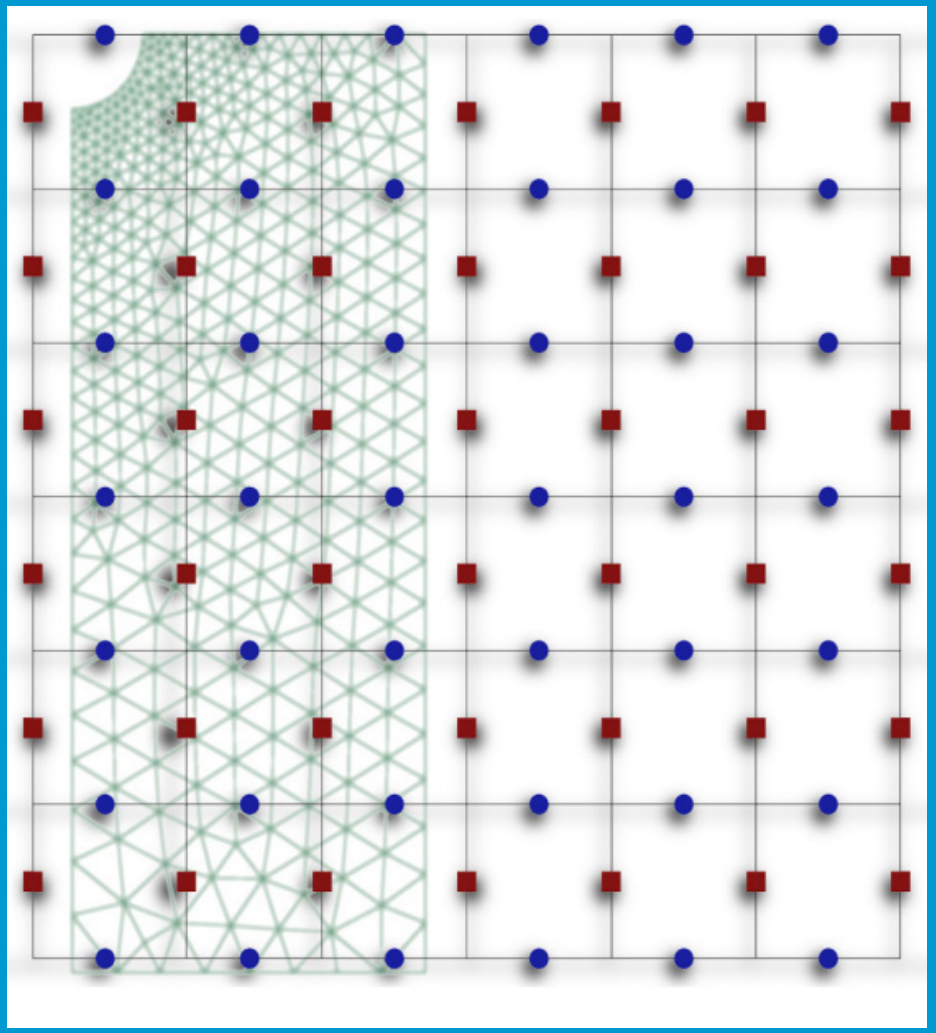
Numerical Methods in Scientific Computing

Jos van Kan, Guus Segal, Fred Vermolen



TU Delft OPEN

Numerical Methods in Scientific Computing



Jos van Kan (1944) graduated in 1968 from Delft University of Technology, Delft, Netherlands, in Numerical Analysis and has been assistant professor at the Department of Mathematics of that institute ever since. He wrote several articles on Numerical Fuid Mechanics (pressure correction methods) and has written a multigrid pressure solver for the Delft software package to solve the Navier Stokes equations. He has been teaching classes in Numerical Analysis since 1971 and wrote several books on the subject.



Guus Segal (1948) graduated in 1971 from Delft University of Technology, Delft, Netherlands, in Numerical Analysis and has been part time assistant professor at the Department of Mathematics of that institute ever since. He is also working in the consultancy and numerical software company SEPRA in Den Haag, The Netherlands. He wrote a number of articles on Finite Element Methods and several articles on curvilinear Finite Volume Methods and Numerical Fluid Mechanics. He has written a book on Finite Element Methods and Navier-Stokes equations. He is the main developer of the finite element package SEPRAN. He has been teaching classes in Numerical Analysis since 1973.



Fred Vermolen (1969) graduated in 1993 from Delft University of Technology, Delft, Netherlands. He wrote his PhD-thesis supervised by the promotores prof Pieter Wesseling (Numerical Analysis) and prof Sybrand van der Zwaag (Materials Science). He wrote several articles on Stefan problems and transport in porous media. His present interest is in mathematical issues in medicine. He has been teaching courses in Numerical Analysis since 2002.

Numerical methods in Scientific Computing

J. van Kan
A. Segal
F. Vermolen

Delft Institute of Applied Mathematics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

© Delft Academic Press
First edition 2005, second edition 2014
ISBN print version: 97890-6562-3638
ISBN electronic version: 978-90-6562-3645

© 2023 TU Delft OPEN Publishing
ISBN paperback: 978-94-6366-738-8
ISBN Ebook: 978-94-6366-740-1
DOI: https://doi.org/10.59490/t.2023.009



This work is licensed under a Creative Commons Attribution 4.0 International license



NUR 919

Key words: numerical mathematics

Preface

This is a book about numerically solving partial differential equations occurring in technical and physical contexts and we (the authors) have set ourselves a more ambitious target than to just talk about the numerics. Our aim is to show the place of numerical solutions in the general modeling process and this must inevitably lead to considerations about modeling itself. Partial differential equations usually are a consequence of applying first principles to a technical or physical problem at hand. That means, that most of the time the physics also have to be taken into account especially for validation of the numerical solution obtained.

This book in other words is especially aimed at engineers and scientists who have 'real world' problems and it will concern itself less with pesky mathematical detail. For the interested reader though, we have included sections on mathematical theory to provide the necessary mathematical background. Since this treatment had to be on the superficial side we have provided further reference to the literature where necessary.

Delft, June 2005

Jos van Kan
Guus Segal
Fred Vermolen

Note to the first edition improvements

In this improved first edition exercises and theory are more separately presented. Furthermore, some parts, such as the parts on boundary fitted coordinates, on coordinate transformation, the treatment of essential boundary conditions for FEM and the solution of non-linear systems of equations, have been rewritten to make them easier to understand.
Newmark-type solvers for the wave equation have been added.

Delft, April 2008

Jos van Kan
Guus Segal
Fred Vermolen

Note to the second edition improvements

In this improved second edition the treatment of boundary conditions for all types of discretization methods has been extended. Periodical boundary conditions have been included. Furthermore,the description of the FEM has been simplified.

Delft, August 2014

Guus Segal
Fred Vermolen

Contents

1 Review of some basic mathematical concepts **1**
1.1 Preliminaries . . . 1
1.2 Global contents of the book . . . 1
1.3 Building blocks for mathematical modeling . . . 2
1.3.1 Gradient of a scalar . . . 2
1.3.2 Directional derivative . . . 4
1.3.3 Divergence of a vector field . . . 4
1.3.4 Gauss' divergence theorem . . . 5
1.3.5 Conservation laws . . . 7
1.4 Minimization . . . 8
1.4.1 Elastic string . . . 8
1.5 Preliminaries from linear algebra . . . 9
1.6 Some theorems used in the mathematical theory . . . 11
1.7 Summary of Chapter 1 . . . 13

2 A crash course in PDE's **15**
Objectives . . . 15
2.1 Classification . . . 15
2.1.1 Three or more independent variables . . . 17
2.2 Boundary and initial conditions . . . 17
2.2.1 Boundary conditions . . . 17
2.2.2 Initial conditions . . . 19
2.3 Existence and uniqueness of a solution . . . 19
2.3.1 The Laplacian operator . . . 19
2.3.2 The maximum principle and uniqueness . . . 20
2.3.3 Existence . . . 22
2.4 Examples . . . 22
2.4.1 Flows driven by a potential . . . 22
2.4.2 Convection-Diffusion . . . 23
2.4.3 Navier-Stokes equations . . . 23
2.4.4 Plane stress . . . 25
2.4.5 Biharmonic equation . . . 26
2.5 Summary of Chapter 2 . . . 27

3 Finite difference methods **29**
Objectives . . . 29
3.1 The cable equation . . . 29
3.1.1 Discretization . . . 30
3.1.2 Properties of the discretization matrix A . . . 31
3.1.3 Global error . . . 33
3.2 Some simple extensions of the cable equation . . . 34
3.2.1 Discretization of the diffusion equation . . . 34

3.2.2 Boundary conditions . . . 35
3.3 Singularly perturbed problems . . . 38
3.3.1 Analytical solution . . . 38
3.3.2 Numerical approximation . . . 39
3.4 The Laplacian equation on a rectangle . . . 42
3.4.1 Matrix vector form . . . 43
3.5 Boundary conditions extended . . . 45
3.5.1 Natural boundary conditions . . . 45
3.5.2 Dirichlet boundary conditions on non rectangular regions . . . 45
3.6 Global error estimate . . . 47
3.6.1 A discrete maximum principle . . . 47
3.6.2 Super solutions . . . 49
3.7 Boundary fitted coordinates . . . 51
3.8 Summary of Chapter 3 . . . 52

4 Finite volume methods **53**
Objectives . . . 53
4.1 Heat transfer with varying coefficient . . . 53
4.1.1 The boundaries . . . 55
4.1.2 Conservation . . . 56
4.1.3 Error in the temperatures . . . 56
4.2 The stationary diffusion equation in 2 dimensions . . . 57
4.2.1 Boundary conditions . . . 59
4.2.2 Boundary conditions in case of a cell centered method . . . 60
4.2.3 Boundary cells in case of a skewed boundary . . . 60
4.2.4 Error considerations in the interior . . . 62
4.2.5 Error considerations at the boundary . . . 62
4.3 Laplacian in general coordinates . . . 62
4.3.1 Discrete transformation from Cartesian to General coordinates 62
4.3.2 An example of finite volume integration in polar co-ordinates 64
4.3.3 Boundary conditions . . . 66
4.3.4 Error analysis . . . 66
4.4 Finite volumes on two component fields . . . 67
4.4.1 Staggered grids . . . 68
4.4.2 Boundary conditions . . . 69
4.5 Project: Stokes equations for incompressible flow . . . 71
4.6 Summary of Chapter 4. . . . 73

5 Minimization problems in physics **75**
Objectives . . . 75
5.1 Introduction . . . 75
5.1.1 Minimal potential energy . . . 75
5.1.2 Derivation of the differential equation . . . 76
5.2 A general one-dimensional problem with first order derivatives . . . 78
5.3 A simple two-dimensional case . . . 79
5.4 Examples of minimization problems . . . 81
5.4.1 Minimal surface problem . . . 81
5.4.2 Minimal potential energy . . . 82
5.4.3 Small displacement theory of elasticity (Plane stress) . . . 83
5.4.4 Loaded and clamped plate . . . 84
5.5 A two-dimensional problem . . . 85
5.6 Theoretical remarks . . . 85
5.6.1 Smoothness requirements . . . 85
5.6.2 Boundary conditions . . . 86

5.6.3 Weak formulation . . . 86
5.7 Exercises . . . 87
5.8 From PDE to minimization problem . . . 88
5.8.1 Introduction . . . 88
5.8.2 Linear problems with homogeneous boundary conditions . . . 88
5.8.3 Linear problems with non-homogeneous boundary conditions 90
5.8.4 Exercises . . . 92
5.9 Mathematical theory of minimization . . . 93
5.10 Summary of Chapter 5 . . . 95

6 The numerical solution of minimization problems 97
Objectives . . . 97
6.1 Ritz's method . . . 97
6.1.1 Introduction . . . 97
6.1.2 A simple one-dimensional example . . . 98
6.1.3 Some observations concerning the basis functions . . . 100
6.1.4 Mathematical theory: convergence of Ritz's method . . . 101
6.2 The finite element method in $\mathbb{R}^1$ . . . 103
6.2.1 Introduction . . . 103
6.2.2 The Poisson equation in $\mathbb{R}^1$ . . . 103
6.2.3 Numerical integration . . . 106
6.2.4 Boundary conditions . . . 108
6.2.5 Element matrices and element vectors . . . 110
6.2.6 Assembly of the large matrix and vector . . . 110
6.2.7 Boundary conditions and assembly . . . 112
6.2.8 Periodical boundary conditions . . . 113
6.2.9 The structure of finite element packages . . . 114
6.3 The finite element method in $\mathbb{R}^2$ . . . 114
6.3.1 The Poisson equation in $\mathbb{R}^2$ . . . 114
6.3.2 Linear elements in $\mathbb{R}^2$ . . . 116
6.3.3 Numerical integration in $\mathbb{R}^n$ . . . 118
6.3.4 Boundary conditions . . . 120
6.4 Theoretical remarks . . . 122
6.4.1 Smoothness requirements . . . 122
6.4.2 Mathematical theory of FEM . . . 124
6.4.3 Approximation errors . . . 125
6.5 Summary of Chapter 6 . . . 126

7 The weak formulation and Galerkin's method 127
Objectives . . . 127
7.1 The weak formulation for a symmetrical problem . . . 127
7.1.1 Introduction . . . 127
7.1.2 Natural boundary conditions . . . 128
7.1.3 Non-homogeneous essential boundary conditions . . . 129
7.1.4 Periodical boundary conditions . . . 130
7.2 The weak formulation for a non-symmetric problem . . . 130
7.3 Galerkin's method . . . 131
7.3.1 Introduction . . . 131
7.3.2 Galerkin's method applied to the convection-diffusion equation . . . 132
7.3.3 The convection-diffusion equation in $\mathbb{R}^1$ by finite elements . . . 133
7.3.4 The convection-diffusion equation in $\mathbb{R}^2$ by finite elements . . . 134
7.4 Petrov-Galerkin . . . 135
7.4.1 Introduction . . . 135

7.4.2 Upwinding in $\mathbb{R}^1$ by Petrov-Galerkin . . . 135
7.4.3 SUPG: stream line upwinding in $\mathbb{R}^2$ by Petrov-Galerkin . . . 137
7.5 An example of a system of coupled PDEs . . . 138
7.6 Mathematical theory . . . 141
7.7 Summary of Chapter 7 . . . 142

8 Extension of the FEM 145
Objectives . . . 145
8.1 (Straight) quadratic triangles . . . 145
8.2 Linear triangles revisited . . . 147
8.3 Quadrilaterals . . . 150
8.4 Curved quadratic triangles . . . 153
8.5 Application to the Stokes equations . . . 154
8.6 Circle symmetry . . . 156
8.7 Theoretical remarks . . . 158
8.8 Fourth order problems . . . 159
8.8.1 The clamped beam . . . 159
8.8.2 A simple example of the mixed approach . . . 161
8.9 Summary of Chapter 8 . . . 162

9 Solution of large systems of equations 163
Objectives . . . 163
9.1 Direct methods . . . 164
9.1.1 Introduction . . . 164
9.1.2 Gaussian elimination . . . 164
9.1.3 LU-decomposition . . . 166
9.1.4 Band method . . . 168
9.1.5 Profile method . . . 168
9.1.6 Renumbering techniques . . . 171
9.2 Generic iterative process. . . . 172
9.3 Defect correction . . . 172
9.3.1 Algorithm . . . 172
9.3.2 Convergence of defect correction . . . 172
9.3.3 Error estimate for defect correction . . . 173
9.3.4 Estimate of the spectral radius . . . 174
9.3.5 M-matrices . . . 174
9.4 Classical preconditioners . . . 175
9.4.1 Jacobi . . . 175
9.4.2 Gauss-Seidel . . . 175
9.4.3 Successive Overrelaxation SOR . . . 177
9.4.4 Block variations . . . 179
9.4.5 Operation count . . . 179
9.5 Krylov Space Methods . . . 180
9.5.1 Introduction . . . 180
9.5.2 The Krylov Space . . . 180
9.5.3 Conjugate Gradients . . . 181
9.5.4 CG algorithm . . . 182
9.5.5 Preconditioning . . . 184
9.5.6 Convergence . . . 185
9.5.7 Krylov space methods for non symmetric matrices. . . . 187
9.5.8 Preconditioners . . . 187
9.6 The multigrid algorithm . . . 190
9.6.1 A one-dimensional example . . . 191
9.6.2 Smooth and rough part of the spectrum . . . 192

9.6.3 Two grid algorithm . . . 193
9.6.4 From two grid to multigrid . . . 195
9.6.5 Convergence of the two grid algorithm . . . 195
9.6.6 Restriction and prolongation in two dimensions . . . 198
9.6.7 Concluding remarks about MG . . . 198
9.7 Non-linear equations . . . 198
9.7.1 Picard iteration . . . 199
9.7.2 Newton's method in more dimensions . . . 200
9.7.3 Starting values . . . 202
9.8 Summary of Chapter 9 . . . 203

10 The heat- or diffusion equation 205
Objectives . . . 205
10.1 A fundamental inequality . . . 205
10.2 Method of lines . . . 207
10.2.1 One dimensional examples . . . 208
10.2.2 Two-dimensional example . . . 209
10.3 Consistency of the spatial discretization . . . 210
10.4 Time integration . . . 212
10.5 Stability of the numerical integration . . . 213
10.5.1 Gershgorin's circle theorem . . . 214
10.5.2 Stability analysis of Von Neumann . . . 216
10.6 The accuracy of the time integration . . . 218
10.7 Conclusions for the method of lines . . . 219
10.8 Special difference methods for the heat equation . . . 220
10.8.1 The principle of the ADI method . . . 220
10.8.2 Formal description of the ADI method . . . 221
10.9 Summary of Chapter 10 . . . 223

11 The wave equation 225
Objectives . . . 225
11.1 A fundamental equality . . . 225
11.2 The method of lines . . . 227
11.2.1 The error in the solution of the system . . . 227
11.3 Numerical time integration . . . 229
11.4 Stability of the numerical integration . . . 230
11.5 Total dissipation and dispersion . . . 230
11.6 Direct time integration of the second order system . . . 232
11.7 The CFL criterion . . . 235
11.8 Summary of Chapter 11 . . . 237

12 The transport equation 239
Objectives . . . 239
12.1 Introduction . . . 239
12.2 Characteristics . . . 240
12.3 Some classical numerical procedures . . . 242
12.3.1 Central discretization and upwind discretization . . . 242
12.4 Mathematical theory for the transport equation . . . 250
12.4.1 Burgers equation . . . 251
12.4.2 The Buckley-Leverett equation . . . 253
12.5 Summary of Chapter 12 . . . 261
12.6 Appendix: requirements on flux-limiters . . . 262

13 Moving boundary problems **265**
Objectives 265
13.1 The formulation of a classical Stefan problem: ice and water 265
13.2 An exact (self-similar) solution for an unbounded region 267
13.3 Numerical methods 268
13.3.1 Moving grid methods 268
13.3.2 A fixed domain method: the level set method 274
13.3.3 Other applications of Stefan problems 280
13.4 Summary of Chapter 13 280

Chapter 1

Review of some basic mathematical concepts

1.1 Preliminaries

In this chapter we take a bird's eye view of the contents of the book. Furthermore we establish a physical interpretation of certain mathematical notions, operators and theorems. As a first application we formulate a general conservation law, since conservation laws are the back bone of physical modeling. Finally we treat some mathematical theorems, that will be used in the remainder of this book.

1.2 Global contents of the book

We first take a look at second order partial differential equations and their relation with various physical problems. Then we look at numerical methods for those equations. First we look at finite difference methods, of respectable age but still very much in use. Subsequently we take on finite volume methods, a typical engineers option, constructed for conservation laws. Finally we turn to finite element methods (FEM) which have gained tremendous popularity over the last decades. Before we can move to FEM, however, we have to delve a bit into minimization problems to provide a proper background. We shall show, that FEM may be considered as a special case of Ritz's method, a particular way of obtaining an approximate solution to a minimization problem. We shall establish a relation between minimization problems and partial differential equations. But not all PDEs can be formulated as a minimization problem and we shall consider a generalization that will enable us to apply the FEM also to those problems.

These methods generally leave us with a large set of linear or non-linear equations and we consider ways of how to solve them. In particular we shall pay some attention to efficient methods that are relatively young, like preconditioned Krylov space methods and multi-grid methods. The treatment can be only cursory but further references will be provided.

We also pay some attention to special methods for specific problems like heat and wave equations. Finally we consider transport equations. They do not fall within the previous context, being only first order, yet they are very important and deserve a chapter of their own. The last chapter will be dedicated to miscellaneous problems that fall outside the classification so far.

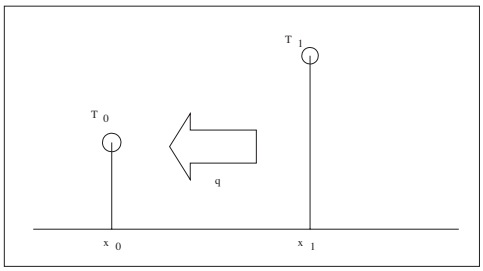


Figure 1.1: 1-dimensional heat flow.

1.3 Building blocks for mathematical modeling

Several mathematical concepts used in modeling are directly derived from a physical context. We shall consider a few of those and see how they can be used to formulate a fundamental mathematical model: conservation.

1.3.1 Gradient of a scalar

Given a scalar function, u, of two variables, differentiable with respect to both variables, then the gradient is defined as

$$\operatorname{grad} u = \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{pmatrix}. \tag{1.3.1}$$

Instead of the notation grad u also ∇u (pronounce: nabla u) is used. To get to the core of what a gradient really is, think of temperature. If you have a temperature difference between two points, then you get a flow of heat between those points that only will stop when the temperature difference has been annihilated. If the difference is bigger, the flow will be larger. If the points are closer together the flow will be larger. The simplest one dimensional model to reflect this is the following linear model. Let q be the generated flow, directly proportional to the temperature difference ΔT and inversely proportional to the distance Δx. This leads to:

$$q = -\lambda \frac{\Delta T}{\Delta x}, \tag{1.3.2}$$

where λ is a material constant, the *heat conduction* coefficient. The minus sign reflects the facts that

1. heat flows from high to low temperatures;
2. physicists hate negative constants.

In a continuous temperature field $T(x)$ we may take limits and obtain a flow that is derived from (driven by) the temperature:

$$q = -\lambda \frac{dT}{dx}. \tag{1.3.3}$$

How is this in more than one dimension? Suppose we have a two-dimensional temperature field $T(x, y)$ which we can represent nicely by considering the contour lines which for temperature are called *isotherms*, lines that connect points of equal temperature.

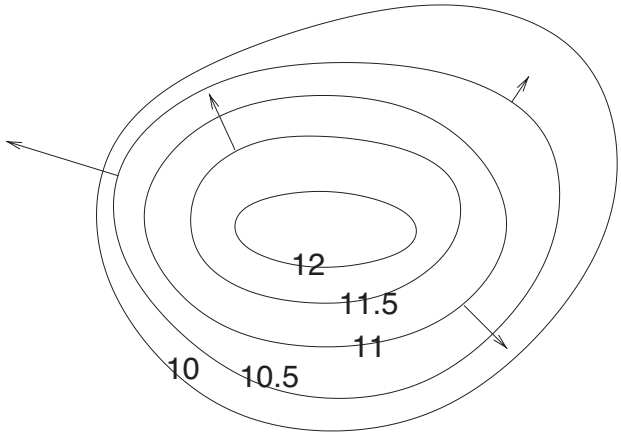


Figure 1.2: Isotherms.

Since there cannot be heat flow between points of equal temperature, the heat flow must be orthogonal to the contour lines at every point. Two vectors $\mathbf{v}$ and $\mathbf{w}$ are orthogonal if their inner product $(\mathbf{v}, \mathbf{w})$ vanishes. In other words: let $x(s), y(s)$ be a parameterization of a contour line and let $\begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$ be the components of the heat flow field. We then have:

$$q_1 \frac{dx}{ds} + q_2 \frac{dy}{ds} = 0, \tag{1.3.4}$$

at every point $x(s), y(s)$ of the isotherm, for all isotherms. Let us substitute the equation of an isotherm into the temperature field: $T(x(s), y(s))$. Doing this makes T a function of s only, *which is constant* because we are on an isotherm. In other words along an isotherm:

$$\frac{dT}{ds} = \frac{\partial T}{\partial x}\frac{dx}{ds} + \frac{\partial T}{\partial y}\frac{dy}{ds} = 0. \tag{1.3.5}$$

If we compare Equation (1.3.4) with (1.3.5) we see that these can only be satisfied if

$$\mathbf{q} = -\lambda \operatorname{grad} T. \tag{1.3.6}$$

For three dimensions you can tell basically the same story that also ends in Equation (1.3.6). This is known as *Fourier's law* and it is at the core of the theory of heat conduction.

Exercise 1.3.1 *(Darcy's Law). In ground water flow the velocities are very small, a few centimeters per day. This makes ground water flow basically a* hydrostatic *problem, in which the flow is driven by differences in hydrostatic pressure. This hydrostatic pressure depends linearly on the height of the ground water level h. So how does the flow* $\mathbf{q}$ *depend on h?* □

Exercise 1.3.2 *(Fick's Law) In diffusion the flow of matter,* $\mathbf{q}$*, is driven by differences in concentration c. Express* $\mathbf{q}$ *in c.* □

Scalar fields like T, h and c that drive a gradient flow field, $\mathbf{q}$, are called *potentials*. Not all flow fields are generated by the gradient of a potential. But those that are, are called *solenoidal* or *irrotational.*

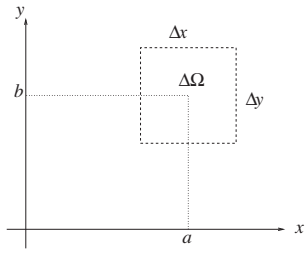


Figure 1.3: Square volume in river.

Exercise 1.3.3 *Let C be a closed contour in the x-y-plane and* $\mathbf{q}$ *a solenoidal vector field. Show that* $\int_C \mathbf{q} \cdot d\mathbf{s} = 0$. □

1.3.2 Directional derivative

In the previous paragraph we saw, how the temperature, T, changes along a curve $x(s), y(s)$. The actual value of $\frac{dT}{ds}$ depends on the parameterization. A natural parameterization is the *arc length* of the curve.
Note, that in that case $(\frac{dx}{ds})^2 + (\frac{dy}{ds})^2 = 1$. This forms the basis of the following definition:

Definition 1.3.1 *Let* $\mathbf{n}$ *be a* unit *vector, then the directional derivative of T in the direction of* $\mathbf{n}$ *is given by*

$$\frac{\partial T}{\partial n} = \frac{\partial T}{\partial x} n_1 + \frac{\partial T}{\partial y} n_2 = (\text{grad}\, T, \mathbf{n}) = (\mathbf{n} \cdot \nabla) T.$$

Exercise 1.3.4 *Compute the directional derivative of* $z = x^2 + y^3$ *in* $(1, 1)$ *in the direction* $(1, -1)$. *(Answer:* $-\frac{1}{2}\sqrt{2}$). □

Exercise 1.3.5 *For what value of* $\mathbf{n}$ *is the directional derivative precisely* $\frac{\partial T}{\partial x}$*?* □

1.3.3 Divergence of a vector field

The mathematical definition of divergence is equally uninspiring. Given a continuously differentiable vector field, $\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, the divergence of $\mathbf{v}$ is defined by:

$$\text{div}\, \mathbf{v} = \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y}. \tag{1.3.7}$$

For $\mathbb{R}^3$ you have the obvious generalization and there is also a nabla notation: $\text{div}\, \mathbf{v} = \nabla \cdot \mathbf{v}$. You will appreciate the correspondence of a genuine inner product of two vectors and the inner product of the "nabla vector" and a vector field. Take care, however. In a genuine inner product you can change the order of the vectors, in the divergence you cannot.

What is the physical meaning of divergence? You could think of a vector field as a river: at any place in the river the water has a certain velocity with direction and magnitude. Now consider a fixed rectangular volume in the river (see Figure 1.3).

Water is flowing in through the left and bottom wall and flowing out through the right and top wall. How much is flowing *in* through the left wall? If you think

about it, you will notice that the $y-$component of the velocity gives no contribution to the inflow, because that is parallel to the left wall. So the inflow through the left wall is equal to $v_{1L}\Delta y$, the outflow through the right wall $v_{1R}\Delta y$. By the same reasoning the inflow through the bottom equals $v_{2B}\Delta x$, the outflow through the top equals $v_{2T}\Delta x$. What's left behind? If the net outflow is larger than the net inflow we are losing matter in the volume, if on the other hand the net inflow is larger we're gaining. The net outflow out of control volume $\Delta\Omega$, in Figure 1.3 is given by

$$\begin{aligned}\Delta\phi(a,b) &= v_1(a+\frac{\Delta x}{2},b)\Delta y - v_1(a-\frac{\Delta x}{2},b)\Delta y + v_2(a,b+\frac{\Delta y}{2})\Delta x - v_2(a,b-\frac{\Delta y}{2})\Delta x \\ &= \Delta x\Delta y(\frac{v_1(a+\frac{\Delta x}{2},b)-v_1(a-\frac{\Delta x}{2},b)}{\Delta x} + \frac{v_2(a,b+\frac{\Delta y}{2})-v_2(a,b-\frac{\Delta y}{2})}{\Delta x}) \\ &= \Delta x\Delta y(\frac{\partial v_1}{\partial x}(\xi,b) + \frac{\partial v_2}{\partial x}(a,\eta)),\end{aligned} \tag{1.3.8}$$

for a $\xi \in (a-\frac{\Delta x}{2}, a+\frac{\Delta x}{2})$, $\eta \in (b-\frac{\Delta y}{2}, b+\frac{\Delta y}{2})$ from the Mean Value Theorem and continuity of the partial derivatives. This implies

$$\lim_{(\Delta x,\Delta y)\to(0.0)} \frac{\Delta\phi(a,b)}{\Delta x\Delta y} = \operatorname{div}\mathbf{v}(a,b). \tag{1.3.9}$$

From this formula, we see that $\operatorname{div}\mathbf{v}(a,b)$ is the outflow density (outflow per unit area) at point (a,b). Integration of the outflow density over an entire domain gives the total outflow. Since the total outflow can also be computed from evaluation of the flux over its boundary, we obtain a very important relation between the integral of the divergence of a vector-field over the domain and the integral of the flux over its boundary. This relation is formulated in terms of the Divergence Theorem, which we shall state in the next subsection.

Exercise 1.3.6 *Explain that for an* incompressible *flow field,* $\mathbf{u}$*, we must have* $\operatorname{div}\mathbf{u} = 0$. □

Exercise 1.3.7 *Derive in the same way as above that divergence is an outflow density in* $\mathbb{R}^3$. □

1.3.4 Gauss' divergence theorem

In the previous section, we informally derived the Divergence Theorem, which was initially proposed by Gauss. In words: the outflow density integrated over an arbitrary volume gives the total outflow out of this volume. But this is mathematics, so we have to be more precise.

Theorem 1.3.1 *Gauss' divergence theorem.*
Let Ω *be a bounded domain in* $\mathbb{R}^2$ *(*$\mathbb{R}^3$*) with piecewise smooth boundary* Γ*. Let* $\mathbf{n}$ *be the outward normal and* $\mathbf{v}$ *a continuously differentiable vector field. Then*

$$\int_\Omega \operatorname{div}\mathbf{v}\, d\Omega = \int_\Gamma \mathbf{v}\cdot\mathbf{n}\, d\Gamma. \tag{1.3.10}$$

□

Remark

1. The expression $\mathbf{v} \cdot \mathbf{n}$ is the outward normal component of the vector-field, $\mathbf{v}$, with respect to the boundary. If this quantity is positive you have outflow, otherwise inflow.

2. Any good book on multivariate analysis will have a proper proof of Gauss' theorem. (See for instance [2] or [35]). A good insight will be obtained however, by subdividing the region Ω in small rectangles and using (1.3.8). Note in particular, that the common side (plane in $\mathbb{R}^3$) of two neighboring volumes cancel: what flows out of one flows into the other. The proof is finalized by taking a limit $\Delta x, \Delta y \to 0$ (contraction) in the Riemann sum.

The Divergence theorem has many important implications and these implications are used frequently in various numerical methods, such as the finite element method. First, one can use the component-wise product rule for differentiation to arrive at the following theorem

Theorem 1.3.2 *For a continuously differentiable scalar field, c, and vector field, $\mathbf{u}$, we have*

$$div\ (c\mathbf{u}) = grad\ c \cdot \mathbf{u} + c\ div\ \mathbf{u}. \tag{1.3.11}$$

Exercise 1.3.8 *Prove Theorem 1.3.2.*

As a result of this assertion, one can prove the following theorem.

Theorem 1.3.3 *Green's Theorem*
For a sufficiently smooth c, $\mathbf{u}$, we have

$$\int_\Omega c\mathrm{div}\,\mathbf{u}\, d\Omega = -\int_\Omega (\mathrm{grad}\, c)\cdot \mathbf{u} d\Omega + \oint_\Gamma c\mathbf{u}\cdot\mathbf{n}\, d\Gamma. \tag{1.3.12}$$

Exercise 1.3.9 *Prove Theorem 1.3.3.*

By the use of Theorem 1.3.3, the following assertion can be demonstrated:

Theorem 1.3.4 *Partial integration in 2 D*
For sufficiently smooth scalar functions ϕ and ψ, we have;

$$\int_\Omega \phi\frac{\partial\psi}{\partial x}\, d\Omega = -\int_\Omega \frac{\partial\phi}{\partial x}\psi\, d\Omega + \oint_\Gamma \phi\psi n_1 d\Gamma, \tag{1.3.13}$$

and

$$\int_\Omega \phi\frac{\partial\psi}{\partial y}\, d\Omega = -\int_\Omega \frac{\partial\phi}{\partial y}\psi\, d\Omega + \oint_\Gamma \phi\psi n_2 d\Gamma. \tag{1.3.14}$$

Exercise 1.3.10 *Prove Theorem 1.3.4.*
Hint: choose an appropriate vector field, $\mathbf{u}$, in the previous exercise. □

1.3.5 Conservation laws

Let us consider some flow field, $\mathbf{u}$, in a volume V with boundary Γ. If the net inflow into this volume is positive *something* in this volume must increase (whatever it is). That is the basic form of a conservation law:

$$\frac{\partial}{\partial t}\int_V S\,dV = -\int_\Gamma \mathbf{u}\cdot\mathbf{n}\,d\Gamma + \int_V f(t,\mathbf{x})\,dV. \tag{1.3.15}$$

The term $f(t,\mathbf{x})$ is a production *density*, it tells how much S is produced any time, any place within V. The boundary integral describes the net inflow into V (mark the minus sign). The flow field, $\mathbf{u}$, is also called the *flux vector* of the model. S just like f has the dimension of a *density.* Since Equation (1.3.15) has to hold for every conceivable volume in the flow field we may formulate a *point wise* conservation law as follows. First we apply Gauss' Theorem 1.3.10 to Equation (1.3.15) to obtain

$$\frac{\partial}{\partial t}\int_V S\,dV = -\int_V \operatorname{div}\mathbf{u}\,dV + \int_V f(t,\mathbf{x})\,dV. \tag{1.3.16}$$

Subsequently we invoke the mean-value theorem of integral calculus for each integral separately, assuming all integrands are continuous:

$$\frac{\partial S}{\partial t}(\mathbf{x}_1) = -\operatorname{div}\mathbf{u}(\mathbf{x}_2) + f(t,\mathbf{x}_3). \tag{1.3.17}$$

Observe that we have divided out a factor $\int_V dV$ and that $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ all lie within V. Finally we let V contract to a single point $\mathbf{x}$ to obtain a point wise conservation law in the form of a PDE:

$$\frac{\partial S}{\partial t} = -\operatorname{div}\mathbf{u} + f(t,\mathbf{x}). \tag{1.3.18}$$

This is all rather abstract, so let us look at an example.

1.3.5.1 Example: Heat flow

In heat flow, conservation law (1.3.18) takes the form

$$\frac{\partial h}{\partial t} = -\operatorname{div}\mathbf{q} + f(t,\mathbf{x}), \tag{1.3.19}$$

in which h is the heat density, $\mathbf{q}$ the heat flux vector and f the production density. Remember, that all quantities in such a point wise conservation law are densities. The heat density, h, stored in a material can be related to the materials (absolute) temperature T:

$$h = \rho c T, \tag{1.3.20}$$

in which ρ is the density and c the heat capacity of the material. These material properties have to be measured. As we already saw in Section 1.3.1 the heat flow, $\mathbf{q}$, is driven by the temperature gradient: $\mathbf{q} = -\lambda\,\nabla T$. This enables us to formulate everything in terms of temperature. Substituting this all we get:

$$\frac{\partial \rho c T}{\partial t} = \operatorname{div}\lambda\,\operatorname{grad} T + f(t,\mathbf{x}). \tag{1.3.21}$$

If ρ, c are constant throughout the material and if there is no internal heat production this transforms into the celebrated heat conduction equation:

$$\frac{\partial T}{\partial t} = \operatorname{div}(k\operatorname{grad} T), \tag{1.3.22}$$

with $k = \lambda/(\rho c)$.

1.4 Minimization

Another way of deriving models is by looking at the potential energy. This is most often used in mechanical problems, but can also be used in different contexts. An equilibrium state can be found by minimizing that potential energy. We also meet minimization problems in optics (optical length) and economics (cost).

1.4.1 Elastic string

As an example consider an elastic string fixed in $(0,0)$ and $(0,1)$, see Figure 1.4.

Without load, the string is undeformed: $u(x) = 0$. When we apply a load f the string deforms. What is the potential energy of the deformed string? First of all, there is an elastic energy proportional to the increase in length: $\Delta P_e = k\Delta L$. Over a small interval Δx this increase amounts to

$$\Delta L = \sqrt{\Delta x^2 + \Delta u^2} - \Delta x. \tag{1.4.1}$$

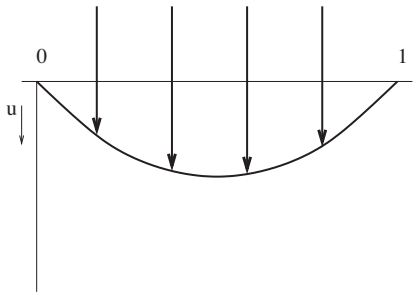


Figure 1.4: Deformed elastic string.

When the inclination $\Delta u/\Delta x$ is small (this is true in a realistic problem), this is approximately equal to

$$\Delta L = \Delta x(1 + \tfrac{1}{2}(\frac{\Delta u}{\Delta x})^2) - \Delta x, \tag{1.4.2}$$

$$= \tfrac{1}{2}\left(\frac{\Delta u}{\Delta x}\right)^2 \Delta x. \tag{1.4.3}$$

The work done by the load f per fragment Δx equals $\Delta W = uf\Delta x$, assuming we take the positive u–axis pointing down. The potential energy per fragment Δx then is given by $\Delta P_e - \Delta W$ and the potential energy over the whole string is obtained by integrating over the whole interval $(0,1)$:

$$P = P_e - W = \int_0^1 (\tfrac{1}{2}k\left(\frac{du}{dx}\right)^2 - uf)\,dx. \tag{1.4.4}$$

So any (sufficiently smooth) function u satisfying $u(0) = 0$ and $u(1) = 0$ yields a potential energy. The solution to the mechanical problem is that function u for which the potential energy P is minimal. In Chapter 5 we shall see how to deal with this.

Exercise 1.4.1 *Show by Taylor's theorem that* $\sqrt{1+x} = 1 + \frac{1}{2}x + O(x^2)$. □

1.5 Preliminaries from linear algebra

Let $\mathbf{x}, \mathbf{y}$ be vectors in $\mathbb{C}^n$. In this chapter we use the inner product $(\mathbf{x}, \mathbf{y})$ defined by

$$(\mathbf{x}, \mathbf{y}) = \sum_j x_j \bar{y}_j = \mathbf{x}^T \bar{\mathbf{y}} = \overline{(\mathbf{y}, \mathbf{y})}, \tag{1.5.1}$$

where $\bar{\mathbf{y}}$ is the conjugate complex of $\mathbf{y}$. Further $||x|| = \sqrt{(x, x)}$.

Definition 1.5.1 *Let A be a $n \times n$ matrix. Let λ be a complex number and $\mathbf{v}$ a complex vector such that*

$$A\,\mathbf{v} \;=\; \lambda\,\mathbf{v}, \qquad \mathbf{v} \neq 0, \tag{1.5.2}$$

then λ is called an eigenvalue and $\mathbf{v}$ an eigenvector of A.

Theorem 1.5.1 *All eigenvalues of a real symmetrical matrix are real, and eigenvectors corresponding to different eigenvalues are orthogonal.*

Proof Multiplication of Equation (1.5.2) by the vector $\bar{\mathbf{v}}^T$:

$$\bar{\mathbf{v}}^T A\mathbf{v} \;=\; \lambda \bar{\mathbf{v}}^T \mathbf{v}. \tag{1.5.3}$$

$\bar{\mathbf{v}}^T A\mathbf{v}$ is real, since

$$\overline{(\bar{\mathbf{v}}^T A\mathbf{v})}^T \;=\; \bar{\mathbf{v}}^T \bar{A}^T \mathbf{v} \;=\; \bar{\mathbf{v}}^T A\mathbf{v} \qquad \text{A symmetrical real.} \tag{1.5.4}$$

In the same way $\bar{\mathbf{v}}^T \mathbf{v}$ is real, hence λ is real.
Further $(\mathbf{v}_i, A\mathbf{v}_j) = (A\mathbf{v}_i, \mathbf{v}_i)$, where $\mathbf{v}_i$ and $\mathbf{v}_j$ are eigenvectors associated with eigenvalues λ_i, λ_j $(\lambda_i \neq \lambda_j)$. This implies $\lambda_j(\mathbf{v}_i, \mathbf{v}_j) = \lambda_i(\mathbf{v}, \mathbf{v}_j)$. Since $\lambda_i \neq \lambda_j$ it immediately follows that $(\mathbf{v}_i, \mathbf{v}_j) = 0$. □

Definition 1.5.2 *A matrix A is called skewed symmetric if $A^T = -A$.*

Theorem 1.5.2 *All eigenvalues of a real skewed symmetrical matrix are purely imaginary.*

Exercise 1.5.1 *Prove Theorem (1.5.2) analogously to the proof of Theorem (1.5.1).* □

Definition 1.5.3 *The Rayleigh quotient, $R(A, \mathbf{x})$, of a symmetrical matrix A is given by:*

$$R(A, \mathbf{x}) = \frac{(\mathbf{x}, \mathbf{A}\mathbf{x})}{(\mathbf{x}, \mathbf{x})}. \tag{1.5.5}$$

Theorem 1.5.3 *If $\mathbf{x}$ is an eigenvector of A, then R is equal to the corresponding eigenvalue.*

Proof
Let λ_i be an eigenvalue of A, then

$$A\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{1.5.6}$$

$$R(A, \mathbf{v}_i) \;=\; \frac{(\mathbf{v}, A\mathbf{v})}{(\mathbf{v}, \mathbf{v})} \;=\; \lambda_i. \tag{1.5.7}$$

□

Theorem 1.5.4 *For the Rayleigh quotient, $R(A, \mathbf{x})$, of a symmetrical matrix A we have*

$$\lambda_1 \leq R(A, \mathbf{x}) \leq \lambda_n \quad \forall \mathbf{x}, \tag{1.5.8}$$

with λ_1 the smallest and λ_n the largest eigenvalue of A.

