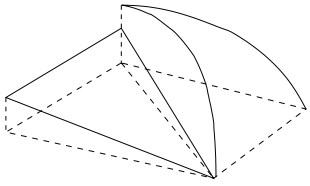
2.6. Prove Young's inequality (2.15) and Cauchy–Schwarz's inequality (2.14).

2.7. Show that the form a_ϵ is continuous on $(\mathcal{D}_k(\mathcal{E}_h))^2$ if $\sigma_e^0 > 0$; i.e, show that for all $v, w \in \mathcal{D}_k(\mathcal{E}_h)$

$$a_\epsilon(v, w) \leq M \|v\|_{\mathcal{E}} \|w\|_{\mathcal{E}}.$$



Part II
Parabolic Problems



Chapter 3
Purely parabolic problems

This chapter studies the numerical solution of the linear parabolic problem following the method of lines. First, the problem is discretized in space using the DG method. The resulting system of ordinary differential equations is called the semidiscrete formulation. Second, the fully discrete problem is obtained by discretizing in time. We study several choices for the time discretization. We refer the reader to Chapter 2 for the definition of notation used in this chapter.

3.1 Preliminaries

3.1.1 Functional spaces

If $z(t, \boldsymbol{x})$ is a function of time t and space $\boldsymbol{x}$, we freely use the notation

$$\forall t, \forall \boldsymbol{x}, \quad z(t)(\boldsymbol{x}) = z(t, \boldsymbol{x}).$$

We consider spaces of functions mapping the time interval $(0, T)$ to a normed space V equipped with the norm $\|\cdot\|_V$. More precisely, for any number $r \geq 1$, we define [81]

$$L^r(0, T; V) = \left\{ z : (0, T) \to V \text{ measurable} : \int_0^T \|z(t)\|_V^r dt < \infty \right\}$$

with

$$\|z\|_{L^r(0,T;V)} = \left(\int_0^T \|z(t)\|_V^r dt \right)^{1/r}.$$

If in addition V is complete (Banach space), then $L^r(0, T; V)$ is also complete.

3.1.2 Gronwall's inequalities

Gronwall's inequalities are important tools for analyzing time-dependent problems. We present both continuous and discrete versions [68].

Lemma 3.1 (Continuous Gronwall inequality). *Let f, g, h be piecewise continuous nonnegative functions defined on (a, b). Assume that g is nondecreasing. Assume that there is a positive constant C independent of t such that*

$$\forall t \in (a, b), \quad f(t) + h(t) \leq g(t) + C \int_a^t f(s)ds.$$

Then,

$$\forall t \in (a, b), \quad f(t) + h(t) \leq e^{C(t-a)} g(t).$$

Lemma 3.2 (Discrete Gronwall inequality). *Let $\Delta t, B, C > 0$ and let $(a_n)_n$, $(b_n)_n$, $(c_n)_n$, $(d_n)_n$ be sequences of nonnegative numbers satisfying*

$$\forall n \geq 0, \quad a_n + \Delta t \sum_{i=0}^{n} b_i \leq B + C\Delta t \sum_{i=0}^{n} a_i + \Delta t \sum_{i=0}^{n} c_i.$$

Then, if $C\Delta t < 1$,

$$\forall n \geq 0, \quad a_n + \Delta t \sum_{i=0}^{n} b_i \leq e^{C(n+1)\Delta t} \left(B + \Delta t \sum_{i=0}^{n} c_i \right).$$

3.1.3 Taylor's expansions

Let f be a $\mathcal{C}^{n+1}$ function on the interval (a, b). Then, the Taylor expansions state that there exists a number ξ between a and b such that

$$f(b) = f(a) + (b-a)f'(a) + \cdots + \frac{(b-a)^n}{n!} f^{(n)}(a) + \frac{(b-a)^{n+1}}{(n+1)!} f^{(n+1)}(\xi). \quad (3.1)$$

Another form of Taylor expansion uses an integral remainder.

$$f(b) = f(a) + (b-a)f'(a) + \cdots + \frac{(b-a)^n}{n!} f^{(n)}(a) + \int_a^b \frac{(b-t)^n}{n!} f^{(n+1)}(t)dt. \quad (3.2)$$

3.1.4 Poincaré's inequalities

The classical Poincaré–Friedrichs inequality in $H^1(\Omega)$ says that there is a constant C such that [82]

$$\forall v \in H^1(\Omega), \quad \|v\|_{L^2(\Omega)} \leq C \left(\|\nabla v\|_{L^2(\Omega)} + \left| \int_{\partial\Omega} v \right| \right).$$

Consequently, we have

$$\forall v \in H_0^1(\Omega), \quad \|v\|_{L^2(\Omega)} \leq C \|\nabla v\|_{L^2(\Omega)}. \quad (3.3)$$

We remark that (3.3) implies that the mapping $v \mapsto \|\nabla v\|_{L^2(\Omega)}$ is a norm for the space $H_0^1(\Omega)$. Assume that Γ_D is a subset of the boundary $\partial\Omega$ with $|\Gamma_D| > 0$. A generalization of this inequality to the broken Sobolev space $H^1(\mathcal{E}_h)$ is (see [61, 16])

$$\forall v \in H^1(\mathcal{E}_h), \quad \|v\|_{L^2(\Omega)} \leq C \left(|||\nabla v|||^2_{H^0(\mathcal{E}_h)} + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{1}{|e|^{\frac{1}{d-1}}} \|[v]\|^2_{L^2(e)} \right)^{1/2}. \quad (3.4)$$

If in addition $\beta > (d-1)^{-1}$, a straightforward corollary is the bound

$$\forall v \in H^1(\mathcal{E}_h), \quad \|v\|_{L^2(\Omega)} \le C \left(|||\nabla v|||^2_{H^0(\mathcal{E}_h)} + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{1}{|e|^\beta} \|[v]\|^2_{L^2(e)} \right)^{1/2}. \tag{3.5}$$

3.1.5 Inverse inequalities

Let E be a bounded domain in $\mathbb{R}^d$ with diameter h_E. Then, there is a constant C independent of h_E such that for any polynomial function v of degree k defined on E we have

$$\forall 0 \le j \le k, \quad \|\nabla^j v\|_{L^2(E)} \le C h_E^{-j} \|v\|_{L^2(E)}. \tag{3.6}$$

3.2 Model problem

Let Ω be a bounded polygonal domain in $\mathbb{R}^d$, $d = 1$, 2 or 3, and let $(0, T)$ be a time interval. For $f \in L^2(0, T; L^2(\Omega))$, g_D in $L^2(0, T; H^{\frac{1}{2}}(\partial\Omega))$, and $z_0 \in L^2(\Omega)$ we consider the linear parabolic problem with Dirichlet boundary condition:

$$\frac{\partial z}{\partial t} - \nabla \cdot (K \nabla z) = f \quad \text{in} \quad (0, T) \times \Omega, \tag{3.7}$$

$$z = g_D \quad \text{on} \quad (0, T) \times \partial\Omega, \tag{3.8}$$

$$z = z_0 \quad \text{on} \quad \{0\} \times \Omega. \tag{3.9}$$

This problem models the conduction of heat in Ω over the time period $[0, T]$, with z being the body temperature and K the heat diffusion coefficient. This problem also models the diffusion of a chemical species of concentration z in a porous medium. We assume that K is bounded uniformly below and above:

$$\forall \boldsymbol{x} \in \Omega, \quad 0 < K_0 \le K(\boldsymbol{x}) \le K_1.$$

A strong solution of the parabolic problem belongs to $\mathcal{C}^2([0, T] \times \Omega)$ and satisfies (3.7)–(3.9) pointwisely. A weak solution of the parabolic problem belongs to the space $L^2(0, T; H^1(\Omega)) \cap H^1(0, T; L^2(\Omega))$ and satisfies the variational formulation

$$\forall t > 0, \quad \forall v \in H^1_0(\Omega), \quad \left(\frac{\partial z}{\partial t}, v \right)_\Omega + (K \nabla z, \nabla v)_\Omega = (f, v)_\Omega,$$

$$\forall v \in H^0_1(\Omega), \quad (z(0), v)_\Omega = (z_0, v)_\Omega.$$

Let us now define a semidiscrete solution of the parabolic problem.

3.3 Semidiscrete formulation

In this section, we approximate the solution $z(t)$ by a function $Z_h(t)$ that belongs to the finite-dimensional space $\mathcal{D}_k(\mathcal{E}_h)$ for all $t \ge 0$. The solution Z_h is referred to as the *semidiscrete* solution, or sometimes as the *continuous in time* solution.

Let $v \in H^s(\mathcal{E}_h)$ for $s > 3/2$, multiply (3.7) by v, integrate over one mesh element, use Green's theorem, and sum over all elements to obtain

$$\forall t > 0, \quad \int_\Omega \frac{\partial z}{\partial t} v + \sum_{E \in \mathcal{E}_h} \int_E K \nabla z(t) \cdot \nabla v - \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{K \nabla z(t) \cdot \boldsymbol{n}_e\} [v]$$
$$+ \epsilon \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{K \nabla v \cdot \boldsymbol{n}_e\} [z(t)] + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e [z(t)][v] = L(t; v), \quad (3.10)$$

where

$$L(t; v) = \int_\Omega f(t) v + \sum_{e \in \partial\Omega} \int_e g_{\mathrm{D}}(t) \left(\epsilon (K \nabla v \cdot \boldsymbol{n}_e) + \frac{\sigma_e^0}{|e|^{\beta_0}} v \right).$$

We have skipped many details, as the derivation is similar to the one given in Section 2.4.1. The only difference is the time derivative term. Similarly to (2.36), we define the energy norm for the parabolic problem

$$\|v\|_{\mathcal{E}} = \left(\sum_E \|K^{1/2} \nabla v\|_{L^2(E)}^2 + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|^{\beta_0}} \|v\|_{L^2(e)}^2 \right)^{1/2}.$$

We still denote the bilinear form by a_ϵ as in (2.23):

$$a_\epsilon(w, v) = \sum_{E \in \mathcal{E}_h} \int_E K \nabla w \cdot \nabla v - \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{K \nabla w \cdot \boldsymbol{n}_e\} [v]$$
$$+ \epsilon \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{K \nabla v \cdot \boldsymbol{n}_e\} [w] + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e [w][v],$$

and we assume that coercivity of a_ϵ holds true for some $\kappa > 0$:

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \kappa \|v\|_{\mathcal{E}}^2 \le a_\epsilon(v, v). \quad (3.11)$$

Thus, the semidiscrete variational formulation is as follows: For all $t \ge 0$, find $Z_h(t) \in \mathcal{D}_k(\mathcal{E}_h)$ such that

$$\forall t > 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{\partial Z_h}{\partial t}, v \right)_\Omega + a_\epsilon(Z_h(t), v) = L(t; v), \quad (3.12)$$
$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad (Z_h(0), v)_\Omega = (\tilde{z}_0, v)_\Omega. \quad (3.13)$$

Depending on the value of the parameter ϵ, the method is called SIPG ($\epsilon = -1$), NIPG ($\epsilon = 1$), or IIPG ($\epsilon = 0$). The initial condition $\tilde{z}_0$ can be chosen to be z_0 if z_0 belongs to the discrete space $\mathcal{D}_k(\mathcal{E}_h)$, or it can be chosen to be $\tilde{z}(0)$, where $\tilde{z}$ is an approximation of z to be specified later. Using the global basis functions defined by (2.34), we can expand the semidiscrete solution

$$\forall t \in (0, T), \quad \forall \boldsymbol{x} \in \Omega, \quad Z_h(t, \boldsymbol{x}) = \sum_{E \in \mathcal{E}_h} \sum_{i=1}^{N_{\mathrm{loc}}} \xi_i^E(t) \phi_i^E(\boldsymbol{x}). \quad (3.14)$$

The degrees of freedom ξ_i^E's are functions of time. Let N_{el} denote the number of elements in the mesh. We can rename the basis functions and the degrees of freedom such that

$$\{\phi_i^E : 1 \le i \le N_{\mathrm{loc}},\ E \in \mathcal{E}_h\} = \{\tilde{\phi}_j : 1 \le j \le N_{\mathrm{loc}} N_{\mathrm{el}}\},$$
$$\{\xi_i^E : 1 \le i \le N_{\mathrm{loc}},\ E \in \mathcal{E}_h\} = \{\tilde{\xi}_j : 1 \le j \le N_{\mathrm{loc}} N_{\mathrm{el}}\}.$$

Plugging (3.14) into (3.12)–(3.13) yields a linear system of ordinary differential equations with the vector of unknowns $\tilde{\boldsymbol{\xi}} = (\tilde{\xi}_j)_j$:

$$\boldsymbol{M}\frac{d\tilde{\boldsymbol{\xi}}}{dt}(t) + \boldsymbol{A}\tilde{\boldsymbol{\xi}}(t) = \boldsymbol{F}(t),$$
$$\boldsymbol{M}\tilde{\boldsymbol{\xi}}(0) = \tilde{\mathbf{Z}}_0.$$

The matrices $\boldsymbol{M} = (M_{ij})_{ij}$, $\boldsymbol{A} = (A_{ij})_{ij}$ are called the mass and stiffness matrices, and they are defined by

$$\forall 1 \le i, j \le N_{\mathrm{loc}} N_{\mathrm{el}}, \quad M_{ij} = (\tilde{\phi}_j, \tilde{\phi}_i)_\Omega, \quad A_{ij} = a_\epsilon(\tilde{\phi}_j, \tilde{\phi}_i). \tag{3.15}$$

From (3.11), the matrix $\boldsymbol{A}$ is positive definite. In fact, it is the matrix resulting from the DG method applied to an elliptic problem. The matrix $\boldsymbol{M}$ is block diagonal, symmetric positive definite, and thus it is invertible. The vectors $\boldsymbol{F}(t)$ and $\tilde{\mathbf{Z}}_0$ have components $(L(t; \tilde{\phi}_i))_i$ and $((\tilde{z}_0, \tilde{\phi}_i)_\Omega)_i$. The existence and uniqueness of $\tilde{\boldsymbol{\xi}}$ is obtained from the theory of ordinary differential equations.

3.3.1 A priori bounds

In this section, we derive a priori bounds (also called stability bounds) for the numerical solution. Choosing $v = Z_h(t)$ in (3.12) and using the coercivity result (3.11), we have

$$\frac{1}{2}\frac{d}{dt}\|Z_h\|_{L^2(\Omega)}^2 + \kappa\|Z_h(t)\|_{\mathcal{E}}^2 \le |L(t; Z_h(t))|.$$

From Cauchy–Schwarz's inequality, the right-hand side is bounded by

$$|L(t; Z_h(t))| \le \|f(t)\|_{L^2(\Omega)}\|Z_h(t)\|_{L^2(\Omega)} + \sum_{e\in\partial\Omega}\left(\|K\nabla Z_h(t)\cdot\boldsymbol{n}_e\|_{L^2(e)} + \frac{\sigma_e^0}{|e|^{\beta_0}}\|Z_h(t)\|_{L^2(e)}\right)\|g_{\mathrm{D}}(t)\|_{L^2(e)}.$$

Next, we use the trace inequality (2.5) and Young's inequality. As usual, the constant C is independent of the mesh size h. We skip the details, as the derivation of similar bounds is done several times in Chapter 2:

$$|L(t; Z_h(t))| \le \|f(t)\|_{L^2(\Omega)}\|Z_h(t)\|_{L^2(\Omega)} + \frac{\kappa}{2}\|Z_h(t)\|_{\mathcal{E}}^2 + C\sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\|g_{\mathrm{D}}(t)\|_{L^2(e)}^2. \tag{3.16}$$

Therefore, we obtain the intermediate result:

$$\frac{1}{2}\frac{d}{dt}\|Z_h\|^2_{L^2(\Omega)} + \frac{\kappa}{2}\|Z_h(t)\|^2_{\mathcal{E}} \le \|f(t)\|_{L^2(\Omega)}\|Z_h(t)\|_{L^2(\Omega)} + C\sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\|g_{\mathrm{D}}(t)\|^2_{L^2(e)}. \tag{3.17}$$

We present two possible approaches for obtaining the final a priori bound. The first one is more standard and uses Gronwall's inequality (3.1). The second approach takes advantage of Poincaré's inequality (3.4).

Approach using Gronwall inequality:

We simply bound

$$\|f(t)\|_{L^2(\Omega)}\|Z_h(t)\|_{L^2(\Omega)} \le \frac{1}{2}\|f(t)\|^2_{L^2(\Omega)} + \frac{1}{2}\|Z_h(t)\|^2_{L^2(\Omega)},$$

multiply the equation by 2, and integrate from 0 to t:

$$\begin{aligned}\|Z_h(t)\|^2_{L^2(\Omega)} + \kappa\int_0^t \|Z_h(s)\|^2_{\mathcal{E}} \le \int_0^t \|f(s)\|^2_{L^2(\Omega)} + \int_0^t \|Z_h(s)\|^2_{L^2(\Omega)} \\ + \|Z_h(0)\|^2_{L^2(\Omega)} + C\sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\int_0^t \|g_{\mathrm{D}}(s)\|^2_{0,e}.\end{aligned}$$

Then, by the continuous Gronwall inequality (Lemma 3.1), we conclude that

$$\begin{aligned}\|Z_h(t)\|^2_{L^2(\Omega)} + \kappa\int_0^t \|Z_h(s)\|^2_{\mathcal{E}} \\ \le C\left(\int_0^t \|f(s)\|^2_{L^2(\Omega)} + \|Z_h(0)\|^2_{L^2(\Omega)} + \sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\int_0^t \|g_{\mathrm{D}}(s)\|^2_{0,e}\right).\end{aligned} \tag{3.18}$$

The constant C grows exponentially in time. We observe that this approach is valid for all primal DG methods, in particular for the NIPG method with zero penalty.

Approach using Poincaré's inequality:

If we use (3.5) and Young's inequality to bound $\|Z_h(t)\|_{L^2(\Omega)}$, we have

$$\frac{1}{2}\frac{d}{dt}\|Z_h\|^2_{L^2(\Omega)} + \frac{\kappa}{2}\|Z_h(t)\|^2_{\mathcal{E}} \le \frac{\kappa}{4}\|Z_h(t)\|^2_{\mathcal{E}} + C\|f(t)\|^2_{L^2(\Omega)} + C\sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\|g_{\mathrm{D}}(t)\|^2_{L^2(e)}.$$

After multiplying by 2 and integrating from 0 to t, we obtain

$$\begin{aligned}\|Z_h(t)\|^2_{L^2(\Omega)} + \frac{\kappa}{2}\int_0^t \|Z_h(s)\|^2_{\mathcal{E}} \le \|\tilde{z}_0\|^2_{L^2(\Omega)} \\ + C\int_0^t \|f(s)\|^2_{L^2(\Omega)} + C\sum_{e\in\partial\Omega}\frac{1}{|e|^{\beta_0}}\int_0^t \|g_{\mathrm{D}}(s)\|^2_{L^2(e)},\end{aligned}$$

which is the same inequality as (3.18) modulo some multiplicative constants. However, here the constant C is independent of time. This approach is valid if the penalty value σ_e^0 is positive for all faces e. The final result is stated in the following lemma.

Lemma 3.3. *Assume that $\beta_0 \geq (d-1)^{-1}$. There exists a positive constant C independent of h such that*

$$\|Z_h\|^2_{L^\infty(0,T;L^2(\Omega))} + \int_0^T \|Z_h\|^2_{\mathcal{E}} \leq C\|\tilde{z}_0\|^2_{L^2(\Omega)}$$
$$+C\|f\|^2_{L^2(0,T;L^2(\Omega))} + C\sum_{e\in\partial\Omega} \frac{1}{|e|^{\beta_0}}\|g_{\mathrm{D}}(s)\|^2_{L^2(0,T;L^2(e))}. \tag{3.19}$$

Remark: The reader will notice that the last term on the right-hand side of (3.19) blows up as the mesh size h tends to zero. This has nothing to do with the discontinuous finite element spaces, but it comes from the fact that the Dirichlet boundary condition is imposed weakly. One can eliminate this issue by choosing to impose the boundary condition strongly. The space of test functions is then defined as

$$\mathcal{D}^0_k(\mathcal{E}_h) = \{v \in \mathcal{D}_k(\mathcal{E}_h) : v = 0 \text{ on } \partial\Omega\}.$$

In that case, the a priori estimates are

$$\|Z_h\|^2_{L^\infty(0,T;L^2(\Omega))} + \int_0^T \|Z_h\|^2_{\mathcal{E}} \leq C\|\tilde{z}_0\|^2_{L^2(\Omega)} + C\|f\|^2_{L^2(0,T;L^2(\Omega))},$$

and the solution is equal to $Z_h + g_h$, where $g_h \in \mathcal{D}_k(\mathcal{E}_h)$ is an interpolant of a lift of the Dirichlet boundary condition g_{D}.

3.3.2 Error estimates

In this section, we derive error estimates for the numerical error $z - Z_h$ in the $L^\infty(0, T; L^2(\Omega))$ and $L^2(0, T; H^1(\mathcal{E}_h))$ norms. We first define the elliptic projection $\tilde{z}$ of the exact solution z:

$$\forall t \geq 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(z(t) - \tilde{z}(t), v) = 0. \tag{3.20}$$

From the analysis of elliptic problems described in Chapter 2, we know that if z belongs to $L^2(0, T; H^s(\mathcal{E}_h))$ for $s > 3/2$, the following error estimate holds:

$$\forall t \geq 0, \quad \|z(t) - \tilde{z}(t)\|_{\mathcal{E}} \leq Ch^{\min(k+1,s)-1}|||z(t)|||_{H^s(\mathcal{E}_h)}. \tag{3.21}$$

In addition, if Ω is convex, error estimates in L^2 norm are

$$\forall t \geq 0, \quad \|z(t) - \tilde{z}(t)\|_{L^2(\Omega)} \leq Ch^{\min(k+1,s)}|||z(t)|||_{H^s(\mathcal{E}_h)} \qquad \text{for SIPG,} \tag{3.22}$$
$$\forall t \geq 0, \quad \|z(t) - \tilde{z}(t)\|_{L^2(\Omega)} \leq Ch^{\min(k+1,s)-1}|||z(t)|||_{H^s(\mathcal{E}_h)} \quad \text{for NIPG and IIPG.} \tag{3.23}$$

Under some conditions such as superpenalization ($\beta_0 \geq 3(d-1)^{-1}$), the estimates in L^2 norm are optimal for NIPG and IIPG.

What can we say about the time derivatives of $z - \tilde{z}$? Using linearity of the bilinear form, we have that

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon\left(\frac{d}{dt}(z - \tilde{z})(t), v\right) = 0.$$

The time derivative of the elliptic projection is the elliptic projection of the time derivative. Thus, similar error estimates are valid.

Now, we can state and prove a priori error estimates.

Theorem 3.4. *Assume that z belongs to $H^1(0,T;H^s(\mathcal{E}_h))$ and that z_0 belongs to $H^s(\mathcal{E}_h)$ for $s > 3/2$. Assume that $\beta_0(d-1) \geq 1$. In the case of SIPG and IIPG, assume that σ_e^0 is sufficiently large for all e. Then, there is a constant C independent of h such that*

$$\left(\int_0^T \|z(t) - Z_h(t)\|_{\mathcal{E}}^2\right)^{1/2} \leq Ch^{\min(k+1,s)-1}\|z\|_{H^1(0,T;H^s(\mathcal{E}_h))},$$
$$\|z - Z_h\|_{L^\infty(L^2(\Omega))} \leq Ch^{\min(k+1,s)-\delta}\|z\|_{H^1(0,T;H^s(\mathcal{E}_h))},$$

where $\delta = 0$ for SIPG, and $\delta = 0$ for NIPG and IIPG if $\beta_0 \geq 3(d-1)^{-1}$, if the mesh consists only of triangles and tetrahedra, and if $g_{\mathrm{D}} \in \mathcal{D}_k(\mathcal{E}_h)$. Otherwise, $\delta = 1$ for NIPG and IIPG.

Proof. Since the scheme is consistent, we obtain the following orthogonality equation:

$$\forall t > 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{\partial(Z_h - z)}{\partial t}, v\right)_\Omega + a_\epsilon(Z_h(t) - z(t), v) = 0.$$

Defining $\chi = Z_h - \tilde{z}$, we have for all $t > 0$ and for all $v \in \mathcal{D}_k(\mathcal{E}_h)$

$$\left(\frac{\partial \chi}{\partial t}, v\right)_\Omega + a_\epsilon(\chi(t), v) = \left(\frac{\partial(z - \tilde{z})}{\partial t}, v\right)_\Omega + a_\epsilon(z(t) - \tilde{z}(t), v). \tag{3.24}$$

Using the definition of the elliptic projection, we obtain

$$\left(\frac{\partial \chi}{\partial t}, v\right)_\Omega + a_\epsilon(\chi(t), v) = \left(\frac{\partial(z - \tilde{z})}{\partial t}, v\right)_\Omega. \tag{3.25}$$

Choosing $v = \chi(t)$ and using the coercivity of a_ϵ and the definition of the elliptic projection, we have

$$\forall t > 0, \quad \frac{1}{2}\frac{d}{dt}\|\chi\|_{L^2(\Omega)}^2 + \kappa\|\chi(t)\|_{\mathcal{E}}^2 \leq \left|\left(\frac{\partial(z - \tilde{z})}{\partial t}, \chi(t)\right)_\Omega\right|.$$

As in the proof of the stability bound, we can use either Gronwall's inequality or Poincaré's inequality to obtain the final estimate. If the penalty parameters σ_e^0 are positive for all e, we can bound the right-hand side of the equation above as

$$\left|\left(\frac{\partial(z - \tilde{z})}{\partial t}, \chi(t)\right)_\Omega\right| \leq \left\|\frac{\partial(z - \tilde{z})}{\partial t}\right\|_{L^2(\Omega)} \|\chi(t)\|_{L^2(\Omega)}$$
$$\leq \frac{\kappa}{2}\|\chi(t)\|_{\mathcal{E}}^2 + \frac{1}{2\kappa}\left\|\frac{\partial(z - \tilde{z})}{\partial t}\right\|_{L^2(\Omega)}^2.$$

Therefore, using the error estimates satisfied by the elliptic projection, we obtain

$$\frac{1}{2}\frac{d}{dt}\|\chi\|^2_{L^2(\Omega)} + \frac{\kappa}{2}\|\chi(t)\|^2_{\mathcal{E}} \leq Ch^{2\min(k+1,s)-2\delta}\left|\left|\left|\frac{\partial z}{\partial t}\right|\right|\right|^2_{H^s(\mathcal{E}_h)}. \tag{3.26}$$

The parameter δ is zero unconditionally for the SIPG method, and it is in general equal to one for both IIPG and NIPG. Under certain conditions given in Theorem 2.14, δ is zero for IIPG and NIPG. Next, we multiply (3.26) by 2 and integrate from 0 to t:

$$\|\chi(t)\|^2_{L^2(\Omega)} + \kappa\int_0^t \|\chi(\tau)\|^2_{\mathcal{E}} \leq \|\chi(0)\|^2_{L^2(\Omega)} + Ch^{2\min(k+1,s)-2\delta}\left|\left|\left|\frac{\partial z}{\partial t}\right|\right|\right|^2_{L^2(0,T;H^s(\mathcal{E}_h))}.$$

We then conclude by noting that $\chi(0) = 0$ and by using the triangle inequalities in the L^2 norm

$$\|z(t) - Z_h(t)\|_{L^2(\Omega)} \leq \|\chi(t)\|_{L^2(\Omega)} + \|z(t) - \tilde{z}(t)\|_{L^2(\Omega)},$$

the triangle inequalities in the energy norm

$$\left(\int_0^T \|z(t) - Z_h(t)\|^2_{\mathcal{E}}\right)^{1/2} \leq \left(\int_0^T \|z(t) - \tilde{z}(t)\|^2_{\mathcal{E}}\right)^{1/2} + \left(\int_0^T \|\tilde{z}(t) - Z_h(t)\|^2_{\mathcal{E}}\right)^{1/2},$$

and the error estimates satisfied by $\tilde{z}$. □

The following result is obtained only in the case of the SIPG method, as the proof heavily uses the symmetry of the bilinear form.

Theorem 3.5. *Let $\epsilon = -1$. Under the assumptions of Theorem* 3.4*, there exists a constant C independent of h such that*

$$\left\|\frac{\partial(z - Z_h)}{\partial t}\right\|_{L^2(0,t;L^2(\Omega))} \leq Ch^{\min(k+1,s)}\|z\|_{H^1(0,T;H^s(\mathcal{E}_h))}.$$

Proof. In the error equation (3.25), we choose $v = \frac{\partial\chi}{\partial t}$:

$$\left\|\frac{\partial\chi}{\partial t}\right\|^2_{L^2(\Omega)} + a_\epsilon\left(\chi(t), \frac{\partial\chi}{\partial t}\right) = \left(\frac{\partial(z - \tilde{z})}{\partial t}, \frac{\partial\chi}{\partial t}\right)_\Omega.$$

Thus, using the symmetry property of a_ϵ, we have

$$\begin{aligned}\left\|\frac{\partial\chi}{\partial t}\right\|^2_{L^2(\Omega)} + \frac{1}{2}\frac{d}{dt}a_\epsilon(\chi(t), \chi(t)) &= \left(\frac{\partial(z - \tilde{z})}{\partial t}, \frac{\partial\chi}{\partial t}\right)_\Omega \\ &\leq \frac{1}{2}\left\|\frac{\partial(z - \tilde{z})}{\partial t}\right\|^2_{L^2(\Omega)} + \frac{1}{2}\left\|\frac{\partial\chi}{\partial t}\right\|^2_{L^2(\Omega)}.\end{aligned}$$

Integrating from 0 to t and using the fact that $\chi(0) = 0$, we obtain

$$\int_0^t \left\| \frac{\partial \chi}{\partial t} \right\|^2_{L^2(\Omega)} + \frac{1}{2} a_\epsilon(\chi(t), \chi(t)) \leq \frac{1}{2} a_\epsilon(\chi(0), \chi(0)) + \frac{1}{2} \int_0^t \left\| \frac{\partial (z - \tilde{z})}{\partial t} \right\|^2_{L^2(\Omega)} \leq C h^{2\min(k+1,s)} \left\| \frac{\partial z}{\partial t} \right\|^2_{L^2(0,T;H^s(\mathcal{E}_h))}.$$

Using coercivity of a_ϵ and the triangle inequality, we have

$$\left\| \frac{\partial (z - Z_h)}{\partial t} \right\|_{L^2(0,T;L^2(\Omega))} \leq \left\| \frac{\partial (z - \tilde{z})}{\partial t} \right\|_{L^2(0,T;L^2(\Omega))} + \left\| \frac{\partial \chi}{\partial t} \right\|_{L^2(0,T;L^2(\Omega))} \leq C h^{\min(k+1,s)} \left\| \frac{\partial z}{\partial t} \right\|_{L^2(0,T;H^s(\Omega))}.$$

This concludes the proof. □

3.4 Fully discrete formulation

We now discretize the time derivative by using finite differences in time. More precisely, we study some time stepping methods that are of low order such as backward Euler and forward Euler and some that are of high order such as Crank–Nicolson and Runge–Kutta methods. Finally, we also present the DG in time method.

Let N_T be a positive integer and let $\Delta t = T/N_T$ denote the time step. We also use the following notation for any function $u = u(t, \boldsymbol{x})$:

$$\forall n \geq 0, \quad t^n = n\Delta t, \quad u^n(\boldsymbol{x}) = u(t^n)(\boldsymbol{x}) = u(t^n, \boldsymbol{x}).$$

3.4.1 Backward Euler discretization

The fully discrete variational formulation is as follows: Find a sequence $(Z_h^n)_{n\geq 0}$ of functions in $\mathcal{D}_k(\mathcal{E}_h)$ such that $Z_h^0 = \tilde{z}_0$ and

$$\forall n \geq 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{Z_h^{n+1} - Z_h^n}{\Delta t}, v \right)_\Omega + a_\epsilon(Z_h^{n+1}, v) = L(t^{n+1}; v). \tag{3.27}$$

We expand the fully discrete solution Z_h^n using the basis functions of $\mathcal{D}_k(\mathcal{E}_h)$

$$\forall n \geq 0, \quad Z_h^n = \sum_{j=1}^{N_{\mathrm{loc}} N_{\mathrm{el}}} \tilde{\xi}_j^n \tilde{\phi}_j.$$

Problem (3.27) is then equivalent to a linear system with vector of unknowns $\tilde{\boldsymbol{\xi}}^n = (\tilde{\xi}_i^n)_i$

$$(\boldsymbol{M} + \Delta t \boldsymbol{A}) \tilde{\boldsymbol{\xi}}^{n+1} = \boldsymbol{M} \tilde{\boldsymbol{\xi}}^n + \Delta t \boldsymbol{F}^{n+1}.$$

The matrices $\boldsymbol{M}$, $\boldsymbol{A}$ are defined in Section 3.3. The vector $\boldsymbol{F}^{n+1}$ has components $(L(t^{n+1}, \tilde{\phi}_i))_i$. The coercivity of a_ϵ implies that $\boldsymbol{M} + \Delta t \boldsymbol{A}$ is positive definite and in particular invertible.

Thus, there exists a unique solution Z_h^n at each time step. Problem (3.27) is called *implicit in time*.

Next, we state and prove a priori bounds and a priori error estimates.

Lemma 3.6. *There exists a constant C independent of h and Δt such that for all $m > 0$*

$$\|Z_h^m\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{N_T} \|Z_h^n\|_{\mathcal{E}}^2 \le C \left(\|\tilde{z}_0\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{N_T} \|f^n\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{N_T} \sum_{e \in \partial\Omega} \frac{1}{|e|^{\beta_0}} \|g_D^n\|_{L^2(e)}^2 \right).$$

If $\sigma_e^0 = 0$ for some face e, then this bound holds true if $\Delta t < 1$.

Proof. Take $v = Z_h^{n+1}$ in (3.27) and use coercivity of a_ϵ:

$$\frac{1}{\Delta t}(Z_h^{n+1} - Z_h^n, Z_h^{n+1})_\Omega + \kappa \|Z_h^{n+1}\|_{\mathcal{E}}^2 \le |L(t^{n+1}; Z_h^{n+1})|.$$

Next, we observe the simple fact that

$$\forall x, y \in \mathbb{R}, \quad \frac{1}{2}(x^2 - y^2) \le \frac{1}{2}(x^2 - y^2 + (x-y)^2) = (x-y)x. \tag{3.28}$$

Therefore, we obtain

$$\frac{1}{2\Delta t}(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2) + \kappa \|Z_h^{n+1}\|_{\mathcal{E}}^2 \le |L(t^{n+1}; Z_h^{n+1})|.$$

The right-hand side is bounded by (3.16):

$$\frac{1}{2\Delta t}(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2) + \frac{\kappa}{2}\|Z_h^{n+1}\|_{\mathcal{E}}^2 \le \|f^{n+1}\|_{L^2(\Omega)} \|Z_h^{n+1}\|_{L^2(\Omega)} + C \sum_{e \in \partial\Omega} \frac{1}{|e|^{\beta_0}} \|g_D(t^{n+1})\|_{L^2(e)}^2. \tag{3.29}$$

As with the semidiscrete stability bounds, we can proceed with two approaches, and we choose to use Gronwall's inequality. This approach is valid for all methods but imposes a constraint on the time step. The reader can prove the estimates by using Poincaré's inequality (3.5), which is valid for any time step and if the penalty values are nonzero. We multiply (3.29) by $2\Delta t$ and sum from $n = 0$ to $n = m - 1$:

$$\|Z_h^m\|_{L^2(\Omega)}^2 - \|Z_h^0\|_{L^2(\Omega)}^2 + \kappa \Delta t \sum_{n=1}^{m} \|Z_h^n\|_{\mathcal{E}}^2 \le 2\Delta t \sum_{n=1}^{m} \|f^n\|_{L^2(\Omega)} \|Z_h^n\|_{L^2(\Omega)} + C\Delta t \sum_{n=1}^{m} \sum_{e \in \partial\Omega} \frac{1}{|e|^{\beta_0}} \|g_D(t^n)\|_{L^2(e)}^2.$$

Equivalently, we have

$$\forall m \geq 1, \quad \|Z_h^m\|_{L^2(\Omega)}^2 + \kappa \Delta t \sum_{n=1}^{m} \|Z_h^n\|_{\mathcal{E}}^2 \leq \|\tilde{z}_0\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{m} \|Z_h^n\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{m} \|f^n\|_{L^2(\Omega)}^2 + C \Delta t \sum_{n=1}^{m} \sum_{e \in \partial\Omega} \frac{1}{|e|^{\beta_0}} \|g_{\mathrm{D}}(t^n)\|_{L^2(e)}^2.$$

The final result is obtained by using Lemma 3.2. □

The next theorem states that the numerical error is first order in time.

Theorem 3.7. *For $s > 3/2$, assume that the exact solution to problem* (3.7)–(3.9) *satisfies*

$$z \in H^1(0, T; H^s(\mathcal{E}_h)), \quad \frac{\partial^2 z}{\partial t^2} \in L^2(0, T; L^2(\Omega)).$$

There exists a constant C independent of h and Δt such that for all $m > 0$

$$\|Z_h^m - z^m\|_{L^2(\Omega)} \leq C h^{\min(k+1,s)-\delta} \left\| \frac{\partial z}{\partial t} \right\|_{H^1(0,T;H^s(\mathcal{E}_h))} + C \Delta t \left\| \frac{\partial^2 z}{\partial t^2} \right\|_{L^2(0,T;L^2(\Omega))},$$

$$\left(\Delta t \sum_{n=1}^{m} \|Z_h^n - z^n\|_{\mathcal{E}}^2 \right)^{1/2} \leq C h^{\min(k+1,s)-1} \left\| \frac{\partial z}{\partial t} \right\|_{H^1(0,T;H^s(\mathcal{E}_h))} + C \Delta t \left\| \frac{\partial^2 z}{\partial t^2} \right\|_{L^2(0,T;L^2(\Omega))}.$$

The definition of the power δ is given in Theorem 3.4.

Proof. Let $\tilde{z}$ be the elliptic projection of z defined by (3.20). We use the notation $z^n = z(t^n)$ and $\tilde{z}^n = \tilde{z}(t^n)$. Subtracting (3.10) from (3.27) and writing $z^n - Z_h^n = \rho^n - \chi^n$ with $\chi^n = Z_h^n - \tilde{z}^n$ and $\rho^n = z^n - \tilde{z}^n$, we obtain

$$\left(\frac{\chi^{n+1} - \chi^n}{\Delta t}, v \right)_\Omega + a_\epsilon(\chi^{n+1}, v) = \left(\frac{\partial z}{\partial t}^{n+1} - \frac{z^{n+1} - z^n}{\Delta t}, v \right)_\Omega + \left(\frac{\rho^{n+1} - \rho^n}{\Delta t}, v \right)_\Omega.$$

Now, the coercivity of a_ϵ and the choice $v = \chi^{n+1}$ yield

$$\frac{1}{2\Delta t}(\|\chi^{n+1}\|_{L^2(\Omega)}^2 - \|\chi^n\|_{L^2(\Omega)}^2) + \kappa \|\chi^{n+1}\|_{\mathcal{E}}^2 \leq \left| (\theta^{n+1}, \chi^{n+1})_\Omega + \left(\frac{\rho^{n+1} - \rho^n}{\Delta t}, \chi^{n+1} \right)_\Omega \right|$$

with the definition

$$\theta^{n+1} = \frac{\partial z}{\partial t}^{n+1} - \frac{z^{n+1} - z^n}{\Delta t}.$$

We use Cauchy–Schwarz's and Poincaré's inequalities:

$$\begin{aligned}\frac{1}{2\Delta t}(\|\chi^{n+1}\|^2_{L^2(\Omega)} - \|\chi^n\|^2_{L^2(\Omega)}) + \kappa\|\chi^{n+1}\|^2_{\mathcal{E}} \\ \leq C\|\chi^{n+1}\|_{\mathcal{E}}\left(\|\theta^{n+1}\|_{L^2(\Omega)} + \left\|\frac{\rho^{n+1} - \rho^n}{\Delta t}\right\|_{L^2(\Omega)}\right).\end{aligned}$$

With Young's inequality, this reduces to

$$\begin{aligned}\frac{1}{2\Delta t}(\|\chi^{n+1}\|^2_{L^2(\Omega)} - \|\chi^n\|^2_{L^2(\Omega)}) + \frac{\kappa}{2}\|\chi^{n+1}\|^2_{\mathcal{E}} \\ \leq C\left(\|\theta^{n+1}\|^2_{L^2(\Omega)} + \left\|\frac{\rho^{n+1} - \rho^n}{\Delta t}\right\|^2_{L^2(\Omega)}\right).\end{aligned}$$

Taylor expansion (3.2) gives

$$\theta^{n+1} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} (t - t_n)\frac{\partial^2 z}{\partial t^2}dt, \quad \rho^{n+1} - \rho^n = \int_{t^n}^{t^{n+1}} \frac{\partial \rho}{\partial t}dt.$$

Using Cauchy–Schwarz's inequality, we can prove that

$$\|\theta^{n+1}\|^2_{L^2(\Omega)} \leq \frac{\Delta t}{3}\int_{t^n}^{t^{n+1}} \left\|\frac{\partial^2 z}{\partial t^2}\right\|^2_{L^2(\Omega)} dt, \tag{3.30}$$

$$\|\rho^{n+1} - \rho^n\|^2_{L^2(\Omega)} \leq \Delta t\int_{t^n}^{t^{n+1}} \left\|\frac{\partial \rho}{\partial t}\right\|^2_{L^2(\Omega)} dt. \tag{3.31}$$

Combining the bounds above, multiplying the resulting inequality by $2\Delta t$ and summing from $n = 0$ to $n = m - 1$, we have

$$\begin{aligned}\|\chi^m\|^2_{L^2(\Omega)} - \|\chi^0\|^2_{L^2(\Omega)} + \kappa\Delta t\sum_{n=1}^{m}\|\chi^n\|^2_{0,\mathcal{E}} \leq C\Delta t^2\int_0^T \left\|\frac{\partial^2 z}{\partial t^2}\right\|^2_{L^2(\Omega)} dt \\ + C\frac{1}{\Delta t}\int_0^T \left\|\frac{\partial \rho}{\partial t}\right\|^2_{L^2(\Omega)} dt.\end{aligned}$$

Thus, since χ^0 is zero by definition and since $\frac{\partial \rho}{\partial t}$ satisfies error estimates of the type (3.22)–(3.23), we have

$$\begin{aligned}\forall m \geq 0, \quad \|\chi^m\|^2_{L^2(\Omega)} + \kappa\Delta t\sum_{n=1}^{m}\|\chi^n\|^2_{\mathcal{E}} \leq C\Delta t^2\int_0^T \left\|\frac{\partial^2 z}{\partial t^2}\right\|^2_{L^2(\Omega)} \\ + Ch^{2\min(k+1,s)-2\delta}\int_0^T \left\|\frac{\partial z}{\partial t}\right\|^2_{H^s(\mathcal{E}_h)}.\end{aligned}$$

The power δ is equal to zero for SIPG and equal to one in general for NIPG and IIPG. Using the triangle inequality, we conclude that for any $m \geq 1$

$$\begin{aligned} \|Z_h^m - z^m\|_{L^2(\Omega)} &\leq Ch^{\min(k+1,s)-\delta} \left\| \frac{\partial z}{\partial t} \right\|_{H^1(0,T;H^s(\mathcal{E}_h))} \\ &\quad + C\Delta t \left\| \frac{\partial^2 z}{\partial t^2} \right\|_{L^2(0,T;L^2(\Omega))}, \\ \left(\Delta t \sum_{n=1}^{m} \|Z_h^n - z^n\|_{\mathcal{E}}^2 \right)^{1/2} &\leq Ch^{\min(k+1,s)-1} \left\| \frac{\partial z}{\partial t} \right\|_{H^1(0,T;H^s(\mathcal{E}_h))} \\ &\quad + C\Delta t \left\| \frac{\partial^2 z}{\partial t^2} \right\|_{L^2(0,T;L^2(\Omega))}. \end{aligned}$$

This concludes the proof. □

3.4.2 Forward Euler discretization

The fully discrete variational formulation is as follows: Find a sequence $(Z_h^n)_{n\geq 0}$ of functions in $\mathcal{D}_k(\mathcal{E}_h)$ such that $Z_h^0 = \tilde{z}_0$ and

$$\forall n \geq 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{Z_h^{n+1} - Z_h^n}{\Delta t}, v \right)_\Omega + a_\epsilon(Z_h^n, v) = L(t^n; v). \tag{3.32}$$

Using notation defined in Section 3.4.1, we can write the resulting linear system as

$$\boldsymbol{M}\tilde{\boldsymbol{\xi}}^{n+1} = (\boldsymbol{M} + \Delta t \boldsymbol{A})\tilde{\boldsymbol{\xi}}^n + \Delta t \boldsymbol{F}^n.$$

The vector $\boldsymbol{F}^n$ has components $(L(t^n, \tilde{\phi}_i))_i$. Since $\boldsymbol{M}$ is block diagonal and invertible, there exists a solution $\tilde{\boldsymbol{\xi}}^{n+1}$ that can be computed locally on each element. Thus, there exists a unique solution Z_h^n at each time step. Problem (3.27) is called *explicit in time*. It is quite popular because of its low computational cost. However, stability properties of forward Euler discretization are very poor.

