Lecture Notes on Fundamentals of Computational Environmental Physics — Numerical Methods

Peter Bastian
Interdisziplinäres Zentrum für Wissenschaftliches Rechnen
Universität Heidelberg, Im Neuenheimer Feld 368, 69120 Heidelberg
Peter.Bastian@iwr.uni-heidelberg.de

October 14, 2020

Contents

Introduction

The earth and environmental sciences heavily rely on modelling and simulation to make predictions and asses uncertainties. The lecture *Fundamentals of Computational Environmental Physics* (FunCEP) lecture addresses this by threefold approach:

- Modelling part: formulate partial differential equations (PDEs) describing the relevant physical processes.
- Numerical part: formulate numerical methods to solve these partial differential equations on a computer.
- Practical part: Really use the methods discussed in theory to solve practical problems (albeit comparably simple ones).

The aims of the numerical part in particular are the following:

- Learn about different numerical methods to solve the relevant PDEs.
- Become aware of the errors involved in numerical simulations.
- Experience simulation results in practical exercises. Ready-to-run applications will be provided but some C++ programming will be required to do the excercises. We use the DUNE software framework [1].

This course covers a broad range of topics and necessarily cannot provide an in-depth discussion of the material. Here are a few recommendations for books where one can find a more detailed treatment. There are many very good books covering the (numerical) solution of PDEs, so this is just my short and biased list. If you want to read more PDE theory you may consider

- L. C. Evans. *Partial Differential Equations*. American Mathematical Society, 2nd edition, 2010. Evans [2010]
- M. Renardy and R. C. Rogers. *An Introduction to Partial Differential Equations*, volume 13 of *Texts in Applied Mathematics*. Springer, 1993. Renardy and Rogers [1993]
- B. Schweizer. *Partielle Differentialgleichungen*. Springer, 2013. Schweizer [2013]

And here are some numerics books:

- K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, 1996. Eriksson et al. [1996] `http://www.csc.kth.se/~jjan/private/cde.pdf`.

[1] `www.dune-project.org`

- A. Ern and J.-L. Guermond. *Theory and practice of finite element methods*. Springer, 2004. Ern and Guermond [2004]
- H. Elman, D. Silvester, and A. Wathen. *Finite Elements and Fast Iterative Solvers*. Oxford University Press. 2nd edition, 2014. Elman et al. [2014]
- W. Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen*. Teubner, 1986. Hackbusch [1986] `http://www.mis.mpg.de/preprints/ln/lecturenote-2805.pdf`. (in german).
- R. J. Leveque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002. Leveque [2002]

These lecture notes were initially typed in LaTeXby Edith Horstmann in the winter semester 2015/16 for which she deserves a huge thank you. In the winter semester 2017/18 I found the time to work over the notes again.

Heidelberg, October 2017 *Peter Bastian*

Chapter 1.

Introduction and Finite Difference Method

1.1. Partial Differential Equations

Let us consider a simple example of a PDE, the *Poisson equation* which reads

$$-\sum_{i=1}^{n} \frac{\partial^2 u}{\partial x_i^2}(x) = -\sum_{i=1}^{n} \partial_{ii}^2 u(x) = -\Delta u(x) = f(x) \qquad (x \in \Omega). \tag{1.1}$$

This equation is a special case of what we will later consider as the *groundwater flow equation*. Why the minus sign? Well, the quantity $j = -\nabla u$ is the vector field governing the flow of water (or heat in case of heat conduction), that is why we prefer the equation with the minus sign. In the special case $f = 0$ equation (1.1) is also called *Laplace equation*.

As it stands, equation (1.1) raises some questions.

What is the domain?

$u : \Omega \to \mathbb{R}$ is a function mapping from a point x in the domain $\Omega \subseteq \mathbb{R}^n$ to $\mathbb{R}$. Domains are open and connected and may be bounded or unbounded. In our environmental applications the domains will most often be bounded. The integer n is the *dimension*. Besides the standard case $n = 3$ we will often consider $n = 1$ or $n = 2$ in order to facilitate analytical (pencil and paper) solutions or to reduce the computational complexity. Then we have to reason about the justification of the dimension reduction.

What is a solution?

A function u is called a *strong solution* if $u \in C^2(\Omega)$ and for every point $x \in \Omega$ we have the identity $-\Delta u(x) = f(x)$. That is the standard way how we understand equation (1.1) and therefore such u is also called *classical solution*.

This now raises some interesting and important questions:

- Does such a function $u \in C^2(\Omega)$ exist? In general no, it depends on the domain and f.

- If it exists, is the solution unique? No, in general not. We have to give additional conditions on the boundary $\partial\Omega$ of the domain or on the behavior towards infinity in the case of an unbounded domain.
- What kind of conditions are sufficient to make the solution unique? For an unbounded domain $\Omega = \mathbb{R}^n$ we might specify $u \to 0$ for $\|x\| \to 0$. If $\Omega \subset \mathbb{R}^n$ is bounded (it fits in a box) then $u = g$ on $\partial\Omega$ are called *Dirichlet boundary conditions* and $\nabla u \cdot \nu = \frac{\partial u}{\partial \nu} = 0$ on $\partial\Omega$ are called (homogeneous) Neumann boundary conditions. Both are standard boundary conditions for equation (1.1). Here ν is the unit outer normal vector to the domain.

Well-posedness

Existence and uniqueness of solutions is not enough. In addition, stability is needed. Since this is important we state it as a definition.

Definition 1.1 (Well-posedness)**.** A PDE problem (set of PDEs + initial and boundary conditions) is said to be *well-posed* in the sense of *Hadamard* if and only if:

1. it has a solution
2. the solution is unique, and
3. the solutions depends continuously on the data.

The last property means that if you change the data of the PDE, i.e. right hand side, boundary conditions, initial conditions or other parameters a little bit, the solution should also change just a little bit. For our example this means that there should exist a constant C such that the following inequality (called a *stability estimate*) holds:

$$\|u\| \leq C\|f\|.$$

A mathematically precise statement of well-posedness requires the selection of a proper setting in *function spaces* in order for the norms to be defined. In the case of $C^2(\Omega)$, $\|.\|$ corresponds to the supremum norm.

Type classification of PDEs

The general m-th order scalar PDE in n space dimensions for an unknown function $u : \Omega \to \mathbb{R}$, $\Omega \subseteq \mathbb{R}^n$ reads

$$F(x, u(x), \partial_1 u(x), \dots \partial_n u(x), \partial_{11}^2 u(x), \dots, \partial_{nn}^2 u(x), \dots, \partial_{n\dots n}^m u(x)) = 0,$$

for all $x \in \Omega$. Unfortunately:

- There is no general theory to analyse the well-posedness of this problem! This is in contrast to initial value problems of ordinary differential equations!
- There is no general numerical method to solve any such PDE!

In order to resolve this situation the idea is to determine classes of PDEs that can be analysed and solved by the same methods. The standard classification corresponds to *second order order linear* PDEs, which cover a large part of important physical problems (including the Poisson problem). The general linear, scalar partial differential equation with variable coefficients reads

$$-\sum_{i,j=1}^{n} a_{ij}(x)\partial_i\partial_j u(x) + \sum_{i=1}^{n} b_i(x)\partial_i u(x) + c(x)u(x) = f(x), \qquad (x \in \Omega), \quad (1.2)$$

where $(A(x))_{ij} = a_{ij}(x)$ is an $n \times n$ matrix for every point $x \in \Omega$, $b(x)$ is a vector and $c(x), f(x)$ are scalar functions. Without loss of generality one can assume that $A(x) = A^T(x)$ because $\partial_i\partial_j u(x) = \partial_j\partial_i u(x)$. Therefore all eigenvalues of $A(x)$ are real.

Definition 1.2 (Type classification)**.** The type of eq. (1.2) in point $x \in \Omega$ is

- *elliptic*, if all eigenvalues have the same sign and are non-zero,
- *hyperbolic*, if all EVs are non-zero, $n-1$ EVs have the same sign and one EV has the opposite sign,
- *parabolic*, if exactly one EV is zero and the remaining ones have the same sign, and the matrix $[A|b]$ has full rank. This condition ensures that 1.2 is actually a PDE and not an ODE.

If the PDE has the same type for all $x \in \Omega$ (which is in practice typically the case) then the PDE is said to have this type.

Example 1.3. Consider

$$-\left(\partial_{11}^2 u + \partial_{22}^2 u\right) = f, \qquad x \in \mathbb{R}^2.$$

Both eigenvalues are -1, therefore the Poisson equation is elliptic. Next consider

$$\partial_{11}^2 u + \partial_1^1 u = f, \qquad x \in \mathbb{R}^2.$$

Here, one EV is 0, one EV is 1, but this is not a PDE since we only have derivatives w.r.t. one variable. According to the definition above this equation is neither elliptic, hyperbolic or parabolic.

Remark 1.4.

- For $n = 2$ the classification is exhausting for all PDEs. Let $A \in \mathbb{R}^{2\times 2}$ be symmetric, then it has two real eigenvalues λ_1, λ_2. Then we

have three cases

$$\begin{aligned}
&\lambda_1, \lambda_2 > 0 \text{ or } \lambda_1, \lambda_2 < 0 && \to \text{elliptic},\\
&\lambda_1 > 0, \lambda_2 < 0 \text{ or } \lambda_1 < 0, \lambda_2 > 0 && \to \text{hyperbolic},\\
&\lambda_1 = 0, \lambda_2 \neq 0 \text{ or } \lambda_1 \neq 0, \lambda_2 = 0 && \to \text{parabolic}.
\end{aligned}$$

- For $n > 2$ the classification is not exhausting, i.e. there are PDEs which are neither elliptic, hyperbolic or parabolic. But the classification can be extended correspondingly.

1.2. Finite Difference Method

As an example for a simple numerical scheme we consider the *finite difference method* (FDM). We wish to solve

$$-\Delta u(x) = f(x), \qquad x \in \Omega = (0,1)^2 \tag{1.3}$$
$$u(x) = g(x) \qquad x \in \Gamma = \partial\Omega. \tag{1.4}$$

The FDM is based on Taylor expansion which reads with the Lagrangian form of the remainder

$$\begin{aligned}
u(x_1 \pm h, x_2) = u(x_1, x_2) &\pm h\partial_1 u(x_1, x_2) + \frac{h^2}{2}\partial_{11}^2 u(x_1, x_2)\\
&\pm \frac{h^3}{6}\partial_{111}^3 u(x_1, x_2) + \frac{h^4}{24}\partial_{1111}^4 u(x_1 + \delta^{\pm}, x_2).
\end{aligned}$$

Adding the two equations for $\pm h$ yields

$$\begin{aligned}
u(x_1 + h, x_2) + u(x_1 - h, x_2) = 2u(x_1, x_2) &+ h^2\partial_{11}^2 u(x_1, x_2)\\
&+ \frac{h^4}{24}\underbrace{\left(\partial_1^4 u(x_1 + \delta^+, x_2) + \partial_{1111}^4 u(x_1 + \delta^+, x_2)\right)}_{=:-\eta_1(x_1, x_2)}
\end{aligned}$$

from which we get a second order approximation of the second partial derivative:

$$\partial_{11}^2 u(x_1, x_2) = \frac{u(x_1 + h, x_2) - 2u(x_1, x_2) + u(x_1 - h, x_2)}{h^2} + \frac{h^2}{24}\eta_1(x_1, x_2).$$

The same derivation can be carried out in x_2-direction. Combining both directions then yields:

$$\begin{aligned}
-\Delta u(x_1, x_2) = \frac{1}{h^2}[&-u(x_1 - h, x_2) - u(x_1, x_2 - h) + 4u(x)\\
&- u(x_1 + h, x_2) - u(x_1, x_2 + h)] + \frac{h^2}{24}(\eta_1(x_1, x_2) + \eta_2(x_1, x_2)).
\end{aligned} \tag{1.5}$$

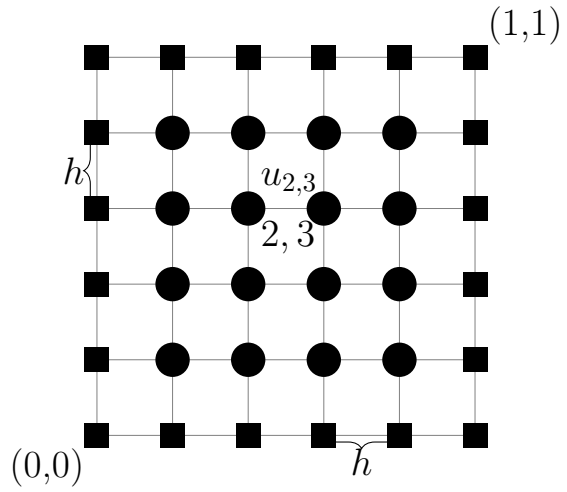


Figure 1.1.: Grid construction for the unit square $\Omega = (0,1)^2$ and $N = 5$. Inner points are given by the circles and boundary points by the squares.

Note that here we have to assume that u is a four times differentiable function!

The idea is now to apply formula (1.5) for a set of points arranged on a grid covering the domain Ω. In the simplest construction each direction is divided into N equal sized intervals resulting in the grid points $x_{i,j} = (ih, jh)$, $0 \le i, j \le N$ and $h = 1/N$. The points with indices $0 < i, j < N$ are called *inner points*, the remaining ones are *boundary points*. This construction is shown by figure 1.1.

By $u_{i,j} \approx u(ih, jh)$ we denote the approximation of $u(ih, jh)$ which is obtained by applying (1.5) to all interior points and ignoring the omitting the unknown error term. For ease of writing we also define $f_{i,j} = f(ih, jh)$ and $g_{i,j} = g(ih, jh)$. The interior grid point give rise to $(N-1)^2$ linear equations of the form

$$\frac{1}{h^2}\left[-u_{i-1,j} - u_{i,j-1} + u_{i,j} - u_{i+1,j} - u_{i,j+1}\right] = f_{i,j}, \qquad 0 < i, j < N.$$

Whenever $i = 0$ or $j = 0$ we replace the corresponding value by the known boundary condition $u_{i,j} = g_{i,j}$. We may arrange these linear equations into matrix form which then read:

$$\frac{1}{h^2}\begin{bmatrix}
4 & -1 & & & -1 & & & & & & & & & & & \\
-1 & 4 & -1 & & & -1 & & & & & & & & & & \\
 & -1 & 4 & -1 & & & -1 & & & & & & & & & \\
 & & -1 & 4 & & & & -1 & & & & & & & & \\
\hline
-1 & & & & 4 & -1 & & & -1 & & & & & & & \\
 & -1 & & & -1 & 4 & -1 & & & -1 & & & & & & \\
 & & -1 & & & -1 & 4 & -1 & & & -1 & & & & & \\
 & & & -1 & & & -1 & 4 & & & & -1 & & & & \\
\hline
 & & & & -1 & & & & 4 & -1 & & & -1 & & & \\
 & & & & & -1 & & & -1 & 4 & -1 & & & -1 & & \\
 & & & & & & -1 & & & -1 & 4 & -1 & & & -1 & \\
 & & & & & & & -1 & & & -1 & 4 & & & & -1 \\
\hline
 & & & & & & & & -1 & & & & 4 & -1 & & \\
 & & & & & & & & & -1 & & & -1 & 4 & -1 & \\
 & & & & & & & & & & -1 & & & -1 & 4 & -1 \\
 & & & & & & & & & & & -1 & & & -1 & 4
\end{bmatrix}
\begin{bmatrix} u_{1,1} \\ u_{2,1} \\ u_{3,1} \\ u_{4,1} \\ \hline u_{1,2} \\ u_{2,2} \\ u_{3,2} \\ u_{4,2} \\ \hline u_{1,3} \\ u_{2,3} \\ u_{3,3} \\ u_{4,3} \\ \hline u_{1,4} \\ u_{2,4} \\ u_{3,4} \\ u_{4,4} \end{bmatrix}
=
\begin{bmatrix} f_{1,1} - g_{1,0} - g_{0,1} \\ f_{2,1} - g_{2,0} \\ f_{3,1} - g_{3,0} \\ f_{4,1} - g_{4,0} - g_{5,1} \\ \hline f_{1,2} - g_{0,2} \\ f_{2,2} \\ f_{3,2} \\ f_{4,2} - g_{5,2} \\ \hline f_{1,3} - g_{0,3} \\ f_{2,3} \\ f_{3,3} \\ f_{4,3} - g_{5,3} \\ \hline f_{1,4} - g_{1,5} - g_{0,4} \\ f_{2,4} - g_{2,5} \\ f_{3,4} - g_{3,5} \\ f_{4,4} - g_{4,5} - g_{5,4} \end{bmatrix}$$

or in short hand notation

$$Az = b.$$

It turns out that A is symmetric, positive definite, and thus has a unique solution for any $N > 1$. Moreover, A is a *sparse* matrix, containing at most $5(N-1)^2$ entries. Still, for fine grids, A becomes very large and the linear system is difficult to solve.

Several questions now arise immediately:

- How large is the *error* $|u(ih, jh) - u_{i,j}|$? One can prove that there exists a constant (depending on the exact solution u) such that

$$\max_{1 \le i,j \le N} |u(ih, jh) - u_{i,j}| \le Ch^2.$$

- A prerequisite for this error bound is that $\partial_1^4 u$, $\partial_2^4 u$ are bounded in the domain Ω. Is this always the case (this is a question about the *regularity* of the solution). What happens if this is not the case?
- How to efficiently solve the potentially large linear system $Az = b$? It turns out this is the main difficulty here. Computing a more accurate numerical solution requires the solution of a larger linear system. This is the reason why solving some earth science problems requires the largest computers in the world.

Chapter 2.

Variational Formulation of Elliptic Boundary Value Problems

2.1. Weak Formulation

Consider the elliptic boundary value problem (e.g. groundwater equation)

$$-\nabla \cdot (K\nabla u) = f \qquad \text{in } \Omega, \tag{2.1}$$

$$u = 0 \qquad \text{on } \Gamma_D \subset \partial\Omega, \tag{2.2}$$

$$-(K\nabla u)\cdot \nu = j \qquad \text{on } \Gamma_N = \partial\Omega \setminus \Gamma_D, \tag{2.3}$$

with $K(x)$ a symmetric, positive definite matrix and $\text{meas}(\Gamma_D) \neq 0$. In case of the groundwater equation, K represents the hydraulic conductivity and the Darcy velocity q is given by $-K\nabla u$.

Equations (2.1) – (2.3) constitute the strong formulation. Strong solutions do exist under rather restrictive assumptions on the smoothness of f, j and the boundary. Mathematicians have devised an alternative definition of a solution that has the advantage to exist under less restrictive assumptions.

Remark 2.1. The boundary conditions on Γ_D are called Dirichlet or essential boundary conditions. The boundary conditions on Γ_N are called Neumann, flux or natural boundary conditions.

In order to derive the weak formulation we multiply equation (2.1) with a function $v \in C^1(\Omega)$ which has homogeneous Dirichlet boundary conditions $v|_{\Gamma_D} = 0$ and use integration by parts:

$$\int_\Omega -\nabla \cdot (K\nabla u) v dx = \int_\Omega f v\, dx$$

$$\iff \int_\Omega (K\nabla u)\cdot \nabla v\, dx - \int_{\partial\Omega} (K\nabla u)\cdot \nu v\, ds = \int_\Omega f v\, dx.$$

Now we seperate the boundary into the two parts $\Gamma = \Gamma_D \cup \Gamma_N$. On Γ_D we have $v = 0$ by assumption and the integral vanishes, on Γ_N we may replace

$-(K\nabla u)\cdot\nu$ by the given boundary condition j (therefore natural boundary condition):

$$\begin{aligned}\int_\Omega (K\nabla u)\cdot\nabla v dx - \underbrace{\int_{\Gamma_D} K\nabla u v\nu ds}_{=0} + \underbrace{\int_{\Gamma_N} jv ds}_{\text{independent of } u} &= \int_\Omega fv dx \\ \iff \underbrace{\int_\Omega (K\nabla u)\nabla v dx}_{=:a(u,v)\text{ "blinear form"}} &= \underbrace{\int_\Omega fv dx - \int_{\Gamma_N} jv ds}_{=:l(v)\text{ "linear form"}}.\end{aligned} \tag{2.4}$$

The left hand side is a so-called *bilinear form* while the right hand side constitutes a so-called *linear form*.

Repetition 2.2. For any vector spaces V, W over $\mathbb{R}$ a function $f : V \to W$ is called linear if it satisfies $f(\alpha v) = \alpha f(v)$ for all $\alpha \in \mathbb{R}$, $v \in V$ and $f(v_1 + v_2) = f(v_1) + f(v_2)$ for all $v_1, v_2 \in V$. A function $a : V \times U \to W$ is called bilinear if it is linear in both arguments. The term form instead of function refers to the fact that $W = \mathbb{R}$ in our application.

From (2.4) we see that $a(u, v) = l(v)$ holds true for the (strong) solution u of 2.1 and any suitable test function $v \in C^1(\Omega)$ with homogenous extension to the boundary. Now the idea is to reverse the argument as follows: Can we find a suitable function space V such that the problem

$$\text{Find } u \in V \text{ such that } a(u, v) = l(v)\ \forall v \in V \tag{2.5}$$

has a unique solution? The answer is yes, as will be detailled below. A solution $u \in V$ of (2.5) is then called a weak solution. If it happens that in addition there holds $u \in C^2(\Omega)$ then the argument given above can be reversed and the *fundamental lemma of calculus of variation* shows that u solves also the original equation (2.1).

In this sense, (2.5) is an alternative formulation of (2.1), the so called *weak formulation* or *variational formulation* (since it can have solutions that are no strong solutions).

Remark 2.3. One may consider a similar situation in the case of linear systems of equations. Suppose $A \in \mathbb{R}^{n\times n}$ is a regular matrix and $b \in \mathbb{R}^n$ a vector, then $\langle Ax, v\rangle = \langle b, v\rangle\ \forall v \in \mathbb{R}^n$ implies $Ax = b$.

Minimization Formulation

There is yet another formulation. The following two assertions are equivalent:

i) u solves $a(u,v) = l(v)\ \forall v \in V$.

ii) u minimizes the functional $J(v) = \frac{1}{2}a(v,v) - l(v)$ (requires a to be symmetric).

Now we can understand (2.5) from a new perspective. Finding a minimum of $J(v)$ boils down to:

$$\text{Set } \phi_v(t) = J(u+vt) \text{ for } t \in \mathbb{R} \text{ and any given function } v.$$
$$\text{Then } \frac{d\phi_v}{dt}(0) = 0 \text{ is a necessary condition for } u \text{ to be a minimum.}$$

And so:

$$\begin{aligned}\phi_v(t) = J(u+vt) &= \frac{1}{2}a(u+vt, u+vt) - l(u+vt) \\ &= \frac{1}{2}a(u,u) + ta(u,v) + \frac{1}{2}t^2(v,v) - l(u) - tl(v) \\ &= J(u) + t[a(u,v) - l(v)] + \frac{1}{2}t^2 a(v,v)\end{aligned}$$

and therefore

$$\frac{d\phi_v}{dt}(0) = a(u,v) - l(v) + ta(v,v)|_{t=0} = a(u,v) - l(v) = 0.$$

Now u, if it exists, is actually a minimum since $a(u,v) > 0$ for any $v \neq 0$. The condition $v|_{\Gamma_D} = 0$ on the test function is explained by the fact that no variation of $u + tv$ is needed on Γ_D since u is already given at these points.

2.2. Function Spaces

The question now is: What is an appropriate function V to use in the weak formulation. The following problem illustrates what can happen when trying to find a minimizer for a functional J defined via an integral.

Suppose we set $V = \{f \in C^0([0,1]) : f(0) = 1, f(1) = 0\}$ and minimize $J(v) = \int_0^1 v^2(x)dx$ over the set V. If we choose for example the sequence of functions $\{\phi_n\}$, $n \in \mathbb{N}$, s.t.

$$\phi_n = \begin{cases} 1 - nx & x \leq 1/n \\ 0 & x > 1/n \end{cases}.$$

It is clear that $J(\phi_n) < J(\phi_m)$ for $n > m$ and $\lim_{n\to\infty} J(\phi_n) = 0$. But the limit of the sequence $\{\phi_n\}$ for $n \to \infty$ is

$$\phi_\infty = \begin{cases} 1 & x = 0 \\ 0 & \text{else} \end{cases}$$

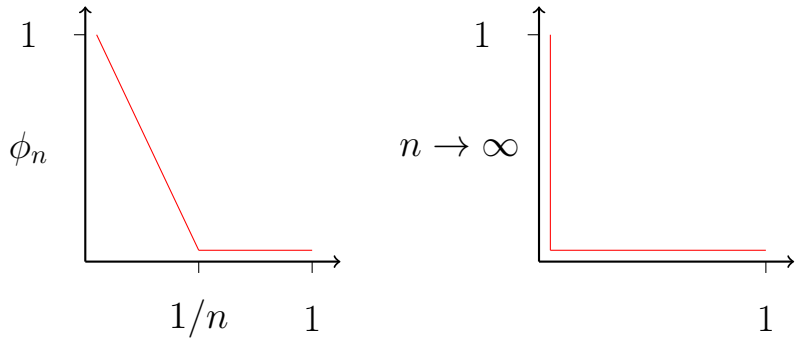


Figure 2.1.: Limit of the sequence $\{\phi_n\}$.

which is *not* a function in V! The problem is that V is not complete w.r.t. to the norm induced by J (note that $\|v\| = \sqrt{J(v)}$ is a norm). The solution is then to choose spaces that are complete with respect to that norm. This can be achieved through a process called *completion* in functional analysis.

The Lebesgue space $L^2(\Omega)$

Using the Lebesgue integral define the following scalar product and norm

$$(u,v)_{0,\Omega} = \int_\Omega u(x)v(x)dx, \qquad \|u\|_{0,\Omega} = \sqrt{(u,u)_{0,\Omega}} = \left(\int_\Omega u(x)^2 dx\right)^{1/2}.$$

Then L^2 is the completion of $C^0(\bar{\Omega})$ w.r.t. the norm $\|.\|_{0,\Omega}$.

Remark 2.4.
- Completion means the limits of all Cauchy sequences of elements are included into the target space, i.e. every $v \in L^2$ is the limit of a Cauchy sequence $(v_k)_{k\in\mathbb{N}} \in C^0(\bar{\Omega})$.
- In the Lebesgue integral $\|u-v\|_{0,\Omega} = \left(\int_\Omega (u-v)^2 dx\right)^{1/2} = 0$ if u and v only differ on sets with *Lebesgue measure* zero, e.g. a line in a two-dimensional domain or a set of countable many points.
- L^2 is a *Hilbert space*, i.e. a *Banach space* endowed with a scalar product. A Banach space is a complete vector space endowed with a norm.

The Sobolev space $H^k(\Omega)$

The next step is to construct subspaces of $L^2(\Omega)$ where elements are differentiable in an appropriate sense. The property of differentiability is transfered to $L^2(\Omega)$ by the introduction of a *weak derivative*.

Definition 2.5. The function $g = \partial_{x_i} f$ is called weak derivative of f iff

$$(g, \phi)_{0,\Omega} = -(f, \partial_{x_i}\phi)_{0,\Omega} \quad \forall \phi \in C_0^\infty(\Omega,)$$

where $C_0^\infty(\Omega)$ is the space of continuous and infinitely often differentiable functions with compact support. Compact support means that the function value is zero for points close enough to the boundary.

Note that the definition of the weak derivative is based on the same argument as the weak formulation. For differentiable functions f we get from integration by parts $-(f, \partial_{x_i}\phi)_{0,\Omega} = (\partial_{x_i} f, \phi)_{0,\Omega}$ where the boundary term vanishes because of the compact support of ϕ.

Now define the scalar product and norm

$$(u, v)_{k,\Omega} = \sum_{0 \le |\alpha| \le k} \int_\Omega (\partial^\alpha u)(\partial^\alpha v) dx, \qquad \|u\|_{k,\Omega} = (u, u)_{k,\Omega}^{1/2}.$$

Then $H^k(\Omega)$ is the completion of $C^k(\bar{\Omega})$ w.r.t. the norm $\|.\|_{k,\Omega}$. Note that formally $H^0(\Omega) = L^2(\Omega)$.

The solution of PDEs requires to prescribe boundary values to functions. However, since the boundary is a set of Lebesgue measure zero no boundary values can be prescribed to L^2-functions. It turns out that for Sobolev spaces $H^k(\Omega)$, $k \ge 1$, one can prescribe boundary values.

$H_0^k(\Omega)$ denotes the completion of $C_0^k(\bar{\Omega})$ (k times differentiable functions with compact support) w.r.t. norm $\|.\|_{k,\Omega}$. Functions in $H_0^k(\Omega)$ are zero at the boundary $\partial\Omega$ almost everywhere. It is also possible to prescribe zero boundary values on part of the boundary. Thus we have the following inclusions:

$$\begin{array}{ccccc} L^2(\Omega) & \supset & H^1(\Omega) & \supset & H^2(\Omega) & \supset \ldots \\ & & \cup & & \cup & \\ & & H_0^1(\Omega) & \supset & H_0^2(\Omega) & \supset \ldots \end{array}$$

2.3. Existence Results

Definition 2.6 (Linear Map)**.** Let V, W be normed spaces. $\mathscr{L}(V; W)$ is the set of all linear and continuous maps from V into W. For any $A \in \mathscr{L}(V; W)$ there exists $C_A \in \mathbb{R}$ s.t.

$$\|Av\|_W \le C_A \|v\|_V \qquad \forall v \in V.$$

Especially, $V' = \mathscr{L}(V; \mathbb{R})$ is called *dual space* of V and $\ell \in \mathscr{L}(V; \mathbb{R})$ is called a *linear functional*.

There are two main results establishing existence of solutions to abstract problems of the form:

$$\text{Find } u \in U: \quad a(u,v) = l(v) \qquad \forall v \in V. \tag{2.6}$$

Note that we may have extended the problem to the situation where *ansatz space* U and *test space* V may be different.

Theorem 2.7 (Banach-Nečas-Babuška)**.** Let U be a Banach space, V a reflexive Banach space, a a continuous bilinear form and $\ell \in V' = \mathscr{L}(\mathbb{R})$. Then 2.6 is well-posed iff

i) $\exists \alpha > 0 : \inf_{u \in U} \sup_{v \in V} \frac{a(u,v)}{\|u\|_U \|v\|_V} \geq \alpha$ *(inf-sup condition).*

ii) $\forall v \in V : (\forall u \in U : a(u,v) = 0) \Rightarrow (v = 0)$.

Some explanation motivation is in order here: For fixed $u \in U$ set $\phi_u(v) := a(u,v)$. Then $\phi_u \in \mathscr{L}(V;\mathbb{R}) = V'$. Thus we can define a linear map $A \in \mathscr{L}(U;V')$ by $Au = \phi_u$. Now 2.6 can be interpreted as

$$\text{Find } u \in U: \quad Au = \ell \ .$$

This linear equation in infinite dimensional function spaces has a unique solution if A is injective and surjective. In the theorem above, i) ensures A is injective and ii) ensures A is surjective.

Theorem 2.8 (Lax-Milgram)**.** Let V be Hilbert space, a a continuous bilinear form, $l \in V'$. Then 2.6 with $U = V$ is well-posed if

$$\exists \alpha > 0 \text{ s.t. } \forall u \in V: \quad a(u,u) \geq \alpha \|u\|_V^2. \tag{2.7}$$

This condition is called *coercivity* of the bilinear form.

Remark 2.9.
- Lax-Milgram requires that $U = V$.
- Coercivity is only a sufficient condition for the existence of a solution (not an equivalent condition as in theorem 2.7).
- Note that a need not be symmetric in both theorems.

Application

Let us recapitulate the connection of this abstract theory to our PDE problem. Consider the second-order elliptic PDE with homogeneous Dirichlet boundary conditions:

$$-\nabla(K\nabla u) = f \text{ on } \Omega \ , \ u = 0 \text{ on } \partial\Omega.$$

It can be shown that the corresponding weak formulation has a unique solution provided

- $\xi^T(K(x)\xi) \geq k_0 > 0 \; \forall x \in \Omega$ *(uniform ellipticity)*, and
- $f \in L^2(\Omega)$ (even less is possible).

To illustrate the power of the weak solution theory we show stability of the weak solution in a one-line-proof:

$$\alpha\|u\|_{1,\Omega}^2 \overset{2.7}{\leq} a(u,u) = l(u) = (f,u)_{0,\Omega} \overset{\text{C-S.}}{\leq} \|f\|_{0,\Omega}\|u\|_{0,\Omega} \leq \|f\|_{0,\Omega}\|u\|_{1,\Omega}.$$

From this we conclude $\|u\|_{1,\Omega} \leq \frac{1}{\alpha}\|f\|_{0,\Omega}$, i.e. u is bounded by the data f. Moreover, the coercivity constant is instrumental in the sensitivity of the norm of the solution with respect to the norm of the source term.

Chapter 3.

Conforming Finite Element Method

The main advantages of the Finite Element Method compared to the Finite Difference method are:

- There is a rigorous convergence theory proving convergence under much less severe assumptions. This does not necessarily mean that the finite difference method is a bad method. It just can only be proven that it is good under more restrictive assumptions.
- Optimal convergence rates are obtained also on unstructured nonuniform grids.
- Full permeability tensors can be treated easily.

3.1. Basic Idea

Here we can discuss only the most basic, so-called *conforming* method. It starts from the usual strong formulation of the elliptic boundary value problem

$$\begin{aligned} -\nabla(K\nabla u) &= f \text{ in } \Omega, \\ u &= g \text{ on } \Gamma_D \subset \partial\Omega, \\ -(K\nabla u)\nu &= j \text{ on } \Gamma_N = \partial\Omega, \backslash \Gamma_D \end{aligned} \tag{3.1}$$

which is then put into the weak formulation

$$\begin{aligned} \text{Find } u \in U: \quad & a(u,v) = l(v) \quad \forall v \in V, \\ & a(u,v) = \int_\Omega (K\nabla u)\nabla v dx,\ l(v) = \int_\Omega fv dx - \int_{\Gamma_N} jv ds, \\ \text{with } V &= \{v \in H^1(\Omega) : v|_{\Gamma_D} = 0 \text{ a.e.}\}, \\ U &= \{u \in H^1(\Omega) : u = u_g + w,\ w \in V, \\ & \qquad u_g \in H^1(\Omega),\ u_g|_{\Gamma_D} = g,\ \text{a.e.}\} = \text{“}u_g + V\text{ ”}. \end{aligned}$$

U is a so-called “affine space”. Note that U is not a vector space as the sum of two elements $u = u_1 + u_2$, $u_1, u_2 \in U$ is not in U. In case of a linear PDE, inhomogeneous Dirichlet boundary conditions can be eliminated beforehand.

Using the property $u = u_g + w$ of any element of U we obtain after insertion in the weak formulation:

$$\begin{aligned} \text{Find } w \in V: \quad & a(u_g + w, v) = l(v) \ \forall v \in V \\ \Leftrightarrow \quad & a(w, v) = l(v) - a(u_g, v) =: \tilde{l}(v), \end{aligned}$$

i.e. the problem is equivalent to:

$$\text{Find } w \in V: \quad a(w, v) = \tilde{l}(v) \ \forall v \in V. \tag{3.2}$$

Now we have the same space on the left side and on the right side and the *Lax-Milgram* theorem can be applied. In the following we therefore just treat the case of homogeneous Dirichlet conditions $g \equiv 0$.

The idea of the conforming Finite Element Method is now to solve (3.2) in a finite dimensional subspace $V_h \subset V$. When V_h is finite with dimension N_h it can be represented by a basis

$$V_h = \text{span}\Phi_h, \quad \Phi_h = \{\varphi_1^h, ..., \varphi_{N_h}^h\}.$$

Inserting the ansatz $V_h \ni u_h = \sum_{j=1}^{N_h} z_j \varphi_j^h$ into the weak formulation we obtain:

$$\begin{aligned} & a(u_h, v) = l(v) && \forall v \in V_h \\ \Leftrightarrow\ & a\left(\sum_{j=1}^{N_h} z_j \varphi_j^h, \varphi_i^h\right) = l(\varphi_i^h) && i = 1, ..., N_h \\ \Leftrightarrow\ & \sum_{j=1}^{N_h} z_j a(\varphi_j^h, \varphi_i^h) = l(\varphi_i^h) && i = 1, ..., N_h \\ \Leftrightarrow\ & Az = b && \text{where } a_{ij} = a(\varphi_j, \varphi_i)\ ,\ b_i = l(\varphi_i). \end{aligned}$$

Here we used first the linearity in the test function and then the linearity of a with respect to its first argument.

Remark 3.1. Some properties of the matrix A follow immediately.

- When $a(u, v)$ is symmetric then A is symmetric.
- When $a(u, v)$ is coercive then A is positive definite:

$$\begin{aligned} y^T A y &= \sum_{i=1}^{N_h} y_i \left(\sum_{j=1}^{N_h} a_{ij} y_j\right) = \sum_{i=1}^{N_h} y_i \left(\sum_{j=1}^{N_h} a(\varphi_j^h, \varphi_i^h) y_j\right) \\ &= \sum_{i=1}^{N_h} y_i a\left(\sum_{j=1}^{N_h} y_j \varphi_j^h, \varphi_i^h\right) = a\left(\sum_{j=1}^{N_h} y_j \varphi_j^h, \underbrace{\sum_{i=1}^{N_h} y_i \varphi_i^h}_{=:v}\right) \\ &= a(v, v) \geq \alpha \|v\|_{\Omega,1}^2 > 0 \text{ when } y \neq 0. \end{aligned}$$

- From $a_{ij} = a(\varphi_j, \varphi_i) = \int_\Omega (K(x)\nabla\varphi_j(x))\nabla\varphi_i(x)dx$ we see that if the supports of φ_i and φ_j do not overlap, the integral vanishes ($\text{supp} f = \{x \in \Omega : f(x) \neq 0\}$). When using basis functions with local support many entries of A can be made zero (A is a *sparse* matrix).

3.2. Conforming Finite Element Spaces

The question now is: How to construct V_h as a finite-dimensional subspace of the Sobolev space $H^1(\Omega)$? The following lemma is helpful in this respect:

Lemma 3.2. Let $k \geq 1$, Ω bounded domain. A piecewise C^∞-function $v : \overline{\Omega} \to \mathbb{R}$ is in $H^k(\Omega)$ iff $v \in C^{k-1}(\overline{\Omega})$.

Proof. See [Braess, 2003, Satz 5.2]. □

As a consequence, for $k = 1$, functions in V_h need to be globally continuous. Locally on each element we choose polynomials (which are C^∞ functions).

The general construction now depends on the space dimension. Let us first consider the one-dimensional case:

- Let $\Omega = (a, b)$.
- Choose a subdivision of Ω into nonoverlapping subintervals called "elements": $\mathcal{T}_h = \{e_1, ..., e_{M_h}\}$, $e_i = (x_{i-1}, x_i)$ with $x_0 = a$, $x_{i-1} < x_i$ and $x_{M_h} = b$.
- Choose a polynomial degree $p \in \mathbb{N}$ and set $V_h^p = \left\{v \in C^0(\bar{\Omega}) : v|_{e_i} \in \mathbb{P}_p\right\}$, where $\mathbb{P}_p$ is the space of polynomials of degree p in one space dimension.

Let us illustrate this construction for $k = 1$, i.e. we consider the space of piecewise linear finite element functions $V_h^1 = \left\{v \in C^0(\bar{\Omega}) : v|_{t_i} \in \mathbb{P}_1\right\}$.