Exercise 1.5.2 *Prove Theorem (1.5.4).*
Hint: use the fact that the eigenvectors of a symmetric matrix form an orthonormal basis of the space $\mathbb{R}^n$ *and expand the vector* $\mathbf{x}$ *as a linear combination of these eigenvectors.* □

Definition 1.5.4 *A matrix* A *is called positive if* $(\mathbf{x}, A\mathbf{x}) \geq 0, \ \forall \mathbf{x} \in \mathbb{R}^n$.

Definition 1.5.5 *A matrix* A *is called positive definite if* $\exists\ \alpha > 0$ *such that* $(\mathbf{x}, A\mathbf{x}) \geq \alpha ||\mathbf{x}||^2, \ \forall \mathbf{x} \in \mathbb{R}^n$.

Theorem 1.5.5

- *If* A *is positive, then its eigenvalues are non-negative.*
- *If* A *is positive definite, then its eigenvalues are positive.*

Theorem 1.5.6 *Let* A *be symmetric.*

- *If the eigenvalues of* A *are non-negative, then* A *is positive.*
- *If the eigenvalues of* A *are positive, then* A *is positive definite.*

Proof of Theorem 1.5.5

- Let $(\lambda, \mathbf{v})$ be an eigenpair of A, then by definition $(\mathbf{v}, A\mathbf{v}) = \lambda(\mathbf{v}, \mathbf{v}) \geq 0$. Since $(\mathbf{v}, \mathbf{v}) > 0$, we have $\lambda \geq 0$.
- Let $(\lambda, \mathbf{v})$ be an eigenpair of A, and let A be positive definite, then $\exists\ \alpha > 0$ such that $(\mathbf{v}, A\mathbf{v}) = \lambda(\mathbf{v}, \mathbf{v}) \geq \alpha(\mathbf{v}, \mathbf{v}) > 0$.
  Since $(\mathbf{v}, \mathbf{v}) > 0$, this immediately implies $\lambda > 0$.

□

Proof of Theorem 1.5.6

- We expand any vector $\mathbf{x}$ as a linear combination of the eigenvectors of A. Symmetry of A enables this procedure. Then

$$\mathbf{x} = \sum_j c_j \mathbf{v}_j. \tag{1.5.9}$$

  This implies with orthogonality of the eigenvectors

$$||\mathbf{x}||^2 = (\mathbf{x}, \mathbf{x}) = \sum_j c_j^2. \tag{1.5.10}$$

  Hence, we get similarly

$$(\mathbf{x}, A\mathbf{x}) = (\sum_j c_j \mathbf{v}_j, \sum_k \lambda_k c_k \mathbf{v}_k) = \sum_j c_j^2 \lambda_j. \tag{1.5.11}$$

  If $\lambda_j \geq 0, \ \forall j$, then this implies

$$(\mathbf{x}, A\mathbf{x}) \geq 0, \tag{1.5.12}$$

  which proves the first assertion.

- The second assertion follows from

$$(\mathbf{x}, A\mathbf{x}) = \sum_j c_j^2 \lambda_j \geq \sum_j c_j^2 \lambda_{min}, \tag{1.5.13}$$

  where $0 < \lambda_{min} = \min_j \lambda_j$.
  Since $(\mathbf{x}, \mathbf{x}) = \sum_j c_j^2$, we get

$$(\mathbf{x}, A\mathbf{x}) \geq \lambda_{min}(\mathbf{x}, \mathbf{x}), \quad \lambda_{min} > 0. \tag{1.5.14}$$

  Hence A is positive definite.

□

As a consequence the Rayleigh quotient of a positive matrix is non-negative, whereas the Rayleigh quotient of a positive definite matrix is positive.

The following theorem can be of great help in estimating bounds for eigenvalues of matrices. This is for example useful in stability analysis.

Theorem 1.5.7 (Gershgorin)
For all eigenvalues λ of the matrix A holds:

$$|\lambda - a_{kk}| \leq \sum_{i=1_{i\neq k}}^{N} |a_{ki}|. \tag{1.5.15}$$

Remark:
Eigenvalues may be complex valued in general and for complex eigenvalues $\lambda = \mu + i\nu$, the absolute value is the *modulus:* $|\lambda| = \sqrt{\mu^2 + \nu^2}$. So the eigenvalues are located within a circle in the complex plane and that is the reason why the theorem is also often referred to as Gershgorin's *circle* theorem. But for symmetric A, the eigenvalues of A are real-valued.

Proof
Let λ be an eigenvalue of the eigenvalue problem with corresponding eigenvector, $\mathbf{v}$, then, $A\mathbf{v} = \lambda\mathbf{v}$, and for each row, p, this gives

$$\sum_i a_{pi} v_i = \lambda v_p, \quad p = 1, \ldots, N. \tag{1.5.16}$$

Let v_k be the component of $\mathbf{v}$ with the largest modulus. For this index k we have

$$\lambda - a_{kk} = \sum_{i \ i\neq k} a_{ki} \frac{v_i}{v_k}, \tag{1.5.17}$$

and because $|v_i/v_k| \leq 1$, we get

$$|\lambda - a_{kk}| \leq \sum_{i \ i\neq k} |a_{ki}|. \tag{1.5.18}$$

This proves the theorem. □

Definition 1.5.6 *A matrix,* **A**, *is called a band-matrix if all elements, a_{ij}, outside a certain band are equal to zero. In formula: $a_{ij} = 0$* if *$i - j > b_1$* or *$j - i > b_2$.*
The bandwidth *of the matrix is in that case $b_1 + b_2 + 1$.*

1.6 Some theorems used in the mathematical theory

In some of the proofs used in this book we shall use the following theorems.
Let $L^2(\Omega) := \{u : \Omega \to \mathbb{R} : \int\limits_{\Omega} |u^2| d\Omega < \infty \}$.

Theorem 1.6.1 *Inequality of Poincaré (Friedrichs)*
Let $\Omega \subset \mathbb{R}^m$, $u \in H^1(\Omega) = \{u \in L^2(\Omega) | \frac{\partial u}{\partial x_1}, ..., \frac{\partial u}{\partial x_m} \in L^2(\Omega)\}$ and $u|_\Gamma = 0$, then $\exists \ K > 0$ such that

$$\int\limits_{\Omega} \sum_{i=1}^{m} (\frac{\partial u}{\partial x_i})^2 \, d\Omega \geq K \int\limits_{\Omega} u^2 \, d\Omega. \tag{1.6.1}$$

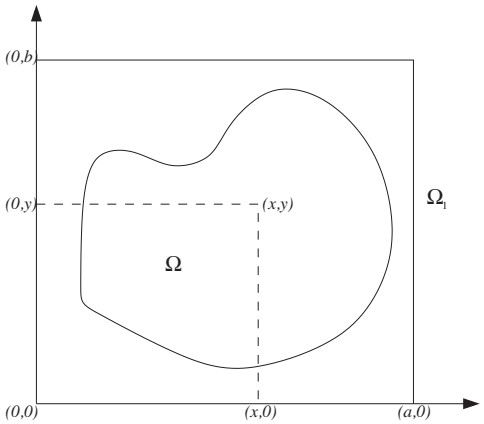


Figure 1.5: 2-dimensional region.

Proof We shall prove the theorem for $m = 2$.

By shifting coordinates we may assume that $(x, y) \in \Omega$ implies $x > 0$ and $y > 0$. The region Ω is enclosed by a rectangle Ω_1, given by $(a, 0) \times (0, b)$ as in Figure (1.5). Since $u(x, y) \in C^1(\Omega)$ and $u(x, y) = 0$ on Γ, we may extend $u(x, y)$ continuously to the whole domain Ω_1 by defining

$$u(x, y) = 0, \quad (x, y) \in \Omega_1 \backslash \Omega. \tag{1.6.2}$$

Let (x_1, y_1) be an arbitrary point in Ω_1. Then

$$u(x_1, y_1) - u(0, y_1) = \int_0^{x_1} \frac{\partial u(x, y_1)}{\partial x}\, dx, \tag{1.6.3}$$

$$u(0, y_1) = 0 \quad \text{follows from Figure (1.5).} \tag{1.6.4}$$

According to Cauchy-Schwartz we have:

$$\{\int_\Omega uv\, d\Omega\}^2 < \int_\Omega u^2\, d\Omega \int_\Omega v^2\, d\Omega. \tag{1.6.5}$$

Hence

$$u^2(x_1, y_1) = \{\int_0^{x_1} \frac{\partial u(x, y_1)}{\partial x}\, dx\}^2 \leq x_1 \int_0^{x_1} \{\frac{\partial u(x, y_1)}{\partial x}\}^2\, dx \tag{1.6.6}$$

$$\leq a \int_0^{a} (\frac{\partial u(x, y_1)}{\partial x})^2\, dx. \tag{1.6.7}$$

Integration of Equation (1.6.6) over Ω_1 gives

$$\int_{\Omega_1} u^2(x, y)\, d\Omega \leq a^2 \int_{\Omega_1} (\frac{\partial u}{\partial x})^2\, d\Omega \leq a^2 \int_{\Omega_1} (\frac{\partial u}{\partial x})^2 + (\frac{\partial u}{\partial y})^2\, d\Omega \tag{1.6.8}$$

This proves the theorem with $K = 1/a^2$. □

Exercise 1.6.1 *Prove Theorem (1.6.1) with $K = 1/b^2$.* □

From the proof of Theorem (1.6.1) and Exercise (1.6.1) if follows that K is overestimated by $K = \max(1/a^2, 1/b^2)$.
This theorem is used to prove that the Laplace operator is positive definite, when using Dirichlet or Robin boundary conditions. We will specify these boundary conditions in Chapter 2.

Definition 1.6.1 *A Banach space is a complete vector space defined over the real or complex numbers provided with a norm.*

Definition 1.6.2 *A Hilbert space is a Banach space provided with an inner product which defines the norm of the space.*

Definition 1.6.3 *A bilinear form $a(u, v)$ in V has the following properties*

- $a(u, v + w) = a(u, v) + a(u, w), \forall u, v, w, \in V,$
- $a(\lambda u, v) = \lambda a(u, v), \forall \lambda \in \mathbb{R}, \forall u, v \in V.$

Definition 1.6.4 *Let $a(.,.)$ be a bilinear form in V, then*

- *$a(.,.)$ is bounded if $\exists\, C > 0$ such that $|a(u, v)| \leq C||u||_V||v||_V, \forall u, v \in V.$*
- *$a(.,.)$ is coercive if $\exists\, C > 0$ such that $a(u, u) \geq C||u||_V^2, \forall u, v \in V.$*

The next theorem is used in existence and uniqueness proofs.

Theorem 1.6.2 *Lax-Milgram*
Let V be a Hilbert space and let $a(.,.)$ be a coercive and bounded, bilinear form on V. Further let $f \in V'$, where V' denotes the set (space) of linear functionals on V, then there is a unique solution $u \in V$, such that

$$a(u, v) = f(v), \forall v \in V. \tag{1.6.9}$$

This solution satisfies

$$||u|| \leq \frac{1}{c}||f||_{V'}. \tag{1.6.10}$$

For a proof of this theorem see for example [22].

1.7 Summary of Chapter 1

In this chapter we have seen the importance of conservation in the development of models and the role the mathematical operators *divergence* and *gradient* play in that development. We have met the famous divergence theorem of Gauss as an expression of global conservation.

We have looked at various applications deriving from conservation: heat transfer, diffusion and ground water flow. We concluded the chapter with an example of minimization as an instrument to derive a physical model. Besides that some standard mathematical theorems have been reviewed.

Chapter 2

A crash course in PDE's

Objectives

In the previous chapter we looked at PDE's from the *modeling* point of view, but now we shall look at them from a *mathematical* angle. Apparently you need partial derivatives and at least *two* independent variables to speak of a PDE (with fewer variables you would have an ordinary differential equation), so the simplest case to consider is a PDE with exactly two independent variables. A second aspect is the *order* of the PDE, that is the order of the highest derivative occurring in it. First order PDE's are a class of their own: the *transport* equations. We shall consider them in Chapter 11. In this chapter we shall take a look at second order PDE's and show that (for two independent variables) they can be classified into three types. We shall provide boundary and initial conditions that are needed to guarantee a unique solution and we will consider a few properties of the solutions to these PDE's. We conclude the chapter with a few examples of second and fourth order equations that occur in various fields of physics and technology.

2.1 Classification

Consider a second order PDE in two independent variables *with constant coefficients.*

$$a_{11}\frac{\partial^2 u}{\partial x^2} + 2a_{12}\frac{\partial^2 u}{\partial x \partial y} + a_{22}\frac{\partial^2 u}{\partial y^2} + b_1\frac{\partial u}{\partial x} + b_2\frac{\partial u}{\partial y} + cu + d = 0. \tag{2.1.1}$$

By *rotating* the coordinate system we can make the term with the mixed second derivative vanish. This is the basis of the classification. To carry out this rotation, we keep in mind that

$$(\frac{\partial}{\partial x}, \frac{\partial}{\partial y})A\begin{pmatrix}\frac{\partial u}{\partial x}\\ \frac{\partial u}{\partial y}\end{pmatrix} = a_{11}\frac{\partial^2 u}{\partial x^2} + 2a_{12}\frac{\partial^2 u}{\partial x \partial y} + a_{22}\frac{\partial^2 u}{\partial y^2}, \tag{2.1.2}$$

where $A = \begin{pmatrix}a_{11} & a_{12}\\ a_{12} & a_{22}\end{pmatrix}$. Since A is symmetric, we can factorize A into $A = Q\Lambda Q^T$, where $\Lambda = \text{diag}\,(\alpha_{11}, \alpha_{22})$, in which α_{11} and α_{22} are eigenvalues of A. The columns of Q are the normalized (with length one) eigenvectors of A. Note that $Q^T = Q^{-1}$

due to symmetry of A. Hence, one obtains from equation (2.1.2)

$$\begin{aligned} a_{11}\frac{\partial^2 u}{\partial x^2} + 2a_{12}\frac{\partial^2 u}{\partial x \partial y} + a_{22}\frac{\partial^2 u}{\partial y^2} = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}) Q \Lambda Q^T \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{pmatrix} = \\ (\frac{\partial}{\partial \xi}, \frac{\partial}{\partial \eta}) \Lambda \begin{pmatrix} \frac{\partial u}{\partial \xi} \\ \frac{\partial u}{\partial \eta} \end{pmatrix} = \alpha_{11}\frac{\partial^2 u}{\partial \xi^2} + \alpha_{22}\frac{\partial^2 u}{\partial \eta^2}. \end{aligned} \tag{2.1.3}$$

The resulting equation will look like:

$$\alpha_{11}\frac{\partial^2 u}{\partial \xi^2} + \alpha_{22}\frac{\partial^2 u}{\partial \eta^2} + \beta_1\frac{\partial u}{\partial \xi} + \beta_2\frac{\partial u}{\partial \eta} + cu + d = 0. \tag{2.1.4}$$

Exercise 2.1.1 *Show that $a_{12}^2 - a_{11}a_{22} < 0$, $a_{12}^2 - a_{11}a_{22} = 0$ and $a_{12}^2 - a_{11}a_{22} > 0$, respectively correspond to $\alpha_{11}\alpha_{22} > 0$, $\alpha_{11}\alpha_{22} = 0$ and $\alpha_{11}\alpha_{22} < 0$ (these cases correspond to the situations in which the eigenvalues of A have the same sign, one of the eigenvalues of A is zero and opposite signs of the eigenvalues of A respectively).* □

There are three possibilities:

1. $\alpha_{11}\alpha_{22} > 0$. (I.e. both coefficients have the same sign) The equation is called *elliptic*. An example of this case is *Poisson's equation*

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f. \tag{2.1.5}$$

2. $\alpha_{11}\alpha_{22} < 0$. (I.e. both coefficients have opposite sign) The equation is called *hyperbolic*. An example of this case is the wave equation

$$\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 0. \tag{2.1.6}$$

3. $\alpha_{11}\alpha_{22} = 0$. (I.e. either coefficient vanishes). The equation is called *parabolic*. An example is the heat equation in one space dimension:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}. \tag{2.1.7}$$

Exercise 2.1.2 *Let $D = a_{11}a_{22} - a_{12}^2$. Show that the condition for hyperbolic, parabolic or elliptic in the original coefficients a_{ij} is given by $D < 0$, $D = 0$ and $D > 0$ respectively. Use the result of Exercise 2.1.1.* □

For the classification only the second order part of the PDE is important. The three different types have very different physical and mathematical properties. To begin with, elliptic equations are time-independent and often describe an equilibrium. Parabolic and hyperbolic equations are time-dependent: they describe the evolution in time or *transient behavior* of a process. The difference in nature between parabolic and hyperbolic equations is that the first class describes an evolution towards an equilibrium, whereas the second class mimics wave phenomena.

This classification strictly spoken holds only for equations with constant coefficients. For equations with varying coefficients this classification only holds *locally*. If the coefficients depend on the solution itself the type of equation may depend on the solution itself.

2.1.1 Three or more independent variables

In this section, we consider a generalization of the simple classification. The general second order part of a *quasi-linear* PDE in $N > 2$ independent variables is given by:

$$\sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}\frac{\partial^2 u}{\partial x_i \partial x_j}. \tag{2.1.8}$$

$a_{ij} = a_{ji}$ and in a way similar to that in the previous section one may remove the mixed derivatives. This leads to:

$$\sum_{i=1}^{N} \alpha_{ii}\frac{\partial^2 u}{\partial \xi_i^2}. \tag{2.1.9}$$

We treat the following cases in this book:

1. All α_{ii} have the same sign. In this case all independent variables ξ_i are space variables. The equation is called *elliptic*. Example: 3D Laplacian
$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = 0. \tag{2.1.10}$$
2. Exactly one α_{ii}, say α_{11} has different sign from the rest. In this case ξ_1 is a time variable, all other ξ_i are space variables. The equation is called *hyperbolic*. Example: 3D Wave equation
$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}. \tag{2.1.11}$$
3. Exactly one α_{ii} vanishes, say α_{11}. Then ξ_1 is a time variable and the equation is called *parabolic*. Example: 3D Heat equation
$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}. \tag{2.1.12}$$

Exercise 2.1.3 *If A is a symmetric $n \times n$ matrix there exists a real unitary matrix C such that $C^T AC = \Lambda$. Λ is a diagonal matrix containing the eigenvalues of A on the diagonal. Show that the substitution $\xi = C^T\mathbf{x}$ eliminates the mixed derivatives in the differential operator* div A grad u. □

2.2 Boundary and initial conditions

To ensure a unique solution to our PDE we need to prescribe appropriate boundary conditions and in time-dependent problems we need initial conditions too. We will just consider here second order PDE's because the considerations for first order PDE's are very different and will be considered in Chapter 11.

2.2.1 Boundary conditions

Consider the bounded region in $\mathbb{R}^2$, Ω with boundary Γ in Figure 2.1. Let Γ consist of three *disjoint* pieces Γ_0, Γ_1 and Γ_2. For an elliptic equation of the form

$$\text{div}\, k\, \text{grad}\, u = f, \tag{2.2.1}$$

with $k > 0$ $\forall \mathbf{x} \in \overline{\Omega}$, the following boundary conditions guarantee a unique solution:

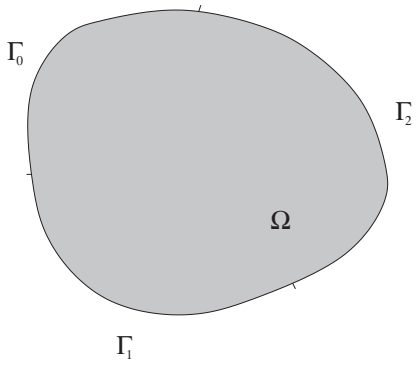


Figure 2.1: The bounded region Ω.

1.
$$u = g_0(\mathbf{x}), \quad \mathbf{x} \in \Gamma_0, \tag{2.2.2}$$
the *Dirichlet boundary condition*.

2.
$$k\frac{\partial u}{\partial n} = g_1(\mathbf{x}), \quad \mathbf{x} \in \Gamma_1, \tag{2.2.3}$$
the *Neumann boundary condition*.

3.
$$k\frac{\partial u}{\partial n} + \sigma u = g_2(\mathbf{x}), \quad \sigma \geq 0, \, \mathbf{x} \in \Gamma_2, \tag{2.2.4}$$
the *Robin*, *radiation*, *kinetic* or *mixed* boundary condition.

These boundary conditions do not have to occur together, each (but not all) of Γ_0, Γ_1 or Γ_2 could be empty. Because the pieces are disjoint exactly *one* boundary condition occurs on each point of the boundary. There is a small problem if $\Gamma = \Gamma_1$ in other words if there is a Neumann boundary condition on all of the boundary. Physically this may be understood, as that the *inflow* at each point of the boundary is prescribed. And since we have an equilibrium the net inflow over the whole region must be annihilated inside or the net outflow must be produced inside. This result is stated in mathematical form in the following theorem.

Theorem 2.2.1 *If a Neumann boundary condition is given on all of* Γ*, then the solution* u *of Equation* (2.2.1) *is determined up to an additive constant only. Moreover the following* compatibility condition *must be satisfied:*

$$\int_\Gamma g_1 \, d\Gamma = \int_\Omega f \, d\Omega \tag{2.2.5}$$

□

Exercise 2.2.1 *Prove Theorem 2.2.1. Use Gauss' divergence theorem on the PDE. It is not necessary to prove the only part.* □

Remarks

1. Only the highest order part of the PDE determines what type of boundary conditions are needed, so the same set is needed if first and zeroth order terms are added to Equation (2.2.1).

2. On each part of the boundary *precisely one* boundary condition applies. (For second order PDE's)

3. Boundary conditions involving the flux vector (Neumann, Robin) are also called *natural boundary conditions.* (For second order PDE's) This term will be explained in Chapter 5.

4. The boundary conditions needed in parabolic and hyperbolic equations are determined by the spatial part of the equation.

5. If the coefficients of the terms of the highest order are *very small* compared to the coefficients of the lower order terms it is to be expected that the nature of the solution is mostly determined by those lower order terms. Such problems are called *singularly perturbed*. An example is the convection dominated convection-diffusion equation (see Section 3.3).

2.2.2 Initial conditions

Initial conditions only play a role in time-dependent problems, and we can be very short. If the equation is first order in time, u has to be given on all of Ω at $t = t_0$. If the equation is second order in time in addition $\frac{\partial u}{\partial t}$ has to be given on all of Ω at $t = t_0$.

Exercise 2.2.2 *Consider the transversal vibrations of membrane that is fixed to an iron ring. These vibrations are described by the wave equation. What is the type of boundary condition? What initial conditions are needed?* □

2.3 Existence and uniqueness of a solution

Physicists and technicians usually consider the mathematical chore of proving existence and uniqueness of a solution a waste of time. 'I know the process behaves in precisely one way', they will claim and of course they are right in that. What they do not know is if their mathematical model describes their process with any accuracy then existence and uniqueness of a solution is an acid test for that. In ODE's a practical way to go about this is try and find one. In PDE's this is not much of an option, since solutions in closed form are rarely available.

Proving existence and uniqueness is usually a very difficult assignment, but to get some of the flavor we shall look at a relatively simple example: Poisson's Equation (2.1.5). We shall prove that a solution to this equation with Dirichlet boundary conditions on all of Γ is unique.

2.3.1 The Laplacian operator

The Laplacian operator div grad is such a fundamental operator that it has a special symbol in the literature: Δ. So the following notations are equivalent:

$$\nabla \cdot \nabla u \equiv \text{div grad}\, u \equiv \Delta\, u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}. \tag{2.3.1}$$

In a technical context div grad is mostly used, in mathematical contexts the other three.
In a physical context it is clear that if there are no sources, a heat equation in equilibrium takes its minimum and maximum at the boundary. Mathematically this is also true as we shall show in the next subsection.

2.3.2 The maximum principle and uniqueness

Solutions to Laplace's and Poisson's equation satisfy certain properties with respect to existence, uniqueness and the occurrence of extremal values at the boundaries of a bounded domain or in the interior of such a domain. We note that a continuous function $u(\mathbf{x})$ has an isolated maximum in some point $\mathbf{x}_0 \in \Omega$ if there exists a $\delta > 0$ such that $u(\mathbf{x}_0) > u(\mathbf{x})$ for $||\mathbf{x} - \mathbf{x}_0|| < \delta$.

Definition 2.3.1 *The Hessian matrix in* $\mathbb{R}^2$ *is defined by*

$$H(\mathbf{x_0}) = \begin{pmatrix} \frac{\partial^2 u}{\partial x^2}(\mathbf{x_0}) & \frac{\partial^2 u}{\partial x \partial y}(\mathbf{x_0}) \\ \frac{\partial^2 u}{\partial y \partial x}(\mathbf{x_0}) & \frac{\partial^2 u}{\partial y^2}(\mathbf{x_0}) \end{pmatrix}. \tag{2.3.2}$$

Theorem 2.3.1 *If the function* $u(\mathbf{x})$ *in* $C^2(\Omega)$*, i.e. the second order derivatives are continuous, the Hessian matrix in an isolated maximum must be negative definite.*

Proof Consider the 2-D Taylor expansion of u around $\mathbf{x}_0$:

$$u(\mathbf{x}) = u(\mathbf{x}_0) + \nabla u(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0, H(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)) + O(||\mathbf{x} - \mathbf{x}_0||^3). \tag{2.3.3}$$

Since u has a maximum for $\mathbf{x}_0$, the smoothness of u implies $\nabla u(x_0) = \mathbf{0}$. When $\mathbf{x}$ approaches $\mathbf{x}_0$ the third order error term becomes arbitrarily small. This implies, with $u(\mathbf{x}) - u(\mathbf{x}_0) < 0$, that there exists a $\delta > 0$ such that
$(\mathbf{x} - \mathbf{x}_0, H(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)) < 0$ for $||\mathbf{x} - \mathbf{x}_0|| < \delta$ and hence $H(\mathbf{x}_0)$ is negative definite. □

Exercise 2.3.1 *Prove that* $H(\mathbf{x}_0)$ *is positive definite if* u *has a minimum in* $\mathbf{x}_0$. □

Exercise 2.3.2 *Show that if* H *is positive definite both diagonal elements must be positive. Hint: Make special choices for* $\mathbf{u}$ *in* $(\mathbf{u}, H\mathbf{u})$. □

Next we are going to consider solutions to Laplace's equation, $-\Delta u = 0$.

Definition 2.3.2 *A function satisfying Laplace's equation* $-\Delta u = 0$ *in* Ω *is called* harmonic *in* Ω.

From Exercise 2.3.2 it is clear that $u_{xx}(\mathbf{x_0})$ and $u_{yy}(\mathbf{x_0})$ are negative if $u(\mathbf{x})$ has an isolated maximum in $\mathbf{x}_0$. This suggests the following theorem

Theorem 2.3.2 *(Strong maximum principle) Let* Ω *be an open bounded domain with boundary* Γ *and closure* $\overline{\Omega}$*, that is* $\overline{\Omega} = \Omega \cup \Gamma$*. Suppose* $u \in C^2(\Omega) \cap C(\overline{\Omega})$ *is harmonic within* Ω*.*
Then

(i) u takes its maximum at the boundary Γ*, hence* $\max_{\mathbf{x} \in \overline{\Omega}} u = \max_{\mathbf{x} \in \Gamma} u$*.*

(ii) Furthermore, if Ω *is connected and if there is an internal point* $\mathbf{x}_0$*, where u reaches its maximum* $(u(\mathbf{x}_0) = \max_{\mathbf{x} \in \overline{\Omega}} u)$*, then u is constant on* $\overline{\Omega}$*,*
that is $u(\mathbf{x}) = u(\mathbf{x}_0)$ *on* $\overline{\Omega}$*.*

This theorem is formulated and proved in Evans [16] among others. To prove the maximum principle, we shall use the arguments given in Protter and Weinberger [30]. Theorem 2.3.2 says that the maximum of a harmonic function is always found on the boundary Γ unless the function is constant. By replacing u by $-u$, we recover similar assertions as in Theorem 2.3.2 with *min* replacing *max*. Before we prove the theorem we give several consequences of the assertion.

Theorem 2.3.3 *Laplace's equation in Ω with a homogeneous Dirichlet boundary condition, that is $u = 0$ on Γ, has only the trivial solution, that is $u = 0$ in Ω.*

Exercise 2.3.3 *Prove Theorem 2.3.3.* □

Theorem 2.3.4 *(uniqueness) Let Ω be a bounded region in $\mathbb{R}^2$ with boundary Γ, and suppose that $u \in C^2(\Omega) \cap C(\overline{\Omega})$ satisfies*

$$-\Delta u = f(x,y), \qquad (x,y) \in \Omega, \tag{2.3.4}$$

$$u = g(x,y), \qquad (x,y) \in \Gamma. \tag{2.3.5}$$

Then there exists at most one solution u.

Exercise 2.3.4 *Prove Theorem 2.3.4.*
Hint: assume that there are two solutions u_1 and u_2 and consider the difference. □

Next we prove Theorem 2.3.2.

Proof of Theorem 2.3.2.
We prove the theorem for $\Omega \in \mathbb{R}^2$. Any dimensionality is dealt with analogously. Let u_m be the maximum on Γ, that is $u \leq u_m$ on Γ. We introduce the function

$$v(x,y) = u(x,y) + \epsilon(x^2 + y^2), \text{ with } \epsilon > 0 \text{ arbitrarily.} \tag{2.3.6}$$

Since u is harmonic, this implies

$$\Delta v = 4\epsilon > 0, \quad \text{in } \Omega. \tag{2.3.7}$$

Suppose that v has a maximum in the open domain Ω, then $\Delta v \leq 0$. This contradicts with the strict inequality (2.3.7), and hence v cannot have a maximum in Ω. Since Ω is a bounded domain in $\mathbb{R}^2$, there exists a radius R such that

$$R = \max_{\mathbf{x} \in \mathbf{\Gamma}} ||\mathbf{x}|| = \max_{\mathbf{x} \in \mathbf{\Gamma}} \sqrt{\mathbf{x^2} + \mathbf{y^2}}. \tag{2.3.8}$$

This implies $v(x,y) \leq u_m + \epsilon R^2$ on Γ. Since v does not have a maximum within the interior Ω, we deduce

$$u(\mathbf{x}) \leq v(\mathbf{x}) \leq u_m + \epsilon R^2 \text{ in } \overline{\Omega} (= \Omega \cup \Gamma). \tag{2.3.9}$$

Since $\epsilon > 0$ can be taken arbitrarily small, we get $u \leq u_m$ in $\overline{\Omega}$. Since u_m is attained on Γ, it follows that a maximum can only be assumed on boundary Γ unless u is constant on $\overline{\Omega}$. □
Uniqueness for the solution to the Poisson equation with Robin conditions can also be proved easily.

Theorem 2.3.5 *Let Ω be a bounded domain in $\mathbb{R}^2$ with boundary Γ, and let $u \in C^2(\Omega) \cap C^1(\overline{\Omega})$ satisfy*

$$-\Delta u = f(x,y)), \qquad (x,y) \in \Omega, \tag{2.3.10}$$

$$\sigma u + \frac{\partial u}{\partial n} = g(x,y), \qquad (x,y) \in \Gamma \tag{2.3.11}$$

with $\sigma > 0$. Then there exists at most one solution u.

Exercise 2.3.5 *Prove Theorem 2.3.5.*
Hints: Assume that there are two solutions u_1 and u_2 and consider the difference $v = u_1 - u_2$. Use multiplication by v and integration by parts to conclude that $v = 0$ on $\overline{\Omega}$. □

Theorem 2.3.6 *Let $u \in C^2(\Omega) \cap C(\overline{\Omega})$ satisfy*

$$-\Delta u \geq 0, \qquad \text{in } \Omega, \tag{2.3.12}$$

$$u = 0, \qquad \text{on } \Gamma, \tag{2.3.13}$$

where Ω is a bounded domain with boundary Γ, then $u \geq 0$ in Ω

Exercise 2.3.6 *Prove Theorem 2.3.6.*
Reason by contradiction and use the Completeness Principle which is: if $u \in C(\overline{\Omega})$ where $\overline{\Omega}$ is a closed bounded set, then u must have a global maximum and minimum on $\overline{\Omega}$. □

Exercise 2.3.7 *Show that the elliptic operator $au_{xx} + 2bu_{xy} + cu_{yy}$, a,b,c constant, $ac - b^2 > 0$ satisfies the same maximum principle as the Laplacian operator.*
Use scaling and rotation of the coordinates. □

Qualitative properties of the solutions to Poisson's or Laplace's equation like the maximum principle are an important tool to evaluate the quality of numerical solutions. Indeed we want our numerical solution to inherit these properties.

2.3.3 Existence

To prove *existence* of a solution of Poisson's equation is very hard. In general one needs extra requirements on the smoothness of the boundary. This is far outside the scope of this book, the interested reader may look at [12]. As we shall see in Chapter 7, there is an alternative way to obtain a *generalized* solution to these problems. The existence proof of such a solution is somewhat easier.

2.4 Examples

In this section we give a few examples of PDE's that describe physical and technical problems. For all problems we consider a bounded region $\Omega \subset \mathbb{R}^2$ with boundary Γ.

2.4.1 Flows driven by a potential

Flows driven by a potential we already met in Chapter 1. They all have the form

$$\frac{\partial c(u)}{\partial t} = \operatorname{div} \lambda \operatorname{grad} u + f(t, \mathbf{x}, u). \tag{2.4.1}$$

For uniqueness c must be a monotone function of u and for stability it must be non-decreasing. In ordinary heat transfer, ground water flow and diffusion, c is linear. In phase transition problems and diffusion in porous media it is non linear. If f depends on u, the function f may influence stability of the equation.

2.4.1.1 Boundary conditions

In Section 2.2 three types of linear boundary conditions have been introduced. These conditions may occur in any combination. This is not a limitative enumeration, there are other ways to couple the heat flow at the boundary to the temperature difference one way or another, mostly non linear.

2.4.1.2 Initial condition

To guarantee that Problem 2.4.1 with boundary conditions (2.2.2) to (2.2.4) has a unique solution $u(\mathbf{x}, t)$, it is necessary that u is prescribed at $t = t_0$: $u(\mathbf{x}, t_0) = u_0(\mathbf{x}), \forall \mathbf{x} \in \Omega$.

2.4.1.3 Equilibrium

An equilibrium of Equation (2.4.1) is reached when all temporal dependence has disappeared. But this problem can also be considered in its own right:

$$-\text{div}\, \lambda\, \text{grad}\, u = f(\mathbf{x}, u), \tag{2.4.2}$$

with boundary conditions (2.2.2) to (2.2.4).

2.4.2 Convection-Diffusion

The *convection-diffusion* equation describes the transport of a pollutant with concentration, c, by a transporting medium with given velocity, $\mathbf{u}$. The equation is

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \text{grad}\, c = \text{div}\, \lambda\, \text{grad}\, c + f(t, \mathbf{x}, c). \tag{2.4.3}$$

Comparison of Equation (2.4.3) with (2.4.1) shows that a *convection* term $\mathbf{u} \cdot \text{grad}\, c$ has been added. Boundary and initial conditions are the same as for the potential driven flows.

In cases where the diffusion coefficient, λ, is small compared to the velocity, $\mathbf{u}$, the flow is *dominated* by the convection. The problem then becomes *singularly perturbed* and in these cases the influence of the second order term is mostly felt at the boundary in the form of *boundary layers*. This causes specific difficulties in the numerical treatment.

2.4.3 Navier-Stokes equations

The Navier-Stokes Equations describe the dynamics of material flow. The momentum equations are given by:

$$\rho(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}) = \text{div}\, \mathbf{s}_x + \rho b_x, \tag{2.4.4a}$$

$$\rho(\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}) = \text{div}\, \mathbf{s}_y + \rho b_y. \tag{2.4.4b}$$

We shall not derive the equations (see for instance [3]), but we will say a few things about their interpretation. The equations describe Newton's second law on a small volume V of fluid with density, ρ, and velocity, $\mathbf{u} = \begin{pmatrix} u \\ v \end{pmatrix}$, *moving along with the flow*. Thus, a particle $P \in V$ with coordinates $\mathbf{x}$ at time $t$has at time $t + \Delta t$, with $\Delta t \to 0$, coordinates $\mathbf{x} + \mathbf{u}\Delta t$. Therefore the change in velocity of a *moving* particle is described by

$$\Delta \mathbf{u} = \mathbf{u}(\mathbf{x} + \mathbf{u}\Delta t, t + \Delta t) - \mathbf{u}(\mathbf{x}, t). \tag{2.4.5}$$

We recall Taylors theorem in three variables:

$$f(x + h, y + k, t + \tau) = f(x, y) + h\frac{\partial f}{\partial x} + k\frac{\partial f}{\partial y} + \tau\frac{\partial f}{\partial t} + O(h^2 + k^2 + \tau^2). \tag{2.4.6}$$

Applying this to Equation (2.4.5) we get:

$$\Delta u = u\Delta t\frac{\partial u}{\partial x} + v\Delta t\frac{\partial u}{\partial y} + \Delta t\frac{\partial u}{\partial t}, \tag{2.4.7a}$$

$$\Delta v = u\Delta t\frac{\partial v}{\partial x} + v\Delta t\frac{\partial v}{\partial y} + \Delta t\frac{\partial v}{\partial t}. \tag{2.4.7b}$$

If we divide both sides by Δt and let $\Delta t \to 0$ we find the *material derivative*

$$\frac{Du}{Dt} = u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{\partial u}{\partial t}, \tag{2.4.8a}$$

$$\frac{Dv}{Dt} = u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{\partial v}{\partial t}. \tag{2.4.8b}$$

The right hand side of Equations (2.4.4) consists of the forces exerted on a (small) volume of fluid. The first term describes surface forces like viscous friction and pressure, the second term describes body forces like gravity. The quantity

$$\Sigma = \begin{pmatrix} \mathbf{s}_x^T \\ \mathbf{s}_y^T \end{pmatrix} = \begin{pmatrix} \sigma_{xx} & \tau_{xy} \\ \tau_{yx} & \sigma_{yy} \end{pmatrix} \tag{2.4.9}$$

is called the *stress tensor*.

The form of the stress tensor depends on the fluid. A *Newtonian fluid* has a stress tensor of the form:

$$\sigma_{xx} = -p + 2\mu\frac{\partial u}{\partial x}, \tag{2.4.10a}$$

$$\sigma_{yy} = -p + 2\mu\frac{\partial v}{\partial y}, \tag{2.4.10b}$$

$$\tau_{xy} = \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}), \tag{2.4.10c}$$

in which p is the pressure and μ the dynamic viscosity. The minimum configuration to be of practical importance requires a mass conservation equation in addition to (2.4.4):

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho\mathbf{u}) = 0, \tag{2.4.11}$$

and a functional relation between ρ and p like for instance *Boyle's law.*

An important special case is where ρ is constant and Equation (2.4.11) changes into

$$\operatorname{div}\mathbf{u} = 0, \tag{2.4.12}$$

the *incompressibility condition*. In this case ρ can be scaled out of Equation (2.4.4) and together with (2.4.10) and (2.4.12) we obtain

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{\partial \bar{p}}{\partial x} = \nu\Delta u + b_x, \tag{2.4.13a}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{\partial \bar{p}}{\partial y} = \nu\Delta v + b_y, \tag{2.4.13b}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{2.4.13c}$$

with $\nu = \frac{\mu}{\rho}$ the kinematic viscosity and $\bar{p} = \frac{p}{\rho}$ the kinematic pressure. In this case $\bar{p}$ is determined by the equations.

Exercise 2.4.1 *Derive Equation* (2.4.13). □

2.4.3.1 Boundary conditions

On each boundary *two* boundary conditions are needed, one in the normal direction and one in the tangential direction. This can be either the velocity or the stress. The tangential stress is computed by $(\mathbf{t}, \Sigma \cdot \mathbf{n})$ for given unit tangent vector, $\mathbf{t}$, and unit normal vector, $\mathbf{n}$. For reasons that go beyond the scope of this book, no boundary conditions for the pressure are required. For an extensive treatment of the Navier-Stokes equations see [39] and [15].

2.4.4 Plane stress

Consider the flat plate in Figure 2.2.

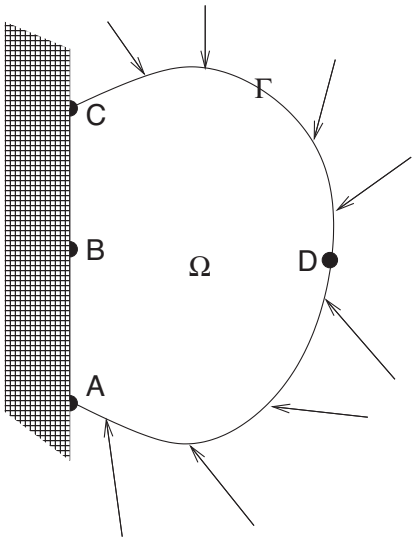


Figure 2.2: Fixed plate with forces applied along the boundary.

The plate is fixed along side ABC but forces are applied along the free boundary ADB as a consequence of which the plate deforms in the x-y-plane. We are interested in the stresses $\Sigma = \begin{pmatrix} \sigma_{xx} & \tau_{xy} \\ \tau_{xy} & \sigma_{yy} \end{pmatrix}$ and the *displacements* $\mathbf{u} = \begin{pmatrix} u \\ v \end{pmatrix}$. The differential equations for the stresses (compare also (2.4.4)) are given by

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + b_1 = 0, \tag{2.4.14a}$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + b_2 = 0, \tag{2.4.14b}$$

in which $\mathbf{b}$ is the (given) body force per unit volume. Usually only gravity contributes to the body force term. We transform Equations (2.4.14) in two stages into a set of PDE's in the displacements. If the medium is *isotropic* we have a a very simple form of *Hooke's Law* relating stresses and strains:

$$E\varepsilon_x = \sigma_{xx} - \nu\sigma_{yy}, \tag{2.4.15a}$$

$$E\varepsilon_y = -\nu\sigma_{xx} + \sigma_{yy}, \tag{2.4.15b}$$

$$E\gamma_{xy} = 2(1+\nu)\tau_{xy}. \tag{2.4.15c}$$

E, the *modulus of elasticity* and ν, Poisson's constant, are material constants. Furthermore, for infinitesimal strains, there is a relation between strain and displacement:

$$\varepsilon_x = \frac{\partial u}{\partial x}, \tag{2.4.16a}$$

$$\varepsilon_y = \frac{\partial v}{\partial y}, \tag{2.4.16b}$$

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}. \tag{2.4.16c}$$

This leads to the following set of PDE's in the displacements $\mathbf{u}$:

$$\frac{E}{1-\nu^2}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x} + \nu\frac{\partial v}{\partial y}\right) + \frac{E}{2(1+\nu)}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) = -b_1, \tag{2.4.17a}$$

$$\frac{E}{2(1+\nu)}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{E}{1-\nu^2}\frac{\partial}{\partial y}\left(\nu\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = -b_2. \tag{2.4.17b}$$

Exercise 2.4.2 *Derive Equations* (2.4.17) □

2.4.4.1 Boundary conditions

The boundary conditions are comparable to those of the Navier-Stokes equations. At each boundary point we need a normal and a tangential piece of data, either the displacement or the stress.