We show that very strict constraints on the time step are required for obtaining a priori bounds and a priori error estimates. In this section, we assume that the mesh is quasi-uniform: there is a constant $\tau > 0$ such that $\frac{h}{h_E} \leq \tau$ for all mesh elements E.

Lemma 3.8. *There is a constant C_b independent of h (but depending on the quasi-uniformity constant τ) such that*

$$\forall v, w \in \mathcal{D}_k(\mathcal{E}_h), \quad a_\epsilon(v, w) \leq C_b h^{-1} \|v\|_{L^2(\Omega)} \|w\|_{\mathcal{E}}.$$

Proof. The proof uses the continuity of a_ϵ in $\mathcal{D}_k(\mathcal{E}_h)$ with respect to $\|\cdot\|_{\mathcal{E}}$, the trace inequality (2.3), and the inverse inequality (3.6). □

Theorem 3.9. *Assume that Δt is of the order of h^{-2}, namely*

$$\Delta t < \frac{\kappa}{C_b^2} h^2.$$

Then, there exists a constant C independent of h and Δt such that for all $m > 0$

$$\begin{aligned} &\|Z_h^m\|_{L^2(\Omega)}^2 + \Delta t \sum_{n=1}^{m} \|Z_h^n\|_{\mathcal{E}}^2 \\ &\leq C\|\tilde{z}_0\|_{L^2(\Omega)}^2 + C\Delta t \sum_{n=0}^{m-1} \|f^n\|_{L^2(\Omega)}^2 + Ch^{-3}\Delta t \sum_{n=1}^{m-1} \sum_{e\in\partial\Omega} \|g_{\mathrm{D}}^n\|_{L^2(e)}^2. \end{aligned} \tag{3.33}$$

Proof. Using (3.28) and choosing $v = Z_h^{n+1}$ in (3.32) gives

$$\frac{1}{2\Delta t}\Big(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2 + \|Z_h^{n+1} - Z_h^n\|_{L^2(\Omega)}^2\Big) + a_\epsilon(Z_h^n, Z_h^{n+1}) = L(t^n; Z_h^{n+1});$$

equivalently,

$$\begin{aligned} \frac{1}{2\Delta t}\Big(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2 + \|Z_h^{n+1} - Z_h^n\|_{L^2(\Omega)}^2\Big) + a_\epsilon(Z_h^{n+1}, Z_h^{n+1}) \\ = a_\epsilon(Z_h^{n+1} - Z_h^n, Z_h^{n+1}) + L(t^n; Z_h^{n+1}). \end{aligned}$$

Using the coercivity of a_ϵ and Lemma 3.8, we obtain

$$\begin{aligned} \frac{1}{2\Delta t}\Big(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2 + \|Z_h^{n+1} - Z_h^n\|_{L^2(\Omega)}^2\Big) + \kappa\|Z_h^{n+1}\|_{\mathcal{E}}^2 \\ \leq C_b h^{-1}\|Z_h^{n+1} - Z_h^n\|_{L^2(\Omega)}\|Z_h^{n+1}\|_{\mathcal{E}} + |L(t^n; Z_h^{n+1})|. \end{aligned}$$

The term $L(t^n; Z_h^{n+1})$ is bounded similarly as in (3.16):

$$\begin{aligned} |L(t^n; Z_h^{n+1})| &\leq \|f^n\|_{L^2(\Omega)}\|Z_h^{n+1}\|_{L^2(\Omega)} + Ch^{-1/2}\|Z_h^{n+1}\|_{\mathcal{E}}\left(\sum_{e\in\partial\Omega}\|g_{\mathrm{D}}^n\|_{L^2(e)}^2\right)^{1/2} \\ &\leq \frac{1}{2}\|f^n\|_{L^2(\Omega)}^2 + \frac{1}{2}\|Z_h^{n+1}\|_{L^2(\Omega)}^2 + Ch^{-3/2}\|Z_h^{n+1}\|_{L^2(\Omega)}\left(\sum_{e\in\partial\Omega}\|g_{\mathrm{D}}^n\|_{L^2(e)}^2\right)^{1/2} \\ &\leq \frac{1}{2}\|f^n\|_{L^2(\Omega)}^2 + \|Z_h^{n+1}\|_{L^2(\Omega)}^2 + Ch^{-3}\sum_{e\in\partial\Omega}\|g_{\mathrm{D}}^n\|_{L^2(e)}^2. \end{aligned}$$

Combining the bounds above gives

$$\begin{aligned} \frac{1}{2\Delta t}\Big(\|Z_h^{n+1}\|_{L^2(\Omega)}^2 - \|Z_h^n\|_{L^2(\Omega)}^2\Big) + \left(\frac{1}{2\Delta t} - \frac{C_b^2 h^{-2}}{2\kappa}\right)\|Z_h^{n+1} - Z_h^n\|_{L^2(\Omega)}^2 + \frac{\kappa}{2}\|Z_h^{n+1}\|_{\mathcal{E}}^2 \\ \leq \frac{1}{2}\|f^n\|_{L^2(\Omega)}^2 + \|Z_h^{n+1}\|_{L^2(\Omega)}^2 + Ch^{-3}\sum_{e\in\partial\Omega}\|g_{\mathrm{D}}^n\|_{L^2(e)}^2. \end{aligned}$$

Therefore, if $\frac{C_b^2 h^{-2}}{\kappa}\Delta t < 1$, we can conclude by using, for instance, Gronwall's inequality. We skip the details. □

Remark: As in the semidiscrete case (see Section 3.3.1), the a priori bounds for both backward Euler and forward Euler discretizations do not imply stability of the solution with respect to h. As the mesh size tends to zero, the constant in the a priori bounds blows up even faster for the forward Euler method. True stability bounds are obtained if the Dirichlet boundary condition is imposed strongly in the discrete space.

Next, we obtain error estimates in a similar fashion as the a priori bounds and the error estimates for backward Euler scheme. We skip the proof of the theorem.

Theorem 3.10. *For $s > 3/2$, assume that the exact solution to problem* (3.7)–(3.9) *satisfies*

$$z \in H^1(0,T;H^s(\mathcal{E}_h)), \quad \frac{\partial^2 z}{\partial t^2} \in L^2(0,T;L^2(\Omega)).$$

In addition, assume that Δt is of the same order of h^2. Then, there exists a constant C independent of h and Δt such that for all $m > 0$

$$\begin{aligned}\|Z_h^m - z^m\|_{L^2(\Omega)} &\le Ch^{\min(k+1,s)-\delta}\left\|\frac{\partial z}{\partial t}\right\|_{H^1(0,T;H^s(\mathcal{E}_h))} \\ &\quad + C\Delta t\left\|\frac{\partial^2 z}{\partial t^2}\right\|_{L^2(0,T;L^2(\Omega))}, \\ \left(\Delta t\sum_{n=1}^{m}\|Z_h^n - z^n\|_{\mathcal{E}}^2\right)^{1/2} &\le Ch^{\min(k+1,s)-1}\left\|\frac{\partial z}{\partial t}\right\|_{H^1(0,T;H^s(\mathcal{E}_h))} \\ &\quad + C\Delta t\left\|\frac{\partial^2 z}{\partial t^2}\right\|_{L^2(0,T;L^2(\Omega))}.\end{aligned}$$

The definition of the power δ is given in Theorem 3.4.

A consequence of the convergence theorem is a stability bound for the numerical solution.

3.4.3 Crank–Nicolson discretization

We introduce the notation

$$t^{n+1/2} = \frac{t^{n+1} + t^n}{2}.$$

The fully discrete variational formulation is as follows: Find a sequence $(Z_h^n)_{n\ge 0}$ of functions in $\mathcal{D}_k(\mathcal{E}_h)$ such that $Z_h^0 = \tilde{z}_0$ and

$$\begin{aligned}&\forall n \ge 0, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{Z_h^{n+1} - Z_h^n}{\Delta t}, v\right) + a_\epsilon\left(\frac{Z_h^{n+1} + Z_h^n}{2}, v\right) \\ &\quad = L(t^{n+1/2}; v).\end{aligned} \tag{3.34}$$

The resulting linear system is

$$\left(\boldsymbol{M} + \frac{\Delta t}{2}\boldsymbol{A}\right)\tilde{\boldsymbol{\xi}}^{n+1} = \left(\boldsymbol{M} - \frac{\Delta t}{2}\boldsymbol{A}\right)\tilde{\boldsymbol{\xi}}^{n} + \Delta t\, \boldsymbol{F}^{n+1/2}$$

with $\boldsymbol{F}^{n+1/2} = (L(t^{n+1/2}; \tilde{\phi}_j))_j$. Since the matrix $\boldsymbol{M} + \frac{\Delta t}{2}\boldsymbol{A}$ is invertible, there is a unique solution Z_h^n for all n. Without going into details, we state that the method (3.34) is of second order in time and under some smoothness assumptions for the exact solution, one can prove the following error bounds:

$$\|Z_h^n - z^n\|_{L^2(\Omega)} = \mathcal{O}(h^{\min(k+1,s)-\delta} + \Delta t^2), \tag{3.35}$$

$$\forall m > 0, \quad \left(\Delta t \sum_{n=0}^{m} \|Z_h^n - z^n\|_{\mathcal{E}}^2\right)^{1/2} = \mathcal{O}(h^{\min(k+1,s)-1} + \Delta t^2). \tag{3.36}$$

The definition of the power δ is given in Theorem 3.4.

3.4.4 Runge–Kutta discretization

DG methods are well suited for high order approximation in space. It is natural to combine them with high order time discretization such as the Runge–Kutta methods.

The fully discrete variational formulation is as follows: Find a sequence $(Z_h^n)_{n\geq 0}$ of functions in $\mathcal{D}_k(\mathcal{E}_h)$ such that $Z_h^0 = \tilde{z}_0$ and

$$\begin{aligned}
&\forall 1 \leq i \leq S, \quad \forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{Y_i - Z_h^n}{\Delta t}, v\right)_\Omega + \sum_{j=1}^{S} \check{a}_{ij} a_\epsilon(Y_j, v) \\
&\quad = \sum_{j=1}^{S} \check{a}_{ij} L(t_n + \check{c}_j \Delta t; v), \\
&\forall v \in \mathcal{D}_k(\mathcal{E}_h), \quad \left(\frac{Z_h^{n+1} - Z_h^n}{\Delta t}, v\right)_\Omega + \sum_{j=1}^{S} \check{b}_j a_\epsilon(Y_j, v) \\
&\quad = \sum_{j=1}^{S} \check{b}_j L(t_n + c_j \Delta t; v).
\end{aligned}$$

In order to obtain the solution at the next time step, S additional values Y_i's are computed; they are intermediate approximations to the solution z at times $t_n + \check{c}_j \Delta t$. The S-state Runge–Kutta method is uniquely defined by the coefficients $(\check{a}_{ij})_{1\leq i,j\leq S}$, $(\check{b}_i)_{1\leq i\leq S}$, and $(\check{c}_i)_{1\leq i\leq S}$. These coefficients are carefully chosen so that the desired accuracy in time is reached [5]. In addition, we have

$$\forall 1 \leq i \leq S, \quad \check{c}_i = \sum_{j=1}^{S} \check{a}_{ij}.$$

The compact way of writing out the coefficients of the Runge–Kutta method is

$$\begin{array}{c|cccc} \check{c}_1 & \check{a}_{11} & \check{a}_{12} & \dots & \check{a}_{1S} \\ \check{c}_2 & \check{a}_{21} & \check{a}_{22} & \dots & \check{a}_{2S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \check{c}_S & \check{a}_{S1} & \check{a}_{S2} & \dots & \check{a}_{SS} \\ \hline & \check{b}_1 & \check{b}_2 & \dots & \check{b}_S \end{array}$$

If $\check{a}_{ij} = 0$ for all $i \leq j$, the method is explicit in time; otherwise it is implicit in time. The order in time increases with the number S but not in a linear fashion. Indeed, if q is the order in time, we have the following relationship between S and q for explicit Runge–Kutta methods [21]:

S	1	2	3	4	5	6	7	8	9	10
q	1	2	3	4	4	5	6	6	7	7

Since the number of stages also characterizes the computational cost of the method, the relationship above clearly shows that the explicit Runge–Kutta of order four optimizes the amount of work with respect to the gain in accuracy. It is a popular method, defined by the following coefficients:

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

Some Runge–Kutta methods such as the strong-stability-preserving Runge–Kutta methods possess attractive stability properties [64].

3.4.5 DG in time discretization

All the fully discrete schemes presented above employ finite differences to approximate the time derivative. In this section, we apply a DG method both in time and space by considering test functions that vary in time and space. Denote the interval $I_n = (t_n, t_{n+1})$ and define the discrete space:

$$W_{\Delta t,h} = \{v : (0,T) \mapsto \mathcal{D}_k(\mathcal{E}_h) : \forall 0 \leq n \leq (N_{\mathrm{T}} - 1),\, v|_{I_n} \in \mathcal{P}_q(I_n)\} \tag{3.37}$$

with

$$\mathcal{P}_q(I_n) = \left\{ v : I_n \mapsto \mathcal{D}_k(\mathcal{E}_h) : \forall t \in I_n,\, v(t) = \sum_{j=0}^{q} v_{j,n} t^j,\ v_{j,n} \in \mathcal{D}_k(\mathcal{E}_h) \right\}. \tag{3.38}$$

Clearly, the functions in $W_{\Delta t,h}$ are discontinuous both in time and space. The discontinuity points in time are the endpoints of the intervals I_n, and the discontinuity points in space are the boundaries of the mesh elements. We need to introduce the jump of a function at time t^n, which we denote by $[\cdot]_t$ to differentiate from the jump in space defined in Section 2.3.1:

$$v_+^n = \lim_{s \to 0^+} v(t_n + s), \quad v_-^n = \lim_{s \to 0^+} v(t_n - s), \quad [v^n]_t = v_+^n - v_-^n.$$

The fully discrete variational formulation is as follows: Find $Z_h \in W_{\Delta t,h}$ such that

$$\forall v \in W_{\Delta t,h}, \quad \sum_{n=0}^{N_T-1} \int_{I_n} \left(\left(\frac{\partial Z_h}{\partial t}, v \right)_\Omega + a_\epsilon(Z_h, v) \right) dt$$
$$+ \sum_{n=1}^{N_T-1} ([Z_h^n]_t, v_+^n) + (Z_{h+}^0, v_+^0)_\Omega = \sum_{n=0}^{N_T-1} \int_{I_n} L(t; v) dt + (z_0, v_+^0)_\Omega. \tag{3.39}$$

The exact solution satisfies (3.39) since $[z^n]_t = 0$, but $z(0)$ is not $\tilde{z}_0$. Since the functions in $W_{\Delta t,h}$ are not required to be continuous at the time t^n, we can fix one interval I_n and choose v to vanish outside I_n. Therefore, (3.39) is equivalent to the following system of equations:

$$\forall 0 \le n \le (N_T - 1), \quad \forall v \in \mathcal{P}_q(I_n), \quad \int_{I_n} \left(\left(\frac{\partial Z_h}{\partial t}, v \right)_\Omega + a_\epsilon(Z_h, v) \right) dt$$
$$+ (Z_{h+}^n, v_+^n)_\Omega = \int_{I_n} L(t; v) dt + (Z_{h-}^n, v_+^n)_\Omega. \tag{3.40}$$

Here, we denote $Z_{h-}^0 = \tilde{z}_0$.

Lemma 3.11. *There exists a unique solution to* (3.39).

Proof. Since problem (3.39) is linear and finite-dimensional, it suffices to prove uniqueness of the solution. We use the equivalent formulation (3.40). Let $W_h \in W_{h,\Delta t}$ denote the difference between two solutions. We show by induction on n that $W_h|_{I_n}$ is zero. First, if $n = 0$, we have

$$\forall v \in \mathcal{P}_q(I_n), \quad \int_{I_n} \left(\left(\frac{\partial W_h}{\partial t}, v \right)_\Omega + a_\epsilon(W_h, v) \right) dt + (W_{h+}^n, v_+^n)_\Omega = 0.$$

Choosing $v = W_h$ yields

$$\int_{I_n} \frac{1}{2} \left(\frac{d}{dt} \|W_h\|_{L^2(\Omega)}^2 + a_\epsilon(W_h, W_h) \right) dt + \|W_{h+}^n\|_{L^2(\Omega)}^2 = 0.$$

Thus, from the coercivity of a_ϵ

$$\frac{1}{2}(\|W_{h-}^{n+1}\|_{L^2(\Omega)}^2 - \|W_{h+}^n\|_{L^2(\Omega)}^2) + \kappa \int_{I_n} \|W_h\|_{\mathcal{E}}^2 dt + \|W_{h+}^n\|_{L^2(\Omega)}^2 = 0;$$

equivalently,

$$\frac{1}{2}(\|W_{h-}^{n+1}\|_{L^2(\Omega)}^2 + \|W_{h+}^n\|_{L^2(\Omega)}^2) + \kappa \int_{I_n} \|W_h\|_{\mathcal{E}}^2 dt = 0.$$

This implies that $W_h = 0$ on the interval I_n for $n = 0$. To finish the induction argument, we assume that $W_h = 0$ for I_{n-1} for some $n \ge 1$ and show that $W_h = 0$ on I_n by repeating the argument. □

Example 3.12. In the case $q = 0$, the solution Z_h is piecewise constant in time on each interval I_n, and in particular $\frac{\partial Z_h}{\partial t} = 0$. The scheme (3.39) becomes for all $0 \le n \le (N_{\rm T} - 1)$

$$\forall v \in \mathcal{D}_k(\mathcal{E}_h),\ \left(\frac{Z_h|_{I_n} - Z_h|_{I_{n-1}}}{\Delta t}, v\right)_\Omega + a_\epsilon(Z_h|_{I_n}, v) = \frac{1}{\Delta t}\int_{I_n} L(t; v)$$

with the notation $Z_h|_{I_{-1}} = \tilde{z}_0$. We have recovered the backward Euler time discretization with a modified right-hand side.

Example 3.13. In the case $q = 1$, the solution Z_h is piecewise linear in time on each interval. For example, one may write

$$\forall t \in I_n, \quad Z_h(t) = Z_{1;n} + \frac{t - t_n}{\Delta t} Z_{2;n}\,,$$

where the functions $Z_{1;n}$ and $Z_{2;n}$ belong to $\mathcal{D}_k(\mathcal{E}_h)$. The scheme (3.39) becomes for all $0 \le n \le (N_{\rm T} - 1)$

$$\begin{aligned}
&(Z_{1;n} + Z_{2;n}, v)_\Omega + \Delta t a_\epsilon(Z_{1;n}, v) + \frac{\Delta t}{2} a_\epsilon(Z_{2;n}, v)\\
&\quad = \int_{I_n} L(t; v)dt + (Z_{1;n-1} + Z_{2;n-1}, v)_\Omega,\\
&\frac{1}{2}(Z_{2;n}, v)_\Omega + \frac{\Delta t}{2} a_\epsilon(Z_{1;n}, v) + \frac{\Delta t}{3} a_\epsilon(Z_{2;n}, v) = \frac{1}{\Delta t}\int_{I_n} (t - t_n) L(t; v)dt
\end{aligned}$$

with the additional notation $Z_{1;-1} = \tilde{z}_0$ and $Z_{2;-1} = 0$.

Lemma 3.14. *There is a constant C independent of h such that for all $1 \le m \le N_{\rm T}$*

$$\|Z_{h-}^m\|_{L^2(\Omega)}^2 + \int_0^{t_m} \|Z_h\|_{\mathcal{E}}^2 dt \le C\left(\|\tilde{z}_0\|_{L^2(\Omega)}^2 + \|f\|_{L^2(0,T;L^2(\Omega))}^2 + \sum_{e\in\partial\Omega} \frac{1}{|e|^{\beta_0}} \|g_{\rm D}\|_{L^2(0,T;L^2(e))}^2\right).$$

Proof. Choosing $v = Z_h$ in (3.40) yields

$$\frac{1}{2}(\|Z_{h-}^{n+1}\|_{L^2(\Omega)}^2 + \|Z_{h+}^n\|_{L^2(\Omega)}^2) + \kappa \int_{I_n} \|Z_h\|_{\mathcal{E}}^2 dt \le \left| \int_{I_n} L(t; Z_h)dt + (Z_{h-}^n, Z_{h+}^n)_\Omega \right|.$$

Using Cauchy–Schwarz's inequality, we have

$$\frac{1}{2}(\|Z_{h-}^{n+1}\|_{L^2(\Omega)}^2 - \|Z_{h-}^n\|_{L^2(\Omega)}^2) + \kappa \int_{I_n} \|Z_h\|_{\mathcal{E}}^2 dt \le \int_{I_n} |L(t; Z_h)|dt.$$

Summing from $n = 0$ to $n = m$, we have

$$\|Z_{h-}^m\|_{L^2(\Omega)}^2 + 2\kappa \int_0^{t_m} \|Z_h\|_{\mathcal{E}}^2 dt \le \|\tilde{z}_0\|_{L^2(\Omega)}^2 + 2\int_0^{t_m} |L(t; Z_h)|dt.$$

The final result is obtained by using the bounds (3.16) and (3.5). □

Error estimates are obtained in a similar way.

3.5 Implementation

Once the DG method has been implemented for solving elliptic equations, it is a simple task to modify the software so that it solves parabolic equations. All time discretization methods presented in this chapter are one-step methods: in order to determine the solution at time t^{n+1}, we need to know the solution at the previous time step t^n. For DG in time discretization, the solution on the previous interval I_{n-1} is needed. We use the same data structure defined in Section 2.9, but we modify the attributes of the structure `element`.

```
typedef struct {
    int    face[3];
    int    parent;
    int    child[4];
    int    degree;
    int    reftype;
    double *soldofs_prev;
    double *soldofs_curr;
} element;
```

The array `soldofs_prev` stores the local degrees of freedom for the solution at the previous time step, whereas the array `soldofs_curr` stores the local degrees of freedom for the solution at the current time step.

If the forward Euler time discretization is used, the mass matrix $\boldsymbol{M}$ is block diagonal, and thus local linear systems on each mesh element are solved. This is an attractive feature for parallelizing the code. In addition, basis functions can be chosen so that $\boldsymbol{M}$ becomes the identity matrix.

We finish this section by giving the algorithms for the backward Euler and forward Euler schemes.

Algorithm 3.1.
Backward Euler DG scheme
initialize $\Delta t = 1/N_T$
compute mass matrix $\boldsymbol{M}$
compute stiffness matrix $\boldsymbol{A}$
compute right-hand side $(\tilde{\boldsymbol{b}})_i = (\tilde{z}_0, \tilde{\phi}_i)_\Omega$
solve $\boldsymbol{M}\tilde{\boldsymbol{\xi}}_0 = \tilde{\boldsymbol{b}}$
loop over time steps: for $n = 1$ to N_T do
  compute right-hand side $\boldsymbol{F}^{n+1}$
  solve $(\boldsymbol{M} + \Delta t\boldsymbol{A})\tilde{\boldsymbol{\xi}}_1 = \boldsymbol{M}\tilde{\boldsymbol{\xi}}_0 + \Delta t\boldsymbol{F}^n$
  copy $\tilde{\boldsymbol{\xi}}_1 = \tilde{\boldsymbol{\xi}}_0$
end

Algorithm 3.2.
Forward Euler DG scheme
initialize $\Delta t = 1/N_T$
compute mass matrix $\boldsymbol{M}$
compute stiffness matrix $\boldsymbol{A}$

compute right-hand side $(\tilde{\boldsymbol{b}})_i = (\tilde{z}_0, \tilde{\phi}_i)_\Omega$
solve $\boldsymbol{M}\tilde{\boldsymbol{\xi}}_0 = \tilde{\boldsymbol{b}}$
loop over time steps: for $n = 1$ to N_T do
    compute right-hand side $\boldsymbol{F}^{n+1}$
    solve $\boldsymbol{M}\tilde{\boldsymbol{\xi}}_1 = (\boldsymbol{M} + \Delta t\boldsymbol{A})\tilde{\boldsymbol{\xi}}_0 + \Delta t\boldsymbol{F}^{n-1}$
    copy $\tilde{\boldsymbol{\xi}}_1 = \tilde{\boldsymbol{\xi}}_0$
end

3.6 Bibliographical remarks

Primal DG methods combined with backward Euler or Crank–Nicolson time stepping techniques for solving parabolic problems are analyzed in [1, 91]. A general study of the Runge–Kutta methods can be found in [22]. DG in time methods for partial differential equations have been thoroughly studied (see, for example, [80, 53, 54, 106] and the references therein). Recent developments can be found in [55, 98, 99, 78].

Exercises

3.1. Let $\partial\Omega = \Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}}$. Define the DG semidiscrete scheme for the parabolic problem (3.7) with the boundary conditions

$$\begin{aligned} z &= z_{\mathrm{D}} \quad \text{on} \quad \Gamma_{\mathrm{D}}, \\ \boldsymbol{K}\nabla z \cdot \boldsymbol{n} &= z_{\mathrm{N}} \quad \text{on} \quad \Gamma_{\mathrm{N}}. \end{aligned}$$

3.2. Show that the matrix $\boldsymbol{M}$ is block diagonal and symmetric positive definite.

3.3. Show that the matrix $\boldsymbol{A}$ is positive definite.

3.4. Show that

$$\frac{1}{2}(\|v\|_{0,\Omega}^2 - \|w\|_{0,\Omega}^2) \leq (v - w, v).$$

3.5. Prove error estimates without using the elliptic projection. Rather use a standard approximation (continuous Lagrange interpolant for instance). State all conditions on the parameters.

3.6. Modify the proof of Lemma 3.6 so that Poincaré's inequality is used instead of Gronwall's inequality. The resulting constant C is then independent of T.

3.7. Prove the bounds (3.30) and (3.31).

3.8. Prove Lemma 3.8.

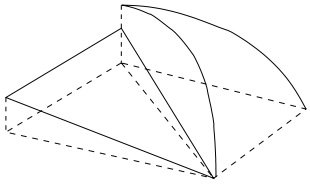
3.9. Prove Theorem 3.10.

3.10. Derive the error estimates (3.35) and (3.36) for the Crank–Nicolson time discretization.

3.11. Write in C or MATLAB® a program that simulates the diffusion of a contaminant through a porous medium, modeled by the following equation:

$$\begin{aligned} \tfrac{\partial z}{\partial t}(t,x) - \tfrac{\partial^2 z}{\partial x^2}(t,x) &= e^{-x^2}(\cos(t) - 2(2x^2-1)\sin(t)) \quad \text{in} \quad (0,\pi)\times(0,1), \\ z(t,0) &= \sin(t), \\ z(t,1) &= e^{-1}\sin(t), \\ z(0,x) &= 0, \quad x \in (0,1). \end{aligned}$$

For the space discretization, use the SIPG and NIPG methods with the following parameters: $\sigma \in \{1; 10; 100\}$. For the time discretization, use the backward Euler method and the DG in time method of degree 1. Given the exact solution $z(t,x) = e^{-x^2}\cos(t)$, compute the numerical errors in $l^\infty(0,\pi; L^2(0,1))$ and $l^2(0,\pi; H_0^1(0,1))$ and the convergence rates for each of the 12 cases. For this, one needs to choose a step size Δt and a mesh size h accordingly; for example one may start with $h = 0.1$ and then divide h successively into two. Plot the numerical solutions in all cases.



Chapter 4
Parabolic problems with convection

This chapter deals with the transport equation, i.e., a parabolic equation with a convective term. Issues such as slope limiting for overshoot and undershoot phenomena may arise if the convective term dominates the diffusive term.

4.1 Model problem

Let Ω be a polygonal domain in $\mathbb{R}^d$, $d = 1, 2$, or 3. Let $\boldsymbol{u} : \mathbb{R}^d \to \mathbb{R}^d$ be a vector function satisfying

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

The boundary of the domain Ω is decomposed into two parts: the inflow part Γ_{in} and outflow part Γ_{out} defined by

$$\Gamma_{\text{in}} = \{\boldsymbol{x} \in \partial\Omega : \boldsymbol{u} \cdot \boldsymbol{n} < 0\}, \quad \Gamma_{\text{out}} = \partial\Omega \setminus \Gamma_{\text{in}}.$$

For $f(z) \in L^2(0, T; L^2(\Omega))$, z_{in} in $L^2(0, T; H^{\frac{1}{2}}(\Gamma_{\text{in}}))$, and $z_0 \in L^2(\Omega)$, we consider the parabolic problem

$$\frac{\partial z}{\partial t} + \nabla \cdot (\boldsymbol{u}z - D\nabla z) = f(z) \text{ in } (0, T) \times \Omega, \tag{4.1}$$

$$(\boldsymbol{u}z - D\nabla z) \cdot \boldsymbol{n} = z_{\text{in}}\boldsymbol{u} \cdot \boldsymbol{n} \quad \text{on } (0, T) \times \Gamma_{\text{in}}, \tag{4.2}$$

$$-D\nabla z \cdot \boldsymbol{n} = 0 \quad \text{on } (0, T) \times \Gamma_{\text{out}}, \tag{4.3}$$

$$z = z_0 \quad \text{in } \{0\} \times \Omega. \tag{4.4}$$

This problem models, for example, the transport of a chemical species through a porous medium. The vector $\boldsymbol{u}$ is a given divergence-free velocity field, and the function z is the concentration of the contaminant. The function $f(z)$ is a source function that is Lipschitz with respect to z:

$$\forall w, v, \quad |f(w) - f(v)| \leq C|w - v|. \tag{4.5}$$

The parameter D is a diffusion coefficient that may vary in space, but it is bounded above and below by positive constants:

$$\forall \boldsymbol{x}, \quad 0 < D_0 \leq D(\boldsymbol{x}) \leq D_1. \tag{4.6}$$

The concentrations z_{in} and z_0 are, respectively, the concentration at the inflow boundary and the concentration at the initial time.

Problem (4.1)–(4.4) differs from problem (3.7)–(3.9) in the sense that the convective term $\nabla \cdot (\boldsymbol{u}z)$ has been added, the source function depends on the solution in a nonlinear fashion, and the boundary conditions are of mixed type on part of the boundary and of Neumann type on the other part. The transport equation (4.1) is written in a conservative form. Using the fact that $\boldsymbol{u}$ is divergence free, we have

$$\nabla \cdot (\boldsymbol{u}z) = (\nabla \cdot \boldsymbol{u})z + \boldsymbol{u} \cdot \nabla z = \boldsymbol{u} \cdot \nabla z.$$

This yields a nonconservative form of the transport equation:

$$\frac{\partial z}{\partial t} + \boldsymbol{u} \cdot \nabla z - \nabla \cdot D \nabla z = f(z) \text{ in } (0, T) \times \Omega.$$

4.2 Semidiscrete formulation

Problem (4.1)–(4.4) is first discretized in space by the DG method. The domain Ω is subdivided into elements (see notation defined in Section 2.3). First, the diffusive term $\nabla \cdot (D \nabla z)$ is discretized by the usual bilinear form a_ϵ similar to the one given in (2.23). Because of the mixed boundary conditions, the jump term penalizes the interior faces only:

$$\begin{aligned} a_\epsilon(w, v) = & \sum_{E \in \mathcal{E}_h} \int_E D \nabla w \cdot \nabla v + \sum_{e \in \Gamma_h} \frac{\sigma_e^0}{|e|^{\beta_0}} \int_e [w][v] \\ & - \sum_{e \in \Gamma_h} \int_e \{D \nabla w \cdot \boldsymbol{n}_e\} [v] + \epsilon \sum_{e \in \Gamma_h} \int_e \{D \nabla v \cdot \boldsymbol{n}_e\} [w]. \end{aligned} \tag{4.7}$$

This bilinear form yields the following energy seminorm:

$$\|v\|_{\mathcal{E}} = \left(\sum_{E \in \mathcal{E}_h} \|D^{1/2} \nabla v\|^2_{L^2(E)} + \sum_{e \in \Gamma_h} \frac{\sigma_e^0}{|e|^{\beta_0}} \|[v]\|^2_{L^2(e)} \right)^{1/2}.$$

Second, the convection term $\nabla \cdot (\boldsymbol{u}z)$ is approximated by an upwind discretization. Let us denote the upwind value of a function w by w^{up}. We recall that $\boldsymbol{n}_e$ is a unit normal vector pointing from E_e^1 to E_e^2:

$$w^{\text{up}} = \begin{cases} w|_{E_e^1} & \text{if} \quad \boldsymbol{u} \cdot \boldsymbol{n}_e \geq 0 \\ w|_{E_e^2} & \text{if} \quad \boldsymbol{u} \cdot \boldsymbol{n}_e < 0 \end{cases} \qquad \forall e = \partial E_e^1 \cap \partial E_e^2. \tag{4.8}$$

The dependence of the upwind discretization b on the vector $\boldsymbol{u}$ is explicitly given:

$$b(\boldsymbol{u}; w, v) = - \sum_{E \in \mathcal{E}_h} \int_E \boldsymbol{u} w \cdot \nabla v + \sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e w^{\text{up}} [v] + \sum_{e \in \Gamma_{\text{out}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e w v. \tag{4.9}$$

The semidiscrete DG approximation $Z_h \in L^2(0,T;\mathcal{D}_k(\mathcal{E}_h))$ satisfies the variational formulation

$$\forall t>0,\ \forall v\in\mathcal{D}_k(\mathcal{E}_h),\ \left(\frac{\partial Z_h}{\partial t},v\right)_\Omega + a_\epsilon(Z_h(t),v)+b(\boldsymbol{u};Z_h(t),v)=L(\boldsymbol{u},Z_h(t);v), \tag{4.10}$$

$$\forall v\in\mathcal{D}_k(\mathcal{E}_h),\quad (Z_h(0),v)_\Omega=(z_0,v)_\Omega, \tag{4.11}$$

where the form L is

$$L(\boldsymbol{u},w;v)=\int_\Omega f(w)v-\sum_{e\in\Gamma_{\text{in}}}\int_e \boldsymbol{u}\cdot\boldsymbol{n}_e z_{\text{in}}v.$$

4.2.1 Existence and uniqueness of solution

Assume that $\{\tilde{\phi}_i : 1\le i\le N_{\text{loc}}N_{\text{el}}\}$ is a basis of the finite-dimensional space $\mathcal{D}_k(\mathcal{E}_h)$ (see Section 3.3). For all $t>0$, we can write the solution $Z_h(t)$ as a linear combination of the $\tilde{\phi}_i$'s with coefficients $\tilde{\xi}_i$'s. We obtain a system of ordinary differential equations:

$$\boldsymbol{M}\frac{d\tilde{\boldsymbol{\xi}}}{dt}(t)+(\boldsymbol{A}+\boldsymbol{B})\tilde{\boldsymbol{\xi}}=\boldsymbol{G}(\tilde{\boldsymbol{\xi}}),$$
$$\boldsymbol{M}\tilde{\boldsymbol{\xi}}(0)=\tilde{\boldsymbol{Z}}_0.$$

The matrices $\boldsymbol{M}$, $\boldsymbol{A}$ are defined by (3.15). The matrix $\boldsymbol{B}$ results from the convective term, and the vector $\boldsymbol{G}(\tilde{\boldsymbol{\xi}})$ depends on the vector solution

$$\forall 1\le i,j\le N_{\text{loc}}N_{\text{el}},\quad (\boldsymbol{B})_{ij}=b(\boldsymbol{u};\tilde{\phi}_j,\tilde{\phi}_i),$$
$$\forall 1\le i\le N_{\text{loc}}N_{\text{el}},\quad (\boldsymbol{G})_i=L(\boldsymbol{u};\tilde{\boldsymbol{\xi}};\tilde{\phi}_i).$$

Since the matrix $\boldsymbol{M}$ is invertible and the vector function $\boldsymbol{G}(\tilde{\boldsymbol{\xi}})$ is Lipschitz with respect to $\tilde{\boldsymbol{\xi}}$, there exists a unique solution to (4.10)–(4.11).

4.2.2 Consistency

Lemma 4.1. *If $z\in H^1(0,T;H^2(\mathcal{E}_h))$ is the solution of* (4.1)–(4.4)*, then z satisfies* (4.10)–(4.11).

Proof. Let v be a test function in $\mathcal{D}_k(\mathcal{E}_h)$. We multiply (4.1) by $v|_E$ and integrate by parts on one element $E\in\mathcal{E}_h$:

$$\left(\frac{\partial z}{\partial t},v\right)_E-\int_E(\boldsymbol{u}z-D\nabla z)\cdot\nabla v+\int_{\partial E}(\boldsymbol{u}z-D\nabla z)\cdot\boldsymbol{n}_E v=\int_E f(z)v.$$

Summing over all E and using the regularity of the exact solution, we obtain

$$\left(\frac{\partial z}{\partial t}, v\right)_\Omega + a_\epsilon(z, v) - \sum_{E\in\mathcal{E}_h}\int_E z\boldsymbol{u}\cdot\nabla v - \sum_{w\in\Gamma_{\text{out}}}\int_e D\nabla z\cdot\boldsymbol{n}_e v$$
$$+\sum_{e\in\Gamma_h}\int_e \boldsymbol{u}\cdot\boldsymbol{n}_e z[v] + \sum_{e\in\Gamma_{\text{out}}}\int_e \boldsymbol{u}\cdot\boldsymbol{n}_e zv + \sum_{e\in\Gamma_{\text{in}}}\int_e (\boldsymbol{u}z - D\nabla z)\cdot\boldsymbol{n}_e v = (f(z), v)_\Omega.$$

Using the boundary conditions (4.2), (4.3) and noting that $z^{\text{up}} = z$, we clearly have (4.10). Equation (4.11) is trivially satisfied. □

4.2.3 Error estimates

In this section, we state a priori error estimates for the semidiscrete scheme [94].