An example of such a function is given in Figure 3.1. Every $v \in V_h^1$ is determined by its values at the positions x_i, $i = 0, \ldots, M_h$ and therefore the dimension of V_h^1 is $M_h + 1$.

The construction is extended to higher polynomial degrees $k \geq 2$ by introducing $k - 1$ additional positions per element where the function value can be prescribed. Therefore the dimension of V_h^k is $M_h + 1 + \min(0, k - 1)M_h$. An example of a piecewise quadratic function is shown in Figure 3.2.

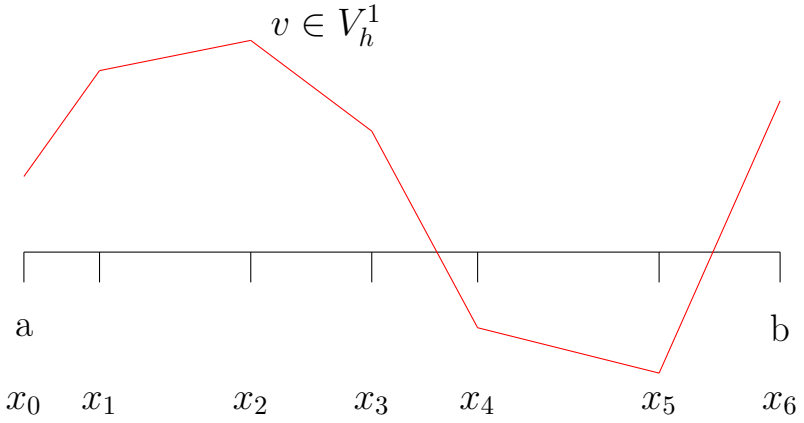


Figure 3.1.: 1D example in which v is taken from the space of piecewise linear functions

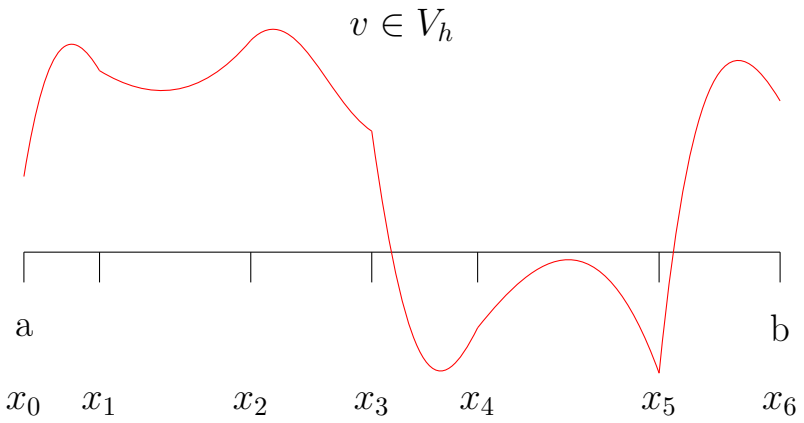
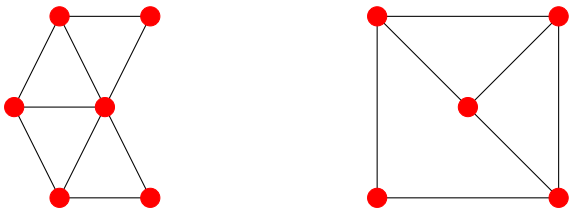


Figure 3.2.: 1D example in which v is taken from the space of piecewise quadratic functions

Remark 3.3.
- Functions in V_h^p are continuous but not differentiable in the classical sense (but in the weak sense).
- The space V_h^p has to be adapted to the boundary conditions. In the case of Dirichlet boundary conditions the value at the boundary is given by the boundary condition for the ansatz space and it is zero for the test space. Consequently the dimension reduces to $M_h - 1 + \min(0, k-1)M_h$.

Mesh Construction in Higher Dimension

The one-dimensional case is special because domains and elements are always intervals. If $n > 1$ domains and elements may have any shape. The requirement of continuity on the finite element functions requires a restriction of this freedom. The following definition of meshes will enable us to define conforming finite functions in higher space dimensions.



conforming nonconforming

Figure 3.3.: Examples of conforming and nonconforming meshes.

Let Ω be a polyhedron (i.e. it is bounded by planar faces). A mesh $\mathcal{T}_h = \{e_1, ..., e_{M_h}\}$ is a set of elements (sometimes called also cells) $e_i \subset \mathbb{R}^d$ which are open and connected and form a nonoverlapping subdivision of Ω:

$$\bigcup_{i=1}^{M_h} \bar{e}_i = \overline{\Omega}, \qquad e_i \cap e_j = \emptyset \ \forall i \neq j.$$

The elements e_i are either *simplices* (triangles, tetrahedra) or *cuboids* (quadrilaterals, hexahedra). By $h_e = \operatorname{diam}(e)$ we denote the diameter of e and by $h = \max_{e \in \mathcal{T}_h} \operatorname{diam}(e)$ the maximum diameter of any element in $\mathcal{T}_h$.

In addition a mesh is called

- *affine* if $e \in \mathcal{T}_h$ is the image of the corresponding reference simplex/cube under an affine linear map $\mu_e : \hat{e} \to e$, $\mu_e(\hat{x}) = B_e \hat{x} + b_e$.
- *conforming* if it is affine and $e_i \cap e_j$ is either empty or a common facet (lower-dimensional polyhedron including a single point) of both elements (see fig. 3.3).
- *uniform* if there exisits a constant κ such that every element $e \in \mathcal{T}_h$ contains a ball of diameter ρ_e with $\rho_e \geq h/\kappa$.
- *quasi-uniform* (or *shape regular*) if there exisits a constant κ such that every element $e \in \mathcal{T}_h$ contains a ball of diameter ρ_e with $\rho_e \geq h_e/\kappa$.

Note that uniformity implies quasi-uniformity as $\rho_e \geq h/\kappa \geq h_e/\kappa$ since $h \geq h_e$.

In order to analyse convergence of the finite element method it is important to think of sequences (or families) of meshes. A mesh family is called *uniform*, respectively *quasi-uniform* if all meshes in the family are uniform, respectively quasi-uniform, with the same constant κ.

Definition 3.4. On affine, conforming meshes one may define conforming finite element spaces as follows:

$$V_h^k = \left\{ v \in C^0(\bar{\Omega}) : \forall e \in \mathcal{T}_h : v|_e = \left(\hat{v}_e \circ \mu_e^{-1}\right) \wedge \hat{v}_e \in \mathbb{V}_p \right\},$$

where $\mathbb{V}_p$ is either

$$\mathbb{P}_p = \left\{ v \in C^\infty(\mathbb{R}^n) : v = \sum_{0 \le \alpha_1 + \ldots + \alpha_n \le p} c_\alpha x_1^{\alpha_1} \ldots x_n^{\alpha_n} \right\} \text{ for simplical meshes}$$

or

$$\mathbb{Q}_p = \left\{ v \in C^\infty(\mathbb{R}^n) : v = \sum_{0 \le \max\limits_{1 \le i \le n} \alpha_i \le p} c_\alpha x_1^{\alpha_1} \ldots x_n^{\alpha_n} \right\} \text{ for cuboid meshes.}$$

Remark 3.5. The transformation to the reference element μ_e^{-1} in the definition above is only necessary for cuboid meshes because when μ_e is affine then $v \in \mathbb{P}_p$ implies that $v \circ \mu_e^{-1} \in \mathbb{P}_p$.

Here are some examples:

- Consider $p = 1, n = 2$ using triangles. Functions are multi-variate polynomials of degree 1 with three degrees of freedom: $v(x) = ax_1 + bx_2 + c$. By fixing the values of the polynomial at the three vertices of one triangle the polynomial is determined uniquely. The values of the polynomial on an edge are defined by the two values ot the vertices of the edge. Therefor the function is continous over an edge.
- Consider $p = 2, n = 2$ using triangles. Here the polynomial has six degrees of freedom. By one additional value per edge the polynomial is determined uniquely. The values on the edge form a polynomial of degree two which is fixed by three values.

3.3. A Glimpse on Convergence Theory for FEM

Lemma 3.6 (Galerkin orthogonality)**.** The error $e = u - u_h \in V$ satisfies

$$a(u - u_h, v) = 0 \qquad \forall v \in V_h.$$

Proof.

$$\begin{aligned} u \in V : a(u, v) &= l(v) \ \forall v \in V, \\ u_h \in V_h : a(u_h, v) &= l(v) \ \forall v \in V_h, \\ \overset{\text{linearity}}{\Rightarrow} a(u - u_h, v) &= 0 \ \forall v \in V_h. \end{aligned}$$

□

Lemma 3.7 (Céa Lemma)**.** The error $u - u_h$ satisfies the following estimate:

$$\|u - u_h\|_V \leq \frac{C}{\alpha} \underbrace{\inf_{w_h \in V_h} \|u - w_h\|_V}_{\text{"approximation property"}} . \tag{3.3}$$

Proof.

$$\begin{aligned}
\alpha\|u - u_h\|_V^2 &\leq a(u - u_h, u - u_h) && \text{(coercivity)} \\
&= a(u - u_h, u - w_h + w_h - u_h) && \text{(arbitrary } w_h \in V_h\text{)} \\
&= a(u - u_h, u - w_h) + \underbrace{a(u - u_h, \underbrace{w_h - u_h}_{\in V_h})}_{=0 \text{ due to Galerkin orthog.}} \\
&\leq C\|u - u_h\|_V \|u - w_h\|_V && (a \in \mathscr{L}(V \times V; \mathbb{R})) \\
\Leftrightarrow \|u - u_h\|_V &\leq \frac{C}{\alpha} \inf_{w_h \in V_h} \|u - w_h\|_V && (w_h \text{ was arbitrary})
\end{aligned}$$

□

Thus the convergence of the finite element method is reduced to the question how well we can approximate the unknown solution u by finite element functions.

Approximation Results

Assume $u \in H^{p+1}(\Omega)$ ("additional regularity"), $p \geq 1$, $n \leq 3$, then $H^{p+1}(\Omega) \subset C^0(\bar{\Omega})$ by the Sobolev embedding theorem and pointwise evaluation is allowed. The *Lagrange interpolation operator*

$$I_h : C^0(\bar{\Omega}) \to V_h^p$$

is well-defined and

$$\|u - I_h u\|_{1,\Omega} \leq C_I h^p \|u\|_{p+1,\Omega}.$$

The constant C_I depends on the shape regularity of the mesh. For a shape-regular family of meshes the method converges with order p in the H^1-norm provided the solution is in H^{p+1}. If we have, for example, $p = 3$ we have convergence with h^3, but u needs to be in $H^4(\Omega)$.

Remark 3.8.
- C_I depends on the quality of the mesh. Shape-regularity is a sufficient condition. A necessary condition is that the largest angle should not tend towards π.
- For $p = 1$, $u \in H^2(\Omega)$ is required and convergence is of first order. But note that $\|.\|_{1,\Omega}$ includes the derivatives.

- For $u \in H^2(\Omega)$ and $p = 1$, one can also show $|u - u_h|_{0,\Omega} \leq Ch^2|u|_{2,\Omega}$.
- $u \in H^2(\Omega)$ requires a convex domain or a C^2-boundary for the Poisson equation. If we have a non-convex domain with a non-smooth boundary and a discontinuous diffusion coefficient K the regularity can be quite low: $u \in H^{1+\epsilon}$, $0 < \epsilon < 1$. The worst case of this is when we have e.g. a checkerboard pattern of two permeabilities, one (k_1) which is high, one (k_2), which is low, see Fig. 3.4. Then $\epsilon = \sqrt{k_2/k_1}$ which can be arbitrarily small.
- An attractive option is to combine h and p refinement.

3.4. Case Studies

In this section we illustrate the convergence behavior of the finite element method, i.e.

$$\|u - u_h\|_{j,\Omega} \to 0 \qquad \text{for } h \to 0$$

for two different test problems. As norms we investigate $j = 0$ (the L^2-norm) and $j = 1$ (the H^1-norm). We will prove in the next chapter that the convergence is of the form

$$\|u - u_h\|_{j,\Omega} \leq Ch^\beta$$

where the exponent β depends on the polynomial degree, the norm in which the error is measured and the regularity of the solution. The exponent β is called the *convergence rate* of the method (with respect to a given norm).

Fully Regular Problem The first test problem uses a solution that is in $H^k(\Omega)$ for any $k \geq 1$.

Example 3.9 (Full Regularity Problem)**.** We consider the Poisson problem in two space dimensions

$$-\Delta u = 0 \qquad \text{in } \Omega = (0, 2)^2$$

k_2	k_1
k_1	k_2

Figure 3.4.: Checkerboard permeability coefficient.

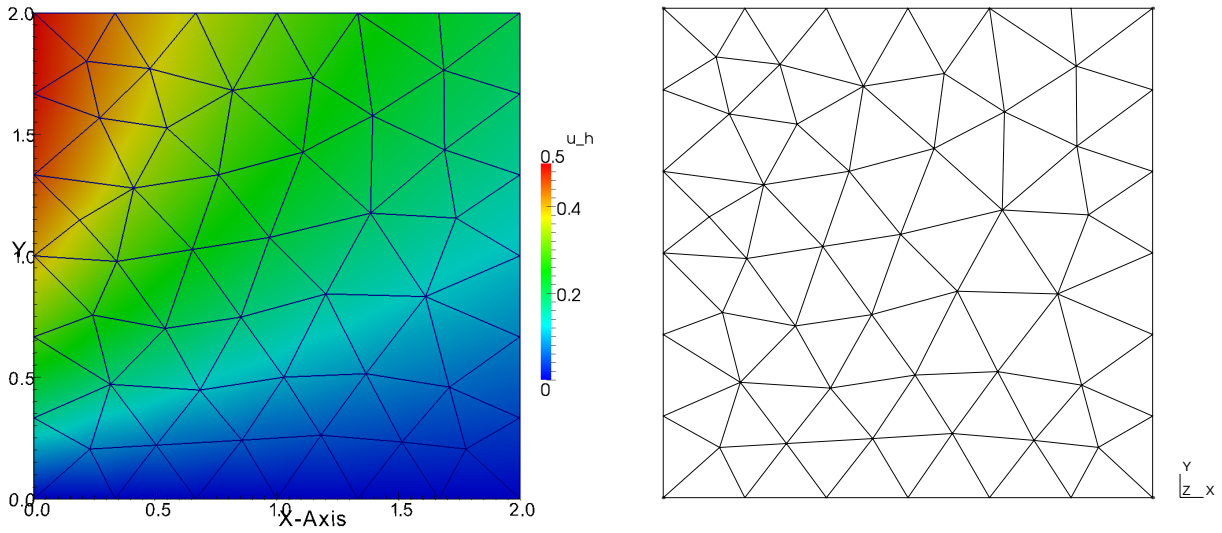


Figure 3.5.: Solution and simplicial coarse mesh for the full regularity example.

with the exact solution

$$u(x,y) = \frac{2y}{(2+x)^2 + y^2}$$

which is taken from [Elman et al., 2014, Example 1.1.3] The exact solution is taken as Dirichlet boundary data. Figure 3.5 shows the coarsest unstructured simplicial mesh generated with `Gmsh`[1], [Geuzaine and Remacle, 2009], used in the computations and the solution as a color plot.

Table 3.1 gives the error in the L^2-norm and the H^1-seminorm for polynomial degree $k = 1, \ldots, 5$ on different meshes. We observe that the convergence rate is $\beta = k + 1$ in L^2 and $\beta = k$ in the H^1-seminorm. We will prove that these convergence factors are optimal.

Figure 3.6 shows plots of the L^2-error $\|u - u_h\|_{0,\Omega}$ in the solution on the coarsest mesh using P_1, P_2, P_3 and P_4. It can be seen that the error has the same overall structure and is just scaled (note the legend!).

Figure 3.7 shows the errors $\|u - u_h\|_{0,\Omega}$ and $\|\nabla(u - u_h)\|_{0,\Omega}$ with respect to (inverse) mesh size h^{-1} for different polynomial degrees $k = 1, \ldots, 5$.

Figure 3.8 compares the errors $\|u - u_h\|_{0,\Omega}$ and $\|\nabla(u - u_h)\|_{0,\Omega}$ with respect to the number of degrees of freedom for P_1, P_2, Q_1 and Q_2. For a given number of degrees of freedom the plot shows that the Q_k solution is slightly more accurate than the P_k solution. □

[1] http://geuz.org/gmsh/

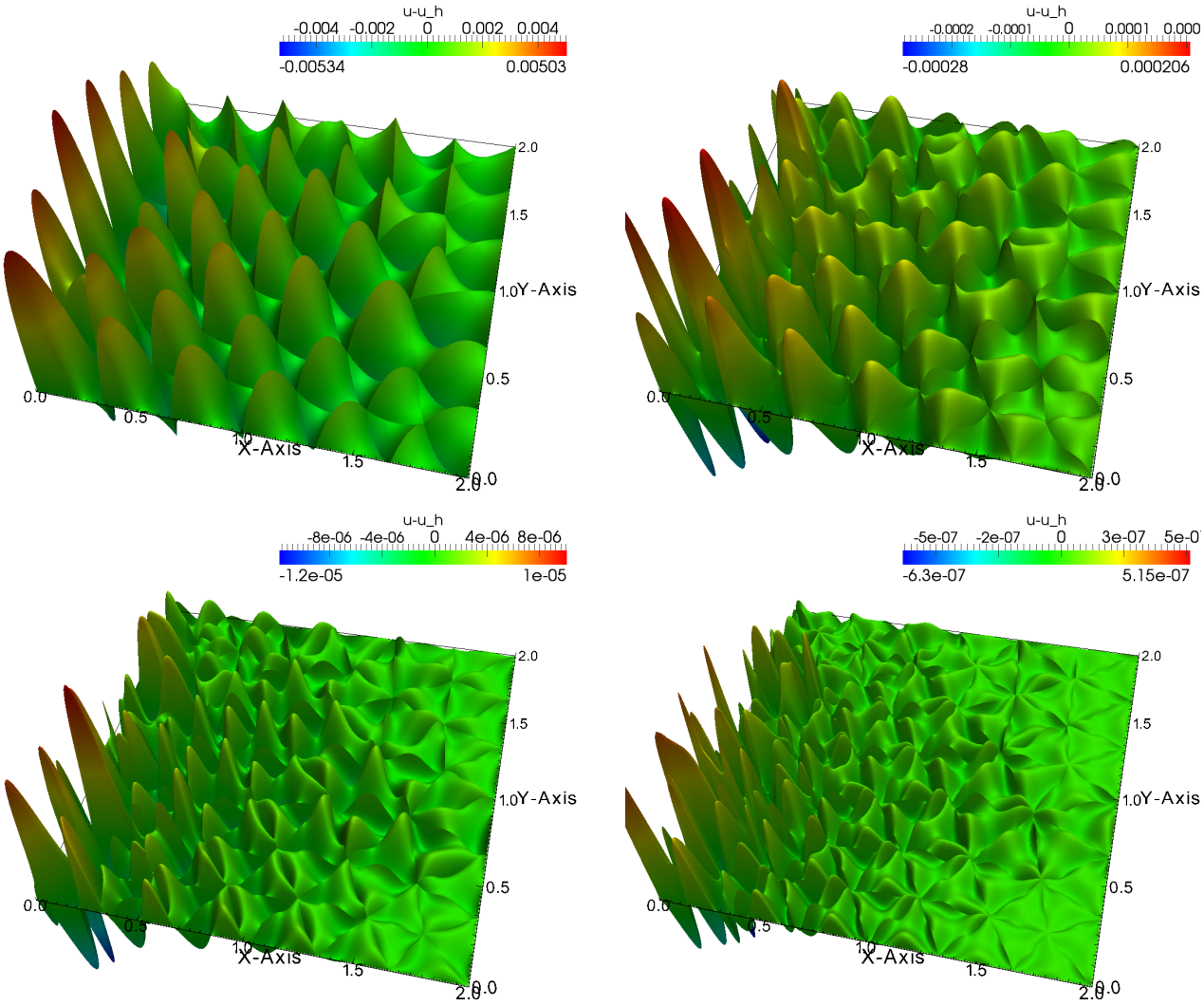


Figure 3.6.: L^2-error in the solution on the coarsest mesh using polynomial degree $k = 1, 2, 3, 4$ (top to bottom , left to right).

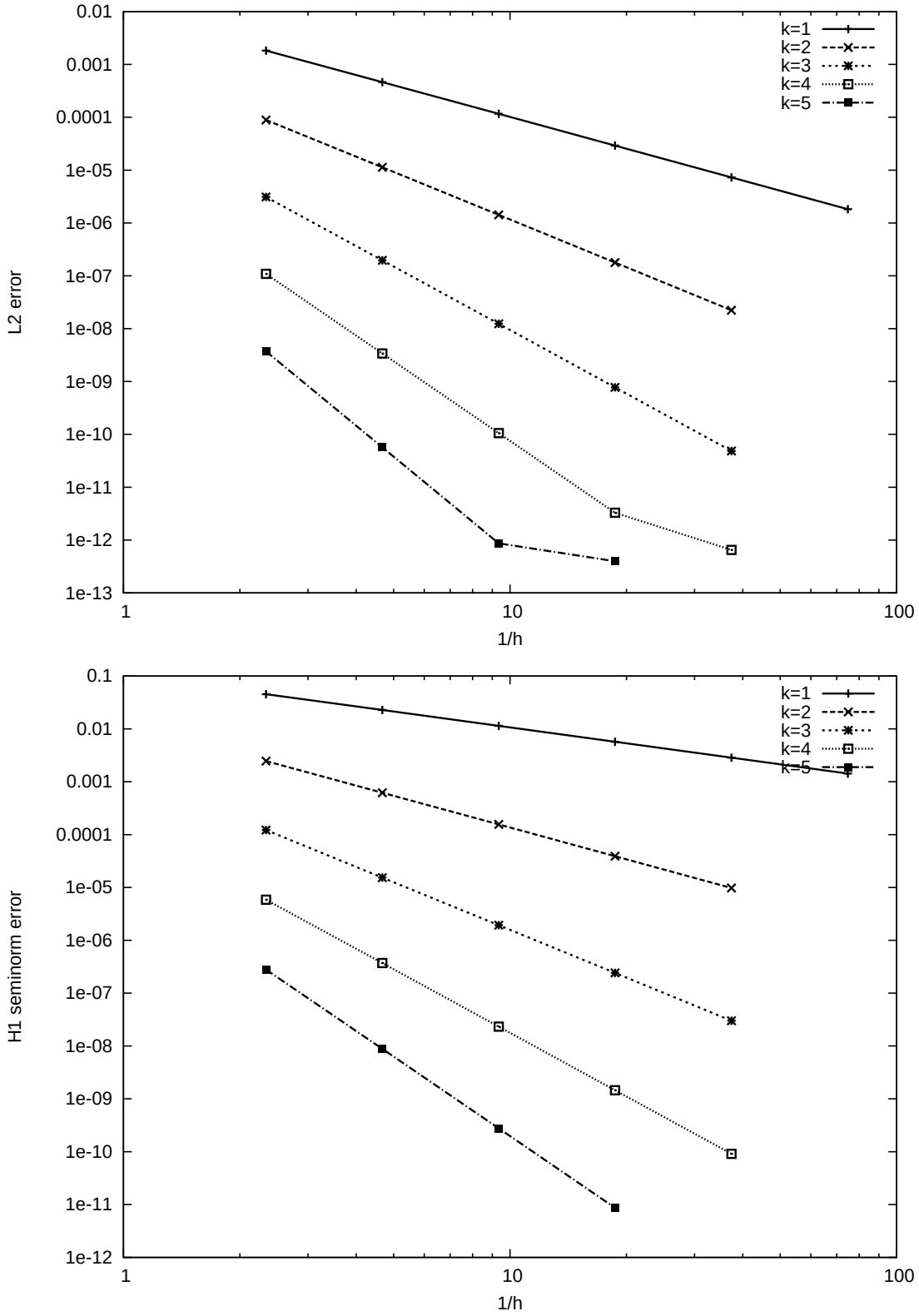


Figure 3.7.: The errors $\|u - u_h\|_{0,\Omega}$ (top) and $\|\nabla(u - u_h)\|_{0,\Omega}$ (bottom) for h- and p-refinement.

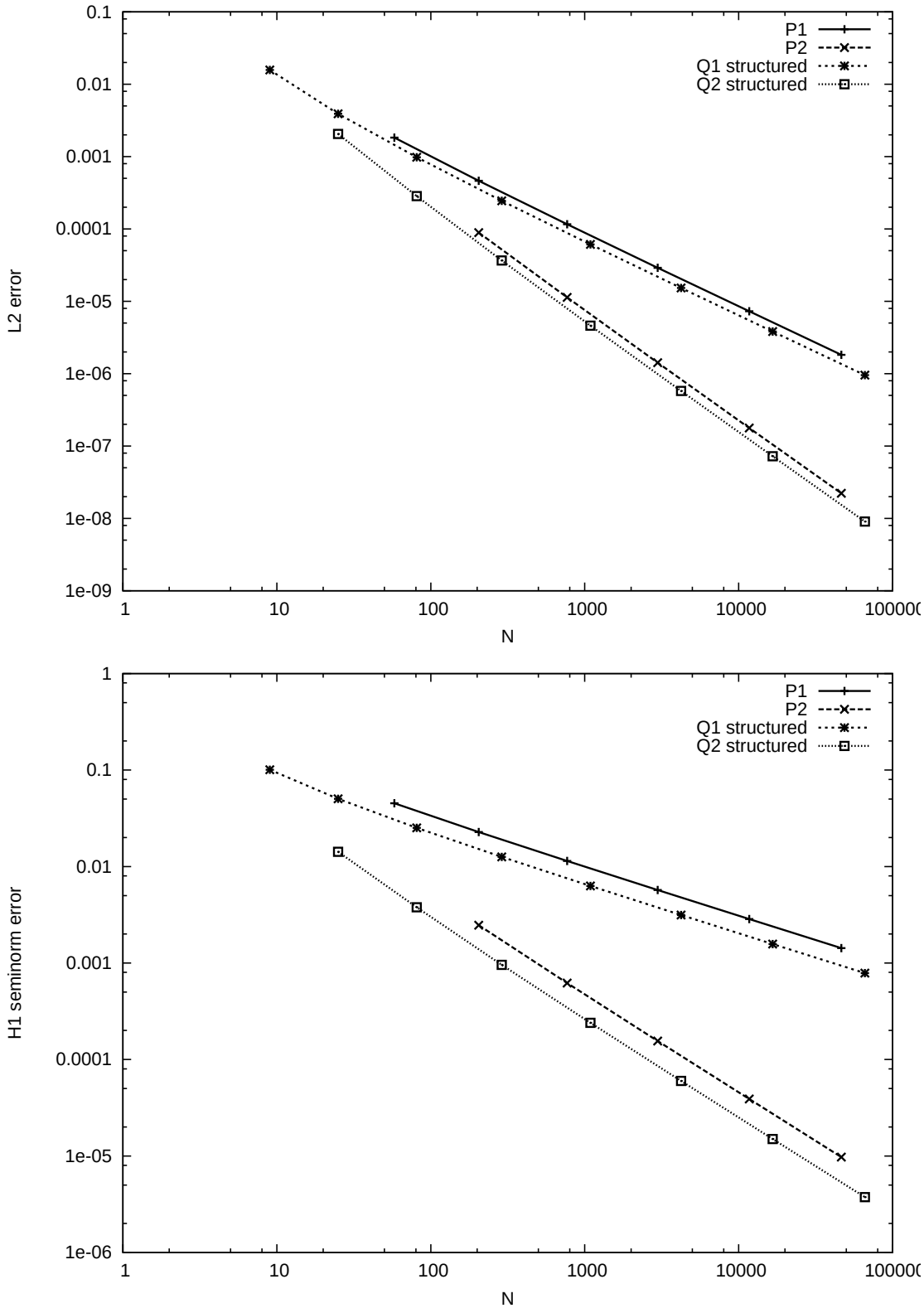


Figure 3.8.: Comparison of $\|u-u_h\|_{0,\Omega}$ (top) and $\|\nabla(u-u_h)\|_{0,\Omega}$ (bottom) using P_1, P_2 and Q_1, Q_2. Note that errors are shown with respect to the number of degrees of freedom.

Table 3.1.: Convergence rates for the example with full regularity using P_k finite elements.

N	$\|u-u_h\|_{0,\Omega}$	L^2-rate	$\|u-u_h\|_{1,\Omega}$	H^1-rate
		$k=1$		
58	1.8313e-03		4.5348e-02	
205	4.6209e-04	1.9866	2.2805e-02	0.99173
769	1.1610e-04	1.9928	1.1423e-02	0.99734
2977	2.9084e-05	1.9970	5.7149e-03	0.99919
11713	7.2765e-06	1.9989	2.8579e-03	0.99976
46465	1.8196e-06	1.9996	1.4290e-03	0.99993
		$k=2$		
205	8.9060e-05		2.4620e-03	
769	1.1318e-05	2.9762	6.2027e-04	1.9889
2977	1.4212e-06	2.9934	1.5556e-04	1.9955
11713	1.7793e-07	2.9978	3.8943e-05	1.9980
46465	2.2255e-08	2.9991	9.7419e-06	1.9991
		$k=3$		
442	3.1084e-06		1.2183e-04	
1693	1.9638e-07	3.9845	1.5378e-05	2.9859
6625	1.2346e-08	3.9915	1.9268e-06	2.9967
26209	7.7398e-10	3.9956	2.4097e-07	2.9993
104257	4.8446e-11	3.9979	3.0124e-08	2.9999
		$k=4$		
769	1.0864e-07		5.8722e-06	
2977	3.3959e-09	4.9996	3.7145e-07	3.9826
11713	1.0548e-10	5.0087	2.3283e-08	3.9958
46465	3.2843e-12	5.0053	1.4561e-09	3.9991
185089	6.4845e-13	2.3405	9.1024e-11	3.9997
		$k=5$		
1186	3.7143e-09		2.7910e-07	
4621	5.6760e-11	6.0321	8.8259e-09	4.9829
18241	8.6560e-13	6.0350	2.7575e-10	5.0003
72481	3.9778e-13	1.1217	8.6568e-12	4.9934

Reentrant Corner Problem In this subsection we consider a problem where the solution is less regular.

Example 3.10 (Reentrant Corner Problem)**.** We consider the Poisson problem in two space dimensions

$$-\Delta u = 0 \qquad \text{in } \Omega = (-1,1)^2 \setminus [0,1) \times (-1,0]$$

with the exact solution in polar coordinates

$$u(r,\theta) = r^{\frac{2}{3}} \sin\left(\frac{2}{3}\theta\right).$$

The exact solution is taken as Dirichlet boundary data. Figure 3.9 shows a color plot of the solution with contour lines and Figure 3.10 shows the two initial meshes generated with `Gmsh` used in the computations.

The exact solution has the regularity $u \in H^{1+\frac{2}{3}}$, see [Hackbusch, 1986, Example 9.7.2]. Table 3.2 shows that the convergence rate in H^1 is $1 + 2/3 - 1 = 2/3$ and in L^2 it is $2(1 + 2/3 - 1) = 4/3$.

Figure 3.11 shows the L^2-error on a fixed mesh with varying polynomial degree. It illustrates that the error is concentrated near the reentrant corner. Moreover, the error is reduced at a greater rate away from the corner with increasing polynomial degree. This suggests that the local convergence depends on the local regularity of the problem.

Figure 3.12 shows the error in L^2 and H^1 norms plotted against the mesh size for various polynomial degrees. Clearly, the convergence rate is independent of the polynomial degree. Here the uniform initial mesh has been used.

Figure 3.13 shows the error in L^2 and H^1 norms now plotted against the number of degrees of freedom for various polynomial degrees using the uniform mesh and the locally refined mesh. Theses results show that with higher polynomial degree the solution is still more accurate for the same number of degrees of freedom. The second observation is that the locally refined mesh is much more efficient in terms of error per degrees of freedom. Together with the observation from 3.11 this suggests that the most efficient method would be a small mesh size in the vicinity of the singularity and a high polynomial degree away from it. The goal of *hp*-methods is to automatically choose the appropriate mesh size and polynomial degree to reach a prescribed error tolerance with the minimum number of degrees of freedom. □

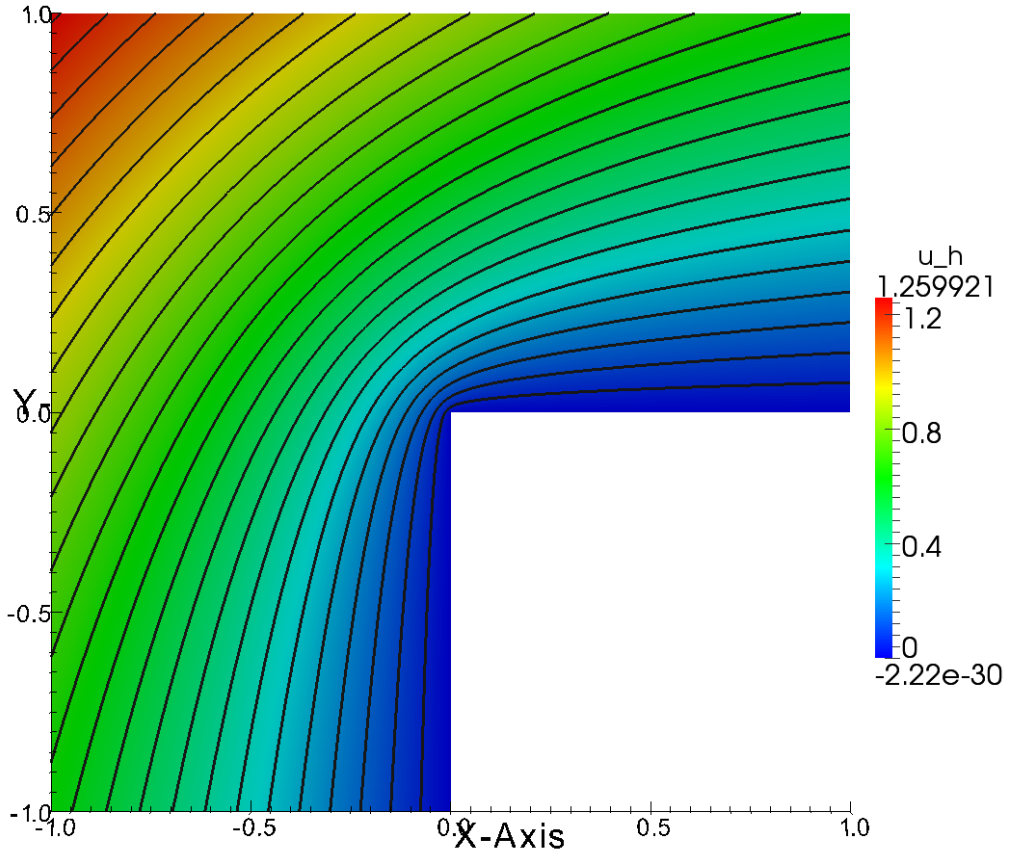


Figure 3.9.: Solution of the L-domain example.

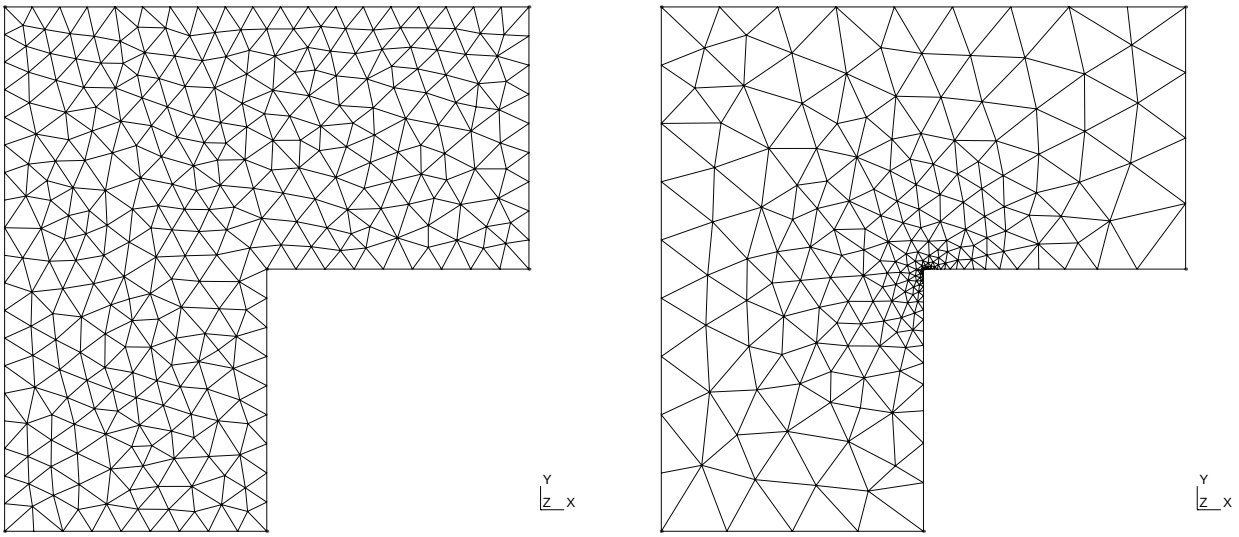


Figure 3.10.: Uniform mesh of the L-shaped domain and mesh with local refinement towards the reentrant corner.

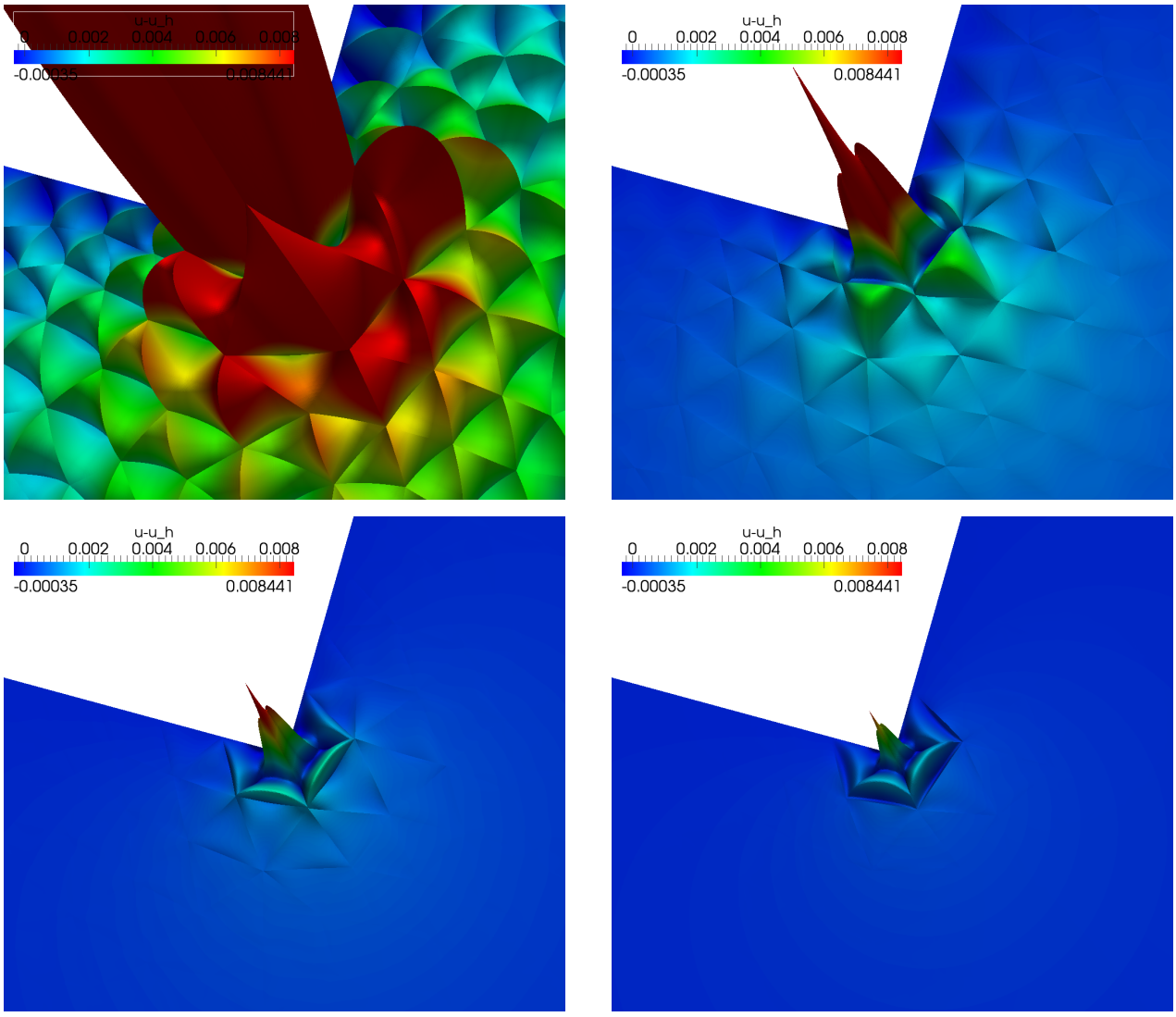


Figure 3.11.: L^2-error in the solution on the coarsest mesh using polynomial degree $k = 1, 2, 3, 4$ (top to bottom , left to right). Note that scaling is the same in all plots!