Exercise 2.4.3 *Formulate the boundary conditions along ABC.* □

Exercise 2.4.4 *Along ADC the force per unit length is given:* $\mathbf{f}$. *Show that*

$$\sigma_{xx}n_x + \tau_{xy}n_y = f_1, \tag{2.4.18a}$$

$$\tau_{xy}n_x + \sigma_{yy}n_y = f_2, \tag{2.4.18b}$$

and hence:

$$\frac{n_x E}{1-\nu^2}\left(\frac{\partial u}{\partial x} + \nu\frac{\partial v}{\partial y}\right) + \frac{n_y E}{2(1+\nu)}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) = f_1, \tag{2.4.19a}$$

$$\frac{n_x E}{2(1+\nu)}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{n_y E}{1-\nu^2}\left(\nu\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = f_2. \tag{2.4.19b}$$

□

2.4.5 Biharmonic equation

The prototype of a fourth order PDE is the biharmonic equation on a bounded region $\Omega \subset \mathbb{R}^2$ with boundary Γ:

$$\Delta\Delta w = f. \tag{2.4.20}$$

It describes the vertical displacement w of a flat plate in the x-y-plane, loaded perpendicularly to that plane with force f. To this problem belong three sets of physical boundary conditions:

1. *Clamped boundary*

$$w = 0, \quad \frac{\partial w}{\partial n} = 0, \quad \mathbf{x} \in \Gamma. \tag{2.4.21}$$

2. *Freely supported boundary*

$$w = 0, \quad \frac{\partial^2 w}{\partial n^2} + \nu \frac{\partial^2 w}{\partial t^2} = 0, \quad \mathbf{x} \in \Gamma. \tag{2.4.22}$$

3. *Free boundary*

$$\frac{\partial^2 w}{\partial n^2} + \nu \frac{\partial^2 w}{\partial t^2} = 0, \quad \frac{\partial^3 w}{\partial n^3} + (2 - \nu) \frac{\partial^3 w}{\partial t^3} = 0, \quad \mathbf{x} \in \Gamma. \tag{2.4.23}$$

$\frac{\partial}{\partial n}$ and $\frac{\partial}{\partial t}$ stand for the *normal* and *tangential* derivative respectively. Further ν is Poisson's constant, which depends on the material. In the biharmonic equation the natural boundary conditions contain derivatives of second order or higher, all other boundary conditions are essential.

2.5 Summary of Chapter 2

In this chapter we obtained a classification of second order PDE's into *hyperbolic*, *parabolic* and *elliptic* equations. We formulated appropriate initial and boundary conditions to guarantee a unique solution. We obtained a maximum principle for harmonic functions and used this to prove uniqueness for elliptic equations. We looked at a few examples of partial differential equations in various fields of physics and technology.

Chapter 3

Finite difference methods

Objectives

In this chapter we shall look at the form of discretization that has been used since the days of Euler (1707-1783): finite difference methods. To grasp the essence of the method we shall first look at some one dimensional examples. After that we consider two-dimensional problems on a *rectangle* because that is a straightforward generalization of the one dimensional case. We take a look at the discretization of the three classical types of boundary conditions. After that we consider more general domains and the specific problems at the boundary. Finally we shall turn our attention to the solvability of the resulting discrete systems and the convergence towards the exact solution.

3.1 The cable equation

As an introduction we consider the displacement y of a cable under a vertical load. (See Figure 3.1)

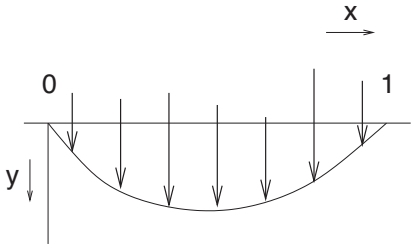


Figure 3.1: Loaded cable.

This problem is described mathematically by the second order ordinary differential equation:

$$-\frac{d^2y}{dx^2} = f, \tag{3.1.1}$$

and since the cable has been fixed at both ends we have a Dirichlet boundary condition at each boundary point:

$$y(0) = 0, \quad y(1) = 0. \tag{3.1.2}$$

Note that here also *one* boundary condition is necessary for the whole boundary, which just consists of two points.

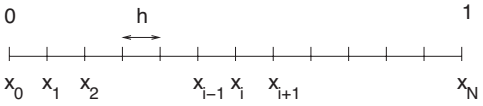


Figure 3.2: Subdivision of the interval $(0,1)$.

3.1.1 Discretization

We divide the interval $(0,1)$ into N subintervals with length $h = 1/N$ (See Figure 3.2). We introduce the notation $x_i = ih$, $y_i = y(x_i)$ and $f_i = f(x_i)$.
In the node point x_i we have:

$$-\frac{d^2y}{dx^2}(x_i) = f_i, \tag{3.1.3}$$

and we shall try to derive an equation that connects the three variables y_{i-1}, y_i, and y_{i+1} with the aid of equation (3.1.3). We recall Taylors formula for sufficiently smooth y:

$$y_{i+1} = y_i + h\frac{dy}{dx}(x_i) + \frac{h^2}{2!}\frac{d^2y}{dx^2}(x_i) + \frac{h^3}{3!}\frac{d^3y}{dx^3}(x_i) + O(h^4), \tag{3.1.4a}$$

$$y_{i-1} = y_i - h\frac{dy}{dx}(x_i) + \frac{h^2}{2!}\frac{d^2y}{dx^2}(x_i) - \frac{h^3}{3!}\frac{d^3y}{dx^3}(x_i) + O(h^4). \tag{3.1.4b}$$

When we sum equations (3.1.4) together the odd order terms drop out, which gives us:

$$y_{i+1} + y_{i-1} = 2y_i + h^2\frac{d^2y}{dx^2}(x_i) + O(h^4). \tag{3.1.5}$$

Rearranging and dividing by h^2 finally gives us the *second divided difference* approximation to the second derivative:

$$\frac{y_{i-1} - 2y_i + y_{i+1}}{h^2} = \frac{d^2y}{dx^2}(x_i) + O(h^2). \tag{3.1.6}$$

The $O(h^2)$ error term is called the *truncation error*, caused by truncating the Taylor series.

Exercise 3.1.1 *Show by the same method that for sufficiently smooth y the* forward divided difference $(y_{i+1} - y_i)/h$ *satisfies*

$$\frac{y_{i+1} - y_i}{h} = \frac{dy}{dx}(x_i) + O(h). \tag{3.1.7}$$

Show that the backward divided difference $(y_i - y_{i-1})/h$ *satisfies*

$$\frac{y_i - y_{i-1}}{h} = \frac{dy}{dx}(x_i) + O(h). \tag{3.1.8}$$

□

Exercise 3.1.2 *Show by the same method that for sufficiently smooth y the* central divided difference $(y_{i+1} - y_{i-1})/2h$ *satisfies*

$$\frac{y_{i+1} - y_{i-1}}{2h} = \frac{dy}{dx}(x_i) + O(h^2). \tag{3.1.9}$$

□

Subsequently, we apply equation (3.1.6) to *every internal node* of the interval, i.e. $x_1, x_2, \ldots, x_{N-1}$, neglecting the $O(h^2)$ error term. Of course by doing so, we only get an approximation (that we denote by u_i) to the exact solution y_i. So we get

$$\begin{aligned} h^{-2}(-u_0 + 2u_1 - u_2) &= f_1 && \text{(3.1.10a)}\\ h^{-2}(-u_1 + 2u_2 - u_3) &= f_2 && \text{(3.1.10b)}\\ \ddots \quad \ddots \quad \ddots &\quad \vdots \\ h^{-2}(-u_{N-2} + 2u_{N-1} - u_N) &= f_{N-1}. && \text{(3.1.10c)} \end{aligned}$$

Taking into account the boundary values $y(0) = y(1) = 0$ we find that $u_0 = u_N = 0$. These values are substituted into equations (3.1.10a) and (3.1.10c) respectively. Hence the system becomes

$$\begin{aligned} h^{-2}(2u_1 - u_2) &= f_1, && \text{(3.1.11a)}\\ h^{-2}(-u_1 + 2u_2 - u_3) &= f_2, && \text{(3.1.11b)}\\ \ddots \quad \ddots \quad \ddots &\quad \vdots \\ h^{-2}(-u_{N-2} + 2u_{N-1}) &= f_{N-1}. && \text{(3.1.11c)} \end{aligned}$$

Or in matrix-vector notation:

$$A\mathbf{u} = \mathbf{f}, \tag{3.1.12}$$

with A an $(N-1) \times (N-1)$ matrix:

$$A = h^{-2} \begin{pmatrix} 2 & -1 & 0 & \ldots & \ldots & 0 \\ -1 & 2 & -1 & 0 & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & -1 & 2 & -1 \\ 0 & \ldots & \ldots & 0 & -1 & 2 \end{pmatrix}. \tag{3.1.13}$$

Exercise 3.1.3 *Show that in case of non-homogeneous boundary conditions, $y(0) = a$, and $y(1) = b$, the matrix A is given by (3.1.13) and that the first and last element of the right-hand side $\mathbf{f}$ are given by $f_1 + h^{-2}a$ respectively $f_{N-1} + h^{-2}b$.* □

The solution of this system can be found by *LU-decomposition*. Since the matrix A is symmetric positive definite, also *Cholesky decomposition* (see[24]) can be used. The proof of positive definiteness will be given in the next section.

3.1.2 Properties of the discretization matrix A

From Expression (3.1.13) it is clear that the matrix A is symmetric. It is easy to prove that the $(N-1) \times (N-1)$ matrix A is positive.

Exercise 3.1.4 *Show that matrix A is positive.*
Hint use Theorem (1.5.7). □

There are several methods to prove that the matrix A is positive definite. The first one is by showing that the inner product $\mathbf{x}^T A\mathbf{x}$ can be written as a sum of squares.

Exercise 3.1.5 *Show that*

$$h^2(\mathbf{x}, A\mathbf{x}) = x_1^2 + \sum_{k=1}^{N-2} (x_{k+1} - x_k)^2 + x_{N-1}^2. \tag{3.1.14}$$

Derive from this result that A is positive definite. □

Another method is to estimate the eigenvalues of the matrix. This can be done by the *von Neumann method*. This approach has the advantage that the smallest eigenvalue can be estimated more accurately than with the bounds that follow from Gershgorin's theorem. Later on it will be used to get a global error estimate. The von Neumann method is based on the fact that the solution of Equation (3.1.3) can be written as

$$y(x) = \sum_{\alpha=1}^{\infty} \rho_\alpha e^{-\pi\alpha x i}, \tag{3.1.15}$$

where i is the imaginary unit ($i^2 = -1$).
In the discrete case we expand the k-th component of u in a similar way ($x_k = kh$)

$$u_k = \sum_{\alpha=1}^{N-1} \rho_\alpha e^{-\pi\alpha k h i}. \tag{3.1.16}$$

The eigenvalue problem $A\mathbf{v} = \lambda\mathbf{v}$ results in

$$\frac{1}{h^2}(-u_{k-1} + 2u_k - u_{k+1}) = \lambda u_k. \tag{3.1.17}$$

Substitution of (3.1.16) in (3.1.17) gives

$$\frac{1}{h^2}\sum_{\alpha=1}^{N-1} \rho_\alpha(-e^{-\pi\alpha(k-1)hi} + 2e^{-\pi\alpha khi} - e^{-\pi\alpha(k+1)hi}) = \lambda \sum_{\alpha=1}^{N-1} \rho_\alpha e^{-\pi\alpha khi}. \tag{3.1.18}$$

This must be true for arbitrary ρ_α, hence each factor following ρ_α in the sum should be zero. Subdivision by $e^{-\pi\alpha khi}$ results in

$$\frac{1}{h^2}(2 - e^{\pi\alpha hi} + e^{-\pi\alpha hi}) = \lambda, \tag{3.1.19}$$

and since

$$cos(\theta) = \frac{e^{i\theta} + e^{-i\theta}}{2}, \tag{3.1.20}$$

we get $N-1$ eigenvalues λ_α:

$$\lambda_\alpha = \frac{2(1 - cos(\pi\alpha h))}{h^2}. \tag{3.1.21}$$

Exercise 3.1.6 *Use the Taylor expansion of the cosine to show that the smallest eigenvalue of the symmetric matrix A is approximately π^2.* □

Since the smallest eigenvalue of the symmetric matrix, A, is positive, it follows that A is positive definite.

Remarks

- The eigenvalue problem corresponding to the Laplace Equation, which is given by

$$-\frac{d^2\varphi}{dx^2} = \mu^2\varphi, \qquad \varphi(0) = \varphi(1) = 0, \tag{3.1.22}$$

  is a special case of the set of Sturm-Liouville problems. The eigenvalues of Equation (3.1.22) form an infinite set given by $\mu^2 = k^2\pi^2$ with k any positive integer. Hence the smallest eigenvalue is exactly π^2.

- The von Neumann method is only applicable for simple cases with constant coefficients, like the one treated here.

3.1.3 Global error

We will estimate the order of the error in our approximate solution u. From Equation (3.1.6) we know that each of the equations of the set (3.1.11) contains an error of $O(h^2)$, provided that y is sufficiently smooth. Suppose that the error in the k-th equation, e_k is given by $e_k = h^2 p_k$. We know that p_k remains bounded as $h \to 0$ by the definition of O. Now let $\Delta y_k = y_k - u_k$, where y_k is the exact solution and u_k our numerical approximation. Then

$$A\mathbf{y} = \mathbf{f} + h^2\mathbf{p}, \tag{3.1.23}$$

and

$$A\mathbf{u} = \mathbf{f}. \tag{3.1.24}$$

We subtract (3.1.24) from (3.1.23) to obtain a set of equations for the error

$$A\Delta\mathbf{y} = h^2\mathbf{p}. \tag{3.1.25}$$

We shall show the global error Δy is of order $O(h^2)$. This is formulated in the following theorem:

Theorem 3.1.1 *The discretization of the Laplace Equation (3.1.1) with boundary conditions (3.1.2) by Equation (3.1.6) gives a global error of $O(h^2)$ in L_2-norm.* □

Proof
From Equation (3.1.25) we get

$$||\Delta\mathbf{y}||_2 \leq ||A^{-1}||_2 h^2 ||\mathbf{p}||_2, \tag{3.1.26}$$

and since the L_2-norm of the inverse of a positive definite matrix is equal to the inverse of the smallest eigenvalue, λ_1, we get

$$||\Delta\mathbf{y}||_2 \leq \frac{h^2}{\lambda_1}||\mathbf{p}||_2 \approx \frac{h^2}{\pi^2}||\mathbf{p}||_2. \tag{3.1.27}$$

□

In this special case it is also possible to estimate the error in the maximum norm as will be shown in the following theorem.

Theorem 3.1.2 *Let the discretization give a truncation error of $e_k = h^2 p_k$ in the k-th equation, then*

$$\|\Delta\mathbf{y}\|_\infty \leq \frac{h^2}{8}\|\mathbf{p}\|_\infty.$$

□

The above theorem will be proved in Exercises 3.1.7 and 3.1.10.

Exercise 3.1.7 *Let $\mathbf{d}$ be a vector with components $d_k = 1, k = 1, 2, \ldots, N-1$. Show by substitution that the solution $\mathbf{e}$ of the set of equations $A\mathbf{e} = \mathbf{d}$ has components $e_k = \frac{1}{2}h^2(N-k)k, k = 1, 2, \ldots, N-1$. Show from this result, that $\|\mathbf{e}\|_\infty \leq 1/8$. (Hint: $Nh = 1$, and by definition $\|\mathbf{p}\|_\infty = \max_k |p_k|$.)* □

In Chapter 2, we saw that the smooth solutions of Laplace's equation satisfy a maximum principle. This should also hold for the numerical solution, which is obtained after the discretization. The following theorem represents the discrete version of the maximum principle:

Theorem 3.1.3 *(Discrete Maximum Principle) The vector inequality* $\mathbf{y} \geq \mathbf{x}$ *means that the inequality is valid for* every *component. Let* A *be the discretization matrix as in* (3.1.13)*, then* $A\mathbf{u} \geq 0$ *implies* $\mathbf{u} \geq 0$. □

Exercise 3.1.8 *Prove Theorem 3.1.3. Reason by contradiction and assume that* $\mathbf{u}$ *has a negative minimum for some component* u_k*. Now consider the k-th equation and show that this is impossible.* □

The next important property is the existence and uniqueness of a numerical solution. This is formulated in the following theorem:

Theorem 3.1.4 *(Existence and uniqueness)*

1. *Let* A *be given as in equation* (3.1.13)*, then* $A\mathbf{u} = 0$ *implies* $\mathbf{u} = 0$.
2. *From this, it follows that the set of equations* $A\mathbf{u} = \mathbf{f}$ *has a solution for every* $\mathbf{f}$ *and that this solution is unique.*

□

Exercise 3.1.9 *Prove Theorem 3.1.4. Use the result from Theorem 3.1.3.* □

Exercise 3.1.10 *With the definitions as in Exercise 3.1.7, show that*

$$-h^2\|p\|_\infty \mathbf{e} \leq \Delta\mathbf{y} \leq h^2\|p\|_\infty \mathbf{e}. \tag{3.1.28}$$

Show that therefore

$$\|\Delta\mathbf{y}\|_\infty \leq \frac{h^2}{8}\|\mathbf{p}\|_\infty. \tag{3.1.29}$$

Hint: use Theorem (3.1.3). □

This concludes the proof of Theorem 3.1.2.

3.2 Some simple extensions of the cable equation

The Laplace Equation (3.1.1) is a special case of the diffusion equation

$$-\frac{d}{dx}(\kappa(x)\frac{d\varphi}{dx}) = f, \tag{3.2.1}$$

with boundary conditions

$$\varphi(0) = a, \quad \varphi(1) = b, \tag{3.2.2}$$

and $\kappa(x)$ a positive function of x.

3.2.1 Discretization of the diffusion equation

There are several possibilities to discretize Equation (3.2.1) with an accuracy of $O(h^2)$. The first one is to rewrite Equation (3.2.1) as

$$-\kappa(x)\frac{d^2\varphi}{dx^2} - \frac{d\kappa(x)}{dx}\frac{d\varphi}{dx} = f. \tag{3.2.3}$$

However, if we apply central differences to discretize (3.2.3), the symmetry that is inherent to Equation (3.2.1) is lost.

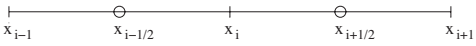


Figure 3.3: Position of discretization points.

One could use Taylor expansion to derive a O(h^2) symmetric discretization of (3.2.1). Unfortunately, such an approach is quite complicated. A better method is to use the central divided differences of Equation (3.1.2) repeatedly. Define

$$y(x) = \kappa(x)\frac{d\varphi}{dx} \tag{3.2.4}$$

and use central differences based on the midpoints $x_{i-\frac{1}{2}}$, $x_{i+\frac{1}{2}}$ (See Figure 3.3) to get

$$\frac{y_{i+\frac{1}{2}} - y_{i-\frac{1}{2}}}{h} = \frac{dy}{dx} + O(h^2). \tag{3.2.5}$$

Substitution of (3.2.4), (3.2.5) into (3.2.1) gives

$$-\frac{\kappa(x_{i+\frac{1}{2}})\frac{d\varphi}{dx}(x_{i+\frac{1}{2}}) - \kappa(x_{i-\frac{1}{2}})\frac{d\varphi}{dx}(x_{i-\frac{1}{2}})}{h} = -\frac{d}{dx}(\kappa(x)\frac{d\varphi}{dx}) + O(h^2). \tag{3.2.6}$$

Next use central differences to discretize $\frac{d\varphi}{dx}$ to get the final expression

$$-\kappa(x_{i+\frac{1}{2}})\frac{\varphi_{i+1} - \varphi_i}{h^2} + \kappa(x_{i-\frac{1}{2}})\frac{\varphi_i - \varphi_{i-1}}{h^2} = f_i. \tag{3.2.7}$$

Exercise 3.2.1 *Use Taylor series expansion to prove that*

$$\kappa(x_{i+\frac{1}{2}}) = \kappa + \frac{h}{2}\kappa' + \frac{h^2}{8}\kappa'' + O(h^3). \tag{3.2.8}$$

Derive a similar expression for $\kappa(x_{i-\frac{1}{2}})$.
Use Taylor series expansion to prove that

$$-\frac{1}{h}[\kappa(x_{i+\frac{1}{2}})\frac{\varphi_{i+1} - \varphi_i}{h} - \kappa(x_{i-\frac{1}{2}})\frac{\varphi_i - \varphi_{i-1}}{h}] = -\frac{d}{dx}[\kappa(x_i)\frac{d\phi}{dx}(x_i)] + O(h^2). \tag{3.2.9}$$

Hint: Use Equation (3.2.3). □

This discretization is clearly symmetric and one can prove that it is also positive definite. Hence the original properties of Equation (3.2.1) are kept.

3.2.2 Boundary conditions

The treatment of Dirichlet boundary conditions is trivial as shown in the previous section. In case the boundary condition contains derivatives, getting an O(h^2) accuracy, requires a thorough discretization.
Consider the Laplace Equation (3.1.1) with boundary conditions

$$y(0) = a, \qquad \frac{dy}{dx}(1) = c. \tag{3.2.10}$$

If we use the subdivision of Figure (3.2) the value of y_N is unknown. Since the discretization (3.1.6) is only applicable to internal points (why?), we need an extra equation to get a square matrix. The most simple method is to use a backward difference to discretize the Neumann boundary condition. This introduces an extra equation, but the truncation error is only O(h) according to Exercise (3.1.1).

A better method is to introduce an extra *virtual point*, x_{N+1}, outside the domain. This implies that the discretization (3.1.6) can be extended to node x_N. The Neumann boundary condition in $x = 1$ can be discretized by central differences. So $y(x_{N+1})$ can be expressed into $y(x_N)$ and $y(x_{N-1})$, and this can be substituted in the discretization of the differential equation in $x = 1$. In fact the virtual point is eliminated in this way. The error in each of the steps is $O(h^2)$, but unfortunately the symmetry of the matrix is lost. Another option is to let the boundary $x = 1$ be in the middle of the interval (x_{N-1}, x_N) as in Figure (3.4). If we omit the truncation

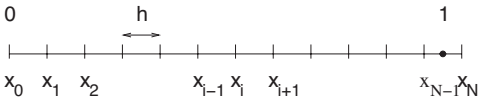


Figure 3.4: Subdivision with virtual point.

error, Equation (3.1.6) in $i = N$ becomes

$$-\frac{y_{N-1} - 2y_N + y_{N+1}}{h^2} = f_N. \tag{3.2.11}$$

Central difference discretization of $\frac{dy}{dx} = c$ gives

$$\frac{y_{N+1} - y_N}{h} = c. \tag{3.2.12}$$

and substitution of (3.2.12) in (3.2.11) results in

$$\frac{-y_{N-1} + y_N}{h^2} = f_N + \frac{c}{h}. \tag{3.2.13}$$

Remark
A more simple way to get a symmetrical matrix would be to use the original matrix and to subdivide the last row of matrix and right-hand side by 2. However, such an approach is only applicable for constant coefficients.

Although in each step of the derivation $O(h^2)$ approximations are used, still the local truncation error of Equation (3.2.13) is $O(h)$, see Exercise (3.2.2)

Exercise 3.2.2 *Show that the Taylor series expansion around x_N of the left-hand side of Equation (3.2.13) can be written as*

$$\frac{y'}{h} - \frac{y''}{2} + \frac{h}{6}y''' + O(h^2), \tag{3.2.14}$$

where $y = y(x_N) = y(1 - \frac{h}{2})$.
Show, using a Taylor series around x_N, that the first derivative of $y(x)$ in point $x = 1$ can be written as

$$y'(1) = y' + \frac{h}{2}y'' + \frac{h^2}{8}y''' + O(h^3). \tag{3.2.15}$$

Show by substitution of (3.2.15) in (3.2.14) and the boundary condition (3.2.10) that the local truncation error of (3.2.13) is O(h). □

It is rather disappointing that the local truncation error is $O(h)$, despite the fact that we used $O(h^2)$ approximations in each step. Fortunately it is possible to prove that the global error is still $O(h^2)$. For that purpose we write the truncation error for

the complete system as $h^2\mathbf{p} + h\mathbf{q}$, where $\mathbf{p}$ is defined as in (3.1.23) and $\mathbf{q}$ is a vector that is completely zero except for the last component which is equal to q_N, so

$$\mathbf{q} = (0, 0, ..., 0, q_N)^T. \tag{3.2.16}$$

The global error Δy can be split into $\Delta y = \Delta y_1 + \Delta y_2$, with

$$A\Delta y_1 = h^2\mathbf{p}, \tag{3.2.17}$$

and

$$A\Delta y_2 = h\mathbf{q}. \tag{3.2.18}$$

From Theorems (3.1.1) and (3.1.2) it follows that $||\Delta y_1|| = O(h^2)$. The exact solution of (3.2.18) is $(\Delta y_2)_i = h^2 x_i$, hence the global error $||\Delta y||$ is also O(h^2).

Exercise 3.2.3 *Show that $\varphi(x) = hq_n x$ is the solution of*
$-\frac{d^2\varphi}{dx^2} = 0, \quad \varphi(0) = 0, \frac{d\varphi}{dx}(1) = hq_N$.
Deduce from this result that $(\Delta y_2)_i = h^2 q_n x_i$, and hence $||\Delta y_2|| = |q_n|h^2$. □

Periodical boundary conditions require a slightly different approach. Such boundary conditions are for example used in case the solution repeats itself endlessly. Consider for example the Poisson equation

$$-\frac{d^2u}{dx^2} \;=\; f(x) \qquad x \in [0, 1], \tag{3.2.19}$$

where $u(x)$ and $f(x)$ are periodical functions with period 1.
Periodicity implies

$$u(x) = u(x + L) \tag{3.2.20}$$

with L the length of the interval. Therefore the trivial boundary condition is

$$u(0) \;=\; u(1). \tag{3.2.21}$$

However, since a second order elliptic equation requires a boundary condition for the whole boundary, two boundary conditions are needed. The second boundary condition one can use is

$$\frac{du}{dx}(0) \;=\; \frac{du}{dx}(1). \tag{3.2.22}$$

Exercise 3.2.4 *Derive* (3.2.22). *Hint use* (3.2.20). □

To discretize Equation (3.2.19) we use the grid of Figure (3.2). The discretization of the differential equation is standard. The discretization of the boundary condition (3.2.21) is trivial. It is sufficient to identify the unknowns u_0 and u_N and represent them by one unknown only (say u_N). To discretize boundary condition (3.2.22) one could use divided differences for both terms in the equation. A more natural way of dealing with this boundary condition is to use the periodicity explicitly by discretizing the differential equation (3.2.19) in $x = 1$ and using the fact that the next point is actually x_1. So we use condition (3.2.22). Hence

$$\frac{-u_{N-1} + 2u_N - u_{N+1}}{h^2} \;=\; f_N. \tag{3.2.23}$$

Exercise 3.2.5 *Why is it sufficient to apply (3.2.23) only for $x = 1$?* □

Exercise 3.2.6 *Show that the discretization of (3.2.19) using (3.2.23) gives the following system of equations*

$$h^{-2}\begin{pmatrix} 2 & -1 & 0 & \dots & 0 & -1 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & -1 & 2 & -1 \\ -1 & 0 & \dots & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ \vdots \\ f_{N-1} \\ f_N \end{pmatrix} \tag{3.2.24}$$

□

Note that for a fast solution of systems of equations of the shape (3.2.24) an adapted solution method is required. See Chapter (9) for the details.

3.3 Singularly perturbed problems

Singularly perturbed problems occur when the coefficient of the highest order derivative is very small compared to the other coefficients. A common example is the *convection diffusion* equation:

$$-\varepsilon\frac{d^2c}{dx^2} + v\frac{dc}{dx} = 0, \qquad c(0) = 0, c(1) = 1, \tag{3.3.1}$$

that describes the transport of a pollutant with concentration c by a convecting medium with known velocity v.

3.3.1 Analytical solution

For constant velocity v and diffusion coefficient ε there is a solution in closed form:

$$c(x) = \frac{e^{\frac{vx}{\varepsilon}} - 1}{e^{\frac{v}{\varepsilon}} - 1}. \tag{3.3.2}$$

For $v/\varepsilon = 40$ the solution has been plotted in Figure 3.5.

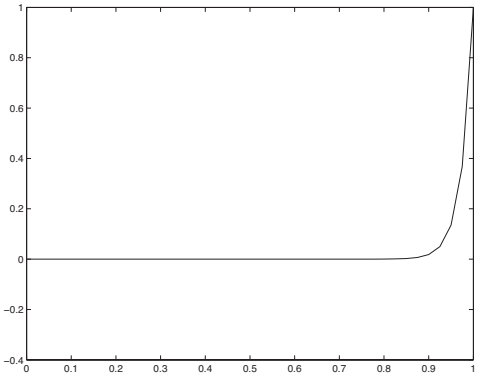


Figure 3.5: Analytic solution.

The quantity vL/ε that occurs regularly in convection diffusion problems is called the *Péclet number Pe*. The quantity L represents a characteristic length. It is a measure for by how much the convection dominates the diffusion. Note that there is a boundary layer at $x = 1$: the right-hand side boundary condition makes itself felt only very close to the boundary. This boundary layer will cause problems in the numerical treatment.

3.3.2 Numerical approximation

Let us take central differences for the first derivative to provide us with an $O(h^2)$ consistent scheme. This gives us a set of equations

$$A\mathbf{c} = \mathbf{f}, \tag{3.3.3}$$

in which A is given by

$$A = h^{-2}\begin{pmatrix} 2 & -1+p_h & 0 & \dots & \dots & 0 \\ -1-p_h & 2 & -1+p_h & 0 & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & 0 & -1-p_h & 2 & -1+p_h \\ 0 & \dots & \dots & 0 & -1-p_h & 2 \end{pmatrix} \tag{3.3.4}$$

and $\mathbf{f}$ by

$$\mathbf{f} = \frac{1}{h^2}\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1-p_h \end{pmatrix}, \tag{3.3.5}$$

in which $p_h = \frac{vh}{2\varepsilon}$ is called the *mesh Péclet number*.

Exercise 3.3.1 *Derive matrix* (3.3.4) *and vector* (3.3.5) □

In Figures 3.6 and 3.7 you see the numerical solution for $Pe = 40$ and $h = 0.1$ and $h = 0.025$ respectively.

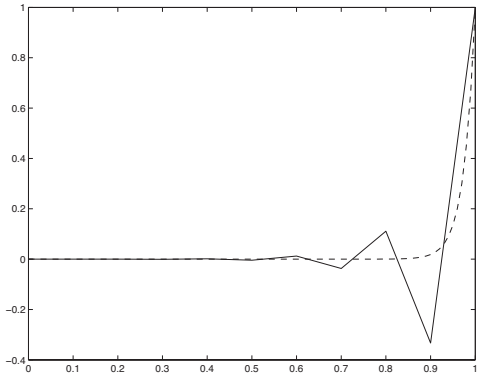


Figure 3.6: Solution (solid) and exact (dotted), coarse grid.

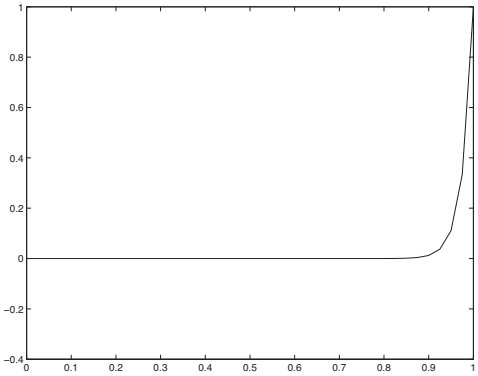


Figure 3.7: Solution, fine grid.

In Figure 3.6 we observe wiggles and negative concentrations. These oscillations are unacceptable from a physical point of view. The wiggles have disappeared in Figure 3.7.

3.3.2.1 Explanation

To explain this phenomenon we consider the following set of *linear difference equations*

$$bu_{k-1} - (b+a)u_k + au_{k+1} = 0, \quad u_0 = 0, u_n = 1. \tag{3.3.6}$$

This system can be solved by substituting $u = r^k$. From Equation (3.3.6) it follows, that

$$b - (b+a)r + ar^2 = 0, \tag{3.3.7}$$

with solutions $r = 1$ and $r = b/a$. The general solution of (3.3.6) can now be written as

$$u_k = A + B\left(\frac{b}{a}\right)^k. \tag{3.3.8}$$

After application of the boundary conditions we find

$$u_k = \frac{\left(\frac{b}{a}\right)^k - 1}{\left(\frac{b}{a}\right)^n - 1}. \tag{3.3.9}$$

Apparently it is necessary that $\frac{b}{a} \geq 0$ to have a monotone, increasing solution.

3.3.2.2 Upwind differencing

For the mesh Péclet number p_h we need the condition $|p_h| \leq 1$ to have a monotone solution. This follows directly from the result of the previous section. To satisfy this inequality we need a condition on the stepsize h: apparently we must have $h \leq \frac{2}{Pe}$. This condition may lead to unrealistically small stepsizes, because in practice Pe can be as large as 10^6. To overcome this you often see the use of *backward* differences for $v > 0$ and *forward* differences for $v < 0$. This is called *upwind differencing*.

Exercise 3.3.2 *Show that taking a backward difference leads to a three term recurrence relation of the form:*

$$(-1-2p_h)u_{k-1}+(2+2p_h)u_k-u_{k+1}=0. \tag{3.3.10}$$

Show that this recurrence relation has a monotone solution if $p_h > 0$. □

Exercise 3.3.3 *Give the three term recurrence relation for* $v < 0$. *Show that this also has a monotone solution.* □

Upwind differencing has a big disadvantage: the accuracy of the solution drops an order and in fact you're having the worst of two worlds: your approximation is bad and you will not be warned that this is the case. See Figure 3.8.

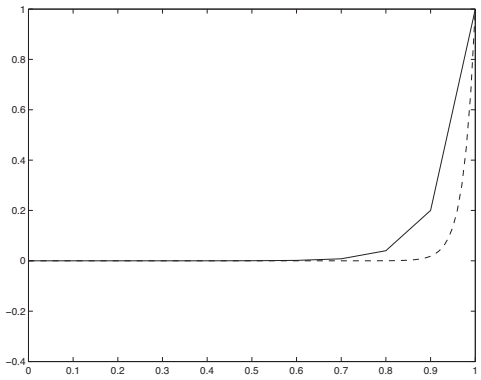


Figure 3.8: Upwind (solid) and exact (dotted) solution.

Why is this approximation so bad? The first order approximation of the first order derivative introduces an artificial diffusion term to suppress the wiggles. This artificial diffusion is an order of magnitude larger than the physical diffusion. So in fact you solve a different problem. See Exercise 3.3.4.

Exercise 3.3.4 *Show that*

$$\frac{c_k-c_{k-1}}{h}=c_k'-\frac{h}{2}c_k''+O(h^2). \tag{3.3.11}$$

Show that this approximation reduces the Péclet number to

$$\widehat{Pe}=\frac{Pe}{1+p_h}. \tag{3.3.12}$$

Deduce from this that $\hat{p}_h < 1$ *for* $v > 0$. *Give analogous relations for* $v < 0$ *and explain why it is necessary to take a forward difference in this case.* □

Effectively, using upwind differencing, you are approximating the solution of

$$-(\varepsilon+\frac{vh}{2})\frac{d^2c}{dx^2}+v\frac{dc}{dx}=0. \tag{3.3.13}$$

It is clear that for a good accuracy $\frac{vh}{2}$ must be small compared to ε. Hence upwind differencing produces nice pictures, but if you need an accurate solution, then, central differences with small h are preferred.

A better way to handle the boundary layer is *mesh refinement* in the boundary layer

itself. The boundary layer contains large gradients and to resolve these you need a sufficient number of points. Actual practice shows that taking sufficient points in the boundary layer suppresses the wiggles. In Figure 3.9 the solution is calculated with 10 points only, but at nodes 0.5, 0.8, 0.85, 0.88, 0.91, 0.93, 0.95, 0.97, 0.99 and 1.

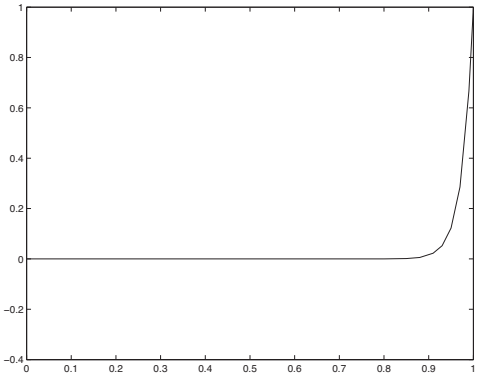


Figure 3.9: Non equidistant node points.

In favor of the upwind differencing method it has to be said that it is the only course of action available in the neighborhood of shocks. As a result you often see methods with a higher accuracy in smooth regions of the solution that fall back on the first order upwind scheme close to shocks.

3.3.2.3 Source terms

If *source terms* in the equation suppress the boundary layer there will be no wiggles in the numerical solution, even if the matrix does not satisfy the *mesh Péclet condition* $p_h \leq 1$.

Exercise 3.3.5 *Calculate with central differences the numerical solution of*

$$-y'' + vy' = \pi^2 \sin \pi x + v\pi \cos \pi x, \quad y(0) = y(1) = 0. \tag{3.3.14}$$

Take $v = 40$ *and* $h = 0.1$. □

Remark
The use of the previous upwind differencing, also called *first order upwind*, may be inaccurate, it usually produces nice pictures. This makes the method attractive from a selling point of view. In the literature more accurate higher upwind schemes can be found. Treatment of these schemes goes beyond the scope of this textbook.

3.4 The Laplacian equation on a rectangle

We now generalize our procedure to two dimensions. Consider a rectangle Ω with length L and width W. In this rectangle we consider *Poisson's equation*:

$$-\Delta u = f, \tag{3.4.1}$$

with *homogeneous boundary conditions* $u = 0$ on Γ.
We divide Ω into small rectangles with sides Δx and Δy such that $M\Delta x = L$ and $N\Delta y = W$. At the intersections of the grid lines we have *nodes* or *nodal points*

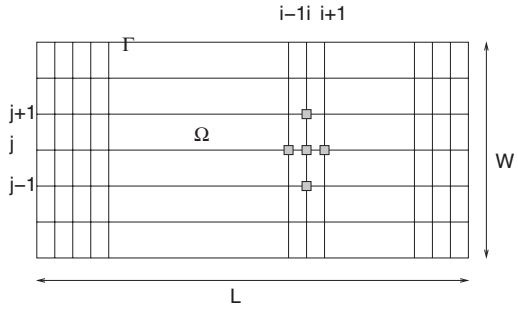


Figure 3.10: Rectangular grid with 5 point molecule.

where we shall try to find approximations of the unknown u. The unknown at node (x_i, y_j) (or (i, j) for short) we denote by $u_{i,j}$. In the same way as in Section 3.1 we replace the differential equation in this node by

$$\frac{-u_{i-1,j} + 2u_{i,j} - u_{i+1,j}}{\Delta x^2} + \frac{-u_{i,j-1} + 2u_{i,j} - u_{i,j+1}}{\Delta y^2} = f_{i,j}. \tag{3.4.2}$$

Exercise 3.4.1 *Use Taylor expansion in two variables to show that the truncation error in* (3.4.2) *is given by*

$$E_{ij} = \frac{1}{12}\left(\Delta x^2 \frac{\partial^4 u}{\partial x^4}(x_i, y_j) + \Delta y^2 \frac{\partial^4 u}{\partial y^4}(x_i, y_j)\right). \tag{3.4.3}$$

In this expression terms of order 5 and higher in the Taylor expansion have been neglected. □

Writing down equation (3.4.2) for every internal node point (i, j), $i = 1, 2, \ldots, M-1$, $j = 1, 2, \ldots, N-1$ presents us with a set of $(M-1) \times (N-1)$ equations with just as many unknowns.

Exercise 3.4.2 *Give the equation with node (1,5) as central node. Substitute the homogeneous boundary conditions.* □

Exercise 3.4.3 *Give the equation with node* $(M-1, N-1)$ *as central node. Substitute the homogeneous boundary conditions.* □

3.4.1 Matrix vector form

Since the system we obtained is a linear system we can represent it in matrix vector form $A\mathbf{u} = \mathbf{f}$. This is not exactly a trivial task, because we have a vector of unknowns with a double index and the conventional matrix vector representation uses a simple index. We shall show how to do this in a specific example, $M = 6, N = 4$. First of all we show how to convert the double index (i, j) into a single index α. This can be done in a number of ways, that are most easily represented in a picture.

3.4.1.1 Horizontal numbering

The nodes are numbered sequentially (see Figure 3.11).

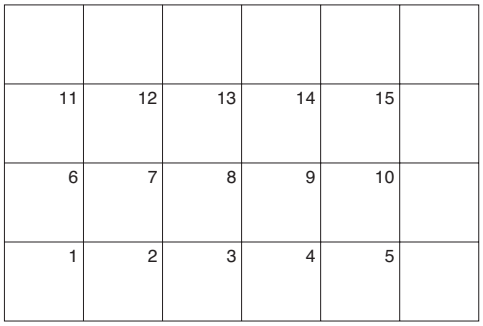


Figure 3.11: Horizontal numbering.

The conversion formula from double index (i, j) to single index α is straightforward:

$$\alpha = i + (j-1) * (M-1). \tag{3.4.4}$$

Exercise 3.4.4 *Show that A is a 3×3 block matrix in which each block is 5×5. What is the band width of A?* □

The diagonal blocks are tridiagonal, the sub and super diagonal blocks are diagonal and all other blocks are 0.

3.4.1.2 Vertical numbering

The nodes are numbered sequentially in vertical direction (see Figure 3.12).

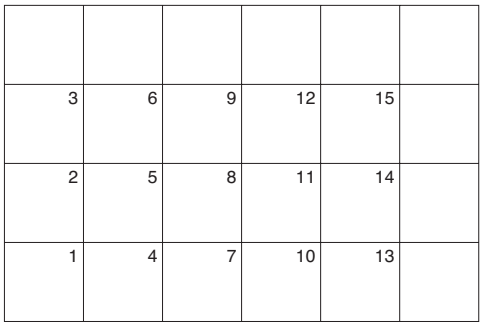


Figure 3.12: Vertical numbering.

The conversion formula from double index (i, j) to single index α is straightforward:

$$\alpha = (i-1) * (N-1) + j. \tag{3.4.5}$$

Exercise 3.4.5 *Show that A is a 5×5 block matrix in which each block is 3×3. What is the band with of A?* □

The diagonal blocks are tridiagonal, the sub and super diagonal blocks are diagonal and all other blocks are 0.

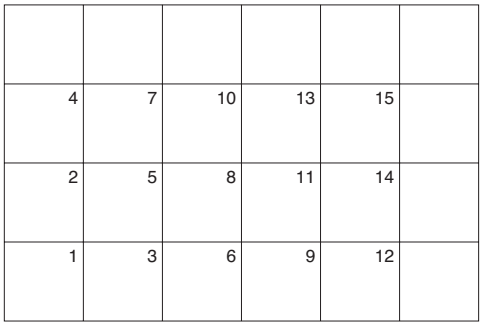


Figure 3.13: Oblique numbering.

3.4.1.3 Oblique numbering

The nodes are numbered sequentially along lines $i + j = k, k = 2, \ldots, 8$ (see Figure 3.13).
The conversion formula from double index (i, j) to single index α is not so straightforward. A is still a block matrix, in which the diagonal blocks increase in size from 1×1 to 3×3. The diagonal blocks are diagonal, the sub and super diagonal blocks are diagonal and all other blocks are 0.

Exercise 3.4.6 *What is the bandwidth of A?* □

3.5 Boundary conditions extended

3.5.1 Natural boundary conditions

Basically *natural boundary conditions* (i.e. Neumann or Robin boundary conditions) involve a flow condition. The treatment in 2D is similar to 1D (see Section 3.2.2). Since these conditions are dealt with in a natural way by Finite Volume Methods we postpone a more detailed discussion of that subject until the next chapter.