Theorem 4.2. *Assume that the solution z to problem (4.1)–(4.4) belongs to $H^1(0, T; H^s(\mathcal{E}_h))$ and that z_0 belongs to $H^s(\mathcal{E}_h)$ for $s > 3/2$. Assume that $\beta_0(d-1) \geq 1$. In the case of SIPG and IIPG, assume that σ_e^0 is sufficiently large for all e. Then, there is a constant C independent of h such that*

$$\|z - Z_h\|_{L^\infty(L^2(\Omega))} + \left(\int_0^T \|z(t) - Z_h(t)\|_{\mathcal{E}}^2 dt\right)^{1/2}$$
$$\leq Ch^{\min(k+1,s)-1}(\|z\|_{H^1(0,T;H^s(\mathcal{E}_h))} + \|z_0\|_{H^s(\mathcal{E}_h)}).$$

Proof. We skip many details, as most of the argument is similar to the proof of Theorem 3.4. We write $z - Z_h = \rho - \chi$ with $\rho = z - \tilde{z}$ and $\chi = Z_h - \tilde{z}$. The function $\tilde{z} \in \mathcal{D}_k(\mathcal{E}_h)$ is an approximation of z that satisfies good error bounds. For instance, $\tilde{z}$ can be the elliptic projection of z (see (3.20)). The error equation is satisfied for all v in $\mathcal{D}_k(\mathcal{E}_h)$:

$$\left(\frac{\partial \chi}{\partial t}, v\right)_\Omega + a_\epsilon(\chi, v) + b(\boldsymbol{u}; \chi, v) = \left(\frac{\partial \rho}{\partial t}, v\right)_\Omega + a_\epsilon(\rho, v)$$
$$+ b(\boldsymbol{u}; \rho, v) + (f(Z_h) - f(z), v)_\Omega.$$

Now, by choosing $v = \chi$ and using the coercivity property of a_ϵ, we obtain

$$\frac{1}{2}\frac{d}{dt}\|\chi\|_{L^2(\Omega)}^2 + \kappa\|\chi\|_{\mathcal{E}}^2 + b(\boldsymbol{u}; \chi, \chi) \leq \left(\frac{\partial \rho}{\partial t}, \chi\right)_\Omega$$
$$+a_\epsilon(\rho, \chi) + b(\boldsymbol{u}; \rho, \chi) + (f(Z_h) - f(z), \chi)_\Omega.$$

Next, we show how to handle the terms $b(\boldsymbol{u}; \chi, \chi)$, $b(\boldsymbol{u}; \rho, \chi)$, and $(f(Z_h) - f(z), \chi)_\Omega$ since the other terms are identical to the ones in the proof of Theorems 2.13 and 3.4. Using a technique introduced in [30], we use Green's formula on the convection term and the fact that $\nabla\cdot\boldsymbol{u} = 0$:

$$\sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{u}\chi\cdot\nabla\chi = \frac{1}{2}\sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{u}\cdot\nabla\chi^2$$

$$= \frac{1}{2} \sum_{E \in \mathcal{E}_h} \int_{\partial E} \boldsymbol{u} \cdot \boldsymbol{n}_E \chi^2$$
$$= \frac{1}{2} \sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e [\chi^2] + \frac{1}{2} \sum_{e \in \partial\Omega} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2.$$

Thus, we obtain

$$b(\boldsymbol{u}; \chi, \chi) = -\sum_{E \in \mathcal{E}_h} \int_E \boldsymbol{u}\chi \cdot \nabla\chi + \sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^{\mathrm{up}}[\chi] + \sum_{e \in \Gamma_{\mathrm{out}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2$$
$$= \sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \left(\chi^{\mathrm{up}}[\chi] - \frac{1}{2}[\chi^2] \right) - \frac{1}{2} \sum_{e \in \partial\Omega} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2 + \sum_{e \in \Gamma_{\mathrm{out}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2$$
$$= \sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e (\chi^{\mathrm{up}}[\chi] - \{\chi\}[\chi]) - \frac{1}{2} \sum_{e \in \Gamma_{\mathrm{in}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2 + \frac{1}{2} \sum_{e \in \Gamma_{\mathrm{out}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \chi^2$$
$$= \frac{1}{2} \sum_{e \in \Gamma_h} \int_e |\boldsymbol{u} \cdot \boldsymbol{n}_e| [\chi]^2 + \frac{1}{2} \sum_{e \in \Gamma_{\mathrm{in}}} \int_e |\boldsymbol{u} \cdot \boldsymbol{n}_e| \chi^2 + \frac{1}{2} \sum_{e \in \Gamma_{\mathrm{out}}} \int_e |\boldsymbol{u} \cdot \boldsymbol{n}_e| \chi^2.$$

Therefore, $b(\boldsymbol{u}; \chi, \chi) \geq 0$. We now bound each term in $b(\boldsymbol{u}; \rho, \chi)$. First, using Cauchy–Schwarz's and Young's inequalities, we have

$$\sum_{E \in \mathcal{E}_h} \int_E \boldsymbol{u}\rho \cdot \nabla\chi \leq C \sum_{E \in \mathcal{E}_h} \|\rho\|_{L^2(E)} \|\nabla\chi\|_{L^2(E)}$$
$$\leq \frac{\kappa}{8} \|\chi\|_{\mathcal{E}}^2 + C\|\rho\|_{L^2(\Omega)}^2.$$

Similarly, we have

$$\sum_{e \in \Gamma_h} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \rho^{\mathrm{up}}[\chi] \leq \sum_{e \in \Gamma_h} \| |\boldsymbol{u} \cdot \boldsymbol{n}_e|^{\frac{1}{2}} [\chi] \|_{0,e} \| |\boldsymbol{u} \cdot \boldsymbol{n}_e|^{\frac{1}{2}} \rho_* \|_{0,e}$$
$$\leq \frac{1}{4} \sum_{e \in \Gamma_h} \| |\boldsymbol{u} \cdot \boldsymbol{n}_e|^{\frac{1}{2}} [\chi] \|_{0,e}^2 + C \sum_{e \in \Gamma_h} \|\rho^{\mathrm{up}}\|_{L^2(e)}^2,$$

$$\sum_{e \in \Gamma_{\mathrm{out}}} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_e \rho\chi \leq \frac{1}{4} \sum_{e \in \Gamma_{\mathrm{out}}} \| |\boldsymbol{u} \cdot \boldsymbol{n}_e|^{\frac{1}{2}} \chi \|_{0,e}^2 + C \sum_{e \in \Gamma_{\mathrm{out}}} \|\rho\|_{L^2(e)}^2.$$

Finally, we bound the nonlinear source term, using the Lipschitz property:

$$\int_\Omega (f(Z_h) - f(z))\chi \leq C\|Z_h - z\|_{L^2(\Omega)} \|\chi\|_{L^2(\Omega)}$$
$$\leq C\|\chi\|_{L^2(\Omega)}^2 + C\|\rho\|_{L^2(\Omega)}^2.$$

The final result is obtained by combining all bounds and using Gronwall's inequality of Lemma 3.1. □

4.3 Fully discrete formulation

We can choose any of the time discretizations described in Chapter 3. The analysis of the resulting fully discrete schemes can be done in a similar way. We do not present the results here. Rather, we discuss the challenging case where the convection term dominates the diffusion term in the transport equation. Two quantities characterize the problem, i.e., the Peclet number P_e and the Courant number C_r:

$$P_e = \frac{\|\boldsymbol{u}\|h}{D}, \quad C_r = \frac{\|\boldsymbol{u}\|\Delta t}{h}.$$

The Peclet number relates the rate of convection to the rate of diffusion. The higher P_e is, the larger advection effects are. The Courant number appears in the stability condition for explicit in time discretizations. This stability condition requires in general that C_r is bounded by a small constant.

4.3.1 Overshoot and undershoot

The solution z of (4.1)–(4.4) is naturally bounded below and above by some constants if the source function $f(z)$ on the right-hand side of (4.1) is equal to zero. If z represents a concentration, we can assume that

$$0 \leq z \leq 1. \tag{4.12}$$

Overshoot and undershoot phenomena occur when the numerical solution Z_h does not satisfy (4.12) or if the value Z_h increases (or decreases) without any physical reason. These phenomena occur with the DG method applied to the transport equation with high Peclet number. Because of the discontinuous approximation, overshoot and undershoot are localized near the front. We illustrate this point with a numerical example.

Let $\Omega = (0, 1)^2$ be subdivided into 10×10 rectangular elements. We solve problem (4.10) with the NIPG method with zero penalty and piecewise linear approximations ($k = 1$). We choose a velocity field $\boldsymbol{u} = (1, 0)$ and a Courant number $C_r = 0.1$. The inflow concentration is set equal to one on the left vertical boundary of Ω. The initial concentration is zero. The resulting numerical solution varies only in the x-direction. Fig. 4.1 shows the numerical concentration at a fixed time for different time discretizations and different Peclet numbers. We see that as P_e increases from 1 to infinity, the amount of overshoot and undershoot increases near the front. There is no overshoot and undershoot for $P_e = 1$ for forward Euler, backward Euler, and Crank–Nicolson schemes. For $P_e = 100$, the backward Euler discretization is the most diffusive one and yields an overshoot of 2.6% and undershoot of 1.6%. Forward Euler yields 12.3% of overshoot and 26.2% of undershoot, whereas Crank–Nicolson yields 5% of overshoot and 6.1% of undershoot. The case $P_e = \infty$ corresponds to a zero diffusion coefficient, and similar overshoot and undershoot quantities are obtained for all three time discretizations. Next, we set the penalty value $\sigma_e^0 = 1$ and use NIPG. Fig. 4.2 shows the amount of overshoot and undershoot for backward Euler and Crank–Nicolson schemes for the case $P_e = \infty$. The amount of overshoot increases (7% for backward Euler and 14.4% for Crank–Nicolson), whereas the amount of undershoot decreases (0.6% for backward Euler and 2.6% for Crank–Nicolson). If we use SIPG or IIPG with $\sigma_e^0 = 1$, we obtain similar results.

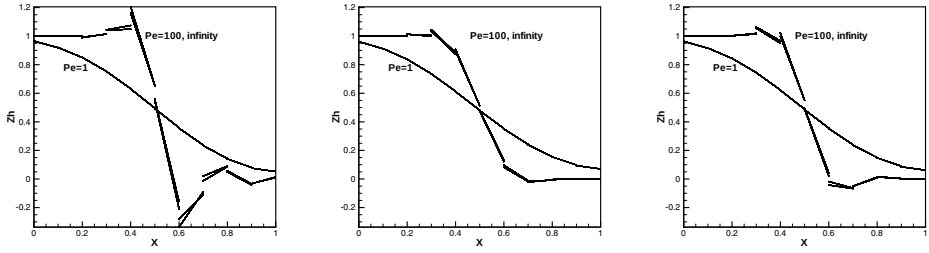


Figure 4.1. *Profiles of numerical concentration obtained for different Peclet numbers and different time discretizations: forward Euler (left), backward Euler (center), and Crank–Nicolson (right) with NIPG* 0*.*

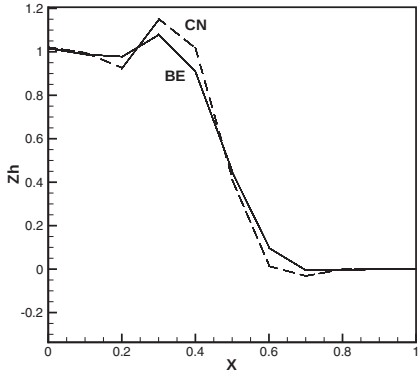


Figure 4.2. *Profiles of numerical concentration obtained with backward Euler (BE) and Crank–Nicolson (CN) and with NIPG* 1 *and* $P_e = \infty$*.*

Remark: If the DG space is the space of piecewise constants, then problem (4.1) reduces to a convection problem as the diffusive terms become zero in (4.10). The solution Z_h is then monotone, and no overshoot/undershoot is observed. However, the solution is too diffusive, and it is necessary to use higher polynomial degrees.

4.3.2 Slope limiters

The overshoot/undershoot phenomena can be greatly reduced by the use of slope limiters. On each mesh element, the limiting procedure replaces the solution Z_h locally by a piecewise linear $\mathcal{L}Z_h$ wherever necessary [37, 46, 70]. We present some slope limiting techniques in 1D and 2D.

Slope limiter in 1D:
Assume that piecewise quadratics are used. On the interval (x_n, x_{n+1}), we can rewrite the DG solution as

$$\forall x_n \le x \le x_{n+1}, \quad Z_h(x) = a_0^n + a_1^n \psi_n(x) + a_2^n \left((\psi_n(x))^2 - \frac{1}{3} \right),$$

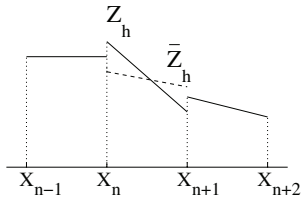


Figure 4.3. *DG solution before limiting Z_h (solid line) and after limiting $\bar{Z}_h$ (dashed line).*

where $a_0^n, a_1^n, a_2^n \in \mathbb{R}$ and ψ_n is the linear function defined by

$$\psi_n(x) = \frac{x - \frac{x_n + x_{n+1}}{2}}{x_{n+1} - x_n}.$$

It is easy to obtain the coefficients a_i^n from the coefficients α_i^n of the expansion (1.9). In this equivalent form, the coefficient a_0^n represents the average of Z_h over the interval (x_n, x_{n+1}). We compute the limited coefficient:

$$\bar{a}_1^n = \begin{cases} a_1^n \quad \text{if} \quad |a_1^n| \le M_{\text{lim}} \\ \text{minmod}(a_1^n, \gamma(a_0^{n+1} - a_0^n), \gamma(a_0^n - a_0^{n-1})) \quad \text{otherwise.} \end{cases}$$

The function minmod is defined by

$$\text{minmod}(y_1, y_2, y_3) = \begin{cases} s \min_{1 \le i \le 3} |a_i| \quad \text{if} \quad s = \text{sign}(y_1) = \text{sign}(y_2) = \text{sign}(y_3), \\ 0 \quad \text{otherwise.} \end{cases}$$

The parameters M_{lim} and γ control the amount of limiting. The amount of overshoot/undershoot decreases as M_{lim} decreases. In practice, $\gamma = 1$ or $\gamma = 0.5$. If $\bar{a}_1^n \neq a_1^n$, then the limited solution is

$$\mathcal{L}Z_h = a_0^n + \bar{a}_1^n \psi_n(x).$$

Fig. 4.3 shows the new limited solution with respect to the original solution.

This limiter relies on the assumption that spurious oscillations are present in Z_h only if they are present in its linear part, which is its L^2 projection into the space of piecewise linear functions. In that case, the higher order part of the approximation is eliminated.

Slope limiter in 2D:
Rectangles: If the element is rectangular, we can apply the one-dimensional limiter in each direction. For instance, if $E = (x_n, x_{n+1}) \times (y_m, y_{m+1})$, we can expand the DG solution as

$$\begin{aligned} \forall (x, y) \in E, \quad Z_h(x, y) = a_0^{nm} + a_1^{nm}\psi_n(x) + a_2^{nm}\xi_m(y) + a_3^{nm}\psi_n(x)\xi_m(y) \\ + a_4^{nm}\left((\psi_n(x))^2 - \frac{1}{3}\right) + a_5^{nm}\left((\xi_m(y))^2 - \frac{1}{3}\right), \end{aligned}$$

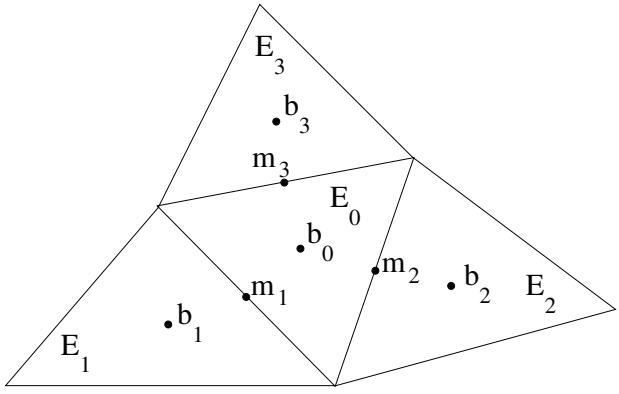


Figure 4.4. *Triangle configuration for building the limiter.*

where ψ_m is defined above and ξ_n is the corresponding linear:

$$\xi_m(y) = \frac{y - \frac{y_m + y_{m+1}}{2}}{y_{m+1} - y_m}.$$

The coefficient a_0^{nm} represents the average of Z_h over E. Then, we compute

$$\bar{a}_1^{nm} = \begin{cases} a_1^{nm} & \text{if} \quad |a_1^{nm}| \le M_{\text{lim}} \\ \text{minmod}(a_1^{nm}, \gamma(a_0^{n+1,m} - a_0^{n,m}), \gamma(a_0^{nm} - a_0^{n-1,m})) & \text{otherwise.} \end{cases}$$

$$\bar{a}_2^{nm} = \begin{cases} a_2^{nm} & \text{if} \quad |a_2^{nm}| \le M_{\text{lim}} \\ \text{minmod}(a_2^{nm}, \gamma(a_0^{n,m+1} - a_0^{n,m}), \gamma(a_0^{nm} - a_0^{n,m-1})) & \text{otherwise.} \end{cases}$$

If $\bar{a}_1^{nm} \neq a_1^{nm}$ or $\bar{a}_2^{nm} \neq a_2^{nm}$, the solution Z_h is replaced by

$$\mathcal{L}Z_h = a_0^n + \bar{a}_1^{nm}\psi_n(x) + \bar{a}_2^{nm}\xi_m(y).$$

Triangles: Next, we define a limiter on triangular elements. This postprocessing consists of several steps.

(i) Step one: compute neighbor averages:
We first compute the average saturation for the element to be limited and all neighboring elements as follows. Assume that a given element E_0 has three neighbors E_1, E_2, E_3 (see Fig. 4.4). Let $\bar{Z}_i$ denote the average solution of Z_h over E_i:

$$\bar{Z}_i = \frac{1}{|E_i|}\int_{E_i} Z_h.$$

(ii) Step two: test if limiting is necessary:
Let m_j denote the midpoints of the edges of E_0. We then compute the saturation $Z_h|_{E_0}(m_j)$, and we check that this value is between $\bar{Z}_j$ and $\bar{Z}_0$. We stop here if the test is successful; otherwise we continue to step three.

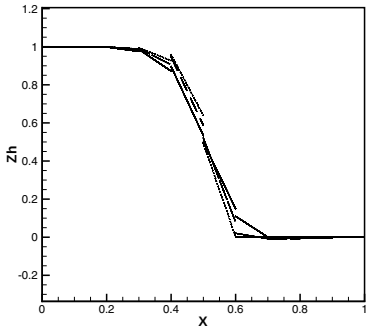


Figure 4.5. *Limited DG solution for NIPG* 0: *backward Euler (solid line), Crank–Nicolson (dashed line), and forward Euler (dotted line).*

(iii) Step three: construct and rank three linears:
We construct three linears using the gravity centers b_j of E_j and the averages $\bar{Z}_j$. For instance, if we write the linears as $L_j(x, y) = a_0^j + a_1^j x + a_2^j y$, for $j \in \{1, 2, 3\}$, they are uniquely determined by

$$L_j(b_0) = \bar{Z}_0 \quad \text{and} \quad L_j(b_l) = \bar{Z}_l \text{ for } l \neq j.$$

We then rank the linears by decreasing $\sqrt{(a_1^j)^2 + (a_2^j)^2}$.

(iv) Step four: select appropriate linear:
We finally check that the values of the linears evaluated at the midpoint m_l, $L_j(m_l)$, are between $\bar{S}_l$ and $\bar{S}_0$ for $1 \geq l \geq 3$. The linears are tested in the order given in the previous step. If one of the linears passes this test, it is chosen to be the limited solution. Otherwise, if none of the constructed linears satisfies the test, then the slope is reduced to 0.

We repeat the experiments described in Section 4.3.1, but we now apply the slope limiter after each time step. We choose $M_{\text{lim}} = 0, \gamma = 1$. The diffusion coefficient is $D = 10^{-3}$, which yields a Peclet number $P_e = 100$. The amount of overshoot is reduced to zero for all three time discretizations. The undershoot is zero for forward Euler and very small for backward Euler (0.4%) and Crank–Nicolson (0.8%). Fig. 4.5 shows the solution Z_h for NIPG 0 (to compare with Fig. 4.1). Similarly, Fig. 4.6 shows the limited solution for NIPG 1 (to compare with Fig. 4.2). If forward Euler is stable, it gives a sharper front than Crank–Nicolson. The more diffuse front is obtained with backward Euler.

4.3.3 An improved DG method

In many applications, the diffusion coefficient D varies over the domain with several orders of magnitude. In this case, some overshoot and undershoot occur at the interface Γ_* through

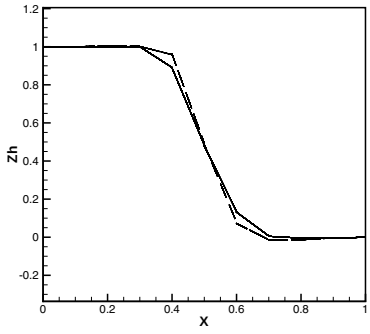


Figure 4.6. *Limited DG solution for NIPG* 1: *backward Euler (solid line) and Crank–Nicolson (dashed line).*

which the flow crosses from a region with low diffusion to a region with high diffusion. We propose an improved DG method without using slope limiters, for which the local oscillations are minimal in the neighborhood of Γ_*. We assume that within each mesh element, the diffusion coefficient is of the same order. We modify the primal DG methods and consider the upwind diffusive fluxes on Γ_*. In (4.10), the bilinear form a_ϵ is replaced by $\tilde{a}_\epsilon$:

$$\begin{aligned}\tilde{a}_\epsilon(w,v) = \sum_{E\in\mathcal{E}_h}\int_E D\nabla w\cdot\nabla v + \sum_{e\in\Gamma_h\setminus\Gamma_*}\frac{\sigma_e^0}{|e|^{\beta_0}}\int_e [w][v]\\ - \sum_{e\in\Gamma_h\setminus\Gamma_*}\int_e \{D\nabla w\cdot\boldsymbol{n}_e\}[v] + \epsilon\sum_{e\in\Gamma_h\setminus\Gamma_*}\int_e \{D\nabla v\cdot\boldsymbol{n}_e\}[w]\\ + \sum_{e\in\Gamma_*}\frac{\sigma_e^0}{|e|^{\beta_0}}\int_e D^{\mathrm{up}}[w][v] - \sum_{e\in\Gamma_*}\int_e (D\nabla w\cdot\boldsymbol{n}_e)^{\mathrm{up}}[v]\\ +\epsilon\sum_{e\in\Gamma_*}\int_e (D\nabla v\cdot\boldsymbol{n}_e)^{\mathrm{up}}[w].\end{aligned}$$

The resulting scheme is still consistent, and optimal a priori error estimates can be obtained. Fig. 4.7 shows a two-dimensional domain with varying diffusion and the numerical solutions obtained with the usual and improved NIPG methods and an explicit forward Euler time discretization. The diffusion coefficient is one thousand times smaller in the black regions than in the white regions. The velocity field is $\boldsymbol{u} = (1, 0)$. The interface Γ_* is thus the union of some of the mesh edges. We observe that the amount of overshoot and undershoot is large for the standard NIPG (without upwinding the diffusive flux), and, after several time steps, the solution blows up. The improved NIPG has nearly zero overshoot and undershoot, and the limiting steady-state solution is obtained as time increases.

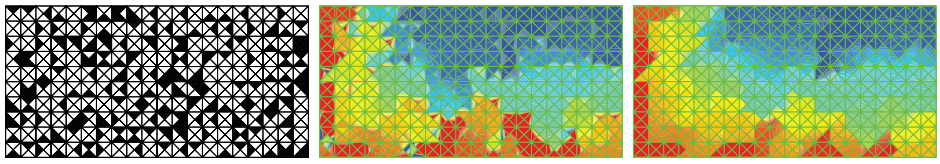


Figure 4.7. *Mesh and diffusion coefficient:* $D = 1$ *in white regions and* $D = 10^{-3}$ *in black regions (left). Contours of standard NIPG solution (center) and improved NIPG solution (right).*

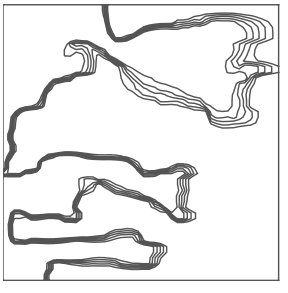
4.4 Bibliographical remarks

A DG method for steady-state convection-diffusion problems is defined in [80]: upwinding is introduced. More recent works are [25, 72]. Primal DG methods for reactive transport are studied in [94, 104, 105, 87]. Analysis of LDG methods applied to transport problems can be found in [36, 41].

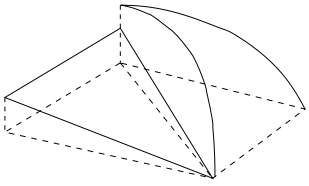
Exercises

4.1. In the case of the parabolic problem with a convection term, write down the local mass conservation property that the DG scheme satisfies.

4.2. In the case of the parabolic problem with a convection term, derive the fully discrete a priori error estimate for the backward Euler discretization and for the SIPG method.



Part III
Applications



Chapter 5
Linear elasticity

This chapter illustrates the use of primal DG methods for a simple solid mechanics problem, namely the linear elasticity problem. We show that the DG scheme is very similar to the one obtained for elliptic problems.

5.1 Preliminaries

5.1.1 Strain and stress tensors

Let $\boldsymbol{u}(\boldsymbol{x})$ be the displacement vector at a point $\boldsymbol{x}$ of a homogeneous elastic body $\Omega \subset \mathbb{R}^d$, $d = 2, 3$. The strain (or deformation) tensor $\boldsymbol{\epsilon}(\boldsymbol{u}) = (\epsilon_{kl}(\boldsymbol{u}))_{1\leq k,l\leq d}$ is defined by

$$\boldsymbol{\epsilon}(\boldsymbol{u}) = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T),$$

or equivalently

$$\forall 1 \leq k, l \leq d, \quad \epsilon_{kl}(\boldsymbol{u}) = \frac{1}{2}\left(\frac{\partial u_k}{\partial x_l} + \frac{\partial u_l}{\partial x_k}\right),$$

since $\nabla \boldsymbol{u} = (\frac{\partial u_k}{\partial x_l})_{k,l}$. The stress tensor is denoted by $\boldsymbol{\sigma}(\boldsymbol{u}) = (\sigma_{ij}(\boldsymbol{u}))_{1\leq i,j\leq d}$ such that $\sigma_{ii}(\boldsymbol{u})$ is the normal stress in the direction x_i and $\sigma_{ij}(\boldsymbol{u})$ for $i \neq j$ are the shear stresses. The stress tensor satisfies the constitutive relationship:

$$\forall 1 \leq i, j \leq d, \quad \sigma_{ij}(\boldsymbol{u}) = \sum_{k,l=1}^{d} D_{ijkl}\epsilon_{kl}(\boldsymbol{u}), \tag{5.1}$$

where $\boldsymbol{D} = (D_{ijkl})_{ijkl}$ is a fourth order tensor satisfying some symmetry properties:

$$D_{ijkl} = D_{jikl} = D_{ijlk} = D_{klij}. \tag{5.2}$$

We assume that $\boldsymbol{D}$ is positive definite and piecewise constant in Ω, i.e,

$$\forall (\gamma_{ij})_{ij} \neq 0, \quad 0 < D_0 \sum_{ij} \gamma_{ij}^2 \leq \sum_{ijkl} \gamma_{ij} D_{ijkl} \gamma_{kl} \leq D_1 \sum_{ij} \gamma_{ij}^2. \tag{5.3}$$

For example, in 2D, by Hooke's law, the stress tensor can be written as

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{pmatrix} = \begin{pmatrix} \lambda + 2\mu & \lambda & 0 \\ \lambda & \lambda + 2\mu & 0 \\ 0 & 0 & \mu \end{pmatrix} \begin{pmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{12} \end{pmatrix},$$

where $\lambda > 0$ and $\mu > 0$ are the Lamé coefficients of the material.

5.1.2 Korn's inequalities

From the definition of the strain tensor, we immediately have

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^d, \quad \|\boldsymbol{\epsilon}(\boldsymbol{v})\|_{L^2(\Omega)} \leq \|\nabla \boldsymbol{v}\|_{L^2(\Omega)}.$$

The reverse inequality is not true in general. If $\boldsymbol{v} \neq \boldsymbol{0}$ belongs to the space of rigid motions on Ω, then $\boldsymbol{\epsilon}(\boldsymbol{v}) = 0$. However, the reverse inequality is valid for functions vanishing on the boundary. This is Korn's first inequality [47, 29] in the usual Sobolev space $H_0^1(\Omega)^d$: there is a constant $C > 0$ such that

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^d, \quad \|\nabla \boldsymbol{v}\|_{L^2(\Omega)} \leq C\|\boldsymbol{\epsilon}(\boldsymbol{v})\|_{L^2(\Omega)}.$$

In the Sobolev space $H^1(\Omega)^d$, the classical Korn inequality states that there is a constant $C > 0$ such that

$$\forall \boldsymbol{v} \in H^1(\Omega)^d, \quad \|\nabla \boldsymbol{v}\|_{L^2(\Omega)} \leq C(\|\boldsymbol{\epsilon}(\boldsymbol{v})\|_{L^2(\Omega)} + \|\boldsymbol{v}\|_{L^2(\Omega)}).$$

Korn's first inequality can be generalized to the broken Sobolev space $H^1(\mathcal{E}_h)^d$ (see [15]). Assume that Γ_{D} is a subset of the boundary $\partial\Omega$ with $|\Gamma_{\mathrm{D}}| > 0$. Then, there exists a positive constant C such that

$$\forall \boldsymbol{v} \in H^1(\mathcal{E}_h)^d, \quad |||\nabla \boldsymbol{v}|||_{H^0(\mathcal{E}_h)} \leq C \left(|||\boldsymbol{\epsilon}(\boldsymbol{v})|||^2_{H^0(\mathcal{E}_h)} + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{1}{|e|^{\frac{1}{d-1}}} \|[\boldsymbol{v}]\|^2_{L^2(e)} \right)^{1/2}. \quad (5.4)$$

5.2 Model problem

Assume that the boundary of the elastic body is divided into two disjoint sets Γ_{D} and Γ_{N} and assume that a system of body forces $\boldsymbol{f} : \Omega \to \mathbb{R}^d$ and surface tractions $\boldsymbol{g}_{\mathrm{N}} : \Gamma_{\mathrm{N}} \to \mathbb{R}^d$ act on the body. On the other part Γ_{D} of the boundary, the body is rigidly fixed in space. Under the assumption of small displacements [65], the displacement $\boldsymbol{u} = (u_i)_{1 \leq i \leq d}$ satisfies the following problem:

$$-\sum_{j=1}^{d} \frac{\partial \sigma_{ij}}{\partial x_j}(\boldsymbol{u}) = f_i \quad \text{in} \quad \Omega \quad \forall i = 1, \ldots, d, \quad (5.5)$$

$$u_i = 0 \quad \text{on} \quad \Gamma_D \quad \forall i = 1, \ldots, d, \quad (5.6)$$

$$\sum_{j=1}^{d} \sigma_{ij}(\boldsymbol{u}) n_j = g_i \quad \text{on} \quad \Gamma_N \quad \forall i = 1, \ldots, d, \quad (5.7)$$

where $\boldsymbol{n} = (n_i)_{1\le i\le d}$ is the unit outward normal to the boundary $\partial\Omega$ and f_i and g_i are the components of the forces $\boldsymbol{f}$ and $\boldsymbol{g}_{\mathrm{N}}$. Equations (5.5) represent the equilibrium equations. From [29], there exists a weak solution $\boldsymbol{u} \in H_0^1(\Omega)^d$ satisfying

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^d, \quad \int_\Omega \sum_{ijkl} D_{ijkl}\epsilon_{kl}(\boldsymbol{u})\epsilon_{ij}(\boldsymbol{v}) = \int_\Omega \boldsymbol{f}\cdot\boldsymbol{v} + \int_{\Gamma_N} \boldsymbol{g}_{\mathrm{N}}\cdot\boldsymbol{v}.$$

Besides, if $\Gamma_N = \emptyset$, then $\boldsymbol{u} \in H^2(\Omega)^d$.

5.3 DG scheme

Let $\mathcal{E}_h$ be a subdivision of Ω. The notation used here is the same as in Section 2.3. We consider the space of vector functions that generalizes the definition (2.29):

$$\boldsymbol{\mathcal{D}}_k(\mathcal{E}_h) = (\mathcal{D}_k(\mathcal{E}_h))^d.$$

The DG approximation $\boldsymbol{U}_h \in \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h)$ satisfies the discrete variational problem

$$\forall \boldsymbol{v} \in \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h), \quad a_\eta(\boldsymbol{U}_h, \boldsymbol{v}) = L(\boldsymbol{v}), \tag{5.8}$$

where the bilinear form $a_\eta : \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h) \times \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h) \to \mathbb{R}$ is given by

$$\begin{aligned} a_\eta(\boldsymbol{w}, \boldsymbol{v}) = &\sum_{E\in\mathcal{E}_h} \int_E \sum_{ijkl} D_{ijkl}\epsilon_{kl}(\boldsymbol{w})\epsilon_{ij}(\boldsymbol{v}) \\ - &\sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \sum_{ijkl} \{D_{ijkl}\epsilon_{kl}(\boldsymbol{w})n_j^e\}[v_i] + \eta \sum_{e\in\Gamma_h\cup\Gamma_D} \int_e \sum_{ijkl} \{D_{ijkl}\epsilon_{kl}(\boldsymbol{v})n_j^e\}[w_i] \\ &+ \sum_{i=1}^d \sum_{e\in\Gamma_h\cup\Gamma_D} \frac{\delta_e}{|e|^\beta} \int_e [w_i][v_i], \end{aligned}$$

and the linear form $L : \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h) \to \mathbb{R}$ is defined by

$$L(\boldsymbol{v}) = \int_\Omega \boldsymbol{f}\cdot\boldsymbol{v} + \int_{\Gamma_N} \boldsymbol{g}_{\mathrm{N}}\cdot\boldsymbol{v}.$$

To avoid any confusion with the strain tensor, the parameter that yields a symmetric or nonsymmetric bilinear form is denoted here by $\eta \in \{-1, 0, 1\}$. The last term in the bilinear form a_η is the penalty term with two additional parameters: the penalty value $\delta_e > 0$ that can vary from face to face and the power β that is usually taken equal to $(d-1)^{-1}$ but can be larger for a superpenalized DG method. The variable n_e^j denotes the jth component of $\boldsymbol{n}_e$.

The energy norm for the linear elasticity problem is defined below:

$$\|\boldsymbol{v}\|_{\mathcal{E}} = \left(\sum_{E\in\mathcal{E}_h} \|\boldsymbol{\epsilon}(\boldsymbol{v})\|_{L^2(E)}^2 + \sum_{e\in\Gamma_h\cup\Gamma_D} \frac{\delta_e}{|e|^\beta} \|[\boldsymbol{v}]\|_{L^2(e)}^2 \right)^{1/2}.$$

5.3.1 Consistency

Let $\boldsymbol{u}$ be the solution of (5.5)–(5.7). Then, following a similar argument as in Section 2.4.1, one can obtain that $\boldsymbol{u}$ satisfies (5.8). The proof requires the additional result, which is an easy consequence of the symmetry of the stress tensor:

$$\sum_{1\le i,j\le d} \sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} = \sum_{1\le i,j\le d} \sigma_{ij}(\boldsymbol{u})\epsilon_{ij}(\boldsymbol{v}).$$

Indeed, we have

$$\begin{aligned}
\sum_{1\le i,j\le d} \sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} &= \sum_{1\le i,j\le d} \frac{1}{2}\sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} + \sum_{1\le i,j\le d} \frac{1}{2}\sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} \\
&= \sum_{1\le i,j\le d} \frac{1}{2}\sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} + \sum_{1\le j,i\le d} \frac{1}{2}\sigma_{ji}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} \\
&= \sum_{1\le i,j\le d} \frac{1}{2}\sigma_{ij}(\boldsymbol{u})\frac{\partial v_i}{\partial x_j} + \sum_{1\le i,j\le d} \frac{1}{2}\sigma_{ij}(\boldsymbol{u})\frac{\partial v_j}{\partial x_i} \\
&= \sum_{1\le i,j\le d} \sigma_{ij}(\boldsymbol{u})\epsilon_{ij}(\boldsymbol{v}).
\end{aligned}$$

5.3.2 Local equilibrium

The analogous to local mass conservation for the elliptic problem (see Section 2.7.3) is here called local equilibrium, as the discretization of (5.5) is satisfied on each mesh element.

Lemma 5.1. *Fix a mesh element $E \in \mathcal{E}_h$, with outward normal $\boldsymbol{n}^E = (n_i^E)_{1\le i\le d}$. Let $\mathcal{N}(e; E)$ denote the element that shares the edge (or face) e with the element E and let U_h^i denote the ith component of $\boldsymbol{U}_h$:*

$$\begin{aligned}
\forall 1 \le i \le d, \quad \int_E f_i = &-\sum_{jkl}\int_{\partial E\setminus\Gamma_N}\{D_{ijkl}\epsilon_{kl}(\boldsymbol{U}_h)n_j^E\} - \int_{\partial E\cap\Gamma_N} g_i \\
&+ \sum_{e\in\partial E\setminus\Gamma_N}\frac{\delta_e}{|e|^\beta}\int_e (U_h^i|_E - U_h^i|_{\mathcal{N}(e;E)}).
\end{aligned}$$

Proof. For a fixed $1 \le i \le d$, choose the test function $\boldsymbol{v} = (v_j)_j$ in (5.8) such that $v_i = 1$ on E and zero elsewhere, and $v_j = 0$ for $j \ne i$. □

5.3.3 Coercivity

Lemma 5.2. *If the parameter η is equal to -1 or 0, assume that the penalty value δ_e is sufficiently large and that $\beta \ge (d-1)^{-1}$. There exists a positive constant κ independent of h such that*

$$\forall \boldsymbol{v} \in \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h), \quad \kappa\|\boldsymbol{v}\|_{\mathcal{E}}^2 \le a_\eta(\boldsymbol{v}, \boldsymbol{v}).$$

Proof. If $\eta = 1$, we simply have from (5.3)

$$a_1(\boldsymbol{v}, \boldsymbol{v}) \geq D_0 \|\boldsymbol{\epsilon}(\boldsymbol{v})\|_0^2 + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{\delta_e}{|e|^\beta} \|[\boldsymbol{v}]\|_{L^2(e)}^2 \geq \min(D_0, 1) \|\boldsymbol{v}\|_{\mathcal{E}}^2.$$

If $\eta = -1$ or $\eta = 0$, we have

$$\begin{aligned} a(\boldsymbol{v}, \boldsymbol{v}) \geq D_0 \|\boldsymbol{\epsilon}(\boldsymbol{v})\|_0^2 + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{\delta_e}{|e|^\beta} \|[\boldsymbol{v}]\|_{L^2(e)}^2 \\ -(1-\eta) \sum_{e \in \Gamma_h \cup \Gamma_D} \sum_{ijkl} \int_e \{D_{ijkl} \epsilon_{kl}(\boldsymbol{v}) n_j^e\} [v_i]. \end{aligned}$$

It suffices to bound the last term of the inequality above. This is done using the trace inequality (2.5) and following a similar argument as in Section 2.7.1. □

A consequence of Korn's inequality (5.4) and the coercivity of the bilinear form a_η is the following lemma.

Lemma 5.3. *There exists a unique solution $\boldsymbol{U}_h$ to problem* (5.8).

Proof. It suffices to prove uniqueness. Denoting by $\boldsymbol{w}_h$ the difference of two solutions $\boldsymbol{U}_h^1$ and $\boldsymbol{U}_h^2$ to problem (5.8), we have

$$\forall \boldsymbol{v} \in \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h), \quad a_\eta(\boldsymbol{w}_h, \boldsymbol{v}) = 0.$$

Choosing $\boldsymbol{v} = \boldsymbol{w}_h$ and using Lemma 5.2, we have

$$\|\boldsymbol{w}_h\|_{\mathcal{E}} = 0.$$

Thus, $\boldsymbol{w}_h = \boldsymbol{0}$ from (5.4). □

5.4 Error analysis

A priori error estimates in the energy norm are given in the following theorem.

Theorem 5.4. *Let $k \geq 1$. Assume that $\beta = (d-1)^{-1}$ if the mesh contains quadrilaterals or hexahedra; otherwise assume that $\beta \geq (d-1)^{-1}$. Assume that the solution $\boldsymbol{u}$ of* (5.5)–(5.7) *belongs to $H^s(\mathcal{E}_h)^d$ for $s \geq 3/2$. Then, under the assumptions of Lemma* 5.2*, there is a constant C independent of h such that*

$$\|\boldsymbol{U}_h - \boldsymbol{u}\|_{\mathcal{E}} \leq C h^{\min(k+1,s)-1} |||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)}. \tag{5.9}$$

Proof. The proof follows closely the proof of the error estimates for the elliptic problem (see Section 2.8). First, we obtain an orthogonality equation by using the consistency of the method:

$$\forall \boldsymbol{v} \in \boldsymbol{\mathcal{D}}_k(\mathcal{E}_h), \quad a_\eta(\boldsymbol{U}_h - \boldsymbol{u}, \boldsymbol{v}) = 0.$$

Let $\tilde{\boldsymbol{u}}$ be an approximation of $\boldsymbol{u}$ satisfying (2.10). Define $\boldsymbol{\chi} = \boldsymbol{U}_h - \tilde{\boldsymbol{u}}$ and choose $\boldsymbol{v} = \boldsymbol{\chi}$ in the equation above,

$$a_\eta(\boldsymbol{\chi}, \boldsymbol{\chi}) = a_\eta(\boldsymbol{u} - \tilde{\boldsymbol{u}}, \boldsymbol{\chi}),$$

or from the coercivity of a_η:

$$\begin{aligned}\kappa\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 \le & \sum_{E\in\mathcal{E}_h}\sum_{ijkl} D_{ijkl}\epsilon_{kl}(\boldsymbol{u}-\tilde{\boldsymbol{u}})\epsilon_{ij}(\boldsymbol{\chi}) - \sum_{e\in\Gamma_h\cup\Gamma_D}\sum_{ijkl}\int_e \{D_{ijkl}\epsilon_{kl}(\boldsymbol{u}-\tilde{\boldsymbol{u}})n_j^e\}[\chi_i] \\ & +\eta\sum_{e\in\Gamma_h\cup\Gamma_D}\sum_{ijkl}\int_e \{D_{ijkl}\epsilon_{kl}(\boldsymbol{\chi})n_j^e\}[u_i-\tilde{u}_i] + \sum_{e\in\Gamma_h\cup\Gamma_D}\frac{\delta_e}{|e|^\beta}\int_e [\boldsymbol{u}-\tilde{\boldsymbol{u}}]\cdot[\boldsymbol{\chi}] \\ & = T_1+\cdots+T_4. \end{aligned} \tag{5.10}$$

We now bound the terms T_i, using Cauchy–Schwarz's, Young's inequalities, and the approximation bounds:

$$\begin{aligned} T_1 &\le \frac{\kappa}{8}\sum_{E\in\mathcal{E}_h}\int_E D_{ijkl}\epsilon_{kl}(\boldsymbol{\chi})\epsilon_{ij}(\boldsymbol{\chi}) + C\sum_{E\in\mathcal{E}_h}\int_E D_{ijkl}\epsilon_{kl}(\boldsymbol{u}-\tilde{\boldsymbol{u}})\epsilon_{ij}(\boldsymbol{u}-\tilde{\boldsymbol{u}}) \\ &\le \frac{\kappa}{8}\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-2}|||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)}^2. \end{aligned}$$

The second term is bounded as follows:

$$\begin{aligned} T_2 &\le \sum_{e\in\Gamma_h\cup\Gamma_D}\sum_{ijkl}\left(\frac{|e|^\beta}{\delta_e}\right)^{\frac{1}{2}-\frac{1}{2}}\|\{D_{ijkl}\epsilon_{kl}(\boldsymbol{u}-\tilde{\boldsymbol{u}})n_j^e\}\|_{0,e}\|[\chi_i]\|_{0,e} \\ &\le \frac{\kappa}{8}\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 + C\sum_{e\in\Gamma_h\cup\Gamma_D}\sum_{ijkl}\frac{|e|^\beta}{\delta_e}\|\{D_{ijkl}\epsilon_{kl}(\boldsymbol{u}-\tilde{\boldsymbol{u}})n_j^e\}\|_{0,e}^2 \\ &\le \frac{\kappa}{8}\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-3+\beta(d-1)}|||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)}^2. \end{aligned}$$

Thus, if $\beta(d-1)\ge 1$, we have

$$T_2 \le \frac{\kappa}{8}\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-2}|||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)}^2.$$

The terms T_3 and T_4 vanish if the approximation $\tilde{\boldsymbol{u}}$ is continuous. Such an approximation can be constructed in $\mathcal{D}_k(\mathcal{E}_h)$ if the mesh contains only triangular elements or tetrahedral elements. Otherwise, these two terms can be bounded using the trace inequalities (2.1), (2.5) and the approximation results (2.10). We remark that the bound for T_4 is valid if $\beta \le (d-1)^{-1}$:

$$|T_3+T_4| \le \frac{\kappa}{4}\|\boldsymbol{\chi}\|_{\mathcal{E}}^2 + Ch^{2\min(k+1,s)-2}|||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)}^2.$$

From the bounds above, we conclude that

$$\|\chi\|_{\mathcal{E}} \le Ch^{\min(k+1,s)-1}|||\boldsymbol{u}|||_{H^s(\mathcal{E}_h)},$$

which, with the triangle inequality, yields (5.9). □

Remark: If one wants to use a larger discrete space on quadrilaterals or hexahedra, namely the space $\mathbb{Q}_k$, then the a priori error estimate is valid for all $\beta \ge (d-1)^{-1}$.