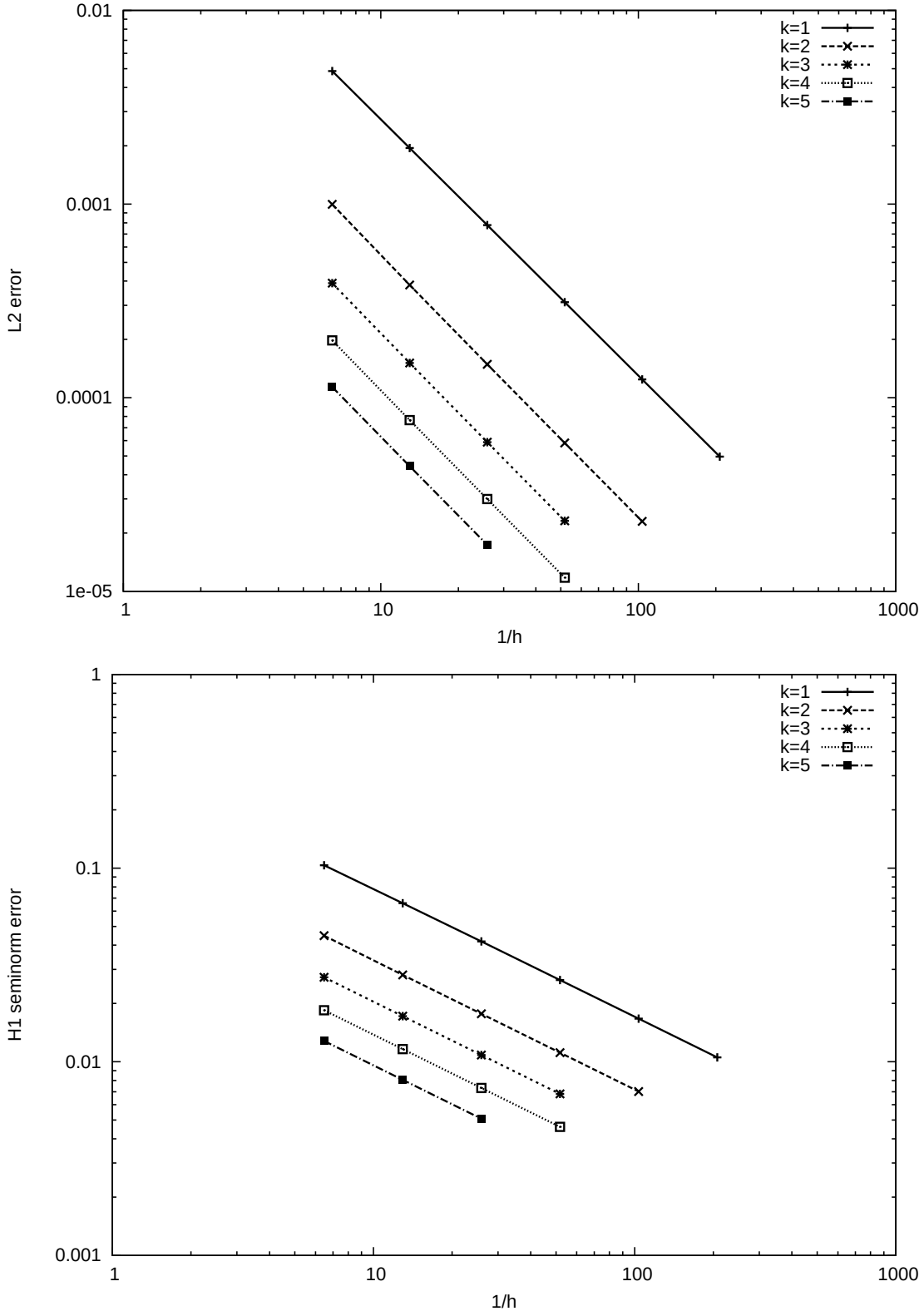


Figure 3.12.: Comparison of $\|u - u_h\|_{0,\Omega}$ (top) and $\|\nabla(u - u_h)\|_{0,\Omega}$ (bottom).

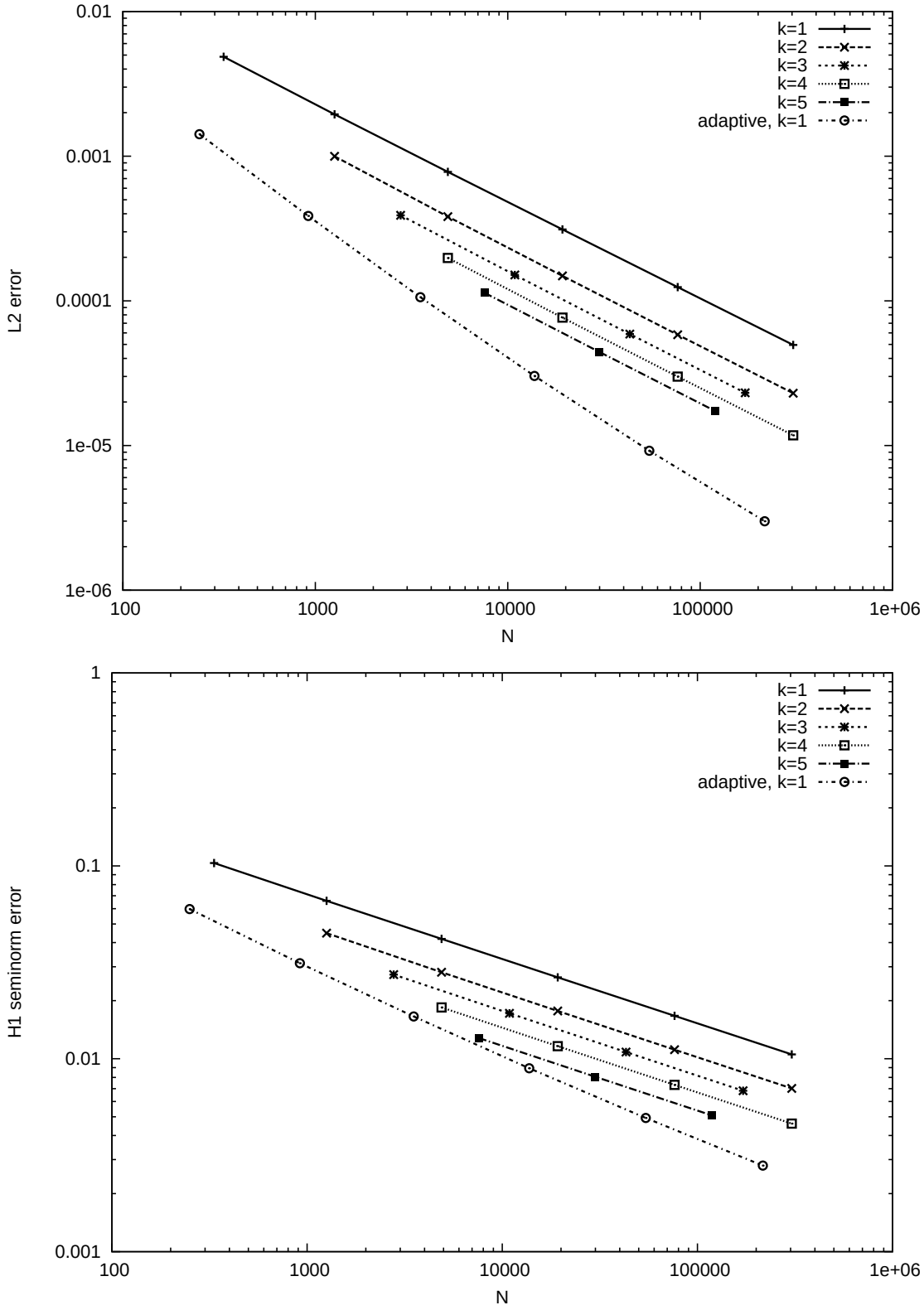


Figure 3.13.: Errors $\|u-u_h\|_{0,\Omega}$ (top) and $\|\nabla(u-u_h)\|_{0,\Omega}$ (bottom) with respect to degrees of freedom.

Table 3.2.: Convergence rates for the L-domain example.

N	$\|u-u_h\|_{0,\Omega}$	L^2-rate	$\|u-u_h\|_{1,\Omega}$	H^1-rate
$k=1$, uniform mesh				
334	4.8614e-03		1.0345e-01	
1259	1.9469e-03	1.3202	6.5916e-02	0.65024
4885	7.7909e-04	1.3213	4.1791e-02	0.65743
19241	3.1151e-04	1.3225	2.6431e-02	0.66099
76369	1.2438e-04	1.3245	1.6692e-02	0.66307
304289	4.9596e-05	1.3265	1.0532e-02	0.66439
$k=1$, adapted mesh				
250	1.4174e-03		5.9661e-02	
919	3.8599e-04	1.8766	3.1300e-02	0.93063
3517	1.0591e-04	1.8657	1.6561e-02	0.91836
13753	3.0263e-05	1.8072	8.9265e-03	0.89164
54385	9.1965e-06	1.7184	4.9287e-03	0.85688
216289	2.9978e-06	1.6172	2.7940e-03	0.81891

3.5. A Note on Representing Wells

Point Sources and Sinks

are important mathematical idealizations when the dimensions of the system under consideration are much larger than the source or sink. Assume we are in dimension $n = 3$. In electrostatics the basic experiment considers a point charge and Coulomb's law predicts a force

$$F(x) = -\frac{q}{4\pi\epsilon}\frac{x}{\|x\|^3},$$

acting on a point-like test charge of unit strength located at point x attracted by a point-like charge of strengh q and opposite sign located at the origin. Here, $\|x\| = \sqrt{\sum_{i=1}^{3} x_i^2}$ is the Euclidean norm, ϵ is the electric field constant and 4π is the surface of the unit sphere.

In gravity the basic experiment considers a point mass and Newton's law predicts a force

$$F(x) = -Gm\frac{x}{\|x\|^3},$$

acting on a point-like test body with unit mass located at x attracted by a point-like body of mass m located at the origin. Here G is the gravitational constant.

Obviously, the form of the force law is the same. It should be noted that the force is the same when the bodies are homogeneous spheres.

Now we transfer this to the domain of fluid flow. Assume the $v(x)$ is a given stationary velocity field of a fluid with constant mass density ϱ. Then

$$M_\omega = \int_{\partial\omega} \varrho v \cdot \nu \, ds$$

is the mass flow rate (mass per unit time) flowing out of the domain ω. Here, $\nu(x)$ is the unit outer normal vector to a point x on the surface of the domain ω and M_ω is positive if there is net mass flowing out of ω. For example, $\omega = B(r,0) = \{x \in \mathbb{R}^n : \|x\| < r\}$ may be the ball of radius r around the origin. Now consider the special velocity field

$$\gamma_n(x) = \begin{cases} \frac{Q}{2\pi}\frac{x}{\|x\|^2} & n = 2, \\ \frac{Q}{4\pi}\frac{x}{\|x\|^3} & n = 3, \end{cases} \tag{3.4}$$

and observe for $n = 2$

$$\int_{\partial B(r,0)} \gamma_2(x) \cdot \nu(x) \, ds = \int_{\partial B(r,0)} \frac{Q}{2\pi}\frac{x}{\|x\|^2} \cdot \frac{x}{\|x\|} \, ds = \frac{Q}{2\pi r} 2\pi r = Q$$

independent of the radius $r > 0$. In the same way we get for $n = 3$:

$$\int_{\partial B(r,0)} \gamma_3(x) \cdot \nu(x) \, ds = \int_{\partial B(r,0)} \frac{Q}{4\pi}\frac{x}{\|x\|^3} \cdot \frac{x}{\|x\|} \, ds = \frac{Q}{4\pi r^2} 4\pi r^2 = Q.$$

Thus we may consider $\gamma_n(x)$ as the velocity field corresponding to a point source with (volumetric) flow rate Q. A generalization to $n > 3$ dimensions is possible. Now observe that the *singularity function*

$$s_n(x) = \begin{cases} -\frac{Q}{2\pi}\ln\|x\| & n = 2, \\ \frac{Q}{4\pi}\frac{1}{\|x\|} & n = 3, \end{cases} \tag{3.5}$$

satisfies the relation

$$\gamma_n(x) = -\nabla s_n(x)$$

and furthermore one may verify that

$$\nabla \cdot \gamma_n(x) = -\Delta s_n(x) = 0 \qquad \forall x \neq 0.$$

We may therefore interpret the singularity function as solution of the Poisson equation in $\mathbb{R}^n \setminus \{0\}$ with a point source of strength Q. For $Q = 1$ we obtain formally

$$1 = \int_{\partial B(r,0)} \gamma_n(x) \cdot \nu(x)\, ds = - \int_{\partial B(r,0)} \nabla s_n(x) \cdot \nu(x)\, ds = - \int_{B(r,0)} \Delta s_n(x)\, dx$$

if Gauß' theorem would be valid. In that sense $-\Delta s_n$ acts "like a delta function" positioned at the origin and we write it as

$$-\Delta s_n(x) = Q\delta_0.$$

Note however that this is not a mathematically rigorous statement! These results can be extended to point sources at arbitrary locations y by translation: $s_n(x, y) := s_n(x - y)$. Sinks are modelled by negative flow rates Q.

When assembling the right hand side vector in the finite element method we use the property of the delta function

$$b_i = \int_\Omega Q\delta_y(x)\varphi_i(x)dx = Q\varphi_i(y).$$

If the position of the source coincides with a mesh vertex $y = x_j$ we get in particular

$$b_i = \begin{cases} Q & i = j, \\ 0 & \text{else.} \end{cases}$$

But note that $s_n(x - y) \notin H^1(\Omega)$ and it is therefore not covered by our convergence theory for the finite element method.

Resolved Wells

Another approach to well modelling is to assume the source/sink term to be nonzero and constant only in a region $\Omega_W \subset \Omega$. We call it a resolved well, if the subdomain Ω_W is resolved by the mesh, i.e. there is $\mathcal{T}_h^W \subset \mathcal{T}_h$ such that

$$\bigcup_{e \in \mathcal{T}_h^W} \bar{e} = \overline{\Omega}_W \qquad e \cap e' = \emptyset \;\; \forall e, e' \in \mathcal{T}_h^W.$$

Now the source term has the form

$$f(x) = \begin{cases} Q & x \in \Omega_W \\ 0 & \text{else} \end{cases}$$

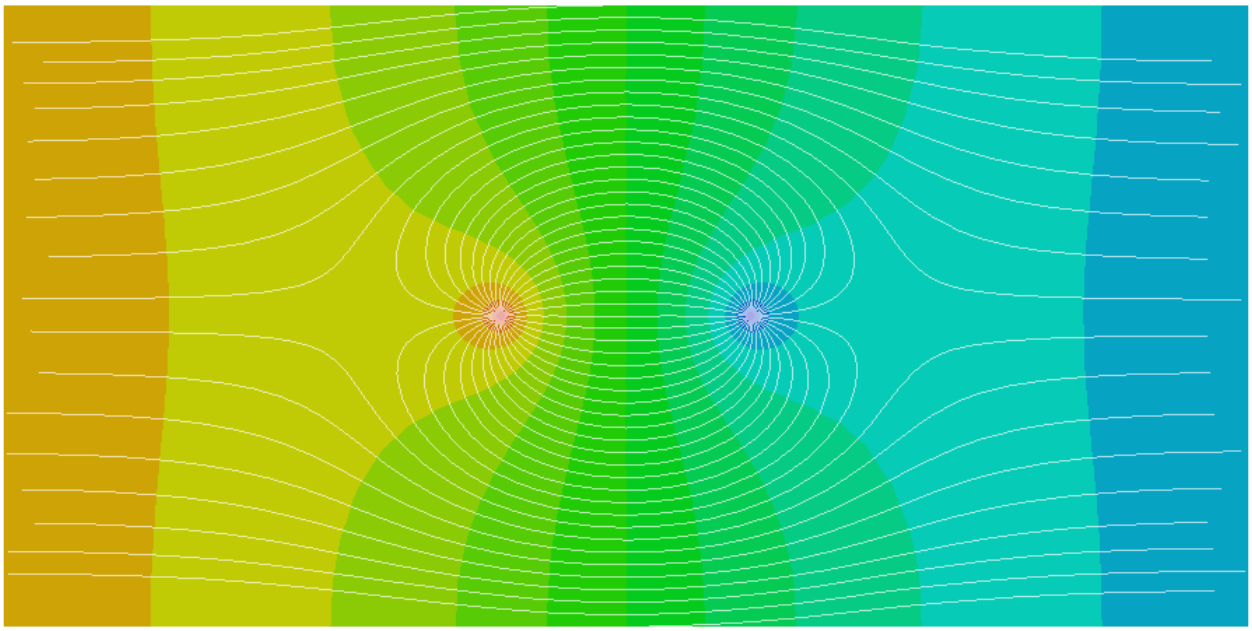


Figure 3.14.: Color plot of the potential and streamlines of the velocity field for the problem stated in Example 3.11.

and can be integrated exactly. But: if $|\Omega_W| \ll |\Omega|$ the mesh needs to be very fine. This can be alleviated by local mesh refinement but this may still prohibitive in three space dimensions.

In addition, for line sources in a 2d vertical model or a full 3d model with heterogeneous permeability, the value of $f(x)$ is in general not known. In high permeability regions there will be more flow in/out of the well as in low permeability regions. Then one may set a very high permeability in the well in the vertical direction and place a point source (or a Neumann boundary condition) within the well.

Numerical Experiment

Example 3.11 (FunCEP exercise 1)**.** In the sense given above

$$u(x) = (a,\ 0)^T \cdot x + u_0 - \frac{1}{2\pi} \ln \|x - x_{source}\|_2 + \frac{1}{2\pi} \ln \|x - x_{sink}\|_2,$$

solves the problem

$$-\Delta u = \delta_{x_{source}} - \delta_{x_{sink}} \quad \text{in } \mathbb{R}^2, \qquad \lim_{\|x\|_2 \to \infty} -\nabla u = (a,\ 0)^T.$$

Figure 3.14 illustrates the solution of this problem. In ground water flow this is called a dublette-flow. All the water from the source will enter the sink and there is a clear separation of this flow from the background flow.

Table 3.3.: Pointwise relative errors $\frac{|u(x)-u_h(x)|}{|u(x)|}$ for various positions and meshes and an approximation of the delta sources by constant functions with support 0.1×0.1. The first two meshes are unstructured with refinement near the delta sources and the last mesh is a structured uniform mesh.

	H=0.1			
DOF	(39.999,25)	(39.9,25)	(35,15)	(20,15)
38533	3.18E-1	1.05E-3	3.04E-6	5.05E-7
153829	3.17E-1	6.68E-4	1.74E-5	1.14E-6
614713	3.17E-1	4.82E-4	5.15E-6	8.52E-7
501501	2.98E-1	2.25E-3	8.26E-7	1.87E-7

Table 3.3 lists pointwise relative errors for the finite element solution. Here the delta sources are approximated by resolved wells of size 0.1×0.1 with constant flow rate. Within the source regions no convergence is observed. At the boundary of the source region convergence is already close to linear (the first three lines of the table were obtained from one mesh and two refinements). Figure 3.15 shows a 3D plot of the relative pointwise error near the source. Observe that the error is highly nonuniform and varies on the element. Therefore it may strongly depend on the exact location.

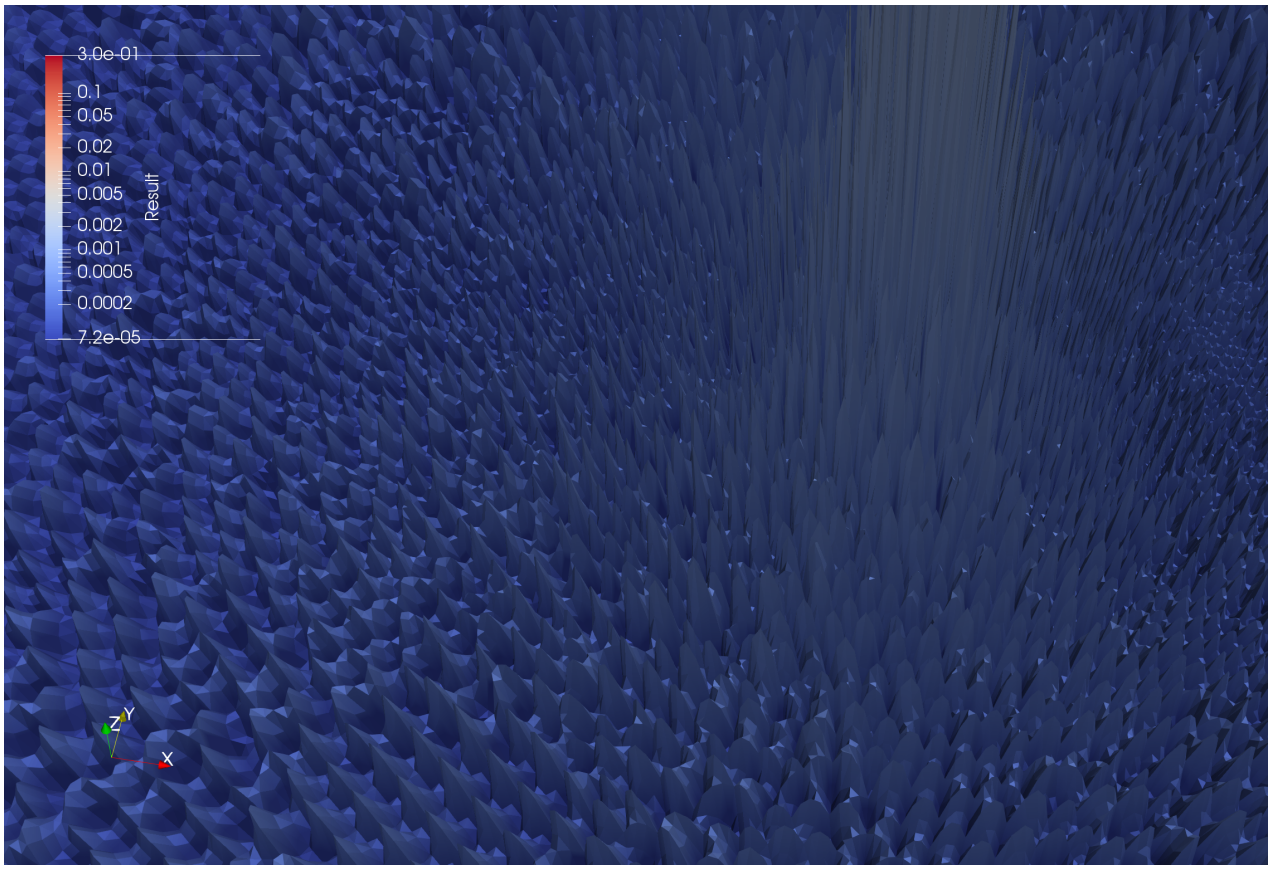


Figure 3.15.: Relative pointwise error in a 3D plot.

Chapter 4.

Advanced FEM: Error Control and Adaptivity

Introduction

Doing simulations may become quite costly in terms of computing time. Without assessing the error in a numerically computed solution we may not be sure whether our result is accurate enough. On the other hand we might "over-resolve" and compute a much too accurate solution. Thus we want to reach the following goals:

1) *Error control:* Use sequence of meshes $\mathcal{T}_{h_1}, \mathcal{T}_{h_2}, \ldots$ and stop the computation when
$$\|u - u_{h_i}\| \leq TOL.$$
Questions: Which norm should be used? What to do if the error is non-uniform and highly local? How to choose TOL?

2) *Adaptivity:* Construct the sequence of meshes $\mathcal{T}_{h_1}, \mathcal{T}_{h_2}, \ldots$ in such a way that $\|u - u_{h_i}\| \leq TOL$ is achieved with minimal effort in computation time (or minimal number of elements).

We might think to achieve error control by using the *a-priori error estimates* from the previous chapter. There, we mentioned

$$\|u - u_h\|_{1,\Omega} \leq C h^{\min(p,r)} \|u\|_{r+1,\Omega}$$

where the constant C depends on the mesh and the coefficients of the PDE, r is the regularity of the problem which is usually difficult to access, and u is the true solution, which is unknown. Therefore, a-priori error bounds are not helpful to assess the error with reasonable accuracy. (This does not mean they are bad, but they essentially only provide the asymptotic rate of convergence as $h \to 0$).

In this chapter we provide so-called *a-posteriori error estimates* which have the form

$$\|u - u_h\| \leq C \left[\sum_{e \in \mathcal{T}_h} \eta_e^2(u_h, f, j, K, \ldots) \right]^{1/2}$$

where η_t is easily computable as it only depends on the data of the PDE and the computed finite element solution. The method discussed here contains a constant C that can be computed with some effort but usually is not computed. There are, however, more elaborate methods which do not contain unknown constants and are thus called "fully computable".

4.1. Residual-based A Posteriori Error Estimation

We recall the weak formulation of the continuous problem:

$$\text{Find } u \in V \subset H^1(\Omega): \quad a(u,v) = l(v) \quad \forall v \in V$$

with

$$a(u,v) = \int_\Omega (K\nabla u)\cdot\nabla v dx, \quad l(v) = \int_\Omega fv dx - \int_{\Gamma_N} jv ds.$$

The error $e_h := u - u_h$ is now the object of interest. For e and any unction $v \in V$ (note, v is just in V and not necessarily in the finite element space V_h!) we can evaluate:

$$a(e_h,v) = a(u-u_h,v) = a(u,v) - a(u_h,v) = \underbrace{l(v) - a(u_h,v)}_{\text{"}residual\text{"}}.$$

Now using element-wise integration by parts we obtain

$$\begin{aligned}
a(e_h,v) &= l(v) - a(u_h,v) \\
&= \sum_{e\in\mathcal{T}_h}\left[\int_e fv\,dx - \int_{\partial e\cap\Gamma_N} jv\,ds - \int_e (K\nabla u_h)\cdot\nabla v\,dx\right] \\
&\overset{\text{int. by parts}}{=} \sum_{e\in\mathcal{T}_h}\Bigg[\int_e \underbrace{(f+\nabla\cdot(K\nabla u_h))}_{=:r} v\,dx \\
&\qquad + \int_{\partial e\cap\Gamma_N} \underbrace{(-(K\nabla u_h)\cdot\nu - j)}_{R_b} v\,ds \\
&\qquad - \int_{\partial e\cap\Omega} (K\nabla u_h)\cdot\nu v\,ds \qquad \underbrace{+0}_{\text{Dirichlet: } v=0 \text{ on } \Gamma_D}\Bigg] \\
&= \sum_{e\in\mathcal{T}_h}\int_e rv\,dx + \sum_{\gamma\in\mathcal{F}_h^N}\int_\gamma R_b v\,ds + \sum_{\gamma\in\mathcal{F}_h^i}\int_\gamma [\![-(K\nabla u_h)\cdot\nu_\gamma]\!] v\,ds.
\end{aligned} \tag{4.1}$$

where $r = f + \nabla(K\nabla u_h)$ is the volume residual, $R_b = -(K\nabla u_h)\cdot\nu - j$ is the boundary residual, $\mathcal{F}_h^N$ are the edges/faces on the Neumann boundary Γ_N, $\mathcal{F}_i^N$ are the interior edges/faces of the mesh and ν_γ is the chosen normal direction for $\gamma \in \mathcal{F}_h^i$. For every interior edge/face $\gamma \in \mathcal{F}_h^i$, we get two contributions (from the cells e and e' adjacent to γ) with opposite normal direction. The two contributions can be combined by introducing the jump of a (discontinuous) function:

$$[\![w]\!](x) = \lim_{\varepsilon\to 0^+} w(x - \varepsilon\nu_\gamma) - \lim_{\varepsilon\to 0^+} w(x + \varepsilon\nu_\gamma) \qquad x \text{ on } \gamma \in \mathcal{F}_h^i.$$

Equation (4.1) is an exact representation of the error. Now we need to estimate it using the following main ingredients:

1) Using the so-called Clément interpolation operator $I_h : H^1(\Omega) \to V_h$ we may produce a discrete function v_h from any given v. Note that here the difficulty is that v is only in $H^1(\Omega)$ and point-wise evaluation is not possible. Therefore the Clément interpolant is defined via averaging, i.e. it is an approximation and not an interpolation in the strict sense).

2) $a(e_h, I_h v) = 0$ Galerkin orthogonality, $\Rightarrow a(e_h, v) = a(e_h, v) - a(e_h, I_h v) = a(e_h, v - I_h v)$.

3) So now, in all 3 terms of (4.1), we can use $v - I_h v$ instead of v, e.g. $\int_e r(v - I_h v)dx \le \|r\|_{0,e}\|v - I_h v\|_{0,e}$ where the estimate is due to Cauchy-Schwarz inequality.

4) Then use element-wise error estimates associated with the Clément interpolant $\|u - I_h u\|_{0,e} \le Ch|u|_{1,\tilde{e}}$, $\|u - I_h u\|_{0,\gamma} \le Ch_t^{1/2}|u|_{1,\tilde{e}}$. Note that the norm on the right hand side is taken over a small region around e and γ including the neighboring elements. With that, one obtains:

$$\begin{aligned} a(e_h, v) \le C\|v\|_{1,\Omega}\Bigg[&\sum_{e\in\mathcal{T}_h} h_e^2 \int_e r^2 dx \\ &+ \sum_{\gamma\in\mathcal{F}_h^N} h_\gamma \int_\gamma R_b^2 ds + \sum_{\gamma\in\mathcal{F}_h^i} h_\gamma \int_\gamma [\![-(K\nabla u_h)\cdot\nu]\!]^2 ds\Bigg]^{1/2} \end{aligned} \tag{4.2}$$

   The fact, that we sum over different elements several times is corrected by the constant C.

5) Last step: take $v = e_h$ and use coercivity:

$$\alpha\|e_h\|_{1,\Omega}^2 \le a(e_h, e_h) \le C\|e\|_{1,\Omega}\eta \quad \Rightarrow \quad \|e_h\|_{1,\Omega} \le \frac{C}{\alpha}\eta.$$

with the computable quantity

$$\eta^2 = \sum_{e \in \mathcal{T}_h} h_e^2 \int_e r^2 dx + \sum_{\gamma \in \mathcal{F}_h^N} h_\gamma \int_\gamma R_b^2 ds + \sum_{\gamma \in \mathcal{F}_h^i} h_\gamma \int_\gamma [\![-(K \nabla u_h) \cdot \nu]\!]^2 ds.$$

Remark 4.1.

- The jump term $[\![.]\!]$ gives an estimate of how well the continuity of fluxes over interior and Neumann boundary edges/faces is satisfied.
- The expression above is not fully computable because of C and α (these could however be determined with some effort).
- The estimate is not robust in permeability for heterogeneous fields as α is related to the smallest permeability coefficient in the whole domain.
- Efficiency: one also needs to ensure that $\eta \leq C\|e\|_{1,\Omega}$ to make sure that $\eta \to 0$ with the same rate as the actual error. $\frac{\eta}{\|e\|_{1,\Omega}}$ is called the efficiency index.

4.2. Adaptive Mesh Refinement

Now we focus on the second goal formulated above, i.e. to achieve error control with small computational effort. A heuristic algorithm to achieve this is given as follows:

1) Choose an initial mesh $\mathcal{T}_{h_0}$ sufficiently fine. Set $i = 0$.
2) Compute u_{h_i} on current mesh $\mathcal{T}_{h_i}$.
3) Compute error estimate $\eta(u_{h_i})$. If $\eta(u_{h_i}) < TOL$: Stop.
4) Select subset of elements $\mathcal{T}'_{h_i} \subset \mathcal{T}_{h_i}$ which should be refined; This set is choosen according to the local distribution of the error $\eta(u_{h_i})$. The idea is to write the error estimate as a sum of local, element-wise contributions

$$\eta(u_{h_i}) = \left(\sum_{e \in \mathcal{T}_{h_i}} \eta_e^2(u_{h_i}) \right)^{1/2} \tag{4.3}$$

   and to refine only those elements where η_e is sufficiently large.
5) Refine mesh and interpolate u_h to new mesh
6) Go to 2)

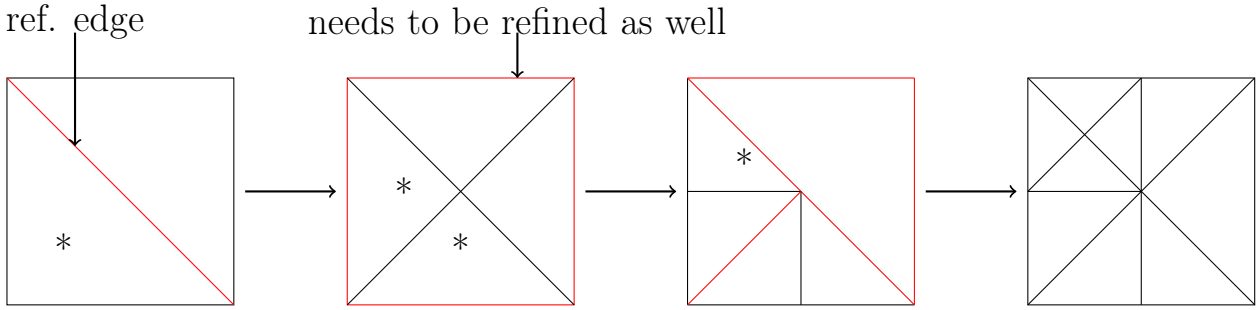


Figure 4.1.: Illustration of bisection refinement.

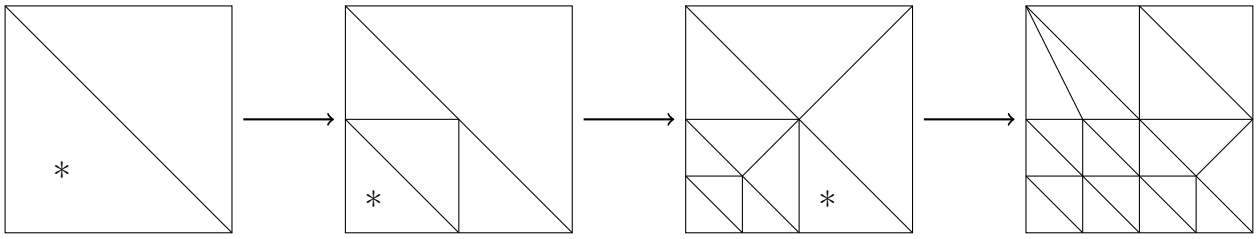


Figure 4.2.: Illustration of red-green refinement.

Local Mesh Refinement Methods

Bisection Refinement

Bisection refinement refers to the fact that an element selected for refinement is subdivided into two smaller elements, independent of the dimension. In all the fidures below the elements to be refined are denoted by an asterisk $*$. The newest vertex bisection algorithm illustrated in Figure 4.1 ensures the conformity of the refined mesh:

- Only specific edges may refined, so-called refinement edges, are allowed to be refined by the algorithm. In the initial mesh the refinement edges are manually chosen.
- A new vertex is introduced when a refinement edge is refined. Then the label refinement edge is given to the edges that are opposite the new vertex.
- Refinement of an element may trigger also the refinement of elements far away.

Red-Green Refinement

Red, also called regular, refinement corresponds to subdividing an element into 2^n elements where n is the spatial dimension. In order to facilitate local refinement and mesh conformity this is combined with bisection refinement which is also called green refinement in this context:

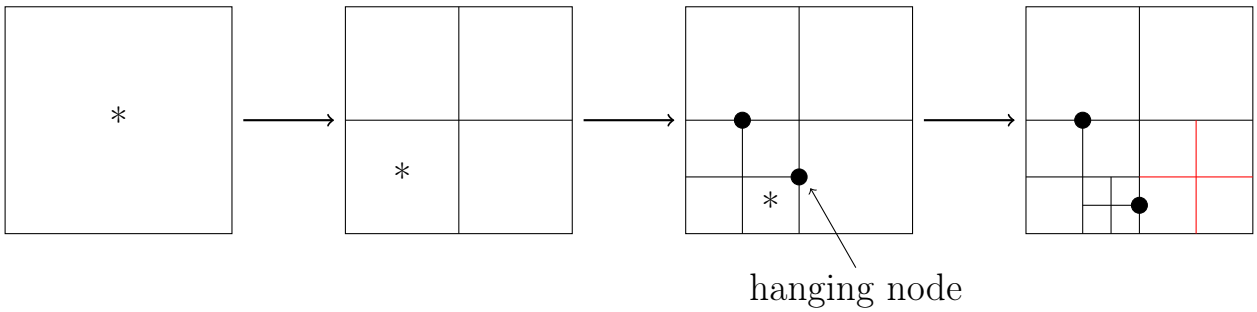


Figure 4.3.: Illustration of hanging node refinement.

- Bisection refinement, if not done properly like in newest vertex bisection, may lead to very small angles of elements. Therefore, bisection is only allowed once, i.e. an element that resulted from green refinement is not allowed to be refined again. In order to still refine that element the idea is to undo the green refinment and replace it with a red refinement instead which pushes the green refinement at another place.
- As with bisection refinement this may imply global changes in the mesh.
- Red refinement is trivial with triangles (two-dimensional simplex) and cuboid elements (any dimension) but non-trivial for simplices of dimension three and higher. In order to refine the simplices in a stable way without producing degenerate angles one can use Freudenthal's algorithm.

Hanging Node Refinement

In this refinement method the idea is to allow certain types of non-conforming meshes. As far as the mesh is concerned, the mesh is identical to red-green refinement but without the green refinement:

- Only one hanging node per edge is allowed (if two hanging nodes would occur on one edge, a refinement needs to be introduced on one of the neighbouring cells to prevent this).
- In order to cope with the mesh non-conformity the finite element functions are defined by interpolating the value in the hanging node from the vertices at the endpoint of the edge. The FE space defined in this way is conforming.