3.5.2 Dirichlet boundary conditions on non rectangular regions

Unfortunately on non rectangular regions the boundary does not coincide with the grid, see Figure 3.14.

For each interior point we have an equation involving function values in five nodes. The black points in Figure 3.14 have to be determined by the Dirichlet boundary condition. It is acceptable to express a black point in a nearby boundary value and the function values in one or more interior points (interior variables). The idea is to end up with a system of equations that only contains interior variables. In this way we can guarantee that we have as many equations as unknowns. We explain the way to proceed by an example. Consider the situation in Figure 3.15.

In this figure we have to express u_S in the known value u_B and the interior variable u_C. Let h be the distance between grid points and sh the fraction that separates the boundary from the S-point. By linear interpolation we have

$$u_B = (1 - s)u_S + su_C + O(h^2), \tag{3.5.1}$$

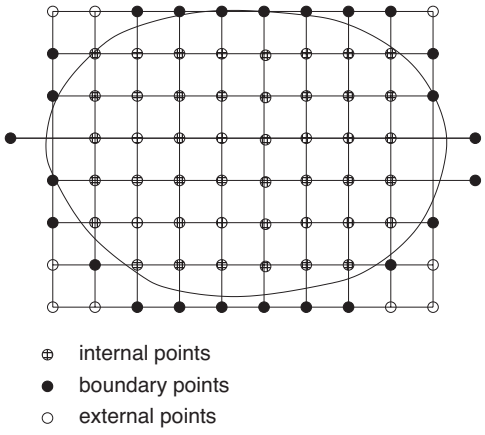


Figure 3.14: Grid on non rectangular region.

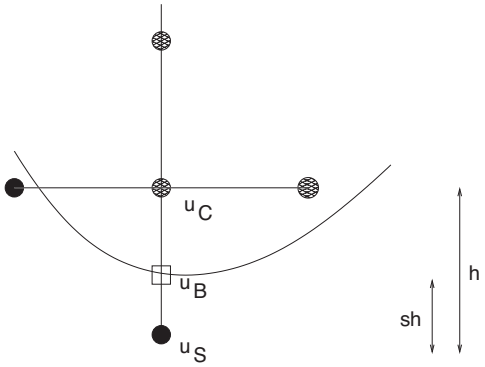


Figure 3.15: Boundary molecule.

and that gives us the relation that we can substitute into the equation:

$$u_S = \frac{u_B - s u_C}{1 - s}. \tag{3.5.2}$$

If s is close to 1 this procedure may lead to an unbalanced set of equations. For that reason we usually consider a point that is closer than say $\frac{1}{4}h$ to the boundary as a *boundary point* even if it belongs to the interior. In that case u_S falls in between u_B and u_C and the formulae change correspondingly.

Here we have

$$u_S = \frac{s u_C + u_B}{1 + s}. \tag{3.5.3}$$

Remark
The method treated here is quite old fashioned. It is better to use either a coordinate transformation (Section 3.7) or alternatively the Finite Element Method (Chapter 6).

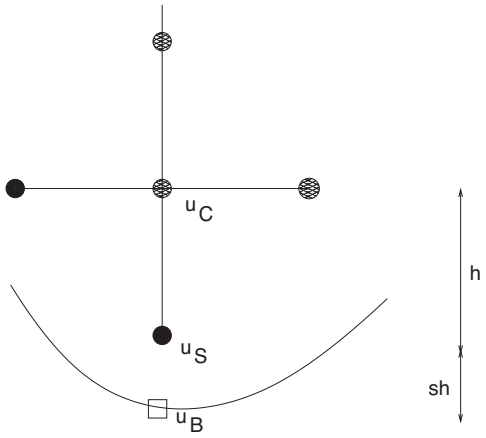


Figure 3.16: Boundary molecule, interior boundary point.

3.6 Global error estimate

We shall try to get some of the flavor of global error estimates for numerical solutions of Problem 3.4.1. The L_2 error estimate can be derived in the same way as in Theorem 3.1.1. Here we shall concentrate ourselves to point wise estimates. In order to do so we need to develop some properties for the discrete Laplace operator. These properties also hold in 3 dimensions, so in a certain way this is a generic treatment of the problem. We will do the estimate on a rectangle with homogeneous Dirichlet boundary conditions, but in subsequent sections we shall hint at ways to apply the theory to more general domains and boundary conditions.

3.6.1 A discrete maximum principle

If the $N \times N$ system of equations $A\mathbf{u} = \mathbf{f}$ is a Finite Difference discretization of Problem 3.4.1 with Dirichlet boundary conditions then A has the following properties:

$$a_{jk} \leq 0, \quad \text{if } j \neq k, \tag{3.6.1a}$$

$$a_{kk} \geq \sum_{\substack{j=1 \\ j \neq k}}^{N} |a_{kj}|, \quad k = 1, \ldots, N. \tag{3.6.1b}$$

We call matrices with property (3.6.1a) a Z-matrix and matrices with property (3.6.1b) *diagonally dominant*. If inequality (3.6.1b) holds strictly for at least one k the matrix is called *strictly* diagonally dominant. (There are complications if the system can be split into independent subsystems, but that won't bother us right now.)

We use the notation $\mathbf{y} \geq 0$ for $y_k \geq 0, k = 1, \ldots, N$. We formulate a very important theorem for the solution of systems of strictly diagonally dominant Z-matrices.

Theorem 3.6.1 *(Discrete Maximum Principle) Let A be a strictly diagonally dominant Z-matrix and let $A\mathbf{u} \geq 0$. Then*

- *(i)* $\mathbf{u} \geq 0$.

- *(ii)* $\mathbf{u} = \mathbf{0}$ *if and only if* $\mathbf{Au} = \mathbf{0}$.

Proof First we prove (i). Suppose $u_k < 0$ for some k then $\mathbf{u}$ has a negative minimum $-M$ for some index K. Hence $\mathbf{u} \geq -M$. Now consider the inequality with number K:

$$\sum_{j=1}^{N} a_{Kj} u_j \geq 0, \tag{3.6.2}$$

hence

$$a_{KK} u_K \geq -\sum_{\substack{j=1 \\ j \neq K}}^{N} a_{Kj} u_j, \tag{3.6.3}$$

and since $u_K = -M < 0$ by hypothesis, we have

$$a_{KK} M \leq \sum_{\substack{j=1 \\ j \neq K}}^{N} a_{Kj} u_j. \tag{3.6.4}$$

Because $a_{Kj} \leq 0, j \neq K$ the right hand side of the inequality is majorized by taking instead of u_j the negative minimum $-M$. We observe that $-a_{Kj}M = |a_{Kj}|M, j \neq K$ which lead us to:

$$a_{KK} M \leq \sum_{\substack{j=1 \\ j \neq K}}^{N} |a_{Kj}| M. \tag{3.6.5}$$

Since $M > 0$ we can divide both sides by M and arrive at a contradiction unless

$$a_{KK} = \sum_{\substack{j=1 \\ j \neq K}}^{N} |a_{Kj}|. \tag{3.6.6}$$

So (3.6.5) holds only for the equal case, and since $-M$ is the minimum (3.6.4) implies $u_j = -M$ for $a_{Kj} < 0$. This means that u_j is constant for all elements j in the molecule. We can repeat this argument for all molecules that have at least one element j in common with a previously considered molecule. Finally we arrive at a molecule where (3.6.1b) holds strictly. For that molecule we have a contradiction unless $M = 0$. This proves part (i) of the theorem.
The if part of (ii) is trivial. The *only if* part is proven in Exercise (3.6.1). □

Exercise 3.6.1 *Prove, under the hypothesis of Theorem 3.6.1 that* $A\mathbf{u} \leq 0$ *implies* $\mathbf{u} \leq 0$ *(Hint: consider* $-\mathbf{u}$*). Use this result to prove that* $A\mathbf{u} = 0$ *implies* $\mathbf{u} = 0$. □

According to Theorem 2.2.1 the solution of the Poisson equation with Neumann boundary conditions is not unique. In that case the row sum of each row of the matrix is equal to 0. In Exercise 3.6.2 it is shown that also the numerical solution is not unique.

Exercise 3.6.2 *Use the proof of Theorem 3.6.1 and Equation* (3.6.5) *to show that if equality holds in Equation* (3.6.1b) *for all k the system* $A\mathbf{u} = 0$ *(A being a Z-matrix) has a nontrivial solution. Determine that solution.* □

Exercise 3.6.3 *Use Theorem 3.6.1 to prove that if A is a strictly diagonally dominant Z-matrix and* $A\mathbf{u} = \mathbf{f}$ *and* $A\mathbf{w} = |\mathbf{f}|$*, with* $\mathbf{f} \neq 0$ *then* $|\mathbf{u}| \leq \mathbf{w}$.
Hint: also consider $A(-\mathbf{u})$. □

3.6.1.1 Discrete harmonics and linear interpolation

We show an important consequence of the discrete maximum principle. This theorem is in fact the discrete equivalent of the strong maximum principle (Theorem (2.3.2).

Theorem 3.6.2 *A discrete solution to Laplace's equation with Dirichlet boundary conditions has its maximum and minimum on the real boundary, provided the boundary conditions have been approximated by linear interpolation.*

Proof
We only sketch the proof, the reader will have no difficulty in filling in the details. The ordinary five point molecule to approximate the Laplace operator generates a strictly diagonally dominant Z-matrix, and application of linear interpolation does not alter that. The inequality (3.6.1b) only holds for those molecules that contain a Dirichlet boundary condition. So the maximum M will, by a now familiar reasoning be attained by an interior point that is one cell away from the boundary, like u_C in Figure 3.16. This equation has been modified into:

$$-u_W - u_N - u_E + (3 + \frac{1}{1+s})u_C = \frac{1}{1+s}u_B. \tag{3.6.7}$$

But since u_N, u_W and u_E have to be not greater than M this means

$$-3M + (3 + \frac{1}{1+s})u_C \leq \frac{1}{1+s}u_B, \tag{3.6.8}$$

or since $u_C = M$ by assumption

$$M \leq u_B. \tag{3.6.9}$$

An analogous reasoning shows that the minimum m is attained at the physical boundary. □

Exercise 3.6.4 *Derive Equation (3.6.7).* □

3.6.2 Super solutions

The (discrete) maximum principle is used to *bound* (discrete) solutions to Poisson's equation. Why would we want to do such a thing? Remember, that we have an error estimate in the form:

$$A\varepsilon = h^2\mathbf{p}, \tag{3.6.10}$$

in which the vector $\mathbf{p}$ is uniformly bounded as $h \to 0$. Suppose we had a solution $\mathbf{q}$ to the equation $A\mathbf{q} = \mathbf{p}$; we would then have an error estimate $\varepsilon = h^2\mathbf{q}$. Usually this is asking too much. But if we are able to *bound* the vector $\mathbf{p}$ by a vector $\mathbf{r} \geq \mathbf{p}$ then the solution $\mathbf{s}$ to $A\mathbf{s} = \mathbf{r}$ bounds $\mathbf{q}$ by the discrete maximum principle: $\mathbf{q} \leq \mathbf{s}$. This gives us an error estimate as well: $\varepsilon \leq h^2\mathbf{s}$. Such a *super solution* $\mathbf{s}$ is obtained by solving the Laplacian for a specific right-hand side that has the properties:

- the solution can be easily obtained
- it dominates the right-hand side of the equation that we are interested in

An obvious choice for the vector $\mathbf{r}$ would be the constant vector $h^2\|\mathbf{p}\|_\infty$. We will show that to get the solution $\mathbf{s}$, it is sufficient to consider the equation $-\Delta u = 1$.

3.6.2.1 A discrete solution to $-\Delta u = 1$

Consider the problem $-\Delta v = 1$ on a circle of radius 1 and the origin as its midpoint with homogeneous Dirichlet boundary conditions. By substitution it is easily verified that $v = \frac{1}{4}(1 - x^2 - y^2)$ is the solution of this problem. But since second divided differences are *exact* for polynomials of degree 2 (why?) the discrete function $v_{ij} = \frac{1}{4}(1 - x_i^2 - y_j^2)$ is a solution to the discretized equation $A\mathbf{u} = \mathbf{e}$ in which $\mathbf{e}$ contains all ones and the single index vector $\mathbf{u}$ is an appropriate remap of the double index vector v_{ij}. That is, if we disregard the approximation to the boundary conditions for the moment.

Exercise 3.6.5 *Show that* $\|\mathbf{u}\|_\infty = \frac{1}{4}$. □

Exercise 3.6.6 *Give the solution of* $-\Delta u = 1$ *with homogeneous Dirichlet boundary conditions on a circle C with midpoint* $(0, 0)$ *and radius R. Show that this is a super solution to the same problem on an arbitrary G region wholly contained in C. Hint: consider the difference of the two solutions and show that they satisfy a Laplace equation with nonnegative boundary conditions. Use Theorem 3.6.2 to conclude that the difference must be nonnegative also.* □

3.6.2.2 Pesky mathematical details: the boundary condition

To develop our train of thoughts unhampered in the previous section we overlooked a pesky mathematical detail. At a boundary point we used linear interpolation and that has influenced our equation somewhat. As a result, the function v_{ij} as introduced in the previous paragraph is not really the solution of $A\mathbf{u} = \mathbf{e}$ but rather of a perturbed system $A\tilde{\mathbf{u}} = \mathbf{e} + \mathbf{e}_b$. The vector e_b contains the interpolation error of $O(h^2)$ at the boundary.

Exercise 3.6.7 *Consider the discretization of* $-\Delta u = 1$ *with homogeneous Dirichlet boundary conditions on the circle with radius 1 in the neighborhood of the boundary as in Figure 3.16. Show that this discretization is given by:*

$$-u_W - u_N - u_E + (3 + \frac{1}{1+s})u_C = h^2. \tag{3.6.11}$$

Verify, that the discrete solution $v_{ij} = \frac{1}{4}(1 - x_i^2 - y_j^2)$ *does not satisfy* this *equation, but rather the equation:*

$$-u_W - u_N - u_E + (3 + \frac{1}{1+s})u_C = h^2 + \frac{s}{4}h^2. \tag{3.6.12}$$

*(*Hint: $1 - x_i^2 - (y_j - (1+s)h)^2 = 0$.*)*
Show that this is equivalent with an error in the boundary condition Δu_B *of* $O(h^2)$. □

Exercise 3.6.8 *Show by using Theorem 3.6.2 and the result of Exercise 3.6.7 that* $\tilde{\mathbf{u}} - \mathbf{u} = O(h^2)$. □

In the sequel we shall neglect the influence of linear interpolation error on the boundary conditions.

3.6.2.3 A point wise error estimate to the discrete solution

Let us apply the results of the previous sections to our error estimate. We have the following theorem:

Theorem 3.6.3 *Let $A\mathbf{u} = \mathbf{f}$ be the discretization of the Poisson equation with homogeneous Dirichlet boundary conditions on a region G wholly contained in a circle with radius R. Let the discretization error be given by $A\varepsilon = h^2\mathbf{p}$ such that $\|\mathbf{p}\|_\infty$ is bounded as $h \to 0$. Then*

$$\|\varepsilon\|_\infty \leq \frac{1}{4}R^2h^2\|\mathbf{p}\|_\infty \tag{3.6.13}$$

Exercise 3.6.9 *Explain why the midpoint of the circle does not play a role in Theorem 3.6.3. Is it true that we can take the smallest circle that wholly contains G?* □

Exercise 3.6.10 *Show that if $A\mathbf{w} = \|\mathbf{p}\|_\infty\mathbf{e}$, then $|\varepsilon| < h^2\mathbf{w}$.* □

Exercise 3.6.11 *Prove Theorem 3.6.3.* □

3.7 Boundary fitted coordinates

In Section 3.5 we payed attention to boundary conditions on general domains. A different approach is the use of *boundary fitted coordinates* that make the boundary of the domain a coordinate line. This usually leads to a reformulation of the problem in *general curvilinear coordinates*. This solves one problem, but introduces another because usually the PDE (even a simple PDE like the Laplacian) can easily become very complex. This approach can also be used if one wants to apply a local grid refinement. We will explain the principle for a one-dimensional problem. Suppose that one has to solve the following problem:

$$-\frac{d}{dx}\left(D(x)\frac{du}{dx}\right) = f(x), \text{ with } u(0) = 0 \text{ and } u(1) = 1. \tag{3.7.1}$$

Here $D(x)$ and $f(x)$ are given functions. For specific choices of $D(x)$ and $f(x)$ a local grid refinement is desirable at positions where the magnitude of the second derivative is large. One can use a co-ordinate transformation such that the grid spacing is uniform in the transformed co-ordinate. Let this co-ordinate be given by ξ, then in general, the relation between x and ξ can be written as

$$x = \Gamma(\xi), \tag{3.7.2}$$

where Γ represents the function for the co-ordinate transformation and we require that Γ is a *bijection* (that is, Γ is *one-to-one*). Then, differentiation with respect to x yields

$$1 = \Gamma'(\xi)\frac{d\xi}{dx}, \tag{3.7.3}$$

so $\frac{d\xi}{dx} = \frac{1}{\Gamma'(\xi)}$ and this implies, after using the Chain Rule for differentiation

$$\frac{du}{dx} = \frac{1}{\Gamma'(\xi)}\frac{du}{d\xi}. \tag{3.7.4}$$

Hence, the differential equation (3.7.1) in x transforms into the following differential equation for ξ

$$\begin{aligned} -\frac{1}{\Gamma'(\xi)}\frac{d}{d\xi}\left[\frac{D(\Gamma(\xi))}{\Gamma'(\xi)}\frac{du}{d\xi}\right] &= f(\Gamma(\xi)). \\ u(\xi_L) = 0, \qquad u(\xi_R) &= 1, \end{aligned} \tag{3.7.5}$$

where $0 = \Gamma(\xi_L)$ and $1 = \Gamma(\xi_R)$. The above differential equation is much more complicated than equation (3.7.1), but it can be solved on an equidistant grid. After

the equation is solved, the solution is mapped onto the gridnodes on the x-number line. In practice, one often does not know the function $\Gamma(\xi)$ in an explicit form, then one has to use a numerical approximation for the derivative of $\Gamma(\xi)$. We will return to this subject in Section 4.3.1.

Exercise 3.7.1 *Consider equation* (3.7.1), *where*

$$f(x) = \begin{cases} 256\,(x-1/4)^2(x-3/4)^2, & \text{for } 1/4 < x < 3/4 \\ 0, & \text{elsewhere.} \end{cases}$$

Suppose that we prefer to discretize such that the mesh is refined at positions where the error is maximal. Then, one has to use a local mesh refinement near $x = 1/2$. Therefore, we use the transformation $x = \Gamma(\xi) = \xi^2(3-2\xi)$. Show, that this transformation yields a mesh refinement at $x = 1/2$, and give the transformed differential equation expressed in ξ, in which one will use an equidistant grid. □

The extension to two dimensions is quite simple. Consider for example Poisson's equation on a circle.

$$-\text{div grad } u = f(x,y), \text{ for } (x,y) \in \Omega. \tag{3.7.6}$$

In order to get a rectangular grid we map the circle onto a rectangle in (r,θ) space, i.e. we transform to polar coordinates. This transformation is defined by

$$x = rcos\theta, \qquad y = rsin\theta. \tag{3.7.7}$$

Exercise 3.7.2 *Express the derivatives of u with respect to x and y in $\frac{\partial u}{\partial r}$ and $\frac{\partial u}{\partial \theta}$.* □

Exercise 3.7.3 *Show that the derivatives, $\frac{\partial r}{\partial x}$, $\frac{\partial r}{\partial y}$, $\frac{\partial \theta}{\partial x}$ and $\frac{\partial \theta}{\partial y}$ are given by*

$$\begin{pmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} \\ \frac{\partial \theta}{\partial x} & \frac{\partial \theta}{\partial y} \end{pmatrix} = \frac{1}{r} \begin{pmatrix} rcos\theta & rsin\theta \\ -sin\theta & cos\theta \end{pmatrix}. \tag{3.7.8}$$

□

Exercise 3.7.4 *Use the results of Exercises (3.7.2) and (3.7.3) to prove that the Poisson equation (3.7.6) in polar coordinates is defined by*

$$-\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2}\right) = f(rcos\theta, rsin\theta). \tag{3.7.9}$$

□

Remark
Note that $r = 0$ is a singular line in Equation (3.7.9).

Exercise 3.7.5 *Which boundary conditions are needed to get rid of the singularity?* □

Exercise 3.7.6 *Discretize of Poisson's equation on a circle of radius 1 in the (r,θ)-plane. Use homogeneous Dirichlet boundary conditions on the circle. Formulate boundary conditions for $r = 0$, $\theta = 0$ and $\theta = 2\pi$.* □

3.8 Summary of Chapter 3

In this chapter we have seen finite difference methods in one and two dimensions. We have looked at the effect of a boundary layer on numerical approximations. We have derived point-wise error estimates for problems with homogeneous Dirichlet boundary conditions using a discrete maximum principle. A method to include Dirichlet boundary conditions on more general regions has been shown and finally we have presented the formula of the Laplacian operator in general coordinates.

Chapter 4

Finite volume methods

Objectives

In the previous chapter we got to know discretization by finite differences. This discretization has two major disadvantages: it is not very clear how to proceed with non equidistant grids; moreover natural boundary conditions are very hard to implement, especially in two or three dimensions. The finite volume discretization that we are about to introduce do not possess these disadvantages. But *they* apply only to differential operators in *divergence* or *conservation* form. For physical problems this is rather a feature than a bug: usually the conservation property of the continuous model will be inherited by the discrete numerical model.

We shall start out with a one dimensional example that we left dangling in our previous chapter: a second order equation on a non equidistant grid. We shall pay attention to Neumann and Robin boundary conditions too. Subsequently we shall turn our attention to two dimensions and discretize the Laplacian in general coordinates. Then we will look at problems with two components: fluid flow and plane stress. We shall introduce the concept of *staggered grids* and show that that is a natural way to treat these problems. There will be a problem at the boundaries in this case that we have to pay attention to.

4.1 Heat transfer with varying coefficient

We consider the diffusion equation on the interval $(0,1)$:

$$-\frac{d}{dx}\left(\lambda\frac{dT}{dx}\right) = f, \quad \lambda\frac{dT}{dx}(0) = 0, -\lambda\frac{dT}{dx}(1) = \alpha(T - T_R). \tag{4.1.1}$$

In this equation λ may depend on the space coordinate x. T_R is a (given) reference temperature and as you see we have natural boundary conditions on both sides of the interval. We divide the interval in (not necessarily equal) subintervals $e_k, k = 1, \ldots, N$, where e_k is bounded by the nodal points (x_{k-1}, x_k). See Figure 4.1.

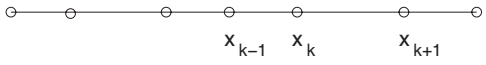


Figure 4.1: Non equidistant grid.

To derive a discrete equation to this problem we consider three subsequent nodes in isolation x_{k-1}, x_k and x_{k+1}, see Figure 4.2. We let $h_k = x_k - x_{k-1}$, $h_{k+1} =$

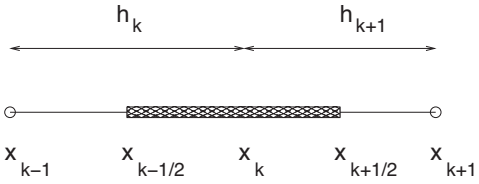


Figure 4.2: Control volume.

$x_{k+1} - x_k$ and define $x_{k-1/2} = x_k - \frac{1}{2}h_k$ and $x_{k+1/2} = x_k + \frac{1}{2}h_{k+1}$. We now integrate Equation (4.1.1) over the *control volume* $(x_{k-1/2}, x_{k+1/2})$ to obtain

$$\int_{x_{k-1/2}}^{x_{k+1/2}} -\frac{d}{dx}\left(\lambda \frac{dT}{dx}\right)\, dx = \int_{x_{k-1/2}}^{x_{k+1/2}} f\, dx, \tag{4.1.2}$$

$$-\lambda \frac{dT}{dx}\bigg|_{x_{k+1/2}} + \lambda \frac{dT}{dx}\bigg|_{x_{k-1/2}} = \int_{x_{k-1/2}}^{x_{k+1/2}} f\, dx. \tag{4.1.3}$$

Equation (4.1.3) represents the physical conservation law: the difference between the influx and outflux is equal to the production in the control volume. We may approximate the derivatives on the left-hand side by central divided differences and the integral on the right by one point integration to obtain:

$$\lambda_{k-1/2}\frac{T_k - T_{k-1}}{h_k} - \lambda_{k+1/2}\frac{T_{k+1} - T_k}{h_{k+1}} = \frac{1}{2}(h_k + h_{k+1})f_k + E_T, \tag{4.1.4}$$

which after rearrangement becomes:

$$-\frac{\lambda_{k-1/2}}{h_k}T_{k-1} + \left(\frac{\lambda_{k-1/2}}{h_k} + \frac{\lambda_{k+1/2}}{h_{k+1}}\right)T_k - \frac{\lambda_{k+1/2}}{h_{k+1}}T_{k+1} = \frac{1}{2}(h_k + h_{k+1})f_k + E_T. \tag{4.1.5}$$

The structure of the error term E_T will be considered in Exercises 4.1.2 and 4.1.3. To get a set of discrete equations we drop the error term.

Exercise 4.1.1 *Show that in case of an equidistant grid Equation (4.1.5) without the error term is identical to the finite difference discretization of (4.1.1) multiplied by the length h.* □

The error E_T in Equation (4.1.5) consist of two terms, one part of the error, E_1, originates from the use of one point integration instead of exact integration, the other part, E_2, originates from the use of central differences instead of derivatives. In the following exercise, it is shown that $E_1 = O(h_k^2 - h_{k+1}^2)$ and $E_2 = O(h_k^2 - h_{k+1}^2)$. Further, if the grid spacing is determined by $h_{k+1} = h_k(1 + O(h))$, then it can be shown that $E_1 = O(h^3)$ and $E_2 = O(h^3)$. The global error is one order lower, that is $O(h^2)$, since compared to the finite difference method all equations are multiplied by the length h.

Exercise 4.1.2 *Show that the error that originates from the one point integration is given by* $E_1 = O(h_k^2 - h_{k+1}^2)$.
Hint: Assume that $f(x)$ *is the derivative of* $F(x)$. *Express the integral in F and use Taylor series expansion.* □

Exercise 4.1.3 *Show that the error from the use of central differences is given by* $E_2 = O(h_k^2 - h_{k+1}^2)$. *You may assume that* λ *does not depend on* x. □

Exercise 4.1.4 *Show that if* $h_{k+1} = h_k(1 + O(h)), k = 1, \ldots, N$ *then* $h_{k+1} - h_k = O(h^2), k = 1 \ldots N$ *and that therefore both* $E_1 = O(h^3)$ *and* $E_2 = O(h^3)$. □

4.1.1 The boundaries

At the left-hand boundary we take $(x_0, x_{1/2})$ as control volume and we integrate to get:

$$\lambda \frac{dT}{dx}\bigg|_{x_0} - \lambda \frac{dT}{dx}\bigg|_{x_{1/2}} = \int_{x_0}^{x_{1/2}} f \, dx. \tag{4.1.6}$$

The left-hand boundary condition can be substituted directly:

$$-\lambda \frac{dT}{dx}\bigg|_{x_{1/2}} = \int_{x_0}^{x_{1/2}} f \, dx. \tag{4.1.7}$$

Application of central differences and one point integration gives:

$$\frac{\lambda_{1/2}}{h_1} T_0 - \frac{\lambda_{1/2}}{h_1} T_1 = \frac{1}{2} h_1 f_0 + E_T. \tag{4.1.8}$$

The truncation error E_T is $O(h_1^2)$ in this equation.

Exercise 4.1.5 *Show that* E_T *is* $O(h_1^2)$ *in the above equation.* □

At the right-hand boundary we take $(x_{N-1/2}, x_N)$ as control volume and integrate to get:

$$\lambda \frac{dT}{dx}\bigg|_{x_{N-1/2}} - \lambda \frac{dT}{dx}\bigg|_{x_N} = \int_{x_{N-1/2}}^{x_N} f \, dx. \tag{4.1.9}$$

On substitution of the right-hand boundary condition this becomes:

$$\lambda \frac{dT}{dx}\bigg|_{x_{N-1/2}} + \alpha T_N = \int_{x_{N-1/2}}^{x_N} f \, dx + \alpha T_R. \tag{4.1.10}$$

Application of central differences and one point integration gives:

$$-\frac{\lambda_{N-1/2}}{h_N} T_{N-1} + \left(\frac{\lambda_{N-1/2}}{h_N} + \alpha \right) T_N = \frac{1}{2} h_N f_N + \alpha T_R + E_T. \tag{4.1.11}$$

Remark
If we have for example a Dirichlet boundary condition $T = T_0$, at the left-hand side there is no need to use the control volume $(x_0, x_{1/2})$. We treat this boundary condition like in Chapter 3, i.e. we substitute the given value and no extra equation is required.

4.1.2 Conservation

Finite volume schemes are often described as *conservative schemes* for the following reason. When we write the finite volume equations in *fluxes* by applying Fick's (Darcy's, Ohm's, Fourier's) law for each finite volume (x_L, x_R) the equation looks like:

$$q_R - q_L = \int_{x_L}^{x_R} f \, dx, \tag{4.1.12}$$

or in words: what flows out minus what flows in equals the local production. This will be true *regardless of the numerical approximation to the fluxes*. If the production is zero, there will be no generation of mass (energy, momentum) by the numerical scheme. The only error that will be made in the fluxes will be caused by the error in approximating the production term.
In the following exercises we shall prove that the error in the flux is equal to the error in inflow flux plus the maximum error in the production provided the flux itself is not discretized.

Exercise 4.1.6 *Show, that if the equation*

$$-\left(\lambda y'\right)' = 0 \tag{4.1.13}$$

is discretized on the interval $(0,1)$ *by the Finite Volume Method, necessarily* $q_0 = q_N$ *with* $q = -\lambda y'$*, regardless of the number of steps* N*.* □

Exercise 4.1.7 *Show that if the equation*

$$-\left(\lambda y'\right)' = 1 \tag{4.1.14}$$

is discretized on the interval $(0,1)$ *by the Finite Volume Method, necessarily* $q_N = q_0 + 1$ *with* $q = -\lambda y'$*, regardless of the number of steps* N*.* □

We call the *error in the fluxes* $\mathbf{dq}$ and we shall calculate the various contributions to it in the following exercises.

Exercise 4.1.8 Propagation of production error
Let $dq_k - dq_{k-1} = h_k E_k$*, where* $\sum_k h_k = 1$*. Show that* $|dq_k| < |dq_0| + \sup_{j \le k} |E_j|$*.* □

Exercise 4.1.9 Propagation of boundary error
Let $dq_k - dq_{k-1} = 0$*. Show that* $dq_k = dq_0, k = 1, \ldots, N$*.* □

4.1.3 Error in the temperatures

The error in the fluxes is in general of the same order as the error in the production terms (see Exercise 4.1.8). Since we have approximated this term with one point integration, we may expect an error of magnitude $O(h^2)$ in the fluxes, q_k, for smoothly varying step sizes. By the same reasoning as in Exercise 4.1.8 we may now show, that the error in the temperatures *remains* $O(h^2)$, because if

$$-\lambda \tilde{T}'(x_{k+1/2}) = q_{k+1/2} + O(h^2), \tag{4.1.15}$$

the approximation with central differences *also* generates an $O(h^2)$ error term and we get for the error dT_k:

$$\lambda_{k+1/2} \frac{dT_k - dT_{k+1}}{h_{k+1}} = E_{k+1}, \tag{4.1.16}$$

where $E_{k+1} = O(h^2)$. Now defining *the error in temperature* $\mathbf{d}T$ in much the same way as in Exercise 4.1.8 we can show that

$$|dT_k| < |dT_N| + \sup_{j \geq k} |E_j| / \lambda_{j-1/2}. \tag{4.1.17}$$

However, by the right-hand-boundary condition we know that $q_N = \alpha(T_N - T_R)$ and that the numerical approximation to q_N has an error of $O(h^2)$. Therefore $dT_N = O(h^2)$ and backsubstitution into inequality (4.1.17) proves the result.

4.2 The stationary diffusion equation in 2 dimensions

The Finite Volume approximation of the stationary diffusion equation in two dimensions is a straightforward generalization of the previous section. Let us consider:

$$-\text{div}\, \lambda \,\text{grad}\, u = f, \quad \mathbf{x} \in \Omega, \tag{4.2.1a}$$

$$-\lambda \frac{\partial u}{\partial n} = \alpha(u - u_0), \quad \mathbf{x} \in \Gamma. \tag{4.2.1b}$$

Both λ and f are functions of the coordinates x and y. In the boundary condition the radiation coefficient α and the reference temperature u_0 are known functions of $\mathbf{x}$ and $\alpha > 0$. We subdivide the region Ω into cells like in Figure (4.3). Usually

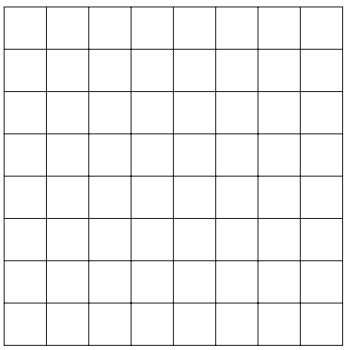


Figure 4.3: Subdivision of rectangular region into cells.

these cells are rectangles, but also quadrilaterals or even triangles are allowed. In the literature one can find two ways of positioning the unknowns. The first one is to place the unknowns in the nodes of the grid. This is called the *vertex-centered* approach. The other one is to put the unknowns in the centers of the cells (*cell-centered*). These methods only differ at the boundary of the domain. For the moment we restrict ourselves to the vertex-centered method, and a rectangular equidistant grid.
We use the same (i, j) notation for the nodes as in Chapter 3. In the literature a node x_{ij} somewhere in the interior of Ω is also denoted by x_C and the surrounding neighbors by their compass names in capitals: N, E, S, W. Cell quantities and quantities in the cell edges are denoted with lower case subscripts: n, s, e, w. If appropriate we shall also apply this notation. We construct a control volume with edges half way between two nodes, like in Figure 4.4. We integrate the equation

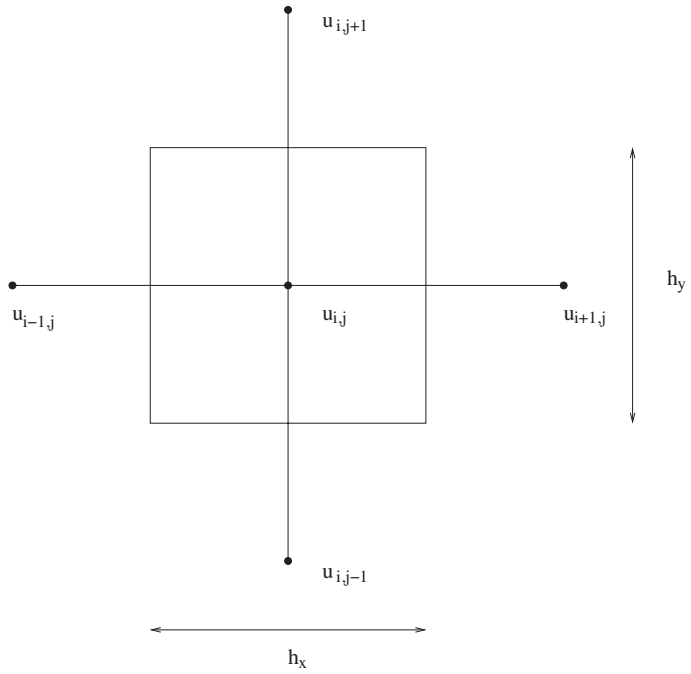


Figure 4.4: Control volume for the Diffusion equation.

over the control volume to obtain:

$$\int_V -\text{div}\,\lambda\,\text{grad}\,u\;dV = \int_V f\;dV, \tag{4.2.2a}$$

$$\oint_\Gamma -\lambda\frac{\partial u}{\partial n}\;d\Gamma = \int_V f\;dV. \tag{4.2.2b}$$

Using central differences for $\frac{\partial u}{\partial n}$ and one point integration for the left-hand-side edges and the right-hand-side volume we get the interior molecule:

$$-\lambda_{i-1/2,j}h_y\frac{u_{i-1,j}-u_{i,j}}{h_x} - \lambda_{i,j-1/2}h_x\frac{u_{i,j-1}-u_{i,j}}{h_y} - \lambda_{i+1/2,j}h_y\frac{u_{i+1,j}-u_{i,j}}{h_x} - \lambda_{i,j+1/2}h_x\frac{u_{i,j+1}-u_{i,j}}{h_y} = h_xh_yf_{i,j}. \tag{4.2.3}$$

Note that Equation (4.2.3) is identical to the finite difference Equation (3.4.2).

Exercise 4.2.1 *Derive the finite volume discretization of (4.2.1) for non-equidistant step sizes.* □

Exercise 4.2.2 *Apply the finite volume method to the* convection-diffusion equation *with incompressible flow:*

$$\text{div}\;(-\varepsilon(\text{grad}\,c) + c\mathbf{u}) = 0, \tag{4.2.4}$$

with ε and $\mathbf{u}$ constant. Show that the contribution of the convection term is non-symmetric. □

4.2.1 Boundary conditions

The treatment of boundary conditions is usually the most difficult part of the finite volume method. Dirichlet boundary conditions are treated in the same way as in 1D. The Robin boundary condition (4.2.1b) requires a special approach. For simplicity we restrict ourselves to the east boundary. All other boundaries can be dealt with in the same way. Since the nodes on the boundary correspond to the unknown function u, it is necessary to define a control volume around these points. The common approach is to take only the half part inside the domain as sketched in Figure (4.5). Integration of the Laplacian equation (4.2.1a) over the

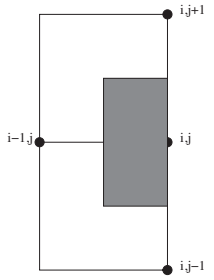


Figure 4.5: Control volume for the Robin boundary condition.

control volume gives Equation (4.2.2b). The integral over the west edge is treated as for the internal points. The integral over the north and south edges are also treated in the same way, however their length is multiplied by $\frac{1}{2}$. On the east edge boundary condition (4.2.1b) is applied to get

$$\int_{\Gamma_e} -\lambda \frac{\partial u}{\partial n}\, d\Gamma \;=\; \int_{\Gamma_e} \alpha(u - u_0)\, d\Gamma. \tag{4.2.5}$$

Discretization of (4.2.5) gives

$$\int_{\Gamma_e} \alpha(u - u_0)\, d\Gamma \approx h_y \alpha(u_{i,j} - u_0). \tag{4.2.6}$$

So the complete discretization for point $u_{i,j}$ at the boundary becomes

$$-\lambda_{i-1/2,j} h_y \frac{u_{i-1,j} - u_{i,j}}{h_x} - \lambda_{i,j-1/2} h_x \frac{u_{i,j-1} - u_{i,j}}{2h_y} - \lambda_{i,j+1/2} h_x \frac{u_{i,j+1} - u_{i,j}}{2h_y} + h_y \alpha_{i,j} u_{i,j} = h_y \alpha_{i,j} u_0 + \frac{h_x h_y}{2} f_{i,j}. \tag{4.2.7}$$

Exercise 4.2.3 *Suppose we want to solve the diffusion equation (4.2.1a) over the square $\Omega = (0,1) \times (0,1)$. Let λ and f be periodic in x-direction. Assume that we have periodical boundary conditions at the boundaries $x = 0$ and $x = 1$. Furthermore boundary condition (4.2.1b) holds for the other two boundaries.*

- *Formulate the periodical boundary conditions at $x = 0$ and $x = 1$. Motivate why the number of boundary conditions is correct.*
- *Derive the finite volume discretization of the equation at the periodical boundaries. Use an equidistant grid with $h_x = h_y$.*

□

4.2.2 Boundary conditions in case of a cell centered method

If a cell centered method is applied, cells and control volumes coincide. All unknowns are positioned in the centers of the cells, which implies that there are no unknowns on the boundary.

Exercise 4.2.4 *Show that the discretization of Equation (4.2.1a) for all internal cells (which have a common edge with the boundary), is given by Equation (4.2.3).* □

The absence of unknowns on the boundary has its effect on the treatment of boundary conditions. Neumann boundary conditions of the type

$$-\lambda \frac{\partial u}{\partial n} = g \text{ at } \Gamma, \tag{4.2.8}$$

are the most easy to implement since (4.2.8) can be substituted immediately in the boundary integrals.

Exercise 4.2.5 *Derive the discretization for a boundary cell with boundary condition (4.2.8).* □

In case of a Dirichlet boundary condition $u = g_2$ on the south boundary, it is necessary to introduce a virtual point $i, j-1$ like in Figure 3.15. The value of $u_{i,j-1}$ can be expressed in $u_{i,j}$ and the boundary value $u_{i,j-1/2}$ using linear extrapolation. Substitution in the 5-point molecule results in a 4-point stencil.

Exercise 4.2.6 *Derive the discretization of Equation (4.2.1a) in a cell adjacent to the Dirichlet boundary.* □

The Robin boundary condition (4.2.1b) is the most difficult to treat. On the boundary we have to evaluate the integral

$$\int_{\Gamma} \alpha(u - u_0)\, d\Gamma. \tag{4.2.9}$$

u is unknown, and not present on the boundary either. In order to keep the second order accuracy, the best way is express u using linear extrapolation from two internal points. Consider for example the south boundary in Figure 4.6. We can express

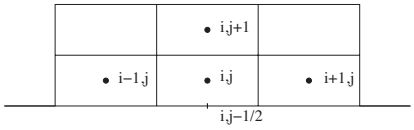


Figure 4.6: Control volume for the cell-centered Robin boundary condition.

u_B in u_C and u_N using linear extrapolation, resulting again in a 4-point molecule.