5.5 Bibliographical remarks

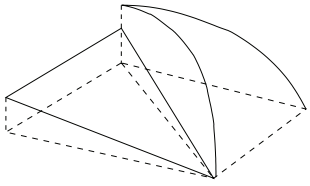
Primal DG methods for linear elasticity have been studied in [111, 110, 66, 90]. Mixed DG methods are considered in [71].

Exercises

5.1. Prove that if $\boldsymbol{u} \in H^2(\Omega)^d$ satisfies problem (5.5)–(5.7), then $\boldsymbol{u}$ satisfies problem (5.8).

5.2. Complete the proof of the coercivity of the bilinear form a_η.

5.3. Derive an L^2 estimate for the numerical error for the SIPG method.



Chapter 6
Stokes flow

This chapter and the following one deal with CFD applications. The fluid flow is characterized by either the Stokes equations or the Navier–Stokes equations. The domain is two-dimensional, but the numerical methods and the analysis can be generalized to three-dimensional domains.

6.1 Preliminaries

6.1.1 Vector notation

The gradient of a vector function $\boldsymbol{v} : \mathbb{R}^d \to \mathbb{R}^d$ is a matrix, and the divergence of a matrix function $\boldsymbol{A} : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ is a vector:

$$\nabla \boldsymbol{v} = \left(\frac{\partial v_i}{\partial x_j}\right)_{1\le i,j\le d}, \quad \nabla \cdot \boldsymbol{A} = \left(\sum_{j=1}^{d} \frac{\partial a_{ij}}{\partial x_j}\right)_{1\le i\le d}.$$

Consequently, we have for a vector function $\boldsymbol{v} = (v_i)_{1\le i\le d}$

$$\Delta \boldsymbol{v} = \nabla \cdot \nabla \boldsymbol{v} = (\Delta v_i)_{1\le i\le d}.$$

The L^2 inner product of two matrix functions $\boldsymbol{A}$, $\boldsymbol{B}$ is defined by

$$(\boldsymbol{A}, \boldsymbol{B})_\Omega = \int_\Omega \boldsymbol{A} : \boldsymbol{B} = \int_\Omega \sum_{1\le i,j\le d} A_{ij} B_{ij}.$$

6.1.2 Barycentric coordinates

Let E be a triangle with vertices A_1, A_2, A_3 and let λ_1, λ_2, λ_3 be the corresponding barycentric coordinates of a point M in E. The point M is the common vertex to three triangles E_1, E_2, E_3 whose union forms E (see Fig. 6.1). For instance, the vertices of triangle E_1 are the points M, A_2, A_3. Let $|E|$ denote the area of E. The barycentric coordinates are

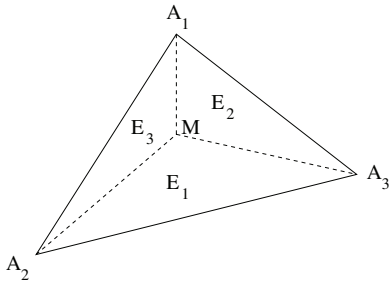


Figure 6.1. *Barycentric coordinates.*

defined by the ratio of two areas:

$$\lambda_i(M) = \frac{|E_i|}{|E|}.$$

Denote the edges of E by e_i such that $e_1 = [A_2, A_3], e_2 = [A_3, A_1], e_3 = [A_1, A_2]$ and denote the midpoint of e_i by B_i. Clearly, we have

$$\lambda_i(A_j) = \begin{cases} 1 & \text{for } i = j, \\ 0 & \text{for } i \neq j, \end{cases} \qquad \lambda_i(B_j) = \begin{cases} 0 & \text{for } i = j, \\ \frac{1}{2} & \text{for } i \neq j. \end{cases}$$

The next result is an application of the Gauss quadrature rule for $Q_G = 1$ given in Appendix A.

Lemma 6.1. *Let E be a triangle and let e denote one edge of E with midpoint B. Then, for all $v \in \mathbb{P}_1(E)$, we have*

$$\int_e v = v(B)|e|. \tag{6.1}$$

We now construct a basis of $\mathbb{P}_1(E)$ from the barycentric coordinates.

Lemma 6.2.

$$\mathbb{P}_1(E) = \operatorname{span}(1 - 2\lambda_1, 1 - 2\lambda_2, 1 - 2\lambda_3).$$

Proof. Since $\dim(\mathbb{P}_1) = 3$, it suffices to show that these functions are linearly independent. Assume that there are coefficients $\alpha_1, \alpha_2, \alpha_3$ such that

$$\sum_{i=1}^{3} \alpha_i(1 - 2\lambda_i) = 0.$$

Fix an edge e_k and integrate over e_k:

$$0 = \int_{e_k} \sum_{i=1}^{3} \alpha_i(1 - 2\lambda_i) = \sum_{i=1}^{3} \alpha_i|e_k|(1 - 2\lambda_i)(B_k) = \alpha_k|e_k|.$$

Thus, $\alpha_k = 0$ for $k = 1, \ldots, 3$. □

Therefore, any linear v can be written as

$$v = \sum_{i=1}^{3} v_i(1 - 2\lambda_i), \quad v_i \in \mathbb{R}.$$

The coefficients v_i are obtained by evaluating v at the midpoint B_i. Hence, we have

$$\forall v \in \mathbb{P}_1(E), \quad v = \sum_{i=1}^{3} v(B_i)(1 - 2\lambda_i). \tag{6.2}$$

6.1.3 An approximation operator of degree one

Let $\mathcal{E}_h$ denote a triangular mesh of a bounded domain Ω (see Section 2.3). We first define a local operator $\pi : H^1(E) \to \mathbb{P}_1(E)$ for any given mesh element.

We fix a triangle E with edges e_1, e_2, e_3, and we define, for any $v \in H^1(E)$, the polynomial $\pi v \in \mathbb{P}_1(E)$ such that

$$\int_{e_k} \pi v = \int_{e_k} v, \quad k = 1, 2, 3.$$

This uniquely defines πv because, from (6.2), it suffices to determine $\pi v(B_k)$ for $k = 1, 2, 3$ (with B_k being the midpoint of the edge e_k). From (6.1), we obtain

$$\int_{e_k} v = \int_{e_k} \pi v = |e_k| \pi v(B_k).$$

Equivalently,

$$\pi v(B_k) = \frac{1}{|e_k|} \int_{e_k} v.$$

The degrees of freedom of the linear πv are associated with the midpoints of the edges of E and defined by $\frac{1}{|e_k|} \int_{e_k} v$.

Furthermore, we have

$$\forall v \in \mathbb{P}_1(E), \quad \pi v = v.$$

Indeed, from (6.2), we have

$$\pi v - v = \sum_{i=1}^{3} (\pi v(B_i) - v(B_i))(1 - 2\lambda_i)$$

and

$$\pi v(B_i) = \frac{1}{|e_i|} \int_{e_i} v = \frac{1}{|e_i|} |e_i| v(B_i) = v(B_i).$$

Let us now define the approximation operator $\boldsymbol{R} : H^1(\Omega)^2 \to \mathcal{D}_1(\mathcal{E}_h)$ such that

$$\forall \boldsymbol{v} = (v_1, v_2) \in H^1(\Omega)^2, \quad \forall E \in \mathcal{E}_h, \quad \boldsymbol{R}\boldsymbol{v}|_E = (\pi v_1, \pi v_2).$$

In other words, if e denotes any edge in the mesh, we have

$$\int_e \boldsymbol{R}\boldsymbol{v} = \int_e \boldsymbol{v}. \tag{6.3}$$

The operator $\boldsymbol{R}$ is called the Crouzeix–Raviart operator [39]. The function $\boldsymbol{R}\boldsymbol{v}$ is discontinuous across the edges except at the midpoints.

Lemma 6.3. *The operator $\boldsymbol{R}$ satisfies*

$$\begin{aligned}
\forall \boldsymbol{v} \in H^1(\Omega)^2, \quad & \int_E \nabla \cdot (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) = 0, \\
\forall \boldsymbol{v} \in H^1(\Omega)^2, \quad \forall e \in \Gamma_h, \quad & \int_e [\boldsymbol{R}(\boldsymbol{v})] = \boldsymbol{0}, \\
\forall \boldsymbol{v} \in H_0^1(\Omega)^2, \quad \forall e \in \partial\Omega, \quad & \int_e \boldsymbol{R}\boldsymbol{v} = \boldsymbol{0}, \\
\forall \boldsymbol{v} \in H^2(\Omega)^2, \quad & \|\nabla(\boldsymbol{v} - \boldsymbol{R}\boldsymbol{v})\|_{L^2(E)} \leq C h_E \|\nabla^2 \boldsymbol{v}\|_{L^2(E)}.
\end{aligned}$$

Proof. The first equality is proved by Green's formula and by (6.3):

$$\int_E \nabla(\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) = \int_{\partial E} (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) \cdot \boldsymbol{n}_E = \sum_{e \in \partial E} \left(\int_e (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) \right) \cdot \boldsymbol{n}_E = 0.$$

The second result is trivial:

$$\int_e [\boldsymbol{R}\boldsymbol{v}] = \int_e [\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}] = \int_e (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v})|_{E_1} - \int_e (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v})|_{E_2} = 0.$$

The result is similar for the third equality:

$$\int_e \boldsymbol{R}\boldsymbol{v} = \int_e (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) = \boldsymbol{0}.$$

Finally, the last result holds true because $\boldsymbol{R}\boldsymbol{v} = \boldsymbol{v}$ if $\boldsymbol{v} \in \mathbb{P}_1(E)^2$ (see [59]). □

6.1.4 An approximation operator of higher degree

There exists a similar operator $\boldsymbol{R}$ such that $\boldsymbol{R}\boldsymbol{v}|_E \in \mathbb{P}_k(E)^2$ for all triangles E and for $k = 2$ and $k = 3$ (see [56, 38, 63]). This operator satisfies for any E

$$\forall \boldsymbol{v} \in H^1(\Omega)^2, \quad \forall q \in \mathbb{P}_{k-1}(E), \quad \int_E q \nabla \cdot (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) = 0, \tag{6.4}$$

$$\forall \boldsymbol{v} \in H^1(\Omega)^2, \quad \forall e \in \Gamma_h, \quad \forall \boldsymbol{q} \in \mathbb{P}_{k-1}(e)^2, \quad \int_e \boldsymbol{q} \cdot [\boldsymbol{R}\boldsymbol{v}] = 0, \tag{6.5}$$

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^2, \quad \forall e \in \partial\Omega, \quad \forall \boldsymbol{q} \in \mathbb{P}_{k-1}(e)^2, \quad \int_e \boldsymbol{q} \cdot \boldsymbol{R}\boldsymbol{v} = 0, \tag{6.6}$$

$$\forall s \in [1, k+1], \quad \forall \boldsymbol{v} \in H^s(\Omega)^2, \quad \|\boldsymbol{v} - \boldsymbol{R}\boldsymbol{v}\|_{L^2(\Omega)} + h_E \|\nabla(\boldsymbol{v} - \boldsymbol{R}\boldsymbol{v})\|_{L^2(E)} \leq C h_E^s |\boldsymbol{v}|_{H^s(\Delta_E)}, \tag{6.7}$$

where Δ_E is a suitable macroelement containing E. Furthermore, each triangle $E \in E_h^i$ has at least one side e such that

$$\forall \boldsymbol{v} \in H^1(\Omega)^2, \quad \int_e (\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) = \boldsymbol{0}. \tag{6.8}$$

6.1.5 Local L^2 projection

We fix a mesh element E. Let $p \in H^s(E)$ and let $\tilde{p}$ denote the L^2 projection of p onto $\mathbb{P}_k(E)$ defined by

$$\forall v \in \mathbb{P}_k(E), \quad \int_E (p - \tilde{p}) v = 0.$$

Then, there is a constant C independent of h_E such that

$$\|p - \tilde{p}\|_{L^2(E)} + h_E \|\nabla(p - \tilde{p})\|_{L^2(E)} \leq C h_E^{\min(k+1,s)} \|p\|_{H^s(E)}.$$

6.1.6 General inf-sup condition

We present the inf-sup condition in a general setting first [59]. Let $b : X \times M \to \mathbb{R}$ be a continuous bilinear form defined on two Hilbert spaces X and M. Let $\|\cdot\|_X$ (resp., $\|\cdot\|_Y$) and $(\cdot,\cdot)_X$ (resp., $(\cdot,\cdot)_Y$) denote the norm and inner product on X (resp., Y). The spaces X and M satisfy an inf-sup condition [7, 18] if there is a constant $\beta > 0$ such that

$$\inf_{q \in M} \sup_{v \in X} \frac{b(v,q)}{\|q\|_M \|v\|_X} \geq \beta. \tag{6.9}$$

We denote by X' and M' the dual spaces of X and M. We define the mappings $B : X \to M'$ and $B' : M \to X'$ by

$$\forall v \in X, \quad \forall q \in M, \quad Bv(q) = B'q(v) = b(v,q).$$

We also define the kernel of B, its orthogonal set, and its polar set:

$$V = \mathrm{Ker}(B) = \{v \in X : \forall q \in M,\ b(v,q) = 0\},$$
$$V^\perp = \{w \in X : \forall v \in V,\ (w,v)_X = 0\},$$
$$V^\circ = \{\phi \in X' : \forall v \in V,\ \phi(v) = 0\}.$$

Lemma 6.4. *The following statements are equivalent.*

(i) *The* inf-sup *condition* (6.9) *holds true.*

(ii) *The mapping B is an isomorphism from $V^\perp$ onto M' and*

$$\forall v \in V^\perp, \quad \|Bv\|_{M'} \geq \beta \|v\|_X.$$

(iii) *The mapping B' is an isomorphism from M onto V° and*

$$\forall q \in M, \quad \|B'q\|_{X'} \geq \beta \|q\|_M.$$

Let $\Omega \subset \mathbb{R}^2$ be a bounded polygonal domain. We now give an example of spaces that satisfy an inf-sup condition, namely the spaces $H_0^1(\Omega)$ and $L_0^2(\Omega)$. The space $L_0^2(\Omega)$ is the space of square-integrable functions with zero average:

$$L_0^2(\Omega) = \left\{ v \in L^2(\Omega) : \quad \int_\Omega v = 0 \right\}.$$

There exists a positive constant β such that

$$\inf_{q \in L_0^2(\Omega)} \sup_{v \in H_0^1(\Omega)^2} \frac{(\nabla \cdot v, q)_\Omega}{\|q\|_{L^2(\Omega)} \|\nabla v\|_{L^2(\Omega)}} \geq \beta. \tag{6.10}$$

Equivalently, from statement (ii), for any $q \in L_0^2(\Omega)$, there is a function $v \in H_0^1(\Omega)^2$ such that

$$\nabla \cdot v = q, \quad \|\nabla v\|_{L^2(\Omega)} \leq \frac{1}{\beta} \|q\|_{L^2(\Omega)}.$$

6.2 Model problem and weak solution

Let Ω be a bounded polygonal domain in $\mathbb{R}^2$. The Stokes equations for an incompressible viscous fluid confined in Ω are

$$-\mu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in} \quad \Omega, \tag{6.11}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in} \quad \Omega, \tag{6.12}$$

$$\boldsymbol{u} = \boldsymbol{0} \quad \text{on} \quad \partial\Omega. \tag{6.13}$$

The unknown variables are the fluid velocity $\boldsymbol{u}$ and the fluid pressure p. The constant $\mu > 0$ is the fluid viscosity; the function $\boldsymbol{f}$ is a body force acting on the fluid. Equation (6.11) is referred to as the momentum equation, whereas (6.12) is the incompressibility equation (or continuity equation). Since p is uniquely defined up to an additive constant, we also assume that $\int_\Omega p = 0$. If we assume that $\boldsymbol{f} \in L^2(\Omega)^2$, a weak solution to (6.11)–(6.13) is the pair $(\boldsymbol{u}, p) \in H_0^1(\Omega)^2 \times L_0^2(\Omega)$ satisfying

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^2, \quad \mu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_\Omega - (\nabla \cdot \boldsymbol{v}, p)_\Omega = (\boldsymbol{f}, \boldsymbol{v})_\Omega, \tag{6.14}$$

$$\forall q \in L_0^2(\Omega), \quad (\nabla \cdot \boldsymbol{u}, q)_\Omega = 0. \tag{6.15}$$

The space of divergence-free vector functions is defined by

$$\boldsymbol{V} = \{\boldsymbol{v} \in H_0^1(\Omega)^2 : \forall q \in L_0^2(\Omega), (\nabla \cdot \boldsymbol{v}, q)_\Omega = 0\}.$$

The space $\boldsymbol{V}$ is equipped with the norm $\boldsymbol{v} \mapsto \|\nabla \boldsymbol{v}\|_{L^2(\Omega)}$. Clearly, if $(\boldsymbol{u}, p)$ is a weak solution satisfying (6.14), (6.15), then $\boldsymbol{u}$ is a solution to the following problem:

$$\forall \boldsymbol{v} \in \boldsymbol{V}, \quad \mu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_\Omega = (\boldsymbol{f}, \boldsymbol{v})_\Omega. \tag{6.16}$$

One can check that the bilinear form $(\boldsymbol{v}, \boldsymbol{w}) \mapsto \mu(\nabla \boldsymbol{v}, \nabla \boldsymbol{w})_\Omega$ is continuous and coercive and that the linear form $\boldsymbol{v} \mapsto (\boldsymbol{f}, \boldsymbol{v})_\Omega$ is continuous. Lax–Milgram's theorem (Theorem 2.8) implies that there is a unique $\boldsymbol{u} \in \boldsymbol{V}$ satisfying (6.16). Next, we consider the mapping

$$\Phi : \boldsymbol{v} \mapsto \mu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})_\Omega - (\boldsymbol{f}, \boldsymbol{v})_\Omega.$$

The mapping Φ belongs to the dual space of $H_0^1(\Omega)^2$ and vanishes on the space $\boldsymbol{V}$ since $\boldsymbol{u}$ satisfies (6.16). Therefore, Φ belongs to the polar space $\boldsymbol{V}^\circ$. From the inf-sup condition (6.10) and from (ii) in Lemma 6.4, there is a unique $p \in L_0^2(\Omega)$ satisfying

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^2, \quad (\nabla \cdot \boldsymbol{v}, p) = \Phi(\boldsymbol{v}).$$

This is equivalent to (6.14). Thus, we have proved that there is a unique weak solution to (6.11)–(6.13).

6.3 DG scheme

Let $\mathcal{E}_h$ be a mesh of Ω as defined in Section 2.3. For any integer $k \geq 1$, we define the discrete velocity and pressure spaces:

$$\boldsymbol{X}_h = \{\boldsymbol{v} \in L^2(\Omega)^2 : \forall E \in \mathcal{E}_h, \boldsymbol{v} \in (\mathbb{P}_k(E))^2\},$$
$$M_h = \{q \in L_0^2(\Omega) : \forall E \in \mathcal{E}_h, q \in \mathbb{P}_{k-1}(E)\}.$$

We introduce the bilinear forms $a_\epsilon : \boldsymbol{X}_h \times \boldsymbol{X}_h \to \mathbb{R}$ and $b : \boldsymbol{X}_h \times M_h \to \mathbb{R}$ corresponding to DG discretizations of the diffusive term $-\Delta \boldsymbol{u}$ and the pressure term ∇p, respectively:

$$a_\epsilon(\boldsymbol{w}, \boldsymbol{v}) = \sum_{E \in \mathcal{E}_h} \int_E \nabla \boldsymbol{w} : \nabla \boldsymbol{v} + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|} \int_e [\boldsymbol{w}] \cdot [\boldsymbol{v}]$$
$$- \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{\nabla \boldsymbol{w}\} \boldsymbol{n}_e \cdot [\boldsymbol{v}] + \epsilon \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{\nabla \boldsymbol{v}\} \boldsymbol{n}_e \cdot [\boldsymbol{w}], \tag{6.17}$$

$$b(\boldsymbol{v}, q) = - \sum_{E \in \mathcal{E}_h} \int_E q \nabla \cdot \boldsymbol{v} + \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{q\} [\boldsymbol{v}] \cdot \boldsymbol{n}_e. \tag{6.18}$$

As usual, the choice of the parameter ϵ will yield the NIPG method ($\epsilon = 1$), the SIPG method ($\epsilon = -1$), or the IIPG method ($\epsilon = 0$). The penalty parameter is denoted by σ_e^0 for an edge e and is strictly positive. For simplicity, we do not assume superpenalization. The derivation of the bilinear form a_ϵ is similar to the one for the elliptic problem. We now give some details on the form b. Using Green's theorem on one mesh element E, we have

$$\int_E \nabla p \cdot \boldsymbol{v} = - \int_E p \nabla \cdot \boldsymbol{v} + \int_{\partial E} p \boldsymbol{v} \cdot \boldsymbol{n}_E.$$

We sum over all mesh elements and use the normal vector $\boldsymbol{n}_e$ fixed for each edge:

$$\sum_{E \in \mathcal{E}_h} \int_E \nabla p \cdot \boldsymbol{v} = - \sum_{E \in \mathcal{E}_h} \int_E p \nabla \cdot \boldsymbol{v} + \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e [p \boldsymbol{v} \cdot \boldsymbol{n}_e].$$

Finally, since p is continuous, we have $p|_e = \{p\}|_e$, and we can write

$$\sum_{E \in \mathcal{E}_h} \int_E \nabla p \cdot \boldsymbol{v} = - \sum_{E \in \mathcal{E}_h} \int_E p \nabla \cdot \boldsymbol{v} + \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e \{p\} [\boldsymbol{v} \cdot \boldsymbol{n}_e],$$

which is exactly the expression $b(\boldsymbol{v}, p)$.

With these spaces and bilinear forms, the numerical method is as follows: Find $(\boldsymbol{U}_h, P_h) \in \boldsymbol{X}_h \times M_h$ such that

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad \mu a_\epsilon(\boldsymbol{U}_h, \boldsymbol{v}) + b(\boldsymbol{v}, P_h) = (\boldsymbol{f}, \boldsymbol{v})_\Omega, \tag{6.19}$$

$$\forall q \in M_h, \quad b(\boldsymbol{U}_h, q) = 0. \tag{6.20}$$

Next, we state a consistency result and a coercivity result. The proofs are omitted, as they are similar to the proofs given in the previous chapters.

Lemma 6.5. *Assume that the weak solution $(\boldsymbol{u}, p)$ also belongs to $H^2(\mathcal{E}_h)^2 \times H^1(\mathcal{E}_h)$. Then, it satisfies the scheme* (6.19)–(6.20).

We define the energy norm for the Stokes problem:

$$\forall \boldsymbol{v} \in H^1(\mathcal{E}_h)^2, \quad \|\boldsymbol{v}\|_{\mathcal{E}} = \left(\sum_{E \in \mathcal{E}_h} \|\nabla \boldsymbol{v}\|^2_{L^2(E)} + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|} \|[\boldsymbol{v}]\|^2_{L^2(e)} \right)^{1/2}. \tag{6.21}$$

Lemma 6.6. *Assume that σ_0^e is sufficiently large if $\epsilon = -1$ or $\epsilon = 0$. Then, there is a constant $\kappa > 0$ independent of h such that*

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad a_\epsilon(\boldsymbol{v}, \boldsymbol{v}) \geq \kappa \|\boldsymbol{v}\|^2_{\mathcal{E}}.$$

6.3.1 Existence and uniqueness of solution

Since problem (6.19), (6.20) results in a square system of linear equations in finite dimension, it suffices to prove uniqueness of the solution. Let $\boldsymbol{W}_h$ denote the difference of two solutions. Set the data $\boldsymbol{f} = \boldsymbol{0}$ and choose $\boldsymbol{v} = \boldsymbol{W}_h$ in (6.19). Since $\boldsymbol{W}_h$ satisfies (6.20), we are left with

$$a_\epsilon(\boldsymbol{W}_h, \boldsymbol{W}_h) = 0.$$

The coercivity of a_ϵ yields that $\boldsymbol{W}_h = 0$. Thus, (6.19) is reduced to

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad b(\boldsymbol{v}, P_h) = 0.$$

At this point, one cannot conclude that $P_h = 0$. This result is a consequence of the inf-sup condition established in Section 6.4.

6.3.2 Local mass conservation

In this context, local mass conservation is a consequence of the discretization of the incompressibility equation. We fix an element $E \in \mathcal{E}_h$ and choose a function $q = 1$ on E and zero elsewhere. Equation (6.20) becomes

$$-\int_E \nabla \cdot \boldsymbol{U}_h + \frac{1}{2} \sum_{e \in \partial E \setminus \partial\Omega} \int_e [\boldsymbol{U}_h] \cdot \boldsymbol{n}_e + \sum_{e \in \partial E \cap \partial\Omega} \int_e \boldsymbol{U}_h \cdot \boldsymbol{n}_e = 0.$$

Denoting by $\boldsymbol{n}_E$ the outward normal to E and using Green's formula, we have

$$-\int_{\partial E} \boldsymbol{U}_h \cdot \boldsymbol{n}_E + \frac{1}{2} \sum_{e \in \partial E \setminus \partial\Omega} \int_e [\boldsymbol{U}_h] \cdot \boldsymbol{n}_e + \sum_{e \in \partial E \cap \partial\Omega} \int_e \boldsymbol{U}_h \cdot \boldsymbol{n}_e = 0,$$

or equivalently

$$\sum_{e \in \partial E \setminus \partial\Omega} \int_e \{\boldsymbol{U}_h\} \cdot \boldsymbol{n}_E = 0.$$

We remark that this equation is comparable to the local mass balance satisfied by the exact solution

$$\sum_{e \in \partial E \setminus \partial\Omega} \int_e \boldsymbol{u} \cdot \boldsymbol{n}_E = 0.$$

6.4 Discrete inf-sup condition

In this section, we prove an inf-sup condition for the spaces $\tilde{\boldsymbol{X}}_h$ and M_h, where $\tilde{\boldsymbol{X}}_h$ is a subspace of $\boldsymbol{X}_h$:

$$\tilde{\boldsymbol{X}}_h = \{\boldsymbol{v}_h \in \boldsymbol{X}_h : \forall e \in \Gamma_h \cup \partial\Omega, [\boldsymbol{v}_h]|_e \cdot \boldsymbol{n}_e = 0\}.$$

The proof relies on the Raviart–Thomas interpolant [88, 59, 89] defined in the following lemma.

Lemma 6.7. *The Raviart–Thomas interpolant $\pi : H^1(\Omega)^2 \to \boldsymbol{X}_h$ satisfies for all $\boldsymbol{v} \in H^1(\Omega)^2$*

$$\forall E \in \mathcal{E}_h,\ \forall q \in \mathbb{P}_{k-1}(E),\ \int_E q \nabla \cdot (\pi \boldsymbol{v} - \boldsymbol{v}) = 0, \tag{6.22}$$

$$\forall e \in \Gamma_h \cup \partial\Omega,\ \forall q \in \mathbb{P}_{k-1}(e),\ \int_e q(\pi \boldsymbol{v} - \boldsymbol{v}) \cdot \boldsymbol{n}_e = 0, \tag{6.23}$$

$$\forall e \in \Gamma_h \cup \partial\Omega,\ \pi \boldsymbol{v}|_e \cdot \boldsymbol{n}_e \in \mathbb{P}_{k-1}(e), \tag{6.24}$$

$$\forall E \in \mathcal{E}_h,\ \|\pi \boldsymbol{v} - \boldsymbol{v}\|_{L^2(E)} + h_E \|\nabla(\pi \boldsymbol{v} - \boldsymbol{v})\|_{L^2(E)} \le C h_E \|\nabla \boldsymbol{v}\|_{L^2(E)}, \tag{6.25}$$

$$\|\pi \boldsymbol{v}\|_{\mathcal{E}} \le C \|\nabla \boldsymbol{v}\|_{L^2(\Omega)} \tag{6.26}$$

with a constant C independent of h_E and h.

Theorem 6.8. *There exists a constant $\beta^* > 0$, independent of h, such that*

$$\inf_{q \in M_h} \sup_{\boldsymbol{v} \in \tilde{\boldsymbol{X}}_h} \frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_{\mathcal{E}} \|q\|_{L^2(\Omega)}} \ge \beta^*. \tag{6.27}$$

Proof. We shall prove that for any $q \in M_h$ there exists $\boldsymbol{v}$ in $\tilde{\boldsymbol{X}}_h$ such that

$$b(\boldsymbol{v}, q) \ge \beta_1^* \|q\|_{L^2(\Omega)}^2, \tag{6.28}$$

$$\|\boldsymbol{v}\|_{\mathcal{E}} \le \beta_2^* \|q\|_{L^2(\Omega)} \tag{6.29}$$

with constants $\beta_1^* > 0$ and $\beta_2^* > 0$ independent of h, q, and $\boldsymbol{v}$. Clearly, this will imply (6.27) because

$$\frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_{\mathcal{E}} \|q\|_{L^2(\Omega)}} \geq \beta_1^* \frac{\|q\|_{L^2(\Omega)}^2}{\|\boldsymbol{v}\|_{\mathcal{E}} \|q\|_{L^2(\Omega)}} = \beta_1^* \frac{\|q\|_{L^2(\Omega)}}{\|\boldsymbol{v}\|_{\mathcal{E}}} \geq \frac{\beta_1^*}{\beta_2^*}.$$

Let $q \in M_h$. Since $q \in L_0^2(\Omega)$ and the spaces $H_0^1(\Omega)^2$, $L_0^2(\Omega)$ satisfy the inf-sup condition (6.10), there exists $\tilde{\boldsymbol{v}} \in H_0^1(\Omega)^2$ such that

$$-\nabla \cdot \tilde{\boldsymbol{v}} = q, \quad \|\nabla \tilde{\boldsymbol{v}}\|_{L^2(\Omega)} \leq \frac{1}{\beta} \|q\|_{L^2(\Omega)}. \tag{6.30}$$

Define the Raviart–Thomas interpolant $\tilde{\boldsymbol{v}}_h = \pi \tilde{\boldsymbol{v}}$. Then, $\tilde{\boldsymbol{v}}_h \in \tilde{\boldsymbol{X}}_h$ from (6.23) and (6.24). We also have from (6.22) and (6.25)

$$-\sum_{E \in \mathcal{E}_h} \int_E (\nabla \cdot \tilde{\boldsymbol{v}}_h) q = -\sum_{E \in \mathcal{E}_h} \int_E (\nabla \cdot \tilde{\boldsymbol{v}}) q = \|q\|_{L^2(\Omega)}^2.$$

Therefore, it follows that

$$b(\tilde{\boldsymbol{v}}_h, q) = \|q\|_{L^2(\Omega)}^2.$$

This implies (6.28) with the constant $\beta_1^* = 1$. Property (6.26) and inequality (6.30) imply that

$$\|\tilde{\boldsymbol{v}}_h\|_{\mathcal{E}} \leq C \|q\|_{L^2(\Omega)}.$$

This concludes the proof. □

Remark: Define the subspace of $\boldsymbol{X}_h$:

$$\boldsymbol{V}_h = \{\boldsymbol{v} \in \boldsymbol{X}_h; \forall q \in M_h, b(\boldsymbol{v}, q) = 0\}.$$

We say that $\boldsymbol{V}_h$ is the space of discretely divergence-free functions. An immediate consequence of Lemma 6.4 is that for a given q in M_h there exists a unique $\boldsymbol{v}$ in $\tilde{\boldsymbol{X}}_h$ such that

$$\forall \boldsymbol{w} \in \boldsymbol{V}_h, \quad \sum_{E \in \mathcal{E}_h} \int_E \nabla \boldsymbol{v} : \nabla \boldsymbol{w} + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{\sigma_e^0}{|e|} \int_e [\boldsymbol{v}] \cdot [\boldsymbol{w}] = 0,$$

$$b(\boldsymbol{v}, q) = -\|q\|_{L^2(\Omega)}^2, \quad \|\boldsymbol{v}\|_{\mathcal{E}} \leq \frac{1}{\beta^*} \|q\|_{L^2(\Omega)}.$$

6.5 Error estimates

We prove optimal error estimates for the velocity and pressure. First, we need a lemma on the approximation operator $\boldsymbol{R}$.

Lemma 6.9.

$$\forall \boldsymbol{v} \in H_0^1(\Omega)^2, \quad \forall q \in M_h, \quad b(\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}, q) = 0. \tag{6.31}$$

Proof. Let $\boldsymbol{v} \in H_0^1(\Omega)^2$ and $q \in M_h$. Then,

$$b(\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}, q) = -\sum_{E\in\mathcal{E}_h} \int_E q\nabla\cdot(\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}) + \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{q\}[\boldsymbol{R}\boldsymbol{v} - \boldsymbol{v}]\cdot\boldsymbol{n}_e.$$

The result is then a consequence of (6.4)–(6.6). □

Theorem 6.10. *Assume that the exact solution* $(\boldsymbol{u}, p)$ *belongs to* $H^{k+1}(\Omega)^2 \times H^k(\Omega)$. *Then, the solution* $(\boldsymbol{U}_h, P_h)$ *of* (6.19), (6.20) *satisfies the error estimate*

$$\|\boldsymbol{u} - \boldsymbol{U}_h\|_{\mathcal{E}} \le Ch^k \left(|\boldsymbol{u}|_{H^{k+1}(\Omega)} + \frac{1}{\mu}|p|_{H^k(\Omega)} \right), \tag{6.32}$$

where C *is independent of* h *and* μ.

Proof. Denote $\boldsymbol{\chi} = \boldsymbol{U}_h - \boldsymbol{R}\boldsymbol{u}$ and $\xi = P_h - \tilde{p}$, where $\tilde{p}$ is the L^2 projection of p. The errors $\boldsymbol{\chi}$ and ξ satisfy the equations

$$\begin{aligned} \forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad \mu a_\epsilon(\boldsymbol{\chi}, \boldsymbol{v}) + b(\boldsymbol{v}, \xi) &= \mu a_\epsilon(\boldsymbol{u} - \tilde{\boldsymbol{u}}, \boldsymbol{v}) + b(\boldsymbol{v}, p - \tilde{p}), \\ \forall q \in M_h, \quad b(\boldsymbol{\chi}, q) &= b(\boldsymbol{u} - \tilde{\boldsymbol{u}}, q). \end{aligned}$$

Choosing $\boldsymbol{v} = \boldsymbol{\chi}$, $q = \xi$ and using (6.31), we obtain

$$a_\epsilon(\boldsymbol{\chi}, \boldsymbol{\chi}) = a_\epsilon(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}, \boldsymbol{\chi}) + \frac{1}{\mu} b(\boldsymbol{\chi}, p - \tilde{p}), \tag{6.33}$$

which yields by coercivity of a_ϵ

$$\kappa \|\boldsymbol{\chi}\|_{\mathcal{E}}^2 \le a_\epsilon(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}, \boldsymbol{\chi}) + \frac{1}{\mu} b(\boldsymbol{\chi}, p - \tilde{p}).$$

Then, we need only bound the two terms on the right-hand side. Throughout this chapter, the generic constant C is independent of the mesh size h and the fluid viscosity μ. By definition, we have

$$\begin{aligned} a_\epsilon(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}, \boldsymbol{\chi}) &= \sum_{E\in\mathcal{E}_h} \int_E \nabla(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}) : \nabla\boldsymbol{\chi} - \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\nabla(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u})\}\boldsymbol{n}_e \cdot [\boldsymbol{\chi}] \\ &\quad + \epsilon \sum_{e\in\Gamma_h\cup\partial\Omega} \int_e \{\nabla\boldsymbol{\chi}\}\boldsymbol{n}_e \cdot [\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}] + \sum_{e\in\Gamma_h\cup\partial\Omega} \frac{\sigma_e^0}{|e|} \int_e [\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u}]\cdot[\boldsymbol{\chi}] \\ &= T_1 + T_2 + T_3 + T_4. \end{aligned}$$

Using Cauchy–Schwarz's inequality and the bound (6.7), we have

$$\begin{aligned} T_1 &\le \frac{\kappa}{10} |||\nabla\boldsymbol{\chi}|||_{L^2(\Omega)}^2 + C \sum_{E\in\mathcal{E}_h} \|\nabla(\boldsymbol{u} - \boldsymbol{R}\boldsymbol{u})\|_{L^2(E)}^2 \\ &\le \frac{\kappa}{10} |||\nabla\boldsymbol{\chi}|||_{L^2(\Omega)}^2 + Ch^k |\boldsymbol{u}|_{H^{k+1}(\Omega)}, \\ T_4 &\le \frac{\kappa}{10} |||\nabla\boldsymbol{\chi}|||_{L^2(\Omega)}^2 + Ch^k |\boldsymbol{u}|_{H^{k+1}(\Omega)}. \end{aligned}$$

Similarly,

$$T_2 \le \frac{\kappa}{10} \sum_{e\in\Gamma_h\cup\partial\Omega} \frac{1}{|e|}\|[\boldsymbol{\chi}]\|^2_{L^2(e)} + C \sum_{e\in\Gamma_h\cup\partial\Omega} |e|\|\{\nabla(\boldsymbol{u}-\boldsymbol{Ru})\}\boldsymbol{n}_e\|^2_{L^2(e)}.$$

The term $\|\{\nabla(\boldsymbol{u}-\boldsymbol{Ru})\}\boldsymbol{n}_e\|_{L^2(e)}$ is bounded using trace inequality (2.2). For instance, if the edge e belongs to the element E_e, we have

$$\|\nabla(\boldsymbol{u}-\boldsymbol{Ru})\boldsymbol{n}_e\|_{L^2(e)} \le Ch_E^{-1/2}\Big(\|\nabla(\boldsymbol{u}-\boldsymbol{Ru})\|_{L^2(E)} + h_E\|\nabla^2(\boldsymbol{u}-\boldsymbol{Ru})\|_{L^2(E)}\Big).$$

Now since we do not have an estimate of $\|\nabla^2(\boldsymbol{u}-\boldsymbol{Ru})\|_{L^2(E)}$, we introduce an approximation $\tilde{\boldsymbol{u}}$ of $\boldsymbol{u}$ satisfying (2.10), and we use an inverse inequality (3.6):

$$\begin{aligned}\|\nabla^2(\boldsymbol{u}-\boldsymbol{Ru})\|_{L^2(E} &\le \|\nabla^2(\boldsymbol{u}-\tilde{\boldsymbol{u}})\|_{L^2(E)} + \|\nabla^2(\tilde{\boldsymbol{u}}-\boldsymbol{Ru})\|_{L^2(E)}\\ &\le \|\nabla^2(\boldsymbol{u}-\tilde{\boldsymbol{u}})\|_{L^2(E)} + Ch_E^{-1}\|\nabla(\tilde{\boldsymbol{u}}-\boldsymbol{Ru})\|_{L^2(E)}.\end{aligned}$$

We skip many details (left to the reader), and we obtain

$$T_2 \le \frac{\kappa}{10} \sum_{e\in\Gamma_h\cup\partial\Omega} \frac{1}{|e|}\|[\boldsymbol{\chi}]\|^2_{L^2(e)} + Ch^{2k}|\boldsymbol{u}|^2_{H^{k+1}(\Omega)}.$$

The third term is simply bounded using (2.5) and (6.7):

$$T_3 \le \frac{\kappa}{10}|||\nabla\boldsymbol{\chi}|||^2_{L^2(\Omega)} + Ch^{2k}|\boldsymbol{u}|^2_{H^{k+1}(\Omega)}.$$

Finally, since $\nabla\cdot\boldsymbol{\chi} \in \mathbb{P}_{k-1}(E)^2$, the term involving the pressure reduces to

$$\frac{1}{\mu}b(\boldsymbol{\chi}, p-\tilde{p}) = \frac{1}{\mu}\sum_{e\in\Gamma_h\cup\partial\Omega}\int_e \{p-\tilde{p}\}[\boldsymbol{\chi}]\cdot\boldsymbol{n}_e \le \frac{\kappa}{10}\|\boldsymbol{\chi}\|^2_{\mathcal{E}} + \frac{C}{\mu^2}h^{2k}|p|^2_{H^k(\Omega)}.$$

Combining all the bounds above and using the triangle inequality

$$\|\boldsymbol{u}-\boldsymbol{U}_h\|_{\mathcal{E}} \le \|\boldsymbol{u}-\boldsymbol{Ru}\|_{\mathcal{E}} + \|\boldsymbol{\chi}\|_{\mathcal{E}},$$

we obtain the final result. □

Theorem 6.11. *Under the assumptions and notation of Theorem* 6.10*, there is a constant C independent of h and μ such that*

$$\|p-P_h\|_{L^2(\Omega)} \le Ch^k(\mu|\boldsymbol{u}|_{H^{k+1}(\Omega)} + |p|_{H^k(\Omega)}).$$

Proof. Let $\tilde{p}$ be the L^2 projection of p. We can write the error equation as follows:

$$\forall\boldsymbol{v}\in\boldsymbol{X}_h, \quad a_\epsilon(\boldsymbol{U}_h-\boldsymbol{u},\boldsymbol{v}) + \frac{1}{\mu}b(\boldsymbol{v}, P_h-\tilde{p}) = \frac{1}{\mu}b(\boldsymbol{v}, p-\tilde{p}). \tag{6.34}$$

From the remark in Section 6.4, there exists $\boldsymbol{v}\in\tilde{\boldsymbol{X}}_h$ such that

$$b(\boldsymbol{v}, P_h-\tilde{p}) = -\|P_h-\tilde{p}\|^2_{L^2(\Omega)}, \quad \|\boldsymbol{v}\|_{\mathcal{E}} \le \frac{1}{\beta^*}\|P_h-\tilde{p}\|_{L^2(\Omega)}, \tag{6.35}$$

and in particular

$$\sum_{E\in\mathcal{E}_h}\int_E \nabla(\boldsymbol{U}_h-\boldsymbol{Ru}):\nabla\boldsymbol{v}+\sum_{e\in\Gamma_h\cup\partial\Omega}\frac{\sigma_e^0}{|e|}\int_e[\boldsymbol{U}_h-\boldsymbol{Ru}]\cdot[\boldsymbol{v}]=0.$$

Therefore, (6.34) becomes

$$\begin{aligned}\frac{1}{\mu}\|P_h-\tilde{p}\|^2_{L^2(\Omega)} &= a_\epsilon(\boldsymbol{U}_h-\boldsymbol{u},\boldsymbol{v})-\frac{1}{\mu}b(\boldsymbol{v},p-\tilde{p})\\ &= \sum_{E\in\mathcal{E}_h}\int_E\nabla(\boldsymbol{Ru}-\boldsymbol{u}):\nabla\boldsymbol{v}+\sum_{e\in\Gamma_h\cup\partial\Omega}\frac{\sigma_e^0}{|e|}\int_e[\boldsymbol{Ru}-\boldsymbol{u}]\cdot[\boldsymbol{v}]-\frac{1}{\mu}b(\boldsymbol{v},p-\tilde{p})\\ &\quad-\sum_{e\in\Gamma}\int_e\{\nabla(\boldsymbol{U}-\boldsymbol{u})\}\boldsymbol{n}_e\cdot[\boldsymbol{v}]+\epsilon\sum_{e\in\Gamma}\int_e\{\nabla\boldsymbol{v}\}\boldsymbol{n}_e\cdot[\boldsymbol{U}-\boldsymbol{u}].\end{aligned}$$

The terms on the right-hand side can be easily bounded, and we obtain the final result. □

We now state the a priori error estimate for the velocity in the L^2 norm. The proof uses the Aubin–Nitsche lift technique as in the proof of Theorem 2.14, and thus it is omitted. The estimate is optimal for the SIPG method and suboptimal for the IIPG and NIPG methods.