Hierarchical Refinement

Efficient implementations of the above mentioned stretgies rely on a hierarchical (tree) data structure:

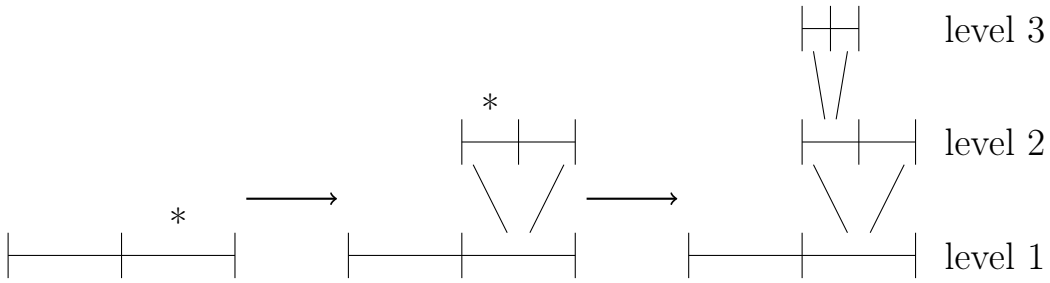


Figure 4.4.: Illustration of hanging node refinement.

- When an element is refined it is actually not removed from the grid but only the new resulting elements are added and the ancestor information is stored. In this way a refinement can also be removed easily, e.g. in instationary problems.
- All these methods are available in DUNE.

Fixed fraction strategy

In step 4) of the adaptive algorithm the question was which elements should be refined?

Idea: On an optimal mesh, the error contribution should be approximately equal on each element. Therefore, refine a fraction of elements with the largest error contribution. Determine largest η^* such that

$$\sum_{\eta_e \geq \eta^*} \eta_e^2 \geq \xi \eta(u_{h_i}),$$

where η_e is the error contribution of element e and $\xi \in [0,1]$ is a parameter, called bulk fraction parameter. Then refine all elements with $\eta_e \geq \eta^*$.

4.3. Numerical Results

AMR for the Reentrant Corner Problem

In order to evaluate the adaptive algorithm we consider again the reentrant corner problem given in example 3.10.

In a first experiment the choice of the bulk fraction parameter ξ is investigated for P_1 finite elements. Table 4.1 shows some properties of the meshes generated for a fixed tolerance value of 0.05. For small values of $\xi < 0.5$ the tolerance is reached with about 5000 degrees of freedom. Large values of $\xi > 0.6$ results in about 10000 degrees of freedom. On the other hand the number of iterations

Table 4.1.: Efficiency in terms of accuracy per degrees of freedom of meshes generated with different values for the bulk fraction parameter ξ and a tolerance of 0.05.

ξ	iterations	depth	N	$\lvert u - u_h \rvert_{1,\Omega}$	η
0.1	39	15	5395	1.30e-02	4.94e-02
0.2	22	14	5498	1.29e-02	4.89e-02
0.3	15	14	5323	1.30e-02	4.99e-02
0.4	12	14	5933	1.22e-02	4.71e-02
0.5	10	12	6996	1.12e-02	4.30e-02
0.6	9	11	8312	1.04e-02	3.99e-02
0.7	8	10	10129	1.01e-02	3.77e-02
0.8	7	9	10501	1.15e-02	4.05e-02
0.9	6	8	10714	1.47e-02	4.83e-02

of the adaptive algorithm (each time requiring the solution of a finite element problem) decreases from 39 to 6. So in terms of computation time a value of $\xi \approx 0.5 \ldots 0.8$ is most effective. Also note that the meshes resulting in about the same finite element error may be quite different. For $\xi = 0.1$ the smallest mesh size is $h_t \approx 2^{-15} = 1/32768$ while for $\xi = 0.9$ the smallest mesh size is $h_t \approx 2^{-8} = 1/256$.

Next we compare different types of mesh refinement for P_1 and Q_1 finite elements in figure 4.5. The figure shows the estimated and the true error for non-conforming refinement (hanging nodes) with triangles, conforming triangular meshes, bisection type refinement and non-conforming quadrilateral mesh refinement. With respect to degrees of freedom conforming and non-conforming regular refinement on triangles is asymptotically identical (with small advantages for conforming refinement on coarse meshes). Bisection refinement is a bit more efficient and quadrilateral meshes are substantially more efficient. The efficiency index is about 3 for all types of meshes. The corresponding meshes for the four different refinement types are shown in figure 4.6. In the top row nonconforming and conforming regular refinement with triangles is shown. The refinement regions have very similar shapes (the conforming mesh is more refined). The bottom row shows bisection and nonconforming quadrilateral refinement. Clearly, these two meshes look different. What is very interesting that there is less refinement along the diagonal $y = -x$ in the quadrilateral case.

Finally, we turn to the combination of adaptive refinement and higher (but fixed) polynomial degree. Figure 4.7 shows the true error versus number of degrees of freedom (as a measure of computational complexity) for polynomial degree 1 and uniform refinement as well as polynomial degrees $1 \ldots 4$ using

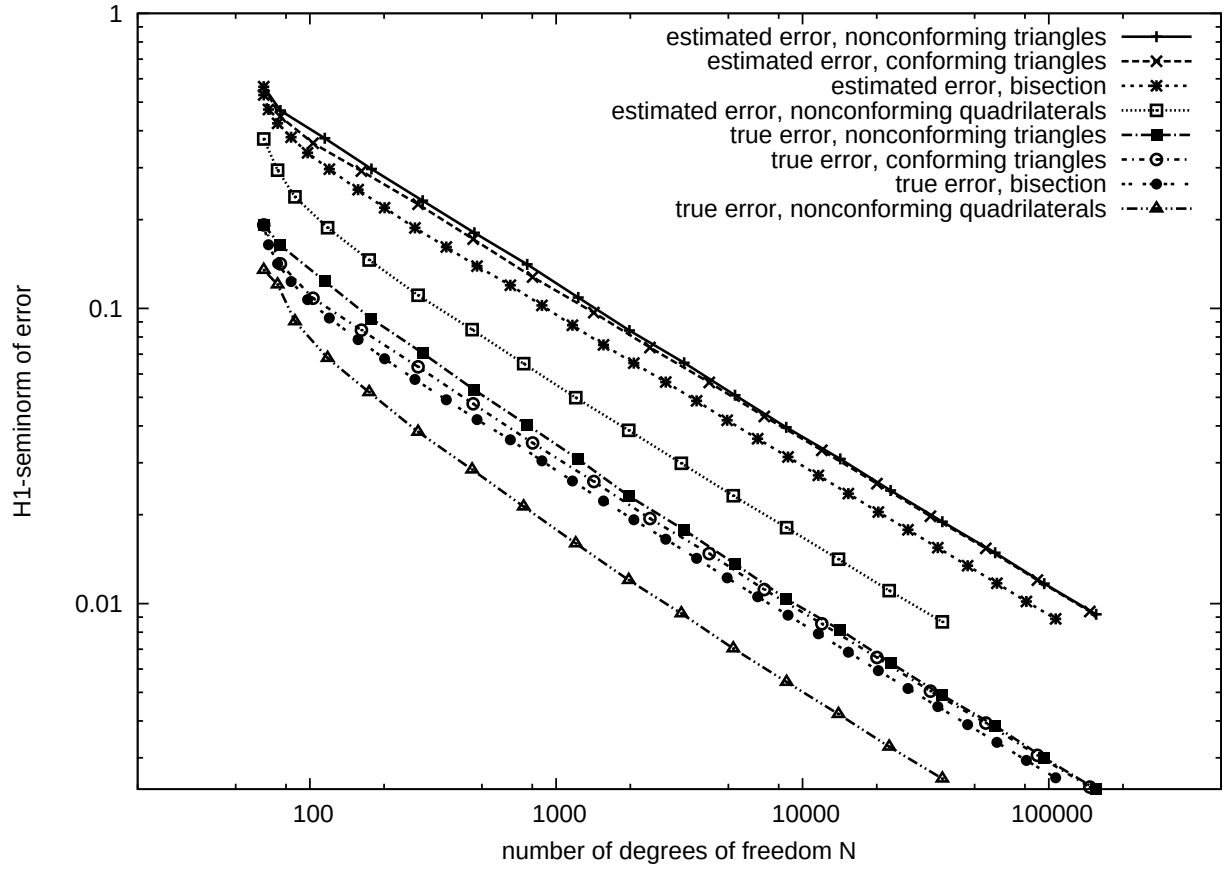


Figure 4.5.: Estimated and true error versus number of degrees of freedom for different local mesh refinement types. A bulk fraction parameter $\xi = 1/2$ was used.

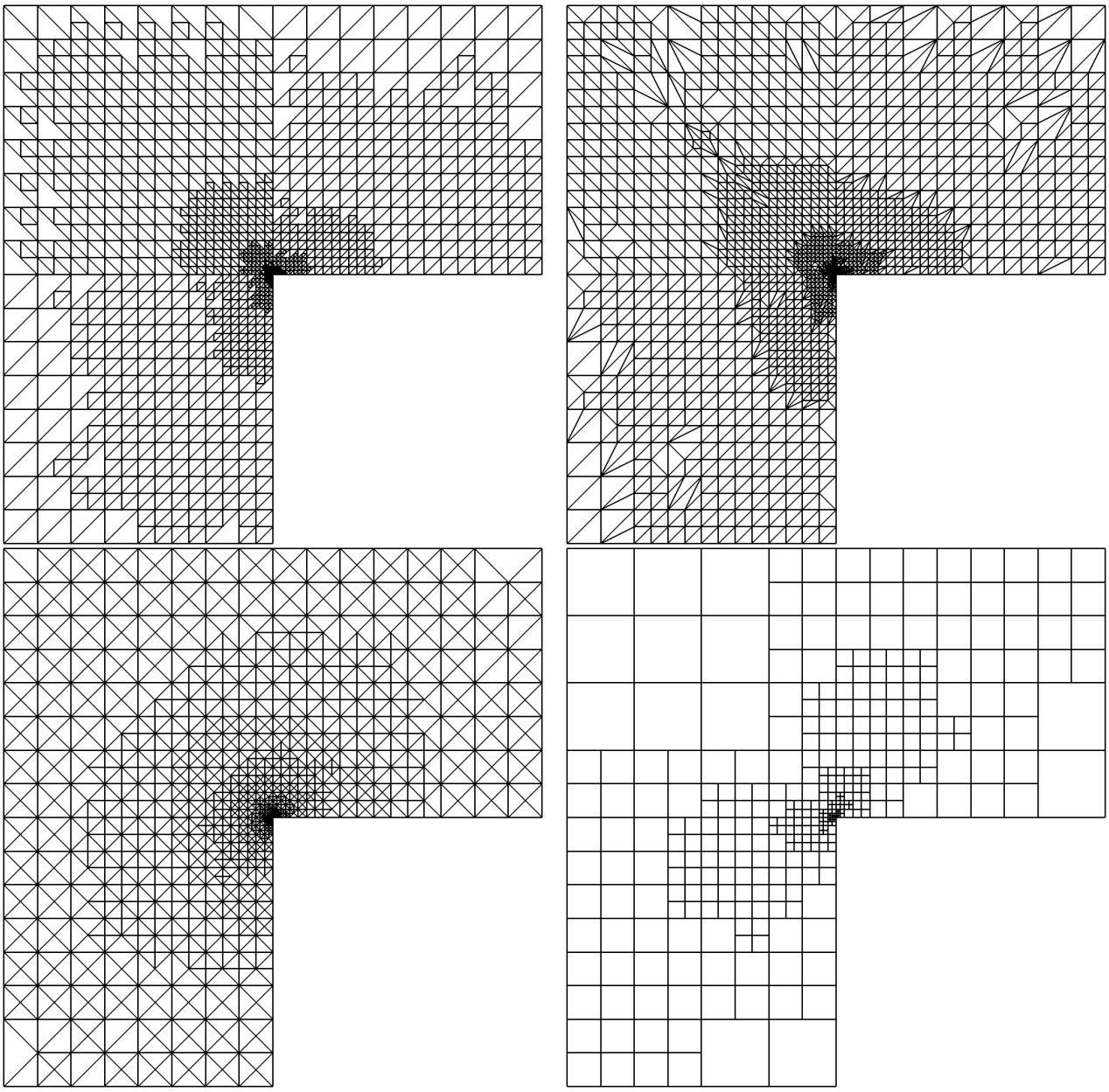


Figure 4.6.: Comparison of different local mesh refinement techniques at about the same absolute error of $|u - u_h|_{1,\Omega} = 0.03$: Nonconforming and conforming refinement with triangles, bisection refinement and nonconforming refinement with quadrilaterals (from top to bottom, left to right).

bisection type refinement. This figure can be compared directly to figure 3.13. Clearly, in comparison to the case of uniform refinement shown in figure 3.13 the asymptotic convergence rate now improves with increasing polynomial degree. A quantitative comparison is shown in table 4.2. In the case of full regularity the convergence rate in the H^1-norm is $O(h^k)$ for polynomial degree k. For uniform refinement we have $h = N^{-1/2}$, i.e. the optimal convergence rate with respect to N is $O(N^{-k/2})$. The third column in table 4.2 shows that *we can recover the convergence rate expected for a fully regular solution* with respect to number of degrees of freedom! The situation can be improved further by varying also the polynomial degree from element to element, choosing a high polynomial degree away from the reentrant corner and a low polynomial degree close to the corner. This leads to exponential convergence with respect to N. Table 4.2 also illustrates that the efficiency index depends slightly on the polynomial degree.

The meshes generated by the adaptive algorithm using P_1 and P_2 elements are shown in figure 4.8. The P_2 mesh is much more locally refined as is expected from the equilibration strategy. Away from the corner the error is reduced by $(1/2)^2$ for each refinement and near the corner it is only reduced by $(1/2)^{2/3}$ due to the low regularity.

AMR for Heterogeneous Permeability Fields

Now we consider adaptive mesh refinement for the groundwater flow problem with heterogeneous permeability. We point out that the error estimate is not robust with respect to coefficient variations, still the local contributions to the error estimator may be used heuristically for adaptive mesh refinement.

Figure 4.9 shows the magnitude of the Darcy velocity in a heterogeneous groundwater flow problem. Clearly, preferential flow paths can be identified. Although the variation in permability is large it is smooth enough in this case to allow the solution u to be in H^2 and therefore the finite element method shows optimal convergence of order 1 in the H^1-norm for polynomial degree 1. This is confirmed also by the error estimator in Table 4.3 where uniform and adaptive refinement is compared. In fact the table shows that both variants achieve about the same error for the same number of degrees of freedom. The difference, however, lies in the spatial distribution of the error contribution which is shown in Figure 4.10. In the case of uniform refinement (upper plot) there are about equally many elements with small, medium and large contributions to the total error, on the adaptively refined mesh each element contributes about the same error and there are much fewer elements with very small or very large contribution.

The general behaviour of adaptive mesh refinement for a fully regular problem is the following: In a first phase, the local mesh refinement algorithm tries to

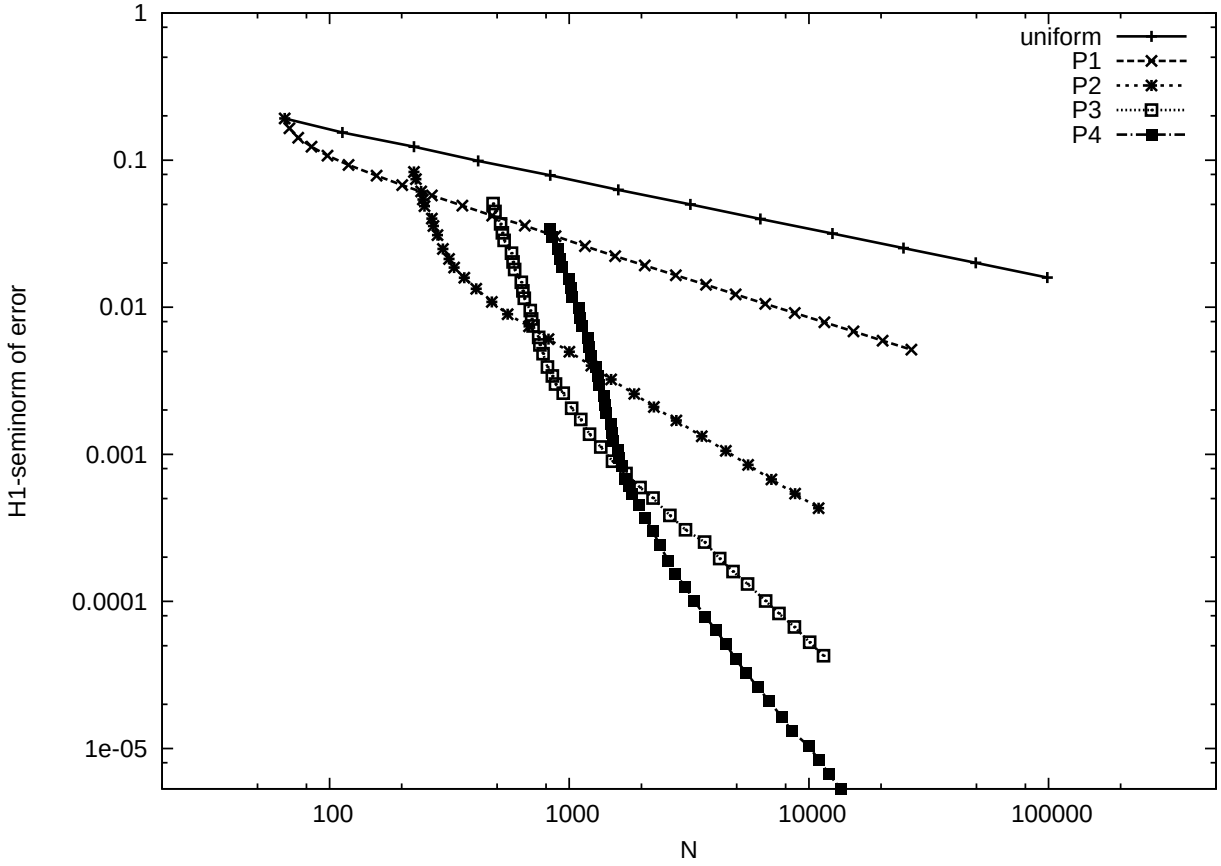


Figure 4.7.: Error versus number of degrees of freedom when combining adaptive mesh refinement with increasing polynomial degree.

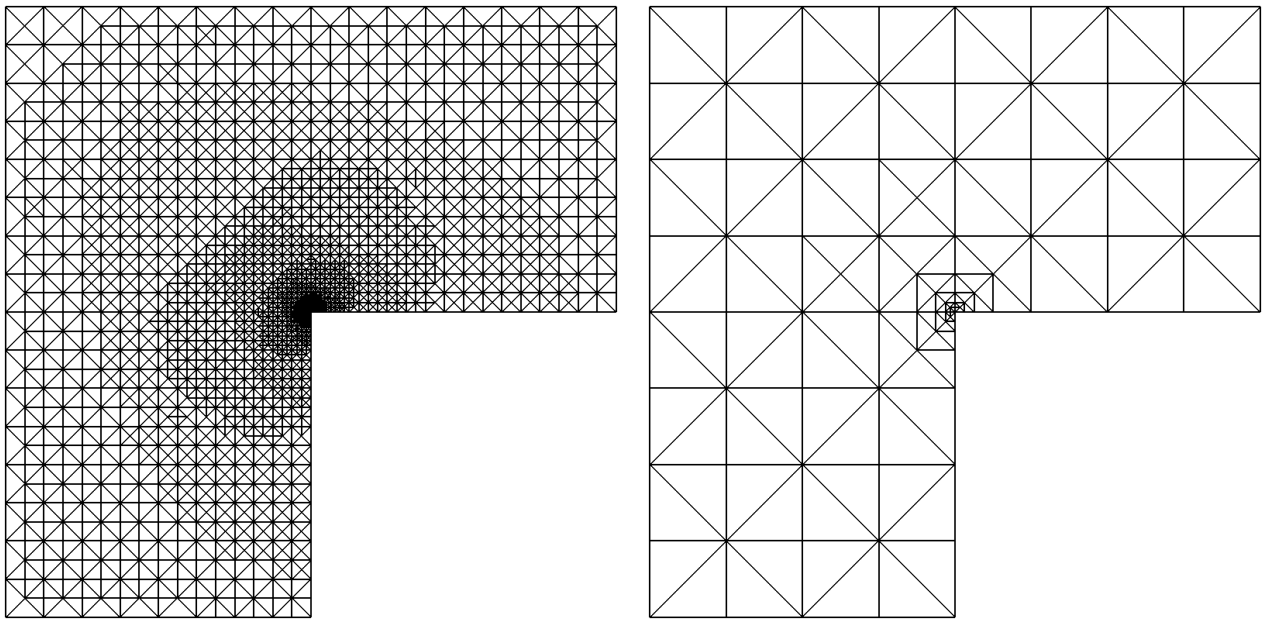


Figure 4.8.: Comparison of adaptive meshes using P_1 (left) and P_2 conforming finite elements at the same absolute error of $|u - u_h|_{1,\Omega} \approx 0.02$.

Table 4.2.: Convergence order and efficiency index for adaptive mesh refinement with varying polynomial degree. Bisection type refinement was used as refinement technique.

N	$\|u - u_h\|_{1,\Omega}$	|rate|	η	$\eta/\|u - u_h\|_{1,\Omega}$
$k = 1$				
15355	6.84e-03	0.51	2.36e-02	3.44
20312	5.92e-03	0.52	2.04e-02	3.44
26779	5.15e-03	0.50	1.78e-02	3.44
$k = 2$				
6979	6.76e-04	1.01	2.61e-03	3.86
8765	5.41e-04	0.98	2.08e-03	3.84
10985	4.30e-04	1.02	1.66e-03	3.86
$k = 3$				
8701	6.70e-05	1.42	2.88e-04	4.30
10072	5.28e-05	1.63	2.27e-04	4.31
11515	4.28e-05	1.57	1.83e-04	4.27
$k = 4$				
10993	8.39e-06	2.19	4.76e-05	5.68
12109	6.74e-06	2.26	3.81e-05	5.65
13541	5.32e-06	2.12	3.00e-05	5.64

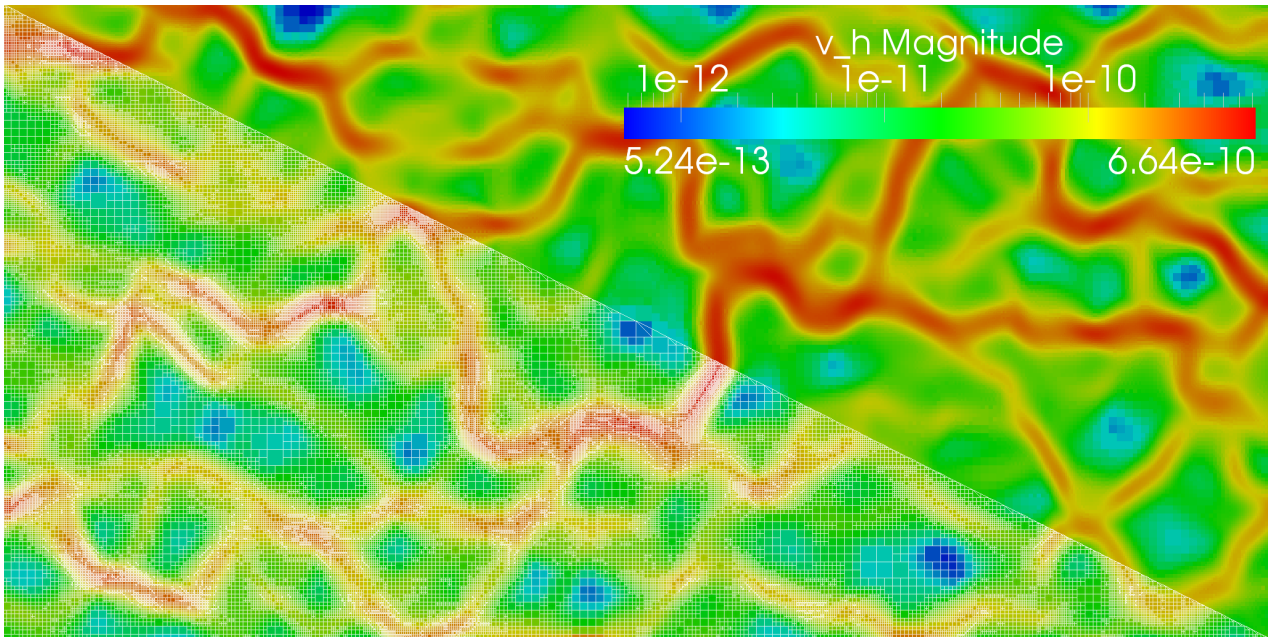


Figure 4.9.: Adaptively refined mesh and velocity magnitude for a heterogenous groundwater flow problem.

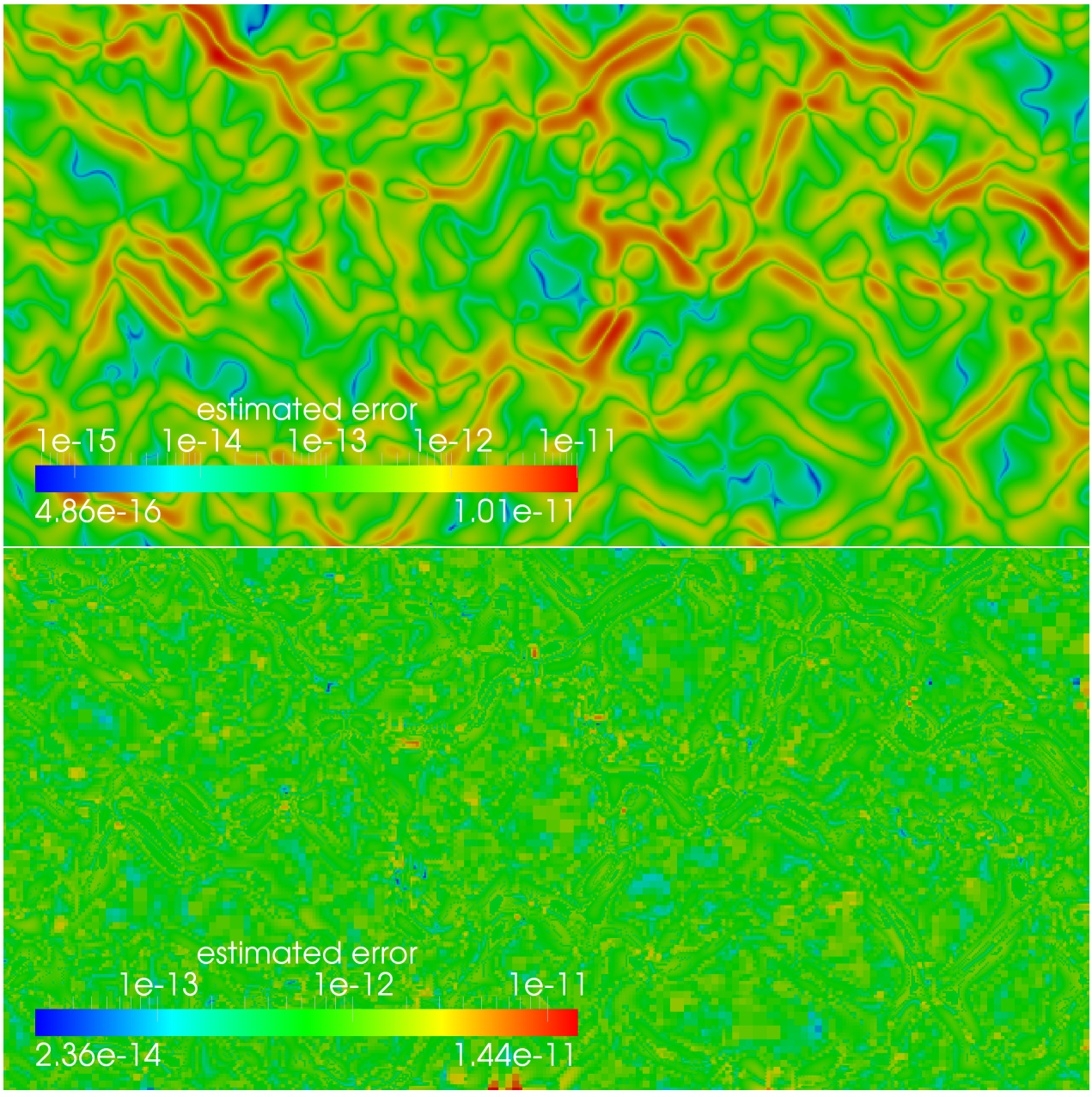


Figure 4.10.: Spatial distribution of the local error contributions η_e.

uniform		adaptive	
DOF	est. error	DOF	est. error
861	5.02e-09	861	5.02e-09
3321	2.74e-09	1864	4.09e-09
13041	1.45e-09	4562	2.63e-09
51681	7.37e-10	11405	1.58e-09
205761	3.70e-10	28816	9.36e-10
		76504	5.52e-10
		207618	3.28e-10

Table 4.3.: Comparison of uniform and adaptive refinement for a heterogeneous problem.

equi-distribute the error contribution for each element. After this is achieved, uniform mesh refinement takes place.

AMR for Wells

Finally, we consider adaptive mesh refinement for problems with sources and sinks.

Table 4.4 shows results for adaptive mesh refinement in a problem with resolved wells, i.e. the source term is distributed over a finite and fixed region. In this case enormous savings (factor 10 in number of degrees of freedom for the same error) can be achieved in comparison to uniform mesh refinement.

Table 4.5 shows results for adaptive refinement in a problem with a point source, i.e. the source region is shrinked as the mesh is refined. As has been pointed out above the continuous solution is not in H^1 in this case and convergence can not be shown with the theory presented here. It turns out that the error estimator also reflects this situtation by indicating no convergence.

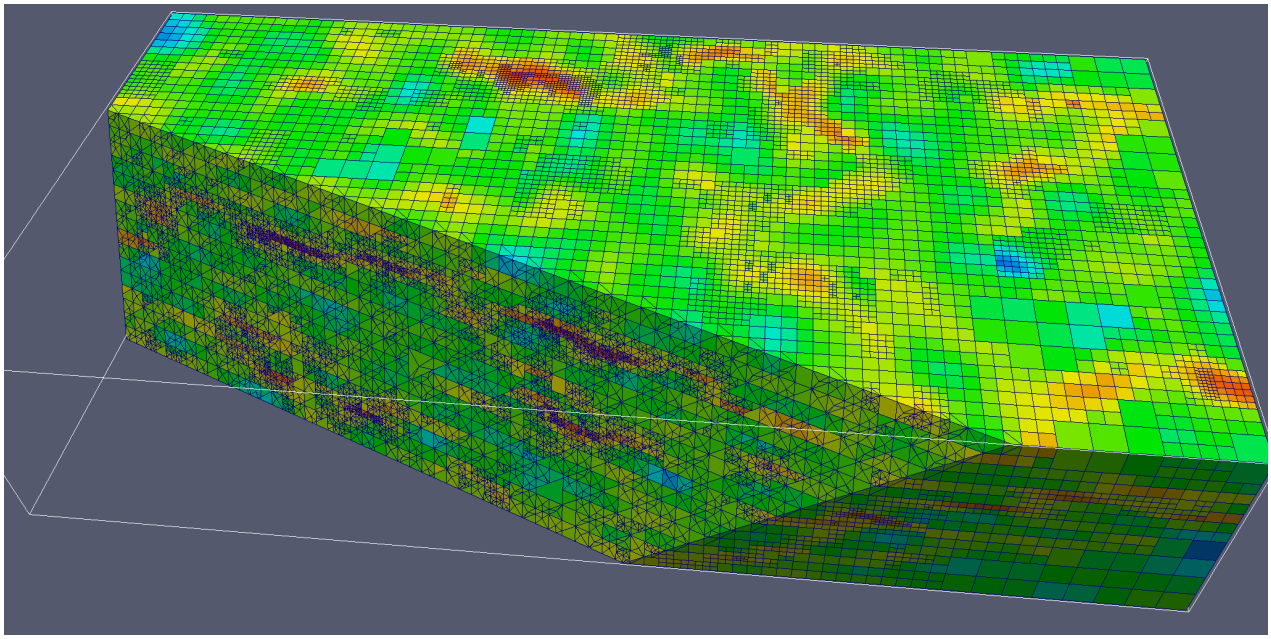


Figure 4.11.: Adaptively refined mesh for heterogeneous groundwater flow in three space dimensions.

uniform		adaptive	
DOF	est. error	DOF	est. error
3321	2.19352e-07	3321	2.19352e-07
13041	1.14575e-07	3326	1.82352e-07
51681	5.84299e-08	3379	1.17863e-07
205761	2.95422e-08	3837	7.70912e-08
		6218	5.29155e-08
		13533	3.44439e-08
		31977	2.14247e-08
		79809	1.33291e-08

Table 4.4.: Comparison of uniform and adaptive refinement for a resolved well.

uniform		adaptive	
DOF	est. error	DOF	est. error
3321	2.19352e-07	3321	2.19352e-07
13041	2.11367e-07	3326	2.22618e-07
51681	2.08426e-07	3347	2.26191e-07
205761	2.07497e-07	3379	2.27344e-07
		3411	2.29072e-07
		3443	2.30945e-07
		3479	2.32823e-07

Table 4.5.: Comparison of uniform and adaptive refinement for a point source (δ right hand side).

Chapter 5.

Cell-centered Finite Volume Method for Elliptic Problems

5.1. Motivation

Figure 5.1 shows the flow field of a heterogeneous groundwater flow problem with a source located in a single element computed on a relatively coarse mesh. When we zoom in close to the source and consider the solution (visualized by glyphs here) on individual elements we obtain the results shown in Figure 5.2. The left image shows the solution obtained by the finite element method. Arrows are pointing in opposite directions across the edge shown in the center of the image. This is clearly unphysical. The solution shown on the right is computed on the same mesh with the so-called cell-centered finite volume method (CCFV). It does not show this effect. In fact the x-component of velocity, $v_x = v \cdot \nu$ with $\nu = (0,1)^T$ is continuous across the edge shown in the center. Velocity components are shown in detail in Figure 5.3.

5.2. Cell Centered Finite Volume Method (CCFV)

We consider again the elliptic problem with inhomogeneous Dirichlet and flux type boundary conditions:

$$\begin{aligned} -\nabla \cdot (K \nabla u) &= f \text{ in } \Omega, \\ u &= g \text{ on } \Gamma_D \subset \partial\Omega, \\ -(K \nabla u) \cdot \nu &= j \text{ on } \Gamma_N = \partial\Omega \setminus \Gamma_D. \end{aligned} \tag{5.1}$$

This equation satisfies the conservation property

$$\int_{\partial\Omega} -(K \nabla u) \cdot \nu ds = \int_{\Gamma_N} j ds + \int_{\Gamma_D} -(K \nabla u) \cdot \nu ds = \int_{\Omega} f dx \tag{5.2}$$

In so-called conservative methods the numerical solution u_h satisfies a similar property. The solution u_h from FEM does satisfy this property only in an approximate sense.

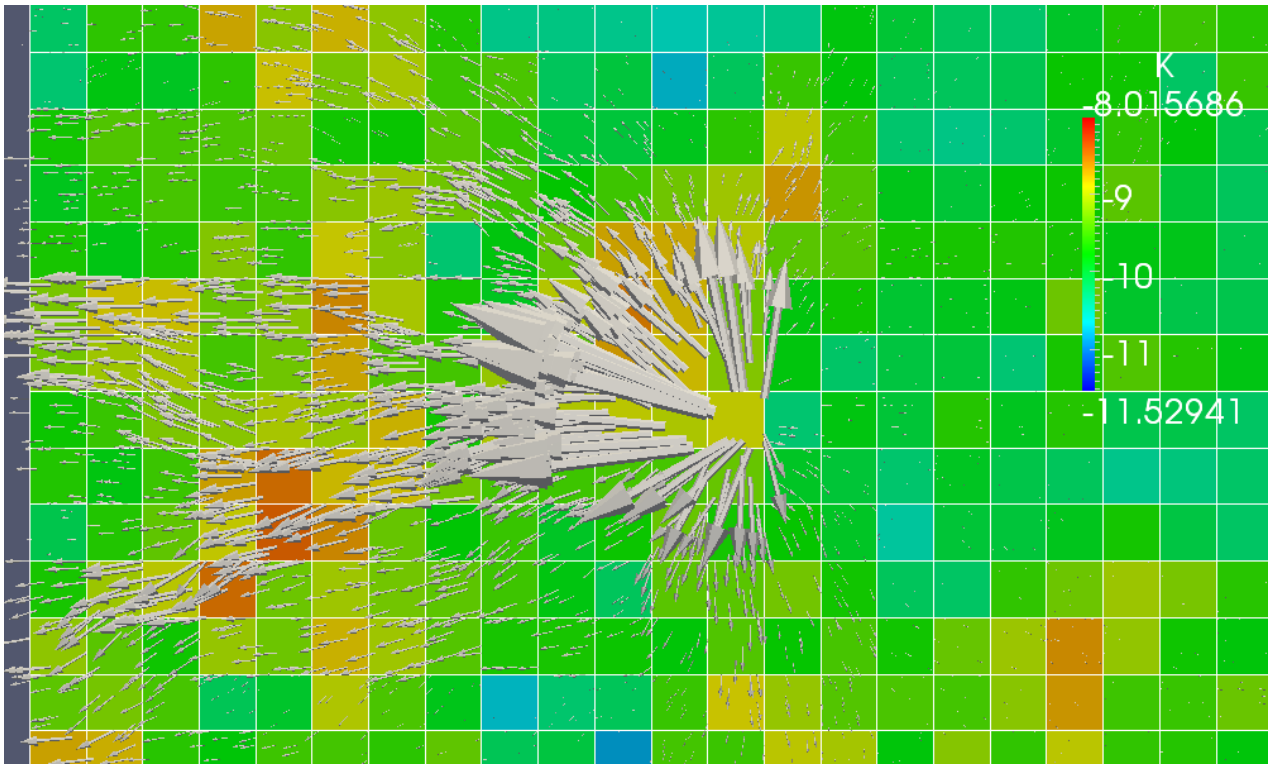


Figure 5.1.: FEM solution of flow near a source.

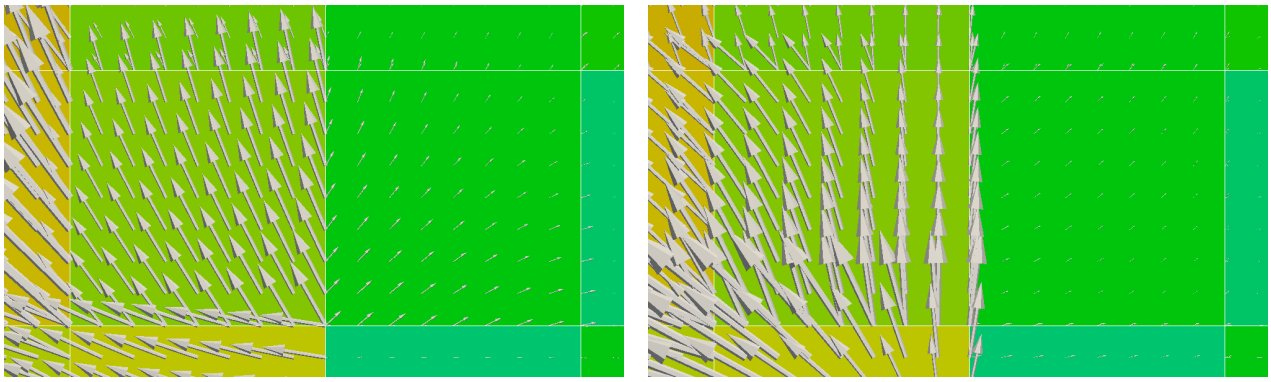


Figure 5.2.: Zoom of solution near a source. Comparison of finite element method (left) and finite volume method (right).