Exercise 4.2.7 *Derive the 4-point molecule.* □

4.2.3 Boundary cells in case of a skewed boundary

The best way to treat a skewed boundary is to make sure, that control volume vertices fall *on* the boundary. This leads to triangle shaped grid-cells at the boundary, see Figure 4.7. Integration over the triangle and substitution of central differences give with the notations of Figure 4.7:

$$-\beta_W u_W - \beta_S u_S + (\beta_W + \beta_S) u_C + \int_{hyp} -\lambda \frac{\partial u}{\partial n}\, d\Gamma = \frac{1}{2} h_x h_y f_C, \tag{4.2.10}$$

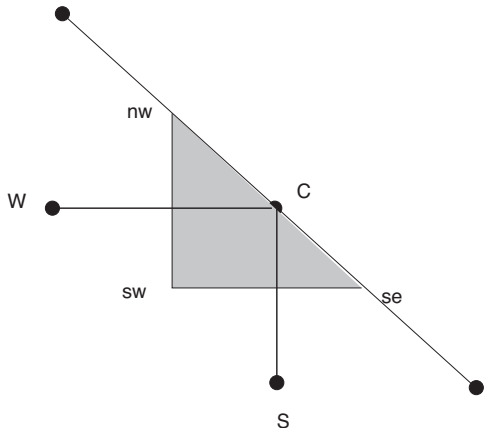


Figure 4.7: Boundary cell.

where the integral has to be taken over the hypotenuse of the triangle. Writing h_h for the length of the hypotenuse and substituting the boundary condition we get:

$$-\beta_W u_W - \beta_S u_S + (\beta_W + \beta_S + \alpha h_h) u_C = \alpha h_h u_{0C} + \frac{1}{2} h_x h_y f_C. \tag{4.2.11}$$

Remark 4.2.1 *(Symmetry and diagonal dominance)*

1. *The discretization matrix generated by the FVM is symmetric.*

2. *The numerical approximation of Problem 4.2.1 with the FVM leads to a diagonally dominant Z-matrix.*

Exercise 4.2.8 *Prove the first statement in Remark 4.2.1. (Hint: across a volume edge between, say, volumes $V_{i+1,j}$ and $V_{i,j}$ the flux is approximated in the same way for the equations of $u_{i+1,j}$ and $u_{i,j}$).* □

Exercise 4.2.9 *Prove the second statement of Remark 4.2.1.* □

Theorem 4.2.1 *Consider the Finite Volume discretization in this section of equations* (4.2.1). *If $f \geq 0$, for $\mathbf{x} \in \Omega$ and $u_0 \geq 0$, for $\mathbf{x} \in \Gamma$, then the solution of the discrete problem is positive.*

Exercise 4.2.10 *Prove Theorem 4.2.1. (Hint: use the second statement of Remark 4.2.1.)* □

Theorem 4.2.2 *Consider the Finite Volume discretization in this section of equations* (4.2.1). *The solution of the discrete problem with $f = 0$ has a maximum and minimum at the boundary.*

Exercise 4.2.11 *Prove Theorem 4.2.2. (Hint: use the second statement of Remark 4.2.1.)* □

If the boundary is curved, then the discretization with a rectangular Cartesian grid is toilsome. An alternative could be to introduce boundary fitted coordinates.

4.2.4 Error considerations in the interior

We shall not go into great detail in error analysis, but indicate sources of error. We started out by integrating the conservation law of the flux vector *exactly*:

$$\Phi_w + \Phi_n + \Phi_e + \Phi_s = \int_V f \, dV, \tag{4.2.12}$$

where Φ stands for the net outflow through that particular edge of the control volume. After that we made a number of approximations:

1. Approximate integrals over the sides by one point integration. $O(h^2)$ accurate for smoothly changing step sizes, otherwise $O(h)$.
2. Approximate derivatives by central differences. $O(h^2)$ accurate for smoothly changing step sizes otherwise $O(h)$.
3. Approximate the right-hand side by one point integration. $O(h^2)$ accurate for smoothly changing step sizes otherwise $O(h)$.

It gets monotonous. From finite difference approximations we already know, that *equidistant* step sizes lead to overall $O(h^2)$ accuracy. So what are smoothly varying step sizes? Roughly speaking it says, that between neighboring step sizes there may be a factor $1 + O(h)$. This still gives pretty much leeway in stretching grids, so that should not be regarded as too restrictive.

4.2.5 Error considerations at the boundary

At the boundary one point integration of the right-hand side is always $O(h)$, because the integration point has to be the gravicenter for order $O(h^2)$ accuracy, whereas the integration point is always on the edge. (Note that in fact the absolute magnitude of the error is $O(h^3)$, but that is because the volume of integration is itself $O(h^2)$.)

So the situation looks grim, but in fact there is nothing that should worry us. And that is because of the following phenomenon: for the solution $\mathbf{u}$ of the discrete equations with $f = 0$, we have

$$\|\mathbf{u}\|_\infty \leq \sup_{\mathbf{x}\in\Gamma} |u_0|. \tag{4.2.13}$$

Exercise 4.2.12 *Prove Inequality* (4.2.13)*. Use the results of Exercises 4.2.9 sseq.* □

Exercise 4.2.13 *Prove that if $\tilde{u}_0 = u_0 + \varepsilon_0$ the perturbation ε in the solution of the homogeneous discrete problem is less than* $\sup|\varepsilon_0|$ *for all components of ε. (Hint: subtract the equations and boundary conditions of u and $\tilde{u}$ to obtain an equation and boundary condition for ε. Then use* (4.2.13)*)* □

From all this we see, that a perturbation of $O(h^3)$ in the right-hand side of equations for the boundary cells leads to an error of $O(h^2)$ in the solution. But one point integration of the right-hand side *also* gives a perturbation of $O(h^3)$. So the effect on the solution should *also* be no worse than $O(h^2)$.

4.3 Laplacian in general coordinates

4.3.1 Discrete transformation from Cartesian to General coordinates

Consider a region in the x-y-plane as in Figure 4.8 that we want to transform into a rectangular region in the ξ-η-plane. We assume that there is a formal mapping

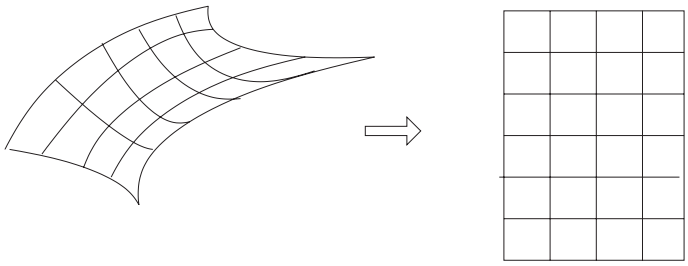


Figure 4.8: General region transformation.

$x(\xi,\eta)$ and $y(\xi,\eta)$ and its inverse $\xi(x,y)$ and $\eta(x,y)$ exists. Coordinate lines in the ξ-η-plane transform to the curves $\mathbf{x}(\xi_0,\eta)$ and $\mathbf{x}(\xi,\eta_0)$ respectively. Such a transformation is called regular if it has an inverse, otherwise it is singular. Sufficient conditions for regularity is, that the *Jacobian* exists and is non-singular. The quantities needed to calculate the transformations are the derivatives of the transformation matrices:

$$J = \begin{pmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{pmatrix} \tag{4.3.1}$$

and its inverse

$$J^{-1} = \begin{pmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{pmatrix}. \tag{4.3.2}$$

Usually the mapping is only known in the cell vertices. This means that we do not have an analytical expression for the derivatives and we must compute them by finite differences. Unfortunately not all derivatives are easily available. Take a look at a cell in the ξ-η-plane (Figure 4.9): Given the configuration in Figure 4.9, central

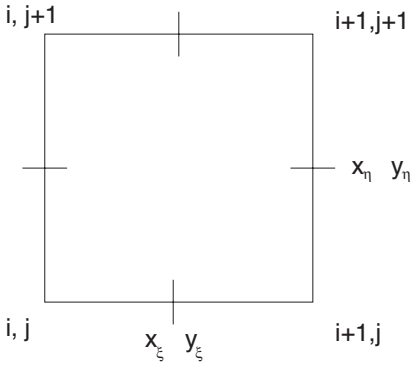


Figure 4.9: One cell with natural place of coordinate derivatives.

differences can be applied to compute x_ξ and y_ξ at the midpoints of the horizontal cell boundaries. Analogously, central differences are applied to compute x_η and y_η at the vertical cell boundaries. Everything else has to be computed by averaging over the neighbors. The quantities ξ_x, ξ_y etcetera have to be calculated by inverting J.

Exercise 4.3.1 *Explain how to express $x_\xi, x_\eta, y_\xi, y_\eta$ in the cell center in the $\xi\eta-$plane (see Figure 4.9) in the cell coordinates in the $xy-$plane. Explain how to calculate ξ_x, ξ_y, η_x and η_y.* □

In the Finite Volume Method, we consider integration of a function, or of a differential expression. If a regular transformation is applied from (x, y) to (ξ, η), then the *Jacobian* enters the picture. Suppose that we integrate over a domain Ω that is defined in the (x, y)-space, and suppose that Ω is mapped onto $\overline{\Omega}$ in the (ξ, η)-space, then, from Calculus, it follows

$$\int_{\Omega_{xy}} f(x,y)d\Omega_{xy} = \int_{\Omega_{\xi\eta}} f(x(\xi,\eta), y(\xi,\eta))|\frac{\partial(x,y)}{\partial(\xi,\eta)}|d\Omega_{\xi\eta}, \tag{4.3.3}$$

where the Jacobian is defined by

$$|\frac{\partial(x,y)}{\partial(\xi,\eta)}| = |det(J)|,$$

which is expressed in the co-ordinate framework (ξ, η). We use the notation $d\Omega_{xy}$ and $d\Omega_{\xi\eta}$ to emphasize that the integral is in the (x, y) and (ξ, η) framework respectively. For the derivation of this procedure, we refer to a textbook on Calculus, like Steward, Adam or Almering. This procedure is applied in general to all integrals that are involved in the Finite Volume discretization. We will illustrate how the finite volume method works in a polar co-ordinate system.

4.3.2 An example of finite volume integration in polar co-ordinates

We will consider an example on a cut piece of cake, on which Poisson's equation is imposed

$$-\text{div grad } u = f(x,y), \text{ on } \Omega, \tag{4.3.4}$$

where Ω is described in polar co-ordinates by

$$\Omega_{r\theta} = \{(r,\theta) \in R^2 : 1 < r < 3,\ 0 < \theta < \pi/4\}.$$

To solve the above equation by Finite Volumes, the equation is integrated over a control volume V, to obtain

$$-\int_V \text{div grad } u d\Omega_{xy} = \int_V f(x,y)d\Omega_{xy}. \tag{4.3.5}$$

From Equation (3.7.9), we know that the above PDE (4.3.4) is transformed into

$$-\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2}\right) = f(rcos\theta, rsin\theta). \tag{4.3.6}$$

Note that $\Omega_{r\theta}$ is a rectangular domain in the (r, θ)- co-ordinate framework. The Jacobian of the transformation from Cartesian co-ordinates to polar co-ordinates is given by

$$|\frac{\partial(x,y)}{\partial(r,\theta)}| = r. \tag{4.3.7}$$

Exercise 4.3.2 *Prove the above formula.* □

Next, we integrate the transformed PDE (4.3.6) over the transformed control volume, which is rectangular and hence much easier to work with, to get

$$\int_V -\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2}\right) r d\Omega_{r\theta} = \int_V f(rcos\theta, rsin\theta) r d\Omega_{r\theta}. \tag{4.3.8}$$

Note that the Jacobian has been implemented on both sides of the above equation. The integral of the left-hand side of the above equation can be worked out such that

$$\int_V -\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) - \frac{1}{r}\frac{\partial^2 u}{\partial \theta^2} d\Omega_{r\theta} = -\int_V (\frac{\partial}{\partial r}, \frac{\partial}{\partial \theta}) \cdot (r\frac{\partial u}{\partial r}, \frac{1}{r}\frac{\partial u}{\partial \theta}) d\Omega_{r\theta}. \tag{4.3.9}$$

The integrand in the right-hand side of the above equation, consists of an inner product of the Divergence operator and a vector field. Both vectors are in the (r, θ) frame. The domain over which the integral is determined is closed and hence the Divergence Theorem can be applied in this volume with piecewise straight boundaries. This implies that equation (4.3.8) can be written as

$$-\int_{\delta V} (n_r, n_\theta) \cdot (r\frac{\partial u}{\partial r}, \frac{1}{r}\frac{\partial u}{\partial \theta}) d\Gamma = \int_V f(rcos\theta, rsin\theta) r d\Omega_{r\theta}. \tag{4.3.10}$$

This equation contains a volume integral with the function f over a control volume and a line integral related to the Laplacian over the boundary of the control volume. The treatment of both integrals is analogous to the Cartesian case: Consider the control volume, with length Δr and $\Delta\theta$, around C, with co-ordinates (r_C, θ_C) in Figure 4.10. The integral at the right-hand side in the above equation is approximated by

$$\int_V f(rcos\theta, rsin\theta) r d\Omega_{r\theta} \approx f(r_C, \theta_C) r_C \Delta r \Delta\theta. \tag{4.3.11}$$

The boundary integral is obtained by the sum of the approximations of integrals over all the boundary segments. Substitution of these approximations into (4.3.10), gives the final result for an internal control volume:

$$\frac{1}{r_C}\frac{u_S - u_C}{\Delta\theta}\Delta r + r_e\frac{u_E - u_C}{\Delta r}\Delta\theta + \frac{1}{r_C}\frac{u_N - u_C}{\Delta\theta}\Delta r + r_w\frac{u_W - u_C}{\Delta r}\Delta\theta = f(r_C, \theta_C) r_C \Delta r \Delta\theta. \tag{4.3.12}$$

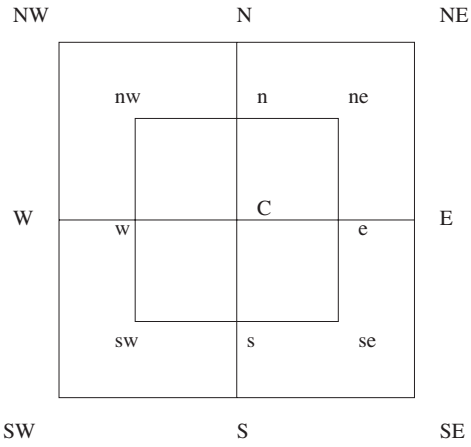


Figure 4.10: General control volume.

4.3.3 Boundary conditions

Boundary conditions of Dirichlet type do not present any problem, so we shall turn our attention to radiation boundary conditions of the form

$$\frac{\partial u}{\partial n} = \alpha(u_0 - u).$$

From an implementation point of view, it is easiest to take the nodal points *on* the boundary, which gives us a half cell control volume at the boundary like in Figure 4.11. Integrating over the half volume and applying the divergence theorem

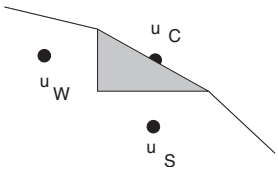


Figure 4.11: Boundary cell.

we get:

$$\frac{1}{r_C}\frac{u_S - u_C}{\Delta\theta}\frac{\Delta r}{2} + r_C\alpha(u_0 - u_C)\Delta\theta + \frac{1}{r_C}\frac{u_N - u_C}{\Delta\theta}\frac{\Delta r}{2} + r_w\frac{u_W - u_C}{\Delta r}\Delta\theta = f(r_C, \theta_C)r_C\Delta r\frac{\Delta\theta}{2}, \tag{4.3.13}$$

where the radiation boundary condition has been substituted into the boundary integral of the right (east) boundary of the control volume.

4.3.4 Error analysis

We did a comparable error analysis of the Laplace equation in Cartesian coordinates in Section 4.2.5. Consider the one point integration of the boundary volume on the right-hand side:

$$\int_V \sqrt{g} f dV = \frac{1}{2}h^2(\sqrt{g}f)_C + O(h^3), \tag{4.3.14}$$

as you can simply verify by Taylor expansion. So this "inaccurate" integration of the right-hand side gives a perturbation of $O(h^3)$ in the approximated right-hand side. The same is true for the integrations along n and s sides, *because they integrate the same integrand and are subtracted from each other.*

Theorem 4.3.1 *For sufficiently smooth f*

$$\int_x^{x+h} f(x, y+h) - f(x, y)dx = h(f(x, y+h) - f(x, y)) + O(h^3). \tag{4.3.15}$$

Proof
Let $F(x, y)$ be such, that $F_x(x, y) = f(x, y)$, then apparently

$$\int_x^{x+h} f(x, y+h) - f(x, y)dx = F(x+h, y+h) - F(x, y+h) - F(x+h, y) + F(x, y). \tag{4.3.16}$$

Now by Taylors theorem:

$$F(x+h,y+h) = F(x,y+h) + hf(x,y+h) + \frac{h^2}{2}f_x(x,y+h) + O(h^3), \quad (4.3.17a)$$

$$F(x+h,y) = F(x,y) + hf(x,y) + \frac{h^2}{2}f_x(x,y) + O(h^3). \quad (4.3.17b)$$

Subtracting the two equations in (4.3.17) we get:

$$\int_x^{x+h} f(x,y+h) - f(x,y)dx = h(f(x,y+h) - f(x,y)) + \frac{h^2}{2}(f_x(x,y+h) - f_x(x,y)) + O(h^3). \quad (4.3.18)$$

From the mean value theorem we note that $f_x(x,y+h) - f_x(x,y) = O(h)$ and the result follows. □

So all "inaccurate" integrations produce an $O(h^3)$ perturbation in the right-hand side. And now we use the same argument as in Section 4.2.5. The perturbation in the solution of the homogeneous Laplacian is always *less* than the perturbation in the boundary condition. But a perturbation of $O(h^3)$ in the right-hand side of a boundary volume equation is equivalent to an $O(h^2)$ perturbation in the boundary condition, hence causes an $O(h^2)$ perturbation in the solution. Because we no longer have the discrete maximum principle at our disposal it is not so easy to formally prove this assertion. But the least we can say is, that if our discrete solution converges to the solution of the continuous problem it is $O(h^2)$ accurate, because we *do* have a maximum principle for the continuous problem.

4.4 Finite volumes on two component fields

We shall show an example of application of the FVM on a two component field. We recall the problem for planar stress from Section 2.4.4. We consider a rectangular plate fixed at the sides ABC and subject to a body force $\mathbf{b}$ inside $\Omega = ABCD$ and boundary stresses t at the two free sides CDA. See Figure 4.12 The equation for the stresses are:

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + b_1 = 0, \quad (4.4.1a)$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + b_2 = 0, \quad (4.4.1b)$$

We integrate the first equation over a control volume V_1 and the second one over a control volume V_2. We define

$$\mathbf{s}_x = \begin{pmatrix} \sigma_{xx} \\ \tau_{xy} \end{pmatrix} \quad \text{and} \quad \mathbf{s}_y = \begin{pmatrix} \tau_{xy} \\ \sigma_{yy} \end{pmatrix}. \quad (4.4.2)$$

After application of Gauss' divergence theorem we obtain:

$$\oint_{\Gamma_1} \mathbf{s}_x \cdot \mathbf{n}\, d\Gamma + \int_{V_1} b_1\, dV = 0, \quad (4.4.3a)$$

$$\oint_{\Gamma_2} \mathbf{s}_y \cdot \mathbf{n}\, d\Gamma + \int_{V_2} b_2\, dV = 0, \quad (4.4.3b)$$

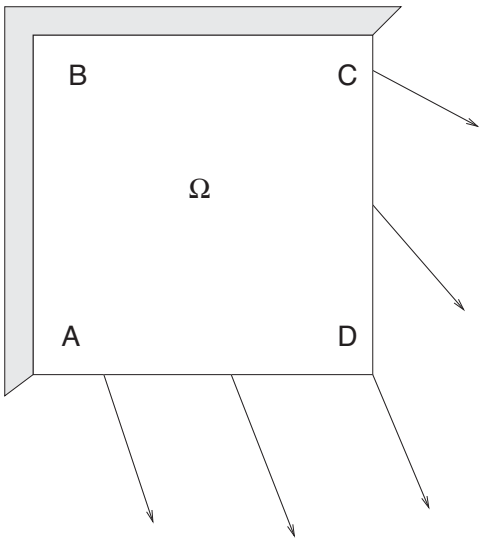


Figure 4.12: Square plate.

or

$$\int_{e_1} \sigma_{xx}\,dy - \int_{w_1} \sigma_{xx}\,dy + \int_{n_1} \tau_{xy}\,dx - \int_{s_1} \tau_{xy}dx = -h_x h_y b_1, \tag{4.4.4a}$$

$$\int_{e_2} \tau_{xy}\,dy - \int_{w_2} \tau_{xy}\,dy + \int_{n_2} \sigma_{yy}\,dx - \int_{s_2} \sigma_{yy}dx = -h_x h_y b_2. \tag{4.4.4b}$$

It is not self evident, that the control volumes for the two force components should be the same for Equation (4.4.4a) as for Equation (4.4.4b) and in fact we shall see that a very natural choice will make them different.

4.4.1 Staggered grids

We apply the finite volume method with volume V_1 to Equation (4.4.4a) and we express the stress tensor components in the *displacements* u and v. In e_1 we now need to have u_x and v_y, so in fact we would like to have u_E, u_C, v_{ne} and v_{se} in order to make compact central differences around e_1. Checking the rest of the sides of V_1 makes it clear, that we need: u_E, u_S, u_W, u_N, u_C and $v_{ne}, v_{nw}, v_{sw}, v_{se}$, see Figure 4.13.

Exercise 4.4.1 *Derive the discretization in the* displacement *variables u and v for Equation* (4.4.4a) *in the V_1 volume.* □

When we apply FVM with volume V_2 to Equation (4.4.4b) we need u_y and v_x in e_2, so now we would like to have v_E, v_C, u_{ne} and u_{se}.

Exercise 4.4.2 *Derive the discretization in the* displacement *variables u and v for Equation* (4.4.4b) *in the V_2 volume.* □

So apparently we must choose a grid in such a way that both V_1 and V_2 can be accommodated and the natural way to do that is take u and v in different nodal points, like in Figure 4.14.
Such an arrangement of nodal point is called a *staggered grid*. This means that in general different problem variables reside in different nodes.

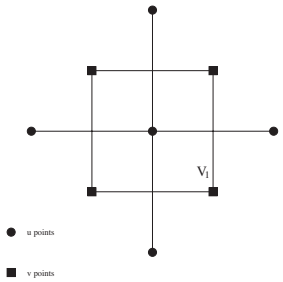


Figure 4.13: V_1-variables.

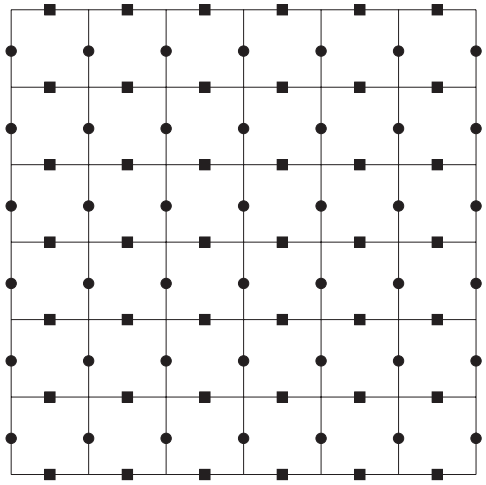


Figure 4.14: Staggered grid.

4.4.2 Boundary conditions

When discretizing a scalar equation you can often choose the grid in such a fashion, that the boundary conditions can be easily implemented. With two or more components especially on a staggered grid this is no longer true.

Consider the W boundary of our fixed plate in Figure 4.12. On this boundary we have the boundary conditions $u = 0$ and $v = 0$. A quick look at the staggered grid of Figure 4.14 shows a fly in the ointment. The u-points are on the boundary all right. Let us distinguish between equations derived from Equation (4.4.4a) (type 1) and those derived from Equation (4.4.4b) (type 2). In equations of type 1 you can easily implement the boundary conditions on the W-boundary. By the same token, you can easily implement the boundary condition on the N-boundary in type 2 equations. For equations of the "wrong" type you have to resort to a trick. The generic form of an equation of type 2 in the displacement variables is:

$$B_W v_W + B_{nw} u_{nw} + B_N v_N + B_{ne} u_{ne} + B_E v_e + B_{se} u_{se} + B_S v_S + B_{sw} u_{sw} + B_C v_C = h^2 b_C. \tag{4.4.5}$$

To implement the boundary condition on the W-side in equations of type 2, we assume a virtual ("ghost") grid point on the other side of the wall acting as W-point, see Figure 4.15

Now we eliminate v_W by linear interpolation: $(v_W + v_C)/2 = 0$, hence $v_W =$

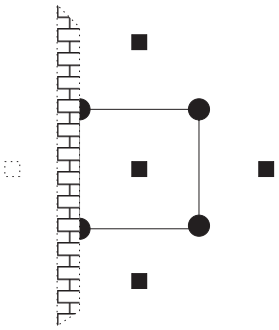


Figure 4.15: Ghost point.

$-v_C$ and Equation (4.4.5) transforms into

$$B_{nw}u_{nw} + B_N v_N + B_{ne}u_{ne} + B_E v_e + B_{se}u_{se} + B_S v_S + B_{sw}u_{sw} + (B_C - B_W)v_C = h^2 b_C. \tag{4.4.6}$$

Exercise 4.4.3 *Explain how to implement the boundary condition on the N-boundary in equations of type 1.* □

The boundary conditions on the E- and S boundary are natural boundary conditions. When a boundary of a full volume coincides with such a boundary, there are no problems, the boundary condition can be substituted directly. That is equations of type 2 are easy at the E-boundary, equations of type 1 are easy at the S-boundary.

Exercise 4.4.4 *Derive the equation of type 1 at the S-boundary in the displacements and substitute the natural boundary condition.* □

What of the half volumes? Consider an equation of type 1 at the E-boundary. (Figure 4.16)
Let us integrate Equation (4.4.1a) over a half volume V_1 to obtain:

$$h(-s_{xxw} + s_{xxC}) + \frac{1}{2}h(\tau_{xyn} - \tau_{xys}) = -\frac{1}{2}h^2 b_{1C} \tag{4.4.7}$$

Since by the natural boundary conditions $s_{xx} = f_1$ and $\tau_{xy} = f_2$ are given quantities at the boundary this transforms into

$$hs_{xxW} = hf_{1C} + \frac{1}{2}h(f_{2n} - f_{2s})) + \frac{1}{2}h^2 b_{1C}. \tag{4.4.8}$$

Again one point integration of the right-hand side causes a perturbation of $O(h^3)$, because it is not in the gravicenter of the volume, and also the integration along the n- and s-sides of the volume has an error of $O(h^3)$.

Exercise 4.4.5 *Prove these last two assertions. Compare your results with Section 4.3.4* □

Since this perturbation is of the same order as a perturbation of $O(h^2)$ in the stresses applied at the boundary, we may expect that this gives a perturbation of the same order in the displacements u and v.

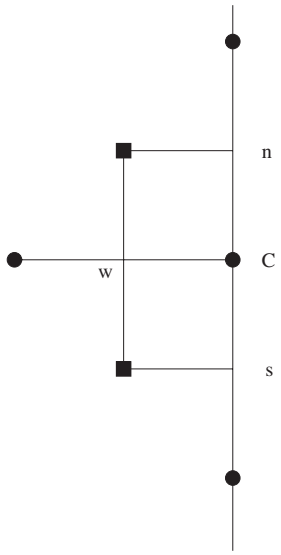


Figure 4.16: Half volume at natural boundary.

4.5 Project: Stokes equations for incompressible flow

A fairly simple and admittedly artificial model for stationary viscous incompressible flow is represented by the *Stokes Equations*:

$$-\operatorname{div} \mu \operatorname{grad} u + \frac{\partial p}{\partial x} = 0 \tag{4.5.1a}$$

$$-\operatorname{div} \mu \operatorname{grad} v + \frac{\partial p}{\partial y} = 0 \tag{4.5.1b}$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{4.5.1c}$$

In these equations the first two ones describe the equilibrium of the viscous stresses, the third equation is the incompressibility condition. The viscosity μ is a given material constant, but the velocities u and v and the pressure p have to be calculated. Let us consider this problem in a straight channel (see Figure 4.17).

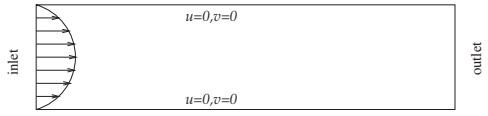


Figure 4.17: Channel for Stokes flow.

At the inlet the velocities are given: $u = u_0(y), v = v_0(y)$, the channel walls allow no slip, so $u = 0$ and $v = 0$ at both walls. At the outlet there is a reference pressure p_0 in the natural boundary conditions: $-\mu\frac{\partial u}{\partial x} + p = p_0$ and $\frac{\partial v}{\partial x} = 0$.

To solve the equations, we use a staggered approach, in which the unknowns are ordered as in Figure 4.18. For the horizontal component of the velocity u, the

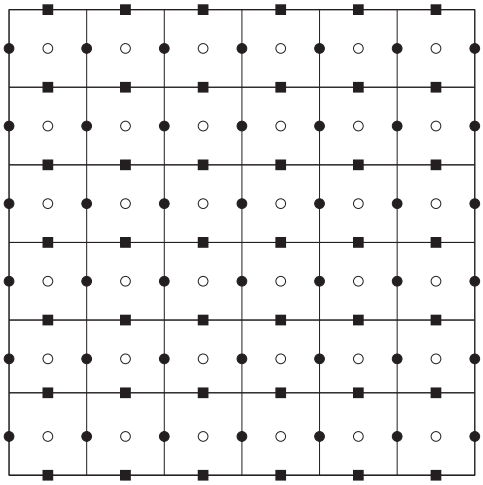


Figure 4.18: The ordering of the unknowns in a staggered approach for the Stokes equations. The solid circles and squares respectively correspond to u and v indicating the horizontal and vertical components of the fluid velocity. The open circles denote the pressure nodes.

finite volume method gives

$$-\int_{\Omega_u} \nabla \cdot (\mu \nabla u) d\Omega + \int_{\Omega_u} \frac{\partial p}{\partial x} d\Omega = 0, \tag{4.5.2}$$

where Ω_u is a control volume with a u-node as the center. The Divergence Theorem yields

$$-\int_{\Gamma_u} \mu \frac{\partial u}{\partial n} d\Gamma + \int_{\Gamma_u} p n_x d\Gamma = 0. \tag{4.5.3}$$

This equation is discretized by similar procedures as the Laplace equation. Note that n_x represents the horizontal component of the unit outward normal vector. The equation for the vertical component of the velocity is worked out similarly, to get

$$-\int_{\Gamma_v} \mu \frac{\partial v}{\partial n} d\Gamma + \int_{\Gamma_v} p n_y d\Gamma = 0. \tag{4.5.4}$$

Subsequently, we consider the continuity equation div $\mathbf{u} = 0$. This equation is discretized using a control volume with a pressure node as the center:

$$\int_{\Omega_p} \text{div } \mathbf{u} d\Gamma = \int_{\Gamma_p} \mathbf{u} \cdot \mathbf{n} d\Gamma. \tag{4.5.5}$$

For the implementation of the outlet condition $-\mu \frac{\partial u}{\partial x} + p = p_0$, we use half a cell over a u-node, in which the integral over the right (east) boundary is given by

$$\int_{\delta\Omega_u^R} \left(-\mu \frac{\partial u}{\partial x} + p n_x \right) d\Gamma = \int_{\Gamma_u^R} p_0 d\Gamma \approx p_0 h.$$

Exercise 4.5.1 *Derive discrete equations for all three volumes Ω_u, Ω_v and Ω_p. Note that the pressure and equation of continuity are coupled, that is, the continuity equation is integrated over a pressure cell.* □

Exercise 4.5.2 *Explain how the no slip boundary conditions are implemented in the equations (*Hint: Use ghost points and averaging in the spirit of Section 4.4.2.*).* □

Exercise 4.5.3 *Explain how to implement the inlet boundary conditions.* □

Exercise 4.5.4 *Take care to end in a vertical line with u points at the outlet. Now explain how to implement the outlet boundary conditions. Argue why you ended up with as many equations as unknowns.* □

Exercise 4.5.5 *In the half Ω_u volume at the outlet boundary the one point integrations over the horizontal edges cause an error of $O(h^3)$. Show this and argue, that this is equivalent to a perturbation of $O(h^2)$ in the reference pressure p_0.* □

4.6 Summary of Chapter 4.

We have learned a new way to discretize: the *Finite Volume Method*, especially suited to conservation laws. We have seen a one dimensional and a two dimensional example with non equidistant stepsizes and radiation boundary conditions. Despite the fact, that at the boundary the accurate midpoint integration rule was replaced by less accurate one point integration we have shown or made plausible that that would not affect the overall accuracy of the solution. We concluded the chapter with extensive treatment of the Laplacian in curvilinear coordinates and an example of the two component problem of planar stress. We have seen, that for problems of that kind it is sometimes useful to take the variables in different node points: staggered grids.

Chapter 5

Minimization problems in physics

Objectives

In Chapter 1 we have seen that many physical partial differential equations (PDEs) are the result of conservation laws. A completely different way to derive PDEs is by minimizing an integral. Examples of this approach are: shortest path and minimal potential energy. In this chapter we shall show how to derive a PDE with corresponding boundary conditions starting from a minimization problem.

On the other hand it is possible, under certain conditions, to derive a minimization problem, that in some sense is equivalent to a given PDE. If the PDE has a solution in the classical sense, solution of the minimization problem means also solution of the corresponding PDE and vice versa.

Minimization problems usually admit a larger solution class than a PDE formulation and therefore the solution of the minimization problem is referred to as generalized solution of the PDE. A similar formulation is possible for problems that do not fit the minimization frame work. This formulation, the *weak formulation* will be treated in Chapter 7. It will be shown that this formulation may be considered as a kind of conservation law.

5.1 Introduction

Mathematical models in physics are often derived from conservation laws (see Section 1.3.5), but also from minimization problems (Section 1.4). In this chapter we shall focus on the latter category.

Before analyzing these minimization problems in general, we shall start with the simple example of minimum potential energy, already treated in Section 1.4.1.

5.1.1 Minimal potential energy

In Section 1.4.1 we have seen that the potential energy of an elastic string fixed in $(0, 0)$ and $(0, 1)$ with a given load is defined by

$$\int_0^1 \left\{ \frac{1}{2} k \left(\frac{du}{dx} \right)^2 - uf \right\} dx \,. \tag{5.1.1}$$

Hence the displacement u minimizes the integral (5.1.1) under the conditions:

$$u(0) = 0\,, \quad u(1) = 0\,.$$

We shall consider a slightly more general minimization problem:

Find the function u that minimizes $I(u)$ defined by (5.1.1) such that

$$u(0) = u_0\,. \tag{5.1.2}$$

In the remainder of this chapter we shall always assume that u is sufficiently smooth, which means that implicitly we suppose that all expressions we use and operations we apply are allowed. Later on we shall specify this more precisely.

In Section 5.4 we shall give a number of often classical examples of minimization problems.

The minimization problem (5.1.1), (5.1.2) is different from standard minimization problems in the sense that we have to find a continuous function instead of a finite set of parameters. In fact we may consider this as a problem with an infinite number of unknowns.

5.1.2 Derivation of the differential equation

In order to show that the solution of the minimization problem satisfies a certain differential equation we use a reasoning due to Euler.
We suppose that (5.1.1) with boundary condition (5.1.2) has a smooth solution, which we call $\hat{u}(x)$. We consider a class of functions $u(x)$ defined as

$$u(x) = \hat{u}(x) + \varepsilon\eta(x)\,. \tag{5.1.3}$$

(5.1.3) will be referred to as a variation around $\hat{u}(x)$.

ε is a variable parameter and η is some arbitrary but fixed function. Since both $\hat{u}(x)$ and $u(x)$ must satisfy the boundary condition (5.1.2) it is necessary that

$$\eta(0) = 0\,. \tag{5.1.4}$$

Also η is assumed to be sufficiently smooth.
Substitution of (5.1.3) in (5.1.1) gives

$$I(\hat{u}+\varepsilon\eta) = \int_0^1 \left\{ \frac{1}{2}k\left(\frac{d(\hat{u}+\varepsilon\eta)}{dx}\right)^2 - (\hat{u}+\varepsilon\eta)f \right\} dx\,, \tag{5.1.5}$$

under the conditions (5.1.2) and (5.1.4).

I is a function of ε only (why?), and according to the classical theory of minimization problems, a necessary condition for the existence of an extreme is

$$\frac{dI}{d\varepsilon} = 0\,, \tag{5.1.6}$$

hence

$$\int_0^1 \left\{ k\frac{d(\hat{u}+\varepsilon\eta)}{dx}\frac{d\eta}{dx} - \eta f \right\} dx = 0\,. \tag{5.1.7}$$

Exercise 5.1.1 *Show that Equation (5.1.7) follows from (5.1.5) and (5.1.6)* □

From (5.1.3) it is clear that $I(\varepsilon)$ reaches its minimum for $\varepsilon = 0$, hence (5.1.7) reduces to

$$\int_0^1 \left\{ k\frac{d\hat{u}}{dx}\frac{d\eta}{dx} - \eta f \right\} dx = 0 \,, \qquad (5.1.8)$$
$$\hat{u}(0) = u_0 \,, \quad \eta(0) = 0 \,.$$

Since $\eta(x)$ is an arbitrary function, (5.1.8) must be valid for any η satisfying (5.1.4). Unfortunately we see in (5.1.8) both η and $\frac{d\eta}{dx}$ in the integral. In order to get an expression in η only, we apply integration by parts to the first term. This results in

$$\int_0^1 \left\{ -\eta\frac{d}{dx}\left(k\frac{d\hat{u}}{dx}\right) - \eta f \right\} dx + \eta k\frac{d\hat{u}}{dx}\Big|_0^1 = 0 \,. \qquad (5.1.9)$$

Due to the boundary condition (5.1.4) this reduces to:

$$\int_0^1 \eta\left(-\frac{d}{dx}\left(k\frac{d\hat{u}}{dx}\right) - f \right) dx + \eta(1)k(1)\frac{d\hat{u}}{dx}(1) = 0 \,. \qquad (5.1.10)$$

(5.1.10) must be valid for all $\eta(x)$ with $\eta(0) = 0$. Let us first consider the subset that also satisfies $\eta(0) = \eta(1) = 0$. Now (5.1.10) reduces to

$$\int_0^1 \eta\left(-\frac{d}{dx}\left(k\frac{d\hat{u}}{dx}\right) - f \right) dx = 0 \,, \qquad (5.1.11)$$

for all η with $\eta(0) = \eta(1) = 0$.

Hence the solution $\hat{u}(x)$ must satisfy the differential equation (using Lemma of Dubois-Reymond 5.2.2)

$$-\frac{d}{dx}\left(k\frac{du}{dx}\right) = f \,, \qquad (5.1.12)$$

with boundary condition $u(0) = u_0$.

(We shall show this more rigorously in Section 5.2).

(5.1.12) is a second order linear differential equation. So we need two boundary conditions in order to get a unique solution. To that end we consider the complete class of functions $\eta(x)$, with $\eta(0) = 0$. Substituting (5.1.12) in (5.1.10) gives:

$$\eta(1)k(1)\frac{d\hat{u}}{dx}(1) = 0 \,, \qquad (5.1.13)$$

with $\eta(1)$ arbitrary.

Hence, we arrive at

$$k(1)\frac{du}{dx}(1) = 0 \,, \qquad (5.1.14)$$

which is our second boundary condition.

So we started with the minimization problem (5.1.1) with one boundary condition (5.1.2) and we showed that the solution must satisfy the second order differential equation (5.1.12) with two boundary conditions (5.1.2) and (5.1.14). Apparently boundary condition (5.1.14) is hidden in the minimization problem. Such a boundary condition, that is not imposed explicitly, is called a *natural boundary condition*. Boundary condition (5.1.2), which must be satisfied both by the minimization problem and the differential equation is called an *essential boundary condition*. It limits the class in which to look for a solution.

In the next section we shall consider a more general problem in one dimension.

5.2 A general one-dimensional problem with first order derivatives

In the previous section we have seen how one can derive a differential equation from a minimization problem. In this section we consider a general minimization in 1-d with first order derivatives.

Theorem 5.2.1

Let $f(x,u,p)$ be a sufficiently smooth function.

Consider the minimization problem

$$\min_u \quad l(u) = \min_u \quad \int_{x_0}^{x_1} f(x,u,u') \, dx \,, \tag{5.2.1}$$

with boundary condition

$$u(x_0) = u_0 \,. \tag{5.2.2}$$

u' is a short notation for $\frac{du}{dx}$.
If a solution $\hat{u}$ of problem (5.2.1), (5.2.2) exists, then this solution must satisfy the differential equation

$$\frac{\partial f}{\partial u} - \frac{d}{dx}\frac{\partial f}{\partial u'} = 0 \,, \tag{5.2.3}$$

with boundary conditions

$$\hat{u}(x_0) = u_0 \quad \textit{(essential)} \,, \tag{5.2.4}$$

and

$$\frac{\partial f}{\partial u'}(x_1) = 0 \quad \textit{(natural)} \,. \tag{5.2.5}$$

REMARK: with $\frac{\partial f}{\partial u'}$ we mean: differentiate $f(x,u,p)$ to p and substitute $\frac{du}{dx}$ for p.

Proof
Consider the following family of curves around the solution $\hat{u}(x)$:

$$u(x) = \hat{u}(x) + \varepsilon\eta(x) \,, \tag{5.2.6}$$

with ε an arbitrary parameter and $\eta(x)$ an arbitrary, sufficiently smooth curve satisfying $\eta(x_0) = 0$.

Substitution of (5.2.6) in (5.2.1) gives

$$l(u) = \int_{x_0}^{x_1} f(x, \hat{u} + \varepsilon\eta(x), \hat{u}' + \varepsilon\eta'(x)) \, dx \,. \tag{5.2.7}$$

The integral in (5.2.7) is a function of ε denoted by $I(\varepsilon)$.
$I(\varepsilon)$ is minimal for $u = \hat{u}(x)$, hence $\varepsilon = 0$.
A necessary condition for the existence of a minimum in $\varepsilon = 0$ is

$$\left.\frac{dI(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0} = 0 \,, \tag{5.2.8}$$

or

$$\int_{x_0}^{x_1} \left\{ \frac{\partial f}{\partial u}\left(x, \hat{u}, \hat{u}'\right)\eta(x) + \frac{\partial f}{\partial u'}(x, \hat{u}, \hat{u}')\eta' \right\} dx = 0 \,. \tag{5.2.9}$$

Integration by parts of the last term results in

$$\int_{x_0}^{x_1} \left[\frac{\partial f}{\partial u} - \frac{d}{dx}\frac{\partial f}{\partial u'} \right] \eta(x)\, dx + \left[\eta(x) \frac{\partial f}{\partial u'} \right]_{x_0}^{x_1} = 0 \,. \tag{5.2.10}$$

$\eta(x)$ is an arbitrary smooth function with $\eta(x_0) = 0$. We first restrict ourselves to the subset of functions that also satisfy $\eta(x_1) = 0$. Then according to the Lemma of Dubois-Reymond, (lemma 5.2.2) it follows that $\hat{u}$ satisfies differential equation (5.2.3).

Subsequently we consider the complete set of functions $\eta(x)$. It is clear that natural boundary condition (5.2.5) must be satisfied. □

REMARK:
Differential equations that follow in this way from a minimization problem are known as *Euler-Lagrange equations*.

Lemma 5.2.2 *(Dubois-Reymond)*

Let $M(x) \in C([a,b])$ *and let*

$$\int_a^b M(x)\eta(x)\, dx = 0 \,, \tag{5.2.11}$$

for all $\eta(x) \in C([a,b])$ *with* $\eta(a) = \eta(b) = 0$.

Then

$$M(x) = 0 \quad on \quad [a,b] \,. \tag{5.2.12}$$

Proof
Suppose there is an $x_0 \in (a,b)$ such that $M(x_0) \neq 0$, for example $M(x_0) > 0$. Since $M(x) \in C(a,b)$ there exists a δ-neighborhood of x_0, $(x_0 - \delta, x_0 + \delta) \subset (a,b)$ such that $M(x) > 0$ if $|x - x_0| < \delta$, $(\delta > 0)$.

Now choose $\eta(x)$ as follows

$$\eta(x) = \begin{cases} (x - x_0 - \delta)^2 (x - x_0 + \delta)^2 & \text{if} \quad |x - x_0| < \delta \\ 0 & \text{elsewhere,} \end{cases}$$

then $\int_a^b M(x)\eta(x)\, dx = \int_{x_0-\delta}^{x_0+\delta} M(x)(x - x_0 - \delta)^2 (x - x_0 + \delta)^2\, dx > 0$.