Theorem 6.12. *Assume that Ω is convex. Then, under the hypotheses of Theorem* 6.10*, there exists a constant C independent of h and μ such that*

$$\|\boldsymbol{u}-\boldsymbol{U}\|_{L^2(\Omega)}\le Ch^{k+1-\delta}\left(|\boldsymbol{u}|_{H^{k+1}(\Omega)}+\frac{1}{\mu}|p|_{H^k(\Omega)}\right),\tag{6.36}$$

where $\delta=0$ for SIPG and $\delta=0$ for IIPG and NIPG.

6.6 Numerical results

We solve (6.19)–(6.20) in the case of a known smooth function. We vary the polynomial degree k from 1 to 3. The domain $\Omega=(0,1)^2$ is subdivided into 2048 triangles. We assume that the penalty parameter takes the same value for all edges e. Numerical errors for the velocity and pressure are given in Table 6.1 and Table 6.2, respectively. Rates are computed from errors obtained on two successive meshes. In the SIPG case, we obtain optimal error estimates as predicted by the theory. In the NIPG case with positive penalty, we obtain optimal error estimates for the velocity in the energy norm and the pressure in the L^2 norm. We also observe, as for the elliptic problem, optimal convergence rates for the velocity in the L^2 norm if the polynomial degree is odd and suboptimal if the polynomial degree is even. Finally, we add convergence rates for the NIPG 0 method. This method numerically converges for $k\ge 2$.

6.7 Bibliographical remarks

Primal DG methods for Stokes are introduced in [107, 61]. A method using discontinuous divergence-free approximations of the velocity and continuous approximations of the pressure is described in [11]. Mixed DG methods for Stokes are analyzed and studied in [34, 100]: an additional unknown, namely the gradient of velocity, is introduced.

Table 6.1. *Numerical errors and convergence rates for Stokes velocity.*

Method	k	σ_e^0	$\|\|\|\nabla(\boldsymbol{u}-\boldsymbol{U}_h)\|\|\|_{H^0(\mathcal{E}_h)}$	Rate	$\|\boldsymbol{u}-\boldsymbol{U}_h\|_{L^2(\Omega)}$	Rate
NIPG	1	1	5.8810×10^{-03}	1.0259	7.6486×10^{-05}	2.0013
	2	1	1.3406×10^{-04}	2.0041	4.7542×10^{-06}	1.9699
	3	1	3.6084×10^{-06}	2.9843	1.1940×10^{-08}	3.9685
SIPG	1	10	4.1955×10^{-03}	1.0345	6.8338×10^{-05}	1.8462
	2	10	1.3995×10^{-04}	2.0251	3.8299×10^{-07}	3.0411
	3	10	7.4763×10^{-06}	3.4034	1.7002×10^{-08}	4.4039
IIPG	1	10	4.1446×10^{-03}	1.0159	4.8448×10^{-05}	1.8866
	2	10	1.2701×10^{-04}	2.0012	1.8436×10^{-06}	2.0620
	3	10	3.2272×10^{-06}	3.0023	9.2767×10^{-09}	3.9536
NIPG	2	0	1.4465×10^{-04}	2.0093	5.8801×10^{-06}	1.9718
	3	0	3.9253×10^{-06}	2.9767	1.3147×10^{-08}	3.9715

Table 6.2. *Numerical errors and convergence rates for Stokes pressure.*

Method	k	σ_e^0	$\|p-P_h\|_{L^2(\Omega)}$	Rate
NIPG	1	1	9.8746×10^{-3}	1.0248
	2	1	5.1239×10^{-5}	2.2044
	3	1	2.9978×10^{-6}	2.9778
SIPG	1	10	2.5143×10^{-2}	0.9874
	2	10	5.7527×10^{-5}	1.8978
	3	10	1.7230×10^{-6}	3.3224
IIPG	1	10	2.3386×10^{-2}	0.9863
	2	10	4.7975×10^{-5}	1.9923
	3	10	1.5089×10^{-6}	2.9383
NIPG	2	0	6.5898×10^{-5}	2.1872
	3	0	3.6788×10^{-6}	2.9465

Exercises

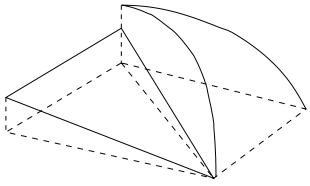
6.1. Prove Lemma 6.1.

6.2. Prove Lemma 6.5.

6.3. Let $(\phi_1, \ldots, \phi_{N_u})$ be a basis of $\boldsymbol{X}_h$ and let $(\psi_1, \ldots, \psi_{N_p})$ be a basis of M_h. Let ξ_i's and η_i's denote the coefficients of the solutions $\boldsymbol{U}_h$ and P_h, respectively, with respect to the basis functions ϕ_i's and ψ_i's. Derive the linear system resulting from (6.19)–(6.20) of the form $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ if the unknown vector is

$$\boldsymbol{x} = (\xi_1, \ldots, \xi_{N_u}, \eta_1, \ldots, \eta_{N_p})^T.$$

6.4. Prove Theorem 6.12.



Chapter 7
Navier–Stokes flow

The Navier–Stokes equations differ from the Stokes equations by the addition of a nonlinear term in the momentum balance. We propose a DG discretization of the nonlinear term based on an upwind technique. As in the previous chapter, only two-dimensional domains are considered.

7.1 Preliminaries

7.1.1 Sobolev imbedding

The space $L^2(\Omega)$ is an example of the space $L^r(\Omega)$:

$$L^r(\Omega) = \left\{ v \text{ measurable } : \int_\Omega |v|^r < \infty \right\}.$$

For $1 \leq r \leq \infty$, this space is a Banach space with the norm

$$\|v\|_{L^r(\Omega)} = \left(\int_\Omega |v|^r \right)^{1/r}.$$

If $\Omega \subset \mathbb{R}^2$, the space $H_0^1(\Omega)$ is imbedded into $L^r(\Omega)$ for any real number $r < \infty$:

$$\forall v \in H_0^1(\Omega), \quad \|v\|_{L^r(\Omega)} \leq C \|\nabla v\|_{L^2(\Omega)}. \tag{7.1}$$

A generalization of this Sobolev imbedding to the broken space $H^1(\mathcal{E}_h)$ is [61]

$$\forall v \in H^1(\mathcal{E}_h), \quad \|v\|_{L^r(\Omega)} \leq C \left(|||\nabla v|||^2_{H^0(\mathcal{E}_h)} + \sum_{e \in \Gamma_h \cup \partial\Omega} \frac{1}{|e|} \|[v]\|^2_{L^2(e)} \right)^{1/2}, \tag{7.2}$$

where C is a constant independent of h. When $r = 2$, we recover Poincaré's inequality (3.5).

7.1.2 Hölder's inequality

Let $1 \leq p, q \leq \infty$ such that $\frac{1}{p} + \frac{1}{q} = 1$. Then, if $u \in L^p(\Omega)$ and $v \in L^q(\Omega)$, the product uv belongs to $L^1(\Omega)$, and we have

$$\int_\Omega |uv| \leq \|u\|_{L^p(\Omega)} \|v\|_{L^q(\Omega)}.$$

When $p = q = 2$, we recover Cauchy–Schwarz's inequality (2.14). Hölder's inequality implies that for any $r > 2$

$$\forall u \in L^r(\Omega), \quad \forall v \in L^{\frac{2r}{r-2}}(\Omega), \quad \|uv\|_{L^2(\Omega)} \leq \|u\|_{L^r(\Omega)} \|v\|_{L^{\frac{2r}{r-2}}(\Omega)}.$$

7.1.3 Brouwer's fixed point theorem

Theorem 7.1. *Let K denote a nonvoid convex and compact subset of a finite-dimensional space and let F be a continuous mapping from K into K. Then, F has at least one fixed point.*

A consequence of this theorem is the following lemma (see [59]).

Lemma 7.2. *Let H be a finite-dimensional Hilbert space whose scalar product is denoted by $(\cdot, \cdot)_H$ and the corresponding norm is $\| \cdot \|_H$. Let Ψ be a continuous mapping from H into H with the following property: there exists $M > 0$ such that*

$$\forall v \in H, \quad \|v\|_H = M, \quad (\Psi(v), v) \geq 0.$$

Then, there exists an element v_0 in H such that

$$\Psi(v_0) = 0, \quad \|v_0\|_H \leq M.$$

Proof. The proof proceeds by contradiction. Suppose $\Psi(v) \neq 0$ in the closed sphere $S = \{v \in H : \|v\|_H \leq M\}$. Then, the mapping

$$F(v) = -M\Psi(v)/\|\Psi(v)\|_H$$

is continuous from S into S. Since H is finite-dimensional and S is nonempty convex and compact, we apply the Brouwer fixed point theorem and obtain the existence of $v_0 \in S$ such that

$$v_0 = -M\Psi(v_0)/\|\Psi(v_0)\|_H.$$

Thus, we have constructed an element of H such that $\|v_0\|_H = M$ and

$$(\Psi(v_0), v_0)_H = -M\|\Psi(v_0)\|_H < 0.$$

This contradicts the assumption. □

7.2 Model problem and weak solution

Let Ω be a bounded polygonal domain in $\mathbb{R}^2$. The Navier–Stokes equations for an incompressible fluid confined in Ω are

$$-\mu\Delta\boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in} \quad \Omega, \tag{7.3}$$

$$\nabla\cdot\boldsymbol{u} = 0 \quad \text{in} \quad \Omega, \tag{7.4}$$

$$\boldsymbol{u} = \boldsymbol{0} \quad \text{on} \quad \partial\Omega. \tag{7.5}$$

As in Section 6.2, the unknown variables are the fluid velocity $\boldsymbol{u}$ and pressure p. The data consists of the fluid viscosity $\mu > 0$ and the body force $\boldsymbol{f}$. Problem (7.3)–(7.5) differs from the Stokes equations in the addition of a nonlinear convective term $\boldsymbol{u}\cdot\nabla\boldsymbol{u}$ in the momentum equation. If the components of $\boldsymbol{u}$ are u_1 and u_2, this term is defined as

$$\boldsymbol{u}\cdot\nabla\boldsymbol{u} = u_1\frac{\partial\boldsymbol{u}}{\partial x_1} + u_2\frac{\partial\boldsymbol{u}}{\partial x_2}.$$

A weak solution to (7.3)–(7.5) is the pair $(\boldsymbol{u}, p) \in H_0^1(\Omega)^2 \times L_0^2(\Omega)$ satisfying

$$\forall\boldsymbol{v} \in H_0^1(\Omega)^2, \quad \mu(\nabla\boldsymbol{u}, \nabla\boldsymbol{v})_\Omega + (\boldsymbol{u}\cdot\nabla\boldsymbol{u}, \boldsymbol{v})_\Omega - (\nabla\cdot\boldsymbol{v}, p)_\Omega = (\boldsymbol{f}, \boldsymbol{v})_\Omega, \tag{7.6}$$

$$\forall q \in L_0^2(\Omega), \quad (\nabla\cdot\boldsymbol{u}, q)_\Omega = 0. \tag{7.7}$$

One can show the existence of a weak solution. Uniqueness of the weak solution is guaranteed under a condition on the data $\boldsymbol{f}$ and μ.

In the rest of the chapter, we mainly focus on the discretization of the nonlinear term.

7.3 DG discretization

Let $\mathcal{E}_h$ be a regular subdivision of Ω. We seek an approximation of velocity and pressure in the finite-dimensional spaces $\boldsymbol{X}_h$ and M_h, respectively:

$$\boldsymbol{X}_h = \{\boldsymbol{v} \in L^2(\Omega)^2 : \forall E \in \mathcal{E}_h, \boldsymbol{v} \in (\mathbb{P}_k(E))^2\},$$
$$M_h = \{q \in L_0^2(\Omega) : \forall E \in \mathcal{E}_h, q \in \mathbb{P}_{k-1}(E)\}.$$

We recall the energy norm:

$$\forall\boldsymbol{v} \in H^1(\mathcal{E}_h)^2, \quad \|\boldsymbol{v}\|_{\mathcal{E}} = \left(\sum_{E\in\mathcal{E}_h}\|\nabla\boldsymbol{v}\|_{L^2(E)}^2 + \sum_{e\in\Gamma_h\cup\partial\Omega}\frac{\sigma_e^0}{|e|}\|[\boldsymbol{v}]\|_{L^2(e)}^2\right)^{1/2}. \tag{7.8}$$

The DG discretization of the diffusive term $-\Delta\boldsymbol{u}$ and the pressure term ∇p is done with the bilinear forms a_ϵ and b defined in Section 6.3. We assume that the penalty parameter σ_e^0 is strictly positive for all edges and that the form a_ϵ is coercive. Next, we present an upwind discretization of the nonlinear term $\boldsymbol{u}\cdot\nabla\boldsymbol{u}$.

7.3.1 Nonlinear convective term

Given a mesh element E and a function $\boldsymbol{v} \in \boldsymbol{X}_h$, we define the inflow $\partial E^{\boldsymbol{v}}_-$ and outflow $\partial E^{\boldsymbol{v}}_+$ boundaries of E with respect to $\boldsymbol{v}$ as

$$\partial E^{\boldsymbol{v}}_- = \{\boldsymbol{x} \in \partial E : \{\boldsymbol{v}\} \cdot \boldsymbol{n}_E < 0\}, \quad \partial E^{\boldsymbol{v}}_+ = \partial E \setminus \partial E^{\boldsymbol{v}}_-.$$

We denote by $\boldsymbol{v}^{\text{int}}$ the trace of the function $\boldsymbol{v}$ on a side of E coming from the interior of E and by $\boldsymbol{v}^{\text{ext}}$ the trace of $\boldsymbol{v}$ coming from the exterior of E. The quantity $\boldsymbol{v}^{\text{int}} - \boldsymbol{v}^{\text{ext}}$ is just another way of writing the jump of $\boldsymbol{v}$. By convention, if the edge lies on the boundary of the domain, we have $\boldsymbol{v}^{\text{int}} = \boldsymbol{v}$ and $\boldsymbol{v}^{\text{ext}} = \boldsymbol{0}$.

The DG discretization of the nonlinear term $\boldsymbol{u} \cdot \nabla \boldsymbol{u}$ is defined by the form c:

$$\forall \boldsymbol{w}, \boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\theta} \in \boldsymbol{X}, \quad c(\boldsymbol{w}, \boldsymbol{v}; \boldsymbol{z}, \boldsymbol{\theta}) = \sum_{E \in \mathcal{E}_h} \int_E (\boldsymbol{v} \cdot \nabla \boldsymbol{z}) \cdot \boldsymbol{\theta} + \frac{1}{2} \sum_{E \in \mathcal{E}_h} \int_E (\nabla \cdot \boldsymbol{v}) \boldsymbol{z} \cdot \boldsymbol{\theta} \tag{7.9}$$

$$+ \sum_{E \in \mathcal{E}_h} \int_{\partial E^{\boldsymbol{w}}_-} |\{\boldsymbol{v}\} \cdot \boldsymbol{n}_E| (\boldsymbol{z}^{\text{int}} - \boldsymbol{z}^{\text{ext}}) \cdot \boldsymbol{\theta}^{\text{int}} - \frac{1}{2} \sum_{e \in \Gamma_h \cup \partial\Omega} \int_e [\boldsymbol{v}] \cdot \boldsymbol{n}_e \{\boldsymbol{z} \cdot \boldsymbol{\theta}\}. \tag{7.10}$$

The first argument of c appears only in the domain of integration $\partial E^{\boldsymbol{w}}_-$. The form c is linear with respect to its three other arguments. Before stating important properties of the form c, we need a lemma on the term involving $\boldsymbol{w}$. The proof of the lemma is technical and can be found in [60].

Lemma 7.3. *Denote*

$$\ell(\boldsymbol{w}, \boldsymbol{v}; \boldsymbol{z}, \boldsymbol{\theta}) = \sum_{E \in \mathcal{E}_h} \int_{\partial E^{\boldsymbol{w}}_-} |\{\boldsymbol{v}\} \cdot \boldsymbol{n}_E| (\boldsymbol{z}^{\text{int}} - \boldsymbol{z}^{\text{ext}}) \cdot \boldsymbol{\theta}^{\text{int}}. \tag{7.11}$$

Then, there is a constant C_0 independent of h such that

$$\left| \ell(\boldsymbol{z}, \boldsymbol{z}; \boldsymbol{z}, \boldsymbol{w}) - \ell(\boldsymbol{v}, \boldsymbol{v}; \boldsymbol{v}, \boldsymbol{w}) \right| \leq C_0 \|\boldsymbol{z} - \boldsymbol{v}\|_{\mathcal{E}} \|\boldsymbol{w}\|_{\mathcal{E}} (\|\boldsymbol{z}\|_{\mathcal{E}} + \|\boldsymbol{v}\|_{\mathcal{E}}). \tag{7.12}$$

Lemma 7.4. *The form c satisfies the following positivity property:*

$$\forall \boldsymbol{v}, \boldsymbol{z} \in \boldsymbol{X}^h, \quad c(\boldsymbol{v}, \boldsymbol{v}; \boldsymbol{z}, \boldsymbol{z}) = \frac{1}{2} \sum_{E \in \mathcal{E}_h} \| |\{\boldsymbol{v}\} \cdot \boldsymbol{n}_E|^{1/2} (\boldsymbol{z}^{\text{int}} - \boldsymbol{z}^{\text{ext}}) \|^2_{L^2(\partial E^{\boldsymbol{v}}_- \setminus \partial\Omega)}$$

$$+ \| |\boldsymbol{v} \cdot \boldsymbol{n}|^{1/2} \boldsymbol{z} \|^2_{L^2(\partial\Omega^{\text{e}}_+)}. \tag{7.13}$$

Proof. We perform an integration by parts on a given mesh element E with unit normal vector $\boldsymbol{n}_E = (n_i)_i$:

$$\int_E \boldsymbol{v} \cdot \nabla \boldsymbol{z} \cdot \boldsymbol{\theta} = \sum_{1 \leq i,j \leq 2} \int_E v_i \theta_j \frac{\partial z_j}{\partial x_i}$$

$$= \sum_{1 \leq i,j \leq 2} \int_E v_i \frac{\partial (z_j \theta_j)}{\partial x_i} - \sum_{1 \leq i,j \leq 2} \int_E z_j v_i \frac{\partial \theta_j}{\partial x_i}$$

$$
\begin{aligned}
&= -\sum_{1\le i,j\le 2}\int_E \frac{\partial v_i}{x_i} z_j\theta_j + \sum_{1\le i,j\le 2}\int_{\partial E} v_i z_j \theta_j n_i - \sum_{1\le i,j\le 2}\int_E z_j v_i \frac{\partial \theta_j}{\partial x_i} \\
&= -\int_E (\nabla\cdot \boldsymbol{v})\boldsymbol{z}\cdot\boldsymbol{\theta} + \int_{\partial E}(\boldsymbol{v}\cdot\boldsymbol{n}_E)\boldsymbol{z}\cdot\boldsymbol{\theta} - \int_E \boldsymbol{v}\cdot\nabla\boldsymbol{\theta}\cdot\boldsymbol{z}.
\end{aligned}
$$

Therefore, we can write

$$
\begin{aligned}
c(\boldsymbol{v},\boldsymbol{v};\boldsymbol{z},\boldsymbol{\theta}) = &-\sum_{E\in\mathcal{E}_h}\int_E (\boldsymbol{v}\cdot\nabla\boldsymbol{\theta})\cdot\boldsymbol{z} - \frac{1}{2}\sum_{E\in\mathcal{E}_h}\int_E (\nabla\cdot\boldsymbol{v})\boldsymbol{z}\cdot\boldsymbol{\theta} - \frac{1}{2}\sum_{e\in\Gamma_h\cup\partial\Omega}\int_e [\boldsymbol{v}]\cdot\boldsymbol{n}_e\{\boldsymbol{z}\cdot\boldsymbol{\theta}\} \\
&+ \sum_{E\in\mathcal{E}_h}\int_{\partial E_-^{\boldsymbol{v}}} |\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E|(\boldsymbol{z}^{\mathrm{int}}-\boldsymbol{z}^{\mathrm{ext}})\cdot\boldsymbol{\theta}^{\mathrm{int}} \\
&+ \sum_{E\in\mathcal{E}_h}\int_{\partial E}\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}}(\boldsymbol{v}^{\mathrm{int}}\cdot\boldsymbol{n}_E). \qquad (7.14)
\end{aligned}
$$

We now rewrite the trace of $\boldsymbol{v}$ on the boundary ∂E as follows:

$$
\boldsymbol{v}^{\mathrm{int}} = \{\boldsymbol{v}\} + \frac{1}{2}(\boldsymbol{v}^{\mathrm{int}} - \boldsymbol{v}^{\mathrm{ext}}) \quad \text{on} \quad \partial E.
$$

Thus, we obtain

$$
\begin{aligned}
&\sum_{E\in\mathcal{E}_h}\int_{\partial E}\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}}(\boldsymbol{v}^{\mathrm{int}}\cdot\boldsymbol{n}_E) \\
&\qquad = \sum_{E\in\mathcal{E}_h}\int_{\partial E}\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}} + \frac{1}{2}\sum_{E\in\mathcal{E}_h}\int_{\partial E}(\boldsymbol{v}^{\mathrm{int}}-\boldsymbol{v}^{\mathrm{ext}})\cdot\boldsymbol{n}_E\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}} \\
&\qquad = \sum_{E\in\mathcal{E}_h}\int_{\partial E}\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}} + \sum_{e\in\Gamma_h\cup\partial\Omega}\int_e [\boldsymbol{v}]\cdot\boldsymbol{n}_e\{\boldsymbol{z}\cdot\boldsymbol{\theta}\}. \qquad (7.15)
\end{aligned}
$$

In the last equation we have used the definition of the jump and the normal vector $\boldsymbol{n}_e$. Combining (7.14) with (7.15), we have

$$
\begin{aligned}
c(\boldsymbol{v},\boldsymbol{v};\boldsymbol{z},\boldsymbol{\theta}) = &-\sum_{E\in\mathcal{E}_h}\int_E (\boldsymbol{v}\cdot\nabla\boldsymbol{\theta})\cdot\boldsymbol{z} - \frac{1}{2}\sum_{E\in\mathcal{E}_h}\int_E (\nabla\cdot\boldsymbol{v})\boldsymbol{z}\cdot\boldsymbol{\theta} + \frac{1}{2}\sum_{e\in\Gamma_h\cup\partial\Omega}\int_e [\boldsymbol{v}]\cdot\boldsymbol{n}_e\{\boldsymbol{z}\cdot\boldsymbol{\theta}\} \\
&+ \sum_{E\in\mathcal{E}_h}\int_{\partial E_-^{\boldsymbol{v}}} |\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E|(\boldsymbol{z}^{\mathrm{int}}-\boldsymbol{z}^{\mathrm{ext}})\cdot\boldsymbol{\theta}^{\mathrm{int}} \\
&+ \sum_{E\in\mathcal{E}_h}\int_{\partial E}\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}}. \qquad (7.16)
\end{aligned}
$$

In the last term in (7.16), we decompose $\partial E = \partial E_-^{\boldsymbol{v}} \cup \partial E_+^{\boldsymbol{v}}$ and note that

$$
\sum_{E\in\partial E_+^{\boldsymbol{v}}\cap\partial\Omega}\int_E \{\boldsymbol{v}\}\cdot\boldsymbol{n}_E\boldsymbol{z}^{\mathrm{int}}\cdot\boldsymbol{\theta}^{\mathrm{int}} = \sum_{E\in\partial E_-^{\boldsymbol{v}}\cap\partial\Omega}\int_E |\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E|\boldsymbol{z}^{\mathrm{ext}}\cdot\boldsymbol{\theta}^{\mathrm{ext}}.
$$

This implies that the last two terms of (7.16) are equal to the following quantity:

$$\sum_{E\in\mathcal{E}_h}\int_{\partial E^v_-}|\{\boldsymbol{v}\}\cdot\boldsymbol{n}_E|\boldsymbol{z}^{\text{ext}}\cdot(\boldsymbol{\theta}^{\text{ext}}-\boldsymbol{\theta}^{\text{int}})+\int_{\partial\Omega^v_+}\boldsymbol{v}\cdot\boldsymbol{n}\boldsymbol{z}\cdot\boldsymbol{\theta}. \tag{7.17}$$

Using (7.16) and (7.17), we obtain another expression for $c(\boldsymbol{v},\boldsymbol{v};\boldsymbol{z},\boldsymbol{\theta})$. We now choose $\boldsymbol{z}=\boldsymbol{\theta}$ in the resulting expression and in the definition (7.10) and sum the two quantities to obtain (7.13). □

Define the space

$$\boldsymbol{V}_h=\{\boldsymbol{v}\in\boldsymbol{X}_h:\forall q\in M_h,\, b(\boldsymbol{v},q)=0\}.$$

The next result says that the form c is continuous [61].

Lemma 7.5. *There is a constant C_1 independent of h and μ such that*

$$\forall\boldsymbol{v}\in\boldsymbol{V}^h,\forall\boldsymbol{w},\boldsymbol{z},\boldsymbol{\theta}\in\boldsymbol{X}^h,\quad c(\boldsymbol{w},\boldsymbol{v};\boldsymbol{z},\boldsymbol{\theta})\le C_1\|\boldsymbol{v}\|_{\mathcal{E}}\|\boldsymbol{z}\|_{\mathcal{E}}\|\boldsymbol{\theta}\|_{\mathcal{E}}. \tag{7.18}$$

It is easy to see that, when $\boldsymbol{u},\boldsymbol{z},\boldsymbol{\theta}\in H^1_0(\Omega)^2$, c reduces to

$$c^{\boldsymbol{u}}(\boldsymbol{u};\boldsymbol{z},\boldsymbol{\theta})=\int_\Omega(\boldsymbol{u}\cdot\nabla\boldsymbol{z})\cdot\boldsymbol{\theta}+\frac{1}{2}\int_\Omega(\nabla\cdot\boldsymbol{u})\boldsymbol{z}\cdot\boldsymbol{\theta}. \tag{7.19}$$

7.3.2 Scheme

The DG variational formulation is as follows: Find $(\boldsymbol{U}_h,P_h)\in\boldsymbol{X}_h\times M_h$ such that

$$\forall\boldsymbol{v}\in\boldsymbol{X}_h,\quad \mu a_\epsilon(\boldsymbol{U}_h,\boldsymbol{v})+c(\boldsymbol{U}_h,\boldsymbol{U}_h;\boldsymbol{U}_h,\boldsymbol{v})+b(\boldsymbol{v},P_h)=(\boldsymbol{f},\boldsymbol{v})_\Omega, \tag{7.20}$$

$$\forall q\in M_h,\quad b(\boldsymbol{U}_h,q)=0. \tag{7.21}$$

7.3.3 Consistency

If $(\boldsymbol{u},p)$ satisfies (7.3)–(7.5), then $c(\boldsymbol{u},\boldsymbol{u};\boldsymbol{u},v)=(\boldsymbol{u}\cdot\nabla\boldsymbol{u},\boldsymbol{v})_\Omega$ and one can check that $(\boldsymbol{u},p)$ satisfies (7.20)–(7.21).

7.4 Existence and uniqueness of solution

7.4.1 Existence of discrete velocity

Consider the space $\boldsymbol{V}_h$ equipped with the energy norm $\|\cdot\|_{\mathcal{E}}$ and the corresponding inner-product $(\cdot,\cdot)_{\mathcal{E}}$:

$$(\boldsymbol{v},\boldsymbol{z})_{\mathcal{E}}=\sum_{E\in\mathcal{E}_h}(\nabla\boldsymbol{v},\nabla\boldsymbol{z})_E+\sum_{e\in\Gamma_h\cup\partial\Omega}\frac{\sigma_e^0}{|e|}([\boldsymbol{v}],[\boldsymbol{z}])_e.$$

Define the mapping $\Psi : \boldsymbol{V}_h \to \boldsymbol{V}_h$ by

$$\forall \boldsymbol{v}, \boldsymbol{z} \in \boldsymbol{V}_h, \quad (\Psi(\boldsymbol{v}), \boldsymbol{z})_{\mathcal{E}} = \mu a_\epsilon(\boldsymbol{v}, \boldsymbol{z}) + c(\boldsymbol{v}, \boldsymbol{v}; \boldsymbol{v}, \boldsymbol{z}) - (\boldsymbol{f}, \boldsymbol{z})_\Omega.$$

Then, coercivity of a_ϵ and property (7.13) imply

$$\forall \boldsymbol{v} \in \boldsymbol{V}_h, \quad (\Psi(\boldsymbol{v}), \boldsymbol{v})_{\mathcal{E}} \geq \mu\kappa \|\boldsymbol{v}\|_{\mathcal{E}}^2 - (\boldsymbol{f}, \boldsymbol{v})_\Omega.$$

The term $(\boldsymbol{f}, \boldsymbol{v})_\Omega$ is bounded by Cauchy–Schwarz's inequality and Poincaré's inequality (3.5):

$$(\boldsymbol{f}, \boldsymbol{v})_\Omega \leq \|\boldsymbol{f}\|_{L^2(\Omega)} \|\boldsymbol{v}\|_{L^2(\Omega)} \leq \tilde{C} \|\boldsymbol{f}\|_{L^2(\Omega)} \|\boldsymbol{v}\|_{\mathcal{E}}.$$

Thus, we obtain

$$(\Psi(\boldsymbol{v}), \boldsymbol{v})_{\mathcal{E}} \geq (\mu\kappa \|\boldsymbol{v}\|_{\mathcal{E}} - \tilde{C} \|\boldsymbol{f}\|_{L^2(\Omega)}) \|\boldsymbol{v}\|_{\mathcal{E}}.$$

Define the sphere S in $\boldsymbol{V}_h$:

$$S = \left\{ \boldsymbol{v} \in \boldsymbol{V}_h : \|\boldsymbol{v}\|_{\mathcal{E}} = \frac{\tilde{C}}{\mu\kappa} \|\boldsymbol{f}\|_{L^2(\Omega)} \right\}.$$

Then, we have

$$\forall \boldsymbol{v} \in S, \quad (\Psi(\boldsymbol{v}), \boldsymbol{v})_{\mathcal{E}} \geq 0.$$

From Lemma 7.2, there exists a function $\boldsymbol{U}_h \in \boldsymbol{V}_h$ such that

$$\Psi(\boldsymbol{U}_h) = 0, \quad \|\boldsymbol{U}_h\|_{\mathcal{E}} \leq \frac{\tilde{C}}{\mu\kappa} \|\boldsymbol{f}\|_{L^2(\Omega)}.$$

In particular $(\Psi(\boldsymbol{U}_h), \boldsymbol{v})_{\mathcal{E}} = 0$ for all $\boldsymbol{v} \in \boldsymbol{V}_h$, and $\boldsymbol{U}_h$ satisfies (7.20) restricted to the space $\boldsymbol{V}_h$. From the definition of $\boldsymbol{V}_h$, the function $\boldsymbol{U}_h$ automatically satisfies (7.21).

7.4.2 Existence of discrete pressure

The existence of P_h is a consequence of the inf-sup condition (6.27). Using similar notation as in Section 6.1.6, we define the polar set of V_h:

$$V_h^\circ = \{\phi \in \boldsymbol{X}_h' : \forall \boldsymbol{v} \in \boldsymbol{V}_h, \phi(\boldsymbol{v}) = 0\}.$$

We define the mapping $B' : M_h \to V_h^\circ$ by

$$\forall q \in M_h, \forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad B'q(\boldsymbol{v}) = b(\boldsymbol{v}, q),$$

and, using the solution $\boldsymbol{U}_h$ found in the preceding section, we define the mapping $\phi \in \boldsymbol{X}_h'$ by

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad \phi(\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v})_\Omega - \mu a_\epsilon(\boldsymbol{U}_h, \boldsymbol{v}) - c(\boldsymbol{U}_h, \boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{v}).$$

This linear functional ϕ belongs to V_h° since $\phi(\boldsymbol{v}) = 0$ for all $\boldsymbol{v} \in \boldsymbol{V}_h$. From Lemma 6.4, there is a unique function $P_h \in M_h$ such that

$$B' P_h = \phi;$$

equivalently,

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad b(\boldsymbol{v}, P_h) = (\boldsymbol{f}, \boldsymbol{v})_\Omega - \mu a_\epsilon(\boldsymbol{U}_h, \boldsymbol{v}) - c(\boldsymbol{U}_h, \boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{v}).$$

Thus, the pair $(\boldsymbol{U}_h, P_h)$ is a solution to (7.20)–(7.21).

7.4.3 A priori bounds

Lemma 7.6. *Let $(\boldsymbol{U}_h, P_h) \in \boldsymbol{X}_h \times M_h$ be a solution to the discrete Navier–Stokes problem. There are constants C_2, C_3 independent of h and μ such that*

$$\mu \|\boldsymbol{U}_h\|_{\mathcal{E}} \le C_2 \|\boldsymbol{f}\|_{L^2(\Omega)}, \tag{7.22}$$

$$\|P_h\|_{L^2(\Omega)} \le C_3(\|\boldsymbol{f}\|_{L^2(\Omega)} + \|\boldsymbol{U}_h\|_{\mathcal{E}}^2). \tag{7.23}$$

Proof. The bound $\|\boldsymbol{U}_h\|_{\mathcal{E}} \le \frac{C}{\mu}\|\boldsymbol{f}\|_{L^2(\Omega)}$ is obtained by choosing $\boldsymbol{v} = \boldsymbol{U}_h$ in (7.20) and by using coercivity of a_ϵ and property (7.13). From the remark in Section 6.4, there is a function $\boldsymbol{v}_h \in \boldsymbol{X}_h$ such that

$$b(\boldsymbol{v}_h, P_h) = -\|P_h\|_{L^2(\Omega)}^2, \quad \|\boldsymbol{v}_h\|_{\mathcal{E}} \le \frac{1}{\beta^*}\|P_h\|_{L^2(\Omega)}. \tag{7.24}$$

Then, from (7.20), we have

$$\|P_h\|_{L^2(\Omega)}^2 = \mu a_\epsilon(\boldsymbol{U}_h, \boldsymbol{v}_h) + c(\boldsymbol{U}_h, \boldsymbol{U}_h; \boldsymbol{U}_h, \boldsymbol{v}_h) - (\boldsymbol{f}, \boldsymbol{v}_h)_\Omega.$$

Using continuity of a_ϵ and property (7.18), we obtain

$$\|P_h\|_{L^2(\Omega)}^2 = C(\mu\|\boldsymbol{U}_h\|_{\mathcal{E}}\|\boldsymbol{v}_h\|_{\mathcal{E}} + \|\boldsymbol{U}_h\|_{\mathcal{E}}^2\|\boldsymbol{v}_h\|_{\mathcal{E}} + \|\boldsymbol{f}\|_{L^2(\Omega)}\|\boldsymbol{v}_h\|_{L^2(\Omega)}).$$

We then conclude from (3.5) and the bounds (7.24) and (7.22). □

7.4.4 Uniqueness

Theorem 7.7. *Under the condition*

$$\mu^2 > \frac{2C_2(C_0 + C_1)}{\kappa}\|\boldsymbol{f}\|_{L^2(\Omega)}, \tag{7.25}$$

there is a unique solution to (7.20)–(7.21).

Proof. Assume that $(\boldsymbol{U}_h^1, P_h^1)$ and $(\boldsymbol{U}_h^2, P_h^2)$ are two solutions to (7.20)–(7.21) and denote their differences by $\boldsymbol{W}_h = \boldsymbol{U}_h^1 - \boldsymbol{U}_h^2$ and $\xi_h = P_h^1 - P_h^2$. Then, we have

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad \mu a_\epsilon(\boldsymbol{W}_h, \boldsymbol{v}) + c(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{v}) - c(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{v}) + b(\boldsymbol{v}, \xi_h) = 0,$$
$$\forall q \in M_h, \quad b(\boldsymbol{W}_h, q) = 0.$$

Choosing $\boldsymbol{v} = \boldsymbol{W}_h$ and $q = \xi_h$ in the equations above and using coercivity of a_ϵ gives

$$\mu\kappa\|\boldsymbol{W}_h\|_{\mathcal{E}}^2 + c(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{W}_h) - c(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{W}_h) \le 0. \tag{7.26}$$

We recall the form ℓ introduced in (7.11), and we denote $d(\boldsymbol{v}, \boldsymbol{z}, \boldsymbol{\theta}) = c(\boldsymbol{w}, \boldsymbol{v}; \boldsymbol{z}, \boldsymbol{\theta}) - \ell(\boldsymbol{w}, \boldsymbol{v}; \boldsymbol{z}, \boldsymbol{\theta})$. We rewrite the nonlinear terms as

$$\begin{aligned} c(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{W}_h) - c(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{W}_h) = d(\boldsymbol{U}_h^1; \boldsymbol{W}_h, \boldsymbol{W}_h) \\ + d(\boldsymbol{W}_h; \boldsymbol{U}_h^2, \boldsymbol{W}_h) + \ell(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{W}_h) - \ell(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{W}_h). \end{aligned}$$

From Lemma 7.3, we have

$$\left|\ell(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{W}_h) - \ell(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{W}_h)\right| \leq C_0 \|\boldsymbol{W}_h\|_{\mathcal{E}}^2 (\|\boldsymbol{U}_h^1\|_{\mathcal{E}} + \|\boldsymbol{U}_h^2\|_{\mathcal{E}}).$$

Using the a priori bound (7.22), we obtain

$$\left|\ell(\boldsymbol{U}_h^1, \boldsymbol{U}_h^1; \boldsymbol{U}_h^1, \boldsymbol{W}_h) - \ell(\boldsymbol{U}_h^2, \boldsymbol{U}_h^2; \boldsymbol{U}_h^2, \boldsymbol{W}_h)\right| \leq \frac{2C_0C_2}{\mu} \|\boldsymbol{f}\|_{L^2(\Omega)} \|\boldsymbol{W}_h\|_{\mathcal{E}}^2.$$

Similarly, from (7.18) and (7.22), we have

$$\begin{aligned} \left|d(\boldsymbol{U}_h^1; \boldsymbol{W}_h, \boldsymbol{W}_h) + d(\boldsymbol{W}_h; \boldsymbol{U}_h^2, \boldsymbol{W}_h)\right| &\leq C_1 \|\boldsymbol{W}_h\|_{\mathcal{E}}^2 (\|\boldsymbol{U}_h^1\|_{\mathcal{E}} + \|\boldsymbol{U}_h^2\|_{\mathcal{E}}) \\ &\leq \frac{2C_1C_2}{\mu} \|\boldsymbol{f}\|_{L^2(\Omega)} \|\boldsymbol{W}_h\|_{\mathcal{E}}^2. \end{aligned}$$

Therefore, (7.26) becomes

$$\left(\mu\kappa - \frac{2C_2(C_0 + C_1)}{\mu} \|\boldsymbol{f}\|_{L^2(\Omega)}\right) \|\boldsymbol{W}_h\|_{\mathcal{E}}^2 \leq 0.$$

From this, we can easily conclude that $\boldsymbol{W}_h = \boldsymbol{0}$ if

$$\mu\kappa > \frac{2C_2(C_0 + C_1)}{\mu} \|\boldsymbol{f}\|_{L^2(\Omega)}.$$

Uniqueness of pressure is a consequence of the inf-sup condition (6.27) as proved in Section 7.4.2. □

7.5 A priori error estimates

We state the convergence result for the method. The proof is left to the reader.