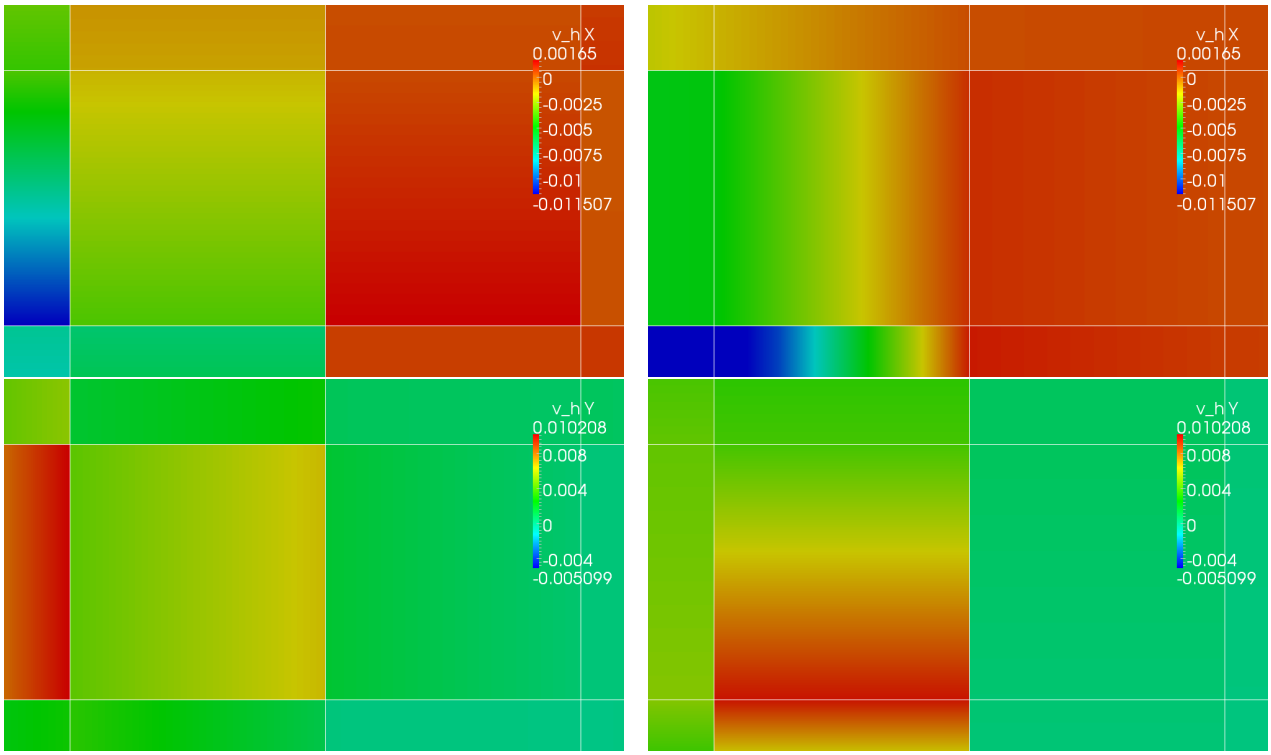


Figure 5.3.: Zoom of solution near a source. Finite volume method on the right and finite element method on the left. x-component of velocity in upper row and y-component of velocity in bottom row.

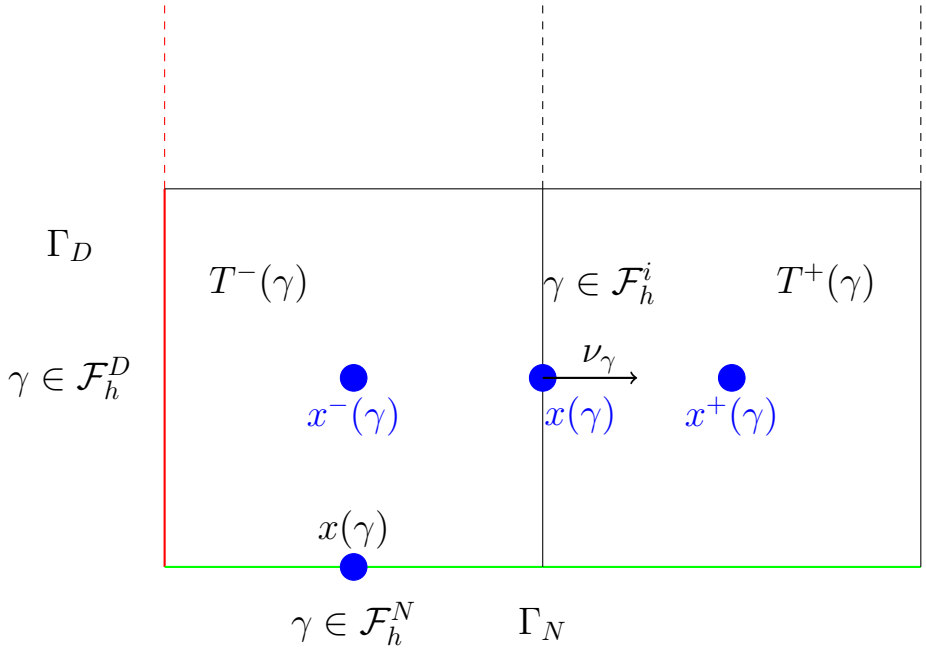


Figure 5.4.: Notation for faces, face normals and adjacent elements in the finite volume method.

Integration of the PDE over a set $\omega \subset \Omega$: $\int_{\partial\omega} -(K\nabla u)\nu ds = \int_\omega f dx$ can be viewed as a kind of weak formulation with a discontinuous test function

$$v_\omega(x) = \begin{cases} 1 & x \in \omega \\ 0 & \text{else} \end{cases}.$$

However, such $v_\omega \notin H^1(\Omega)$ so it is not covered by our theory and the conforming finite element method. The full importance of the conservation property only becomes clear for

- Numerically obtained flow fields used in transport (see motivation).
- Nonlinear hyperbolic problems (not treated in this course).

Our aim is now to construct a numerical methods satisfying an exact conservation property for domains ω which are unions of cells, i.e.

$$\omega = \bigcup_{e \in \mathcal{T}_h'} \overline{e}, \quad \mathcal{T}_h' \subset \mathcal{T}_h.$$

For simplicity we consider a structured axiparallel mesh $\mathcal{T}_h$ consisting of cubes in $d = 1, 2, 3$ dimensions, where $\mathcal{F}_h = \mathcal{F}_h^D \cup \mathcal{F}_h^N \cup \mathcal{F}_h^i$ is the set of Dirichlet, Neumann and interior faces. Figure 5.4 introduces some notation useful for the definition of the CCFV method.

As an approximation space we choose the space of element-wise constant functions

$$W_h = \left\{ v \in L^2(\Omega) : v|_e = \text{const } \forall e \in \mathcal{T}_h \right\}.$$

Since $W_h \not\subset H^1(\Omega)$ (functions are discontinuous) such a method is called "non-conforming".

Again, we follow the idea to multiply (5.1) by a test function $v \in W_h$, integrate and use elementwise integration by parts:

$$\begin{aligned}
-\int_\Omega \nabla\cdot(K\nabla u)v dx &= -\sum_{e\in\mathcal{T}_h}\int_e \nabla\cdot(K\nabla u)v dx \\
&= -\sum_{e\in\mathcal{T}_h}\int_{\partial e}(K\nabla u)\cdot\nu v ds \\
&= \sum_{e\in\mathcal{T}_h}\Bigg[\int_{\partial e\cap\Omega} -(K\nabla u)\cdot\nu v ds \\
&\qquad + \int_{\partial e\cap\Gamma_D} -(K\nabla u)\cdot\nu v ds + \int_{\partial e\cap\Gamma_N} jv ds\Bigg] \\
&= \sum_{\gamma\in\mathcal{F}_h^i}\int_\gamma [\![-(K\nabla u)\cdot\nu_\gamma v]\!] ds \\
&\qquad + \sum_{\gamma\in\mathcal{F}_h^D}\int_\gamma -(K\nabla u)\cdot\nu v ds + \sum_{\gamma\in\mathcal{F}_h^N}\int_\gamma jv ds.
\end{aligned} \tag{5.3}$$

Note that here v has to be in the brackets $[\![.]\!]$ for the inner faces since v is discontinuous over two elements e, e'.

Equation (5.3) holds for the solution of the continuous problem. Now we consider the discrete case where we approximate u by $u_h \in W_h$. We introduce a "numerical flux" (here it is a velocity) at the interior faces by:

$$j_h^i(\gamma) = \underbrace{-k_{\text{eff}}}_{\substack{\in\mathbb{R},\text{ scalar,}\\ \text{to be defined}}} \underbrace{\frac{u_h(x^+(\gamma)) - u_h(x^-(\gamma))}{\|x^+(\gamma) - x^-(\gamma)\|}}_{\approx\frac{\partial u}{\partial\nu_\gamma}} \tag{5.4}$$

where k_{eff} is an effective permeability. For (5.4) to be an accurate flux we assume that

1) $x^+(\gamma) - x^-(\gamma)$ is orthogonal to γ, i.e. it has the same direction as ν_γ.
2) $K\nu_\gamma = \lambda\cdot\nu_\gamma$, i.e. ν_γ is an eigenvector of K. Grids satisfying this property are called K-orthogonal.

At a Dirichlet boundary a numerical flux can be defined in a similar way by

$$j_h^D(\gamma) = -k(T^-(\gamma))\frac{g(x(\gamma)) - u_h(x^-(\gamma))}{\|x(\gamma) - x^-(\gamma)\|}, \tag{5.5}$$

where g is the boundary condition function.

By putting this in equation (5.3), using the midpoint rule for quadrature, and rearranging the terms such that we have the known quantities on the right hand side, the numerical method becomes:

$$\begin{aligned} -\sum_{\gamma\in\mathcal{F}_h^i} k_{\text{eff}}(\gamma)\frac{u_h(x^+(\gamma))-u_h(x^-(\gamma))}{\|x^+(\gamma)-x^-(\gamma)\|}[\![v]\!]|\gamma| \\ + \sum_{\gamma\in\mathcal{F}_h^D} k(T^-(\gamma))\frac{u_h(x^-(\gamma))}{\|x^+(\gamma)-x^-(\gamma)\|}v|\gamma| \\ = \sum_{t\in\mathcal{T}_h} f(x(t))v(x(t))|t| - \sum_{\gamma\in\mathcal{F}_h^N} j(x(\gamma))v|\gamma| \\ + \sum_{\gamma\in\mathcal{F}_h^D} k(T^-(\gamma))\frac{g(x(\gamma))}{\|x(\gamma)-x^-(\gamma)\|}v|\gamma| \end{aligned} \tag{5.6}$$

where $|e|$ denotes cell volume and $|\gamma|$ denotes face area. This can be written in short form as

$$\text{Find } u_h \in W_h: \qquad a_h(u_h, v) = l_h(v) \qquad \forall v \in W_h \tag{5.7}$$

where the bilinear form a_h and linear form l_h are given by the left and right hand side of equation (5.6).

Remark 5.1.
- The bilinear form a_h and the linear form l_h are mesh dependent. This is typical for a nonconforming method.
- Dirichlet boundary conditions are only fulfilled for $h \to 0$.
- The approximation is second order accurate in L^2 for equidistant grids.
- No constraints are necessary for the function space W_h as the Dirichlet as well as the Neumann condition are built into the bilinear form and right hand side linear functional.
- Equation (5.7) results in a linear system for the values of u_h at the cell centers. Formally this is done in the same way as in the finite element method by introducing a basis. A natural basis is $\Phi_h = \{\phi_e : e \in \mathcal{T}_h\}$ with
$$\phi_e(x) = \begin{cases} 1 & x \in e \\ 0 & \text{else} \end{cases}.$$

5.3. Conservation Properties

We now check that the solution u_h solving (5.7) satisfies the desired conservation properties.

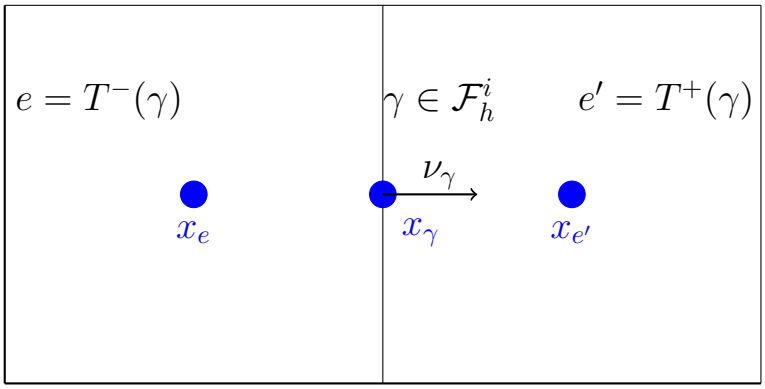


Figure 5.5.: Situation at an interior face γ.

Local Conservation

Consider first a single interior face $\gamma \in \mathcal{F}_h^i$ as shown in Figure 5.5. Setting the test function v to $v = \phi_e$ and considering the face γ results in the flux from e to e':

$$F_{e,e'} = k_{\text{eff}}(\gamma)\frac{u_h(x_{e'}) - u_h(x_e)}{\|x_{e'} - x_e\|}(\underbrace{\phi_e^-}_{=1} - \underbrace{\phi_e^+}_{=0})|\gamma| = k_{\text{eff}}(\gamma)\frac{u_h(x_{e'}) - u_h(x_e)}{\|x_{e'} - x_e\|}|\gamma|.$$

Setting the test function v to $v = \phi_{e'}$ and considering the face γ results in the flux $F_{e',e} = -F_{e',e}$.

Also observe that when we set the test function v to $v = \phi_e + \phi_{e'}$ the contribution of face γ in equation (5.6) is zero due to linearity in v (or since $[\![v]\!] = 0$ on γ).

Global Conservation

Consider now any connected subdomain made up of a collection of elements $\overline{\Omega}' = \bigcup_{e \in \mathcal{T}_h' \subset \mathcal{T}_h} \bar{e}$ and set $v(x) = \chi_{\Omega'} \in W_h$. Observe $[\![v]\!] = 0$ for all interior faces γ in Ω' and assume ν_γ points outward of Ω' for $\gamma \subset \partial\Omega'$. Then (5.7) reduces to (with slight abuse of notation):

$$\begin{aligned}
&-\sum_{\gamma \in \mathcal{F}_h^i \cap \partial\Omega'} k_{\text{eff}}(\gamma)\frac{u_h(x^+(\gamma)) - u_h(x^-(\gamma))}{\|x^+(\gamma) - x^-(\gamma)\|}|\gamma| \\
&\qquad - \sum_{\gamma \in \mathcal{F}_h^D \cap \partial\Omega'} k(T^-(\gamma))\frac{g(x(\gamma)) - u_h(x^-(\gamma))}{\|x^+(\gamma) - x^-(\gamma)\|}|\gamma| + \sum_{\gamma \in \mathcal{F}_h^N \cap \partial\Omega'} j(x(\gamma))v|\gamma| \\
&= \sum_{e \in \mathcal{T}_h} f(x(e))|e|
\end{aligned}$$

If $\Omega' = \Omega$, then the first term on the left hand side vanishes as well and we obtain global conservation.

5.4. Effective Permeability

In order to derive the effective permeability we consider the one-dimensional flow problem in $\Omega = (x', x'')$:

$$\frac{d\sigma(x)}{dx} = 0, \qquad \sigma(x) = -k(x)\frac{du(x)}{dx}.$$

From the first equation it follows that the flux $\sigma(x)$ is a constant:

$$\sigma(x) = C = const$$

and from the main theorem of calculus and the second equation we get

$$\int_{x'}^{x''} \frac{du}{dx}dx = u(x'') - u(x') = \int_{x'}^{x''} -\frac{\sigma(x)}{k(x)}dx = -C\int_{x'}^{x''} \frac{1}{k(x)}dx.$$

From this we can now get the flux constant C as

$$C = -\frac{\|x' - x''\|}{\int_{x'}^{x''} \frac{1}{k(x)}dx}\ \frac{u(x'') - u(x')}{\|x'' - x'\|} = k_{\text{eff}}\frac{u(x'') - u(x')}{\|x'' - x'\|}$$

and we identify the effective permeability as

$$k_{\text{eff}} = \frac{\|x'' - x'\|}{\int_{x'}^{x''} \frac{1}{k(x)}dx}.$$

Now consider the special situation where the permeability is two-valued

$$k(x) = \begin{cases} k_l & x \le (x' + x'')/2 \\ k_r & x > (x' + x'')/2 \end{cases}.$$

Then we obtain the effective permeability as the harmonic average:

$$k_{\text{eff}} = \frac{\|x'' - x'\|}{\int_{x'}^{(x'+x'')/2} \frac{1}{k(x)}dx + \int_{(x'+x'')/2}^{x''} \frac{1}{k(x)}dx} = \frac{\|x'' - x'\|}{\frac{\|x''-x'\|}{2}(\frac{1}{k_l} + \frac{1}{k_r})} = \frac{2}{(\frac{1}{k_l} + \frac{1}{k_r})}. \quad (5.8)$$

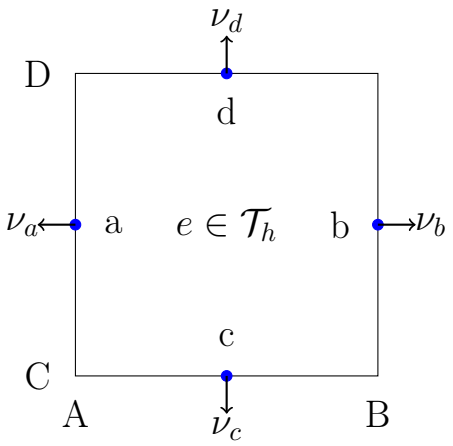


Figure 5.6.: Notation for velocity interpolation with Raviart-Thomas elements.

5.5. Velocity Reconstruction

The cell-centred Finite Volume Method computes

- cell pressure values and
- face velocities (or fluxes, which is the same for $\rho = const.$) .

For transport computations also a velocity field in the interior of elements is required. Then we use the following interpolation scheme.

Number the faces of a given cell as $i \in a, b, c, d$ (see Figure 5.6) and denote the fluxes through the faces as $J_i = j_e(x_i)\nu_i$ where ν_i denotes the unit outer normal vector of the face. The normal fluxes J_i are given by the finite volume scheme while we wish to reconstruct the flux vector field $j_e(x)$ defined on the cell e. This is done by simple linear interpolation of each component:

$$j_e(x,y) = \begin{pmatrix} j_{e,x}(x,y) \\ j_{e,y}(x,y) \end{pmatrix} = \begin{pmatrix} J_a(\frac{x-B}{B-A}) + J_b(\frac{x-A}{B-A}) \\ J_c(\frac{x-D}{C-D}) + J_d(\frac{x-C}{C-D}), \end{pmatrix}$$

where $e = (A, B) \times (C, D)$ is the cell.

This way we obtain a velocity field defined in each element and therefore in the complete domain Ω. This velocity field satisfies the property that normal component is continuous. The construction here is valid for axiparallel cuboid meshes but it can be extended to affine simplicial meshes es well.

We may also compute the divergence, i.e. $\partial_x j_{e,x} = \frac{J_a+J_b}{B-A}$, $\partial_y j_{e,y} = \frac{J_c+J_d}{D-C}$ and

$$\nabla \cdot j_e = \frac{J_a + J_b}{B - A} + \frac{J_c + J_d}{D - C}.$$

On the other hand, since the discrete fluxes J_i computed by the scheme satisfy (insert $v = \phi_e$ in equation (5.6)):

$$\int_{\partial e} j_e \nu ds = J_a(D-C) + J_b(D-C) + J_c(B-A) + J_d(B-A) = \int_e f dx.$$

Dividing by $(B-A)(D-C)$ and assuming $f = 0$ we obtain

$$\frac{J_a + J_b}{B-A} + \frac{J_c + J_d}{D-C} = \nabla \cdot j_e = 0,$$

i.e. the reconstructed flux field is even pointwise divergence free for $f = 0$ (if not, it balances exactly the given source/sink term). This is a very desirable property.

The considerations in this subsection can be generalized to continuous functions as well. A desirable property of a continuous flux field $j : \Omega \to \mathbb{R}^d$ is

$$j \in H(\mathrm{div}; \Omega) = \{\sigma \in (L^2(\Omega))^d : \nabla \cdot \sigma \in L^2(\Omega)\}$$

Similarly to Lemma 3.2 one has

Lemma 5.2. Let $j : \Omega \to \mathbb{R}^d$ be piece-wise and componentwise C^∞-function on a subdivision $\mathcal{T}_h$ of Ω. Then $j \in H(div; \Omega)$ if and only if $j \cdot \nu$ is continuous at internal subdomain boundaries.

Chapter 6.

Solution of Linear Systems

6.1. Motivation

- Numerical solution of linear elliptic PDEs with the methods discussed in this course results in the solution of systems of linear algebraic equations.
- Typically, these systems are sparse, i.e. the have $\mathcal{O}(n)$ non- zero entries instead n^2. Solution algorithms should exploit these zeros by employing special matrix formats (data structures).
- Sparsity of the linear system depends on the chosen basis in the FE method.
- Assembling the FE system matrix costs $\mathcal{O}(n)$ operations, solving the system costs $\mathcal{O}(n^\alpha)$ where typically $\alpha > 1$ for many methods. E.g. $\alpha = 3$ for naive Gaussian elimination. As a consequence, solving the linear system dominates computation time for large n.

6.2. Overview of Solvers

There are basically two approaches: direct and iterative solvers.

Direct Solvers

These are based on Gaussian elimination/ LU decomposition. Such methods suffer from the "fill-in" problem illustrated in Figure 6.1.

- Gaussian elimination depends heavily on the ordering of equations and unknowns as shown in Figure 6.1.
- Fill-in can be minimized by reordering, however, exact fill-in minimalization is NP-complete (best known algorithms have exponential runtime $\mathcal{O}(\alpha^n)$).
- In addition, the reordering might conflict with pivoting. Therefore there are two classes of algorithms for matrices which can be factorized without pivoting (e.g. symmetric positive definite matrices) and thos which can not).

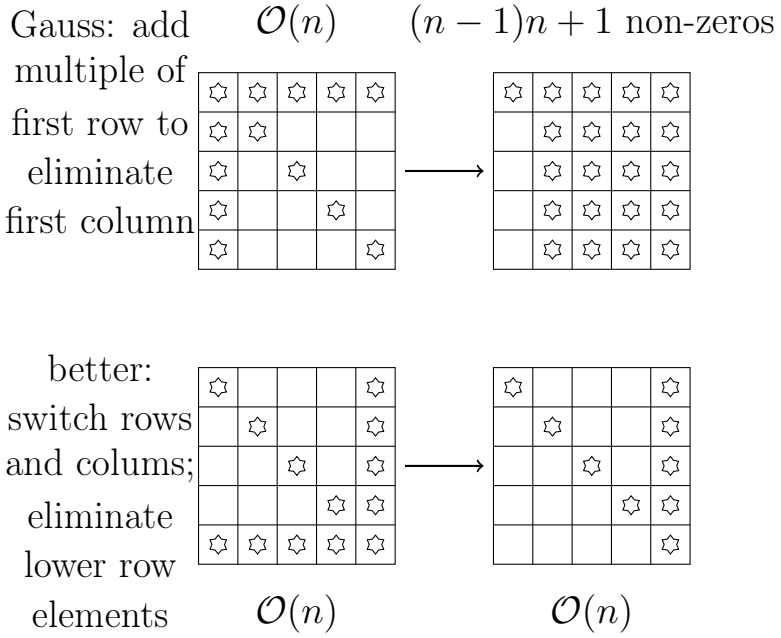


Figure 6.1.: Fill-in problem in sparse Gaussian elimination. Amount of fill-in depends on the ordering of the equations (rows) and unknowns(columns).

- There are very good heuristic methods which give close to exact fill-in minimalization, e.g. nested dissection ordering based on graph partitioning algorithms.

Nested dissection works as follows:

- Use ideas from graph theory: A graph $G = (V, E)$; $E \subseteq V \times V$ can be defined for any $A \in \mathbb{R}^{n \times n}$ by setting $V = \{1, \ldots, n\}$ and $e = (i, j) \in E \Leftrightarrow a_{ij} \neq 0$.
- A graph seperator $S \subseteq V$ is a subset of vertices which, when taken from the graph, leaves two disconnected subgraphs.
- The graph bisection problem is to find a seperator such that both remaining subgraphs have the same number of vertices (up to one) and the number of vertices in the seperator is minimal.
- Nested dissection is a recursive algorithm that splits the graph in two by solving a graph bisection problem. Then it orders first the two subgraphs, then the seperator and then splits the two subgraphs recursively.

Iterative Solvers

for the solution of the linear system $Ax = b$ construct a sequence $x^{(0)}, x^{(1)}, x^{(2)}, \ldots$ such that

$$\lim_{k \to \infty} x^{(k)} = x.$$

The convergence of these methods typically depends on the eigenvalues of A. Some methods do not converge on certain matrices even if they are invertible. For A s.p.d. the spectral condition number

$$\kappa(A) = \frac{\lambda_{max}(A)}{\lambda_{min}(A)}$$

is relevant. For the Poisson equation, uniform mesh refinement and standard FE/FV method one finds $\kappa(A) \leq Ch^{-2}$. For heterogeneous groundwater flow, the constant C depends on the maximal and minimal permeabilities.

Condition number and number of iterations. The condition number determines the number of iterations it takes to reduce the initial error to an acceptable size. Assume that convergence is geometric, i.e. we have

$$\underbrace{\|x - x^{(k)}\|}_{\text{error in } k\text{-th iterate}} \leq \rho \|x - x^{(k-1)}\| \leq \rho^k \underbrace{\|x - x^{(0)}\|}_{\text{initial error}}.$$

From this we get the number of iterations it takes to reduce the norm of the initial error by a given tolerance:

$$\frac{\|x - x^{(k)}\|}{\|x - x^{(0)}\|} \leq \rho^k \leq TOL \qquad \Rightarrow \qquad k = \frac{\log(TOL)}{\log(\rho)}$$

where ρ is the convergence factor ($\rho < 1$) which is a function of the the condition number $\rho = f(\kappa(A))$.

The error $\|x - x^{(k)}\|$ is in general not accessible, but one can compute $\|A(x - x^{(k)})\| = \|Ax - Ax^{(k)}\| = \|b - Ax^{(k)}\|$ and stops the computation when $\|b - Ax^{(k)}\| \leq TOL\|b - Ax^{(0)}\|$.

Figure 6.2 gives a comparison of the complexity of direct and iterative methods for the discretization of a Poisson problem. It is important to note that the complexity of iterative methods decreases with the dimension d.

6.3. Linear Iterative Methods

Our goal is to solve $Ax = b$. Given an initial guess $x^{(0)}$, the error in the k-th iterate $x^{(k)}$ is

$$e^{(k)} = x - x^{(k)}. \tag{6.1}$$

Then the defect is given by the identity

$$Ae^{(k)} = A(x - x^{(k)}) = Ax - Ax^{(k)} = b - Ax^{(k)} =: d^{(k)} \quad \text{“defect”} \tag{6.2}$$

Scheme	$d=2$	$d=3$
Gaussian elimination (GEM)	n^3	n^3
Banded GEM	n^2	$n^{7/3}$
nested dissection ordering GEM	$n^{3/2}$	n^2
Gauss-Seidel, Jacobi	n^2	$n^{5/3}$
conjugate gradient, SOR	$n^{3/2}$	$n^{4/3}$
multigrid	n	n

Figure 6.2.: Comparison of complexity of direct and iterative methods.

$Ae^{(k)} = d^{(k)}$ is the error equation. Note that the right hand side is fully computable. The error equation is not easier to solve than the original equation but the idea is now to solve it approximately by replacing the matrix A by a simpler matrix W:

$$Wv^{(k)} = d^{(k)} \quad \Rightarrow \quad v^{(k)} = W^{-1}d^{(k)}.$$

From (6.1) we obtain $x = x^{(k)} + e^{(k)} \approx x^{(k)} + v^{(k)}$ and the iteration becomes

$$x^{(k+1)} = x^{(k)} + v^{(k)} = x^{(k)} + W^{-1}(b - Ax^{(k)}). \tag{6.3}$$

Consider articular choices for W. With the splitting $A = L + U + D$, into L and U strict lower/upper triangular matrices, D a diagonal matrix we set:

$$\begin{aligned} W_R &= \frac{1}{\omega}\mathbb{I},\ \omega \in \mathbb{R} && \text{(Richardson)},\\ W_{Jac} &= D && \text{J(acobi)},\\ W_{GS} &= L + D && \text{(Gauss-Seidel)}. \end{aligned}$$

Convergence of linear iterative methods is based on the analysis of error propagation:

$$\begin{aligned} e^{(k+1)} &= x - x^{(k+1)} = x - x^{(k)} - W^{-1}(b - Ax^{(k)}) = e^{(k)} - W^{-1}(Ax - Ax^{(k)})\\ &= e^{(k)} - W^{-1}Ae^{(k)} = (\mathbb{I} - W^{-1}A)e^{(k)} \end{aligned}$$

where $S = (\mathbb{I} - W^{-1}A)$ is the iteration matrix and $e^{(k+1)} = Se^{(k)} = S^{k+1}e^{(0)}$ is the error propagation equation. Clearly, $S^k \to 0$ implies convergence of the method

Theorem 6.1. A, W regular matrices. The iteration (6.3) converges iff $\rho(S) < 1$, where $\rho(B)$ is the spectral radius of matrix B, i.e. the largest eigenvalue $\rho(B) = \max\{|\lambda| : \lambda \text{ is EV of } B\}$. The convergence is independent of $x^{(0)}$ and b.

The following Theorem gives a characterization of the spectral radius for a particular class of matrices.

Theorem 6.2. A, W symmetric and positive matrices. Then the iteration

$$x^{(k+1)} = x^{(k)} + \frac{1}{\lambda_{max}(W^{-1}A)} W^{-1}(b - Ax^{(k)})$$

converges with rate $\rho = 1 - \frac{1}{\kappa(W^{-1}A)}$, where $\kappa(B) = \frac{\lambda_{max}(B)}{\lambda_{max}(B)}$ is the spectral condition number.

6.4. Descent Methods

Assume A is symmetric and positive definite. Descent methods are based on minimizing the functional

$$F(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle \tag{6.4}$$

where $\langle ., .\rangle$ denotes the Euclidean scalar product.

Theorem 6.3. The unique minimum of (6.4) coincides with the solution of $Ax = b$.

Proof. (Idea). Take any $y \in \mathbb{R}^n$:, assume x^* is the minimum of the functional F and write $y = x^* + v$. Then one can show that $F(y) = F(x^* + v) = F(x^*) + \langle Av, v\rangle$. The last term is positive for $v \neq 0$ since A is positive definite. □

The idea for construction of an iterative method is to minimization the functional along a search direction: Given an iterate $x^{(k)}$ and a "search direction" $p^{(k)}$ find $\alpha^{(k)} \in \mathbb{R}$ s.t. $F(x^{(k)} + \alpha^{(k)} p^{(k)})$ is minimal. Setting $g(\alpha) = F(x^{(k)} + \alpha^{(k)} p^{(k)})$ a necessary condition for the minimum is $\frac{dg}{d\alpha}(\alpha) = 0$. which has the solution

$$\alpha^{(k)} = \frac{\langle p^{(k)}, b - Ax^{(k)}\rangle}{\langle p^{(k)}, Ap^{(k)}\rangle}.$$

In the gradient descent method the search direction is choosen as the steepest descent direction, i.e. the negative gradient of F: $p^{(k)} = -\nabla F(x^{(k)}) = b - Ax^{(k)}$. This results in the scheme:

$$x^{(k+1)} = x^{(k)} + \frac{\langle b - Ax^{(k)}, b - Ax^{(k)}\rangle}{\langle b - Ax^{(k)}, b - Ax^{(k)}\rangle}(b - Ax^{(k)}). \tag{6.5}$$

Theorem 6.4. If A is symmetric and positive definite the gradient descent method converges and

$$\|x - x^{(k+1)}\|_A \leq \frac{\kappa(A) - 1}{\kappa(A) + 1} \|x - x^{(k)}\|_A$$

where $\|x\|_A = \sqrt{\langle x, Ax \rangle}$. Thus, $\kappa(A)$ should be small for fast convergence $\left(\frac{\kappa(A)-1}{\kappa(A)+1} \approx 1 - \frac{1}{\kappa(A)}\right)$.

Improvements of this method are possible:

1) Conjugate gradient method: remember previous search directions and ensure that new search direction is orthogonal to all previous ones: $\langle p^{(k)}, p^{(j)} \rangle_A = 0$ for $j < k$. The convergence rate of the conjugate gradient method is much better than the steepest descent method: $\rho = \frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}$.
2) Preconditioning: Combine descent method with a linear iterative method by applying it to the system $BAx = Bb$ where the condition number of BA is smaller than the condition number of A.

6.5. Multigrid Methods

Smoothing Property

Assume A is symmetric and positive definite with the eigenvalues $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\} \subset \mathbb{R}^+$ with $0 < \lambda_{min}(A) = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n = \lambda_{max}$.

The Richardson iteration $x^{(k+1)} = x^{(k)} + \omega(b - Ax^{(k)})$, $\omega \in \mathbb{R}$, has the iteration matrix $e^{(k+1)} = (\mathbb{I} - \omega A)e^{(k)}$. Let (λ_i, z_i) be an eigenpair: $Az_i = \lambda_i z_i$ for $i = 1...n$. The z_i form a basis of $\mathbb{R}^n$ and we can expand the error in terms of the eigenvectors:

$$\begin{aligned} e^{(k)} &= \sum_{i=1}^{n} c_i^{(k)} z_i \\ e^{(k+1)} &= (\mathbb{I} - \omega A)e^{(k)} = (\mathbb{I} - \omega A) \sum_{i=1}^{n} c_i^{(k)} z_i = \sum_{i=1}^{n} c_i^{(k)} (z_i - \omega \lambda_i z_i) \\ &= \sum_{i=1}^{n} c_i^{(k)} (1 - \omega \lambda_i) z_i = \sum_{i=1}^{n} c_i^{(k+1)} z_i \end{aligned}$$

Now take $\omega = 1/\lambda_n = 1/\lambda_{max}(A)$. Then

$$1 - \frac{\lambda_i}{\lambda_{max}} = \begin{cases} \text{small} & \text{if i is large (i.e. } \lambda_i > 0.5\lambda_{max}) \\ \text{large} & \text{if i is small} \end{cases}$$

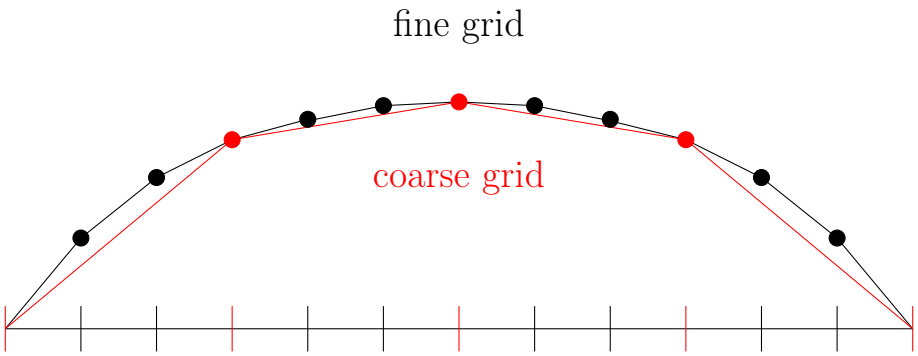


Figure 6.3.: Approximation of smooth fine grid error on a coarse grid.

For the Poisson equation, large $i \approx n$ correspond to high frequencies and small $i \approx 0$ correspond to low frequencies. Richardson, Jacobi and Gauss-Seidel methods effectively remove high frequency errors but not low frequency errors (only very slowly).

Coarse Grid Correction

Approximate low frequency error on a coarser grid, where it has high frequency. Use projection $P : \mathbb{R}^{n_c} \to \mathbb{R}^{n_f}$ e.g. linear interpolation. Solve $Ae = d$ approximately by replacing the fine grid error e by the (interpolated) error v_c on the coarse grid:

$$APv_c = d$$

Multiplying with P^T from the left results in the coarse grid system

$$A_c v_c = P^T A P v_c = P^T d$$

with A_c is a $n_c \times n_c$ matrix. One step of the coarse grid correction results in the update

$$\Rightarrow x^{new} = x^{old} + PA_c^{-1}P^T(b - Ax^{old}).$$

The matrix $PA_c^{-1}P^T$ has a nontrivial kernel, i.e. there are errors that can not be corrected by the coarse grid correction. Therefore, combine it with a smoother, e.g. Richardson iteration, to obtain the two grid method:

$$\boxed{\begin{array}{c} \text{Smoothing: } x^{(k+1/2)} = x^{(k)} \dfrac{1}{\lambda_{max}(A)}(b - Ax^{(k)}) \\ \text{Coarse grid corr.: } x^{(k+1)} = x^{(k+1/2)} PA_c^{-1}P^T(b - x^{(k+1/2)}) \end{array}} \tag{6.6}$$

If this principle is applied recursively, it is called multigrid method.

Chapter 7.

Low-order Schemes for Linear Transport

7.1. Introduction

We are now concerned with the solution of the first-order linear hyperbolic problem

$$\partial_t u + \nabla \cdot (\beta u) = f \qquad \text{in } \Omega \times \Sigma, \tag{7.1a}$$

$$u(x,0) = u_0(x) \qquad \text{at } t = 0,\ x \in \Omega, \tag{7.1b}$$

$$u(x,t) = g(x,t) \qquad \text{on } \Gamma^-(t) = \{x \in \partial\Omega : \beta(x,t) \cdot \nu(x) < 0\}, \tag{7.1c}$$

also called the transport equation. Here, $\Sigma = (0, t_F)$ is a time interval with final time t_F, $\Omega \subseteq \mathbb{R}^n$ is a spatial domain as before, $u : \Omega \times \Sigma \to \mathbb{R}$ is the unknown solution describing e.g. the concentration of a dissolved substance and $\beta : \Omega \times \Sigma \to \mathbb{R}^n$ is a given velocity field. The boundary Γ^- is the in-flow boundary.

Assuming the vector field β is smooth enough, we can put (7.1a) in "non-conservative" form:

$$\partial_t u + \beta \cdot \nabla u + (\nabla \cdot \beta) u = f. \tag{7.2}$$

Observe that the zero order term $(\nabla \cdot \beta)u$ vanishes when the vector field is "divergence free", i.e. $\nabla \cdot \beta = 0$ for all times.

Method of Characteristics

The method of characteristics provides an explicit solution formula for the transport equation and explains also the choice of boundary conditions.

Proposition 7.1. Assume $f = 0$, $\beta \in [C^1(\Omega \times \Sigma)]$ and $\nabla \cdot \beta = 0$ (this could be relaxed). For any given point in space-time $(\bar{x}, \bar{t}) \in \Omega \times \Sigma$ define the curve $(\hat{x}(s), \hat{t}(s))$:

$$\frac{d\hat{x}}{ds}(s) = \beta(\hat{x}(s), \hat{t}(s)) \ (s > 0), \qquad \hat{x}(0) = \bar{x} \tag{7.3a}$$

$$\frac{d\hat{t}}{ds}(s) = 1 \ (s > 0), \qquad \hat{t}(0) = \bar{t}. \tag{7.3b}$$

Then the solution of (7.1a) is constant along the curve given by (7.3) where it has the value $u(\bar{x}, \bar{t})$.