This contradicts (5.2.11) for $x \in (a,b)$.
So from the continuity of $M(x)$ it follows that $M(x) = 0$ for $x \in [a,b]$. □

In the next section we shall extend the Euler Lagrange equations to $\mathbb{R}^2$.

5.3 A simple two-dimensional case

We have seen how the Euler-Lagrange equations are derived in one dimension. Now we shall extend the theory to two dimensions. First we shall consider a simple two-dimensional example. It will be shown that the only difference with $\mathbb{R}^1$ is that the integration by parts is replaced by Gauss' divergence theorem.

Consider a region Ω (Figure 5.1) in $\mathbb{R}^2$. The boundary Γ is subdivided into 2 parts Γ_1 and Γ_2.

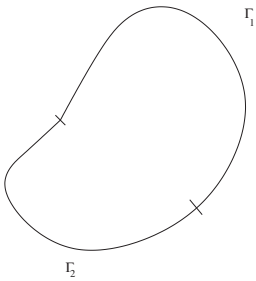


Figure 5.1: Region Ω with 2 boundary parts Γ_1 and Γ_2.

On Ω we consider the following minimization problem:
minimize the integral

$$I(u) = \int_{\Omega} \{\frac{k}{2}|\nabla u|^2 - uf\}\, d\Omega - \int_{\Gamma_2} u\, d\Gamma\,, \tag{5.3.1}$$

over the class of functions satisfying the boundary condition

$$u|_{\Gamma_1} = 0\,, \tag{5.3.2}$$

with $(k : \Omega \to \mathbb{R}^+)$.
With $|\nabla u|^2$ we mean $\nabla u \cdot \nabla u$.
To derive the Euler-Lagrange equations we proceed in exactly the same way as in $\mathbb{R}^1$.

So let $\hat{u}(x, y)$ be the function minimizing (5.3.1), (5.3.2) and consider the set of functions

$$u(\mathbf{x}) = \hat{u}(\mathbf{x}) + \varepsilon\eta(\mathbf{x})\,. \tag{5.3.3}$$

Substitution of (5.3.3) in (5.3.1) gives

$$I(\varepsilon) = \int_{\Omega} \{\frac{k}{2}|\nabla(\hat{u} + \varepsilon\eta)|^2 - (\hat{u} + \varepsilon\eta)f\}\, d\Omega - \int_{\Gamma_2} (\hat{u} + \varepsilon\eta)\, d\Gamma\,. \tag{5.3.4}$$

So the necessary condition for the existence of a minimum of (5.3.4) at $\varepsilon = 0$ is given by:

$$\int_{\Omega} \{k(\nabla\hat{u} \cdot \nabla\eta) - \eta f\}\, d\Omega - \int_{\Gamma_2} \eta\, d\Gamma = 0\,. \tag{5.3.5}$$

Exercise 5.3.1 *Derive formula (5.3.5)* □

In order to apply the two-dimensional version of the Lemma of Dubois-Reymond it is necessary to remove the term $\nabla\eta$.

Instead of classical integration by parts we use Gauss' divergence theorem (1.3.10):

$$\int_{\Omega} \mathrm{div}\, \mathbf{w}\, d\Omega = \oint_{\Gamma} \mathbf{w} \cdot \mathbf{n}\, d\Gamma\,.$$

By substituting $\mathbf{w} = \eta(k\nabla\hat{u})$ we get

$$\int_{\Omega} \nabla\eta \cdot (k\nabla\hat{u})\, d\Omega = -\int_{\Omega} \eta \mathrm{div}\,(k\nabla\hat{u})\, d\Omega + \oint_{\Gamma} \eta k\nabla\hat{u} \cdot \mathbf{n}\, d\Gamma\,. \tag{5.3.6}$$

REMARK: This is in fact the first equation of Green (see Exercise 1.3.8) .

Exercise 5.3.2 *Derive Equation (5.3.6).* □

A combination of (5.3.5) and (5.3.6) results in

$$\int_{\Omega} \{-\text{div}\,(k\nabla\hat{u}) - f\}\eta \; d\Omega + \int_{\Gamma_2} (k\nabla\hat{u}\cdot\mathbf{n} - 1)\eta \; d\Gamma = 0\,. \tag{5.3.7}$$

(Why?)

The two dimensional version of Dubois-Reymond's lemma leads to the PDE

$$-\text{div}\,(k\nabla\hat{u}) = f\,, \tag{5.3.8}$$

with boundary conditions,

$$u|_{\Gamma_1} = 0\,, \quad \text{(essential)} \tag{5.3.9}$$

and

$$k\frac{\partial u}{\partial n}\bigg|_{\Gamma_2} = 1\,. \quad \text{(natural)} \tag{5.3.10}$$

The technique applied here can also be used to solve a more general problem. This is done in Section 5.5. Before doing so we give a number of examples of minimization problems.

Exercise 5.3.3 *Prove Dubois-Reymond's lemma for a bounded region in two dimensions.* □

5.4 Examples of minimization problems

In this section we consider the following examples of minimization problems:

- Minimal surface problem, Section 5.4.1.
- Minimal potential energy, Section 5.4.2. This problem corresponds to a simple Poisson equation and is very suitable to demonstrate numerical techniques.
- Plane stress, Section 5.4.3. This is also a minimum potential energy problem, however, now we have an unknown vector instead of a scalar. As a consequence the corresponding PDE consists of a set of 2 ($\mathbb{R}^2$) or 3 ($\mathbb{R}^3$) coupled PDEs.
- Loaded and clamped plate (normal load), Section 5.4.4.
  Here we have a minimum potential energy problem involving second order derivatives. The corresponding PDE is of order four (Biharmonic equation).

5.4.1 Minimal surface problem

Let $u_c(\mathbf{x})$ be a given closed curve in $\mathbb{R}^3$. Let $c(\mathbf{x})$ be the projection of the curve in the plane ($\mathbb{R}^2$), and define the region Ω as the domain enclosed by c. We assume that u_c has a unique value for every $\mathbf{x} \in c$. The problem is to find the surface in $\mathbb{R}^3$ passing through u_c with minimum area in the domain Ω.

The area of the surface $z = u(x, y)$ is given by

$$s(u) = \int_{\Omega} \sqrt{1 + \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2}\; d\Omega\,. \tag{5.4.1}$$

The problem can be formulated as:
Find u smooth enough, satisfying the boundary conditions:

$$u(c) = u_c \ , \tag{5.4.2}$$

Such that $s(u)$ is minimal. Figure 5.2 shows an example of the solution of such a problem.

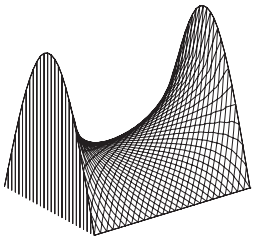


Figure 5.2: Solution of minimal surface problem.

5.4.2 Minimal potential energy

Consider the two rectangular conductors in Figure 5.3.

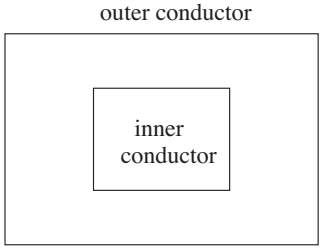


Figure 5.3: Two rectangular conductors.

Let the potential u at the inner conductor be equal to 0, and on the outer conductor be equal to 1. What is the potential u in the region between inner and outer conductor?
Due to symmetry arguments it is sufficient to consider only one quarter of the region, see Figure 5.4.
The principle of minimum potential energy requires that the potential distribution is such that the field energy is minimal.
The energy is given by [32],

$$p(u) = \frac{1}{2} \int_{\Omega} |\nabla u|^2 \, d\Omega \ . \tag{5.4.3}$$

The mathematical formulation of this problem is:

> Minimize the integral $p(u)$ over the class of sufficiently smooth functions with boundary conditions
>
> $$\begin{array}{ll} u = 0 & \text{on } \Gamma_1 \ , \\ u = 1 & \text{on } \Gamma_2 \ . \end{array} \tag{5.4.4}$$
>
> (see Figure 5.4).

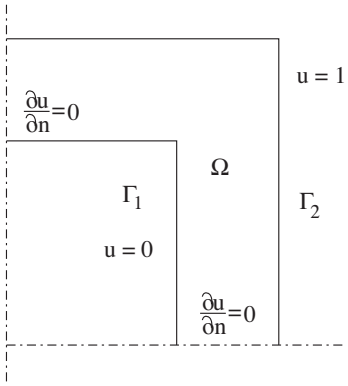


Figure 5.4: One quarter of the potential problem.

5.4.3 Small displacement theory of elasticity (Plane stress)

Consider the flat thin plate of Figure 5.5. See for example [31]. We assume that the thickness of the plate is small compared to its diameter. The outer load is uniform over the cross-section. The load is applied in the same plane as the plate. Along Γ_1 the plate is clamped.

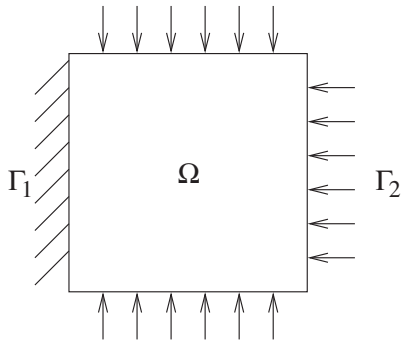


Figure 5.5: Flat plate clamped in Γ_1 and with uniform load on Γ_2.

Unknowns that we want to determine in this problem are the displacement vector $\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}$ and the stress tensor $\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix}$. These are not independent.

The potential energy of the plate is defined as:

$$P(\mathbf{u}) = \frac{1}{2}\int_{\Omega} (\sigma_{xx}\varepsilon_x + \sigma_{yy}\varepsilon_y + \gamma_{xy}\tau_{xy})\, d\Omega - \int_{\Gamma_2} (t_1 u + t_2 v)\, d\Gamma\,, \tag{5.4.5}$$

where $\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_x & \gamma_{xy} \\ \gamma_{xy} & \varepsilon_y \end{bmatrix}$ denotes the strain tensor and

$\mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$ the external load vector.

We suppose that are no body forces in this case.

In order to get expression (5.4.5) in one type of unknown we need the *strain-displacement relations*:

$$\varepsilon_x = \frac{\partial u}{\partial x}\,, \quad \varepsilon_y = \frac{\partial v}{\partial y}\,, \quad \gamma_{xy} = [\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}]\,. \tag{5.4.6}$$

We also need a *constitutive equation* which relates stress to strain.
If we assume that the material is elastic, i.e. satisfies Hooke's Law then we get the following relations:

$$\begin{aligned} \sigma_{xx} &= \frac{E}{1-\nu^2}\,(\varepsilon_x + \nu\varepsilon_y), \\ \sigma_{yy} &= \frac{E}{1-\nu^2}\,(\nu\varepsilon_x + \varepsilon_y), \\ \tau_{xy} &= \frac{E}{1-\nu^2}\,\frac{1-\nu}{2}\,\gamma_{xy}, \end{aligned} \tag{5.4.7}$$

with E the elasticity modulus and ν Poisson's ratio.
With these relations we can express the potential energy in the displacements only.

Exercise 5.4.1 *Show that the potential energy (5.4.5) with the relations (5.4.6) and (5.4.7) can be written as:*

$$\begin{aligned} P(\mathbf{u}) &= \frac{1}{2}\int_\Omega \{A\frac{\partial u}{\partial x}(\frac{\partial u}{\partial x} + \nu\frac{\partial v}{\partial y}) + B(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) \\ &\quad + A\frac{\partial v}{\partial y}(\nu\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y})\}\, d\Omega - \int_{\Gamma_2} (t_1 u + t_2 v)\, d\Gamma\,, \end{aligned} \tag{5.4.8}$$

with $A = \frac{E}{(1-\nu^2)}$ *and* $B = \frac{E}{2(1+\nu)}$ *(E and ν constant).* □

The mathematical formulation of this problem is:

Find u, v with

$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_1 \tag{5.4.9}$$

such that the integral $P(u)$ in (5.4.8) is minimal.

5.4.4 Loaded and clamped plate

Consider the small plate of Figure 5.6.

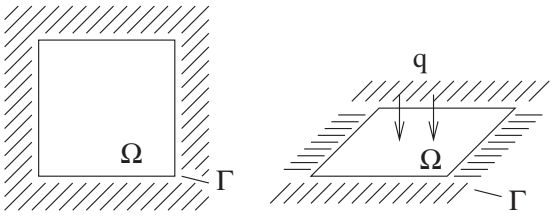


Figure 5.6: Clamped plate with normal load q, domain Ω and boundary Γ.

The load q is normal to the plate and the potential energy is given by

$$I(u) = \int_\Omega \frac{1}{2}[w_{xx}^2 + 2w_{xx}w_{yy} + w_{yy}^2 - 2qw]\, d\Omega\,, \tag{5.4.10}$$

where w is the displacement of the neutral face. We wish to determine w for a given load q. The mathematical formulation of this problem is:

Find w satisfying the boundary conditions:

$$\begin{array}{ll} w = 0 & \text{on } \Gamma\,, \\ \frac{\partial w}{\partial n} = 0 & \text{on } \Gamma\,. \end{array} \tag{5.4.11}$$

such that the integral $I(u)$ in (5.4.10) is minimal.

5.5 A two-dimensional problem

Theorem 5.2.1 can be generalized to two dimensions.

Theorem 5.5.1 *Let Ω be a domain in $\mathbb{R}^2$ with boundary Γ. Let Γ be subdivided into three parts Γ_1, Γ_2 and Γ_3:*

$$\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3\,.$$

The class of functions in which we try to find a solution is given by

$$\Sigma = \{u |\ u(\mathbf{x}) = g(\mathbf{x}),\ \ \forall \mathbf{x} \in \Gamma_1\}\,.$$

Let $F(x, y, u, p, q)$ and $f(x, y, u)$ be sufficiently smooth functions. Consider the following minimization problem

$$\min_{u\in\Sigma} J[u] = \min_{u\in\Sigma} \int_\Omega F(x, y, u, u_x, u_y)\, d\Omega + \int_{\Gamma_2} f(x, y, u)\, d\Gamma\,. \tag{5.5.1}$$

If there exists a solution $\hat{u}$ of this problem then $\hat{u}$ satisfies the PDE

$$\frac{\partial F}{\partial u} - \frac{\partial}{\partial x}\frac{\partial F}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial F}{\partial u_y} = 0\,, \tag{5.5.2}$$

with boundary conditions

$$u = g\,, \quad \forall \mathbf{x} \in \Gamma_1 \tag{5.5.3}$$

$$\frac{\partial F}{\partial u_x} n_1 + \frac{\partial F}{\partial u_y} n_2 + \frac{\partial f}{\partial u} = 0\,, \quad \forall \mathbf{x} \in \Gamma_2 \tag{5.5.4}$$

$$\frac{\partial F}{\partial u_x} n_1 + \frac{\partial F}{\partial u_y} n_2 = 0\,, \quad \forall \mathbf{x} \in \Gamma_3\,. \tag{5.5.5}$$

where $\mathbf{n} = \begin{pmatrix} n_1 \\ n_2 \end{pmatrix}$ is the outward normal vector at the boundary.

Exercise 5.5.1 *Prove theorem 5.5.1 using the technique of Section 5.3.* □

5.6 Theoretical remarks

5.6.1 Smoothness requirements

In our derivations of the PDEs we have assumed that the solution is sufficiently smooth. This means that the solution must be so smooth that the differential equation exists. Hence if the PDE is of second order, it is necessary that the solution is in $C^2(\Omega)$. However in the original minimization problem we have only first order derivatives. Hence for the existence of a solution of the minimization problem it is sufficient that the first derivatives exist, or more precisely that the integral

makes sense. Usually this is translated by requiring that both the unknown u as its derivatives u_x and u_y are square integrable, i.e.

$$\int_\Omega u^2 \, d\Omega \,, \quad \int_\Omega u_x^2 \, d\Omega \quad \text{and} \quad \int_\Omega u_y^2 \, d\Omega$$

must exist and be finite.
In general this is even weaker than requiring that the derivatives exist everywhere in Ω.

If the solution of the minimization problem is not twice differentiable, it cannot satisfy the PDE. So actually a minimization problem may have a solution in a larger class of functions than the corresponding PDE. In fact the minimization problem can be seen as a generalization of that PDE.

5.6.2 Boundary conditions

We have seen that we must distinguish between essential and natural boundary conditions. Essential boundary conditions are conditions that have to be satisfied by all functions in the function class where we seek the solution. Natural boundary conditions appear naturally from the minimization problem once we derive the corresponding Euler-Lagrange equations.

In general we can state the following:

> If a minimization problem contains derivatives of first order and not higher, the corresponding Euler-Lagrange equation will be of second order. For such a problem essential boundary conditions have always the form $u = g_0, \mathbf{x} \in \Gamma_0$. If a boundary condition contains first derivatives for this type of problems, it is always a natural boundary condition. See [37].

> If a minimization problem contains derivatives of second order and not higher, the corresponding PDE will be of fourth order. For such problem boundary conditions involving only u or first order derivatives of u are essential, boundary conditions involving second or third derivatives will be natural.

5.6.3 Weak formulation

Consider minimization problem (5.3.1) without the integral over Γ_2 and with Γ_1 equal to the whole boundary Γ, hence

$$\min_{u \in \Sigma} \; I(u) = \int_\Omega \{ \frac{k}{2} |\nabla u|^2 - uf \} \, d\Omega \,, \tag{5.6.1}$$

$$u|_\Gamma = 0 \,. \tag{5.6.2}$$

According to (5.3.7), the solution must satisfy

$$\int_\Omega \{ -\text{div}\,(k \nabla u) - f \} \eta \, d\Omega = 0 \,, \tag{5.6.3}$$

for all η in Σ.

This is precisely the differential equation multiplied by η and integrated over the domain. In Chapter 7 we shall use such a method to arrive at the *weak formulation*. In that case η is called a test function.

5.7 Exercises

Exercise 5.7.1 *Find the Euler-Lagrange equation for the minimal surface problem (5.4.1), with boundary conditions (5.4.2). Do not use theorem 5.5.1.* □

Exercise 5.7.2 *Find the Euler-Lagrange equations for the minimization problem in Section 5.4.2 by direct variation around the solution. Which boundary conditions are essential and which are natural?* □

Exercise 5.7.3 *Find the Euler-Lagrange equations for the minimization problem of Exercise 5.4.1 in Section 5.4.3. (assume $u = \hat{u} + \varepsilon\eta, v = \hat{v} + \varepsilon\xi$).*

Use the strain-displacement relations (5.4.6) and stress-strain relations (5.4.7) to rewrite the Euler-Lagrange equations in the form

$$-\frac{\partial \sigma_{xx}}{\partial x} - \frac{\partial \tau_{xy}}{\partial y} = 0\,,$$
$$-\frac{\partial \tau_{xy}}{\partial x} - \frac{\partial \sigma_{yy}}{\partial y} = 0\,.$$

□

Exercise 5.7.4 *Find the Euler-Lagrange equations for the minimization problem in Section 5.4.4.* □

Exercise 5.7.5 *Find the Euler-Lagrange equations for the rotation surface with minimal area defined by*

$$\min J[u] = 2\pi \int_{x_0}^{x_1} u\sqrt{1 + \left(\frac{du}{dx}\right)^2}\, dx\,,$$
$$u(x_0) = y_0\,.$$

□

Exercise 5.7.6 *Consider the region Ω of Figure 5.7.*

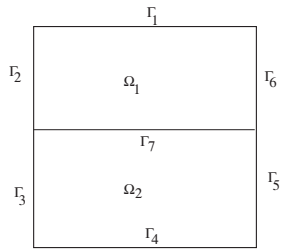


Figure 5.7: Region consisting of 2 layers.

In this region we have two layers Ω_1 and Ω_2 with different values of the permeability κ (κ_1 and κ_2). The pressure p in this layer satisfies the minimization problem

$$\min_{p\in\Sigma} \int_{\Omega} \frac{1}{2}\kappa |\nabla p|^2\, d\Omega, \tag{5.7.1}$$

subject to the essential boundary condition $p|_{\Gamma_1} = g(x)$.
Find the Euler-Lagrange equations for this problem. What are the natural boundary conditions on Γ_3. Derive also the interface conditions on Γ_7.
Hint: Split the integral into two parts. □

5.8 From PDE to minimization problem

5.8.1 Introduction

We have seen in Section 5.3 that if the solution of a minimization problem is smooth, it satisfies a partial differential equation. Furthermore, on those parts of the boundary where no essential boundary condition has been prescribed, natural boundary conditions result from the minimization problem. Before trying to solve these minimization problems numerically we ask ourselves the question: is it always possible to find a minimization problem corresponding to a (partial) differential equation? The answer to this question is no. Under certain conditions only, one can find an equivalent minimization problem. The key property will be symmetry, which will be defined later on in this section. Nevertheless a large class of important PDEs satisfies the requirements necessary to derive an equivalent minimization problem. In Chapter 7 we shall generalize the theory in such a way that the numerical techniques of Chapter 6 can be applied even for cases where no minimization problem can be found. For simplicity we shall restrict ourselves to linear problems only. However, we have seen in 5.1 that also non-linear PDEs may correspond to minimization problems.

In first instance we consider only homogeneous boundary conditions. The general case will be treated later.

5.8.2 Linear problems with homogeneous boundary conditions

Consider the linear PDE (5.3.8)-(5.3.10), but with homogeneous boundary conditions

$$-\operatorname{div} k\nabla u = f\,, \tag{5.8.1}$$

$$u|_{\Gamma_1} = 0\,, \tag{5.8.2}$$

$$k\frac{\partial u}{\partial n}\bigg|_{\Gamma_2} = 0\,. \tag{5.8.3}$$

The solution of (5.8.1)-(5.8.3) must be found in the vector space Σ:

$$\Sigma = \{u \text{ smooth} \mid \; u|_{\Gamma_1} = 0\,; \;\; \frac{\partial u}{\partial n}\bigg|_{\Gamma_2} = 0\}\,.$$

In general we shall write a linear PDE like (5.8.1) in the form

$$Lu = f\,. \tag{5.8.4}$$

Hence in (5.8.1) we have $Lu \equiv -\operatorname{div}(k\nabla u)$.
It can be shown that a minimization problem for (5.8.4) can be found if L satisfies the two following properties:

$$\text{symmetry (self adjointness)} \quad \int_\Omega uLv\,d\Omega = \int_\Omega vLu\,d\Omega\,, \quad \forall u, v \in \Sigma\,. \tag{5.8.5}$$

$$\text{positiveness} \quad \int_\Omega uLu\,d\Omega \geq 0\,,\ \forall u \in \Sigma\,. \tag{5.8.6}$$

In practice it turns out that the two properties are also necessary for the existence of a corresponding minimization problem.
Differential operators satisfying properties (5.8.5) and (5.8.6) are called *strongly elliptic*.

Before constructing a minimization problem we shall check if properties (5.8.5), (5.8.6) are satisfied by problem (5.8.1)-(5.8.3). To this end we multiply (5.8.1) by $v \in \Sigma$, integrate over Ω and apply the divergence theorem twice:

$$\int_\Omega -v(\operatorname{div} k\nabla u)\, d\Omega = \int_\Omega k\nabla u \cdot \nabla v\, d\Omega - \oint_\Gamma vk\frac{\partial u}{\partial n}\, d\Gamma\,. \tag{5.8.7}$$

Due to the boundary conditions the boundary integral vanishes.

$$\int_\Omega k\nabla u \cdot \nabla v\, d\Omega = -\int_\Omega u(\operatorname{div} k\nabla v)\, d\Omega + \oint_\Gamma uk\frac{\partial v}{\partial n}\, d\Gamma\,. \tag{5.8.8}$$

Again the boundary integral vanishes.
In fact (5.8.7)-(5.8.8) already demonstrate symmetry.

To prove positivity it is sufficient to substitute u for v in (5.8.7):

$$\int_\Omega -u(\operatorname{div} k\nabla u)\, d\Omega = \int_\Omega k\nabla u \cdot \nabla u\, d\Omega \geq 0\,, \text{ for } u \in \Sigma\,. \tag{5.8.9}$$

With properties (5.8.5) and (5.8.6) it is easy to prove the following theorem:

Theorem 5.8.1 *Let L be a linear, symmetric, positive differential operator defined over a space Σ and let*

$$Lu = f\,. \tag{5.8.10}$$

Then the solution u minimizes the functional

$$I(u) = \int_\Omega \{\frac{1}{2}uLu - uf\}\, d\Omega\,, \text{ over the space } \Sigma\,. \tag{5.8.11}$$

On the other hand if u minimizes (5.8.11) then u satisfies (5.8.10).

Proof
First suppose that u_0 is the solution of (5.8.10), hence $Lu_0 = f$.
Substituting this in (5.8.11) gives (using the symmetry of L)

$$\begin{aligned} I(u) &= \int_\Omega \{\frac{1}{2}uLu - uLu_0\}\, d\Omega = \\ &\int_\Omega \{\frac{1}{2}(u-u_0)L(u-u_0) - \frac{1}{2}u_0Lu_0\}\, d\Omega\,. \end{aligned} \tag{5.8.12}$$

Since $\int_u \frac{1}{2}u_0Lu_0\, d\Omega$ is fixed and L is positive we know that the minimum is reached if

$$\int_\Omega \frac{1}{2}(u-u_0)L(u-u_0)\, d\Omega = 0\,.$$

Hence u_0 minimizes (5.8.11). □

Exercise 5.8.1 *Show that the minimum of $I(u)$ over Σ satisfies (5.8.10). Use the standard Euler-Lagrange approach and symmetry of L.* □

If we apply this theorem to example (5.8.1)-(5.8.3), it immediately follows that the corresponding functional $I(u)$ is given by

$$\begin{aligned} I(u) &= \int_\Omega \{\frac{1}{2}u(-\text{div}\, k\nabla u) - uf\}\, d\Omega \\ &= \int_\Omega \{\frac{1}{2}k|\nabla u|^2 - uf\}\, d\Omega\,, \end{aligned}$$

and this is the same as (5.3.1) except for the boundary integral. Actually with respect to the minimization problem it is not necessary to satisfy the natural boundary condition and it is sufficient to consider the space

$$\Sigma = \{u \text{ smooth} \mid \ u|_{\Gamma_1} = 0\}\,.$$

REMARK:

The proof of this theorem is based upon the symmetry and the positivity of the differential operator. These properties are sufficient. In practice these properties are also necessary. As a consequence no equivalent minimization problem for the convection-diffusion equation equation can be found.

In this section we have restricted ourselves to homogeneous boundary conditions, because they are necessary for the symmetry property (5.8.5). Otherwise the boundary integral in (5.8.7) would not vanish. It is only a small extension to consider also non-homogeneous boundary conditions, as will be demonstrated in Section 5.8.3.

Exercise 5.8.2 *Show that the operator in the convection-diffusion equation*

$$-\text{div}\,(k\nabla c) + \mathbf{u}\cdot\nabla c = f$$

is not symmetric. □

5.8.3 Linear problems with non-homogeneous boundary conditions

Theorem 5.8.1 relates a PDE with an equivalent minimization problem. However, this theorem is only applicable for homogeneous boundary conditions. In case of non-homogeneous boundary conditions we have to adapt the theorem or make the boundary conditions homogeneous. The last solution is the most simple one.

Theorem 5.8.2 *Let $Lu = f$ with non-homogeneous boundary conditions. Suppose that there is a smooth function w satisfying the non-homogeneous boundary conditions. If this function does not exist, the original problem has no solution.*
Then u satisfies the minimization problem

$$I(u) = \frac{1}{2}\int_\Omega (u-w)(Lu+Lw)\, d\Omega - \int_\Omega fu\, d\Omega\,. \tag{5.8.13}$$

Proof
Consider

$$v = u - w. \tag{5.8.14}$$

Clearly v satisfies homogeneous boundary conditions and since

$$Lv = Lu - Lw = f - Lw, \tag{5.8.15}$$

Theorem 5.8.1 can be applied for v with right-hand side $f - Lw$.

So the corresponding minimization problem is:

$$\min_{v\in\Sigma} I(v) = \frac{1}{2}\int_\Omega vLv\, d\Omega - \int_\Omega v(f - Lw)\, d\Omega\,, \tag{5.8.16}$$

with Σ provided with homogeneous boundary conditions.
Substituting (5.8.14) in (5.8.16) it is easy to see that

$$\tilde{I}(u) = \frac{1}{2}\int_\Omega (u - w)(Lu + Lw)\, d\Omega - \int_\Omega fu\, d\Omega + \int_\Omega fw\, d\Omega\,. \tag{5.8.17}$$

Since w and f are given functions independent of the solution, the minimum of (5.8.17) does not change if we skip the last term and we end up with (5.8.13). □
Mark that in this case we do not have $\int_\Omega uLw\, d\Omega = \int_\Omega wLu\, d\Omega$ (why?). As a consequence Equation (5.8.13) can not be simplified furthermore.
It is of course necessary to remove w from this expression, since w is unknown. This will be done in the following example.

Theorem 5.8.3 *Consider a region Ω with boundary Γ. Γ consists of 3 parts Γ_1, Γ_2 and Γ_3 such that*

$$\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3\,.$$

Consider the differential equation

$$-\mathrm{div}\, k\nabla u = f \quad (k > 0) \text{ in } \Omega\,, \tag{5.8.18}$$

with boundary conditions

$$u = g_1 \quad on \quad \Gamma_1\,, \tag{5.8.19}$$

$$k\frac{\partial u}{\partial n} = g_2 \quad on \quad \Gamma_2\,, \tag{5.8.20}$$

$$cu + k\frac{\partial u}{\partial n} = g_3 \quad on \quad \Gamma_3 \quad (c > 0)\,. \tag{5.8.21}$$

Then u satisfies the minimization problem

$$\min_{u\in\Sigma} \int_\Omega \{\frac{1}{2}k|\nabla u|^2 - uf\}\, d\Omega - \int_{\Gamma_2} g_2 u\, d\Gamma + \int_{\Gamma_3} \{\frac{1}{2}cu^2 - g_3 u\}\, d\Gamma\,, \tag{5.8.22}$$

with $\Sigma : \{u \mid u|_{\Gamma_1} = g_1\}$.

Proof
The function w satisfies (5.8.19) to (5.8.21).
Substitution of these terms in (5.8.13) gives

$$\min_u I(u) = \frac{1}{2}\int_\Omega (u - w)(-\mathrm{div}\,(k\nabla(u + w))\, d\Omega - \int_\Omega fu\, d\Omega\,.$$

Gauss theorem applied to the first terms gives

$$\begin{aligned} I(u) \;=\; & \frac{1}{2}\int_\Omega \nabla(u - w)\cdot k\nabla(u + w)\, d\Omega - \int_\Omega fu\, d\Omega \\ & -\frac{1}{2}\int_\Gamma (u - w)k\nabla(u + w)\cdot \mathbf{n}\, d\Gamma\,. \end{aligned} \tag{5.8.23}$$

The first term can be written as

$$\frac{1}{2}\int_{\Omega} \nabla(u-w)\cdot k\nabla(u+w)\,d\Omega = \frac{1}{2}\int_{\Omega} \nabla u\cdot k\nabla u\,d\Omega - \frac{1}{2}\int_{\Omega} \nabla w\cdot k\nabla w\,d\Omega\,. \tag{5.8.24}$$

Since the last term does not depend on u it can be removed from the minimization problem, without effect on u.
The last term of (5.8.23) is split over the 3 boundaries Γ_1, Γ_2 and Γ_3.
On Γ_1 this term is equal to 0 because of $u - w = 0$.
On Γ_2 it can be written as:

$$-\frac{1}{2}\int_{\Gamma_2}\{uk\frac{\partial(u+w)}{\partial n} - wk\frac{\partial(u+w)}{\partial n}\}\,d\Gamma = -\int_{\Gamma_2}\{ug_2 - wg_2\}\,d\Gamma$$

and again the last term can be skipped from the minimization problem since it does not depend on u.
On Γ_3 as

$$-\int_{\Gamma_3}\{ug_3 - wg_3 - \frac{1}{2}cu^2 + \frac{1}{2}cw^2\}\,d\Gamma\,,$$

and now the second and fourth term can be removed (why ?).
So at last we arrive at the minimization problem (5.8.22).

So the Dirichlet boundary condition (5.8.19) is an essential boundary condition. □

5.8.4 Exercises

Exercise 5.8.3 *Find the equivalent minimization problem of the three-dimensional Poisson equation*

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = f(x,y,z) \quad in\ \Omega\,,$$

with boundary condition

$$u = g \quad on\ \Gamma\,.$$

□

Exercise 5.8.4 *Find the minimization problem corresponding to the differential equation*

$$-\frac{d}{dx}(p(x)\frac{du}{dx}) = f$$

with boundary condition

$$u(0) = 1\,, \quad u(1) + \frac{du}{dx}(1) = a\,.$$

□

Exercise 5.8.5 *Find the minimization problem corresponding to the system of s ordinary differential equations*

$$\sum_{k=1}^{s}\left[-\frac{d}{dx}(p_{jk}(x)\frac{du_k}{dx}) + q_{jk}(x)u_k\right] = f_j(x) \qquad j = 1,2,...,s\,,$$
$$a < x < b\,,$$

with boundary conditions $u_j(a) = u_j(b) = 0$ $(j = 1, 2, ..., s)$ and P a symmetric positive definite matrix with elements $p_{jk}(x)$ and Q a symmetric positive semi-definite matrix with elements $q_{jk}(x)$.
Use theorem 5.8.1 with u a vector instead of a scalar. □

Exercise 5.8.6 *Find the minimization problem corresponding to the differential equation*

$$\frac{d^4u}{dx^4} = f \,,$$

with boundary conditions

$$u(0) = \frac{du}{dx}(0) = 0 \,, \quad \frac{du}{dx}(1) = 0 \,, \quad \frac{d^3u}{dx^3}(1) = 1 \,.$$

□

5.9 Mathematical theory of minimization

Section 5.8 shows that linear PDEs which satisfy some extra properties like symmetry and positivity, are equivalent to a minimization problem. In the proof of Theorem 5.8.1 we needed a function space Σ satisfying some smoothness requirements. In this section we shall consider this theory from a more fundamental (mathematical) point of view. However, this will not be a complete and thorough mathematical treatment of the problem, since that is beyond the scope of this book.

Let us first introduce some notations.

In Section 5.8 we have used expressions like $\int_\Omega uf \, d\Omega$. So it is naturally to use the L^2 inner product

$$(u, v) = \int_\Omega uv \, d\Omega \,, \tag{5.9.1}$$

which is defined for all functions $u, v \in L^2(\Omega)$.

Besides that we have introduced the integral

$$\int_\Omega uLv \, d\Omega \tag{5.9.2}$$

in (5.8.5). For this integral we have required some properties like symmetry (5.8.5) and positivity (5.8.6).

Definition 5.9.1
The operator L is called positive definite if there exist a constant $\gamma > 0$, such that

$$\int_\Omega uLu \, d\Omega \geq \gamma \int_\Omega u^2 \, d\Omega \,, \quad \forall u \in \Sigma \,. \tag{5.9.3}$$

If (5.8.5), (5.8.6) and (5.9.3) are satisfied we can define a new inner product, the *energy product*, by:

$$(u, v)_L = \int_\Omega uLv \, d\Omega \tag{5.9.4}$$

and corresponding (energy) norm $\|u\|_L^2 = (u, u)_L$.

Exercise 5.9.1 *Prove that (5.9.4) satisfies all the requirements of an inner product.* □

It is necessary that the definition space Σ is such that the integral in (5.9.4) makes sense (i.e. is finite) and besides that, the space must be a vector space. This means that elements in Σ must satisfy the following linearity property

$$\begin{aligned} &\text{if} \quad u,\ v \in \Sigma \quad \text{then also} \\ &\alpha u + \beta v \in \Sigma \quad \text{with} \quad \alpha, \beta \in \mathbb{R}^1 \,. \end{aligned}$$

The space Σ is a space with smoothness requirements for its elements, but also each function in Σ must satisfy essential boundary conditions.

Exercise 5.9.2 *Show that Σ can be a vector space only if homogeneous boundary conditions are satisfied.* □

Now we can formulate Theorem 5.8.1 in a more mathematical way:

Let L be a linear operator defined on a Hilbert space Σ satisfying

$$(Lu, v) = (v, Lu) \quad \forall u, v \in \Sigma \text{ i.e. } L \text{ is self-adjoint .} \tag{5.9.5}$$

$$(u, Lu) \geq 0 \quad \forall u \in \Sigma \quad \text{i.e } L \text{ is positive .} \tag{5.9.6}$$

$$(u, Lu) \geq \gamma(u, u) \quad \forall u \in \Sigma \quad \text{i.e } L \text{ is positive definite .} \tag{5.9.7}$$

In fact (5.9.7) implies (5.9.6).
(u, u) is the L^2 inner product and (Lu, v) is the inner product in Σ.

Then the solution of

$$Lu = f \,, \quad u \in \Sigma \,, \quad f \in L^2(\Omega) \,, \tag{5.9.8}$$

minimizes the functional with $J(u)$

$$\min_{u \in \Sigma} J(u) = \frac{1}{2}(u, Lu) - (u, f) \,, \tag{5.9.9}$$

and the minimum of (5.9.9) satisfies (5.9.8).

Exercise 5.9.3 *Prove this theorem in the same way as in Section 5.8.* □

REMARK:
The property that L must be positive definite is not necessary in the theorem. It is only important in order to define an inner product. Also it enables us to prove uniqueness of the solution.

Theorem 5.9.1 *There is exactly one $u \in \Sigma$ that minimizes the functional $J(u)$ defined in (5.9.9).*

Proof

$$(u, f) \leq \|u\| \, \|f\| \,, \tag{5.9.10}$$

where $\|u\|$ is the L^2-norm. ($\|u\| = (u, u)^{1/2}$).

From (5.9.3) it follows that

$$\|u\|^2 \leq \frac{1}{\gamma} \|u\|_L^2 \,. \tag{5.9.11}$$

(5.9.10) together with (5.9.11) gives

$$(u, f) \leq \frac{1}{\sqrt{\gamma}} \|u\|_L \|f\| \,. \tag{5.9.12}$$

Now we apply Riesz' representation theorem (see [22]), This theorem states:

for every bounded linear functional $\ell(u)$ defined on a Hilbert space H there is exactly one element $u_0 \in H$ such that

$$\ell(u) = (u, u_0)_H \quad \forall u \in H\,,$$

with $(u, v)_H$ the inner product in H.

Hence there is exactly one $u_0 \in \Sigma$ such that

$$(u, f) = (u, u_0)_L \quad \forall u \in \Sigma\,. \tag{5.9.13}$$

Now consider the minimization problem:

$$\begin{aligned} J[u] &= \frac{1}{2}\|u\|_L^2 - (u, f) \\ &= \frac{1}{2}\|u\|_L^2 - (u, u_0)_L \\ &= \frac{1}{2}(u - u_0, u - u_0)_L - \frac{1}{2}(u_0, u_0)_L\,. \end{aligned} \tag{5.9.14}$$

Since $(v, v)_L > 0\ \forall v \in \Sigma$, $J[u]$ takes its minimum for $u = u_0$. Since u_0 is unique (from the Riesz' representation theorem) we have proven the theorem. □

So the minimization problem has always a unique solution in the Hilbert space Σ. However, this solution does not have to be the solution of the original PDE. The reason is that for second order PDEs, the inner product (u, Lv) only contains first order derivatives.
For elements in Σ it is sufficient that (u, Lu) is finite, hence the first derivative must be in $L^2(\Omega)$. For a second order PDE it is necessary that the second derivatives exist. So we may have a solution of the minimization problem in Σ that does not satisfy the PDE in classical sense.

According to Theorem 5.8.1 a solution of the minimization problem satisfies

$$Lu = f \quad u \in \Sigma\,. \tag{5.9.15}$$

But if u is not smooth we cannot consider this as a classical solution. So a solution of (5.9.9) is called a 'generalized' or 'weak' solution of the PDE. If the solution is also sufficiently smooth, it is called a 'strong' solution. In fact we have proved that there is always a weak solution. To prove that there is a strong solution we need extra smoothness requirements for both f and the boundary of Ω. Such a proof is general not simple. See for example [12]. But if a strong solution exists, it is equal to the weak solution. (Why?).

The Hilbert spaces introduced in this Chapter are usually *Sobolev spaces* denoted as $H^k(\Omega)$, where k refers to the highest order derivatives in the inner product. For example the space $H^1(\Omega)$ is defined in Theorem (1.6.1). More theory about Sobolev spaces can for example be found in [1].

5.10 Summary of Chapter 5

A number of physical problems can be formulated in the following form:

find a function $u(\mathbf{x})$ in a class of functions such that an integral of a function of $u(\mathbf{x})$ and some of its derivatives is minimal.

It has been demonstrated by variation, that the solution of the minimization problems satisfies a PDE, the *Euler-Lagrange equation*.

Some boundary conditions can be prescribed on the solution class of the minimization problems; these are called essential boundary conditions. Others arise naturally when the Euler-Lagrange equations are derived from the minimization problem. These boundary conditions are called natural. They always involve first order derivatives for second order equations and second and third order derivatives for fourth order equations.

On the other hand it has been shown that under certain conditions (symmetry and positiveness) an equivalent minimization problem can be derived from a PDE. The minimization problem has always lower order derivatives than the PDE. For example if the minimization problem contains first order derivatives the PDE is of second order and in case the minimization problem contains second order derivatives, the PDE is of order four. As a consequence the smoothness requirements for the solution of the PDE are more restrictive than those of the minimization problem.

If the PDE and minimization problem are equivalent, solution of one of the two automatically solves the other.

Chapter 6

The numerical solution of minimization problems

Objectives

Chapter 5 showed the equivalence between a certain class of PDEs and minimization problems. As a consequence solving the PDE also solves the minimization problem and vice versa. Chapters 3, 4 were devoted to solving the PDE directly by finite differences or, after integration over a volume, by finite volumes. In this chapter we shall solve the corresponding minimization problem numerically. Hence the PDE is solved in an indirect way.
The numerical technique that will be applied is the classical *Ritz's method* based on expressing the solution as a linear combination of previously chosen functions: the basis functions. These are in general not related to the problem, but chosen beforehand. This method itself is not very practical, but combined with a clever choice of basis functions we arrive at the finite element method (FEM). The FEM is well suited for unstructured grids and has a strict local character. All information in one element is used, without considering neighbors. This makes the method very attractive for computer implementation. For certain types of PDEs, for example those arising from elasticity and plasticity problems, the FEM is the most popular method at this moment.
Another way of looking at the FEM, is to consider it as an automatic tool to derive finite difference formula for unstructured grids. An important advantage of the FEM is that the treatment of boundary conditions is almost always very natural and therefore simpler than in classical difference methods.

6.1 Ritz's method

6.1.1 Introduction

Suppose we want to solve the general minimization problem

$$\min_u J[u] \; ; \quad J[u] = \int_\Omega F(x, y, u, u_x, u_y) \; d\Omega \; , \tag{6.1.1}$$

where the minimum must be found over a class of functions in the target space Σ:

$$\Sigma = \{u \text{ sufficiently smooth}; \;\; u|_\Gamma = g\} \; . \tag{6.1.2}$$

Chapter 5 already demonstrated that the solution of this problem is not simple. Actually we transformed it to a minimization problem with one unknown (ε), thus deriving the Euler-Lagrange equations.
Direct minimization of (6.1.1), (6.1.2) is in general only possible if we have a finite number of unknowns.

This can be achieved by approximating the solution by a linear combination of a finite fixed set of functions $\varphi_i(\mathbf{x})$:

$$u^n(\mathbf{x}) = \sum_{j=1}^{n} a_j \varphi_j(\mathbf{x}) \,. \tag{6.1.3}$$

In the remainder of this section we assume homogeneous essential boundary conditions, i.e. $g = 0$.