Theorem 7.8. *Assume that condition* (7.25) *holds true. If the exact solution* $(\boldsymbol{u}, p)$ *belongs to* $H^{k+1}(\Omega)^2 \times H^k(\Omega)$*, there is a constant* C *independent of* h *and* μ *such that the numerical error satisfies*

$$\begin{aligned} \|\boldsymbol{u} - \boldsymbol{U}_h\|_{\mathcal{E}} &\leq Ch^k \left(\left(1 + \frac{1}{\mu^2}\right) |\boldsymbol{u}|_{H^{k+1}(\Omega)} + \frac{1}{\mu} |p|_{H^k(\Omega)}\right), \\ \|p - P_h\|_{L^2(\Omega)} &\leq Ch^k \left(\left(\mu + \frac{1}{\mu} + \frac{1}{\mu^3}\right) |\boldsymbol{u}|_{H^{k+1}(\Omega)} + \left(1 + \frac{1}{\mu^2}\right) |p|_{H^k(\Omega)}\right). \end{aligned}$$

7.6 Numerical experiments

The nonlinear scheme (7.20)–(7.21) has been linearized by Picard's iteration method. Given an initial velocity ($\boldsymbol{U}_h^0 = \boldsymbol{0}$), one computes a sequence ($\boldsymbol{U}_h^n$, P_h^n) satisfying for $n \geq 0$

$$\forall \boldsymbol{v} \in \boldsymbol{X}_h, \quad \mu a_\epsilon(\boldsymbol{U}_h^{n+1}, \boldsymbol{v}) + c(\boldsymbol{U}_h^n, \boldsymbol{U}_h^n; \boldsymbol{U}_h^{n+1}, \boldsymbol{v}) + b(\boldsymbol{v}, P_h^{n+1}) = (\boldsymbol{f}, \boldsymbol{v})_\Omega,$$
$$\forall q \in M_h, \quad b(\boldsymbol{U}_h^{n+1}, q) = 0.$$

The termination criterion for the Picard iterations is

$$\|\boldsymbol{U}_h^{n+1} - \boldsymbol{U}_h^n\|_{L^2(\Omega)} \leq 10^{-10}.$$

7.6.1 Effects of penalty size

We can repeat the numerical convergence studies done in Section 6.6 for the Navier–Stokes problem. We obtain the same convergence rates, not shown here. We now study the effect of the penalty parameter σ_e^0 on the accuracy of the solution. We fix a mesh, and we choose to approximate the velocity by piecewise polynomials of degree two and the pressure by piecewise polynomials of degree one. We vary the penalty parameter between $\sigma_e^0 = 10^{-3}$ and $\sigma_e^0 = 10^5$. Fig. 7.1 shows the variation of the energy norm of the error in velocity for NIPG, IIPG, and SIPG. First, for $\sigma_e^0 \geq 100$, all three methods yield the same numerical error. This is explained by the fact that the jump term dominates the flux terms computed on the edges. As the penalty increases in size, the error also increases. Second, if $\sigma_e^0 \leq 1$, the NIPG method yields comparable accurate solutions even as σ_e^0 decreases. This is not the case for the other two methods. The error obtained with the SIPG method slightly increases as the penalty value tends to zero. The IIPG solution also loses accuracy, and the method does not converge after 40 Picard iterations if the penalty value is 10^{-3}. Similar conclusions are made on the effects of the penalty size on the L^2 norm of the error in the velocity (see

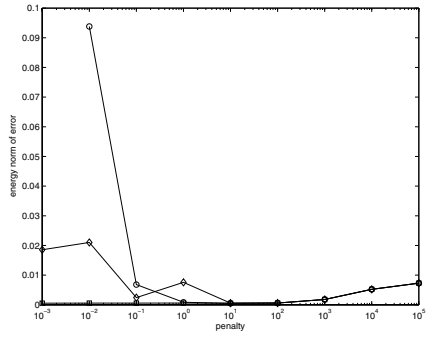


Figure 7.1. *Variations of the energy norm of numerical error in velocity with respect to the penalty value for NIPG (line with squares), SIPG (line with diamonds), and IIPG (line with circles).*

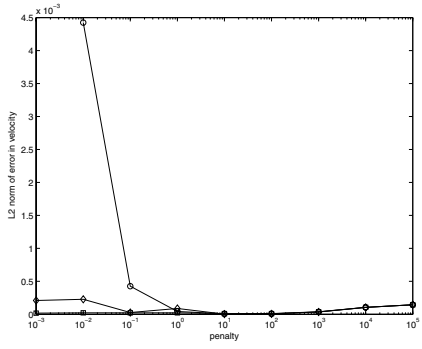


Figure 7.2. *Variations of the L^2 norm of numerical error in velocity with respect to the penalty value for NIPG (line with squares), SIPG (line with diamonds), and IIPG (line with circles).*

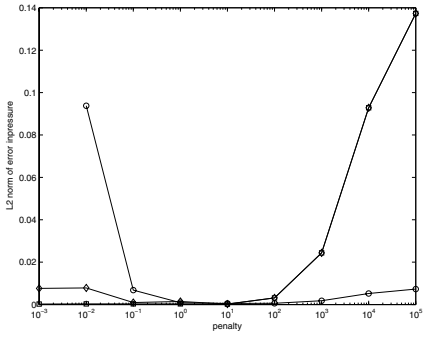


Figure 7.3. *Variations of the L^2 norm of numerical error in pressure with respect to the penalty value for NIPG (line with squares), SIPG (line with diamonds), and IIPG (line with circles).*

Fig. 7.2). The error for the pressure in L^2 norm behaves the same way for small penalties. However, for large penalties, Fig. 7.3 shows that the error in the pressure remains small for the IIPG method and increases for both the NIPG and SIPG methods.

7.6.2 Step channel problem

We consider the flow in a channel described in Fig. 7.4. The fluid enters the channel through the left vertical boundary denoted by Γ_- and exits through the right vertical boundary Γ_+. At both inflow Γ_- and outflow Γ_+ boundaries, we impose the following parabolic velocity field:

$$\forall (x_1, x_2) \in \Gamma_- \cup \Gamma_+, \quad \boldsymbol{u}(x_1, x_2) = (x_2(1 - x_2), 0).$$

On the rest of $\partial\Omega$, the velocity is set to $\mathbf{0}$. The fluid viscosity is equal to one. The Navier–Stokes velocity and pressure are approximated by piecewise polynomials of degree two and one, respectively. The velocity field and the pressure isocontours are given in Fig. 7.5 for NIPG 1 and in Fig. 7.6 for SIPG 10. The total number of degrees of freedom equal to 30000.

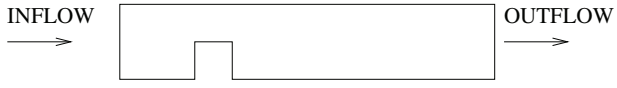


Figure 7.4. *Step channel problem setting.*

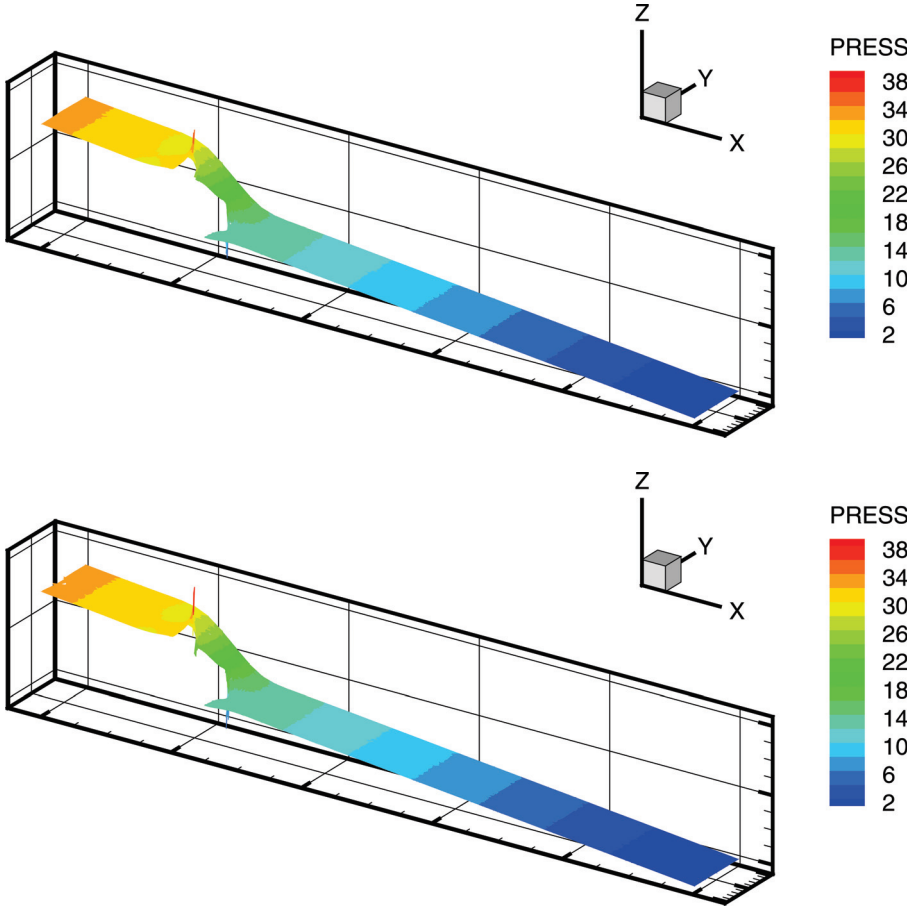


Figure 7.5. *Pressure isocontours for NIPG* 1 *(top) and SIPG* 10 *(bottom).*

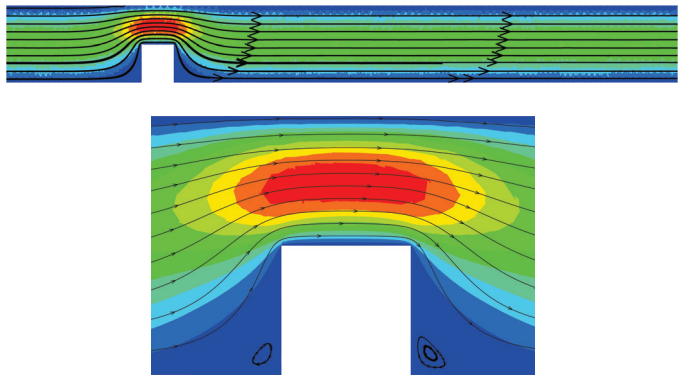
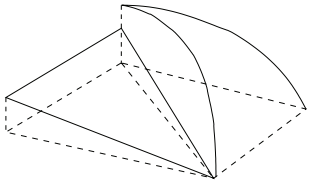


Figure 7.6. *Streamlines and velocity field for NIPG* 1.

7.7 Bibliographical remarks

The analysis of primal DG methods for Navier–Stokes equations can be found in [61, 89, 60]. LDG methods for the Oseen equations (linearized Navier–Stokes) are studied in [32]. Divergence-free DG solutions are studied in [33]. In [73], the approximation of velocity is discontinuous and pointwise divergence-free, and the approximation of pressure is continuous. For the time-dependent Navier–Stokes equations, primal DG methods were proposed and analyzed in [62, 74].



Chapter 8
Flow in porous media

In this chapter, examples of complex flow and transport phenomena in porous media are numerically solved by using primal DG methods. We consider the cases of miscible displacement and immiscible two-phase flow, which arise, for instance, in the environmental problem of subsurface contamination and in the production of oil from petroleum reservoirs.

A petroleum reservoir consists of a porous medium whose pores contain some hydrocarbon components, usually referred to as oil. Oil can be extracted by three recovery processes. In the primary recovery, oil is produced through wells by simple natural decompression. Very little is obtained, as this process ends quickly when the pressure equilibrium between the oil field and atmosphere is attained. In the second recovery, the wells are divided into two sets: injection wells and production wells. An inexpensive fluid (water) is injected into the reservoir to push the oil toward the production wells. Pressure inside the reservoir is maintained high enough to avoid collapse of the rock and to avoid gas production. In that case, the flow in the reservoir is a two-phase immiscible flow with no mass transfer between the phases. This process is described in Section 8.1. Only 40% of the oil is recovered with the secondary process. After both primary and secondary recovery processes, the capillary forces, and the interfacial tension between the injected and resident fluids causes some of the oil to remain in the pores of the reservoir rock. In order to increase oil recovery, the capillary forces have to be eliminated or reduced. This can be done if the injected fluid is miscible with the resident fluid. A solvent (polymers) is therefore injected at certain wells in the petroleum reservoir. The solvent mixes with the oil to form a single phase, and this mixture flows to other wells where oil is produced. The last process is called tertiary recovery or enhanced recovery and is described by the miscible displacement problem given in Section 8.2.

8.1 Two-phase flow

We consider pressure-saturation formulations of the incompressible two-phase flow. The reader can refer to [67, 27, 26] for a thorough treatment of the different models of two-phase flow.

8.1.1 Model problem

We assume that there are two flowing phases in the porous medium Ω: a wetting phase (such as water) and a nonwetting phase (such as oil). The saturation of a given phase is the ratio of the void volume filled with the phase to the total of the void volume in the porous medium. The unknown variables are the phase pressures p_w, p_n and the phase saturations s_w, s_n. The subscript w (resp., n) is used for the wetting phase (resp., nonwetting phase).

The Darcy velocity for each phase is given by

$$\boldsymbol{u}_\alpha = -\lambda_\alpha \boldsymbol{K}(\nabla p_\alpha - \rho_\alpha \boldsymbol{g}), \quad \alpha = w, n, \tag{8.1}$$

where λ_α is the phase mobility, $\boldsymbol{K}$ is the permeability of the porous medium, ρ_α is the phase density, and $\boldsymbol{g}$ is the constant gravitational vector. Phase mobilities are defined by

$$\lambda_\alpha = \frac{k_{r_\alpha}}{\mu_\alpha}, \quad \alpha = w, n, \tag{8.2}$$

where μ_α is the constant phase viscosity and k_{r_α} is the relative permeability of phase α. Relative permeabilities are nonlinear functions of the phase saturation, usually determined experimentally. Popular models for the relative permeability are the van Genuchten model [57] and the Brooks–Corey model [20]. For instance, using the Brooks–Corey formula, we have

$$k_{rw}(s_e) = s_e^{\frac{2+3\theta}{\theta}}, \quad k_{rn}(s_e) = (1 - s_e)^2 \left(1 - s_e^{\frac{2+\theta}{\theta}}\right), \tag{8.3}$$

where s_e is the effective saturation defined as

$$\forall s_{rw} \le s_w \le 1 - s_{rn}, \quad s_e = \frac{s_w - s_{rw}}{1 - s_{rw} - s_{rn}}.$$

The residual saturations s_{rw}, s_{rn} correspond to macroscopic averaging of local surface tension in the pores. For instance, if the nonwetting phase saturation s_n is less than s_{rn}, then the nonwetting phase fluid cannot be displaced by the wetting phase fluid. The additional parameter $\theta \in [0.2, 3.0]$ in the definition of the relative permeabilities is a result of the inhomogeneity of the medium. A highly heterogeneous porous medium is characterized by a large θ. The total velocity and total mobility are the sum of the phase velocities and phase mobilities, respectively:

$$\boldsymbol{u}_t = \boldsymbol{u}_w + \boldsymbol{u}_n, \quad \lambda_t = \lambda_w + \lambda_n.$$

We define the fractional flow function f_w:

$$f_w = \frac{\lambda_w}{\lambda_t}.$$

The permeability $\boldsymbol{K}$ corresponds to an average on the continuum scale of the microscopic heterogeneities of the medium. It may be discontinuous and may greatly vary in space. We assume that $\boldsymbol{K}$ is symmetric positive definite. The balance of mass for each phase yields the following equation:

$$\phi \frac{\partial s_\alpha}{\partial t} + \nabla \cdot \boldsymbol{u}_\alpha = q_\alpha, \quad \alpha = w, n. \tag{8.4}$$

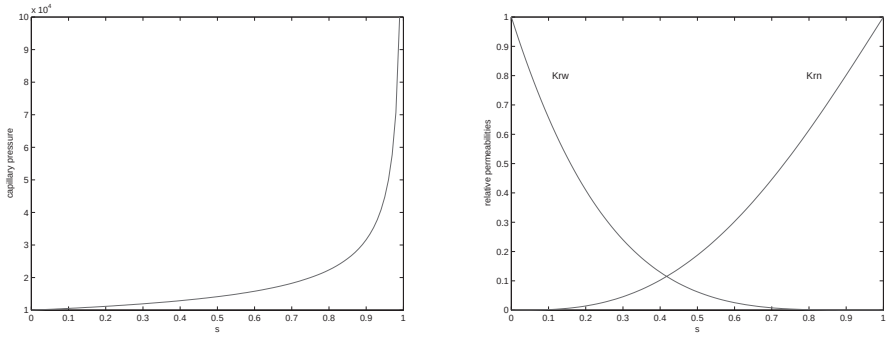


Figure 8.1. *Capillary pressure (left) and relative permeabilities (right) curves.*

The variable ϕ is the porosity: it is the ratio of the void volume to the total volume. The function q_α is a source/sink term, in general used to model wells located inside the domain. The system of equations (8.1), (8.4) is closed under the following relationships:

$$s_w + s_n = 1, \tag{8.5}$$

$$p_c = p_n - p_w. \tag{8.6}$$

Clearly, the definition of saturation yields (8.5). Equation (8.6) introduces the capillary pressure p_c, which can be thought of as the macroscopic average of the microscopic forces due to surface tension. The capillary pressure is a nonlinear function of the saturation and can be determined experimentally. With the Brooks–Corey model, the capillary pressure is

$$p_c(s_w) = p_d s_w^{-\frac{1}{\theta}}.$$

The constant pressure p_d corresponds to the capillary pressure needed to displace the fluid from the largest pore. Fig. 8.1 shows the capillary pressure and the relative permeabilities curves for $\theta = 2$ and for $s_{rw} = s_{rn} = 0$. Finally, an initial condition and boundary conditions complete the system. Possible boundary conditions are Dirichlet and Neumann for the pressure and Dirichlet, Neumann, or mixed for the saturation. For instance, we may have some of the following conditions on parts of the boundary:

$$p_w = p_{\mathrm{wD}} \quad \text{on} \quad \Gamma_{\mathrm{pwD}}, \tag{8.7}$$

$$s_n = s_{\mathrm{nD}} \quad \text{on} \quad \Gamma_{\mathrm{snD}}, \tag{8.8}$$

$$\boldsymbol{u}_w \cdot \boldsymbol{n} = 0 \quad \text{on} \quad \Gamma_{\mathrm{pwN}}, \tag{8.9}$$

$$\boldsymbol{u}_n \cdot \boldsymbol{n} = 0 \quad \text{on} \quad \Gamma_{\mathrm{pnN}}, \tag{8.10}$$

$$(s_w \boldsymbol{u}_t + \lambda_n f_w \boldsymbol{K} \nabla p_c) \cdot \boldsymbol{n} = s_{\mathrm{in}} \boldsymbol{u}_t \cdot \boldsymbol{n} \quad \text{on} \quad \Gamma_{\mathrm{swM}}, \tag{8.11}$$

$$\lambda_n f_w \boldsymbol{K} \nabla p_c \cdot \boldsymbol{n} = 0 \quad \text{on} \quad \Gamma_{\mathrm{swN}}. \tag{8.12}$$

From the constitutive relations (8.5) and (8.6), we can reduce the problem to a system of two equations with two unknowns. There are many possibilities, and we present a few possible choices.

Wetting phase pressure-wetting phase saturation formulation

Summing the two equations from (8.4) and using (8.5), we obtain

$$-\nabla \cdot (\lambda_t \boldsymbol{K} \nabla p_w) - \nabla \cdot (\lambda_n \boldsymbol{K} \nabla p_c) = q_w + q_n, \tag{8.13}$$

$$\phi \frac{\partial s_w}{\partial t} + \nabla \cdot (\lambda_n f_w \boldsymbol{K} \nabla p_c) + \nabla \cdot (f_w \boldsymbol{u}_t) = q_w. \tag{8.14}$$

Wetting phase pressure-nonwetting phase saturation formulation

Another equivalent formulation is

$$-\nabla \cdot (\lambda_t \boldsymbol{K} \nabla p_w) - \nabla \cdot (\lambda_n \boldsymbol{K} \nabla p_c) = q_w + q_n, \tag{8.15}$$

$$-\phi \frac{\partial s_n}{\partial t} - \nabla \cdot (\lambda_w \boldsymbol{K} \nabla p_w) = q_w. \tag{8.16}$$

Global pressure-nonwetting phase saturation formulation

Define the global pressure by

$$p = p_n - \int_{1-s_{nr}}^{1-s_n} f_w p_c' + p_c(1 - s_{nr}). \tag{8.17}$$

Then, an equivalent formulation is

$$-\nabla \cdot (\lambda_t \boldsymbol{K} \nabla p) = q_w + q_n, \tag{8.18}$$

$$\phi \frac{\partial s_n}{\partial t} + \nabla \cdot (\lambda_w \boldsymbol{K} \nabla p - \lambda_n f_w \boldsymbol{K} \nabla p_c) = -q_w. \tag{8.19}$$

The reader can refer to [26] for an extensive analysis of these mathematical models.

We discretize the partial differential equations in space using the DG method. Thus, the pressure is approximated by discontinuous polynomials of degree k_p and the saturation by discontinuous polynomials of degree k_s. The following two sections describe two different fully discrete schemes.

8.1.2 A sequential approach

We present a discretization of the model (8.13)–(8.14). The technique we describe in this section can be applied to any of the models given above. In a sequential approach, the idea is to decouple the equations by time-lagging the coefficients. Let N_{T} be a positive integer and let $\Delta t = T/N_{\mathrm{T}}$ denote the time step. Let $t^i = i\Delta t$. At each time t^i, we compute DG approximations of $p_w(t^i)$ and $s_w(t^i)$. We assume that $\partial\Omega = \Gamma_{\mathrm{pwD}} \cup \Gamma_{\mathrm{pwN}} = \Gamma_{\mathrm{swM}} \cup \Gamma_{\mathrm{swN}}$ and that boundary conditions (8.7), (8.9), (8.11), and (8.12) hold.

The fully discrete scheme is as follows: Given $(P_w^i, S_w^i) \in \mathcal{D}_{k_p}(\mathcal{E}_h) \times \mathcal{D}_{k_s}(\mathcal{E}_h)$, find $(P_w^{i+1}, S_w^{i+1}) \in \mathcal{D}_{k_p}(\mathcal{E}_h) \times \mathcal{D}_{k_s}(\mathcal{E}_h)$ such that for all $(v, z) \in \mathcal{D}_{k_p}(\mathcal{E}_h) \times \mathcal{D}_{k_s}(\mathcal{E}_h)$ we have the following set of equations.

Pressure equation

$$\sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{K}\lambda_t(S_w^i)\nabla P_w^{i+1}\cdot\nabla v - \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{pwD}}}\int_e\{\boldsymbol{K}\lambda_t(S_w^i)\nabla P_w^{i+1}\cdot\boldsymbol{n}_e\}[v]$$
$$+\epsilon\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{pwD}}}\int_e\{\boldsymbol{K}\lambda_t(S_w^i)\nabla v\cdot\boldsymbol{n}_e\}[P_w^{i+1}] + \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{pwD}}}\frac{\sigma_e^0}{|e|^\beta}\int_e[P_w^{i+1}][v]$$
$$= \sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{\chi}^i\cdot\nabla v - \sum_{e\in\Gamma_h\cup\partial\Omega}\int_e(\boldsymbol{\chi}^i)^{\mathrm{up}}\cdot\boldsymbol{n}_e[v]$$
$$+\sum_{e\in\Gamma_{\mathrm{pwD}}}\frac{\sigma_e^0}{|e|^\beta}\int_e p_{\mathrm{wD}}v + \epsilon\sum_{e\in\Gamma_{\mathrm{pwD}}}\int_e(\boldsymbol{K}\lambda_t(S_w^i)\nabla v\cdot\boldsymbol{n})p_{\mathrm{wD}}. \tag{8.20}$$

Saturation equation

$$\int_\Omega\frac{\phi}{\Delta t}S_w^{i+1}z + \sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{K}\lambda_n(S_w^i)f_w(S_w^i)|p_c'(S_w^i)|\nabla S_w^{i+1}\cdot\nabla z$$
$$-\sum_{e\in\Gamma_{s1}}\int_e S_w^{i+1}\boldsymbol{U}_t^i\cdot\boldsymbol{n}_e z - \sum_{e\in\Gamma_h}\int_e\{\boldsymbol{K}\lambda_n(S_w^i)f_w(S_w^i)|p_c'(S_w^i)|\nabla S_w^{i+1}\cdot\boldsymbol{n}_e\}[z]$$
$$+\epsilon\sum_{e\in\Gamma_h}\int_e\{\boldsymbol{K}\lambda_n(S_w^i)f_w(S_w^i)|p_c'(S_w^i)|\nabla z\cdot\boldsymbol{n}_e\}[S_w^{i+1}] + \sum_{e\in\Gamma_h}\frac{\sigma_e^0}{|e|^\beta}\int_e[S_w^i][z]$$
$$= \int_\Omega\frac{\phi}{\Delta t}S_w^i z + \sum_{E\in\mathcal{E}_h}\int_E\boldsymbol{\zeta}^i\cdot\nabla z - \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{pwD}}}\int_e(\boldsymbol{\zeta}^i)^{\mathrm{up}}\cdot\boldsymbol{n}_e[z] - \sum_{e\in\Gamma_{s1}}\int_e s_{\mathrm{in}}\boldsymbol{U}_t^i\cdot\boldsymbol{n}_e$$
$$-\sum_{e\in\Gamma_h}\int_e f_w(S_w^i)\{\boldsymbol{K}\lambda_t(S_w^i)\nabla z\cdot\boldsymbol{n}_e\}[P_w^i] - \sum_{e\in\Gamma_{\mathrm{pwD}}}\int_e\boldsymbol{K}\lambda_w(S_w^i)\nabla z\cdot\boldsymbol{n}_e(P_w^i - p_{\mathrm{wD}}), \tag{8.21}$$

where $\boldsymbol{U}_t^i$, $\boldsymbol{\zeta}^i$, and $\boldsymbol{\chi}^i$ are defined as

$$\boldsymbol{U}_t^i = -\boldsymbol{K}\lambda_w(S_w^i)\nabla P_w^i - \boldsymbol{K}\lambda_n(S_w^i)(p_c'(S_w^i)\nabla S_w^i + \nabla P_w^i),$$
$$\boldsymbol{\chi}^i = \boldsymbol{K}\lambda_n(S_w^i)|p_c'(S_w^i)|\nabla S_w^i,$$
$$\boldsymbol{\zeta}^i = f_w\{\boldsymbol{U}_t^i\}.$$

Because of the discontinuous approximations, there are two values for the functions $\boldsymbol{\chi}^i$ and $\boldsymbol{\zeta}^i$ on an interior edge. These quantities are then replaced by the upwind numerical fluxes $(\boldsymbol{\chi}^i)^{\mathrm{up}}$ and $(\boldsymbol{\zeta}^i)^{\mathrm{up}}$. Upwinding is done with respect to the normal component of the average of the total velocity $\boldsymbol{U}_t$ (see Section 4.2):

$$w^{\mathrm{up}} = \begin{cases} w|_{E_e^1} & \text{if} \quad \{\boldsymbol{U}_t\}\cdot\boldsymbol{n}_e \geq 0 \\ w|_{E_e^2} & \text{if} \quad \{\boldsymbol{U}_t\}\cdot\boldsymbol{n}_e < 0 \end{cases} \qquad \forall e = \partial E_e^1\cap\partial E_e^2. \tag{8.22}$$

The approximation S_w^0 is defined as the L^2 projection of the initial saturation $s_w(t=0)$.

Depending on the choice of ϵ and σ_e^0, we obtain the NIPG, IIPG, and SIPG variations of the DG methods. One may use different penalty values for the pressure and saturation equations. By time-lagging the coefficients, we obtain linear equations in P_w^{i+1} and S_w^{i+1}. Equation (8.20) corresponds to the DG method applied to an elliptic problem, and (8.21) is the DG method applied to a parabolic problem with convection. In realistic two-phase flow, convection dominates the transport, and thus one has to apply a slope limiter to the approximation of the saturation at each time step. Without slope limiting, the sequential approach yields large overshoot and undershoot that blow up at finite time. Besides, since slope limiting restricts the approximation to piecewise linears, it seems pointless to use high order of approximation.

8.1.3 A coupled approach

Let us consider the global pressure-phase saturation problem (8.18)–(8.19). We assume that the global pressure p satisfies the Dirichlet boundary condition of the form (8.7) on Γ_{D} and Neumann boundary conditions (8.9) and (8.10) on Γ_{N}. The nonwetting phase saturation is prescribed on the Dirichlet boundary Γ_{D}. For each $1 \le i \le N_{\mathrm{T}}$, the approximations $(P^i, S_n^i) \in \mathcal{D}_{k_p}(\mathcal{E}_h) \times \mathcal{D}_{k_s}(\mathcal{E}_h)$ of the functions $(p(\cdot, t^i), s_n(\cdot, t^i))$ satisfy the following set of equations.

Pressure equation

$$\begin{aligned}
\forall v \in \mathcal{D}_{k_p}(\mathcal{E}_h),\ & \sum_{E\in\mathcal{E}_h}\int_E \lambda_t(S_n^{i+1})\boldsymbol{K}\nabla P^{i+1}\cdot\nabla v + \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|^\beta}\int_e [P^{i+1}][v] \\
& - \sum_{e\in\Gamma_h}\int_e \{\lambda_t(S_n^{i+1})\boldsymbol{K}\nabla P^{i+1}\cdot\boldsymbol{n}_e\}[v] - \sum_{e\in\Gamma_{\mathrm{D}}}\int_e (\lambda_t(s_{\mathrm{nD}})\boldsymbol{K}\nabla P^{i+1}\cdot\boldsymbol{n}_e)v \\
& + \varepsilon\sum_{e\in\Gamma_h}\int_e \{\lambda_t(S_n^{i+1})\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\}[P^{i+1}] + \varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e (\lambda_t(s_{\mathrm{nD}})\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e)P^{i+1} \\
= \varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e & (\lambda_t(s_{\mathrm{nD}})\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e)p_{\mathrm{D}} + \sum_{e\in\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|^\beta}\int_e p_{\mathrm{D}}v + \int_\Omega (q_w(t^{i+1}) + q_n(t^{i+1}))v. \quad (8.23)
\end{aligned}$$

Saturation equation

$$\begin{aligned}
\forall z \in \mathcal{D}_{k_s}(\mathcal{E}_h),\ & \int_\Omega \frac{\phi}{\Delta t}(S_n^{i+1} - S_n^i)z - \sum_{E\in\mathcal{E}_h}\int_E \lambda_w(S_n^{i+1})\boldsymbol{K}\nabla P^{i+1}\cdot\nabla z \\
& + \sum_{E\in\mathcal{E}_h}\int_E \lambda_n(S_n^{i+1})f_w(S_n^{i+1})p_c'(S_n^{i+1})\boldsymbol{K}\nabla S_n^{i+1}\cdot\nabla z \\
& + \sum_{e\in\Gamma_h}\int_e \{\lambda_w(S_n^{i+1})\boldsymbol{K}\nabla P^{i+1}\cdot\boldsymbol{n}_e\}[z] + \sum_{e\in\Gamma_{\mathrm{D}}}\int_e (\lambda_w(s_{\mathrm{nD}})\boldsymbol{K}\nabla P^{i+1}\cdot\boldsymbol{n}_e)z \\
& - \sum_{e\in\Gamma_h}\int_e \{\lambda_n(S_n^{i+1})f_w(S_n^{i+1})p_c'(S_n^{i+1})\boldsymbol{K}\nabla S_n^{i+1}\cdot\boldsymbol{n}_e\}[z] \\
& - \sum_{e\in\Gamma_{\mathrm{D}}}\int_e (\lambda_n(s_{\mathrm{nD}})f_w(s_{\mathrm{nD}})p_c'(s_{\mathrm{nD}})\boldsymbol{K}\nabla S_n^{i+1}\cdot\boldsymbol{n}_e)z
\end{aligned}$$

$$
\begin{aligned}
-\varepsilon \sum_{e\in\Gamma_h}\int_e \{\lambda_w(S_n^{i+1})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e\}[P^{i+1}] - \varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e (\lambda_w(s_{\mathrm{nD}})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e)P^{i+1}\\
+\varepsilon\sum_{e\in\Gamma_h}\int_e \{\lambda_n(S_n^{i+1})f_w(S_n^{i+1})p_c'(S_n^{i+1})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e\}[S_n^{i+1}]\\
+\varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e \lambda_n(s_{\mathrm{nD}})f_w(s_{\mathrm{nD}})p_c'(s_{\mathrm{nD}})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e)S_n^{i+1}\\
+\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|^\beta}\int_e [S_n^{i+1}][z] = \sum_{e\in\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|^\beta}\int_e s_{\mathrm{nD}}z - \int_\Omega q_w(t^{i+1})z\\
-\varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e(\lambda_w(s_{\mathrm{nD}})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e)p_{\mathrm{D}} + \varepsilon\sum_{e\in\Gamma_{\mathrm{D}}}\int_e(\lambda_n(s_{\mathrm{nD}})f_w(s_{\mathrm{nD}})p_c'(s_{\mathrm{nD}})\boldsymbol{K}\nabla z\cdot\boldsymbol{n}_e)s_{\mathrm{nD}}.
\end{aligned}
\tag{8.24}
$$

The approximation S_n^0 is chosen as the L^2 projection of the saturation $s_n(t=0)$. Problem (8.23)–(8.24) is a system of nonlinear equations to be solved at each time step. We either use a Picard iteration or Newton's method. The coupled approach is therefore more costly than the sequential approach. The main advantage of this approach lies in the fact that even though some overshoot and undershoot phenomena occur, they are small and remain bounded. No slope limiting technique is needed. In the nondegenerate case, we can analyze the scheme and show that it converges with optimal rate [49]. Besides, increasing the degrees of polynomials improves the accuracy of the solution and diminishes the amount of overshoot/undershoot.

8.1.4 Numerical examples

The Buckley–Leverett problem: This is the two-phase flow problem in 1D and without capillary pressure effects. In that case, a quasi-analytical solution can be derived. Fig. 8.2

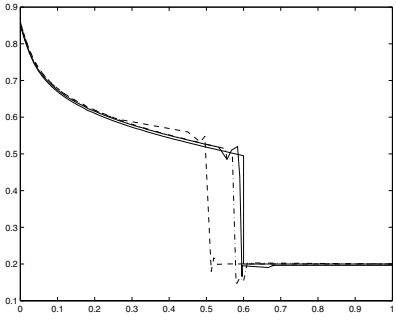


Figure 8.2. *Buckley–Leverett problem: exact solution (solid line), DG solution with $k_s = 1$ (dashed line), DG solution with $k_s = 2$ (dash-dotted line), and DG solution with $k_s = 3$ (solid line).*

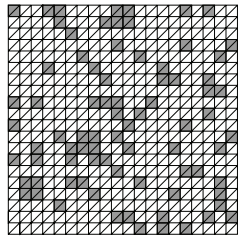


Figure 8.3. *Permeability field and coarse mesh: permeability is* 10^{-11} *in white regions and* 10^{-16} *elsewhere.*

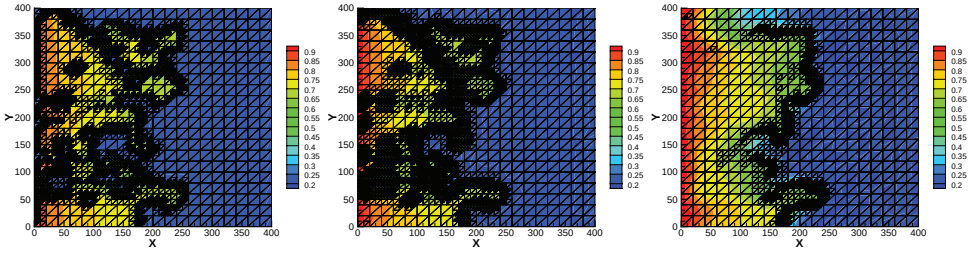


Figure 8.4. *NIPG: wetting phase saturation contours at* 35 *days:* $\sigma_e^0 = 0$ *(left),* $\sigma_e^0 = 10^{-5}$ *(center),* $\sigma_e^0 = 1$ *(right).*

compares the DG solution obtained with the coupled approach to the quasi-analytical solution. As the polynomial degree increases, the solution is more accurate. Small overshoot and undershoot occur, but they remain stable.

Inhomogeneous medium: We consider a square domain $(0, 400)^2$ with varying permeability as shown in Fig. 8.3. The permeability is $10^{-11}\mathbf{I}\,\mathrm{m}^2$ except in several small regions where it is 10^5 times smaller (see [45]). The regions with small permeability act as impermeable zones into which the injected wetting phase does not flow. The simulation is run for 35 days. The vertical boundaries correspond to Γ_{pwD}: a difference in pressure drives the flow from left to right. The left vertical boundary corresponds to Γ_{swM}. The sequential approach (8.20)–(8.21) is used with slope limiting described in Section 4.3.2. The DG approximations are piecewise linears for saturation and piecewise quadratics for pressure. At each time step, the mesh is refined or derefined according to computed error indicators [75]. Fig. 8.4 shows the wetting phase saturation contours for the NIPG method with various penalty values. There is very little numerical diffusion for small values of the penalty, namely $\sigma_e^0 = 0$ or $\sigma_e^0 = 10^{-5}$. However, if the penalty increases ($\sigma_e^0 = 1$), then the numerical solution is diffusive and does not "see" the regions of small permeability values. This sensitivity to the choice of the penalty is pronounced for highly varying permeability. Fig. 8.5 shows the contours of the saturation obtained with the SIPG and IIPG methods. Similar conclusions are made.

Five-spot problem: The domain and the coarse mesh are given in Fig. 8.6. Four production wells are located at each corner of the domain, and one injection well is located in the center of the domain. Wetting phase pressure is imposed on the well bores. The meshes

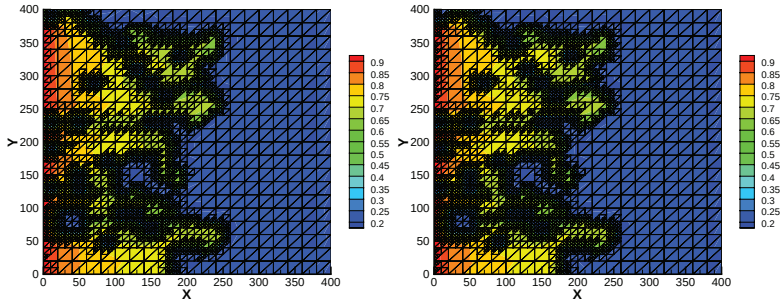


Figure 8.5. *Two-dimensional wetting phase saturation contours at* 35 *days for* $\sigma_e^0 = 10^{-6}$*: IIPG (left) and SIPG (right).*

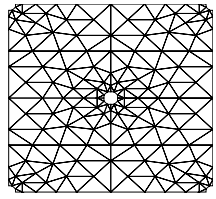


Figure 8.6. *Five-spot problem: domain with coarse mesh.*

are adaptively refined and derefined as the front of the wetting phase moves through the domain. Here, the DG method with the sequential approach is used. Fig. 8.7 shows the contours of the wetting phase pressure and saturation at a given time.