Proof. Differentiating the function $u(x,t)$ along the curve given by (7.3) we obtain:

$$\begin{aligned}\frac{d}{ds}u(\hat{x}(s),\hat{t}(s)) &= \frac{\partial u}{\partial t}(\hat{x}(s),\hat{t}(s))\frac{d\hat{t}}{ds}(s) + \sum_{i=1}^{n}\frac{\partial u}{\partial x_i}(\hat{x}(s),\hat{t}(s))\frac{d\hat{x}_i}{ds}(s)\\ &= \frac{\partial u}{\partial t}(\hat{x}(s),\hat{t}(s)) + \beta(\hat{x}(s),\hat{t}(s))\cdot\nabla u(\hat{x}(s),\hat{t}(s))\\ &= 0.\end{aligned}$$

Here we used the chain rule, then the definition of the curve and finally the nonconservative form of the equation. □

The following consequences result from the method of characteristics:

1) Any point $(\bar{x},\bar{t})$ with $\bar{t}=0$, i.e. at initial time, gives rise to a curve that traces through the space time domain and will possibly hit the outflow boundary $\Gamma^+(t)=\{x\in\partial\Omega : \beta(x,t)\cdot\nu(x)>0\}$. The value of the solution along that curve is $u_0(\bar{x})$.

2) The same holds true for a point $(\bar{x},\bar{t})$ with $\bar{x}\in\Gamma^-(\bar{t})$, i.e. at the inflow boundary. The value of the solution along that curve is then $g(\bar{x},\bar{t})$.

3) Since the velocity field β is continuously differentiable, any point $(\bar{x},\bar{t})\in\Omega\times\Sigma$ is only on one characteristic curve and the solution is uniquely defined.

4) A discontinuous initial condition $u_0(x)$ gives rise to a discontinuous solution at later times. This, as it turns out, is in fact a notorious difficulty when solving the transport equation numerically.

7.2. Finite Volume Methods

Consider an affine conforming spatial mesh $\mathcal{T}_h$ and a subdivision of the time interval Σ:

$$0=t_0<t_1<\ldots<t_K=t_F,\ \Delta t_k=t_k-t_{k-1},\ 1\le k\le K.$$

Again, we employ the space of piecewise constant functions

$$W_h=\{v\in L^2(\Omega) : v|_e = const\ \ \forall e\in\mathcal{T}_h\}$$

to construct a finite volume method: For a test function $v \in W_h$ and time interval (t_{k-1}, t_k) we get by element-wise integration by parts

$$\begin{aligned}
&\int_{t_{k-1}}^{t_k} \int_{\Omega} \big[\partial_t u + \nabla \cdot (\beta u)\big] v \, dxdt \\
&\qquad = \sum_{e \in \mathcal{T}_h} \left[\int_e \int_{t_{k-1}}^{t_k} \partial_t uv \, dtdx + \int_{t_{k-1}}^{t_k} \int_e \nabla \cdot (\beta u) v \, dxdt \right] \\
&\qquad = \sum_{e \in \mathcal{T}_h} \int_e \left(u(t_k) - u(t_{k-1})\right) v \, dx + \sum_{e \in \mathcal{T}_h} \int_{t_{k-1}}^{t_k} \int_{\partial e} (\beta \cdot \nu) uv \, dsdt \\
&\qquad = \sum_{e \in \mathcal{T}_h} \int_e \left(u(t_k) - u(t_{k-1})\right) v \, dx + \sum_{\gamma \in \mathcal{F}_h^i} \int_{t_{k-1}}^{t_k} \int_{\gamma} [\![\beta \cdot \nu_\gamma uv]\!] \, dsdt \\
&\qquad\quad + \sum_{\gamma \in \mathcal{F}_h^+} \int_{t_{k-1}}^{t_k} \int_{\gamma} (\beta \cdot \nu_\gamma) uv \, dsdt + \sum_{\gamma \in \mathcal{F}_h^-} \int_{t_{k-1}}^{t_k} \int_{\gamma} (\beta \cdot \nu_\gamma) gv \, dsdt.
\end{aligned}$$

Here we have used the following notation:

- $\mathcal{F}_h^i$: interior edges/faces as above.
- $\mathcal{F}_h^- = \{\gamma \in \mathcal{F}_h^{\partial\Omega} : \gamma \subseteq \Gamma^-\}$: edges/faces on the inflow boundary.
- $\mathcal{F}_h^+ = \{\gamma \in \mathcal{F}_h^{\partial\Omega} : \gamma \subseteq \Gamma^+\}$: edges/faces on the outflow boundary $\Gamma^+(t) = \{x \in \partial\Omega : \beta(x,t) \cdot \nu(x) > 0\}$.

Putting known quantities to the right hand side we obtain the identity

$$\begin{aligned}
&\sum_{e \in \mathcal{T}_h} \int_e \left(u(t_k) - u(t_{k-1})\right) v \, dx + \sum_{\gamma \in \mathcal{F}_h^i} \int_{t_{k-1}}^{t_k} \int_{\gamma} [\![\beta \cdot \nu_\gamma uv]\!] \, dsdt \\
&\quad + \sum_{\gamma \in \mathcal{F}_h^+} \int_{t_{k-1}}^{t_k} \int_{\gamma} (\beta \cdot \nu_\gamma) uv \, dsdt \\
&= \int_{t_{k-1}}^{t_k} \int_{\Omega} fv \, dxdt - \sum_{\gamma \in \mathcal{F}_h^-} \int_{t_{k-1}}^{t_k} \int_{\gamma} (\beta \cdot \nu_\gamma) gv \, dsdt.
\end{aligned}$$

A fully-discrete formulation for the numerical solution $u_h^k \in W_h$ is obtained by first replacing the time integrals either by a first-order explicit or first-order

implicit approximation (quadrature) and then introducing a *numerical flux function*

$$[\![\beta \cdot \nu_\gamma uv]\!] = \Phi(\beta \cdot \nu_\gamma, u^-, u^+)[\![v]\!]$$

for the evaluation of u in the jump term at interior faces. Two choices for numerical flux functions are:

i) $\Phi_c(\beta \cdot \nu_\gamma, u^-, u^+) = (\beta \cdot \nu_\gamma)\frac{u^- + u^+}{2}$ (centred flux, average value), where u^- and u^+ are the values in cells $T^-(\gamma)$ and $T^+(\gamma)$ for the edge/face γ.

ii) $\Phi_u(\beta \cdot \nu_\gamma, u^-, u^+) = (\beta \cdot \nu_\gamma)\frac{u^- + u^+}{2} + \frac{|\beta \cdot \nu_\gamma|}{2}(u^- - u^+)$

$$= \begin{cases} (\beta \cdot \nu_\gamma)u^- & \text{if } \beta \cdot \nu_\gamma > 0 \\ (\beta \cdot \nu_\gamma)u^+ & \text{if } \beta \cdot \nu_\gamma < 0 \end{cases}$$

which is called "upwind flux". The upwind flux can also be interpreted as a stabilized centered flux. A physical motivation for the upwind flux is that the value of u at a point on an edge/face should be taken from the element where the flow comes from, i.e. which is upwind. A mathematical motivation for the upwind flux is that, when inserting the solution u as a test function, the additional term $\frac{|\beta \cdot \nu_\gamma|}{2}(u^- - u^+)^2$ is positive and ensures the coercivity of the bilinear form. In contrast, the centered flux may lead to unstable numerical schemes.

Depending on the evaluation of the time integral we now obtain the following schemes. Spatial integrals are approximated by the midpoint rule where necessary.

Explicit Finite Volume Scheme

Let $u_h^{k-1} \in W_h$ be given. Then $u_h^k \in W_h$ is obtained from

$$\begin{aligned} &\sum_{e \in \mathcal{T}_h} \left(u_h^k - u_h^{k-1}\right) v|e| + \sum_{\gamma \in \mathcal{F}_h^i} \Phi_u(\beta \cdot \nu_\gamma, u_h^{k-1,-}, u_h^{k-1,+})[\![v]\!]|\gamma|\Delta t_k \\ &\quad + \sum_{\gamma \in \mathcal{F}_h^+} (\beta \cdot \nu_\gamma) u_h^{k-1} v|\gamma|\Delta t_k = \sum_{e \in \mathcal{T}_h} f v|e|\Delta t_k - \sum_{\gamma \in \mathcal{F}_h^-} (\beta \cdot \nu_\gamma) g v|\gamma|\Delta t_k. \end{aligned} \tag{7.4}$$

Inserting the basis function representation $u_h^k = \sum_{e' \in \mathcal{T}_h} z_{e'}^k \varphi_{e'}$ and using the test function $v = \varphi_e$ one may explicitly solve for the coefficient z_e^k.

Implicit Finite Volume Scheme

Let $u_h^{k-1} \in W_h$ be given. Then $u_h^k \in W_h$ is obtained from

$$\begin{aligned}\sum_{e\in\mathcal{T}_h} \left(u_h^k - u_h^{k-1}\right) v|e| + \sum_{\gamma\in\mathcal{F}_h^i} \Phi_u(\beta\cdot\nu_\gamma, u_h^{k,-}, u_h^{k,+})[\![v]\!]|\gamma|\Delta t_k \\ + \sum_{\gamma\in\mathcal{F}_h^+} (\beta\cdot\nu_\gamma) u_h^k v|\gamma|\Delta t_k = \sum_{e\in\mathcal{T}_h} fv|e|\Delta t_k - \sum_{\gamma\in\mathcal{F}_h^-} (\beta\cdot\nu_\gamma) gv|\gamma|\Delta t_k.\end{aligned} \tag{7.5}$$

This approach requires the solution of a linear system at each timestep.

7.3. Stability Condition

Consider the explicit upwind scheme (7.4). Inserting the test function $\varphi_e = \chi_e$ for $e \in \mathcal{T}_h$ we obtain the explicit representation of the solution

$$(z_e^k - z_e^{k-1})|e| + \sum_{\substack{\gamma\in\mathcal{F}_h^e \\ \beta\cdot\nu_e>0}} \Delta t_k(\beta\cdot\nu_e) z_e^{k-1}|\gamma| + \sum_{\substack{\gamma\in\mathcal{F}_h^e, \\ \beta\cdot\nu_e<0}} \Delta t_k(\beta\cdot\nu_e) z_{nb(e,\gamma)}^{k-1}|\gamma| = b_e^{k-1}$$

where $\mathcal{F}_h^e = \{\gamma \in \mathcal{F}_h^i \cup \mathcal{F}_h^D : e = T^-(\gamma) \vee e = T^+(\gamma)\}$,

$$nb(e,\gamma) = \begin{cases} T^-(\gamma) & e = T^+(\gamma) \\ T^+(\gamma) & e = T^-(\gamma) \end{cases}$$

and b_e^{k-1} collects all contributions from the source term and boundary condition. Note that ν_e is the unit outer normal to the element e and not ν_γ. Solving for z_e^k we obtain

$$z_e^k = \left(1 - \Delta t_k \sum_{\substack{\gamma\in\mathcal{F}_h^e \\ \beta\cdot\nu_e>0}} \frac{(\beta\cdot\nu_e)|\gamma|}{|e|}\right) z_e^{k-1} - \Delta t_k \sum_{\substack{\gamma\in\mathcal{F}_h^e \\ \beta\cdot\nu_e<0}} \frac{(\beta\cdot\nu_e)|\gamma|}{|T|} z_{nb(e,\gamma)}^{k-1} + b_e^{k-1}. \tag{7.6}$$

Collecting all coefficients z_e^k in a vector z^k we obtain a recursion of the form

$$z^k = Az^{k-1} + b^{k-1}. \tag{7.7}$$

Assuming that $b^k = 0$ for all k, which means there are no sources and sinks and a zero inflow boundary condition, as well as a divergence free velocity field $\nabla\cdot\beta = 0$, we would like to ensure the following two properties of the numerical solution:

i) Positivity: if $z^{k-1} \geq 0$ (understood element-wise) then also $z^k \geq 0$.

ii) Boundedness: $\|z^k\|_\infty \leq M$ for some constant M and all times k.

Considering the signs on the right hand side of (7.6) the first term is critical. Positivity can only be ensured if

$$\forall e : \Delta t_k \sum_{\substack{\gamma \in \mathcal{F}_h^e \\ \beta \cdot \nu_e > 0}} \frac{(\beta \cdot \nu_e)|\gamma|}{|T|} \leq 1 \Leftrightarrow \Delta t_k \leq \left(\sum_{\substack{\gamma \in \mathcal{F}_h^e \\ \beta \cdot \nu_e > 0}} \frac{(\beta \cdot \nu_e)|\gamma|}{|T|} \right)^{-1}. \tag{7.8}$$

It can also be shown that the same condition, provided $\nabla \cdot \beta = 0$, ensures that $\|A\|_\infty = 1$ and therefore the solution stays bounded:

$$\|z^k\|_\infty \leq \|A\|_\infty \|z^0\|_\infty = \|z^0\|_\infty.$$

For a structured, equidistant mesh $|\gamma| = h^{n-1}$, $|e| = h^n$ and (7.8) is equivalent to

$$\Delta t_k \leq \frac{h}{\sum_{\substack{\gamma \in \mathcal{F}_h^e \\ \beta \cdot \nu_e > 0}} (\beta \cdot \nu_e)}$$

which is the famous *Courant-Friedrich-Levy* or CFL condition.

Implicit case

The implicit scheme results in a linear system

$$Lz^k = z^{k-1} + b^k$$

for each time step. L is a so-called M-matrix with $\|L^{-1}\|_\infty \leq 1$ for *any* Δt_k provided $\nabla \cdot \beta = 0$. Assuming again $b^k = 0$ the numerical solution is unconditionally stable.

Conclusion on Stability

Above we only covered the stability of the finite volume scheme using the upwind flux. Considering also the central flux with either explicit or implicit time stepping the following can be observed:

	upwind flux	central flux
explicit	stable under CFL	unconditionally unstable
implicit	unconditionally stable	stable if Δt large enough

7.4. Accuracy and Numerical Diffusion

Here we consider the accuracy of the finite volume upwind scheme by determining the local consistency order using Taylor expansion. We restrict the analysis to one spatial dimension $n = 1$, uniform time stepping $\Delta t_k = \Delta t$, uniform mesh size $|T_i| = h$, constant velocity $\mathbb{R} \ni \beta > 0$, no sources and sinks $f = 0$ and zero Dirichlet boundary condition $g = 0$.

Under these conditions the upwind finite volume scheme reads (note $u_h(x_i, t_k) = z_i^k$):

$$\frac{z_i^k - z_i^{k-1}}{\Delta t} + \beta \frac{z_i^{k-1} - z_{i-1}^{k-1}}{h} = 0 \qquad \text{(explicit scheme)}, \tag{7.9}$$

$$\frac{z_i^k - z_i^{k-1}}{\Delta t} + \beta \frac{z_i^k - z_{i-1}^{k-1}}{h} = 0 \qquad \text{(implicit scheme)}. \tag{7.10}$$

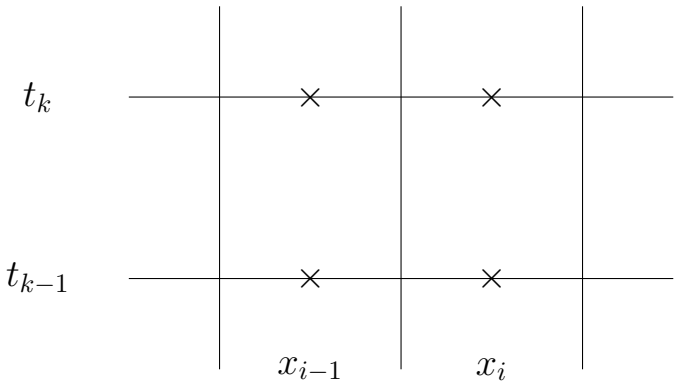


Figure 7.1.: Spatial and temporal positions in the Taylor expansion.

Defining the cell centers x_i and time steps t_k as shown in Figure 7.1, Taylor expansion yields:

$$\frac{u(x_i, t_k) - u(x_i, t_{k-1})}{\Delta t} = \frac{\partial u}{\partial t}(x_i, t_{k-1}) + \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}(x_i, t_{k-1}) + \mathcal{O}(\Delta t^2),$$

$$\frac{u(x_i, t_k) - u(x_{i-1}, t_{k-1})}{h} = \frac{\partial u}{\partial x}(x_i, t_{k-1}) - \frac{h}{2}\frac{\partial^2 u}{\partial x^2}(x_i, t_{k-1}) + \mathcal{O}(h^2),$$

where we assume enough smoothness of the solution and kept the first term of the error expansion.

Differentiating the PDE $\partial_t u + \beta \partial_x u = 0$ with repect to t as well as x yields

$$\left.\begin{aligned} \partial_t^2 u + \beta \partial_t \partial_x u &= 0 \\ \partial_t \partial_x u + \beta \partial_x^2 u &= 0 \end{aligned}\right\} \Rightarrow \partial_t^2 u - \beta^2 \partial_x^2 u = 0 \Leftrightarrow \boxed{\partial_t^2 u = \beta^2 \partial_x^2 u}$$

Inserting the true (smooth) solution into (7.9) and using the relation between spatial and temporal derivative yields:

$$\begin{aligned}
&\frac{u(x_i,t_k)-u(x_i,t_{k-1})}{\Delta t}+\beta\frac{u(x_i,t_k)-u(x_{i-1},t_{k-1})}{\Delta x}\\
&\quad=\left(\frac{\partial u}{\partial t}+\beta\frac{\partial u}{\partial x}\right)|_{x_i,t_{k-1}}+\left(\frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}-\frac{h}{2}\beta\frac{\partial^2 u}{\partial x^2}\right)|_{x_i,t_{k-1}}+\mathcal{O}(\Delta t^2+h^2)\\
&\quad=\left(\frac{\partial u}{\partial t}+\beta\frac{\partial u}{\partial x}\right)|_{x_i,t_{k-1}}\underbrace{-\left(\frac{\beta h-\beta^2\Delta t}{2}\right)\frac{\partial^2 u}{\partial x^2}|_{x_i,t_{k-1}}}_{\substack{\text{added diffusion term}\\ \text{"numerical diffusion"}}}+\mathcal{O}(\Delta t^2+h^2)
\end{aligned}$$

As a consequence, we may interpret the explicit upwind finite volume scheme (7.9) as a second-order accurate discretization of the modified PDE $\partial_t u+\beta\partial_x u-((\beta h-\beta^2\Delta t)/2)\partial_x^2 u$ which has an additional diffusion term. This also explains the diffusive nature of the numerical results using the upwind scheme: The error of the scheme appears like adding "numerical diffusion".

Observe also that the maximum stable time step is given by $\frac{\Delta t}{h}\beta=1\Rightarrow\Delta t=\frac{h}{\beta}$. Then the numerical diffusion coefficient becomes $\frac{\beta h-\beta^2\Delta t}{2}=\frac{\beta h-\beta h}{2}=0$. The scheme becomes exact in this case. However, this effect is restricted to one spatial dimension, equidistant meshes and constant velocity.

The same analysis for the implicit finite volume upwind scheme yields

$$\begin{aligned}
&\frac{u(x_i,t_k)-u(x_i,t_{k-1})}{\Delta t}+\beta\frac{u(x_i,t_k)-u(x_{i-1},t_{k-1})}{\Delta x}\\
&\quad=\left(\frac{\partial u}{\partial t}+\beta\frac{\partial u}{\partial x}\right)|_{x_i,t_{k-1}}-\left(\frac{\beta h+\beta^2\Delta t}{2}\right)\frac{\partial^2 u}{\partial x^2}|_{x_i,t_{k-1}}+\mathcal{O}(\Delta t^2+h^2).
\end{aligned}$$

Note that here in contrast the numerical diffusion coefficient is $\frac{\beta h+\beta^2\Delta t}{2}$ and the temporal error does not compensate the spatial error. In the implicit upwind scheme, both, the temporal and the spatial error act like a diffusion term. This also explains the (conditional) stability of the central flux when using implicit time stepping: when the time step is *large enough* the scheme becomes stable.

7.5. Numerical Results

Figure 7.2 gives numerical results for the proposed schemes. The domain is $\Omega=(0,1)^2$ and the velocity field constant at an angle 30° with $\|\beta\|=1$. The initial condition is discontinuous and is shown in the top row of images. At the left boundary (part of the the inflow) a Dirichlet condition is prescribed. The mesh is quadrilateral with an equidistant size $h=1/100$.

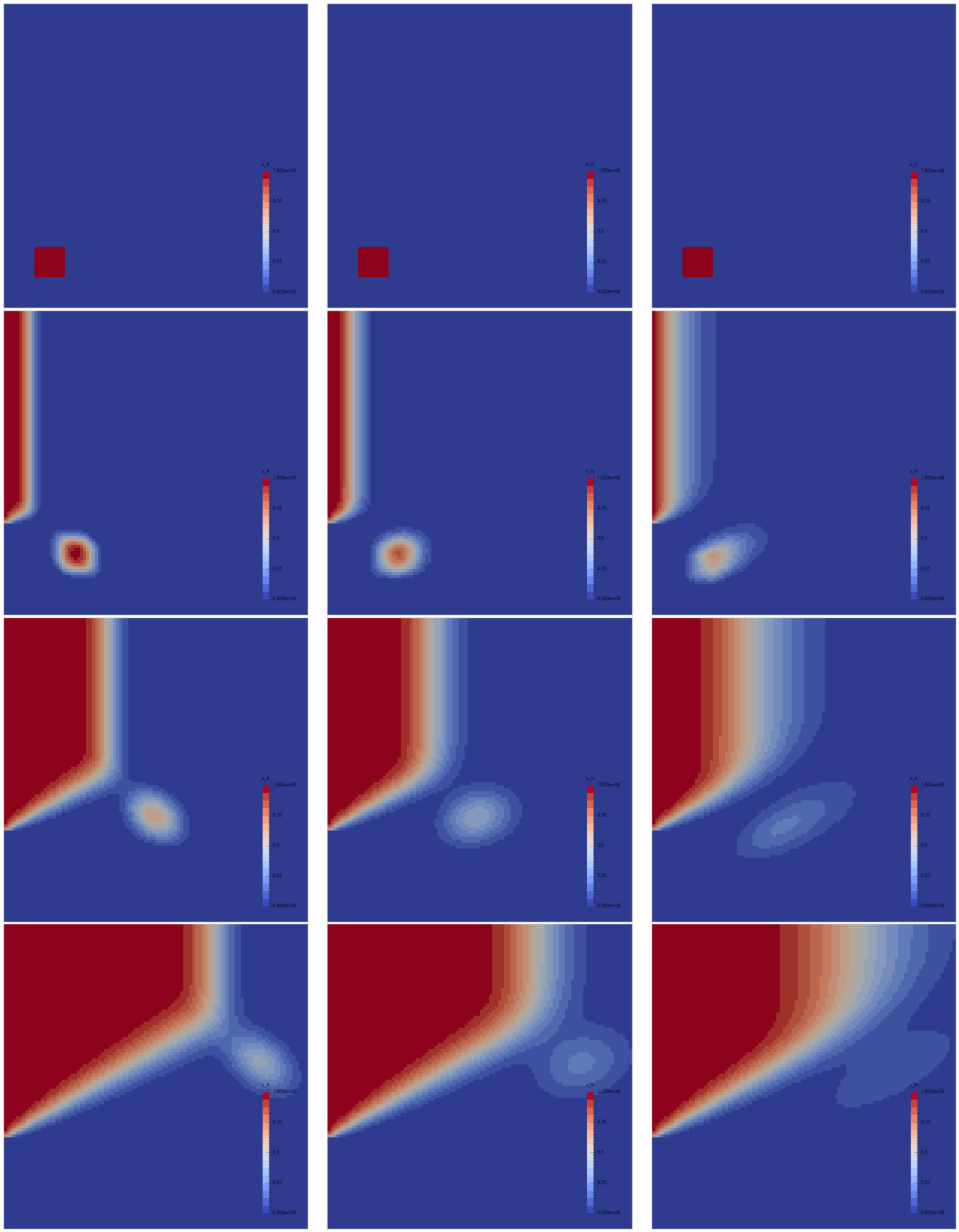


Figure 7.2.: Results for a model problem with discontinuous initial condition, $\beta = (\cos(\pi 30/180), \sin(\pi 30/180))^T$, $h = 1/100$. First column: explicit scheme with $\Delta t = 1/200$, runtime 2.8s, middle column: implicit scheme with $\Delta t = 1/200$, runtime 6.3s right column: implicit scheme with $\Delta t = 1/20$, runtime 0.9s.

The left column shows the explicit upwind finite volume scheme operating close to the stability limit at $\Delta t = 1/200$. The middle column shows the implicit upwind finite volume scheme at the same time step while the right column shows the implicit scheme operating at the much larger time step $\Delta t = 1/20$.

The true solution is discontinuous with the initial condition just moving to right and up while a wedge-formed by the boundary condition comes in from the left. All schemes show an excessive smearing of the front with more smearing exhibited by the implicit scheme at the same time step. The implicit scheme is unconditionally stable but the smearing is very pronounced at the large time step.

Chapter 8.

High-order Schemes for Linear Transport

Again, we are interested in solving the PDE

$$\partial_t u + \nabla \cdot (\beta u) = f \qquad \text{in } \Omega \times \Sigma \tag{8.1a}$$
$$u(x,0) = u_0(x) \qquad \text{at } t = 0,\ x \in \Omega \tag{8.1b}$$
$$u(x,t) = g(x,t) \qquad \text{on } \Gamma^- = \{x \in \partial\Omega : \beta(x) \cdot \nu(x) < 0\} \tag{8.1c}$$

8.1. Discontinuous Galerkin Space Discretization

In the previous chapter, we used simultaneous discretization of space and time. Here, we first discretize in space and leave time continuous, which results in a system of ODEs, which we then discretize in time. This approach is called "method of lines".

It is also possible to discretize first in time, then in space, an approach called "Rothe's method".

In order to obtain a higher order method, we use piecewise polynomials:

$$W_h^p = \{v \in L^2(\Omega) : v|_e \in \mathbb{V}_p \ \forall e \in \mathcal{T}_h\},$$

where $\mathbb{V}_p = \mathbb{P}_p$ (simplices) or $\mathbb{V}_p = \mathbb{Q}_p$ (cubes).

The first step of the method of lines is obtained as usual: Multiply the PDE with a test function, integrate over the domain and use (element-wise) integration by parts:

$$\begin{aligned}\int_\Omega \left[\partial_t u + \nabla \cdot (\beta u)\right] v dx &= \sum_{e \in \mathcal{T}_h} \left[\partial_t \int_e uv dx - \int_e u\beta \cdot \nabla v dx + \int_{\partial e} \beta u \cdot \nu v ds\right] \\ &= \sum_{e \in \mathcal{T}_h} \left[\partial_t \int_e uv dx - \int_e u\beta \cdot \nabla v dx\right] + \sum_{\gamma \in \mathcal{F}_h^i} \int_\gamma [\![\beta \cdot \nu_\gamma uv]\!] ds \\ &\quad + \sum_{\gamma \in \mathcal{F}_h^+} \int_\gamma (\beta \cdot \nu_\gamma) uv ds + \sum_{\gamma \in \mathcal{F}_h^-} \int_\gamma (\beta \cdot \nu_\gamma) gv ds.\end{aligned}$$

The numerical scheme is now obtained by

- choosing $u_h(.,t) \in W_h^P$ and
- employing a numerical flux function, e.g. the upwind flux $\Phi_u(\beta\cdot\nu_\gamma, u^-, u^+)$.

This results in the scheme

$$\begin{aligned}\sum_{e\in\mathcal{T}_h}\left[\partial_t\int_e u_h v dx - \int_e u_h\beta\cdot\nabla v dx\right] + \sum_{\gamma\in\mathcal{F}_h^i}\int_\gamma \Phi_u(\beta\cdot\nu_\gamma, u_h^-, u_h^+)[\![v]\!]ds \\ + \sum_{\gamma\in\mathcal{F}_h^+}\int_\gamma(\beta\cdot\nu_\gamma)u_h v ds = \sum_{e\in\mathcal{T}_h}\int_e fv dx - \sum_{\gamma\in\mathcal{F}_h^-}\int_\gamma(\beta\cdot\nu_\gamma)gv ds\ \forall v\in W_h^p.\end{aligned} \tag{8.2}$$

For the practical realization choose a basis $W_h^p = \text{span}\{\varphi_{e,i} : e\in\mathcal{T}_h, 1\le i\le \dim\mathbb{V}_p =: P\}$. These basis functions are local to each element, i.e. $\varphi_{e,i}\cdot\varphi_{e',j} = 0\ \forall e\neq e'$. As an example consider $\mathbb{P}_1$ in 2d:

$$\varphi_{e,1}(x) = \begin{cases} 1 & x\in T \\ 0 & \text{else}\end{cases}, \varphi_{e,2}(x) = \begin{cases} x_1 & x\in T \\ 0 & \text{else}\end{cases}, \varphi_{e,3}(x) = \begin{cases} x_2 & x\in T \\ 0 & \text{else}\end{cases}.$$

The ansatz $u_h(x,t) = \sum_{e'\in\mathcal{T}_h}\sum_{j=1}^P z_{e',j}(t)\varphi_{e',j}(x)$ using the test function $v=\varphi_{e,i}$ and employing the locality of the basis results in the linear equations

$$\begin{aligned}&\sum_{j=1}^P\left[\frac{dz_{e,j}}{dt}\int_e\varphi_{e,i}\varphi_{e,j}dx - z_{T,j}\int_e\varphi_{e,j}\beta\cdot\nabla\varphi_{e,j}dx\right] \\ &\quad+\sum_{j=1}^P z_{e,j}\sum_{\substack{\gamma\in\mathcal{F}_h^e\\ \beta\cdot\nu_e>0}}\int_\gamma(\beta\cdot\nu_e)\varphi_{e,j}\varphi_{e,i}ds \\ &\quad+\sum_{j=1}^P z_{e,j}\sum_{\substack{\gamma\in\mathcal{F}_h^e\\ \beta\cdot\nu_e<0}} z_{nb(e,\gamma),j}\int_\gamma(\beta\cdot\nu_e)\varphi_{nb(e,\gamma),j}\varphi_{e,i}ds \\ &\qquad= \int_e f\varphi_{e,i}dx - \sum_{\substack{\gamma\in\mathcal{F}_h^e\\ \beta\cdot\nu_e<0}}\int_\gamma(\beta\cdot\nu_e)g\varphi_{e,i}ds \quad \forall e\in\mathcal{T}_h,\ 1\le i\le P.\end{aligned} \tag{8.3}$$

Here we used the same notation as for the finite volume upwind scheme.

8.2. Time Discretization

Equation (8.3) comprises a system of (linear) ordinary differential equations for the unknown coefficient functions $z_{e,j}(t)$ which can be written in vector form as

$$\frac{dz(t)}{dt} = L_h(t, z(t)) = A(t)z(t) + b(t). \tag{8.4}$$

This system may be solved by explicit Runge-Kutta methods which read in Shu-Osher form:

$$\begin{aligned} z^{(0)} &= z^{k-1} \\ z^{(i)} &= \sum_{j=0}^{i-1} \alpha_{ij} z^{(j)} + \beta_{ij} \Delta t_k L_h(t_{k-1} + d_j \Delta t_k, z^{(j)}) \quad i = 1...s \\ z^k &= z^{(s)} \end{aligned} \tag{8.5}$$

Examples for such schemes are:

	α_{ij}	β_{ij}	d_i
explicit Euler $s = 1$, order 1	1	1	0
Heun $s = 2$, order 2 SSPRK2	$\begin{array}{cc} 1 & \\ \frac{1}{2} & \frac{1}{2} \end{array}$	$\begin{array}{cc} 1 & \\ 0 & \frac{1}{2} \end{array}$	$\begin{array}{c} 0 \\ 1 \end{array}$
SSPRK3 $s = 2$, order 2	$\begin{array}{ccc} 1 & & \\ \frac{3}{4} & \frac{1}{4} & \\ \frac{1}{3} & 0 & \frac{2}{3} \end{array}$	$\begin{array}{ccc} 1 & & \\ 0 & \frac{1}{4} & \\ 0 & 0 & \frac{2}{3} \end{array}$	$\begin{array}{c} 0 \\ 1 \\ \frac{1}{2} \end{array}$

Employing these schemes, the following results can be proven for the fully discrete scheme:

Theorem 8.1. Assume $u \in C^0([0, t_F]; H^1(\Omega)) \cap C^2([0, t_F]; L^2(\Omega))$, polynomial degree 0 (piecewise constants), explicit Euler and a uniform space time mesh. Then the discontiuous Galerkin upwind scheme (which is equivalent to the finite volume scheme) converges with $\mathcal{O}(\Delta t + h^{1/2})$ provided $\Delta t \leq \rho^{Eul} \frac{h}{\|\beta\|}$.

Die Pietro, Ern, Thm 3.7. . □

Here, the following notation is used in the regularity assumption. A space-time function $\Psi : \Omega \times \Sigma \to \mathbb{R}$ can also be interpreted in two steps. First, map from time interval to a space of functions living in space only $\Psi : \Sigma \to V$ where $V = \{u : \Omega \to \mathbb{R}\}$ and which satisfy $\Psi(t)(x) = \Psi(x, t)$.

Then $C^l(\Sigma; V)$ is the space of functions $\Psi : \Sigma \to V$ being l times continuously differentiable (derivative defined as $\frac{d\Psi(t)}{dt} = \lim_{\Delta t \to 0} = \frac{\Psi(t+\Delta t) - \Psi(t)}{\Delta t}$) "with values in V".

Concerning Theorem 8.1 we remark:

- For a finite difference scheme we expect convergence $\mathcal{O}(\Delta t + h)$. But this requires $u(t) \in C^2(\Omega)$. The theorem here requires less regularity.
- Using the stability limit $\Delta t \sim h$ the convergence is of order $\mathcal{O}(h^{1/2})$.

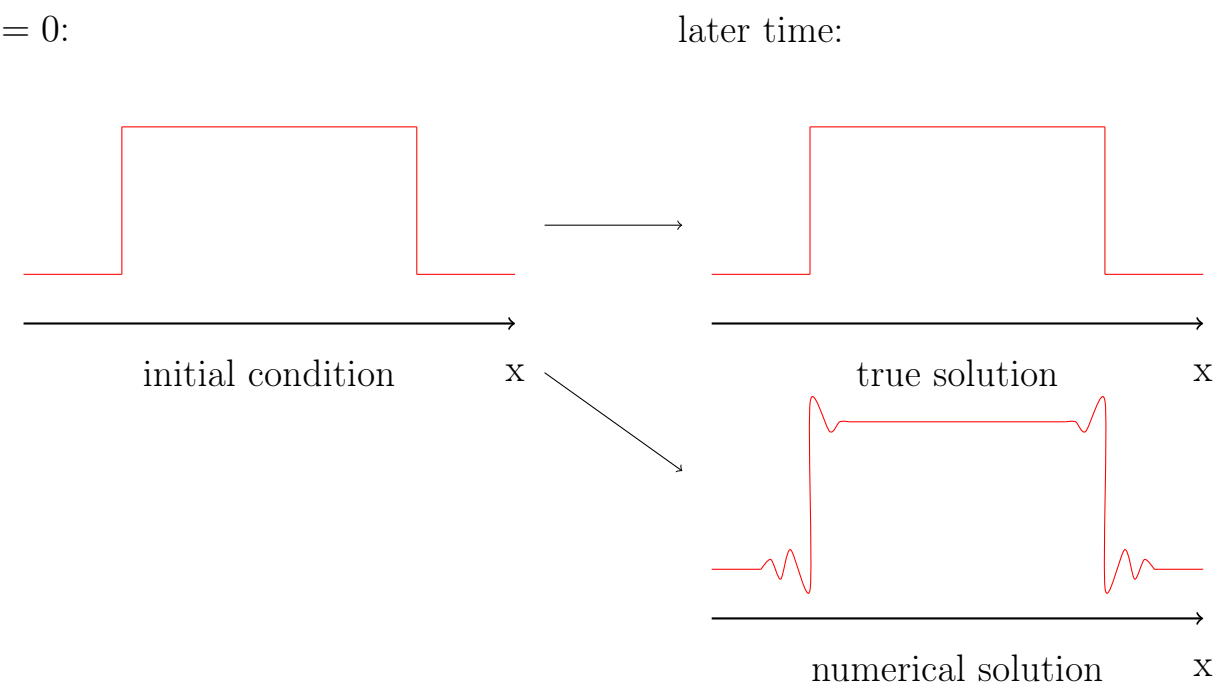


Figure 8.1.: Oscillatory behavior of higher-order schemes.

The following Theorem treats the case $p > 0$.

Theorem 8.2. Assume $u \in C^3(\Sigma; L^2(\Omega)) \cap C^0(\Sigma; H^{k+1}(\Omega))$, $d_t u \in C^0(\Sigma; H^k(\Omega))$, polynomial degree $p \geq 1$, $f \in C^2(L^2(\Omega))$. As time step restriction require

- Case $p = 1$: $\Delta t \leq \rho^{RK2} \frac{h}{\|\beta\|}$. $\rightarrow \Delta t \sim h$
- Case $p \geq 2$: $\Delta t \leq \rho' \left(\frac{h}{\|\beta\|}\right)^{4/3}$. $\rightarrow \Delta t \sim h^{4/3} = h \underbrace{h^{1/3}}_{\rightarrow 0 \text{ for } h \rightarrow 0}$
  ($\rightarrow \Delta t$ must go to zero at a faster rate).

Then the SSPRK2 scheme (Heun) converges with rate $\mathcal{O}(\Delta t^2 + h^{p+1/2})$.

Die Pietro, Ern, Thm 3.10. . □

As presented, these schemes with $p \geq 1$ may produce solutions with unphysical oscillations near sharp fronts with the typical behavior shown in Figure 8.1. In fact, this behavior is universal, as stated by the following Theorem.

Theorem 8.3 (Godunov)**.** Linear schemes without unphysical oscillations are restricted to first order accuracy in space.

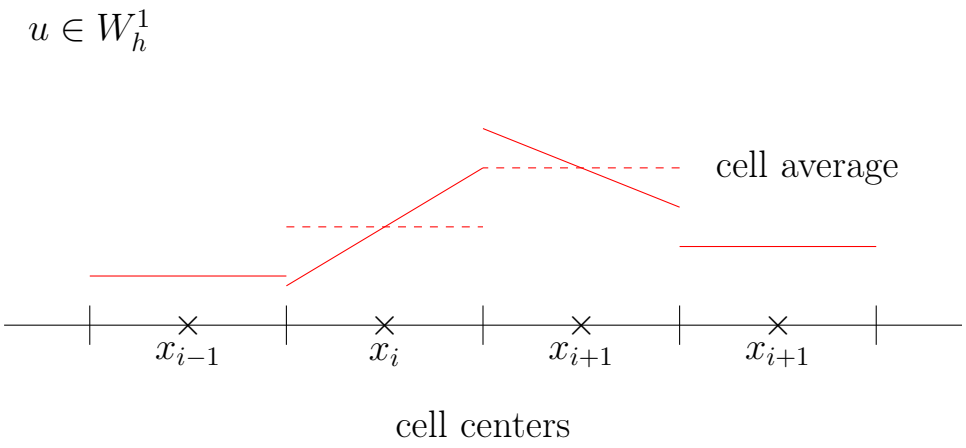


Figure 8.2.: A piecewise linear function.