The functions $\varphi_i(\mathbf{x})$ (the *basis functions*), must be chosen such that:

$$\varphi_i(\mathbf{x}) \in \Sigma \quad \text{for all } i \,.$$

This means that $\varphi_i(\mathbf{x})$ must be sufficiently smooth, such that (6.1.1) makes sense, and also that $\varphi_i(\mathbf{x})$ must satisfy the homogeneous boundary conditions. So in fact the functions $\varphi_i(\mathbf{x})$ span a subspace of Σ. Moreover, the functions $\varphi_i(x)$ should preferably be linearly independent (why?). Now Ritz's method consists of solving the minimization problem over this subspace.

Since only the a_i in (6.1.3) are unknown, this means that the problem reduces to minimizing over the set $a_1 ... a_n$:

$$\min_{a_i \in \mathbb{R}^n} J[a_1, a_2, ..., a_n] \,. \tag{6.1.4}$$

The necessary condition for the existence of a minimum is

$$\frac{\partial J[u^n]}{\partial a_i} = 0 \,, \quad i = 1, 2, ..., n \,. \tag{6.1.5}$$

(6.1.5) forms a set of n equations with n unknowns, which under certain conditions, can be solved uniquely. This produces a solution $u^n(\mathbf{x})$. By increasing the number of basis functions we hope that $u^n(\mathbf{x})$ converges to the solution $u(\mathbf{x})$ of (6.1.1) and (6.1.2). It is clear that the choice of the basis functions $\varphi_i(\mathbf{x})$ is essential for the convergence and especially for the speed of convergence of Ritz's method.

Let us first consider a simple one-dimensional example to show how Ritz's method behaves in practice.

6.1.2 A simple one-dimensional example

Theorem 6.1.1 *Let u satisfy the following minimization problem (cf. Section 5.1.1)*

$$\min_{u \in \Sigma} J[u] = \int_0^1 \{ \frac{1}{2} \left(\frac{du}{dx} \right)^2 - f(x) u(x) \} \, dx \,, \tag{6.1.6}$$

$$\Sigma : \; \{ u \mid u \text{ sufficiently smooth; } u(0) = 0 \} \,.$$

and let $u^n(\mathbf{x})$ be defined by (6.1.3)*, then the set of Ritz equations is given by*

$$\sum_{j=1}^{n} a_j \int_0^1 \frac{d\varphi_i}{dx} \frac{d\varphi_j}{dx} \, dx = \int_0^1 f(x) \varphi_i(x) \, dx, \quad i = 1, 2, ..., n. \tag{6.1.7}$$

Proof
Substitution of (6.1.5) in (6.1.6), (6.1.5) gives

$$\frac{\partial}{\partial a_i}\int_0^1 \{\frac{1}{2}\left(\frac{d\sum_{j=1}^{n} a_j\varphi_j(x)}{dx}\right)^2 - f(x)(\sum_{j=1}^{n} a_j\varphi_j(x))\}\,dx = 0. \tag{6.1.8}$$

Hence a_i satisfies (6.1.7) □

Exercise 6.1.1 *Verify Equation (6.1.7)* □

Exercise 6.1.2 *Show that the solution of (6.1.6) satisfies the DE*

$$-\frac{d^2u}{dx_2} = f(x)\,, \tag{6.1.9}$$

with boundary conditions

$$u(0) = 0\,, \quad \frac{du}{dx}(1) = 0\,, \tag{6.1.10}$$

provided the solution of (6.1.6) is twice differentiable. □

The system of equations (6.1.7) is uniquely solvable if the coefficient matrix S is non-singular. This system can be written in matrix-vector notation by

$$S\mathbf{a} = \mathbf{f}\,, \tag{6.1.11}$$

with S an $(n \times n)$ matrix with elements $s_{ij} = \int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx$,

a an $(n \times 1)$ vector with elements a_j,

f an $(n \times 1)$ vector with elements $f_i = \int_0^1 f(x)\varphi_i(x)\,dx$.

There are many possible choices for the basis functions $\varphi_i(\mathbf{x})$, but we shall restrict ourselves to 2 specific ones.

Theorem 6.1.2 *Let the basis functions $\varphi_i(\mathbf{x})$ be given by*

$$\varphi_k(x) = \sin k\pi x \tag{6.1.12}$$

then the matrix **S** *in* (6.1.11) *has elements*

$$S_{kk} = \frac{k^2\pi^2}{2}\,. \tag{6.1.13}$$

and the solution a_k satisfies

$$a_k = \frac{2}{k^2\pi^2}\int_0^1 f(x)\sin(k\pi x)\,dx\,. \tag{6.1.14}$$

□

The basis functions $\varphi_k(x)$ are elements of Σ, since they are analytical functions and satisfy $\varphi_k(0) = 0$. Note that none of them satisfies the natural boundary condition. Using the orthogonality relations of the cosine we see that the basis functions $\varphi_k(x)$ produce a diagonal matrix S, with diagonal elements (6.1.13)

Exercise 6.1.3 *Prove that Equation (6.1.14) is the result of substituting the basis functions (6.1.12) into (6.1.7)* □

Exercise 6.1.4 *Show that the set a_k defined by (6.1.14) form the coefficients of the Fourier expansion of the exact solution $u(x)$ using functions* $\sin(k\pi x)$.
Hint: substitute 6.1.3 into 6.1.9. □

Theorem 6.1.3 *Let the basis functions $\varphi_i(\mathbf{x})$ be given by*

$$\varphi_k(x) = x^k \tag{6.1.15}$$

then the matrix $\mathbf{S}$ *in* (6.1.11) *is given by*

$$S = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & \frac{4}{3} & \frac{6}{4} & \frac{8}{5} & \frac{10}{6} \\ 1 & \frac{6}{4} & \frac{9}{5} & \frac{12}{6} & \cdots \\ 1 & \frac{8}{5} & \frac{12}{6} & \frac{16}{7} & \cdots \\ 1 & \frac{10}{6} & \cdots & \cdots & \cdots \\ \vdots & & & & \end{bmatrix} . \tag{6.1.16}$$

□

Exercise 6.1.5 *Derive Equation (6.1.16).* □

Matrix S in (6.1.16) is a *Hilbert matrix*. Not only is this matrix full, it is also very badly conditioned. Although the matrix is non-singular, numerically it is not invertible for relatively small values of n (order 10 to 20), on a 16 digits computer.

From these two specific choices for the basis functions we can draw some conclusions with respect to requirements for the basis functions.

6.1.3 Some observations concerning the basis functions

- With respect to the basis function $\varphi_k(x)$ defined in (6.1.12) it is clear that the solution $u^n(x)$ converges to the minimization problem, because the Fourier series is convergent (Exercise 6.1.4).
  One can also prove convergence in case of basis functions $\varphi_k(x)$ in (6.1.15), provided the system of linear equations can be solved.
- Even though the basis functions themselves do not satisfy the natural boundary condition, in the limit the linear combination does in some way, if there is convergence to the exact solution. In practice $\frac{du^n}{dx}(x)$ will be small in some sense, for n large enough.
- We have seen that with the specific choice (6.1.12) of the basis functions, the coefficient matrix is diagonal, and therefore the solution of the system of equations is trivial.
  This is not a coincidence: these functions form the eigenfunctions of the continuous eigenvalue problem

$$-\frac{d^2u}{dx_2} = \lambda u \; ; \; u(0) = 0 \, , \; u(1) = 0 \, . \tag{6.1.17}$$

  These eigenfunctions are orthogonal with respect to the inner product

$$\int_0^1 \frac{d\varphi_i}{dx} \frac{d\varphi_j}{dx} \, dx,$$

which implies that these inner products vanish if $i \neq j$. Also for more general problems one can define such an inner product and again the eigenfunctions have the same property. Unfortunately in practice it is almost impossible to find an analytical expression for the eigenfunctions. Numerical computation of the eigenfunctions is in general a harder task than solving the system of equations (6.1.11).

- On the other hand choosing an arbitrary set of basis functions leads to a full matrix. Unless the number of basis functions is very small, solution of such a system is very expensive.
  In finite difference methods and finite volume methods we always arrived at systems of equations with a sparse structure. If we want a sparse matrix in Ritz's method, it is necessary that most of the integrals

$$\int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx$$

  vanish. So the majority of the basis functions must be orthogonal with respect to the inner product defined by these integrals. We shall call such a set "nearly orthogonal".

- It is obvious that in the limit, the set of basis functions must span the complete space Σ, otherwise there are elements in Σ that can not be represented as linear combination of basis functions. Besides that, it would be nice if arbitrary functions in Σ could be approximated accurately with a small number of basis functions. The basis functions $\varphi_k(x)$ in (6.1.12) do not satisfy this property.

Combining all this we come to the following requirements for our set of basis functions:

1 the basis functions must be linearly independent

2 the basis functions must span the complete space Σ

3 the basis functions should be "nearly orthogonal"

4 arbitrary functions in Σ must be approximated accurately by a limited number of basis functions

At first sight it seems very difficult to satisfy all these demands. However, in Section 6.2 we shall show how to construct such basis functions by the finite element method.

We have treated lightly over the convergence of Ritz's method for a good reason: this is very hard to prove in general. For a specific case of practical importance we provide a proof: strongly elliptic operators (see Section 5.8.2).

6.1.4 Mathematical theory: convergence of Ritz's method

We consider the convergence of Ritz's method for the specific case of a linear operator satisfying properties (5.9.5) – (5.9.7). In order to do that we need a few tools. First we recall the definition of a basis for a Hilbert space.

Definition 6.1.1 *A family* $\{\varphi_\alpha\} \in \Sigma$ *is called a basis for the Hilbert space* Σ, *if the following two properties are satisfied.*

1. *Linear independence*

$$\sum_{i=1}^{N} \beta_i \varphi_i = 0 \ \textit{implies} \ \beta_i = 0\,, \ \ i = 1,..,N.$$

2. *Completeness*
For every $u \in \Sigma$ and a given $\varepsilon > 0$, there is a finite linear combination of basis functions such that the distance between u and this combination is smaller than ε. In formula:
$\forall \varepsilon > 0 \; \exists \; \{\varphi_{\alpha_1}, \varphi_{\alpha_2}, ..., \varphi_{\alpha_N}\}$ and $\{\beta_1, \beta_2, ..., \beta_N\}$, $N < \infty$, such that

$$\|u - \sum_{i=1}^{N} \beta_i \varphi_{\alpha_i}\|_\Sigma < \varepsilon\,,$$

in which $\| \cdot \|_\Sigma$ is the norm in Σ.

□

Theorem 6.1.4 *The Ritz equations with approximate solution* (6.1.3) *to solve the minimization problem (5.9.9):*

$$\min_{u \in \Sigma} J[u]\,, \ \ \textit{with} \ \ J[u] = \frac{1}{2}\|u\|_L^2 - (u, f)\,, \tag{6.1.18}$$

are given by

$$\sum_{j=1}^{n} a_j(\varphi_i, \varphi_j)_L = (f, \varphi_i) \quad i = 1, ..., n\,. \tag{6.1.19}$$

Exercise 6.1.6 *Prove Equation (6.1.19)* □

The system of linear equations (6.1.19) has a unique solution if and only if the coefficient matrix S defined by

$$S = \begin{bmatrix} (\varphi_1, \varphi_1)_L & (\varphi_2, \varphi_1)_L & \cdots & (\varphi_n, \varphi_1)_L \\ (\varphi_1, \varphi_2)_L & (\varphi_2, \varphi_2)_L & \cdots & (\varphi_n, \varphi_2)_L \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ (\varphi_1, \varphi_n)_L & (\varphi_2, \varphi_n)_L & \cdots & (\varphi_n, \varphi_n)_L \end{bmatrix}. \tag{6.1.20}$$

is non-singular.
S is a *Gramm matrix* for the set of functions $\varphi_1, \varphi_2, ..., \varphi_n$ in the space Σ. In the following we assume that $\{\varphi_i\}$ is a basis for Σ.

Theorem 6.1.5 *S defined by (6.1.20) is not singular.*

Proof
Suppose there is a non-zero vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_n)$ such that $S\boldsymbol{\alpha} = \mathbf{0}$.
Then $(\boldsymbol{\alpha}, S\boldsymbol{\alpha}) = 0$ so

$$\sum_i \sum_j \alpha_i \alpha_j (\varphi_i, \varphi_j)_L = 0\,.$$

Since the inner product is bilinear this implies

$$(\sum_i \alpha_i \varphi_i, \sum_j \alpha_j \varphi_j)_L = 0 \ \text{ or}$$
$$\|\sum_i \alpha \varphi_i\|_L = 0 \ \text{ and because } ||.||_L \text{ is a norm } \sum_i \alpha \varphi_i = 0\,.$$

By the linear independence of the basis functions, this implies $\alpha_i = 0$. So $S\boldsymbol{\alpha} = 0$ implies $\boldsymbol{\alpha} = 0$ and S is non-singular. □

Theorem 6.1.6 *If $\{\varphi_i\}$ is a basis for Σ, then approximation (6.1.3) converges to the solution u_0 of the minimization problem (6.1.18), .*

Proof
According to (5.9.14), J[u] can be written as

$$J[u] = \frac{1}{2}\|u - u_0\|_L^2 - \frac{1}{2}\|u_0\|_L^2 \,, \tag{6.1.21}$$

where $u_0 \in \Sigma$ minimizes $J[u]$ over Σ. This is a continuous function of the energy norm (why?), that is

$$\begin{aligned} &\forall \varepsilon > 0 \quad \exists\, \delta > 0 \text{ such that} \\ &\|u - u_0\|_L < \delta \Rightarrow |J[u] - J[u_0]| < \varepsilon \,. \end{aligned} \tag{6.1.22}$$

Let $u_0^n \in \Sigma^n = \text{Span}\,\{\varphi_j\}_{j=1}^n$ minimize $J[u]$ over Σ^n, then

$$J[u_0] \leq J[u_0^n] \leq J[u^n], \;\; \forall u^n \in \Sigma^n. \tag{6.1.23}$$

We choose u^n using completeness:

$$\begin{aligned} &\exists N \geq 1, \alpha_1, \ldots, \alpha_N \text{ such that} \\ &\|u_0 - u^n\|_L < \delta, \text{ with } u^n = \sum_{j=1}^{n} \alpha_j \varphi_j, \forall n \geq N. \end{aligned} \tag{6.1.24}$$

Continuity of $J[u]$ gives $|J[u^n] - J[u_0]| < \varepsilon$. Note that $\varepsilon > 0$ is arbitrary, for which $\delta > 0$ and $N \geq 1$ exist, and hence $J[u^n] \to J[u_0]$ as $n \to \infty$.
The Squeeze Theorem is applied to (6.1.23) to conclude that

$$J[u_0^n] \to J[u_0] \text{ as } n \to \infty. \tag{6.1.25}$$

Equation (6.1.21) finally implies $\|u_0^n - u_0\| \to 0$ as $n \to \infty$. □

6.2 The finite element method in $\mathbb{R}^1$

6.2.1 Introduction

Ritz's method can be used to solve the minimization problem and therefore also the corresponding PDE. The main issue in Ritz' method is the choice of the basis functions. In Section 6.1.3 we have formulated a number of properties the basis functions should satisfy, in order to get an attractive solution method.
We derive a construction technique that creates basis functions satisfying all these properties. The key to this method is the subdivision of the region Ω into subparts (elements) and an element-wise polynomial approximation of the unknown function.

First we demonstrate this construction in $\mathbb{R}^1$, subsequently it will be extended to $\mathbb{R}^2$.

6.2.2 The Poisson equation in $\mathbb{R}^1$

As first example we consider Poisson's equation in one dimension:

$$\begin{aligned} -\frac{d^2u}{dx_2} &= f(x) \,, \\ u(0) &= 0 \,, \\ \frac{du}{dx}(1) &= 0 \,. \end{aligned} \tag{6.2.1}$$

Exercise 6.2.1 *Show that the solution of (6.2.1) satisfies the minimization problem*

$$\min_{u\in\Sigma} J[u]\ ;\ \ J[u] = \int_0^1 \{\frac{1}{2}(\frac{du}{dx})^2 - u(x)f(x)\}\ dx\ . \tag{6.2.2}$$
$$\Sigma : \{u \text{ sufficiently smooth };\ u(0) = 0\}$$

□

The smoothness requirement implies that the integral in (6.2.2) makes sense.

The system of Ritz equations is given by (6.1.7).
In order to construct the basis functions we subdivide the interval [0, 1] into subintervals $e_k = [x_{k-1}, x_k]$ the *elements*, as shown in Figure 6.1.

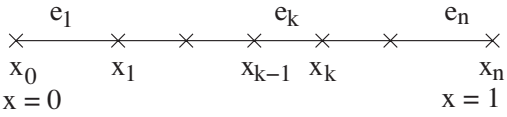


Figure 6.1: Subdivision of the interval [0, 1] in elements.

The solution u is approximated by a piecewise (lower order) polynomial defined element-wise. The most simple approximation is piecewise linear per element. Figure 6.2 shows a typical approximation $\tilde{u}$ of a function u by a piecewise linear polynomial. Note that the boundary condition $u(0) = 0$ is already satisfied by $\tilde{u}(0) = 0$.

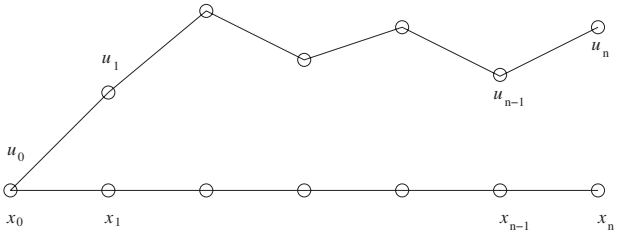


Figure 6.2: Approximation of $u(x)$ by $\tilde{u}(x)$.

Exercise 6.2.2 *Let $\tilde{u}$ be a piecewise linear approximation of u. Then $\tilde{u}$ does not belong to $C^1(0,1)$. Let $f(x)$ be a continuous function.*
Show that the integral in (6.2.2) remains finite when u is replaced by $\tilde{u}$. □

The linear interpolation polynomial of the function $u(x)$ over the element e_k is defined by

$$u_k(x) = \frac{x - x_k}{x_{k-1} - x_k} u(x_{k-1}) + \frac{x - x_{k-1}}{x_k - x_{k-1}} u(x_k)\ . \tag{6.2.3}$$

Exercise 6.2.3 *Show that Formula (6.2.3) is indeed the linear interpolation polynomial.*
□

Formally speaking it is not correct to use $u(x_k)$ since $u(x)$ is unknown. It would be better to use $\tilde{u}(x_k)$. However, as long as there is no confusion possible, we will omit the tilde.

We define linear *Lagrangian polynomials* $l_k(x)$

$$\ell_{k-1}(x) = \frac{x - x_k}{x_{k-1} - x_k}\,; \quad \ell_k(x) = \frac{x - x_{k-1}}{x_k - x_{k-1}}\,, \tag{6.2.4}$$

and write (6.2.3) as

$$u_k(x) = \ell_{k-1}(x)u_{k-1} + \ell_k(x)u_k\,. \tag{6.2.5}$$

u_k denotes $u(x_k)$.

Clearly $\ell_{k-1}(x)$ and $\ell_k(x)$ are linear on e_k and are defined by the relations

$$\ell_j(x_i) = \delta_{ij}\,; \quad i, j = k-1, k\,. \tag{6.2.6}$$

δ_{ij} is the Kronecker delta, defined by

$$\delta_{ij} = 0 \quad if \quad i \neq j \tag{6.2.7}$$

$$\delta_{ij} = 1 \quad if \quad i = j. \tag{6.2.8}$$

These relations define $\ell_j(x)$ uniquely (why?).

From (6.2.5) it is clear that $\tilde{u}(x)$ is a linear function of $u_0, u_1, ..., u_n$ so that we can write

$$\tilde{u}(x) = \sum_{j=0}^{n} u_j \varphi_j(x)\,. \tag{6.2.9}$$

The function $\varphi_i(x)$ consist of piecewise linear *Lagrangian polynomials* and may be considered as a generalized Lagrangian polynomials defined over the whole region Ω.
A typical $\varphi_i(x)$ has been sketched in Figure 6.3.

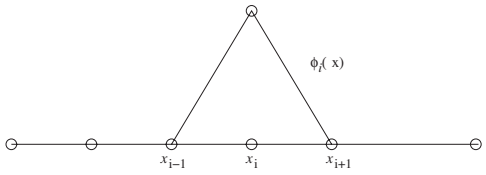


Figure 6.3: Example of a typical generalized Lagrangian polynomial.

φ_i is found taking all coefficients $u_k = 0$ $(i \neq k)$ and $u_i = 1$.

Exercise 6.2.4 *Sketch the basis functions $\varphi_0(x)$ and $\varphi_n(x)$.* □

Note that $\varphi_i(x)$ is only non-zero in the elements that contain the node x_i.
It is immediately clear that $\varphi_i(x)$ is defined by the following rules:

$$\begin{array}{ll} \text{a.} & \varphi_i(x) \text{ is linear in each element.} \\ \text{b.} & \varphi_i(x_j) = \delta_{ij}. \end{array} \tag{6.2.10}$$

Since $u_0 = 0$, (6.2.9) can be written as

$$\tilde{u}(x) = \sum_{j=1}^{n} u_j \varphi_j(x)\,. \tag{6.2.11}$$

The basis function $\varphi_0(x)$ will be used for non-homogeneous boundary conditions (see Section 6.2.4).

Theorem 6.2.1 *Suppose that an equidistant grid is used* $(x_{i+1} - x_i = h)$. *The system of Ritz equations* (6.1.7) *with the basis functions defined by (6.2.10) leads to the following system of equations:*

$$\frac{1}{h}\begin{bmatrix} 2 & -1 & & & & & \\ -1 & 2 & -1 & & & \bigcirc & \\ & -1 & 2 & -1 & & & \\ & & & \ddots & \ddots & & \\ & \bigcirc & & & -1 & 2 & -1 \\ & & & & & -1 & 1 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \\ \vdots \\ \\ u_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \\ \vdots \\ \\ f_n \end{bmatrix}, \tag{6.2.12}$$

with $f_i = \int_0^1 f(x)\varphi_i(x)\,dx$.

Exercise 6.2.5 *Prove Theorem 6.2.1.* □

Exercise 6.2.6 *Compare system (6.2.12) with the system obtained by the FDM.* □

6.2.3 Numerical integration

The right-hand-side vector of (6.2.12) contains an integral over a function $f(x)$. In general one can not compute such an integral analytically, so a numerical approximation is required. Since we are integrating over each element separately an obvious choice is to use a numerical rule based on the same element.

Well-known integration rules are for example

mid-point rule:
$$\int_{x_{k-1}}^{x_k} g(x)\,dx \approx (x_k - x_{k-1})g(x_{k-1/2})\,, \tag{6.2.13}$$

trapezoid rule:
$$\int_{x_{k-1}}^{x_k} g(x)\,dx \approx \frac{x_k - x_{k-1}}{2}\{g(x_{k-1}) + g(x_k)\}\,, \tag{6.2.14}$$

Simpson's rule:
$$\int_{x_{k-1}}^{x_k} g(x)\,dx \approx \frac{x_k - x_{k-1}}{6}\{g(x_{k-1}) + 4g(x_{k-1/2}) + g(x_k)\}\,. \tag{6.2.15}$$

All these rules can be written in the general form:

$$\int_{x_{k-1}}^{x_k} g(x)\,dx \approx \sum_{k=1}^{r} w_k g(v_k)\,, \tag{6.2.16}$$

with r the number of quadrature points,
w_k the weights, and
v_k quadrature points.

Exercise 6.2.7 *Give* r, w_k *and* v_k *for the midpoint rule, the trapezoid rule and Simpson's rule.* □

Another class of integration rules of the shape (6.2.16) are the *Gaussian rules*. These methods are characterized by the fact that integration points and weights are chosen such that the highest order of accuracy is reached with a particular number of

integration points. Weights and integration points of Gaussian integration rules can be found in various text books, like for example [50] and [37].

f_i in (6.2.12) is defined as

$$\int_0^1 f(x)\varphi_i(x)\,dx = \int_{x_{i-1}}^{x_i} f(x)\varphi_i(x)\,dx + \int_{x_i}^{x_{i+1}} f(x)\varphi_i(x)\,dx\,. \tag{6.2.17}$$

The integrand $g(x)$ in (6.2.17) is defined by $f(x)\varphi_i(x)$. We could use every possible integration rule of type (6.2.16) to compute (6.2.17).
We consider integration over the element $[x_{k-1}, x_k]$. In the Finite Element Method, one represents the numerical solution in terms of a linear combination of basis functions. For the case of linear basis functions, one approximates the function $g(x)$ by linear interpolation, that is

$$g(x) \approx g(x_{k-1})\varphi_{k-1}(x) + g(x_k)\varphi_k(x), \tag{6.2.18}$$

over the interval $[x_{k-1}, x_k]$. Subsequently, integration over $[x_{k-1}, x_k]$ gives

$$\begin{aligned} \int_{x_k}^{x_{k-1}} g(x)\,dx &\approx \int_{x_k}^{x_{k-1}} g(x_{k-1})\varphi_{k-1}(x) + g(x_k)\varphi_k(x)\,dx & (6.2.19)\\ &= g(x_{k-1})\int_{x_k}^{x_{k-1}} \varphi_{k-1}(x)\,dx + g(x_k)\int_{x_k}^{x_{k-1}} \varphi_k(x)\,dx. & (6.2.20) \end{aligned}$$

Using linearity of the basis functions and the relation $\varphi_i(x_j) = \delta_{ij}$, we get

$$\int_{x_k}^{x_{k-1}} g(x)\,dx \approx \frac{x_k - x_{k-1}}{2}(g(x_{k-1}) + g(x_k)). \tag{6.2.21}$$

Similar rules are derived for higher order basis functions with more quadrature points. Imagine that one integrates over interval $[x_{k-l}, x_{k+m}]$, $l, m \geq 0$, $l \cdot m \neq 0$. Let this interval contain nodes $x_{k-l}, x_{k-l+1}, \ldots, x_{k+m}$, then using the basis functions $\varphi_{k-l}, \varphi_{k-l+1}, \ldots, \varphi_{k+m}$ one can write the following interpolating approximation for $g(x)$

$$g(x) \approx \sum_{p=k-l}^{k+m} g(x_p)\varphi_p(x). \tag{6.2.22}$$

This interpolation is substituted into the integral over $g(x)$

$$\int_{x_{k-l}}^{x_{k+m}} g(x)\,dx \approx \sum_{p=k-l}^{k+m} g(x_p)\int_{x_{k-l}}^{x_{k+m}} \varphi_p(x)\,dx. \tag{6.2.23}$$

This type of quadrature based on interpolation on the FEM basis functions is called *Newton-Cotes rule*.

Theorem 6.2.2 *The Newton-Cotes rule applied to (6.2.12), the right-hand side vector can be written as*

$$h \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{n-1}) \\ f(x_n)/2 \end{bmatrix}. \tag{6.2.24}$$

Exercise 6.2.8 *Prove Theorem 6.2.2.* □

Note that the Newton-Cotes rule applied for a linear approximation in $\mathbb{R}^1$ is identical to the trapezoid rule. If a quadratic approximation is used this rule is identical to Simpson's rule.

Not only the type of interpolation, also the type of integration rule influences the accuracy of the solution. This subject will be considered in Section 8.7.

6.2.4 Boundary conditions

In our example (6.2.1) we have seen how homogeneous boundary conditions had to be treated. In summary:

- homogeneous natural boundary conditions pose no problem at all. They are an implicit part of the minimization problem.
- homogeneous essential boundary conditions fix the parameters on the boundary. The corresponding interpolation functions are not used as basis functions. In this way all basis functions satisfy the essential boundary conditions.

Non-homogeneous boundary conditions require only a small adaptation. We shall demonstrate this by extending example (6.2.1) with non-homogeneous boundary conditions:

$$\begin{aligned} -\frac{d^2u}{dx_2} &= f(x)\,, \\ u(0) &= a\,, \\ \frac{du}{dx}(1) &= b\,. \end{aligned} \tag{6.2.25}$$

The solution of (6.2.25) satisfies the minimization problem

$$\min_{u\in\Sigma} J[u] = \int_0^1 \{\frac{1}{2}(\frac{du}{dx})^2 - u(x)f(x)\}\,dx - bu(1)\,, \tag{6.2.26}$$
$$\Sigma : \{u \mid u \text{ sufficiently smooth; } \mathrm{u}(0) = \mathrm{a}\}$$

Exercise 6.2.9 *Show that the solution of (6.2.25) satisfies the minimization problem (6.2.26).* □

In order to apply Ritz's method we define

$$\tilde{u}(x) = \sum_{j=0}^{n} u_j\varphi_j(x) = \sum_{j=1}^{n} u_j\varphi_j(x) + u_0\varphi_0(x) \tag{6.2.27}$$

Again we use the linear Lagrangian polynomials $\ell_i(x)$ as basis functions, so $\varphi_i(x)$ is defined by (6.2.10). Now it is clear that $u_0 = a$ (why?).
If we use the approximation (6.2.27), the Ritz equations corresponding to (6.2.26) are equal to

$$\sum_{j=1}^{n} u_j \int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx = \int_0^1 f\varphi_i\,dx - u_0\int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_0}{dx}\,dx + b\varphi_i(1) \tag{6.2.28}$$
$$i = 1, 2, ..., n\,.$$

$$\frac{1}{h}\begin{bmatrix} 1 & -1 & & & & & \\ -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & \ddots & & & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 1 \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{bmatrix} = h\begin{bmatrix} f_0/2 \\ f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n/2 + b/h \end{bmatrix}$$

Figure 6.4: System of equation before applying essential boundary conditions.

Exercise 6.2.10

a. Derive (6.2.28).

b. Why is $i = 0$ not part of (6.2.28)?

c. Which of the functions $\varphi_i(x)$ is non-zero in $x = 1$?

□

From Formula 6.2.28 it will be clear that the non-homogeneous essential boundary condition gives a contribution to the right-hand side. To compute this contribution we first build the matrix and right-hand side as if there are no essential boundary conditions (see Figure 6.4). Following Exercise 6.2.10 row 1 (corresponding to $\varphi_i = \varphi_0$), must be removed. Since the matrix must be square also column 1 must be removed. This is done by multiplying this column by the given value u_0 and subtracting it from the right-hand side as sketched in Figure 6.5.

$$\frac{1}{h}\begin{bmatrix} 1 & -1 & & & & & \\ -1 & 2 & -1 & & & & \\ & 1 & 2 & -1 & & & \\ & & \ddots & & & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 1 \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{bmatrix} = h\begin{bmatrix} f_0/2 \\ f_1 \\ f2 \\ \vdots \\ f_{n-1} \\ f_n/2 + b/h \end{bmatrix} - \begin{bmatrix} u_0/h \\ -u_0/h \\ 0 \\ \vdots \\ \cdot \\ 0 \end{bmatrix}$$

Figure 6.5: Remove row 1. Multiply column 1 by u_0 and put it into the right-hand side.

The result of this operation is in Figure 6.6. The inhomogeneous natural boundary

$$\frac{1}{h}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & \ddots & & & & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 1 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{bmatrix} = h\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n/2 + b/h \end{bmatrix} - \frac{u_0}{h}\begin{bmatrix} -1 \\ 0 \\ \vdots \\ \\ 0 \end{bmatrix}$$

Figure 6.6: System of equations after application of essential boundary conditions.

condition also contributes to the right-hand side. This contribution is an immediate consequence of the minimization problem.

6.2.5 Element matrices and element vectors

In order to construct the matrix in (6.2.12) it was necessary to evaluate the integrals in (6.1.11). Since $\varphi_i(x)$ is defined in an element-wise manner, the natural way to do this is by an element-wise way.

$$\int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx = \sum_{k=1}^{n} \int_{e_k} \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx\,. \tag{6.2.29}$$

So instead of computing the left-hand side for all i and j, one might first compute all integrals

$$\int_{e_k} \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx\,, \tag{6.2.30}$$

for all i and j and add these integrals afterwards to get (6.2.29). This seems a very complicated way to compute the integrals. However at most 4 of the integrals in (6.2.30) are different from zero (Why?). We store these four integrals in a small matrix, the *element matrix*:

$$S^{e_k} = \begin{bmatrix} \int_{e_k} \frac{d\varphi_{k-1}}{dx}\frac{d\varphi_{k-1}}{dx}\,dx & \int_{e_k} \frac{d\varphi_{k-1}}{dx}\frac{d\varphi_k}{dx}\,dx \\ \int_{e_k} \frac{d\varphi_k}{dx}\frac{d\varphi_{k-1}}{dx}\,dx & \int_{e_k} \frac{d\varphi_k}{dx}\frac{d\varphi_k}{dx}\,dx \end{bmatrix}. \tag{6.2.31}$$

In the same way we create the *element vector*:

$$\mathbf{f}^{e_k} = \begin{bmatrix} \int_{e_k} f(x)\varphi_{k-1}(x)\,dx \\ \int_{e_k} f(x)\varphi_k(x)\,dx \end{bmatrix}. \tag{6.2.32}$$

Once all element matrices and vectors are computed, it is a matter of addition to compute the large matrix S and the large right-hand side $\mathbf{F}$. The main advantage of this approach is that all information of the minimization problem, the type of approximation in the element as well as the numerical integration rule applied, is stored locally.
To create the large matrix it is sufficient to know which unknowns are present in the element and to which entries the entries of the element matrix must be added. This is called the *topology* of the problem. The same holds for the large vector on the right-hand side.

This is a big advantage of the FEM. Once the region is subdivided into elements, it is sufficient to give a generic algorithm for the contributions of an arbitrary element. There is no need to worry about neighboring elements. Especially for more-dimensional unstructured grids, this is very attractive.

6.2.6 Assembly of the large matrix and vector

We have seen that all information for the FEM is stored in element matrices, element vectors and problem topology. The question is now: how can we construct the large matrix and vector from this information. The process of creating the large matrix and vector is known as assembly. To demonstrate this process we reuse minimization problem (6.2.2).
We consider the subdivision of the region [0, 1] into 4 elements as shown in Figure 6.7. Element e_i is defined by $e_i = [x_{i-1}, x_i]$ and the nodes are numbered from 0

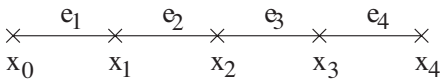


Figure 6.7: Subdivision of [0, 1] into 4 elements, and corresponding numbering of nodes and elements.

to 4. The unknowns have in this special case exactly the same numbering, where we know that $u_0 = 0$, so that the real unknowns are numbered from 1 to 4. In first instance the large matrix has size (5×5) and the large vector (5×1). The actual essential boundary condition is eliminated afterwards. The problem topology of this case is very simple; each element contains two unknowns.

$$\begin{array}{ll} e_1: & (0,1)\,, \\ e_2: & (1,2)\,, \\ e_3: & (2,3)\,, \\ e_4: & (3,4)\,. \end{array} \tag{6.2.33}$$

In the first step the large matrix and vector are cleared:

$$S^0 = \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \end{array} \\ \left[\begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \leftarrow 1 \\ \leftarrow 2 \\ \leftarrow 3 \\ \leftarrow 4 \\ \leftarrow 5 \end{array} \end{array} \quad \mathbf{f}^0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{6.2.34}$$

The element matrix for an arbitrary element e_k has shape

$$S^{e_k} = \frac{1}{x_k - x_{k-1}} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{6.2.35}$$

We apply the Newton-Cotes rule, to obtain the element vector

$$\mathbf{f}^{e_k} = \frac{x_k - x_{k-1}}{2} \begin{bmatrix} f(x_{k-1}) \\ f(x_k) \end{bmatrix}. \tag{6.2.36}$$

For the sake of simplicity we assume an equidistant grid with step size $x_k - x_{k-1} = h$.

Adding the first element matrix and right-hand side to (6.2.34) gives

$$S^1 = \frac{1}{h} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{f}^1 = \frac{h}{2} \begin{bmatrix} f(x_0) \\ f(x_1) \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{6.2.37}$$

Next we add S^{e_2} and f^{e_2} to S^1 and f^1:

$$S^2 = \frac{1}{h} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{f}^2 = \frac{h}{2} \begin{bmatrix} f(x_0) \\ 2f(x_1) \\ f(x_2) \\ 0 \\ 0 \end{bmatrix}. \tag{6.2.38}$$

Repeating this process for e_3 and e_4 gives:

$$S = S^4 = \frac{1}{h}\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}, \quad \mathbf{f} = \mathbf{f}^4 = \frac{h}{2}\begin{bmatrix} f(x_0) \\ 2f(x_1) \\ 2f(x_2) \\ 2f(x_3) \\ f(x_4) \end{bmatrix}. \tag{6.2.39}$$

This is of course the same expression as (6.2.12) and (6.2.24).
After the elimination of $u_0 = 0$ as described in Figure 6.6 the matrix, S, and the right-hand side, $\mathbf{f}$, become

$$S = \frac{1}{h}\begin{bmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & -1 & 2 & -1 \\ & & -1 & 1 \end{bmatrix}, \quad \mathbf{f} = \frac{h}{2}\begin{bmatrix} 2f(x_1) \\ 2f(x_2) \\ 2f(x_3) \\ f(x_4) \end{bmatrix}. \tag{6.2.40}$$

This construction seems very long-winded, especially for such a simple one dimensional problem. However, it is very well suited for computer implementation. All one needs is a topology formed by a subdivision in elements, as well as a procedure to compute an element matrix and element vector for an arbitrary element. The rest is a matter of book keeping. How complicated the mesh may be, the assembly process is always the same. All finite element codes work according to this principle.

6.2.7 Boundary conditions and assembly

We would like to apply the same procedure as in Section 6.2.6, even in the case of non-homogeneous boundary conditions. To that end we consider the DE (6.2.25) with corresponding minimization problem (6.2.26).

In the right-hand side of (6.2.28) we see two extra terms compared to (6.1.11):

$$-u_0 \int_0^1 \frac{d\varphi_i}{dx}\frac{d\varphi_0}{dx}\,dx + b\varphi_i(1)\,. \tag{6.2.41}$$

The integral in the first of these terms is already present in the element matrix of element e_1:

$$S^{e_1} = \begin{bmatrix} \int_{e_1} \frac{d\varphi_0}{dx}\frac{d\varphi_0}{dx}\,dx & \int_{e_1} \frac{d\varphi_1}{dx}\frac{d\varphi_0}{dx}\,dx \\ \int_{e_1} \frac{d\varphi_1}{dx}\frac{d\varphi_0}{dx}\,dx & \int_{e_1} \frac{d\varphi_1}{dx}\frac{d\varphi_1}{dx}\,dx \end{bmatrix}. \tag{6.2.42}$$

If we skip the first row of S^{e_1} and multiply the remaining part of the first column by u_0 and subtract this term of the right-hand side vector, then we get precisely the first term in (6.2.41).
This step can easily be performed by a finite element program, provided point 0 is marked as a point with essential boundary condition.
So, even if the first row of S^{e_1} is not used, it is conceptually simpler always to create a 2×2 matrix for all elements e_k.
The term $-b\varphi(1)$ only influences the element vector in the last element. However, also in this case it is better not to worry about boundary conditions in the element vector.

In order to create this extra term we introduce an extra boundary element (in this case a point element), consisting of 1 point $(x = 1)$ only. This element is solely meant to incorporate the term $b\varphi_i(1)$

Exercise 6.2.11 *Show that the element matrix and element vector for the boundary condition*

$$\left.\frac{du}{dx}\right|_{(x=1)} = b$$

are given by:

$$S^e = [0]\,, \quad \mathbf{f}^e = [b]\,. \tag{6.2.43}$$

□

The elimination of the essential boundary conditions can be described in the following formulae.

Suppose we renumber the unknowns such that we have first all non-prescribed unknowns $(\mathbf{u}_i)$ (also called *degrees of freedom*) and subsequently all unknowns given by the essential boundary conditions $(\mathbf{u}_b)$.

The system of equations can be written as:

$$\begin{bmatrix} S_{ii} & S_{ib} \\ S_{bi} & S_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ \mathbf{u}_b \end{bmatrix} = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{f}_b \end{bmatrix}\,. \tag{6.2.44}$$

Since $\mathbf{u}_b$ is given (6.2.44) can be reduced to

$$S_{ii}\mathbf{u}_i = \mathbf{f}_i - S_{ib}\mathbf{u}_b\,, \tag{6.2.45}$$

and this is the actual system to be solved.

The last set of equations in (6.2.44) contains also some useful information. Suppose that $\mathbf{u}_b$ is not given, but that the flux (natural boundary condition) is prescribed. In that case the last equation would be:

$$S_{bi}\mathbf{u}_i + S_{bb}\mathbf{u}_b = \mathbf{f}_b + \mathbf{b}\,, \tag{6.2.46}$$

where $\mathbf{b}$ is the given flux (see 6.2.43)).

The consequence is that if $\mathbf{u}_b$ is given and $\mathbf{u}_i$ has been solved from (6.2.44), the flux can be approximated by

$$\mathbf{b} = S_{bi}\mathbf{u}_i + S_{bb}\mathbf{u}_b - \mathbf{f}_b\,. \tag{6.2.47}$$

This term is also known under the name *reaction force*. It represents the flux through the boundary with essential boundary conditions.

In the next Section we shall extend our example to two-dimensions.

6.2.8 Periodical boundary conditions

Consider the Poisson equation with periodical boundary conditions.

$$\frac{d^2u}{dx^2} = f, \qquad u(0) = u(1), \quad \frac{du(0)}{dx} = \frac{du(1)}{dx}. \tag{6.2.48}$$

Theorem 6.2.3 *The minimization problem corresponding to* (6.2.48) *is given by*

$$\min_{u\in\Sigma} J[u] = \int_0^1 \{\frac{1}{2}\left(\frac{du}{dx}\right)^2 - f(x)u(x)\}\,dx\,, \tag{6.2.49}$$

$$\Sigma:\ \{u \mid u \text{ sufficiently smooth};\ u(0) = u(1)\}\,.$$

Exercise 6.2.12 *Prove Theorem 6.2.3.* □

Note that the boundary condition $\frac{du(0)}{dx} = \frac{du(1)}{dx}$ is a natural boundary condition for this minimization problem.
In order to apply the Ritz method we set

$$u^n(\mathbf{x}) = \sum_{j=0}^{n} a_j \varphi_j(\mathbf{x}) \,, \tag{6.2.50}$$

with $a_0 = a_n$. Hence the unknowns in the first and last node are identified. By doing so, the first and last element are coupled to each other, which is precisely the idea of periodical boundary conditions.

Exercise 6.2.13 *Compute the matrix and right-hand side for the solution of 6.2.49 using linear basis functions and Newton Cotes quadrature.* □

6.2.9 The structure of finite element packages

In the previous sections it has been made clear that the finite element method is well suited for automatization. As a consequence a lot of (commercial) packages have been developed over the last decades. Most packages subdivide the finite element process in three steps.

- Preprocessing: usually the mesh generation
- Solving: the actual FEM
- Postprocessing: showing the results

The solve part consists globally of the following steps:

```
Read input and mesh
Compute the structure of the large matrix from the topology
Clear large matrix and vector
for all elements (including boundary elements) do
    Compute element matrix and vector
    Add element matrix to large matrix
    Add element vector to large vector
end for
Apply essential boundary conditions
Solve system of equations
Write results for postprocessing
```

The crucial step is the computation of element matrix and vector. In fact this part defines the actual differential equation and type of approximation.
In general one uses preprogrammed finite element subroutines to compute element matrix and vector, however, it is also possible that the user supplies his own element matrix and vector. In this way she may use the general concept of the FEM, and still solve her own specific problem.