8.2 Miscible displacement

We consider the displacement of one incompressible fluid by another in a porous medium Ω in $\mathbb{R}^2$ over a time period $(0, T)$. The invading and the displaced fluids are referred to as the solvent and the resident fluid. We further assume that solvent and resident fluid mix in all proportions forming a single phase, and we neglect the influence of gravity. Miscible displacement is mathematically modeled by a coupled system of an elliptic equation with a parabolic equation that is convection dominated. In that sense, the nature of the problem is similar to the global pressure-phase saturation model of two-phase flow. The classical equations governing the miscible displacement in Ω over $[0, T]$ are

$$\boldsymbol{u} = \frac{1}{\mu(c)} \boldsymbol{K} \nabla p \quad \text{in} \quad [0, T] \times \Omega, \tag{8.25}$$

$$-\nabla \cdot \boldsymbol{u} = q, \qquad \text{in} \quad [0, T] \times \Omega, \tag{8.26}$$

$$\phi \frac{\partial c}{\partial t} + \nabla \cdot (\boldsymbol{u} c - \boldsymbol{D}(\boldsymbol{u}) \nabla c) = \tilde{c} q \qquad \text{in} \quad [0, T] \times \Omega, \tag{8.27}$$

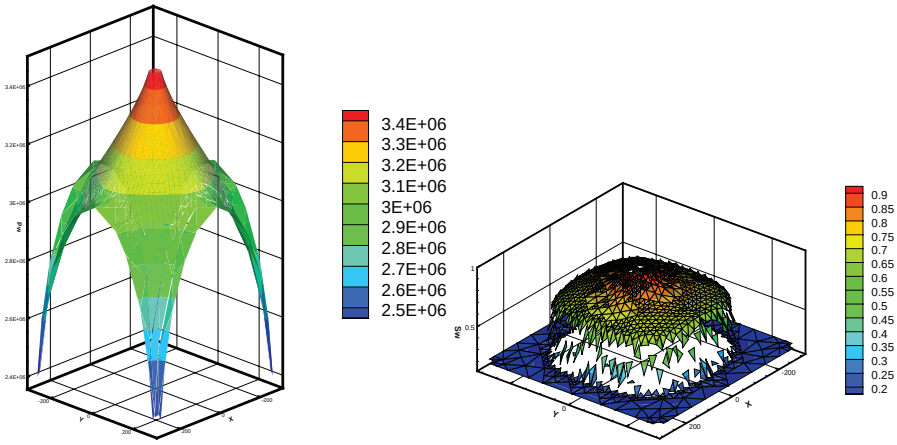


Figure 8.7. *Five-spot problem: wetting phase pressure (left) and wetting phase saturation (right).*

where the unknown variables are p, the pressure in the fluid mixture, and c, the fraction volume (or concentration) of the solvent in the fluid mixture. The permeability $\boldsymbol{K}$ of the medium measures the resistance of the medium to fluid flow; the viscosity μ of the fluid measures the resistance to flow of the fluid mixture; $\boldsymbol{u}$ represents the Darcy velocity (volume flowing across a unit cross-section per unit time); the porosity ϕ is the fraction of the volume of the medium occupied by pores; and $\boldsymbol{D}(\boldsymbol{u})$ is the coefficient of molecular diffusion and mechanical dispersion of one fluid into the other. The imposed external total flow rate q is a sum of point sources and sinks, and, in the case of oil recovery, q represents the flow rates at injection and production wells. The data $\tilde{c}$ is the injected concentration at injection wells and the resident concentration at production wells. Here, $\boldsymbol{D}(\boldsymbol{u})$ is a tensor depending on the velocity:

$$\boldsymbol{D}(\boldsymbol{u}) = (\alpha_t|\boldsymbol{u}| + D_m)\mathbf{I} + (\alpha_l - \alpha_t)\frac{\boldsymbol{u}\boldsymbol{u}^T}{\|\boldsymbol{u}\|},$$

where D_m is the molecular diffusivity, α_l and α_t are the longitudinal and transverse dispersivities, respectively, $\|\boldsymbol{u}\|$ represents the magnitude of the velocity vector, and $\mathbf{I}$ is the identity tensor. In 2D, for $\boldsymbol{u} = (u_x, u_y)$, $\boldsymbol{D}(\boldsymbol{u})$ can be written as

$$\boldsymbol{D}(\boldsymbol{u}) = D_m\mathbf{I} + \frac{\alpha_l}{\|\boldsymbol{u}\|}\begin{bmatrix} u_x^2 & u_xu_y \\ u_xu_y & u_y^2 \end{bmatrix} + \frac{\alpha_t}{\|\boldsymbol{u}\|}\begin{bmatrix} u_y^2 & -u_xu_y \\ -u_xu_y & u_x^2 \end{bmatrix}. \tag{8.28}$$

Equation (8.25) is a formulation of Darcy's law and is referred to as the pressure equation. Equation (8.27) is the transport equation. The boundary of the domain is characterized in two ways. First, it is decomposed into a Dirichlet part Γ_{D} and a Neumann part Γ_{N} such that $\Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}} = \partial\Omega$ and $\Gamma_{\mathrm{D}} \cap \Gamma_{\mathrm{N}} = \emptyset$. Second, the boundary is divided into an inflow part Γ_{in} and an outflow part Γ_{out}:

$$\Gamma_{\mathrm{in}} = \{\boldsymbol{x} \in \partial\Omega : \boldsymbol{u} \cdot \boldsymbol{n} < 0\}, \quad \Gamma_{\mathrm{out}} = \partial\Omega \setminus \Gamma_{\mathrm{in}}.$$

We assume the following boundary conditions for pressure and concentration:

$$p = p_{\mathrm{D}} \quad \text{in} \quad (0,T)\times\Gamma_{\mathrm{D}}, \tag{8.29}$$

$$\frac{1}{\mu(c)}\boldsymbol{K}\nabla p\cdot\boldsymbol{n} = 0 \quad \text{in} \quad (0,T)\times\Gamma_{\mathrm{N}}, \tag{8.30}$$

$$(c\boldsymbol{u} - \boldsymbol{D}(\boldsymbol{u})\nabla c)\cdot\boldsymbol{n} = c_{\mathrm{in}}\boldsymbol{u}\cdot\boldsymbol{n} \quad \text{in} \quad (0,T)\times\Gamma_{\mathrm{in}}, \tag{8.31}$$

$$-\boldsymbol{D}(\boldsymbol{u})\nabla c\cdot\boldsymbol{n} = 0 \quad \text{in} \quad (0,T)\times\Gamma_{\mathrm{out}}. \tag{8.32}$$

Finally, the system is completed by an initial condition on c:

$$c = c_0 \quad \text{in} \quad \{0\}\times\Omega.$$

The viscosity of the fluid mixture is assumed to follow the quarter-power mixing law, commonly applicable to hydrocarbon mixtures [76]:

$$\mu(c) = (c\mu_s^{-0.25} + (1-c)\mu_o^{-0.25})^{-4},$$

where μ_s (resp., μ_o) is the viscosity of the solvent (resp., resident fluid). The stability of the flow is characterized by the mobility ratio, i.e., the ratio of the viscosity of the resident fluid to the viscosity of the solvent:

$$M = \frac{\mu_o}{\mu_s}.$$

Small instabilities in the flow will grow if the mobility ratio is larger than unity. In that case, protrusions referred to as viscous fingering develop through the resident fluid. Other important characteristics are Peclet numbers P_{em}, P_{el}, P_{et} that give the ratios of convective effects to those of molecular diffusion, longitudinal dispersion, and transversal dispersion, respectively. If L denotes a representative length, those numbers are defined by

$$P_{em} = \frac{qL}{\phi D_m}, \quad P_{el} = \frac{L}{\alpha_l}, \quad P_{et} = \frac{L}{\alpha_t}.$$

8.2.1 Semidiscrete formulation

We define in this section a semidiscrete problem for (8.26)–(8.32). Let $\mathcal{E}_h$ be a mesh as described in Section 2.3. We first define forms that are linear with respect to their last two arguments:

$$\begin{aligned} a_\epsilon(c;p,v) = &\sum_{E\in\mathcal{E}_h}\int_E \frac{1}{\mu(c)}\boldsymbol{K}\nabla p\nabla v - \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\int_e\left\{\frac{K}{\mu(c)}\nabla p\cdot\boldsymbol{n}_e\right\}[v] \\ &+ \sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|}\int_e [p][v] + \epsilon\sum_{e\in\Gamma_h\cup\Gamma_{\mathrm{D}}}\int_e\left\{\frac{K}{\mu(c)}\nabla v\cdot\boldsymbol{n}_e\right\}[p], \end{aligned}$$

$$b_\epsilon(\boldsymbol{u}; c, w) = \sum_{E\in\mathcal{E}_h}\int_E \boldsymbol{D}(\boldsymbol{u})\nabla c\nabla w - \int_E c\boldsymbol{u}\cdot\nabla w$$
$$- \sum_{e\in\Gamma_h}\int_e \{\boldsymbol{D}(\boldsymbol{u})\nabla c\cdot\boldsymbol{n}_e\}[w] + \epsilon\sum_{e\in\Gamma_h}\int_e \{\boldsymbol{D}(\boldsymbol{u})\nabla w\cdot\boldsymbol{n}_e\}[c]$$
$$+ \sum_{e\in\Gamma_h}\int_e c^{\mathrm{up}}\{\boldsymbol{u}\}\cdot\boldsymbol{n}_e[w] + \sum_{e\in\Gamma_{\mathrm{out}}}\int_e c\boldsymbol{u}\cdot\boldsymbol{n}_e w + \sum_{e\in\Gamma_h}\frac{\sigma_e^0}{|e|}\int_e [c][w],$$

where c^{up} is the upwind value of c with respect to $\{\boldsymbol{u}\}$ (similar definition as (8.22)). We also define the following functionals that are linear with respect to their last argument:

$$L_1(c; v) = \int_\Omega qv + \sum_{e\in\Gamma_{\mathrm{D}}}\int_e \left\{\frac{1}{\mu(c)}\boldsymbol{K}\nabla v\cdot\boldsymbol{n}_e\right\} p_{\mathrm{D}} + \sum_{e\in\Gamma_{\mathrm{N}}}\int_e gv + \sum_{e\in\Gamma_{\mathrm{D}}}\frac{\sigma_e^0}{|e|}\int_e p_{\mathrm{D}}v,$$
$$L_1(\boldsymbol{u}; w) = \int_\Omega \tilde{c}qw - \sum_{e\in\Gamma_{\mathrm{in}}}\int_e c_{\mathrm{in}}\boldsymbol{u}\cdot\boldsymbol{n}_e w.$$

The continuous in time DG method is given by the map $(P_h, C_h) : [0, T] \to \mathcal{D}_{k_p}(\mathcal{E}_h) \times \mathcal{D}_{k_c}(\mathcal{E}_h)$ determined by the relations, for any t in J,

$$\forall v \in \mathcal{D}_{k_p}(\mathcal{E}_h), \quad a_\epsilon(C_h(t); P_h(t), v) = L_1(C_h(t), v), \tag{8.33}$$
$$\forall w \in \mathcal{D}_{k_c}(\mathcal{E}_h), \quad \left(\phi\frac{\partial C_h}{\partial t}, w\right)_\Omega + b_\epsilon(\mathbf{U}_h(t); C_h, w) = L_2(\boldsymbol{U}_h, w), \tag{8.34}$$
$$\forall w \in \mathcal{D}_{k_c}(\mathcal{E}_h), \quad (C_h(0), w)_\Omega = (c_0, w)_\Omega, \tag{8.35}$$

where

$$\mathbf{U}_h(t) = -\frac{1}{\mu(C_h(t))}\boldsymbol{K}\nabla P_h(t).$$

There are several possible time discretizations. For instance, as in the two-phase flow problem, we can solve the pressure and concentration equations together as a coupled nonlinear system, or we can solve the equations successively by time-lagging the coefficients. The latter approach is presented in the following section.

8.2.2 A fully discrete approach

Let N_{T} be a positive integer and let $\Delta t = T/N_{\mathrm{T}}$ denote the time step. Let $t^i = i\Delta t$. At each time t^i, we compute a DG approximation of $p(t^i)$ and $c(t^i)$ denoted by P_h^i and C_h^i and satisfying for all $i \geq 0$

$$\forall v \in \mathcal{D}_{k_p}(\mathcal{E}_h), \quad a_\epsilon(C_h^i(t); P_h^{i+1}(t), v) = L_1(C_h^i(t), v),$$
$$\forall w \in \mathcal{D}_{k_c}(\mathcal{E}_h), \quad \left(\phi\frac{\tilde{C}_h^{i+1} - C_h^i}{\Delta t}, w\right)_\Omega + b_\epsilon(\mathbf{U}_h^i; \tilde{C}_h^{i+1}, w) = L_2(\boldsymbol{U}_h^i, w),$$
$$C_h^{i+1} = \mathcal{L}(\tilde{C}_h^{i+1}).$$

The initial concentration C_h^0 is the L^2 projection of c_0 onto $\mathcal{D}_{k_c}(\mathcal{E}_h)$. Because of the large convective effects, slope limiters are needed in order to obtain a stable scheme. The slope limiting operator is denoted by $\mathcal{L}$ (see Section 4.3.2).

8.2.3 Numerical examples

We present some simulation results obtained with the sequential approach. Both concentration and pressure are approximated by discontinuous piecewise quadratics. We assume first that the permeability field takes two values. The permeability is 1000 times smaller in a horizontal band located at the center of the domain. Fig. 8.8 shows the concentration contours for a mobility ratio equal to 100. As expected, the solvent does not penetrate the band of lower permeability. Next, we consider a randomly generated permeability field, and we set the mobility ratio equal to 10. Fig. 8.9 shows the contours of pressure and concentration at a given time. Because of the highly discontinuous permeability (representing the heterogeneities of the medium), viscous fingering occurs. As the mobility ratio increases, the length of the fingers increases. Fig. 8.10 shows the contours of pressure and concentration computed at the same time as in Fig. 8.9 but for a mobility ratio equal to 100.

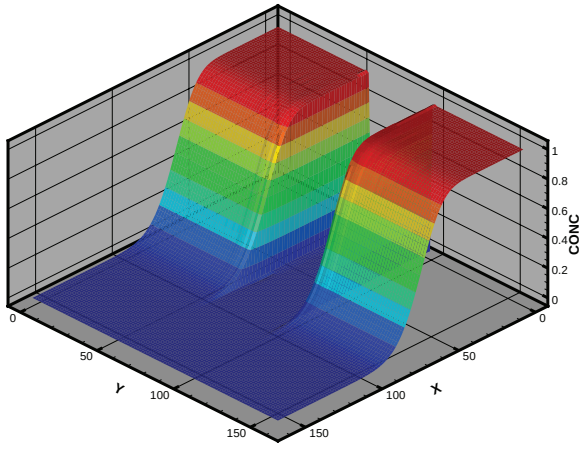


Figure 8.8. *Mobility ratio* 100: *concentration contours.*

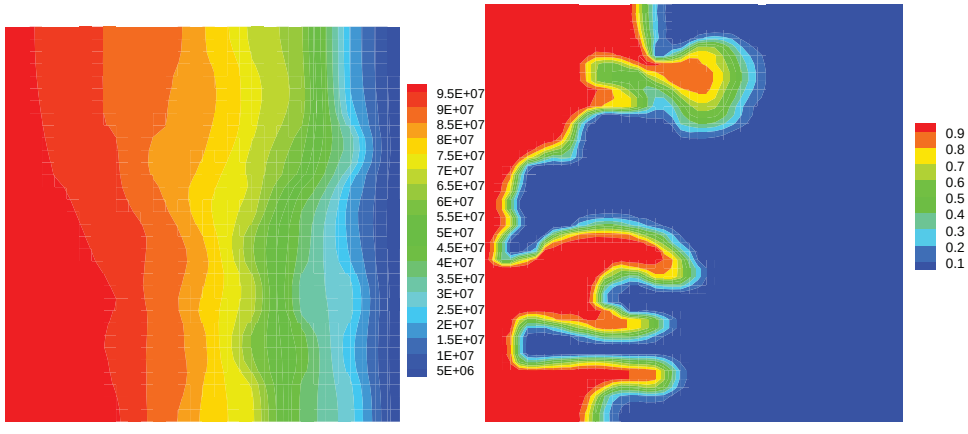


Figure 8.9. *Mobility ratio* 10: *pressure contours (left) and concentration contours (right).*

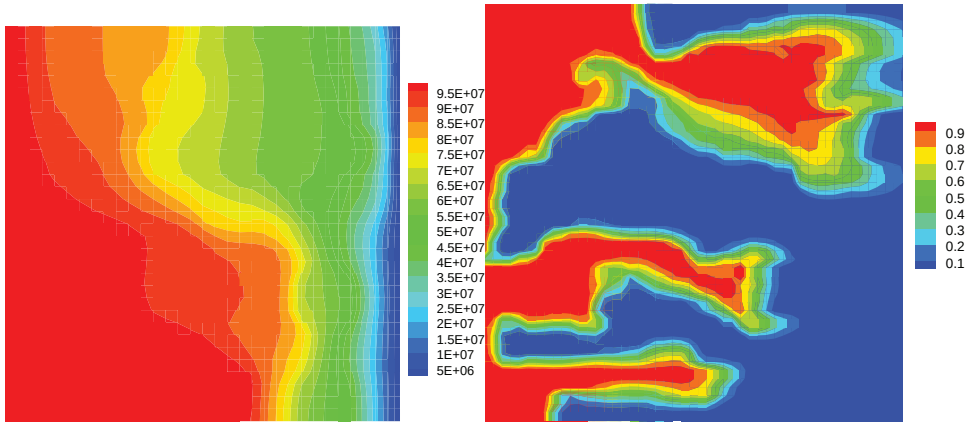
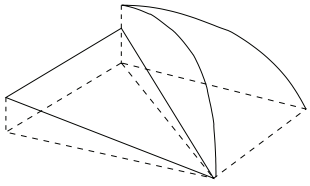


Figure 8.10. *Mobility ratio* 100: *pressure contours (left) and concentration contours (right).*

8.3 Bibliographical remarks

Sequential DG methods for incompressible two-phase flow are formulated in [75]. In [69], the DG method is combined with the mixed finite element method. Coupled DG methods for incompressible two-phase flow are studied in [51, 48, 49]. DG methods for incompressible miscible displacement are considered in [103, 102, 93, 92, 52].



Appendix A
Quadrature rules

Let $\mathcal{O}$ be a domain in $\mathbb{R}^d$. A quadrature rule is defined by a set of nodes $(\boldsymbol{x}_i)_{i\in I}$ in $\mathcal{O}$ and a set of weights $(w_i)_{i\in I}$ in $\mathbb{R}$. The general form of a quadrature rule is

$$\int_{\mathcal{O}} f \approx \sum_{i\in I} w_i f(\boldsymbol{x}_i).$$

In this chapter, we define quadrature rules for an interval, for the reference triangle, and for the reference quadrilateral.

A.1 Gauss quadrature rule on intervals

With a change of variable, we can transform any given integral on the interval (a, b) into an integral on the interval $(-1, 1)$:

$$\int_a^b f(s)ds = \frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}t + \frac{a+b}{2}\right)dt.$$

Therefore, we consider only the Gauss quadrature rule on the interval $(-1, 1)$:

$$\int_{-1}^{1} f(s)ds \approx \sum_{i=1}^{Q_{\mathrm{G}}} w_i f(s_i).$$

The Gauss quadrature rule with Q_{G} nodes is exact for polynomials of degree less than or equal to $2Q_{\mathrm{G}}-1$. Table A.1 lists the quadrature nodes and weights for several values of Q_{G}.

A.2 Quadrature rules on the reference triangle

We present numerical quadrature rules [44] for computing $\int_{\hat{E}} \phi(x, y)dxdy$, where $\hat{E}$ is the reference triangle defined in Section 2.5.1. The quadrature rule is

$$\int_{\hat{E}} \hat{v} \approx \sum_{i=1}^{Q_{\mathrm{D}}} w_i \hat{v}(s_{x,i}, s_{y,i}).$$

Table A.1. *Gauss quadrature nodes and weights on the interval* $(-1, 1)$.

Q_G	k	w_j	s_j
1	1	2.000000000000	0.000000000000
2	3	1.000000000000	−0.577350269189
		1.000000000000	0.577350269189
3	5	0.555555555555	−0.774596669241
		0.888888888888	0.000000000000
		0.555555555555	0.774596669241
4	7	0.347854845137	−0.861136311594
		0.652145154862	−0.33998104358
		0.652145154862	0.339981043584
		0.347854845137	0.861136311594

Table A.2. *Quadrature weights and points for reference triangle.*

Q_D	k	w_i	$s_{x,i}$	$s_{y,i}$
1	1	0.500000000000	0.333333333333	0.333333333333
3	2	0.166666666666	0.666666666667	0.166666666667
		0.166666666666	0.166666666667	0.166666666667
		0.166666666666	0.166666666667	0.666666666667
4	3	−0.281250000000	0.333333333333	0.333333333333
		0.260416666666	0.200000000000	0.200000000000
		0.260416666666	0.600000000000	0.200000000000
		0.260416666666	0.200000000000	0.600000000000
6	4	0.1116907948390	0.108103018168	0.445948490915
		0.1116907948390	0.445948490915	0.445948490915
		0.1116907948390	0.445948490915	0.108103018168
		0.0549758718276	0.816847572980	0.091576213509
		0.0549758718276	0.091576213509	0.091576213509
		0.0549758718276	0.091576213509	0.816847572980
7	5	0.112500000000	0.333333333333	0.333333333333
		0.062969590272	0.101286507323	0.101286507323
		0.062969590272	0.797426985353	0.101286507323
		0.062969590272	0.101286507323	0.797426985353
		0.066197076394	0.470142064105	0.470142064105
		0.066197076394	0.059715871789	0.470142064105
		0.066197076394	0.470142064105	0.059715871789
12	6	0.058393137863	0.501426509658	0.249286745170
		0.058393137863	0.249286745170	0.249286745170
		0.058393137863	0.249286745170	0.501426509658
		0.025422453185	0.873821971016	0.063089014491
		0.025422453185	0.063089014491	0.063089014491
		0.025422453185	0.063089014491	0.873821971016
		0.041425537809	0.053145049844	0.310352451033

Table A.2. *Continued*

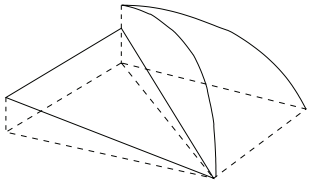
Q_{D}	k	w_i	$s_{x,i}$	$s_{y,i}$
		0.041425537809	0.310352451033	0.053145049844
		0.041425537809	0.053145049844	0.636502499123
		0.041425537809	0.636502499123	0.053145049844
		0.041425537809	0.636502499123	0.310352451033
		0.041425537809	0.310352451033	0.636502499123
13	7	−0.074785022233	0.333333333333	0.333333333333
		0.087807628716	0.479308067841	0.260345966079
		0.087807628716	0.260345966079	0.260345966079
		0.087807628716	0.260345966079	0.479308067841
		0.026673617804	0.869739794195	0.065130102902
		0.026673617804	0.065130102902	0.065130102902
		0.026673617804	0.065130102902	0.869739794195
		0.038556880445	0.048690315425	0.312865496004
		0.038556880445	0.312865496004	0.048690315425
		0.038556880445	0.048690315425	0.638444188569
		0.038556880445	0.638444188569	0.048690315425
		0.038556880445	0.638444188569	0.312865496004
		0.038556880445	0.312865496004	0.638444188569

Let k denote the polynomial degree for which this rule is exact. Table A.2 gives the values of the quadrature nodes $(s_{x,i}, s_{y,i})$ and weights w_i for several values of k and Q_{D}.

A.3 Quadrature rule on the reference quadrilateral

Let $\hat{E}$ be the reference quadrilateral: $\hat{E} = (-1, 1)^2$. We apply the one-dimensional Gauss quadrature rule in each direction:

$$\int_{-1}^{1}\int_{-1}^{1} f(\hat{x}, \hat{y})d\hat{x}d\hat{y} \approx \sum_{i=1}^{Q_{\mathrm{G}}}\sum_{j=1}^{Q_{\mathrm{G}}} w_i w_j f(s_i, s_j).$$



Appendix B
DG codes

B.1 A MATLAB implementation for a one-dimensional problem

The following code solves (1.1)–(1.3) with $K = 1$.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Solution of the elliptic problem
%       -p''(x) = f(x) in (0,1)
%          p(0) = 1
%          p(1) = 0
% with primal discontinuous Galerkin methods
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [Aglobal,rhsglobal,ysol] = DGsimplesolve(nel,ss,penal)
%
% input variables
%       nel: number of subintervals
%       ss: symmetrization parameter equal to 1 for NIPG,
%           0 for IIPG, and -1 for SIPG
%       penal: penalty parameter
% output variables
%       Aglobal: global stiffness matrix
%       rhsglobal: global right-hand side
%       ysol: vector of DG unknowns
% local matrices
  Amat = (nel)*[0 0 0;0 4 0;0 0 16/3];
  Bmat = (nel)*[penal  1-penal  -2+penal;
                -ss-penal -1+ss+penal 2-ss-penal;
                2*ss+penal 1-2*ss-penal -2+2*ss+penal];
  Cmat = (nel)*[penal -1+penal -2+penal;
                ss+penal -1+ss+penal -2+ss+penal;
                2*ss+penal -1+2*ss+penal -2+2*ss+penal];
  Dmat = (nel)*[-penal -1+penal 2-penal;
                -ss-penal -1+ss+penal 2-ss-penal;
                -2*ss-penal -1+2*ss+penal 2-2*ss-penal];
  Emat = (nel)*[-penal 1-penal 2-penal;
                ss+penal -1+ss+penal -2+ss+penal;
                -2*ss-penal 1-2*ss-penal 2-2*ss-penal];
```

```
  F0mat =(nel)*[penal 2-penal -4+penal;
                -2*ss-penal -2+2*ss+penal 4-2*ss-penal;
                4*ss+penal 2-4*ss-penal -4+4*ss+penal];
  FNmat =(nel)*[penal -2+penal -4+penal;
                2*ss+penal -2+2*ss+penal -4+2*ss+penal;
                4*ss+penal -2+4*ss+penal -4+4*ss+penal];

% dimension of local matrices
  locdim = 3;
% dimension of global matrix
  glodim = nel * locdim;
% initialize to zero matrix and right-hand side vector
  Aglobal = zeros(glodim,glodim);
  rhsglobal = zeros(glodim,1);
% Gauss quadrature weights and points
  wg(1) = 1.0;
  wg(2) = 1.0;
  sg(1) = -0.577350269189;
  sg(2) = 0.577350269189;

% assemble global matrix and right-hand side
% first block row
  for ii=1:locdim
    for jj=1:locdim
      Aglobal(ii,jj) = Aglobal(ii,jj)+Amat(ii,jj)+F0mat(ii,jj)+Cmat(ii,jj);
      je = locdim+jj;
      Aglobal(ii,je) = Aglobal(ii,je)+Dmat(ii,jj);
      end; %jj
    end; %ii
% compute right-hand side
  rhsglobal(1) = nel*penal;
  rhsglobal(2) = nel*penal*(-1) - ss*2*nel;
  rhsglobal(3) = nel*penal+ss*4*nel;
  for ig=1:2
    rhsglobal(1) = rhsglobal(1)
                   + wg(ig)*sourcef((sg(ig)+1)/(2*nel))/(2*nel);
    rhsglobal(2) = rhsglobal(2)
                   + wg(ig)*sg(ig)*sourcef((sg(ig)+1)/(2*nel))/(2*nel);
    rhsglobal(3) = rhsglobal(3)
                   + wg(ig)*sg(ig)*sg(ig)*sourcef((sg(ig)+1)/(2*nel))/(2*nel);
    end; %ig

% intermediate block rows
% loop over elements
  for i=2:(nel-1)
   for ii=1:locdim
     ie = ii+(i-1)*locdim;
     for jj=1:locdim
       je = jj+(i-1)*locdim;
       Aglobal(ie,je) = Aglobal(ie,je)+Amat(ii,jj)+Bmat(ii,jj)+Cmat(ii,jj);
       je = jj+(i-2)*locdim;
       Aglobal(ie,je) = Aglobal(ie,je)+Emat(ii,jj);
       je = jj+(i)*locdim;
       Aglobal(ie,je) = Aglobal(ie,je)+Dmat(ii,jj);
       end; %jj
```

```
% compute right-hand side
     for ig=1:2
        rhsglobal(ie) = rhsglobal(ie)
          +wg(ig)*(sg(ig)^(ii-1))*sourcef((sg(ig)+2*(i-1)+1.0)/(2*nel))/(2*nel);
        end; %ig
     end; %ii
   end; %i

% last block row
  for ii=1:locdim
    ie = ii+(nel-1)*locdim;
    for jj=1:locdim
      je = jj+(nel-1)*locdim;
      Aglobal(ie,je) = Aglobal(ie,je)+Amat(ii,jj)+FNmat(ii,jj)+Bmat(ii,jj);
      je = jj+(nel-2)*locdim;
      Aglobal(ie,je) = Aglobal(ie,je)+Emat(ii,jj);
      end; %jj
% compute right-hand side
    for ig=1:2
      rhsglobal(ie) = rhsglobal(ie)
        +wg(ig)*(sg(ig)^(ii-1))*sourcef((sg(ig)+2*(nel-1)+1.0)/(2*nel))/(2*nel);
      end; %ig
    end; %ii

% solve linear system
  ysol = Aglobal\rhsglobal;

  return;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function yval = sourcef(xval)

% source function for exact solution=(1-x)e^(-x^2)
  yval = -(2*xval-2*(1-2*xval)+4*xval*(xval-xval^2))*exp(-xval*xval);

  return;
```

B.2 Selected C routines for higher dimensional problem

We now present several routines written in C that are needed to solve the Poisson problem:

$$\begin{aligned} -\Delta p &= f \quad \text{in} \quad \Omega, \\ p &= 0 \quad \text{on} \quad \partial\Omega. \end{aligned}$$

We consider a two-dimensional domain subdivided into triangles. The data structure is explained in Section 2.9.1. We assume that the same polynomial degree is used everywhere and that the code can be easily modified to handle different polynomial degrees for different elements. In all the routines, the entries in a matrix are actually stored as entries of a long vector, via the definition of macros at the beginning of the routine. The local matrices are assumed to have a maximum number of columns equal to 10, but this can be easily changed in the definition of macros to accommodate polynomial approximation of degree greater

than three. We first give the routine `localmat_vol` that computes the local matrix $\boldsymbol{A}_E$ and the local right-hand side $\boldsymbol{b}_E$ for each mesh element E.

```
// macros for defining the matrices
#define Aloc(i,j) Aloc[ 10*((j)-1) + (i)-1 ]
#define xg(i,j) xg[ 2*((j)-1) + (i)-1 ]
#define der_basis(i,j) der_basis[ 2*((j)-1) + (i)-1 ]

void localmat_vol(
        int E;          // in: number of elements
        double *Aloc;   // out: local matrix
        double *Floc    // out: local right-hand side
        )
{
// local variables
  int k;            // polynomial degree
  int Nloc;         // local dimension
  int ng;           // number of numerical quadrature points
  int ig,idofs,jdofs;
  double *wg;       // quadrature weights
  double *xg;       // local coordinates of quadrature points
  double *val_basis;//values of basis functions
  double *der_basis;//derivatives of basis functions
  double source;    // value of source function
  double determ;    // determinant of Jacobian matrix
  double xx[2];     // global coordinates of quadrature point
// initialize the quadrature weights and points
  ng = 6;
  xg = (double *)malloc (sizeof(double)*2*ng);
  wg = (double *)malloc (sizeof(double)*ng);
  get_quadrature_triangle(ng,xg,wg);

// get polynomial degree
  k = mesh_elt[E].degree;
  Nloc = (k+1)*(k+2)/2;

// allocate memory for the basis
  val_basis = (double *)malloc (sizeof(double)*Nloc);
  der_basis = (double *)malloc (sizeof(double)*2*Nloc);

// initialize to zero the local matrix and right-hand side
  init_zero(Aloc,Nloc*Nloc);
  init_zero(Floc,Nloc*Nloc);

// loop over the quadrature points
  for (ig=1;ig<=ng;ig++) {
// compute values and derivatives of basis functions and determinant
    elem_basis(k,xg(1,ig),xg(2,ig),E,val_basis,der_basis,&determ);
// compute global coordinates of quadrature point
    loc_to_glo_coor(E,xg(1,ig),xg(2,ig),xx);
// get source function
    source_fct(xx,&source);
// compute the entries of local matrix
    for (idofs=1;idofs<=Nloc;idofs++) {
       for (jdofs=1;jdofs<=Nloc;jdofs++) {
          Aloc(idofs,jdofs,Nloc) += (der_basis(1,jdofs)*der_basis(1,idofs)
                                    +der_basis(2,jdofs)*der_basis(2,idofs))
                                    *determ*wg[ig-1];
         }// jdofs
```

```
          Floc[idofs-1] += source*val_basis[idofs-1]*determ*wg[ig-1];
       }// idofs
    }// ig

// free pointers
  free(xg);
  free(wg);
  free(val_basis);
  free(der_basis);

  return;
}
```

The above routine calls other routines that can be written without difficulty. The routine `get_quadrature_triangle` initializes the arrays `xg` and `wg` so that they contain the coordinates of the quadrature points and the weights of the quadrature points given in Table A.2. The routine `init_zero` initializes to zero the arrays `Aloc` and `Floc`. The routine `loc_to_glo_coor` gives the global coordinates of a point $(\hat{x}, \hat{y})$ in the reference triangle, using the mapping defined by (2.30): `xx`$= F_E($`xg`$)$. The routine `source_fct` returns the value of the right-hand side function f in (2.16) evaluated at the point `xx`. Finally, the routine `elem_basis` computes the values and global derivatives of the basis functions, as well as the determinant of the Jacobian matrix $\boldsymbol{B}_E$. This routine is given below: it calls another routine `get_node_coor` that returns the coordinates of the vertices of the triangle E.

```
// macros for defining the matrices
#define dbasis(i,j) dbasis[ 2*((j)-1) + (i)-1 ]
#define inv_BE(i,j) inv_BE[ 2*((j)-1) + (i)-1 ]

void elem_basis(
        int k;          // in: polynomial degree
        double xhat;    // in: x-coordinate of point
        double yhat;    // in: y-coordinate of point
        int E;          // in: element number
        double vbasis;  // out: value of basis function
        double *dbasis; // out: derivative of basis function
        double *determ; // out: determinant of Jacobian matrix
        )
{
// local variables
  int i,ii;
  double nodecoor[6];      // coordinates of vertices of E
  double inv_BE[4];        // inverse of BE
  double *ssn,*ttn;        // monomial values
  double *dttn,*dssn;      // monomial derivatives
  double temp;

// get coordinates of vertices of E
// nodecoor[0],nodecoor[1] are the coordinates of vertex 1
// nodecoor[2],nodecoor[3] are the coordinates of vertex 2
// nodecoor[4],nodecoor[5] are the coordinates of vertex 3
  get_node_coor(E,nodecoor);

// compute det(BE)
  *determ = (nodecoor[2]-nodecoor[0])*(nodecoor[5]-nodecoor[1])
           -(nodecoor[4]-nodecoor[0])*(nodecoor[3]-nodecoor[1]);
```

```
// inverse of BE
  inv_BE(1,1) = (nodecoor[5]-nodecoor[1])/(*determ);
  inv_BE(2,2) = (nodecoor[2]-nodecoor[0])/(*determ);
  inv_BE(1,2) = -(nodecoor[4]-nodecoor[0])/(*determ);
  inv_BE(2,1) = -(nodecoor[3]-nodecoor[1])/(*determ);
// monomial basis
  ssn = (double *)malloc(sizeof(double)*(k+1));
  ttn = (double *)malloc(sizeof(double)*(k+1));
  dssn = (double *)malloc(sizeof(double)*(k+1));
  dttn = (double *)malloc(sizeof(double)*(k+1));
  ssn[0] = 1.0;
  ttn[0] = 1.0;
  dssn[0] = 0.0;

  dttn[0] = 0.0;
  ssn[1] = xhat;
  ttn[1] = yhat;
  dssn[1] = 1.0;
  dttn[1] = 1.0;
  for (i=2;i<=k;i++) {
      ssn[i] = xhat*ssn[i-1];
      ttn[i] = yhat*ttn[i-1];
      dssn[i] = xhat*dssn[i-1];
      dttn[i] = yhat*dttn[i-1];
      }
  for (i=2;i<=k; i++) {
      dssn[i] *= i;
      dttn[i] *= i;
      }

// values of basis functions
   ii=0;
   for (i=0;i<=k;i++) {
      for (j=0;j<=i;j++) {
        vbasis[ii] = ssn[i-j]*ttn[j];
        ii = ii+1;
        }// j
      }// i

// derivatives of basis functions
    ii=1;
    for (i=0;i<=k;i++) {
      for (j=0;j<=i;j++) {
// local derivatives
        dbasis(1,ii) = dssn[i-j]*ttn[j];
        dbasis(2,ii) = ssn[i-j]*dttn[j];
// global derivatives
        temp=dbasis(1,ii)*inv_BE(1,1)+dbasis(2,ii)*inv_BE(2,1);
        dbasis(2,ii)=dbasis(1,ii)*inv_BE(1,2)+dbasis(2,ii)*inv_BE(2,2);
        dbasis(1,ii)=temp;
        ii = ii+1;
        } // j
      } // i

  return;
}
```

The next algorithm computes the local stiffness matrices obtained by the integration over one edge numbered `iface`. We assume that the edge is an interior edge shared by two triangles. From this routine, it is easy to write the routine for the local stiffness matrix associated with a boundary edge. The choice of the method is defined by a variable `penal` and macros `NIPG,SIPG,IIPG`.

```
// macros
#define NIPG (1)
#define SIPG (2)
#define IIPG (3)
#define Bloc11(i,j) Bloc11[ 10*((j)-1) + (i)-1 ]
#define Bloc22(i,j) Bloc22[ 10*((j)-1) + (i)-1 ]
#define Bloc12(i,j) Bloc12[ 10*((j)-1) + (i)-1 ]
#define Bloc21(i,j) Bloc21[ 10*((j)-1) + (i)-1 ]
#define der_basis1(i,j) der_basis1[ 2*((j)-1) + (i)-1 ]
#define der_basis2(i,j) der_basis2[ 2*((j)-1) + (i)-1 ]

void localmat_face(
        int iface;      // in: number of edges
        int penal;      // in: type of primal method
        double *Bloc11; // out: local matrix
        double *Bloc22; // out: local matrix
        double *Bloc12; // out: local matrix
        double *Bloc21  // out: local matrix
        )
{

// local variables
  int k;           // polynomial degree
  int Nloc;        // local dimension
  int ng;          // number of numerical quadrature points
  int E1, E2;      // elements neighbors of edge
  int ig,idofs,jdofs;
  double *wg;      // quadrature weights
  double *xg;      // quadrature points on segment
  double *val_basis1;//values of basis functions for neighbor E1
  double *der_basis1;//derivatives of basis functions for neighbor E1
  double *val_basis2;//values of basis functions for neighbor E2
  double *der_basis2;//derivatives of basis functions for neighbor E2
  double determ1,determ2;// determinant of Jacobian matrix
  double ss1[2],ss2[2]; // local coordinates of quadrature points
  double normal_vec[2];// normal vector to edge pointing from E1 to E2
  double area;     // length of edge
  double eps_ns;   // symmetrization parameter
  double val_penal;// value of penalty

// initialize the quadrature weights and points
  ng = 3;
  xg = (double *)malloc (sizeof(double)*ng);
  wg = (double *)malloc (sizeof(double)*ng);
  get_quadrature_segment(ng,xg,wg);

// get neighbors of edge
  E1 = meshface[iface].neighbor[0];
  E2 = meshface[iface].neighbor[1];
```

```
// get polynomial degree
  k = mesh_elt[E1].degree;
  Nloc = (k+1)*(k+2)/2;

// define the method
  if (penal==NIPG) {
    eps_ns = 1;
    }
  else if (penal==IIPG) {
    eps_ns = 0;
    }
  else if (penal==SIPG) {
    eps_ns = -1;
    }
  get_penalty(&val_penal);

// loop over the quadrature points
  for (ig=0;ig<ng;ig++) {
// compute local coordinates of quadrature point
    loc_coor_quad(iface,E1,E2,xg[ig],ss1,ss2);
// compute normal vector to E1 and length of edge
    calc_normal_face(iface,E1,normal_vec);
    calc_length_face(iface,&area);
// compute values and derivatives of basis functions and determinant
    elem_basis(k,ss1[0],ss1[1],E1,val_basis1,der_basis1,&determ1);
    elem_basis(k,ss2[0],ss2[1],E2,val_basis2,der_basis2,&determ2);

// compute the entries of local matrix Bloc11
    for (idofs=1;idofs<=Nloc;idofs++) {
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
        Bloc11(idofs,jdofs,Nloc) += (der_basis1(1,jdofs)*normal_vec[0]
                                     +der_basis1(2,jdofs)*normal_vec[1])
                                    *val_basis1[idofs-1]*area*wg[ig]*(-0.5);
        Bloc11(idofs,jdofs,Nloc) += (der_basis1(1,idofs)*normal_vec[0]
                                     +der_basis1(2,idofs)*normal_vec[1])
                                    *val_basis1[jdofs-1]*area*wg[ig]*(0.5)
                                    *eps_ns;
        Bloc11(idofs,jdofs,Nloc) += val_penal*val_basis1[idofs-1]
                                    *val_basis1[jdofs-1]*wg[ig];
        }//jdofs
      }//idofs

// compute the entries of local matrix Bloc22
    for (idofs=1;idofs<=Nloc;idofs++) {
       for (jdofs=1;jdofs<=Nloc;jdofs++) {
           Bloc22(idofs,jdofs,Nloc) += (der_basis2(1,jdofs)*normal_vec[0]
                                        +der_basis2(2,jdofs)*normal_vec[1])
                                         *val_basis2[idofs-1]*area*wg[ig]
                                         *(0.5);
           Bloc22(idofs,jdofs,Nloc) += (der_basis2(1,idofs)*normal_vec[0]
                                        +der_basis2(2,idofs)*normal_vec[1])
                                         *val_basis2[jdofs-1]*area*wg[ig]
                                         *(-0.5)*eps_ns;
           Bloc22(idofs,jdofs,Nloc) += val_penal*val_basis2[idofs-1]
                                       *val_basis2[jdofs-1]*wg[ig];
           }//jdofs
       }//idofs
```

```
// compute the entries of local matrix Bloc12
    for (idofs=1;idofs<=Nloc;idofs++) {
       for (jdofs=1;jdofs<=Nloc;jdofs++) {
           Bloc12(idofs,jdofs,Nloc) += (der_basis2(1,jdofs)*normal_vec[0]
                                        +der_basis2(2,jdofs)*normal_vec[1])
                                         *val_basis1[idofs-1]*area*wg[ig]
                                         *(-0.5);
           Bloc12(idofs,jdofs,Nloc) += (der_basis1(1,idofs)*normal_vec[0]
                                        +der_basis1(2,idofs)*normal_vec[1])
                                         *val_basis2[jdofs-1]*area*wg[ig]
                                         *(-0.5)*eps_ns;
           Bloc12(idofs,jdofs,Nloc) -= val_penal*val_basis1[idofs-1]
                                       *val_basis2[jdofs-1]*wg[ig];
           }//jdofs
       }//idofs

// compute the entries of local matrix Bloc21
    for (idofs=1;idofs<=Nloc;idofs++) {
       for (jdofs=1;jdofs<=Nloc;jdofs++) {
           Bloc21(idofs,jdofs,Nloc) += (der_basis1(1,jdofs)*normal_vec[0]
                                        +der_basis1(2,jdofs)*normal_vec[1])
                                         *val_basis2[idofs-1]*area*wg[ig]
                                         *(0.5);
           Bloc21(idofs,jdofs,Nloc) += (der_basis2(1,idofs)*normal_vec[0]
                                        +der_basis2(2,idofs)*normal_vec[1])
                                         *val_basis1[jdofs-1]*area*wg[ig]
                                         *(0.5)*eps_ns;
           Bloc21(idofs,jdofs,Nloc) -= val_penal*val_basis2[idofs-1]
                                       *val_basis1[jdofs-1]*wg[ig];
           }//jdofs
       }//idofs

    }// ig

// free pointers
  free(xg);
  free(wg);
  free(val_basis1);
  free(der_basis1);
  free(val_basis2);
  free(der_basis2);

  return;
}
```

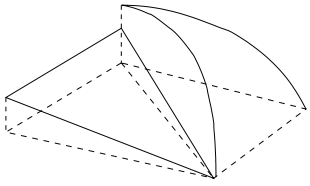
The routine `localmat_face` calls the routine `get_quadrature_segment` that initializes the weights and nodes of the Gauss quadrature rule given in Table A.1. The routine `get_penalty` simply returns the penalty value set by the user. The routine `loc_coor_quad` returns the coordinates of the quadrature point on the edges $\hat{e}_1$ and $\hat{e}_2$ of the reference triangle that correspond to the quadrature point on the segment $[-1, 1]$. The edges $\hat{e}_1$ and $\hat{e}_2$ are such that the edge number `iface` is the image of $\hat{e}_1$ by the mapping F_{E_1} and the image of $\hat{e}_2$ by the mapping F_{E_2}, where E_1 and E_2 are the neighbors of edge `iface`. The routines `calc_normal_face` and `calc_length_face` return the fixed normal vector to the edge and the length of the edge.

The routine for assembling local contributions to global matrix and right-hand sides is provided next.