8.3. Slope Limiting

Avoiding unphysical oscillations requires non-linear schemes. Here, they have the form:

$$\begin{aligned} z^{(0)} &= z^{k-1}, \\ z^{(i)} &= \Lambda\left(\sum_{j=0}^{i-1} \alpha_{ij} z^{(j)} + \beta_{ij} \Delta t_k L_h(t_{k-1} + d_j \Delta t_k, z^{(j)})\right) \qquad i = 1...s, \\ z^k &= z^{(s)}, \end{aligned}$$

with Λ a so-called slope limiter.

To illustrate the concept of slope limiting consider one space dimension, equidistant mesh size h and polynomial degree $p = 1$. A function $u_h \in W_h^1$ can be described in each cell i by: $u_i(x) = \bar{u}_i + \sigma_i(x - x_i)$, where $\bar{u}_i$ is the cell average, σ_i is the slope and x_i is the cell center, see Figure 8.2.

The limited function $u_h' = \Lambda(u_h)$ is also defined defined per cell by $u_i'(x) = \bar{u}_i + \sigma_i'(x - x_i)$ (the cell average is the same due to mass conservation; the slope changes) and the slope given by

$$\sigma_i' = \text{minmod}\left(\sigma_i, \frac{u_{i+1} - u_i}{h}, \frac{u_{i+1} - u_i}{h}\right)$$

with

$$\text{minmod}(a, b, c) = \begin{cases} \alpha \min(|a_1|, |a_2|, |a_3|) & \text{if } \alpha := \text{sgn}(a_1) = \text{sgn}(a_2) = \text{sgn}(a_3) \\ 0 & \text{else} \end{cases}$$

Extensions to higher space dimensions are heuristic.

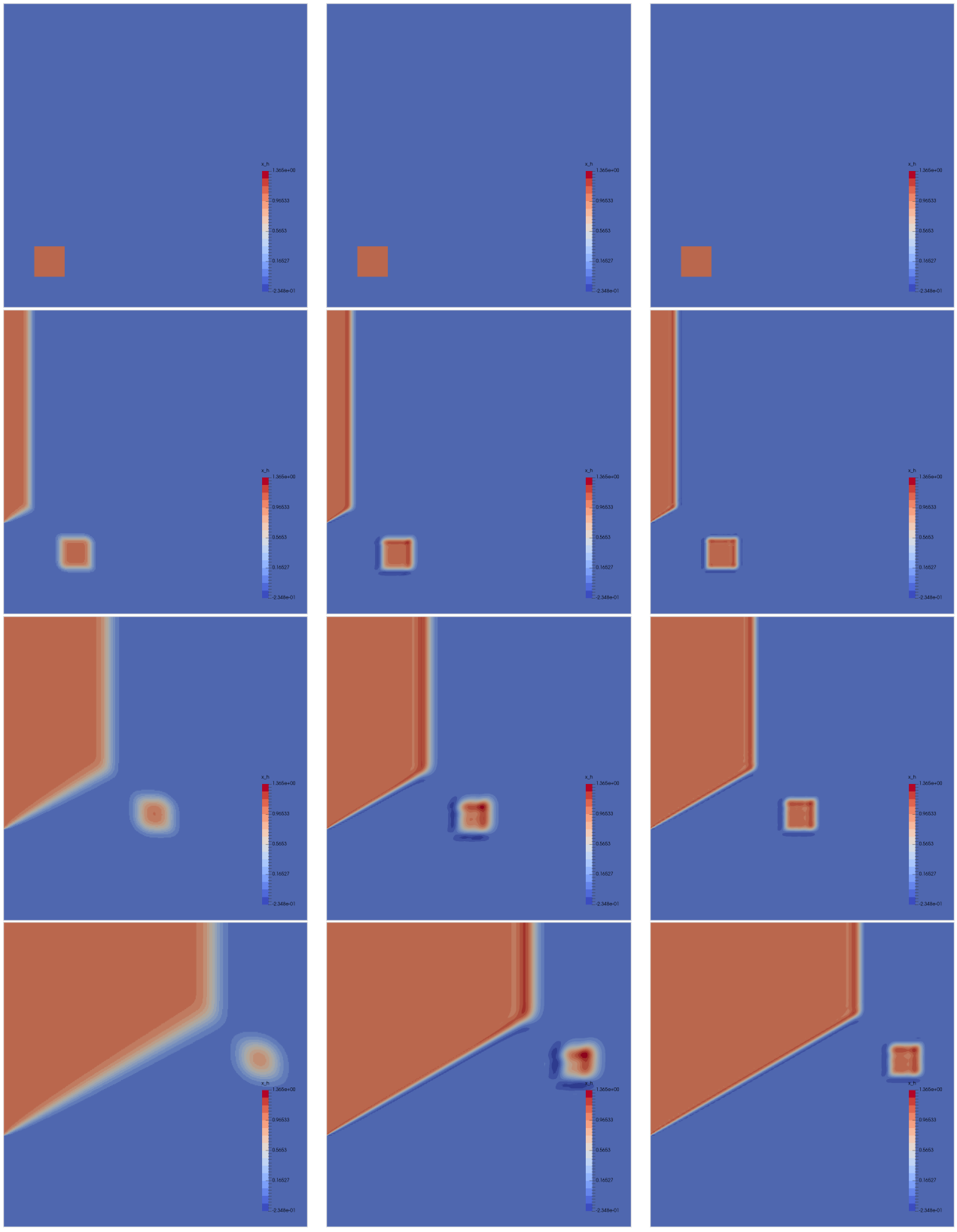


Figure 8.3.: Results for a model problem with discontinuous initial condition using explicit higher order methods, $\beta = (\cos(\pi 30/180), \sin(\pi 30/180))^T$. First column: order 1, $h = 1/400$, runtime 160s, middle column: order 2, $h = 1/200$, runtime 55s, right column: order 3, $h = 1/100$, $\Delta t = 1/400$, runtime 34s.

8.4. Numerical Results

Figure 8.3 shows results for the same problem treated in Section **??** using explicit time discretization. We compare formally first, second and third order accurate schemes using different mesh as well as time step sizes which lead approximately to the same total computation time. Clearly the higher order schemes outperform the lower order schemes but all higher order schemes exhibit unphysical oscillations. We do not treat limiter methods here to enforce a maximim principle. These methods remove a lot of the sharpness of the higher order schemes close to the discontinuity. If the unphysical oscillations do not hurt, just accept them. Of course, when solving nonlinear problems limiters are needed to converge towards to enforce a selection principle.

Figure 8.4 shows results for using implicit time discretizations. We compare increasing the order (in space and time) while keeping the spatial and temporal mesh size constant. The results illustrate that the schemes are stable and the accuracy improves as the order is increased. However, the same comment applies with respect to unphysical oscillations.

Implicit versus Explicit

The question of explicit vs. implicit is not easy to answer. In general, explicit methods are to be prefered with hyperbolic problems as the ordinary differential equations arising after semi-discretization are typically not stiff. However they might become so when the data varies highly, such as the velocity magnitude in a porous medium flow problem with wells. Another application of implicit methods is with quasi-stationary problems.

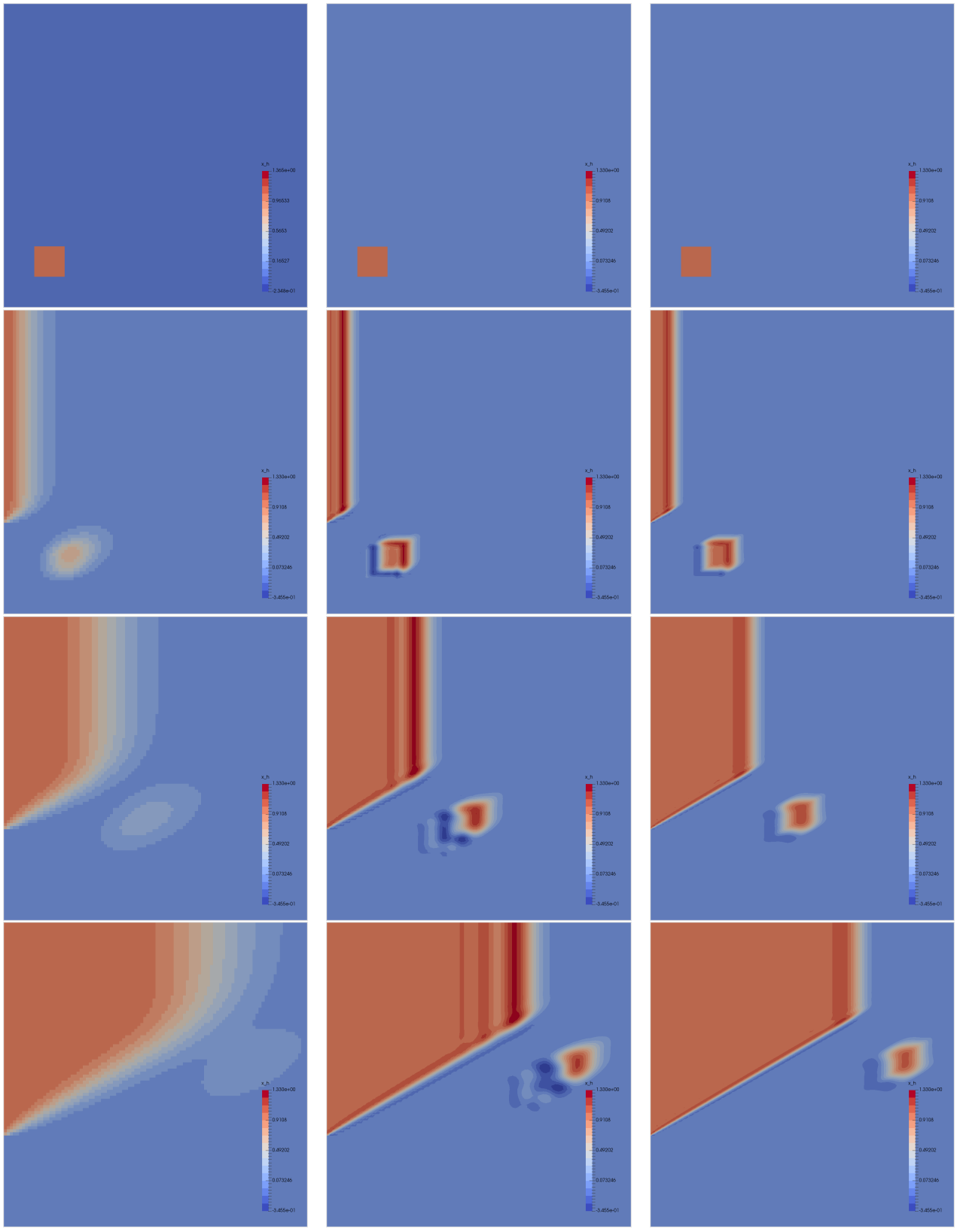


Figure 8.4.: Results for a model problem with discontinuous initial condition using implicit higher order methods using $h = 1/100$, $\Delta t = 1/40$, $\beta = (\cos(\pi 30/180), \sin(\pi 30/180))^T$, $h = 1/100$. First column: order 1, runtime 1s, middle column: order 2, runtime 10s, right column: order 3, runtime 53s.

Chapter 9.

Numerical Solution of Parabolic Equations

9.1. Discussion of the Model

Now we consider convective and diffusive transport and solve the parabolic equation

$$\begin{aligned}
\partial_t u + \nabla \cdot (\beta u - D\nabla u) &= f && \text{in } \Omega \times \Sigma && (9.1a)\\
u(x,0) &= u_0(x) && t = 0 && (9.1b)\\
u(x,t) &= g(x,t) && x \in \Gamma_D(t) && (9.1c)\\
(\beta u - D\nabla u)\cdot \nu &= j(x,t) && x \in \Gamma_N(t) && (9.1d)\\
(D\nabla u)\cdot \nu &= o(x,t) && x \in \Gamma_O(t) \subset \Gamma^+ && (9.1e)
\end{aligned}$$

where we seek the concentration of the dissolved quantity $u : \Omega \times \Sigma \to \mathbb{R}$. The second line gives the initial condition and we apply three different types of boundary conditions: Dirichlet, flux (Neumann) and outflow conditions where we require $\Gamma_D(t) \subset \Gamma^- = \{x \in \partial\Omega : \beta \cdot \nu < 0\}$ and $\Gamma_O(t) \subset \Gamma^+ = \{x \in \partial\Omega : \beta \cdot \nu > 0\}$.

Observe that there is a type change in the equation:

- For $D = 0$ the PDE is first-order hyperbolic. Boundary conditions can only be posed on the inflow boundary Γ^-.
- For $D > 0$ (in the sense that $x^T Dx > 0$) the PDE is parabolic and boundary conditions on all of the boundary $\Gamma = \partial\Omega$ are necessary.
- There is no continuous transition from $D > 0$ to $D = 0$ due to the type change. It turns out that numerics gets more difficult for $D \to 0$.

Boundary Layers

To illustrate the problems when $D \to 0$ we consider the following simplified (stationary) problem in one space dimension:

$$\begin{aligned}
u' - \varepsilon u'' &= 0 && \text{in } \Omega = (0,1),\\
u(0) = 1, u(1) &= 0 && \text{boundary condition,}
\end{aligned}$$

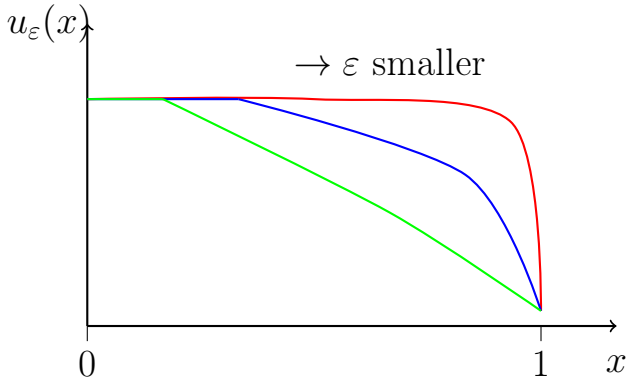


Figure 9.1.: Illustration of boundary layers for various values of ε. Green is a rather large value and red is a rather small value

with diffusion coefficient $\varepsilon > 0$. This problem has the following exact solution

$$u_\varepsilon(x) = \frac{1 - \exp(\frac{1-x}{\varepsilon})}{1 - \exp(\frac{-1}{\varepsilon})}.$$

The solution is illustrated for various values of ε in Figure 9.1.

Now we can solve for the width of the boundary at concentration $u = 1/2$ in dependence of ε:

$$\begin{aligned} u_\varepsilon(x) = \frac{1}{2} &\Leftrightarrow \frac{1 - \exp(\frac{1-x}{\varepsilon})}{1 - \exp(\frac{-1}{\varepsilon})} = \frac{1}{2} \\ &\Leftrightarrow \exp\left(\frac{1-x}{\varepsilon}\right) = 1 - \frac{1}{2}\left(1 - \exp\left(\frac{-1}{\varepsilon}\right)\right) = \frac{1}{2}\left(1 + \exp\left(\frac{-1}{\varepsilon}\right)\right) \\ &\Leftrightarrow x = 1 - \varepsilon \ln\left(\frac{1}{2}\left(1 + \exp\left(\frac{-1}{\varepsilon}\right)\right)\right) \approx 1 - \varepsilon \ln\frac{1}{2}. \end{aligned}$$

We conclude that the boundary layer has a width proportional to ϵ.

Convection versus Diffusion

Consider the stationary equation in n dimensions. We want to consider the question whether convection or diffusion is dominating the system.

Assume $\beta \in \mathbb{R}^n$, $D = \varepsilon\mathbb{I}$ and $\Omega = (0, L)^n$ and consider the PDE in non-conservative form:

$$\beta\nabla u - \varepsilon\Delta u = 0 \qquad \text{in } \Omega.$$

To answer the question, above we transfer the equation to a domain of unit size by introducing the transformation $\hat{u}(\hat{x}) := u(\hat{x}L)$. Then, by employing the

chain rule we get

$$\partial_{\hat{x}_i}\hat{u}(\hat{x}) = \partial_{\hat{x}_i}u(\hat{x}L)L, \qquad \partial^2_{\hat{x}_i}\hat{u}(\hat{x}) = \partial^2_{\hat{x}_i}u(\hat{x}L)L^2,$$

which we can resolve for the derivative with respect to the original variables

$$\partial_{x_i}u(\hat{x}L) = L^{-1}\partial_{\hat{x}_i}\hat{u}(\hat{x}), \qquad \partial^2_{x_i}u(\hat{x}L) = L^{-2}\partial^2_{\hat{x}_i}\hat{u}(\hat{x}).$$

Inserting these expressions into the PDE results in the transformed PDE in the unit domain

$$\beta L^{-1}\hat{\nabla}\hat{u}(\hat{x}) - \varepsilon L^{-2}\hat{\Delta}\hat{u}(\hat{x}) = 0 \qquad \text{in } \hat{\Omega} = (0,1)^n.$$

Now consider the ratio of convection vs. diffusion term and name this ratio the Peclet number:

$$\frac{\|\beta\|L^{-1}}{\varepsilon L^{-2}} = \frac{L\|\beta\|}{\varepsilon} =: \mathrm{Pe} \qquad \text{(Peclet-number)}. \tag{9.2}$$

The Peclet number is dimensionless since β has units ms^{-1} and ε has units m^2s^{-1}. We say that convection dominates if $\mathrm{Pe} \gg 1$, diffusion dominates if $\mathrm{Pe} \ll 1$ and both processes are about the same when Pe is around one.

9.2. Numerics

Now we combine all our knowledge from the previous chapters and proceed in the same way, i.e.:

- Use method of lines.
- Employ the cell-centered FV-method for spatial discretization.

The spatial discretization follows the same recipe. Multiply with a cell-wise constant test function, integrate over the domain, employ cell-wise integration by parts and insert numerical fluxes:

$$\begin{aligned}
\int_\Omega \big[\partial_t u_h + \nabla(\beta u_h - D\nabla u_h)\big]v dx &= \sum_{e\in\mathcal{T}_h}\left[\partial_t \int_T u_h v dx + \int_{\partial_e}(\beta u_h - D\nabla u_h)\nu v ds\right] = \\
&\partial_t \int_\Omega u_h v dx + \sum_{\gamma\in\mathcal{F}_h^i}\int_\gamma \left[\Phi(\beta\cdot\nu_\gamma, u_h^-, u_h^+) - D_{eff}(\gamma)\frac{u_h(x^+(\gamma)) - u_h(x^-(\gamma))}{\|x^+(\gamma) - x^-(\gamma)\|}\right][\![v]\!] ds \\
&\quad + \sum_{\gamma\in\mathcal{F}_h^D}\int_\gamma \left[\beta\cdot g - D(T^-(\gamma))\frac{g(x^(\gamma)) - u_h(x^-(\gamma))}{\|x^(\gamma) - x^-(\gamma)\|}\right] v ds \\
&\quad + \sum_{\gamma\in\mathcal{F}_h^N}\int_\gamma j v ds + \sum_{\gamma\in\mathcal{F}_h^O}\int_\gamma (\beta u_h + o) v ds \\
&= \int_\Omega f v dx
\end{aligned}$$

To make things more simple, let us consider a special case $n = 1$ (one spatial dimension), $\beta > 0$ (flow to the right), upwind flux $\Phi_u(\beta \cdot \nu, u^-, u^+) = \beta \cdot \nu u^-$, $f = 0$ (no sources and sinks). If we consider an interior cell e_i (i.e. we set the test function to $\varphi_{e_i} = \xi_{e_i}$) we get a linear system of ordinary differential equations for the coefficients $z_i(t)$:

$$d_t(z_i(t)h) + \left(\beta z_i(t) - D\frac{z_{i+1}(t) - z_i(t)}{h}\right) + \left(-\beta z_{i-1}(t) - D\frac{z_{i-1}(t) - z_i(t)}{h}\right) = 0,$$
$$\Leftrightarrow d_t z_i(t) - \left(\frac{\beta}{h} + \frac{D}{h^2}\right) z_{i-1}(t) + \left(\frac{\beta}{h} + \frac{2D}{h^2}\right) z_i(t) - \frac{D}{h^2} z_{i+1}(t) = 0.$$

Now do a time discretization on the equidistant grid in time $t_k = k\Delta t$ and employ the explicit Euler method using the dofference quotient $d_t z_i(t_k) \approx \frac{z_i^{k+1} - z_i^k}{\Delta t}$ to get

$$z_i^{k+1} = \frac{\Delta t}{h}\left(\beta + \frac{D}{h}\right) z_{i-1}^k + \left(1 - \frac{\Delta t}{h}\beta - \frac{\Delta t}{h^2}2D\right) z_i^k + \frac{\Delta t}{h}\frac{D}{h} z_{i+1}^k \qquad (9.3)$$

Again we derive a stability condition ensuring positivity of solutions by simply looking at the sign of the matrix coefficients. The offdiagonal terms already are positive. From the diagonal coefficient we get the condition:

$$1 - \frac{\Delta t}{h}\beta - \frac{\Delta t}{h^2}2D \geq 0 \;\Leftrightarrow\; \frac{\Delta t}{h}\left(\beta + \frac{2D}{h}\right) \leq 1$$

from which we conclude the following two extreme cases:

$$\begin{array}{ll} \beta > 0,\ D = 0 & \Delta t \leq \frac{h}{\beta} \quad \text{CFL condition} \\ \beta = 0,\ D > 0 & \Delta t \leq \frac{h^2}{2D} \quad \rightarrow \text{possibly very small timesteps} \end{array}.$$

The condition $\Delta t \sim h^2$ is very restrictive. On the one hand it is required also from an accuracy viewpoint because the error from the time discretization is $O(\Delta t)$ and the error from spatial discretization is $O(h^2)$ and in order to balance them we need to set $\Delta t \sim h^2$. On the other hand, one could use second-order explicit schemes and one would still obtain the same time step restriction which is then overly restrictive. In addition solutions of parabolic problems would decay exponentially in time (without sources and sinks), the temporal error reduces quickly in time and we schould be able to take large time steps. Therefore, implicit schemes are preferred for parabolic problems.

Next we consider the implicit Euler method in time and combine it with the central flux $\Phi_c(\beta \cdot \nu, u^-, u^+) = \beta \cdot \nu \frac{u^- + u^+}{2}$ for the one-dimensional model problem.

We obtain the discrete system

$$h\frac{z_i^{k+1}-z_i^k}{\Delta t}+\left(\beta\frac{z_i^{k+1}+z_{i+1}^{k+1}}{2}-D\frac{z_{i+1}^{k+1}-z_i^{k+1}}{h}\right)-\left(\beta\frac{z_{i-1}^{k+1}+z_i^{k+1}}{2}-D\frac{z_i^{k+1}-z_{i-1}^{k+1}}{h}\right)=0,$$

$$\Leftrightarrow \frac{\Delta t}{h}\underbrace{\left(-\frac{\beta}{2}-\frac{D}{h}\right)}_{<0 \text{ for } \beta>0} z_{i-1}^{k+1}+\underbrace{\left(1+\frac{\Delta t}{h^2}2D\right)}_{>0} z_i^{k+1}+\frac{\Delta t}{h}\underbrace{\left(\frac{\beta}{2}-\frac{D}{h}\right)}_{?} z_{i+1}^{k+1}=z_i^k.$$

In every time step of the implicit scheme we need to solve a linear system $Az^{k+1}=z^k$, where A is an M-matrix if

$$\frac{\Delta t}{h}\left(\frac{\beta}{2}-\frac{D}{h}\right)\leq 0 \Leftrightarrow \frac{\beta}{2}\leq\frac{D}{h}\Leftrightarrow \underbrace{\frac{h\beta}{D}}_{=:Pe_h}\leq 2. \tag{9.4}$$

The M-matrix property ensures that all entries of the inverse A^{-1} are nonnegative. Therefore, with the implicit scheme we can even use the second-order central flux if the grid Peclet number is small enough and obtain an unconditionally stable scheme.

Let us revisit the stability condition of the explicit scheme. For large enough grid Peclet number

$$Pe_h=\frac{h\beta}{D}>2 \;\Leftrightarrow\; \frac{h}{D}>\frac{2}{\beta}$$

we find

$$\frac{h^2}{2D}=\frac{h}{D}\frac{h}{2}>\frac{2}{\beta}\frac{h}{2}=\frac{h}{\beta}$$

which means that the CFL condition $\Delta t\leq\frac{h}{\beta}$ also ensures that the time step restriction arising from the diffusion term is satisfied.

Conclusion

From the stability analysis we conclude which scheme should be used depending on the grid Peclet number:

$$\begin{aligned} Pe_h>2:&\ \text{use explicit upwind scheme,}\\ Pe_h\leq 2:&\ \text{use implicit central scheme.}\end{aligned}$$

9.3. Density Driven Flow

As an application we consider the following coupled problem:

$$-\nabla\cdot\left\{\frac{K}{\mu}\left(\nabla p-\rho(c)g\right)\right\}=0, \tag{9.5}$$

$$\partial_t c+(uc-D\nabla c)=0, \tag{9.6}$$

with appropriate boundary and initial conditions. Here the so-called Boussinesq approximation was employed which only invokes the dependence of density on concentration in the boyancy term.

The individual equations are linear, but the coupled system is non-linear. There are two basic approaches to solve this system:

a) *Fully coupled approach*:

- Solve both equations simultaneously.
- Employing discretization in space and time results now in a non-linear algebraic system of equations which solved iteratively, e.g. with Newton's method.

b) *Operator splitting method*:

- Given c^k, compute p^k and u^k by solving the groundwater flow problem.
- Given u^k, compute c^{k+1} by solving the transport equation.
- $\Delta t^{k+1} = t^{k+1} - t^k$ is the operator splitting time step.
- Iterate these steps.

Discussion:

- In the operator splitting scheme each subproblem is linear, thus no non-linear system needs to be solved.
- In the operator splitting appropriate discretization schemes (such as explicit or implicit methods) can be used for the individual steps.
- The accuracy in time of the operator splitting scheme is $\mathcal{O}(\Delta t)$. There are also second-order variants, but going higher-order is not done.
- The operator splitting time step must be small enough for stability. However, it might not be easy to detect when the time step was too large in practice.
- A variant is iterated operator splitting, which essentially is the fully coupled system solved with a block Gauß-Seidel method.

Chapter 10.

Stationary Stokes Equations

10.1. Strong Form of the Equations

We are interested in solving the incompressible Navier-Stokes equations

$$\partial_t u + \nabla \cdot (uu^T) - \eta \Delta u + \nabla p = f \qquad \text{in } \Omega \times \Sigma, \tag{10.1a}$$
$$\nabla \cdot u = 0 \qquad \text{in } \Omega \times \Sigma, \tag{10.1b}$$

where Ω is the spatial domain, Σ is a time interval, $\eta = \mu/\rho$ is the kinematic viscosity and $p = p_{orig}/\rho$ is the rescaled pressure. Note that in the literature usually the symbol ν is used for the kinematic viscosity. Since we use ν for the unit outer normal in these lecture notes we use η instead.

Further assuming that time derivative $\partial_t u$ and inertial terms $\nabla \cdot (uu^T)$ are small, we arrive at the stationary Stokes equations

$$-\Delta u + \nabla p = f \qquad \text{in } \Omega \times \Sigma, \tag{10.2a}$$
$$\nabla \cdot u = 0 \qquad \text{in } \Omega \times \Sigma, \tag{10.2b}$$

where we have rescaled pressure and right hand side by kinematic viscosity $p = p/\eta$, $f = f/\eta$. The stationary Stokes equations are supplemented with the following boundary conditions

$$u = g \qquad \text{on } \Gamma_D \subset \partial\Omega \qquad \text{(Dirichlet)}, \tag{10.3a}$$
$$\nabla u \nu - p\nu = s \qquad \text{on } \Gamma_N \subset \partial\Omega \setminus \Gamma_D \qquad \text{(natural)}. \tag{10.3b}$$

Observe also that $u : \Omega \to \mathbb{R}^n$ is a vector-valued function and therefore

$$\nabla u = \begin{pmatrix} \partial_1 u_1 & \dots & \partial_n u_1 \\ \vdots & \ddots & \vdots \\ \partial_1 u_n & \dots & \partial_n u_n \end{pmatrix}, \qquad \Delta u = \nabla \cdot \nabla u = \begin{pmatrix} \Delta u_1 \\ \vdots \\ \Delta u_n \end{pmatrix}.$$

Remark 10.1. Some remarks are in order here.

a) There are only n boundary conditions for $n+1$ equations and unknowns respectively. In particular for Dirichlet boundary conditions there is no boundary condition on the pressure. Giving a reason for this will be a major goal of this chapter.

b) If $\Gamma_D = \partial\Omega$ (Dirichlet only), then p is only determined up to a constant (in contrast to the Possion equation) and a compatibility condition on the data is required:

$$0 \overset{(10.2b)}{=} \int_\Omega \nabla \cdot u \, dx = \int_{\partial\Omega} u \cdot \nu \, ds = \int_{\partial\Omega} g \cdot \nu \, ds.$$

c) If $\text{meas}(\Gamma_N) \neq 0$ the pressure is determined uniquely. This can be motivated as follows. Integrating over the natural boundary condition (10.3) over Γ_N:

$$\int_{\Gamma_N} s \cdot \nu ds = \int_{\Gamma_N} [\nabla u \nu - p\nu] \cdot \nu ds \;\Leftrightarrow\; \int_{\Gamma_N} p ds = \int_{\Gamma_N} (\nabla u \nu) \cdot \nu - s \cdot \nu ds$$

shows that assuming u is unique, the average of p over Γ_N is now also fixed.

10.2. Weak Formulation

Existence and uniqueness can be established for a weak formulation of (10.2).

We start with the *Momentum equation*. Scalar multiplication with a vector-valued test function from the Sobolev space

$$v \in V_D = \left\{ v \in (H^1(\Omega))^n : v|_{\Gamma_D} = 0 \; a.e. \right\}$$

and integrating over the domain gives

$$\begin{aligned}
\int_\Omega [-\Delta u + \nabla p] \cdot v \, dx &= \sum_{i=1}^n \int_\Omega [-\Delta u_i + \partial_{x_i} p] \cdot v_i \, dx \\
&= \sum_{i=1}^n \left[\int_\Omega \nabla u_i \cdot \nabla v_i \, dx - \int_{\partial\Omega} (\nabla u_i \cdot \nu) v_i ds - \int_\Omega p \partial_{x_i} v_i dx + \int_{\partial\Omega} p \nu_i v_i ds \right] \\
&= \int_\Omega \sum_{i=1}^n \nabla u_i \cdot \nabla v_i dx - \int_\Omega p(\nabla \cdot v) dx - \int_{\Gamma_N} [(\nabla u \nu) - p\nu] \cdot v ds \\
&= \int_\Omega \nabla u : \nabla v dx - \int_\Omega p(\nabla \cdot v) dx - \int_{\Gamma_N} s \cdot \nu ds.
\end{aligned}$$

We exploited that the test function is zero on the Dirichlet boundary and we introduced the notation

$$T : S = \sum_{i=1}^n \sum_{i=j}^n T_{ij} S_{ij}$$

for any two $n \times n$ matrices T and S. Also observe that the natural boundary condition arises from integration by parts.

For later use we define the two bilinear forms:

$$a(u,v) = \int_\Omega \nabla u : \nabla v \, dx, \qquad b(v,p) = -\int_\Omega p(\nabla \cdot v) \, dx.$$

Next let us consider the *mass conservation equation*. Now multiply with a scalar test function $q \in L^2(\Omega)$ and obtain

$$\int_\Omega (\nabla \cdot u) q dx = -b(u,q)$$

Combining everything, we obtain the following weak formulation of the stationary Stokes equations.
Find $(u,p) \in U \times Q$ such that

$$a(u,v) + b(v,p) = l(v) \qquad \forall v \in V_D \tag{10.4a}$$
$$b(u,q) = 0 \qquad \forall q \in Q \tag{10.4b}$$

where the right hand side linear functional is given by

$$l(v) = \int_\Omega f \cdot v \, dx + \int_{\Gamma_N} s \cdot v \, ds. \tag{10.5}$$

The function space U is given by the affine shifted space

$$U = \{w \in (H^1(\Omega))^n \, : \, w = u_g + u', u' \in V_D, u_g \in (H^1(\Omega))^n, u_g|_{\Gamma_D} = g \ a.e.\}. \tag{10.6}$$

Since the problem is linear, it can be reduced to one with homogeneous Dirichlet boundary conditions by inserting $u = u_g + u'$ (same as in Poisson equation). However, doing this in a straightforward way results in a non-homogeneous mass conservations equation. We will show below how this can be avoided by selecting a specific function u_g. As a result we can then set $U = V_D$.

The space Q has to be adapted according to the boundary conditions:

$$Q = \begin{cases} L^2(\Omega) & \text{if } \mathrm{meas}(\Gamma_N) \neq 0, \\ Q_0 = \{q \in L^2(\Omega) : \int_\Omega q \, dx = 0\} & \text{if } \Gamma_D = \partial\Omega. \end{cases}$$

Existence and uniqueness of the solution can now be established with the Banach-Nečas-Babuška theorem (2.7). This requires establishing the inf-sup condition.

Reduction to Homogeneous Dirichlet Boundary Conditions

In case of the groundwater flow problem we were able to reduce the problem to homogeneous Dirichlet conditions. This was necessary there to be apply to apply the Lax-Milgram theorem. Here, it is necessary for the reformulations that we will study below.

We illustrate how the reduction to homogeneous Dirichlet conditions can be done in such a way that the right hand side of the mass conservation equation stays zero. We restrict ourselves to the case when $\Gamma_D = \partial\Omega$, i.e. $V_D = (H^1(\Omega))^n$, and $Q = Q_0$.

1) Choose any $\tilde{u}_g \in (H^1(\Omega))^n$ such that $\tilde{u}_g = u = g$ on $\partial\Omega$. For this function we obtain:

$$\int_\Omega \nabla \cdot \tilde{u}_g \, dx = \int_{\partial\Omega} \tilde{u}_g \nu \, ds = \int_{\partial\Omega} g\nu \, ds = \int_{\partial\Omega} u\nu \, ds = \int_\Omega \nabla \cdot u \, dx = 0.$$

   From this we observe that all functions with the same boundary values share the same average divergence, in this cas zero.

2) The map $\nabla\cdot : V_D \to Q_0$ is surjective, see [Ern/Guermond, section 4.1.3]. This means for any $q \in Q_0$ there is a function $u_q \in V_D$ such that $\nabla u_q = q$. Since $\nabla \cdot \tilde{u}_g \in Q_0$ (we have just shown $\int_\Omega \nabla \cdot u_g \, dx = 0$) there exists $w \in V_D$ s.t. $\nabla \cdot w = \nabla \cdot \tilde{u}_g$ almost everywhere.

3) Now set $u_g := \tilde{u}_g - w$ and observe

   a) We have $u_g = g$ on $\partial\Omega$ since $w = 0$ on $\partial\Omega$.

   b) And $\nabla \cdot u_g = \nabla \cdot \tilde{u}_g - \nabla \cdot w = \nabla \cdot \tilde{u}_g - \nabla \cdot \tilde{u}_g = 0$.

Now with this u_g we make the Ansatz $u = u_g + u'$ in (10.4) with $u' \in V_D$ and obtain the equivalent weak formulation. Find $(u', p) \in V_D \times Q$ such that

$$a(u', v) + b(v, p) = l(v) - a(u_g, v) := l'(v) \qquad \forall v \in V_D, \tag{10.7a}$$

$$b(u', q) = 0 \qquad \forall q \in Q. \tag{10.7b}$$

The second equation follows now from $\nabla \cdot u_g = 0$:

$$b(u_g + u', q) = -\int_\Omega (\nabla \cdot u_g) q \, dx + b(u', q) = b(u', q).$$

We now embark on deriving two additional equivalent forms of the weak formulation of the stationary Stokes equations. These formulatios shed some light on the fact that there is no boundary condition for pressure.

10.3. Constrained Weak Formulation

Equation (10.7b) ensures that $\nabla \cdot u' = 0$. This constraint could also be included into the definition of the function space for velocity (again we just concentrate on the case $\Gamma_D = \partial\Omega$):

$$V_0 = \left\{ v \in (H_0^1(\Omega))^n \,:\, \nabla \cdot v = 0 \text{ a.e. } \right\} \subset V_D. \tag{10.8}$$

Correspondingly, the test functions are also restricted to the space V_0. Then, for $v \in V_0$ we have $b(v,p) = -\int_\Omega (\nabla \cdot v) p \, dx = 0$ and we obtain the following *constrained* weak formulation. Find $u'' \in V_0$ such that

$$a(u'', v) = l'(v) \qquad \forall v \in V_0. \tag{10.9}$$

This probem has a unique solution as V_0 is a Hilbert space, a is coercive bilinear form (as it is just n times the bilinear form of the Poisson problem) and l' is continuous. Thus one can apply the Lax-Milgram Theorem (2.8) as in the Poisson case.

10.4. Optimization Formulation

A third reformulation will make clear the connection to the original formulation. Since a is symmetric and coercive, we may rewrite (10.9) as an equivalent optimization problem over the space V_0:

$$\min_{v \in V_0} J(v) \quad \text{with} \quad J(v) = \frac{1}{2} a(v,v) - l'(v). \tag{10.10}$$

This unconstrained optimization problem can be rewritten equivalently as a constrained optimization problem over the larger space V_D by employing the definition of the space V_0:

$$\begin{cases} \min\limits_{v \in V_D} J(v) \\ \nabla \cdot v = 0 \quad a.e. \end{cases} \tag{10.11}$$

Constrained optimization problems may be solved with the method of *Lagrange multipliers*. This method converts a constrained optimization problem into an unconstrained optimization problem with a new functional. Since (10.11) involves minimization of a functional, the Lagrange multiplier turns out to be a scalar function which we call p. The Lagrange functional $\Lambda(u,p)$ is defined as

$$\Lambda(u,p) = J(u) + b(u,p).$$

We now show that finding the minimum of Λ, i.e.

$$\min_{u \in V_D,\ p \in Q_0} \Lambda(u, p) \tag{10.12}$$

is equivalent to solving the stationary Stokes problem.

A necessary condition for the solution of (10.12) is that variational derivatives vanish. Let us take the variational derivative with respect to u:

$$\begin{aligned}
\Lambda(u + tv, p) &= \frac{1}{2} a(u + tv, u + tv) - l'(u + tv) + b(u + tv, p) \\
&= \frac{1}{2} a(u, u) + t a(u, v) + \frac{1}{2} t^2 a(v, v) - l'(u) - t l'(v) + b(u, p) + t b(v, p) \\
&= \frac{1}{2} a(u, u) - l'(u) + b(u, p) + t \left[a(u, v) - l'(v) + b(v, p) \right] + \frac{1}{2} t^2 a(v, v)
\end{aligned}$$

and therefore

$$\frac{d}{dt} \Lambda(u + tv, p)|_{t=0} = a(u, v) - l'(v) + b(v, p) = 0 \qquad \forall v \in V_D$$

which is (10.7a).

The variational derivative with respect to p gives then:

$$\begin{aligned}
\Lambda(u, p + tq) &= \frac{1}{2} a(u, u) - l'(u) + b(u, p + tq) \\
&= \frac{1}{2} a(u, u) - l'(u) + b(u, p) + t b(u, q)
\end{aligned}$$

and

$$\frac{d}{dt} \Lambda(u, p + tq)|_{t=0} = b(u, q) = 0 \qquad \forall q \in Q_0$$

which is (10.7b).

Conclusion

1) The formulations (10.7), (10.11) and (10.12) are equivalent.
2) The pressure can be interpreted as a Lagrange multiplier when converting the constrained optimization problem in the divergence free space into an unconstrained optimization problem.
3) Mathematically, this is the reason why there are no boundary conditions on the pressure.

Chapter 11.

Discretization of the Stationary Stokes Equations

This chapter follows the book Elman et al. [2014]. We seek to solve the problem

$$
\begin{aligned}
-\Delta u + \nabla p &= f && \text{in } \Omega, && (11.1a)\\
\nabla \cdot u &= 0 && \text{in } \Omega, && (11.1b)\\
u &= g && \text{on } \Gamma_D \subset \partial\Omega, && (11.1c)\\
\nabla u \nu - p\nu &= s && \text{on } \Gamma_N \subset \partial\Omega \setminus \Gamma_D, && (11.1d)
\end{aligned}
$$

in its weak formulation where we look for $(u, p) \in U \times Q$ such that

$$
\begin{aligned}
a(u, v) + b(v, p) &= l(v) && \forall v \in V_D, && (11.2a)\\
b(u, q) &= 0 && \forall q \in Q, && (11.2b)
\end{aligned}
$$

where the forms are

$$
a(u, v) = \int_\Omega \nabla u : \nabla v \, dx, \qquad b(v, p) = -\int_\Omega p(\nabla \cdot v) \, dx,
$$

$$
l(v) = \int_\Omega f \cdot v \, dx - \int_{\Gamma_N} s \cdot v \, ds
$$

and the spaces are

$$
V_D = \{v \in (H^1(\Omega))^n \, : \, v|_{\Gamma_D} = 0\}, \quad Q = \begin{cases} L^2(\Omega) & \text{if } \operatorname{meas}(\Gamma_N) \neq 0 \\ Q_0 & \text{if } \Gamma_D = \partial\Omega \end{cases},
$$

$$
U_D = \{w \in (H^1(\Omega))^n \, : \, w = u_g + u', u' \in V_D, u_g \in (H^1(\Omega))^n, u_g|_{\Gamma_D} = g\}.
$$

In chapter 10 we have stated that a unique solution exists.

11.1. Finite Element Discretization of the Stokes Equations

In principle we follow the same recipe as for the scalar elliptic problem:

1) Choose appropriate subspaces of the continuous spaces $V_{h,D} \times Q_h \subset V_D \times Q$ (conforming finite element method).

2) Choose a basis for the discrete spaces:

$$V_{h,D} = \text{span}\{\varphi_i \,:\, i = 1, \ldots, N_u\}, \qquad U_{h,D} = u_{h,g} + V_{h,D},$$
$$Q_h = \text{span}\{\psi_i \,:\, i = 1, \ldots, n_p\}.$$

3) Represent the solution in the basis:

$$u_h = \sum_{j=1}^{n_u} x_j \varphi_j + \sum_{j=n_u+1}^{N_u} X_j \varphi_j, \qquad p_h = \sum_{j=1}^{n_p} y_j \psi_j.$$

Here the coefficients X_j depend on the Dirichlet boundary conditions.

4) Set up the linear system by inserting the basis representation and testing with all test functions:

$$\sum_{j=1}^{n_u} x_j a(\varphi_j, \varphi_i) + \sum_{j=1}^{n_p} y_j b(\varphi_i, \psi_j) = l(\varphi_i) - \sum_{j=n_u+1}^{N_u} X_j a(\varphi_j, \varphi_i) \quad i = 1, \ldots, n_u,$$
$$\sum_{j=1}^{n_u} x_j b(\varphi_j, \psi_i) = - \sum_{j=n_u+1}^{N_u} X_j b(\varphi_j, \psi_i) \qquad i = 1, \ldots, n_p.$$

Note that in the discrete setting we use a Lagrange basis and the straightforward reduction to homogeneous Dirichlet condition results in a non-zero right hand side in the mass conservation equation.

The linear system can be written in block form

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \quad \Leftrightarrow \quad Kz = b \tag{11.3}$$

with the obvious definitions

$$(A)_{ij} = a(\varphi_j, \varphi_i), \qquad f_i = l(\varphi_i) - \sum_{j=n_u+1}^{N_u} g(\xi_j) a(\varphi_j, \varphi_i),$$
$$(B)_{ij} = b(\varphi_j, \psi_i), \qquad g_i = - \sum_{j=n_u+1}^{N_u} g(\xi_j) b(\varphi_j, \psi_i).$$

Note that we reuse the symbols f and g for the right hand side vectors.

A standard choice for the velocity space is to generate it from a scalar space $W_{h,D} = \{\eta_1, \ldots, \eta_{n_u/n}\}$ for the individual components, i.e. $V_{h,D} = (W_{h,D})^n$ with

$$\begin{aligned} \varphi_1 &= (\eta_1, 0, \ldots)^T, \quad \varphi_2 = (\eta_2, 0, \ldots)^T, \quad \ldots, \quad \varphi_{n_u/n} = (\eta_{n_u/n}, 0, \ldots)^T, \\ \varphi_{n_u/n+1} &= (0, \eta_1, 0, \ldots), \qquad\qquad\qquad\qquad \ldots \qquad \varphi_{n_u} = (0, \ldots, \eta_{n_u/n}). \end{aligned}$$

Then the matrices A and B have an additional block structure due to the velocity components:

$$A = \begin{bmatrix} A_1 & 0 & \ldots & 0 \\ 0 & A_2 & \ldots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \ldots & 0 & A_n \end{bmatrix}, \qquad B = \begin{bmatrix} B_1 & B_2 & \cdots & B_n \end{bmatrix},$$

where all blocks are the same. In the following we assume thus structure if not mentioned otherwise.

11.2. Solvability of the Discrete Stokes System

We are now interested in the regularity of the matrix K. In contrast to the scalar elliptic problem this is *not* implied by the regularity of the continuous problem (there, the coercivity of the bilinear form immediately implied the postive definiteness of the system matrix). It turns out that in the Stokes problem the finite element spaces $V_{h,D}$ and Q_h cannot be chosen independently of each other. The purpose of this section is to shed light on this issue.

What we know is that the matrix A (the velocity block) is indeed regular since it does only involve the bilinear form from the Poisson equation n times. From this we conclude the following:

a) The matrix K is regular iff $Kz = 0$ implies $z = 0$.

b) So assume $Kz = 0$. This is means

$$\begin{aligned} Ax &+ B^T y &&= 0, \\ Bx & &&= 0, \end{aligned} \tag{11.4}$$

with $z = (x^T, y^T)^T$. Scalar multiplication of the first equation with x results in

$$x^T Ax + x^T B^T y = x^T Ax + (Bx)^T y = x^T Ax = 0$$

since $Bx = 0$ from the second equation. Now since A is regular we conclude that $x = 0$.

c) Setting $x = 0$ in the first equation results in $B^T y = 0$. To conclude that $y = 0$ we need that all columns of the matrix B^T are linearly independent (which is the same as B^T having maximal rank). This is possible since the number of rows of B^T is typically greater than the number of columns (which is a necessary condition) but care must be taken. In practice one chooses a combination of finite element spaces for velocity and pressure and then needs to prove that B^T has maximal rank.

d) This is equivalent to proving the inf-sup condition for the discrete problem. It is important to note that the discrete inf-sup condition does *not* follow automatically from the continuous inf-sup condition.

e) The case of $\Gamma_D = \partial\Omega$ needs additional considerations. In theory one should chose Q_h to be a finite dimensional subspace of Q_0, the space of L_2 functions with vanishing average. This is, however, not the usual approach taken in practice since it might introduce full rows into the system matrix. Therefore one often uses $Q_h \subset L_2(\Omega)$ in practice. In this case B^T is a singular matrix since pressure is only determined up to a constant. This results in the condition that $\text{null}(B^T) = \text{span}\{(1, \ldots, 1)^T\}$ (where we have assumed a Lagrange basis).

f) Alternative version of the argument. Starting from (11.5) we can apply block Gaussion elimination to arrive at the equivalent system

$$\begin{aligned} Ax \;+\; B^T y &= 0, \\ -BA^{-1}B^T y &= 0. \end{aligned} \tag{11.5}$$

Now the question is whether the *Schur complement* $S = BA^{-1}B^T$ is regular. Multiplying the second equation with y^T we conclude from $y^T BA^{-1}B^T y = (B^T y)^T A^{-1} B^T y = 0$ that y must be zero if B^T gas maximal rank. (Again the argument must be modified as above when $\Gamma_D = \partial\Omega$.

11.3. Stable Finite Element Pairs

These are choices for the spaces $V_{h,D}$ and Q_h such that B^T has the desired properties, i.e. either $\text{null}(B^T) = 0$ or $\text{null}(B^T) = \text{span}\{(1, \ldots, 1)^T\}$. In this section we just state some important results from the literature.

Quadrilateral Elements

(i.e. two space dimensions).

- Lowest order equal order elements, i.e. we take piecewise bilinear finite elements for each velocity component and for pressure. This is denoted by $Q_1^2 \times Q_1$ or $Q_1 - Q_1$ for short. This element pair is *not* stable, and is in fact one of the famous examples where numerical analysis could explain why a method would not work. The instability of the equal order element can be understood from simple size considerations. Let us denote by n the number of vertices in the mesh and by $n_{\partial\Omega}$ the number of mesh vertices on the Dirichlet boundary. Then $n_u = 2(n - n_{\partial\Omega})$ is the number of velocity degree of freedoms and $n_p = n$ is the number of pressure degrees of freedom (because there are no boundary conditions on pressure). The matrix B^T has the block form

$$B^T = \begin{bmatrix} B_1^T \\ B_2^T \end{bmatrix}$$

  where $B_1 = B_2$. Due to this, B^T has linearly independent columns iff B_1^T has linearly independent columns. However, the size of B_1 is $(n - n_{\partial\Omega}) \times n$ and therefore the columns can not be linearly independent unless $n_{\partial\Omega} = 0$, a case which we excluded (in that case A is not regular). In the case $\Gamma_D = \partial\Omega$ B_1^T is required to have $n-1$ linearly independent columns. This is not possble since $n_{\partial\Omega} > 1$.

- The analysis of the equal order element gives a route how to obtain stable elements: increase the polynomial degree in the velocity space! Taking polynomials of degree 2, i.e. $(Q_2)^2 \times Q_1$, or $Q_2 - Q_1$ for short is called the *Taylor-Hood* element. It is stable if the mesh contains more than one element.

- The pressure is in $L^2(\Omega)$ in the continuous weak formulation. Therefore in the discrete subspace no continuity is actually needed. It turns out, $(Q_2)^2 \times W_h^1$ (the space from the discontinuous Galerkin method), is a stable element. However, the inf-sup constant depends on the aspect ratio of the element. This might become a problem in the Navier-Stokes equations where boundary layers need to be resolved. An advantage of elements with an element-wise discontinuous pressure space is that they are locally mass conservative (test the mass conservation equation with a test function that is constant on one element).

- One can also reduce the pressure space to piecewise constants. It turns out $(Q_2)^2 \times W_h^0$ is also stable. But from a rate of convergence perspective it is not desired to have a large difference in polynomial degrees as the convergence rate of velocity might be influenced from the low polynomial degree in pressure (which is the case here).

- Unfortunately, the more balanced choice $(Q_1)^2 \times W_h^0$ is an unstable element pair.

Triangular Elements

- $(P_2)^2 \times P_1$, the Taylor-Hood element for triangles is stable for a mesh with at least three elements.
- $(P_2)^2 \times W_h^1$ is *not* stable. However it can be made stable by slightly enlarging the velocity space by a so-called cubic bubble function. On the reference element this is the function

$$\varphi(\hat{x}_1, \hat{x}_2) = \hat{x}_1 \hat{x}_2 (1 - \hat{x}_1 - \hat{x}_2).$$

  Note that this function is zero at the boundary of the element. Therefore the corresponding degree of freedom can be statically eliminated.
- The $(P_2)^2 \times W_h^0$ element is stable but has again suboptimal rate of convergence.
- The trick to enlarge the velocity space by local bubble functions can also be applied to the equal order element. The *Mini* element (P1 + cubic bubble)$^2\times$ P_1 is stable!

11.4. Stabilized Elements

Here the idea is to add terms to the weak formulation in such a way that

- a proper null space is ensured (stability)
- but the accuracy is not harmed.

The modified weak formulation for finding $(u, p) \in U \times Q$ reads

$$a(u, v) + b(v, p) = l(v) \qquad \forall v \in V_D, \tag{11.6a}$$
$$b(u, q) + \beta c(p, q) = 0 \qquad \forall q \in Q, \tag{11.6b}$$

with the new bilinear form $c(p, q)$ and the scalar parameter $\beta > 0$. Correspondingly the discrete system then is

$$\begin{bmatrix} A & B^T \\ B & \beta C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix},$$

where the zero block is now replaced by the matrix C. The Schur complement system now is

$$S_{stab} y = (-BA^{-1}B^T + \beta C)y = g - BA^{-1}f.$$

Since $-BA^{-1}B^T$ is negative definite (see next chapter) taking C positive definite and β large enough will make the system regular. On the other hand β should not be too large as this would harm accuracy.

Here are different choices for c:

- $c(p,q) = \int_\Omega \nabla p \cdot \nabla q \, dx$ works for uniform grids.
- $c(p,q) = \int_\Omega (p - \Pi_0 p)(q - \Pi_0 q) \, dx$ where Π_0 is the L^2-projection from piecewise linears to piecewise constants, is an example of a projection scheme.
- The bilinear form $c(p,q) = \sum_{\gamma \in \mathcal{F}_h^{int}} \frac{|e|}{|\gamma|} \int_\gamma [\![p]\!][\![q]\!] \, ds$ can be used to stabilize the lowest order $Q_1 - P_0$ and $P_1 - P_0$ elements.

11.4.1. Additional Comments

Finite element methods for the Stokes and Navier-Stokes equations are still subject of active research. Recent advances include improved outflow boundary conditions, see Braack and Mucha [2014] as well as so-called pressure robust methods, see John et al. [2017].

Chapter 12.

Solvers for the Discrete Stokes Equations

12.1. Indefinitness of the Discrete System

Discretizing the stationary Stokes system with stable conforming finite elements results in a linear system in saddle point form:

$$Kz = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} = b. \tag{12.1}$$

The system matrix K is symmetric but *indefinite*, i.e. it has positive and negative eigenvalues. This can be shown as follows. The block LU-decomposition of K is

$$K = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} = \begin{bmatrix} I_u & 0 \\ BA^{-1} & I_p \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & -BA^{-1}B^T \end{bmatrix} \begin{bmatrix} I_u & A^{-1}B^T \\ 0 & I_p \end{bmatrix}.$$

From this we observe:

- A is symmetric and positive definite and therefore has real and positive eigenvalues.
- $S = BA^{-1}B^T$ is called the pressure Schur-complement. The rigidity theorem of Sylvester says that the signs of the eigenvalues of a symmetric matrix Z (i.e. with real eigenvalues) do not change under a congruence transform XZX^T when X has maximal rank. When applied to S it follows that S is symmetric and positive definite (since A^{-1} is symmetric and positive definite). As a consequence $-S$ has negative eigenvalues.
- Applying Sylvesters theorem to the LU decomposition of K we see that K has n_u positive and n_p negative eigenvalues. This requires that we use a stable finite element pair where B^T has maximal rank n_p.

Note that the ratio of the number of positive and negative eigenvalues does not change under mesh refinement.

12.2. Preconditioning

Indefinitness of K is a major problem for iterative solvers. The conjugate gradient (CG) method only works for symmetric and positiv definite matrices, however, a similar method, the so-called minimum residual method (MINRES) works also for symmetric indefinite matrices.

In addition, any fast iterative method needs a good preconditioner. Left preconditioning means that CG or MINRES is applied to

$$M^{-1}Kz = M^{-1}b$$

where M is called preconditioner.

Since K has 2×2 block structure it is natural to consider also preconditioners with such a structure, in particular we will investigate the choice

$$M = \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix}. \tag{12.2}$$

Convergence of the MINRES method (and also the CG method) depends crucially on the eigenvalues of the preconditioned system $M^{-1}A$ as well as their distribution. We observe that

$$M^{-1}Az = \lambda z \qquad \Leftrightarrow \qquad Az = \lambda Mz,$$

where the latter system is called a generalized eigenvalue problem. It turns out, its eigenvalues can be easily determined for the specific M from (12.2).

1) Any x with $Bx = 0$ gives rise to an eigenvalue 1, since

$$K \begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} Ax \\ Bx \end{bmatrix} = \begin{bmatrix} Ax \\ 0 \end{bmatrix} = 1 \cdot \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} x \\ 0 \end{bmatrix} = 1 \cdot M \begin{bmatrix} x \\ 0 \end{bmatrix}.$$

   Assuming that B^T has maximal rank n_p then the kernel of B has dimension $n_u - n_p$.

2) Assume now that $\lambda \neq 1$. Then, a corresponding eigenvector satisfies

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} Ax + B^T y \\ Bx \end{bmatrix} = \lambda \cdot \begin{bmatrix} Ax \\ Sy \end{bmatrix},$$

   which is equivalent to

$$\begin{bmatrix} (1-\lambda)A & B^T \\ B & -\lambda S \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0.$$

Since $\lambda \neq 1$ we can apply block Gauß elimination and obtain

$$\begin{bmatrix} (1-\lambda)A & B^T \\ 0 & -\lambda S - \frac{1}{1-\lambda} S \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0 \quad \Leftrightarrow \quad (\lambda^2 - \lambda - 1) S y = 0.$$

In case that B^T has maximal rank and therefore S is nonsingular, the last equation can only be satisfied when λ satisfies

$$\lambda^2 - \lambda - 1 = 0 \quad \Leftrightarrow \quad \lambda_\pm = \frac{1}{2} \pm \frac{\sqrt{5}}{2}.$$

Thus, any $y \in \mathbb{R}^{n_p}$ gives rise to two eigenvalues λ_+ and λ_-, each with corresponding eigenvector $z = \begin{bmatrix} (\lambda - 1)^{-1} A B^T y \\ y \end{bmatrix}$.

3) In total we have now determined $n = n_u - n_p + 2n_p = n_u + n_p$ eigenvalues, which are all eigenvalues.

4) In the case of all Dirichlet conditions and $\text{null}(B^T) = \text{span}\{\mathbb{1}\}$, $\mathbb{1} = (1, \ldots, 1)^T$, we have one eigenvector with eigenvalue zero:

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} 0 \\ \mathbb{1} \end{bmatrix} = 0 \cdot \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} 0 \\ \mathbb{1} \end{bmatrix}.$$

In 1) above the number of eigenvalues 1 is now $n_u - n_p + 1$ and there is one pair $\lambda_\pm$ missing in 2) since S has only rank $n_p - 1$ now. Together we have $n_u - n_p + 1 + 1 + 2(n_p - 1) = n$ eigenvalues.

One can show that the convergence of the MINRES method depends crucially on the distribution of the eigenvalues. If there are only three *different* eigenvalues, then MINRES converges in three steps.

However, note that the preconditioner M is not computationally efficient to realize since S is a dense matrix involving A^{-1}. When applying the preconditioner we need to solve a system $Mz = d$ which boils down to solve two independent systems $Ax = d_u$ and $Sy = d_p$. Whereas the first system can be solved efficiently using multigrid, the second system is problematic to solve since the matrix S is *not available*! Thus it is necessary to approximate M by a computationally efficient preconditioner $\tilde{M}$.

Wathen-Silvester Preconditioner

A computationally efficient version of the Schur complement preconditioner exploits properties of stable finite element methods. The stability property

$\text{null}(B^T) = \{0\}$ in terms of linear algebra can be quivalently expressed by the discrete inf-sup condition on the level of the finite element method:

$$\inf_{0\neq q_h\in Q_h} \sup_{0\neq v_h\in V_h} \frac{|b(v_h, q_h)|}{\|q_h\|_{0,\Omega}\|v_h\|_{1,\Omega}} \geq \beta_h > 0 \tag{12.3}$$

with β_h independent of h.

Observe, that with the basis representations

$$u_h = \sum_{i=1}^{n_u} x_i\varphi_i, \qquad p_h = \sum_{i=1}^{n_p} w_i\psi_i, \qquad v_h = \sum_{i=1}^{n_u} z_i\varphi_i, \qquad q_h = \sum_{i=1}^{n_p} y_i\psi_i,$$

we have

$$\begin{aligned} a(u_h, v_h) &= \langle Ax, z\rangle, \qquad & b(u_h, q_h) &= -(q_h, \nabla\cdot u_h)_{0,\Omega} = \langle Bx, y\rangle, \\ (p_h, q_h)_{0,\Omega} &= \langle Qw, y\rangle. \end{aligned}$$

There should be no confusion of the "mass matrix" Q with entries $q_{i,j} = (\psi_i, \psi_j)_{0,\Omega}$ and the space Q. Also note that

$$c\|v\|_{1,\Omega} \leq \sqrt{a(v, v)} \leq C\|v\|_{1,\Omega} \tag{12.4}$$

with $c, C > 0$ due to coercivity and continuity of the bilinear form a on the appropriate space V used in the weak formulation.

The following Lemma shows that the Schur complement S is spectrally equivalent to the mass matrix Q, i.e. it has the same extreme eigenvalues up to a constant.

Lemma 12.1. There exist constants $c_1, c_2 > 0$ and independent of h such that

$$c_1\langle Qy, y\rangle \leq \langle Sy, y\rangle \leq c_2\langle Qy, y\rangle. \tag{12.5}$$

Proof. First we observe due to the norm equivalence (12.4)

$$c\frac{|\langle Bz, y\rangle|}{\langle Az, z\rangle^{1/2}\langle Qy, y\rangle^{1/2}} \leq \frac{|b(v_h, q_h)|}{\|q_h\|_{0,\Omega}\|v_h\|_{1,\Omega}} \leq C\frac{|\langle Bz, y\rangle|}{\langle Az, z\rangle^{1/2}\langle Qy, y\rangle^{1/2}}.$$

Now for fixed y we have the identity

$$\begin{aligned} \max_{z\neq 0} \frac{|\langle Bz, y\rangle|}{\langle Az, z\rangle^{1/2}\langle Qy, y\rangle^{1/2}} &= \frac{1}{\langle Qy, y\rangle^{1/2}} \max_{w\neq 0} \frac{|\langle BA^{-1/2}w, y\rangle|}{\langle w, w\rangle^{1/2}} \\ &= \frac{1}{\langle Qy, y\rangle^{1/2}} \max_{w\neq 0} \frac{|\langle A^{-1/2}B^Ty, w\rangle|}{\langle w, w\rangle^{1/2}} \\ &= \frac{\langle BA^{-1}B^Ty, y\rangle^{1/2}}{\langle Qy, y\rangle^{1/2}} = \frac{\langle Sy, y\rangle^{1/2}}{\langle Qy, y\rangle^{1/2}}. \end{aligned}$$

In the first step we used the transformation $w = A^{1/2}z$ and in the last step we used the "duality argument" $\sup_{w\neq 0} \frac{\langle x,w\rangle}{\|w\|} = \|x\|$ (follows from Cauchy-Schwarz).

With this we obtain now

$$\begin{aligned}\min_{y\neq 0} \frac{\langle Sy,y\rangle^{1/2}}{\langle Qy,y\rangle^{1/2}} &= \min_{y\neq 0}\max_{z\neq 0} \frac{|\langle Bz,y\rangle|}{\langle Az,z\rangle^{1/2}\langle Qy,y\rangle^{1/2}} \\ &\geq \inf_{0\neq q_h\in Q_h}\sup_{0\neq v_h\in V_h} \frac{1}{C}\frac{|b(v_h,q_h)|}{\|q_h\|_{0,\Omega}\|v_h\|_{1,\Omega}} \geq \frac{\beta_h}{C}.\end{aligned}$$

The second inequality is

$$\begin{aligned}\max_{y\neq 0} \frac{\langle Sy,y\rangle^{1/2}}{\langle Qy,y\rangle^{1/2}} &= \max_{y\neq 0}\max_{z\neq 0} \frac{|\langle Bz,y\rangle|}{\langle Az,z\rangle^{1/2}\langle Qy,y\rangle^{1/2}} \\ &\leq \sup_{0\neq q_h\in Q_h}\sup_{0\neq v_h\in V_h} \frac{1}{c}\frac{|b(v_h,q_h)|}{\|q_h\|_{0,\Omega}\|v_h\|_{1,\Omega}} \\ &\leq \frac{1}{c}\sup_{0\neq v_h\in V_h} \frac{\|\nabla\cdot v_h\|_{0,\Omega}}{\|v_h\|_{1,\Omega}} \qquad \text{(Cauchy-Schwarz)} \\ &= \frac{1}{c}\sup_{0\neq v_h\in V_h} \frac{\|\nabla\cdot v_h\|_{0,\Omega}}{(\|v_h\|_{0,\Omega}^2 + \|\nabla\cdot v_h\|_{0,\Omega}^2 + \|\nabla\times v_h\|_{0,\Omega}^2)^{1/2}} \leq \frac{1}{c}.\end{aligned}$$

The last step uses the Helmholtz decomposition of a vector field. □

The previous Lemma suggests that

$$\tilde{M} = \begin{bmatrix} A & 0 \\ 0 & Q \end{bmatrix}$$

is an efficient preconditioner. In practice systems with A and Q are solved iteratively with few or even one iteration. In case of discontinuous finite elements for the pressure the matrix Q is diagonal.

12.3. Extension to the Navier-Stokes Equations

Finally, a few remarks about solving the instationary Navier-Stokes equations

$$\begin{aligned}\partial_t u + \nabla\cdot(uu^T) - \eta\Delta u + \nabla p &= f \qquad \text{in } \Omega\times\Sigma, \\ \nabla\cdot u &= 0\end{aligned}$$

with appropriate boundary and initial conditions.

In the weak formulation there are now two additional terms

1) From the time derivative and e.g. implicit Euler discretization in time:

$$\partial_t u \approx \frac{1}{\Delta t}(u^k - u^{k-1}, \qquad m(u^k, v) = \int_\Omega u^k v \, dx.$$

2) The convective term gives rise to a nonlinear form

$$c(z; u, v) := \int_\Omega ((\nabla u) z) \cdot v \, dx.$$

Inserting a basis representation leads to a nonlinear algebraic problem per time step which has the form

$$\begin{bmatrix} \frac{1}{\Delta t} Q + C(x^k) + \nu A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x^k \\ y \end{bmatrix} = \mathrm{rhs}(x^{k-1}) \Leftrightarrow K(x^k, \nu, \Delta t) w = \mathrm{rhs}(x^{k-1}).$$

Some remarks about the solution of this system:

- The matrix $K(z, \nu, \Delta t)$ is non-symmetric and indefinite.
- Different regimes can be considered: For $\Delta t \to 0$ it approaches the mass matrix. For large viscosity ν it approaches the Stokes equation. For large Reynolds numbers (small ν) the nonlinearity is dominating. Clearly, one would like to have a method working in all regimes.
- As nonlinear solvers one can use a fixed point iteration or the Newton method.
- The linear solver needs to consider the three regimes discussed above:
  - For small Reynolds number and large time steps one is essentially in the Stokes regime and the Wathen-Silvester preconditioner discussed above works quite well.
  - for very small time steps when the mass matrix is dominating the Schur complement S should be approximated by a (scaled) Poisson matrix.
  - For high Reynolds numbers preconditioning is most difficult and no perfect methods exist. In that case upwinding methods could be used which lead to lower triangular $C(x^k)$ allowing more robust solvers.

Appendix A.

Nabla and Friends

A.1. Notation for Derivatives

The partial derivative

$$\frac{\partial u}{\partial x_i}(x) = \lim_{h\to\infty} \frac{u(x+he_i) - u(x)}{h}$$

of a scalar function $u : \mathbb{R}^n \to \mathbb{R}$ is written in short notation as

$$\partial_{x_i} u(x) = \frac{\partial u}{\partial x_i}(x).$$

Similarly we have for the higher derivatives

$$\partial^2_{x_i} u(x) = \frac{\partial^2 u}{\partial x_i^2}(x), \qquad \partial_{x_i}\partial_{x_j} u(x) = \frac{\partial^2 u}{\partial x_i \partial x_j}(x), \qquad \dots$$

A vector $\alpha = (\alpha_1, \dots, \alpha_n)^T$ of nonnegative integers α_i is called a *multiindex* of order

$$|\alpha| = \sum_{i=1}^{n} \alpha_i.$$

Sometimes

$$|\alpha|_\infty = \max_{i=1,\dots,n} \alpha_i.$$

is referred to as the maximum order.

For a given multiindex α we set

$$\partial^\alpha u(x) = \partial^{\alpha_1}_{x_1} \dots \partial^{\alpha_n}_{x_n} u(x) = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \dots \partial x_n^{\alpha_n}}(x)$$

For a given nonnegative integer k

$$D^k u(x) = \{\partial^\alpha u(x) \,:\, |\alpha| = k\}$$

denotes the ordered set of all partial derivatives of order k at the point x. Note that $D^k u(x)$ has n^k elements, i.e. $\partial_{x_i}\partial_{x_j} u(x)$ and $\partial_{x_j}\partial_{x_i} u(x)$ are different elements although they have the same value.

For the special cases $k = 1$ and $k = 2$ we identify $D^1 u(x)$ with the gradient $\nabla u(x)$ and $D^2 u(x)$ with the Hessian matrix $\nabla^2 u(x)$ (see below for the definition of gradient and Hessian).

In the case of a function $u(x, y)$, $u : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, we write $D^1_x u$ or $D^2_y u$ to indicate the variable with respect to which differentiation is to be applied.

A.2. Vector Differential Calculus

The whole presentation treats the differential operators only in cartesian coordinates.

A.2.1. Nabla Operator

The nabla operator formally is a row or column vector of partial derivatives with respect to all variables of its argument:

$$\nabla = (\partial_1, \ldots, \partial_n)^T \tag{A.1}$$

(when we assume that the argument has n variables).

A.2.2. Gradient

Gradient of a Scalar Nabla applied to a scalar function $u(x_1, \ldots, x_n)$ in n variables gives a vector called "gradient" of the function:

$$\nabla u = (\partial_1 u, \ldots, \partial_n u)^T . \tag{A.2}$$

We can imagine ∇ to be a column vector in this case applied to a scalar which gives a vector.

The gradient of a scalar function in point x is a vector which is perpendicular to the level set $l(c) = \{y : u(y) = c\}$ for $c = u(x)$ pointing in the direction of the steepest increase of the function u.

Gradient of a Vector-valued Function Nabla applied to a vector-valued function

$$u(x) = (u_1(x_1, \ldots, x_n), \ldots, u_m(x_1, \ldots, x_n))^T$$

with m components in n variables gives a matrix called the "Jacobian" of the function:

$$\nabla u = \begin{pmatrix} (\nabla u_1)^T \\ \vdots \\ (\nabla u_m)^T \end{pmatrix} = \begin{pmatrix} \partial_1 u_1 & \ldots & \partial_n u_1 \\ \vdots & & \vdots \\ \partial_1 u_m & \ldots & \partial_n u_m \end{pmatrix} \qquad \text{or} \qquad (\nabla u)_{i,j} = \partial_j u_i. \tag{A.3}$$

If we wish to view the gradient as a column vector and the function u also as a column vector (of possibly different size) then we formally have:

$$\text{“}\nabla u\text{”} := \left(\nabla u^T\right)^T. \tag{A.4}$$

Here ∇u^T acts as an outer product producing a matrix.

In the case of a scalar function u the matrix $\nabla\nabla u = \nabla^2 u$ is called the Hessian matrix.

A.2.3. Divergence

Divergence of a Vector Field The scalar product of nabla with a vector-valued function gives a scalar called the "divergence" of the function:

$$\nabla \cdot u = \sum_{i=1}^{n} \partial_i u_i.$$

Divergence of a Matrix-valued Function The divergence operator applied to a matrix-valued function

$$\sigma(x_1, \ldots, x_n) = \begin{pmatrix} \sigma_1 \\ \vdots \\ \sigma_m \end{pmatrix} = \begin{pmatrix} \sigma_{1,1}(x) & \ldots & \sigma_{1,n}(x) \\ \vdots & & \vdots \\ \sigma_{m,1}(x) & \ldots & \sigma_{m,n}(x) \end{pmatrix}$$

in n variables is defined to yield the divergence for each row of the matrix. Note that σ needs to have as many columns as there are variables. It produces a vector-valued function:

$$\nabla \cdot \sigma = \begin{pmatrix} \nabla \cdot \sigma_1 \\ \vdots \\ \nabla \cdot \sigma_m \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{n} \partial_j \sigma_{1,j} \\ \vdots \\ \sum_{j=1}^{n} \partial_j \sigma_{m,j} \end{pmatrix} \qquad \text{or} \qquad (\nabla \cdot \sigma)_i = \sum_{j=1}^{n} \partial_j \sigma_{i,j}. \tag{A.5}$$

If we regard the divergence as a row vector and σ an $m \times n$ matrix with n also the number of variables, then we can formally write

$$\text{“}\nabla \cdot \sigma\text{”} := \left(\nabla \cdot (\sigma^T)\right)^T. \tag{A.6}$$

Here the inner product $\nabla \cdot (\sigma^T)$ produces a row vector. Note the similarity to the formula (A.4).

A.2.4. Curl

The “curl” (also called “rot”, which is exactly the same thing) of a vector field is defined as

$$\nabla \times u = \begin{pmatrix} \partial_2 u_3 - \partial_3 u_2 \\ \partial_3 u_1 - \partial_1 u_3 \\ \partial_1 u_2 - \partial_2 u_1 \end{pmatrix} \tag{A.7}$$

which corresponds to the vector (cross) product $a \times b = (a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1)^T$. As stated, it makes only sense for $u : \mathbb{R} \to \mathbb{R}^3$ and there is no obvious extension of the curl operator to n dimensions. However, the related Stokes theorem (see below) can be extended to arbitrary dimensions.

A.2.5. Convection Term in Navier-Stokes Equations

For a vector-valued function u, the convection term in the Navier-Stokes equations is written as $u \cdot \nabla u$ which is formally defined as

$$u \cdot \nabla u = (\nabla u) u = \begin{pmatrix} \nabla u_1 \cdot u \\ \vdots \\ \nabla u_n \cdot u \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n u_i \partial_i u_1 \\ \vdots \\ \sum_{i=1}^n u_i \partial_i u_n \end{pmatrix}. \tag{A.8}$$

Note that the scalar product of a vector with a matrix (∇u is a matrix!) is defined as a vector where each component is the scalar multiplication of the vector with a row of the matrix.

A.2.6. Laplacian

Laplacian of a scalar function The Laplacian takes second order derivatives of a scalar function and is defined as

$$\Delta u = \nabla \cdot \nabla u = \sum_{i=1}^n \partial_i^2 u. \tag{A.9}$$

Laplace of Vector-valued function The definition of the Laplacian is extended to vector-valued functions by applying it to each component, i.e. the Laplacian of a vector-valued function is again a vector-valued function. In agreement with the conventions above we have:

$$\Delta u = \nabla \cdot \nabla u = \begin{pmatrix} \nabla \cdot \nabla u_1 \\ \vdots \\ \nabla \cdot \nabla u_n \end{pmatrix} = \begin{pmatrix} \Delta u_1 \\ \vdots \\ \Delta u_n \end{pmatrix}. \tag{A.10}$$

A.3. Vector Integral Calculus

A.3.1. Matrix Product

Let T, S be two $m \times n$ matrices, then we define

$$T : S = \sum_{i=1}^{m} \sum_{j=1}^{n} T_{i,j} S_{i,j}. \tag{A.11}$$

Applied to two vector-valued functions u, v with m components in n variables we have with the definitions from above:

$$\nabla u : \nabla v = \sum_{i=1}^{m} \nabla u_i \cdot \nabla v_i. \tag{A.12}$$

Now let T, S, Q be $n \times n$ matrices. Then the following holds:

$$T : (QSQ^T) = (Q^T T Q) : S. \tag{A.13}$$

This can be shown as follows:

$$\begin{aligned} T : (QSQ^T) &= \sum_{i=1}^{n} \sum_{j=1}^{n} T_{i,j} (e_i^T Q S Q^T e_j) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{n} T_{i,j} \left(\sum_{k=1}^{n} Q_{i,k} \left(\sum_{l=1}^{n} S_{kl} Q_{l,j}^T \right) \right) \\ &= \sum_{k=1}^{n} \sum_{l=1}^{n} S_{kl} \left(\sum_{i=1}^{n} \sum_{j=1}^{n} T_{i,j} Q_{i,k} Q_{l,j}^T \right) \\ &= \sum_{k=1}^{n} \sum_{l=1}^{n} S_{kl} \sum_{i=1}^{n} Q_{k,i}^T \left(\sum_{j=1}^{n} T_{i,j} Q_{j,l} \right) \\ &= (Q^T T Q). \end{aligned}$$

A.3.2. Integration by Parts

Green's formula for sufficiently smooth scalar functions u, v and a suitable bounded domain Ω is

$$\int_\Omega (\partial_i u)v = -\int_\Omega u\partial_i v + \int_{\partial\Omega} uvn_i \tag{A.14}$$

where n_i is the i-th component of the outer unit normal vector n.

For a vector-valued function u and a scalar function v we then have

$$\int_\Omega (\nabla \cdot u)v = -\int_\Omega u \cdot \nabla v + \int_{\partial\Omega} u \cdot n\, v \quad . \tag{A.15}$$

For a matrix-valued function T and a vector valued function v one shows the corresponding formula

$$\int_\Omega (\nabla \cdot T) \cdot v = -\int_\Omega T : \nabla v + \int_{\partial\Omega} (Tn) \cdot v \tag{A.16}$$

which is needed in the variational formulation of the Navier-Stokes equations. Indeed using the definitions above one obtains:

$$\begin{aligned}
\int_\Omega (\nabla \cdot T) \cdot v &= \int_\Omega \sum_{i=1}^n \left(\sum_{j=1}^n \partial_j T_{i,j} \right) v_i \\
&= \sum_{i=1}^n \sum_{j=1}^n \left\{ -\int_\Omega T_{i,j}\partial_j v_i + \int_{\partial\Omega} T_{i,j} v_i n_j \right\} \\
&= -\int_\Omega \sum_{i=1}^n \sum_{j=1}^n T_{i,j}(\nabla v)_{i,j} + \int_{\partial\Omega} \sum_{i=1}^n \left(\sum_{j=1}^n T_{i,j} n_j \right) v_i \quad .
\end{aligned}$$

Bibliography

Malte Braack and Piotr Boguslaw Mucha. Directional do-nothing condition for the navier-stokes equations. Journal of Computational Mathematics, 32(5): 507 – 521, 2014. ISSN 02549409.

D. Braess. Finite Elemente. Springer, 3rd edition, 2003.

H. Elman, D. Silvester, and A. Wathen. Finite Elements and Fast Iterative Solvers. Oxford University Press, 2nd edition edition, 2014.

K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Computational Differential Equations. Cambridge University Press, 1996. `http://www.csc.kth.se/~jjan/private/cde.pdf`.

A. Ern and J.-L. Guermond. Theory and practice of finite element methods. Springer, 2004.

L. C. Evans. Partial Differential Equations. American Mathematical Society, 2010. 2nd edition.

C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering, 79(11):1309–1331, 2009.

W. Hackbusch. Theorie und Numerik elliptischer Differentialgleichungen. Teubner, 1986. `http://www.mis.mpg.de/preprints/ln/lecturenote-2805.pdf`.

V. John, A. Linke, C. Merdon, M. Neilan, and L. Rebholz. On the divergence constraint in mixed finite element methods for incompressible flows. SIAM Review, 59(3):492–544, 2017. doi: 10.1137/15M1047696.

R. J. Leveque. Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, 2002.

M. Renardy and R. C. Rogers. An Introduction to Partial Differential Equations, volume 13. Springer, 1993. Texts in Applied Mathematics.

B. Schweizer. Partielle Differentialgleichungen. Springer, 2013.