```
// macros for defining the matrices
#define Aloc(i,j) Aloc[ 10*((j)-1) + (i)-1 ]
#define Bloc11(i,j) Bloc11[ 10*((j)-1) + (i)-1 ]
#define Bloc22(i,j) Bloc22[ 10*((j)-1) + (i)-1 ]
#define Bloc12(i,j) Bloc12[ 10*((j)-1) + (i)-1 ]
#define Bloc21(i,j) Bloc21[ 10*((j)-1) + (i)-1 ]

void global_system(
        int penaltype;     // in: type of primal method
        double *Aglobal;   // out: global matrix
        double *Fglobal    // out: global right-hand side
        )
{

// local variables
  int k;            // polynomial degree
  int Nloc;         // local dimension
  int iel;          // element number
  int iface;        // edge number
  int kaux,idofs,jdofs,ie,je;
  int Nlocmax;     // maximum local dimension
  int E1,E2;       // edge neighbors
  double *Aloc;    // local matrix for volume
  double *Floc;    // local right-hand side
  double *Bloc11,*Bloc22,*Bloc12,*Bloc21;//local matrices for edge

// allocate memory for local matrices and right-hand side
  Aloc = (double *)malloc (sizeof(double)*Nlocmax*Nlocmax);
  Bloc11 = (double *)malloc (sizeof(double)*Nlocmax*Nlocmax);
  Bloc22 = (double *)malloc (sizeof(double)*Nlocmax*Nlocmax);
  Bloc12 = (double *)malloc (sizeof(double)*Nlocmax*Nlocmax);
  Bloc21 = (double *)malloc (sizeof(double)*Nlocmax*Nlocmax);
  Floc = (double *)malloc (sizeof(double)*Nlocmax);

// assemble volume contributions
  kaux=0;
// loop over the elements
  for (iel=1;iel<=Nel;iel++) {
// get polynomial degree
    k = mesh_elt[iel].degree;
    Nloc = (k+1)*(k+2)/2;
// compute local volume matrix
    localmat_vol(iel,Aloc,Floc);
    for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+kaux;
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+kaux;
         Aglobal(ie,je) = Aglobal(ie,je) + Aloc(idofs,jdofs);
         }//jdofs
      Fglobal[ie-1] += Floc[idofs-1];
      }// idofs
    kaux = kaux+Nloc;
    }// iel
```

```
// assemble edge contributions
// loop over the edges
  for (iface=1;iface<=Nface;iface++) {
// get neighbors of iface
    E1 = meshface[iface].neighbor[0];
    E2 = meshface[iface].neighbor[1];
// get polynomial degree
    k = mesh_elt[E1].degree;
    Nloc = (k+1)*(k+2)/2;

// for interior face only
    if (meshface[iface].bctype==0) {
// compute local matrices
     localmat_face(iface,penaltype,Bloc11,Bloc22,Bloc12,Bloc21);

// assemble Bloc11
     for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+(Nloc*(E1-1));
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+(Nloc*(E1-1));
         Aglobal(ie,je) = Aglobal(ie,je) + Bloc11(idofs,jdofs);
         }// jdofs
      }// idofs

// assemble Bloc22
     for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+(Nloc*(E2-1));
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+(Nloc*(E2-1));
         Aglobal(ie,je) = Aglobal(ie,je) + Bloc22(idofs,jdofs);
         }// jdofs
      }// idofs

// assemble Bloc12
     for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+(Nloc*(E1-1));
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+i(Nloc*(E2-1));
         Aglobal(ie,je) = Aglobal(ie,je) + Bloc12(idofs,jdofs);
         }// jdofs
      }// idofs

// assemble Bloc21
     for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+i(Nloc*(E2-1));
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+(Nloc*(E1-1));
         Aglobal(ie,je) = Aglobal(ie,je) + Bloc21(idofs,jdofs);
         }// jdofs
      }// idofs

    }// interior face only

   else { // Dirichlet boundary face
// compute local matrix
     localmat_bdyface(iface,penaltype,Bloc11);
```

```
// assemble Bloc11
     for (idofs=1;idofs<=Nloc;idofs++) {
      ie = idofs+(Nloc*(E1-1));
      for (jdofs=1;jdofs<=Nloc;jdofs++) {
         je = jdofs+(Nloc*(E1-1));
         Aglobal(ie,je) = Aglobal(ie,je) + Bloc11(idofs,jdofs);
         }// jdofs
      }// idofs
     }

  free(Aloc);
  free(Floc);
  free(Bloc11);
  free(Bloc22);
  free(Bloc12);
  free(Bloc21);

  return;
}
```



Appendix C
An approximation result

The stability and analysis of primal DG methods depend strongly on the value of the penalty parameter. The penalty is either large enough for the SIPG and IIPG methods or strictly positive for the NIPG method. If the penalty term is removed, the bilinear form of the NIPG method remains coercive. The resulting method, denoted NIPG 0, is convergent for polynomial degrees greater than or equal to two in 2D or 3D. The proof of the a priori error estimates uses the following lemmas.

Lemma C.1. *Let E be a triangle or a parallelogram in* 2D *or a tetrahedron in* 3D*. Let $\boldsymbol{n}_E$ be the outward normal to the boundary ∂E. Let $\boldsymbol{K}$ be a constant matrix. Fix one edge (or face) e of E. For any $f \in H^s(E)$, with $s \geq 2$, there exists a polynomial $q \in \mathbb{P}_2(E)$ such that*

$$\int_e \boldsymbol{K}\nabla q \cdot \boldsymbol{n}_E = \int_e \boldsymbol{K}\nabla f \cdot \boldsymbol{n}_E, \tag{C.1}$$

$$\int_{e'} \boldsymbol{K}\nabla q \cdot \boldsymbol{n}_E = 0 \quad \textit{for} \quad e' \in \partial E \setminus e, \tag{C.2}$$

$$\forall i = 0, 1, 2, \quad \|\nabla^i q\|_{L^2(E)} \leq C h_E^{1/2-i} |E|^{1/2} \|\nabla f \cdot \boldsymbol{n}_E\|_{L^2(e)}. \tag{C.3}$$

Proof. We first prove the result for a triangle E with vertices A_1, A_2, A_3 and edges $e_1 = [A_2A_3]$, $e_2 = [A_1A_3]$, $e_3 = [A_1A_2]$ with respective unit exterior normal vectors $\boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3$. Let $\lambda_1, \lambda_2, \lambda_3$ be the barycentric coordinates of A_1, A_2, and A_3 in E. Define the quadratic polynomial

$$q = 4\alpha\lambda_1(1 - \lambda_1).$$

From the definition of the barycentric coordinates, we see that $\alpha = q(A_{12})$, where A_{12} is the midpoint of edge e_3. Next, we compute the gradient of q:

$$\nabla q = 4\alpha\nabla\lambda_1(1 - 2\lambda_1).$$

Since each component of $\nabla\lambda_1$ is a constant, and since $1-2\lambda_1$ is a linear function vanishing at the midpoints of edges e_2 and e_3, we have

$$\int_{e_2} \boldsymbol{K}\nabla q\cdot\boldsymbol{n}_E = \int_{e_3} \boldsymbol{K}\nabla q\cdot\boldsymbol{n}_E = 0.$$

It remains to satisfy condition (C.1) by carefully choosing the scalar α:

$$\int_{e_1} \nabla q\cdot\boldsymbol{n}_E = 4\alpha\int_{e_1} \boldsymbol{K}\nabla\lambda_1(1-2\lambda_1)\cdot\boldsymbol{n}_E = \int_{e_1} \boldsymbol{K}\nabla f\cdot\boldsymbol{n}_E.$$

Equivalently, since λ_1 vanishes on e_1, we have

$$4\alpha\int_{e_1} \boldsymbol{K}\nabla\lambda_1\cdot\boldsymbol{n}_E = \int_{e_1} \boldsymbol{K}\nabla f\cdot\boldsymbol{n}_E.$$

Let $|e_1|$ be the length of e_1 and $|E|$ the area of E. We compute

$$\nabla\lambda_1 = -\frac{\boldsymbol{n}_E}{2}\frac{|e_1|}{|E|}. \tag{C.4}$$

Therefore, we can solve for α:

$$\alpha = -\frac{1}{2}\frac{|E|}{|e_1|^2}\frac{1}{\boldsymbol{K}\boldsymbol{n}_E\cdot\boldsymbol{n}_E}\int_{e_1} \boldsymbol{K}\nabla f\cdot\boldsymbol{n}_E.$$

Hence, using Cauchy–Schwarz's inequality, we have for some constant C

$$|\alpha| \le C\Big|\int_{e_1} \boldsymbol{K}\nabla f\cdot\boldsymbol{n}_E\Big| \le Ch_E^{\frac{1}{2}}\|\nabla f\cdot\boldsymbol{n}_E\|_{L^2(e_1)}.$$

Therefore, we obtain a bound on the L^2 norm of q:

$$\|q\|_{L^2(\Omega)} \le C|\alpha|\|\lambda_1(1-\lambda_1)\|_{L^2(E)} \le C|E|^{1/2}h_E^{1/2}\|\nabla f\cdot\boldsymbol{n}_E\|_{L^2(e_1)}.$$

Using (C.4), we obtain a bound on the gradient norm of q:

$$\|\nabla q\|_{L^2(\Omega)} \le |\alpha|\|\nabla\lambda_1(1-2\lambda_1)\|_{L^2(E)} \le C|\alpha|\frac{|e_1|}{|E|}|E|^{1/2} \le C|E|^{1/2}h_E^{-1/2}\|\nabla f\cdot\boldsymbol{n}_E\|_{L^2(e_1)},$$

and similarly

$$\|\nabla^2 q\|_{L^2(\Omega)} \le C|E|^{1/2}h_E^{-3/2}\|\nabla f\cdot\boldsymbol{n}_E\|_{L^2(e_1)}.$$

Next, we consider a tetrahedron E with vertices A_1, A_2, A_3, A_4, opposite faces e_1, e_2, e_3, e_4, and unit exterior normal vectors $\boldsymbol{n}_1$, $\boldsymbol{n}_2$, $\boldsymbol{n}_3$, $\boldsymbol{n}_4$. The argument is similar to the one used for triangles. Let λ_1, λ_2, λ_3, λ_4 be the barycentric coordinates of A_1, A_2, A_3, and A_4 in E. Consider the polynomial

$$q = \alpha\lambda_1(3\lambda_1-2).$$

We compute the gradient of q:

$$\nabla q = \alpha\nabla\lambda_1(6\lambda_1-2).$$

Each component of ∇q has zero mean value on e_2, e_3, e_4 and $(\nabla\lambda_1)\lambda_1$ vanishes on e_1. Therefore, α is determined by the condition:

$$-2\alpha \int_{e_1} \boldsymbol{K}\nabla\lambda_1 \cdot \boldsymbol{n}_1 = \int_{e_1} \boldsymbol{K}\nabla f \cdot \boldsymbol{n}_1,$$

and since (C.4) is valid in 3D, this holds if

$$\alpha = \frac{|E|}{|e_1|^2} \frac{1}{(\boldsymbol{K}\boldsymbol{n}_1, \boldsymbol{n}_1)} \int_{e_1} \boldsymbol{K}\nabla f \cdot \boldsymbol{n}_1.$$

Hence there exists a constant C such that

$$|\alpha| \le C\|\nabla f \cdot \boldsymbol{n}_E\|_{L^2(e_1)},$$

and for $i = 0, 1, 2$ we obtain

$$\|\nabla^i q\|_{L^2(E)} \le C|E|^{\frac{1}{2}} h_E^{1/2-i} \|\nabla f \cdot \boldsymbol{n}_E\|_{L^2(e_1)}.$$

In the case of parallelograms, the proof is more technical (see [96]). □

Lemma C.2. *Let E be a triangle or a parallelogram in* 2D *or a tetrahedron in* 3D *with diameter h_E. Let $\boldsymbol{n}_E$ be the outward norm to the boundary ∂E. Let $\boldsymbol{K}$ be a constant matrix and let $k \ge 2$. For any $p \in H^s(E)$, with $s \ge 2$, there exist an approximation $\tilde{p} \in \mathbb{P}_k(E)$ and a constant $C > 0$ independent of h_E such that*

$$\forall e \in \partial E, \quad \int_e \boldsymbol{K}\nabla(\tilde{p} - p) \cdot \boldsymbol{n}_E = 0, \tag{C.5}$$

$$\forall i = 0, 1, 2, \quad \|\nabla^i(\tilde{p} - p)\|_{L^2(E)} \le C h_E^{\min(k+1,s)-i} \|p\|_{H^s(E)}. \tag{C.6}$$

Proof. We give the proof for the case of the triangle only. The other cases are treated in a similar way. Let E be a triangle with edges e_1, e_2, and e_3 and with unit normal vector $\boldsymbol{n}_1$, $\boldsymbol{n}_2$, and $\boldsymbol{n}_3$, respectively. Let $p \in H^s(\Omega)$ for $s \ge 2$ and let $p^I \in \mathbb{P}_k(E)$ be an approximation of p satisfying (2.10). Set $f = p - p^I$ and use Lemma C.1 to construct a polynomial q_1 in $\mathbb{P}_2(E)$ such that

$$\int_{e_1} \boldsymbol{K}\nabla(q_1 - f) \cdot \boldsymbol{n}_1 = 0,$$

$$\int_{e_2} \boldsymbol{K}\nabla q_1 \cdot \boldsymbol{n}_2 = 0,$$

$$\int_{e_3} \boldsymbol{K}\nabla q_1 \cdot \boldsymbol{n}_3 = 0.$$

Besides, for $i = 0, 1, 2$,

$$\|\nabla^i q_1\|_{L^2(E)} \le C|E|^{\frac{1}{2}} h_E^{\frac{1}{2}-i} \|\nabla f \cdot \boldsymbol{n}_E\|_{L^2(e_1)}. \tag{C.7}$$

Similarly, we construct polynomials q_2 and q_3 in $\mathbb{P}_2(E)$ such that

$$\int_{e_\delta} \boldsymbol{K}\nabla q_\delta \cdot \boldsymbol{n}_\delta = \int_{e_\delta} \boldsymbol{K}\nabla f \cdot \boldsymbol{n}_\delta \quad \text{for} \quad \delta = 2, 3,$$

$$\int_{e_1} \boldsymbol{K}\nabla q_2 \cdot \boldsymbol{n}_1 = \int_{e_3} \boldsymbol{K}\nabla q_2 \cdot \boldsymbol{n}_3 = 0,$$
$$\int_{e_1} \boldsymbol{K}\nabla q_3 \cdot \boldsymbol{n}_1 = \int_{e_2} \boldsymbol{K}\nabla q_3 \cdot \boldsymbol{n}_2 = 0.$$

Let $q = q_1 + q_2 + q_3$ and set $\tilde{p} = q + p^I$. Then $\tilde{p}$ satisfies (C.5), and we derive, for $i = 0, 1, 2$,

$$\begin{aligned}\|\nabla^i(\tilde{p} - p)\|_{L^2(E)} &\le \|\nabla^i q\|_{L^2(E)} + \|\nabla^i(p^I - p)\|_{L^2(E)} \\ &\le Ch_E^{\min(k+1,s)-i}\|p\|_{H^s(E)} + \|\nabla^i(p^I - p)\|_{L^2(E)},\end{aligned}$$

which has the same order of approximation as $|\nabla^i(p^I - p)|_{L^2(E)}$. □

Bibliography

[1] D. N. ARNOLD, *An interior penalty finite element method with discontinuous elements*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 742–760.

[2] D. ARNOLD, D. BOFFI, AND R. FALK, *Approximation by quadrilateral finite element*, Mathematics of Computation, 239 (2002), pp. 909–922.

[3] D. ARNOLD, F. BREZZI, B. COCKBURN, AND D. MARINI, *Discontinuous Galerkin methods for elliptic problems*, in First International Symposium on Discontinuous Galerkin Methods, B. Cockburn, G. Karniadakis, and C.-W. Shu, eds., vol. 11 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, pp. 89–101.

[4] D. N. ARNOLD, F. BREZZI, B. COCKBURN, AND L. D. MARINI, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM Journal on Numerical Analysis, 39 (2002), pp. 1749–1779.

[5] U. M. ASCHER AND L. R. PETZOLD, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia, 1998.

[6] K. ATKINSON, *An Introduction to Numerical Analysis*, Second Edition, Wiley, New York, 1989.

[7] I. BABUŠKA, *The finite element method with Lagrangian multipliers*, Numerische Mathematik, 20 (1973), pp. 179–192.

[8] I. BABUŠKA, C. BAUMANN, AND J. ODEN, *A discontinuous hp finite element method for diffusion problems:* 1*-D analysis*, Computers & Mathematics with Applications, 37 (1999), pp. 103–122.

[9] I. BABUŠKA AND M. SURI, *The optimal convergence rates of the p-version of the finite element method*, SIAM Journal on Numerical Analysis, 24 (1987), pp. 750–776.

[10] G. BAKER, *Finite element methods for elliptic equations using nonconforming elements*, Mathematics of Computation, 31 (1977), pp. 45–59.

[11] G. A. BAKER, W. N. JUREIDINI, AND O. A. KARAKASHIAN, *Piecewise solenoidal vector fields and the Stokes problem*, SIAM Journal on Numerical Analysis, 27 (1990), pp. 1466–1485.

[12] F. BASSI AND S. REBAY, *A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations*, Journal of Computational Physics, 131 (1997), pp. 267–279.

[13] P. BASTIAN AND B. RIVIÈRE, *Superconvergence and H(div) projection for discontinuous Galerkin methods*, International Journal for Numerical Methods in Fluids, 42 (2003), pp. 1043–1057.

[14] J. BERGH AND L. LOFSTROM, *Interpolation Spaces*, Springer-Verlag, Berlin, 1976.

[15] S. BRENNER, *Korn's inequalities for piecewise H^1 vector fields*, Mathematics of Computation, 73 (2003), pp. 1067–1087.

[16] S. C. BRENNER, *Poincaré–Friedrichs inequalities for piecewise H^1 functions*, SIAM Journal on Numerical Analysis, 41 (2003), pp. 306–324.

[17] S. BRENNER AND L. SCOTT, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[18] F. BREZZI, *On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers*, Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge, 8 (1974), pp. 129–151.

[19] F. BREZZI, B. COCKBURN, L. MARINI, AND E. SÜLI, *Stabilization mechanisms in discontinuous Galerkin finite element methods*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 3293–3310.

[20] R. BROOKS AND A. COREY, *Hydraulic properties of porous media*, Hydrology Paper 3, Colorado State University, Fort Collins, CO, 1964.

[21] J. BUTCHER, *On the attainable order of Runge-Kutta methods*, Mathematics of Computation, 19 (1965), pp. 408–417.

[22] J. BUTCHER, *The Numerical Analysis of Ordinary Differential Equations*, Wiley, New York, 1987.

[23] P. CASTILLO, *Performance of discontinuous Galerkin methods for elliptic PDEs*, SIAM Journal on Scientific Computing, 24 (2002), pp. 524–547.

[24] P. CASTILLO, B. COCKBURN, I. PERUGIA, AND D. SCHÖTZAU, *An a priori error analysis of the local discontinuous Galerkin method for elliptic problems*, SIAM Journal on Numerical Analysis, 38 (2000), pp. 1676–1706.

[25] P. CASTILLO, B. COCKBURN, AND C. SCHWAB, *Optimal a priori error analysis for the hp-version of the local discontinuous Galerkin method for convection-diffusion problems*, Mathematics of Computation, 71 (2002), pp. 455–478.

[26] G. CHAVENT AND J. JAFFRÉ, *Mathematical Models and Finite Elements for Reservoir Simulation*, North–Holland, Amsterdam, 1986.

[27] Z. CHEN, G. HUAN, AND Y. MA, *Computational Methods for Multiphase Flows in Porous Media*, SIAM, Philadelphia, 2006.

[28] P. CIARLET, *The finite element method for elliptic problems*, North–Holland, Amsterdam, 1977.

[29] P. CIARLET, *Mathematical Elasticity. Volume* I: *Three-Dimensional Elasticity*, North–Holland, Amsterdam, 1988.

[30] B. COCKBURN AND C. DAWSON, *Some extensions of the local discontinuous Galerkin method for convection-diffusion equations in multidimensions*, in Mathematics of Finite Elements and Applications: MAFELAP X, J. Whiteman, ed., Elsevier, Oxford, UK, 2000, pp. 225–238.

[31] B. COCKBURN, J. GUNZMAN, AND B. RIVIÈRE, *Convergence of non-symmetric discontinuous Galerkin methods on non-uniform meshes*, in preparation.

[32] B. COCKBURN, G. KANSCHAT, AND D. SCHÖTZAU, *The local discontinuous Galerkin method for the Oseen equations*, Mathematics of Computation, 73 (2003), pp. 569–593.

[33] B. COCKBURN, G. KANSCHAT, AND D. SCHÖTZAU, *A note on discontinuous Galerkin divergence-free solutions of the Navier-Stokes equations*, Journal of Scientific Computing, 31 (2007), pp. 61–73.

[34] B. COCKBURN, G. KANSCHAT, D. SCHÖTZAU, AND C. SCHWAB, *Local discontinuous Galerkin methods for the Stokes system*, SIAM Journal on Numerical Analysis, 40 (2002), pp. 319–343.

[35] B. COCKBURN, G. KARNIADAKIS, AND C.-W. SHU, EDS., *Discontinuous Galerkin Methods: Theory, Computation, and Applications*, vol. 11 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000.

[36] B. COCKBURN AND C.-W. SHU, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM Journal on Numerical Analysis, 35 (1998), pp. 2440–2463.

[37] B. COCKBURN AND C.-W. SHU, *The Runge-Kutta discontinuous Galerkin method for conservation laws* V, Journal of Computational Physics, 141 (1998), pp. 199–224.

[38] M. CROUZEIX AND R. FALK, *Non conforming finite elements for the Stokes problem*, Mathematics of Computation, 52 (1989), pp. 437–456.

[39] M. CROUZEIX AND P.-A. RAVIART, *Conforming and nonconforming finite element methods for solving the stationary Stokes equations*. I, Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge, 7 (1973), pp. 33–75.

[40] C. DAWSON, *The $\mathcal{P}^{k+1} - \mathcal{S}^k$ local discontinuous Galerkin method for elliptic equations*, SIAM Journal on Numerical Analysis, 40 (2002), pp. 2151–2170.

[41] C. DAWSON AND J. PROFT, *A priori error estimates for interior penalty versions of local discontinuous Galerkin methods applied to transport problems*, Numerical Methods for Partial Differential Equations, 17 (2001), pp. 545–564.

[42] C. DAWSON, S. SUN, AND M. WHEELER, *Compatible algorithms for coupled flow and transport*, Computer Methods in Applied Mechanics and Engineering, 193 (2004), pp. 2565–2580.

[43] L. DELVES AND C. HALL, *An implicit matching principle for global element calculations*, Journal of the Institute of Mathematics and its Applications, 23 (1979), pp. 223–234.

[44] D. DUNAVANT, *High degree efficient symmetrical Gaussian quadrature rules for the triangle*, International Journal for Numerical Methods in Engineering, 21 (1985), pp. 1129–1148.

[45] L. DURLOFSKY, *Accuracy of mixed and control volume finite element approximations to Darcy velocity and related quantities*, Water Resources Research, 30 (1994), pp. 965–973.

[46] L. DURLOFSKY, B. ENGQUIST, AND S. OSHER, *Triangle based adaptive stencils for the solution of hyperbolic conservation laws*, Journal of Computational Physics, 98 (1992), pp. 64–73.

[47] G. DUVAUT AND J. LIONS, *Inequalities in Mechanics and Physics*, Springer-Verlag, Berlin, 1976.

[48] Y. EPSHTEYN AND B. RIVIÈRE, *On the solution of incompressible two-phase flow by a p-version discontinuous Galerkin method*, Communications in Numerical Methods in Engineering, 22 (2006), pp. 741–751.

[49] Y. EPSHTEYN AND B. RIVIÈRE, *Analysis of hp discontinuous Galerkin methods for incompressible two-phase flow*, Technical Report TR-MATH 06-17, University of Pittsburgh, Pittsburgh, PA 2006.

[50] Y. EPSHTEYN AND B. RIVIÈRE, *Estimation of penalty parameters for symmetric interior penalty Galerkin methods*, Journal of Computational and Applied Mathematics, 206 (2007), pp. 843–872.

[51] Y. EPSHTEYN AND B. RIVIÈRE, *Fully implicit discontinuous finite element methods for two-phase flow*, Applied Numerical Mathematics, 57 (2007), pp. 383–401.

[52] Y. EPSHTEYN AND B. RIVIÈRE, *High order methods for miscible displacement*, to appear in International Journal of Numerical Analysis and Modeling. Also Technical Report, Carnegie Mellon University, Pittsburgh, PA, 2007.

[53] K. ERIKSSON AND C. JOHNSON, *Error estimates and automatic time step control for nonlinear parabolic problems*, I, SIAM Journal on Numerical Analysis, 24 (1987), pp. 12–23.

[54] K. ERIKSSON AND C. JOHNSON, *Adaptive finite element methods for parabolic problems* II: *Optimal error estimates in $L_\infty L_2$ and $L_\infty L_\infty$*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 706–740.

[55] D. ESTEP, *A posteriori error bounds and global error control for approximation of ordinary differential equations*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 1–48.

[56] M. FORTIN AND M. SOULIE, *A non-conforming piecewise quadratic finite element on triangles*, International Journal for Numerical Methods in Engineering, 19 (1983), pp. 505–520.

[57] M. VAN GENUCHTEN, *A closed-form equation for predicting the hydraulic conductivity of unsaturated soils*, Soil Science Society of America Journal, 44 (1980), pp. 892–898.

[58] V. GIRAULT, *A local projection operator for quadrilateral finite elements*, Mathematics of Computation, 64 (1995), pp. 1421–1431.

[59] V. GIRAULT AND P. RAVIART, *Finite Element Methods for Navier-Stokes Equations: Theory and Algorithms*, vol. 5 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 1986.

[60] V. GIRAULT AND B. RIVIÈRE, *DG approximation of coupled Navier-Stokes and Darcy equations by Beaver-Joseph-Saffman interface condition*, SIAM Journal on Numerical Analysis, submitted, also Technical Report TR-MATH 07-09.

[61] V. GIRAULT, B. RIVIÈRE, AND M. WHEELER, *A discontinuous Galerkin method with non-overlapping domain decomposition for the Stokes and Navier-Stokes problems*, Mathematics of Computation, 74 (2005), pp. 53–84.

[62] V. GIRAULT, B. RIVIÈRE, AND M. WHEELER, *A splitting method using discontinuous Galerkin for the transient incompressible Navier-Stokes equations*, Mathematical Modelling and Numerical Analysis, 39 (2005), pp. 1115–1148.

[63] V. GIRAULT AND R. SCOTT, *A quasi-local interpolation operator preserving the discrete divergence*, Calcolo, 40 (2003), pp. 1–19.

[64] S. GOTTLIEB, C.-W. SHU, AND E. TADMOR, *Strong stability-preserving high-order time discretization methods*, SIAM Review, 43 (2001), pp. 89–112.

[65] M. GURTIN, *The linear theory of elasticity*, in Handbuch der Physik, Springer-Verlag, Berlin, 1972, pp. 1–295.

[66] P. HANSBO AND M. LARSON, *Discontinuous finite element methods for incompressible and nearly incompressible elasticity by use of Nitsche's method*, Computer Methods in Applied Mechanics and Engineering, 191 (2002), pp. 1895–1908.

[67] R. HELMIG, *Multiphase Flow and Transport Processes in the Subsurface*, Springer-Verlag, Berlin, 1997.

[68] J. G. HEYWOOD AND R. RANNACHER, *Finite-element approximation of the nonstationary Navier–Stokes problem. Part* IV: *Error analysis for second-order time discretization*, SIAM Journal on Numerical Analysis, 27 (1990), pp. 353–384.

[69] H. HOTEIT AND A. FIROOZABADI, *Numerical modeling of two-phase flow in heterogeneous permeable media with different capillarity pressures*, Advances in Water Resources, 31 (2008), pp. 56–73.

[70] H. HOTEIT, PH. ACKERER, R. MOSÉ, J. ERHEL, AND B. PHILIPPE, *New two-dimensional slope limiters for discontinuous Galerkin methods on arbitrary meshes*, International Journal for Numerical Methods in Engineering, 61 (2004), pp. 2566–2593.

[71] P. HOUSTON, D. SCHÖTZAU, AND E. SÜLI, *An hp-adaptive mixed discontinuous Galerkin FEM for nearly incompressible linear elasticity*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 3224–3246.

[72] P. HOUSTON, C. SCHWAB, AND E. SÜLI, *Discontinuous hp-finite element methods for advection-diffusion-reaction problems*, SIAM Journal on Numerical Analysis, 39 (2002), pp. 2133–2163.

[73] O. A. KARAKASHIAN AND W. N. JUREIDINI, *A nonconforming finite element method for the stationary Navier–Stokes equations*, SIAM Journal on Numerical Analysis, 35 (1998), pp. 93–120.

[74] S. KAYA AND B. RIVIÈRE, *A discontinuous subgrid eddy viscosity method for the time-dependent Navier–Stokes equations*, SIAM Journal on Numerical Analysis, 43 (2005), pp. 1572–1595.

[75] W. KLIEBER AND B. RIVIÈRE, *Adaptive simulations of two-phase flow by discontinuous Galerkin methods*, Computer Methods in Applied Mechanics and Engineering, 196 (2006), pp. 404–419.

[76] E. KOVAL, *A method for predicting the performance of unstable miscible displacement in heterogeneous media*, Society of Petroleum Engineers Journal, 3 (1963), pp. 145–154.

[77] M. LARSSON AND A. NIKLASSON, *Analysis of a family of discontinuous Galerkin methods for elliptic problems: The one-dimensional case*, Numerische Mathematik, 99 (2004), pp. 113–130.

[78] A. LASIS AND E. SÜLI, *hp-version discontinuous Galerkin finite element method for semilinear parabolic problems*, SIAM Journal on Numerical Analysis, 45 (2007), pp. 1544–1569.

[79] P. LAX AND N. MILGRAM, *Parabolic Equations. Contributions to the Theory of Partial Differential Equations*, Princeton University Press, Princeton, NJ, 1954.

[80] P. LESAINT AND P. RAVIART, *On a finite element method for solving the neutron transport equation*, in Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, New York 1974, pp. 89–123.

[81] J. LIONS AND E. MAGENES, *Problèmes aux Limites non Homogènes et Applications*, I, Dunod, Paris, 1968.

[82] J. NECAS, *Les Méthodes Directes en Théorie des Equations Elliptiques*, Masson, Paris, 1967.

[83] J. NITSCHE, *Über ein Variationsprinzip zur Lösung von Dirichlet Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind*, Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg, 36 (1971), pp. 9–15.

[84] J. ODEN, I. BABUŠKA, AND C. BAUMANN, *A discontinuous hp finite element method for diffusion problems*, Journal of Computational Physics, 146 (1998), pp. 491–519.

[85] P. PERCELL AND M. F. WHEELER, *A local residual finite element procedure for elliptic equations*, SIAM Journal on Numerical Analysis, 15 (1978), pp. 705–714.

[86] I. PERUGIA AND D. SCHÖTZAU, *An hp-analysis of the local discontinuous Galerkin method for diffusion problems*, Journal of Scientific Computing, 17 (2002), pp. 561–571.

[87] J. PROFT AND B. RIVIÈRE, *Analytical and numerical study of diffusive fluxes for transport equations with near-degenerate coefficients*, Technical Report TR-MATH 06-07, University of Pittsburgh, Pittsburgh, PA, 2006.

[88] R. A. RAVIART AND J. M. THOMAS, *A mixed finite element method for second order elliptic problems*, in Mathematical Aspects of the Finite Element Method, vol. 606 of Lecture Notes in Mathematics, Springer-Verlag, New York, 1977, pp. 292–315.

[89] B. RIVIÈRE AND V. GIRAULT, *Discontinuous finite element methods for incompressible flows on subdomains with non-matching interfaces*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 3274–3292.

[90] B. RIVIÈRE, S. SHAW, M. WHEELER, AND J.WHITEMAN, *Discontinuous Galerkin finite element methods for linear elasticity and quasistatic viscoelasticity problems*, Numerische Mathematik, 95 (2003), pp. 347–376.

[91] B. RIVIÈRE AND M. WHEELER, *A discontinuous Galerkin method applied to nonlinear parabolic equations*, in First International Symposium on Discontinuous Galerkin Methods, B. Cockburn, G. Karniadakis, and C.-W. Shu, eds., vol. 11 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, pp. 231–244.

[92] B. RIVIÈRE AND M. WHEELER, *Discontinuous Galerkin methods for flow and transport problems in porous media*, Communications in Numerical Methods in Engineering, 18 (2002), pp. 63–68.

[93] B. RIVIÈRE AND M. WHEELER, *Miscible displacement in porous media*, in Computational Methods in Water Resources, S. Hassanizadeh, R. Schotting, W. Gray, and G. Pinder, eds., vol. 47 of Developments in Water Sciences, Elsevier, Amsterdam, 2002, pp. 907–914.

[94] B. RIVIÈRE AND M. WHEELER, *Non conforming methods for transport with nonlinear reaction*, in Fluid Flow and Transport in Porous Media: Mathematical and Numerical Treatment, Z. Chen and R. Ewing, eds., vol. 295 of Contemporary Mathematics, AMS, Providence, RI, 2002, pp. 421–430.

[95] B. RIVIÈRE, M. WHEELER, AND V. GIRAULT, *Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. Part* I, Computational Geosciences, 3 (1999), pp. 337–360.

[96] B. RIVIÈRE, M. F. WHEELER, AND V. GIRAULT, *A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems*, SIAM Journal on Numerical Analysis, 39 (2001), pp. 902–931.

[97] W. RUDIN, *Real Complex Analysis*, McGraw–Hill, New York, 1987.

[98] D. SCHÖTZAU AND C. SCHWAB, *An hp a priori error analysis of the DG time-stepping method for initial value problems*, Calcolo, 37 (2000), pp. 207–232.

[99] D. SCHÖTZAU AND C. SCHWAB, *Time discretization of parabolic problems by the hp-version of the discontinuous Galerkin finite element method*, SIAM Journal on Numerical Analysis, 38 (2000), pp. 837–875.

[100] D. SCHÖTZAU, C. SCHWAB, AND A. TOSELLI, *Mixed hp-DGFEM for incompressible flows*, SIAM Journal on Numerical Analysis, 40 (2003), pp. 2171–2194.

[101] G. STRANG AND G. FIX, *An Analysis of the Finite Element Method*, Prentice–Hall, Englewood Cliffs, NJ, 1973.

[102] S. SUN, B. RIVIÈRE, AND M. WHEELER, *A combined mixed finite element and discontinuous Galerkin method for miscible displacement problem in porous media*, in Recent Progress in Computational and Applied PDEs, Kluwer/Plenum, New York, 2002, pp. 323–351.

[103] S. SUN AND M. WHEELER, *Discontinuous Galerkin methods for coupled flow and reactive transport problems*, Applied Numerical Mathematics, 52 (2005), pp. 273–298.

[104] S. SUN AND M. F. WHEELER, *Symmetric and nonsymmetric discontinuous Galerkin methods for reactive transport in porous media*, SIAM Journal on Numerical Analysis, 43 (2005), pp. 195–219.

[105] S. SUN AND M. WHEELER, *Analysis of discontinuous Galerkin methods for multicomponent reactive transport problems*, Computers and Mathematics with Applications, 52 (2006), pp. 637–650.

[106] V. THOMÉE, *Galerkin Finite Element Methods for Parabolic Problems*, Springer-Verlag, Berlin, 1997.

[107] A. TOSELLI, *hp-finite element discontinuous Galerkin approximations for the Stokes problem*, Mathematical Models and Methods in Applied Sciences, 12 (2002), pp. 1565–1616.

[108] T. WARBURTON AND J. HESTHAVEN, *On the constants in hp-finite element trace inverse inequalities*, Computer Methods in Applied Mechanics and Engineering, 192 (2003), pp. 2765–2773.

[109] M. F. WHEELER, *An elliptic collocation-finite element method with interior penalties*, SIAM Journal on Numerical Analysis, 15 (1978), pp. 152–161.

[110] T. WIHLER, *Locking-free DGFEM for elasticity problems in polygons*, IMA Journal of Numerical Analysis, 24 (2004), pp. 45–75.

[111] T. WIHLER, *Locking-free adaptive discontinuous Galerkin FEM for linear elasticity problems*, Mathematics of Computation, 75 (2006), pp. 1087–1102.

Index

Discontinuous Galerkin Methods for
Solving Elliptic and Parabolic Equations. Theory and Implementation
B. Riviere
List of typos
June 2012

Here is a list of misprints and clarifications. I would like to thank the readers for helping find the typos.

- page 3 line -3: replace j by n.
- page 29 line -10: the variable ϵ is misplaced. The correct formula is:

$$L(v) = \int_\Omega fv + \sum_{e\in\Gamma_{\rm D}} \int_e \Big(\epsilon \mathbf{K}\nabla v\cdot \mathbf{n}_e + \frac{\sigma_e^0}{|e|^{\beta_0}} v\Big) g_{\rm D} + \sum_{e\in\Gamma_{\rm N}} \int_e v g_{\rm N}.$$

- page 31 line 5: clarification. The space $\mathcal{D}(E)$ is the space of $\mathcal{C}^\infty$ functions with compact support in E.
- page 52 line -5: the sign for the first term in the formula for m_e^{21} is wrong. The correct formula is:

$$m_e^{21} = \frac{1}{2}\int_e \mathbf{K}\nabla P_{h,1}\cdot \mathbf{n}_e v_2 + \frac{\epsilon}{2}\int_e \mathbf{K}\nabla v_2\cdot \mathbf{n}_e P_{h,1} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e P_{h,1}v_2.$$

- page 53 line 2: the sign for the first term in the formula for $\mathbf{M}_e^{21}$ is wrong. The correct formula is:

$$(\mathbf{M}_e^{21})_{ij} = \frac{1}{2}\int_e \mathbf{K}\nabla \phi_{j,E_e^1}\cdot\mathbf{n}_e\phi_{i,E_e^2} + \frac{\epsilon}{2}\int_e \mathbf{K}\nabla\phi_{i,E_e^2}\cdot\mathbf{n}_e\phi_{j,E_e^1} - \frac{\sigma_e^0}{|e|^{\beta_0}}\int_e \phi_{j,E_e^1}\phi_{i,E_e^2}.$$

- page 53 line 7: the correct formula for $(\mathbf{b}_e)_i$ is:

$$(\mathbf{b}_e)_i = \int_e \Big(\epsilon\mathbf{K}\nabla\phi_{i,E_e^1}\cdot\mathbf{n}_e + \frac{\sigma_e^0}{|e|^{\beta_0}}\phi_{i,E_e^1}\Big) g_{\rm D}.$$

- page 55 line 4: the correct formula for $\mathbf{M}_e^{21}(i,j)$ is:

$$\mathbf{M}_e^{21}(i,j) = \mathbf{M}_e^{21}(i,j) - \sigma_e^0 w(k)\phi_{i,E_e^2}(s(k))\phi_{j,E_e^1}(s(k))$$

- page 55 line 5 of algorithm 2.3: the correct sentence is: for $i = 1$ to N_{loc} do.
- page 73 line -6: $H_1^0(\Omega)$ should read $H_0^1(\Omega)$.
- page 74 line 10: in the definition of the energy norm, for the second term, $\|v\|^2_{L^2(e)}$ should read $\|[v]\|^2_{L^2(e)}$.
- page 84 line 13: the term $(\boldsymbol{M} + \Delta t \boldsymbol{A})$ should read $(\boldsymbol{M} - \Delta t \boldsymbol{A})$
- page 127 lines -1, -2: the terms $h^k |\boldsymbol{u}|_{H^{k+1}(\Omega)}$ should read $h^{2k} |\boldsymbol{u}|^2_{H^{k+1}(\Omega)}$.
- page 129 line 6: the variable $\mathbf{U}$ should read $\mathbf{U}_h$.
- page 129 last line of Theorem 6.12: the line should read *where* $\delta = 1$ *for SIPG and* $\delta = 0$ *for IIPG and NIPG.*
- page 166 line -4: the third argument for `Aloc` should be removed: `Aloc(idofs,jdofs)`. The same comment holds for the third argument of variables `Bloc11, Bloc22, Bloc12, Bloc21` on pages 170 and 171.