6.3 The finite element method in $\mathbb{R}^2$

6.3.1 The Poisson equation in $\mathbb{R}^2$

We have demonstrated the FEM for the one-dimensional Poisson equation using linear interpolations. And even that simple equation showed many of the issues of the FEM. In this section we shall extended that example to $\mathbb{R}^2$. More general cases will be the subject of Chapter 7.

Consider Poisson's equation defined on a bounded region $\Omega \subset \mathbb{R}^2$ with boundary $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$.

$$-\Delta u = f\,, \qquad \mathbf{x} \in \Omega \tag{6.3.1}$$

with boundary conditions

$$\begin{aligned} u &= g_1(\mathbf{x})\,, \qquad \mathbf{x} \in \Gamma_1 \\ \frac{\partial u}{\partial n} &= g_2(\mathbf{x})\,, \qquad \mathbf{x} \in \Gamma_2 \\ \alpha u + \frac{\partial u}{\partial n} &= g_3(\mathbf{x})\,, \qquad \mathbf{x} \in \Gamma_3 \quad (\alpha \geq 0)\,. \end{aligned} \tag{6.3.2}$$

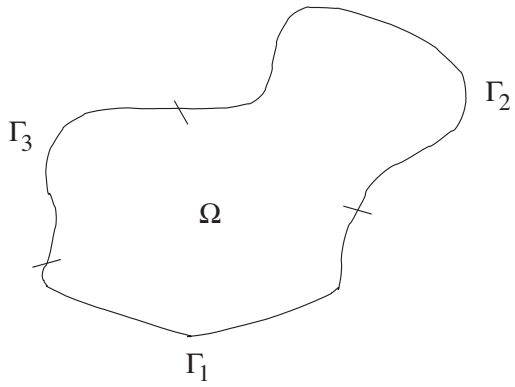


Figure 6.8: Region Ω with boundary $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$.

The minimization problem corresponding to (6.3.1), (6.3.2) is given by

$$\min_{u \in \Sigma} J[u] \tag{6.3.3}$$

with

$$J[u] = \int_\Omega \{\frac{1}{2}|\nabla u|^2 - uf\}\, d\Omega - \int_{\Gamma_2} g_2 u\, d\Gamma - \int_{\Gamma_3} g_3 u\, d\Gamma + \frac{1}{2}\int_{\Gamma_3} \alpha u^2\, d\Gamma$$

and $\Sigma = \{\text{sufficiently smooth} \mid u = g_1|_{\Gamma_1}\}$.

Exercise 6.3.1 *Prove that the PDE formulation (6.3.1) together with (6.3.2) is 'equivalent' to the minimization form (6.3.3).* □

To provide a general framework we first apply Ritz's method formally.

First we choose a set of basis functions $\varphi_i(\mathbf{x}) \in \Sigma_0$ with

$$\Sigma_0 = \{u \mid \ u|_{\Gamma_1} = 0\}\,. \tag{6.3.4}$$

Next we choose an arbitrary but known function u_B that satisfies

$$u_B(\mathbf{x}) = g_1(\mathbf{x}),\ \mathbf{x} \in \Gamma_1\,. \tag{6.3.5}$$

The solution $u(\mathbf{x})$ is approximated by a finite dimensional subset of Σ:

$$u^n(\mathbf{x}) = \sum_{j=1}^{n} u_j \varphi_j(\mathbf{x}) + u_B(\mathbf{x}) \tag{6.3.6}$$

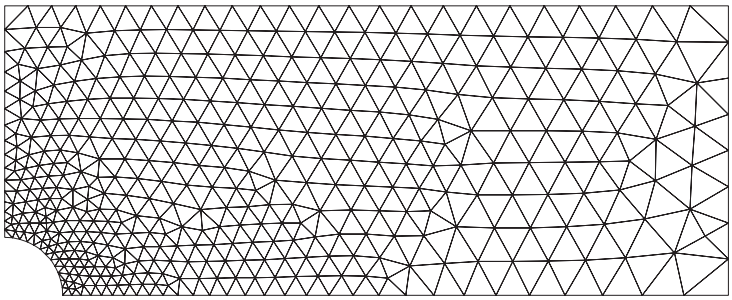


Figure 6.9: Subdivision in triangles.

Clearly we have $u^n(\mathbf{x}) \in \Sigma$. The set of Ritz equations to approximate the minimization problem (6.3.3) by (6.3.6) is given by:

$$\sum_{j=1}^{n} u_j \{ \int_{\Omega} (\nabla \varphi_i \cdot \nabla \varphi_j)\, d\Omega + \int_{\Gamma_3} \alpha \varphi_i \varphi_j \, d\Gamma \} = \int_{\Omega} f \varphi_i \, d\Omega + \int_{\Gamma_2} g_2 \varphi_i \, d\Gamma +$$
$$\int_{\Gamma_3} g_3 \varphi_i \, d\Gamma - \int_{\Omega} \nabla \varphi_i \cdot \nabla u_B \, d\Omega - \int_{\Gamma_3} \alpha \varphi_i u_B \, d\Gamma \,. \tag{6.3.7}$$

Exercise 6.3.2 *Derive (6.3.7).* □

The next step is to provide FEM basis functions. To this end we subdivide the region into elements and define a polynomial approximation on each element.

6.3.2 Linear elements in $\mathbb{R}^2$

The extension of the linear line element in $\mathbb{R}^1$ is the triangle in $\mathbb{R}^2$. Figure 6.9 shows a typical subdivision of a region into triangles. In order to construct a linear polynomial on each triangle we need 3 parameters. A natural choice is to use the function values in the three vertices of the triangle (Figure 6.10).

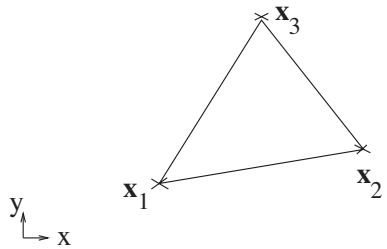


Figure 6.10: Linear triangle with nodal points.

This has the added benefit of making the approximation continuous across element boundaries.
Following the same procedure as in $\mathbb{R}^1$, it will be clear that the corresponding basis functions φ_i have the properties:

$$\begin{aligned} &1) \quad \varphi_i(\mathbf{x}) \quad \text{is linear per triangle ,} \\ &2) \quad \varphi_i(\mathbf{x}_j) = \delta_{ij} \,. \end{aligned} \tag{6.3.8}$$

A typical basis function is sketched in Figure 6.11.

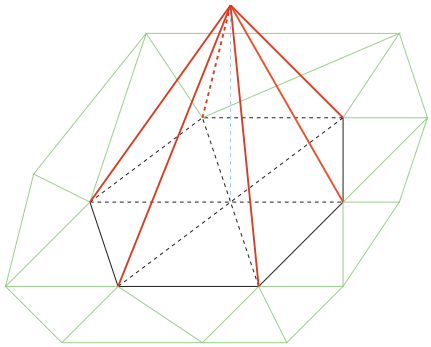


Figure 6.11: Sketch of a typical linear basis function.

(6.3.8) defines the basis functions implicitly. In order to compute the integral in (6.3.7), it is necessary to have an explicit expression per element. Consider the triangle in Figure 6.10).
A linear polynomial is defined by

$$\varphi_i(\mathbf{x}) = \alpha_i + \beta_i x + \gamma_i y \,. \tag{6.3.9}$$

(6.3.8) defines 3 equations for each i to compute the parameters $\alpha_i, \beta_i, \gamma_i$. Substitution of (6.3.8) in (6.3.9) leads to the following system of linear equations:

$$\begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{6.3.10}$$

Exercise 6.3.3 *Verify (6.3.10)* □

The system of equations (6.3.10) has a solution if the coefficient determinant Δ (see (6.3.11)) does not vanish

$$\Delta = \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} . \tag{6.3.11}$$

Δ in (6.3.11) can be expressed as

$$\Delta = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1) \,, \tag{6.3.12}$$

which is twice the area of the triangle in Figure 6.10, as will be shown in Section 8.2.

Exercise 6.3.4 *Prove (6.3.12).*
Hint: subtract the first row from the second and the third row. □

If the orientation of the nodes is counterclockwise Δ is positive, otherwise it is negative.

Exercise 6.3.4 shows that the system is regular as long as the area of the triangle differs from 0.
The solution of system of equations (6.3.10) is given by

$$\begin{aligned} \beta_1 &= \frac{1}{\Delta}(y_2 - y_3) \,, \quad \beta_2 = \frac{1}{\Delta}(y_3 - y_1) \,, \quad \beta_3 = \frac{1}{\Delta}(y_1 - y_2) \,, \\ \gamma_1 &= \frac{1}{\Delta}(x_3 - x_2) \,, \quad \gamma_2 = \frac{1}{\Delta}(x_1 - x_3) \,, \quad \gamma_3 = \frac{1}{\Delta}(x_2 - x_1) \,, \\ \alpha_i &= 1 - \beta_i x_i - \gamma_i y_i \,. \end{aligned} \tag{6.3.13}$$

Exercise 6.3.5 *Show that (6.3.13) is the solution of (6.3.10).*
Hint: formulate the equations for α_1, β_1 and γ_1 and subtract the first equation from the second and third one. Repeat this process for the other unknowns. □

Now we have all ingredients to evaluate the integrals in formula (6.3.7). As we have seen in Section 6.2.5, we only need to compute the element matrix and element vector.
First of all we shall consider the case that α, g_2 and g_3 are all equal to zero, so that all boundary integrals in (6.3.7) vanish. Later on we shall pay attention to these boundary integrals in inhomogeneous boundary problems.
The element matrix for the linear triangle corresponding to (6.3.7) is given by

$$S^{e_k} = \begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix} , \tag{6.3.14}$$

With $S_{ij} = \int_{e_k} \nabla\varphi_i \cdot \nabla\varphi_j \, d\Omega$.

Exercise 6.3.6 *Show that (6.3.14) is the element matrix corresponding to (6.3.7).* □

From (6.3.9) - (6.3.14) it follows that

$$S_{ij} = \frac{|\Delta|}{2}(\beta_i\beta_j + \gamma_i\gamma_j) \, . \tag{6.3.15}$$

The element vector for the linear triangle corresponding to (6.3.7) is given by:

$$\mathbf{f}^{e_k} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} , \tag{6.3.16}$$

with

$$f_i = \int_{e_k} f(\mathbf{x})\varphi_i(\mathbf{x}) \, d\Omega \, . \tag{6.3.17}$$

Exercise 6.3.7 *Verify (6.3.16) and (6.3.17).* □

We shall have to evaluate (6.3.17) numerically.

6.3.3 Numerical integration in $\mathbb{R}^n$

Numerical integration in $\mathbb{R}^1$ has been the subject of Section 6.2.3. In this section we consider the more general case of integration over triangles in $\mathbb{R}^2$ or tetrahedrons in $\mathbb{R}^3$. Integration over other types of elements will be the subject of Section 8.7.
In $\mathbb{R}^2$ and $\mathbb{R}^3$ we can derive integration rules of the same type as mid-point rule, trapezoidal rule or Simpson's rule, by integrating polynomials of a certain degree exactly. Besides that, for these triangles and tetrahedrons it is possible to construct Gaussian integration rules. Weights and integration points can be found in numerous text books. (See for example [50]).

Definition 6.3.1 *A simplex in $\mathbb{R}^n$ is the convex hull of $n+1$ points in $\mathbb{R}^n$.*

A simplex in $\mathbb{R}^1$ is an interval, in $\mathbb{R}^2$ a triangle and in $\mathbb{R}^3$ a tetrahedron. □

The next theorems gives a general formula for integration of powers of linear basis functions over simplices. It is very useful.

Theorem 6.3.1 *Let S be a triangle in* $\mathbb{R}^2$ *and let* Δ *be the determinant defined by*

$$\Delta = \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}, \tag{6.3.18}$$

with $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ *the vertices of S.*

Let $\lambda_i(\mathbf{x})$ *be the linear basis functions over S defined by*

$$\begin{aligned} &\lambda_i(\mathbf{x}) \quad \textit{linear} \\ &\lambda_i(\mathbf{x}_j) = \delta_{ij} \quad i, j, = 1, 2, 3\,. \end{aligned} \tag{6.3.19}$$

Then the following general integration rule holds:

$$\int_S \lambda_1^{m_1} \lambda_2^{m_2} \lambda_3^{m_3} = \frac{m_1! m_2! m_3!}{(m_1 + m_2 + m_3 + 2)!} |\Delta|\,,$$

for all $m_i \geq 0$.

Proof: See Holand and Bell (1969)[20], page 84.

Exercise 6.3.8 *Use Theorem 6.3.1 to show that*

$$\int_S \lambda_i = \frac{|\Delta|}{6}. \tag{6.3.20}$$

□

This theorem can be extended to n dimensions:

Theorem 6.3.2 *Let S be a simplex in* $\mathbb{R}^n$ *and let* Δ *be the determinant defined by*

$$\Delta = \begin{vmatrix} 1 & x_{1,1} & x_{2,1} & \cdots & x_{n,1} \\ 1 & x_{1,2} & x_{2,2} & \cdots & x_{n,2} \\ \vdots & \vdots & \vdots & \vdots & \\ 1 & x_{n+1,1} & x_{n+1,2} & \cdots & x_{n+1,n} \end{vmatrix}, \tag{6.3.21}$$

with $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{n+1}$ *the vertices of S, and* $x_{i,j}$ *the* j^{th} *component of* $\mathbf{x}_i$.

Let $\lambda_i(\mathbf{x})$ *be the linear basis functions over S defined by*

$$\begin{aligned} &\lambda_i(\mathbf{x}) \quad \textit{linear} \\ &\lambda_i(\mathbf{x}^j) = \delta_{ij} \quad i, j, = 1, 2, \cdots, n+1\,. \end{aligned} \tag{6.3.22}$$

Then the following general integration rule holds:

$$\int_S \lambda_1^{m_1} \lambda_2^{m_2} \cdots \lambda_{n+1}^{m_{n+1}}\, d\Omega = \frac{m_1! m_2! \cdots m_{n+1}!}{(\sum_i m_i + n)!} |\Delta|\,,$$

for all $m_i \geq 0$.

Exercise 6.3.9 *Apply Theorem 6.3.2 to show that*

$$\int_{x_1}^{x_2} \lambda_i \, dx = \frac{h}{2}. \tag{6.3.23}$$

□

Theorem 6.3.3 *Let $\lambda_i(\mathbf{x})$ be defined as in (6.3.22). Then*

$$\sum_{i=1}^{n+1} \lambda_i(\mathbf{x}) = 1 \, . \tag{6.3.24}$$

Exercise 6.3.10 *Prove Theorem 6.3.3.*
□

Exercise 6.3.11 *Show that* $\int_S d\Omega = \frac{|\Delta|}{n!}$.
Hint: use Theorem 6.3.2. □

Exercise 6.3.12 *Find the midpoint rule for a triangle and a tetrahedron.*
Hint: the midpoint rule is a one point integration rule that is exact for linear polynomials. Determine this point by integrating the linear basis functions. □

Exercise 6.3.13 *Prove that the Newton-Cotes rule for a triangle in $\mathbb{R}^2$ with linear basis functions is given by*

$$\int_S g(\mathbf{x}) \, d\Omega = \frac{|\Delta|}{6}(g(\mathbf{x}_1) + g(\mathbf{x}_2) + g(\mathbf{x}_3)) \, . \tag{6.3.25}$$

Hint: use Theorem 6.3.2. □

Exercise 6.3.14 *Show that if the Newton-Cotes rule is applied to (6.3.16), (6.3.17), the element vector is given by*

$$\mathbf{f}^{e_k} = \frac{|\Delta|}{6} \begin{bmatrix} f(\mathbf{x}^1) \\ f(\mathbf{x}^2) \\ f(\mathbf{x}^3) \end{bmatrix} . \tag{6.3.26}$$

□

6.3.4 Boundary conditions

The way in which essential boundary conditions are treated is independent of the dimension of the space. With respect to natural boundary conditions we follow a similar approach as in $\mathbb{R}^1$. In that case we introduced point elements to treat the extra term $b\varphi_i(1)$. In equation (6.3.7) we find four boundary integrals, three of which are related to natural boundary conditions.

$$\int_{\Gamma_3} \alpha \varphi_i \varphi_j \, d\Gamma \, , \tag{6.3.27}$$

$$\int_{\Gamma_2} g_2 \varphi_i \, d\Gamma \, , \tag{6.3.28}$$

$$\int_{\Gamma_3} g_3 \varphi_i \, d\Gamma \, . \tag{6.3.29}$$

Since we use linear triangles we actually approximate the boundary by straight lines. In Section 8.7 we shall return to the consequences of this approximation.

For the moment we assume that the boundary is exactly given by the straight boundary lines of the subdivision. Of course it is possible to add the contribution of the integrals (6.3.27)-(6.3.29) to all element matrices and vectors that correspond to boundary triangles that have a side in common with Γ_2 or Γ_3. From a computational point of view this is not so desirable because this means that not all triangles have the same type of element matrix and element vector. So following our discussion in $\mathbb{R}^1$ it is natural to introduce extra line elements just for the computation of the integrals in (6.3.27)-(6.3.29). These line elements (also called boundary elements) are implicitly defined by the boundary and the subdivision in triangles, see for example Figure 6.12.



Figure 6.12: Subdivision in triangles and line elements.

A typical line element is sketched in Figure 6.13.

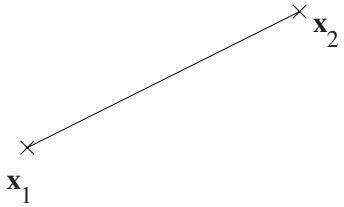


Figure 6.13: Example of a linear line element.

Only two base functions differ from zero on this element (why?), so the element matrix must have size (2×2) and the element vector (2×1). The element matrix for the boundary elements along Γ_3 is given by

$$S^{e_k} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix},$$

with

$$S_{ij} = \int_{e^k} \alpha \varphi_i \varphi_j \, d\Gamma \,. \tag{6.3.30}$$

The element vector for the Γ_3 boundary elements are defined by

$$\mathbf{f}^{e_k} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad f_i = \int_{e^k} g_3 \varphi_i \, d\Gamma \,.$$

Exercise 6.3.15 *Give the line element matrices and vectors along* Γ_2. □

To compute the line integrals along element e_k we map the element $(\mathbf{x}_1, \mathbf{x}_2)$ onto $(0, h)$, with h the length of the element given by

$$h = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}\,. \tag{6.3.31}$$

Hence

$$S_{ij} = \int_{e^k} \alpha \varphi_i \varphi_j \, d\Gamma \;\; = \;\; \int_0^h \alpha(t) \varphi_i(t) \varphi_j(t) \, dt, \tag{6.3.32}$$

where $t = 0$ corresponds to x_1 and $t = h$ to x_2.
Application of Newton-Cotes to (6.3.32) gives

$$S_{ij} = \frac{h}{2} \alpha(t_i) \delta_{ij} \;\; = \;\; \alpha(\mathbf{x}_i) \delta_{ij}\,. \tag{6.3.33}$$

Exercise 6.3.16 *Prove* (6.3.33). □

In the same way we can approximate the elements of the element vector along Γ_3 by

$$f_i = \frac{h}{2} g_3(x_i). \tag{6.3.34}$$

Exercise 6.3.17 *Prove* (6.3.34). □

Exercise 6.3.18 *Compute the element matrix and element vector for the line elements along* Γ_2. □

In case of essential boundary conditions we have to choose $u_B(x)$. It is natural to approximate $u_B(x)$ by a linear combination of basis functions corresponding to the points on Γ_1. Hence

$$u_B(x) = \sum_{j=n+1}^{n+n_B} u_j \phi_j(x) \tag{6.3.35}$$

and the approximation can be written as

$$u^n = \sum_{j=1}^{n+n_B} u_j \phi_j(x), \tag{6.3.36}$$

where the last n_B parameters u_j are prescribed. So Equation (6.3.7) reduces to

$$\sum_{j=1}^{n+n_B} u_j \{ \int_\Omega (\nabla \varphi_i \cdot \nabla \varphi_j) \, d\Omega + \int_{\Gamma_3} \alpha \varphi_i \varphi_j \, d\Gamma \} = \int_\Omega f \varphi_i \, d\Omega + \int_{\Gamma_2} g_2 \varphi_i \, d\Gamma + \int_{\Gamma_3} g_3 \varphi_i \, d\Gamma \,, i = 1, 2, \ldots, n\,. \tag{6.3.37}$$

The implementation of essential boundary conditions is exactly the same as described in (6.2.44) and (6.2.45).

6.4 Theoretical remarks

6.4.1 Smoothness requirements

In Section 5.6 we have seen that it is necessary that integrals like $\int_\Omega u_x^2 \, d\Omega$ and $\int_\Omega u_y^2 \, d\Omega$ must exist and be finite. This must also be true for the approximation and hence for the basis functions $\varphi_i(\mathbf{x})$.

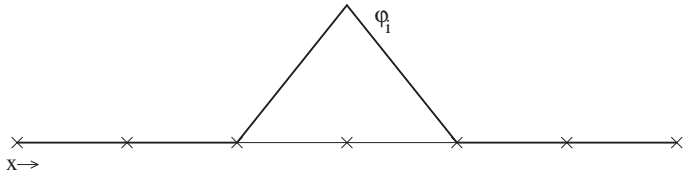


Figure 6.14: One-dimensional basis function $\varphi_i(x)$.

If we consider the one-dimensional basis function $\varphi_i(\mathbf{x})$ sketched in Figure 6.14, then we see that this function is infinitely often differentiable in the interior of each element, but not differentiable on some of the element boundaries (why?). On these boundaries the basis function is continuous. If we split the integral

$$\int_0^1 \frac{d\varphi_i}{dx}\, dx \tag{6.4.1}$$

into

$$\int_0^1 \frac{d\varphi_i}{dx}\, dx = \sum_{k=1}^{m} \int_{e_k} \frac{d\varphi_i}{dx}\, dx\,, \tag{6.4.2}$$

then each of the integrals exists and is finite. This operation is allowed as long as the contribution for the element boundaries in equal to zero.
This is the case if $\varphi_i(x)$ is continuous since the "length" of a point is zero, and therefore the point has no contribution to the integral. Mathematically speaking we say that a point has "zero measure".

However, if $\varphi_i(x)$ would be discontinuous like the one sketched in Figure 6.15, then the derivative on the element interface will be infinite. In fact the derivative is a *delta* function and the contribution of the point with the discontinuity does not vanish. The integral $\int_\Omega u_x^2\, d\Omega$ is no longer finite and such basis functions are not allowed.

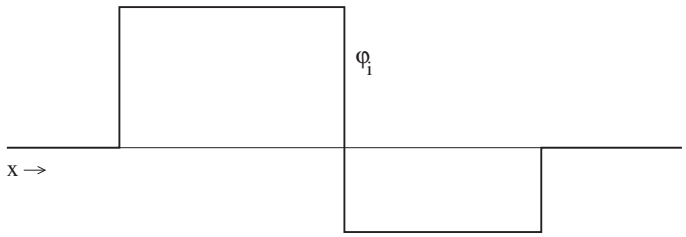


Figure 6.15: Example of a discontinuous basis function.

So for a second order problem in $\mathbb{R}^1$ it is necessary that the basis functions are not only piecewise smooth but also globally continuous.

In $\mathbb{R}^2$ the theory is slightly more complicated but one can say in general that a basis function that is a polynomial per element and continuous over the element boundaries may be used for second order problems. Discontinuous basis functions are in

general not allowed. For fourth order problems we have to require continuity of the first derivative (why?).

Elements with basis functions that satisfy the continuity requirement are called *conforming*, and the basis functions are referred to as admissible. Elements not satisfying this requirement are *non-conforming*. Sometimes they are used for special applications. See Section 8.8.1.

Exercise 6.4.1 *Show that the two-dimensional basis function derived in Section 6.3.2 is admissible.* □

6.4.2 Mathematical theory of FEM

Consider the linear equation

$$Lu = f \quad u \in \Sigma. \tag{6.4.3}$$

The corresponding minimization problem is defined as (see 5.9.9)

$$\min_{u\in\Sigma} J[u], \text{ with } J[u] = \frac{1}{2}\|u\|_L^2 - (u, f). \tag{6.4.4}$$

We have seen that Ritz's method may be formulated as (see 6.1.3)

$$\min_{u_h\in\Sigma_h} J[u_h], \text{ with } \Sigma_h \text{ a finite dimensional subspace of } \Sigma. \tag{6.4.5}$$

Now we can prove the following theorem:

Theorem 6.4.1 *Let $\hat{u}$ be the solution of (6.4.4) and $\hat{u}_h$ be the solution of (6.4.5) then*

$$(\hat{u}, v)_L = (f, v) \quad \forall v \in \Sigma. \tag{6.4.6}$$

$$(\hat{u}_h, v_h)_L = (f, v_h) \quad \forall v_h \in \Sigma_h. \tag{6.4.7}$$

Proof Equation (6.4.6) follows immediately by substituting $u = \hat{u} + \epsilon v$ in (6.4.4), differentiating with respect to ϵ and putting $\epsilon = 0$ like in the derivation of the Euler-Lagrange equations. According to (6.1.19) Ritz's equations $\mathbf{Su} = \mathbf{f}$ can be written as

$$\sum_{j=1}^{n} u_j(\varphi_i, \varphi_j)_L = (f, \varphi_i) \quad (i = 1, ..., n). \tag{6.4.8}$$

Let $v_h \in \Sigma_h$ be given by $v_h = \sum_{i=1}^{n} v_i$. We take the inner product of $\mathbf{Su} = \mathbf{f}$ with the vector $\mathbf{v} = (v_1, v_2, ..., v_n)$ to get

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(\varphi_i, \varphi_j)_L u_j v_i = \sum_{i=1}^{n} v_i(f, \varphi_i). \tag{6.4.9}$$

Using the linearity of the inner product we get

$$(\sum_{i=1}^{n} v_i\varphi_i, \sum_{j=1}^{n} u_j\varphi_j)_L = (f, \sum_{i=1}^{n} v_i\varphi_i). \tag{6.4.10}$$

And this of course equal to

$$(\hat{u}_h, v_h)_L = (f, v_h) \quad \forall v_h \in \Sigma_h. \tag{6.4.11}$$

Since v_h is arbitrary we have proved (6.4.7).

With Theorem 6.4.1 we can prove the following

Theorem 6.4.2 *Let $\hat{u}$ be the solution of (6.4.4) over Σ, $\hat{u}_h$ be the solution of (6.4.5) and let $\tilde{u}$ be the interpolation of $\hat{u}$ by the FEM basis functions. So $\tilde{u}$ is defined by*

$$\tilde{u} = \sum_{k=1}^{n} \hat{u}(x_k, y_k)\varphi_k, \tag{6.4.12}$$

with (x_k, y_k) the nodes of the FEM approximation. Then

$$\|\hat{u} - \hat{u}_h\|_L^2 \leq \|\tilde{u} - \hat{u}\|_L^2. \tag{6.4.13}$$

In other words the error in finite element solution is smaller than the error that we would have if we interpolate the solution by the same set of FEM basis functions, at least measured in the energy norm.
In fact the FEM minimizes $\hat{u} - \hat{u}_h$ in energy norm.

Proof:
Since $v_h \in \Sigma$, (6.4.6) is true for each $v_h \in \Sigma_h$. Subtraction of (6.4.7) from (6.4.6) gives

$$(\hat{u} - \hat{u}_h, v_h)_L = 0, \ \forall v_h \in \Sigma_h. \tag{6.4.14}$$

Now choose

$$v_h = \hat{u} - \hat{u}_h - (\hat{u} - w_h), \text{ with } w_h \text{ arbitrary } \in \Sigma_h. \tag{6.4.15}$$

Then

$$(\hat{u} - \hat{u}_h, \hat{u} - \hat{u}_h)_L = (\hat{u} - \hat{u}_h, \hat{u} - w_h)_L, \ \forall w_h \in \Sigma_h. \tag{6.4.16}$$

Using $ab \leq \frac{1}{2}a^2 + \frac{1}{2}b^2$ it follows that

$$\|\hat{u} - \hat{u}_h\|_L^2 \leq \frac{1}{2}\|\hat{u} - \hat{u}_h\|_L^2 + \frac{1}{2}\|\hat{u} - w_h\|_L^2. \tag{6.4.17}$$

So

$$\|\hat{u} - \hat{u}_h\|_L^2 \leq \|\hat{u} - w_h\|_L^2, \ \forall w_h \in \Sigma_h. \tag{6.4.18}$$

Substitution of $\tilde{u}$ for w_h proves the theorem.

Remark: this error estimate is only true if we use exact integration and also the region is completely identical to the union of all finite elements. In other words when we use linear triangles like in this chapter, the boundary of the region must consist of piecewise straight lines in order that this error estimate holds (i.e. be polygonal).

6.4.3 Approximation errors

By applying the FEM we make a number of errors. First of all we approximate the solution by a polynomial. This produces an approximation error. We might expect that a higher order polynomial reduces this error. We shall return to this subject in Chapter 8.
Besides that we have approximated the region Ω by straight lines. This too introduces an error. Finally the integrals are approximated by a numerical integration rule. Hence another error is made.
It is clear that these errors must be in balance. Each error should be of the same order, thus producing an optimal result. We shall return to this matter in Chapter 8.

6.5 Summary of Chapter 6

The equivalence of a certain class of PDEs and minimization problems has been proven in Chapter 5. A method to approximate the solution of the minimization problem (Ritz) has been derived. The solution is approximated by a finite set of basis functions.
The FEM is a numerical method that constructs the basis functions by subdividing the region into elements and using a simple polynomial approximation per element. In this chapter we have limited ourselves to 1D and 2D linear elements. The most important property of the FEM basis functions is that they are non-zero in a very limited number of elements.
Since all integrals are computed element-wise it is possible to store all contributions per element in an *element matrix* and *element vector* of small size. By using the generic form of these element matrices and vectors it is very simple to construct the large matrix and right-hand side automatically. In order to approximate integrals per element, numerical integration rules are applied. The Newton-Cotes rule derived in Section 6.2.3 is a very attractive rule since it is based on the FEM basis functions.
Essential boundary conditions in the FEM are implemented by direct substitution. Natural boundary conditions require boundary elements or when homogeneous, no special arrangement at all.

Chapter 7

The weak formulation and Galerkin's method

Objectives

Chapter 5 showed that under certain conditions, solving a PDE is equivalent to solving a minimization problem. For an important class of PDEs, for instance those containing a convective term, these conditions are not met. In order to apply the FEM for such problems, it is necessary to have an alternative formulation. This alternative is based on the weak formulation already mentioned in Section 5.6.3. This formulation is applicable for all kinds of PDEs. Usually it is equivalent to the original conservation law used to derive the PDE.

To solve the weak formulation numerically, the Galerkin method is applied. This method is a direct generalization of Ritz. In case an equivalent minimization method exists, Ritz and Galerkin are identical. Since Galerkin is also based on an expansion in basis functions, the FEM is immediately applicable.

As an extension we shall consider the possibility to introduce upwinding in the FEM by using a special variant of Galerkin, the so-called streamline upwind Petrov Galerkin method (SUPG).

7.1 The weak formulation for a symmetrical problem

7.1.1 Introduction

Let us recall the minimization problem (5.3.1) with boundary condition (5.3.2):

$$\min_{u\in\Sigma} I(u) = \int_{\Omega} \{\frac{k}{2}|\nabla u|^2 - uf\}\, d\Omega, \tag{7.1.1}$$

in which the minimization class Σ is defined by $\Sigma = \{u \text{ smooth} \mid u|_\Gamma = 0\}$.

Γ is the complete boundary of Ω. According to (5.3.5) the solution of (7.1.1) must satisfy

$$\int_{\Omega} \{k(\nabla u \cdot \nabla \eta) - \eta f\}\, d\Omega = 0\,, \forall \eta \in \Sigma. \tag{7.1.2}$$

Integration by parts (5.3.7) resulted in

$$\int_{\Omega} \{-\text{div}\,(k\nabla u) - f\}\eta\, d\Omega = 0, \forall \eta \in \Sigma. \tag{7.1.3}$$

And finally in (5.3.8) we arrived at the differential equation:

$$-\text{div}\,(k\nabla u) = f\,, \tag{7.1.4}$$

with boundary condition,

$$u|_\Gamma = 0\,. \tag{7.1.5}$$

In the derivation of the weak formulation we follow the opposite direction. We start with the differential equation (7.1.4), (7.1.5). Next we multiply this equation by an arbitrary function $\eta \in \Sigma$, and we integrate over the domain Ω. This yields exactly formulation (7.1.3). Integration by parts (Gauss' theorem) applied to (7.1.3) results in (7.1.2).

The arbitrary function η is known under the name *test function* and (7.1.2) is called *weak formulation*. Strictly speaking (7.1.3) is also a form of a weak formulation, but in this book we shall limit ourselves to those forms in which by integration by parts the derivatives have been reduced to the lowest order possible.

Note that the form (7.1.2) is symmetric, whereas (7.1.3) is non-symmetric.

There are several reasons to introduce the weak formulation (7.1.2) instead of the differential equation (7.1.4). First of all, it is easier to prove existence and uniqueness of a solution satisfying (7.1.2) than for one satisfying (7.1.4), (7.1.5). It is clear that a solution that satisfies (7.1.4), (7.1.5) is always a solution of (7.1.2). On the other hand a solution of (7.1.2) requires only the existence of the integral over the first derivatives, and it may be possible that the second derivative does not exist at all. For that reason the term *generalized* or *weak* formulation is used.
The second reason to introduce the weak formulation is that it naturally leads to the FEM. Without weak formulation we are not able to derive a FEM for general PDEs. Of course in this specific case there is no need to use a weak formulation, since we can use the minimization problem (7.1.1) and apply Ritz's method.

7.1.2 Natural boundary conditions

In Section 7.1.1 we have seen a simple example with essential boundary conditions only. Let us extend (7.1.4), (7.1.5) to the complete problem treated in Section 5.3.

So we start with the PDE (5.3.8) with boundary conditions (5.3.9) and (5.3.10):

$$-\text{div}\,(k\nabla u) = f\,, \tag{7.1.6}$$

with boundary conditions,

$$u|_{\Gamma_1} = 0\,, \tag{7.1.7}$$

and

$$k\frac{\partial u}{\partial n}|_{\Gamma_2} = 1\,. \tag{7.1.8}$$

In order to derive the weak formulation we use the solution space Σ of functions satisfying the essential boundary conditions (7.1.7): $\Sigma = \{u \text{ smooth } |u|_{\Gamma_1} = 0\ \}$.

Multiplication of (7.1.6) by a test function η and integration over Ω yields:

$$\int_\Omega \{-\text{div}\,(k\nabla u) - f\}\eta\, d\Omega \;=\; 0, \quad \eta \in \Sigma. \tag{7.1.9}$$

Gauss' theorem applied to (7.1.9), while substituting (7.1.8) gives

$$\int_\Omega \{k(\nabla u \cdot \nabla \eta) - \eta f\}\, d\Omega \;-\; \int_{\Gamma_2} \eta\, d\Gamma = \; 0, \quad \forall \eta \in \Sigma, \tag{7.1.10}$$

and this is precisely Equation (5.3.5). So the natural boundary condition (7.1.8) gives rise to a boundary integral in (7.1.10) but does not influence the solution space nor the space of test functions. In fact the natural boundary condition has been applied by replacing $k\frac{\partial u}{\partial n}$ in the boundary integral on Γ_2.

7.1.3 Non-homogeneous essential boundary conditions

The case of non-homogeneous essential boundary conditions has been considered in Chapter 5. For a minimization problem there is no difficulty in applying such a boundary condition. Already in the one-dimensional example of Sections 5.1.1 and 5.1.2, we have seen that the solution u must satisfy the non-homogeneous essential boundary condition, but that the test function $\eta(\mathbf{x})$ must satisfy a homogeneous essential boundary condition.

The derivation of the minimization problem from a PDE with homogeneous boundary conditions was much more complicated (see Section 5.8.3), but the final result is simple.

From these observations it is logical to derive the weak formulation corresponding to the differential equation (7.1.6) with inhomogeneous boundary conditions by demanding that the test functions satisfy the homogeneous essential boundary conditions.

Consider the PDE (7.1.11) with boundary conditions (7.1.12), (7.1.13),

$$-\text{div}\,(k\nabla u) \;=\; f, \tag{7.1.11}$$

$$u|_{\Gamma_1} \;=\; g_1(\mathbf{x}), \tag{7.1.12}$$

$$\sigma u + k\frac{\partial u}{\partial n}|_{\Gamma_2} \;=\; g_2(\mathbf{x}), \quad \sigma \geq 0. \tag{7.1.13}$$

In order to get the weak formulation we multiply (7.1.11) by a test function $\eta(\mathbf{x}) \in \Sigma = \{\eta \mid \eta|_{\Gamma_1} \;=\; 0\}$, and integrate over Ω.

$$\int_\Omega \eta\{-\text{div}\,(k\nabla u) - f\}\, d\Omega \;=\; 0. \tag{7.1.14}$$

Gauss' theorem gives:

$$\int_\Omega (k\nabla u \cdot \nabla\eta - f\eta)\, d\Omega - \int_\Gamma k\frac{\partial u}{\partial n}\eta\, d\Gamma \;=\; 0. \tag{7.1.15}$$

Since $\eta|_{\Gamma_1} = 0$ and $k\frac{\partial u}{\partial n} \;=\; g_2 - \sigma u$, (7.1.15) can be written as:

$$\int_\Omega k\nabla u \cdot \nabla\eta\, d\Omega + \int_{\Gamma_2} \sigma u\eta\, d\Gamma \;=\; \int_\Omega f\eta\, d\Omega + \int_{\Gamma_2} g_2\eta\, d\Gamma \quad \forall\eta \in \Sigma \tag{7.1.16}$$

and

$$u|_{\Gamma_1} = \; g_1. \tag{7.1.17}$$

(7.1.16), (7.1.17) form our weak formulation.

7.1.4 Periodical boundary conditions

Consider the PDE (7.1.11) with boundary conditions (7.1.12) and periodical boundary conditions on the opposite boundaries Γ_2 and Γ_3

$$-\text{div}\,(k\nabla u) \;=\; f, \tag{7.1.18}$$

$$u|_{\Gamma_1} \;=\; g_1(\mathbf{x}), \tag{7.1.19}$$

$$u|_{\Gamma_2} = u|_{\Gamma_3}, \;\; \frac{\partial u}{\partial n}|_{\Gamma_2} = -\frac{\partial u}{\partial n}|_{\Gamma_3}. \tag{7.1.20}$$

Exercise 7.1.1 *Explain the minus sign in 7.1.20* □

In order to get the weak formulation we multiply (7.1.11) by a test function $\eta(\mathbf{x}) \in \Sigma = \{\eta \mid \eta|_{\Gamma_1} \;=\; 0\}$, and integrate over Ω.

$$\int_\Omega \eta\{-\text{div}\,(k\nabla u) - f\}\; d\Omega \;=\; 0. \tag{7.1.21}$$

Gauss' theorem gives:

$$\int_\Omega (k\nabla u \cdot \nabla\eta - f\eta)\; d\Omega - \int_\Gamma k\frac{\partial u}{\partial n}\eta\; d\Gamma \;=\; 0. \tag{7.1.22}$$

Application of the boundary conditions 7.1.19 and 7.1.20 gives

$$\int_\Omega k\nabla u \cdot \nabla\eta\; d\Omega = \int_\Omega f\eta\; d\Omega \;\;\; \forall\eta \in \Sigma \tag{7.1.23}$$

and

$$u|_{\Gamma_1} = \; g_1, \;\; u|_{\Gamma_2} = u|_{\Gamma_3}. \tag{7.1.24}$$

(7.1.23), (7.1.24) form our weak formulation.

Exercise 7.1.2 *Prove 7.1.23, 7.1.24.* □

The extension to non-symmetric problems is straight-forward as will be shown in Section 7.2. The examples in this section, however, already show that the weak formulation is much easier than deriving the corresponding minimization problem if it exists.

7.2 The weak formulation for a non-symmetric problem

As a generalization of the preceding theory we consider the convection-diffusion equation in two space dimensions:

$$-\text{div}\,(\kappa\nabla T) + \rho c_p(\mathbf{u} \cdot \nabla T) + cT \;=\; f. \tag{7.2.1}$$

T is the temperature, κ the heat conduction, ρc_p the heat capacity, c some non-negative constant and f a source term.

We assume that the boundary Γ is subdivided into three parts Γ_1, Γ_2 and Γ_3.

On Γ_1 we prescribe the essential boundary condition

$$T|_{\Gamma_1} \;=\; g_1(\mathbf{x}). \tag{7.2.2}$$

On Γ_2 the flux is given

$$\kappa \frac{\partial T}{\partial n}|_{\Gamma_2} = g_2(\mathbf{x}). \tag{7.2.3}$$

Finally on Γ_3 we assume a mixed boundary condition

$$\sigma T + \kappa \frac{\partial T}{\partial n}|_{\Gamma_3} = g_3(\mathbf{x}), \quad \sigma \geq 0. \tag{7.2.4}$$

In order to derive the weak formulation we proceed as in the symmetrical case. Equation (7.2.1) is multiplied by a test function η satisfying the homogeneous essential boundary condition $\eta|_{\Gamma_1} = 0$ and integrated over the domain Ω. This results in

$$\int_\Omega \{-\text{div}\,(\kappa \nabla T) + \rho c_p(\mathbf{u} \cdot \nabla T) + cT - f\}\eta \, d\Omega = 0. \tag{7.2.5}$$

Now we apply Gauss' theorem, but only on the second derivative. Application to the first order term would not result in lower order derivatives, since the first derivative of the temperature would be replaced by a first derivative of the test function.

$$\int_\Omega \kappa(\nabla T \cdot \nabla \eta) + \{\rho c_p(\mathbf{u} \cdot \nabla T) + cT - f\}\eta \, d\Omega - \int_\Gamma \kappa \frac{\partial T}{\partial n}\eta \, d\Gamma = 0. \tag{7.2.6}$$

Substituting the boundary conditions (7.2.3) and (7.2.4) as well as the essential boundary condition for the test function:

$$\eta|_{\Gamma_1} = 0. \tag{7.2.7}$$

leads to

$$\int_\Omega \kappa(\nabla T \cdot \nabla \eta) + \rho c_p(\mathbf{u} \cdot \nabla T)\eta + cT\eta \, d\Omega + \int_{\Gamma_3} \sigma T \eta \, d\Gamma = \int_\Omega f\eta \, d\Omega + \int_{\Gamma_2} g_2 \eta \, d\Gamma + \int_{\Gamma_3} g_3 \eta \, d\Gamma. \tag{7.2.8}$$

(7.2.8) together with the boundary conditions (7.2.2) and (7.2.7) forms the weak formulation of Equations (7.2.1) to (7.2.4).

We see that the highest derivative in (7.2.8) is of first order which means that it is sufficient to require that the integrals over the first derivatives exist. Strict mathematically speaking the integral over the square of the first derivatives must exist.

If we suppose the existence of a function $T_1(x)$ which is smooth enough and satisfies the boundary condition (7.2.2), then the weak formulation can be stated as:

> Find T such that $T - T_1 \in \Sigma$ and (7.2.8) is satisfied $\forall \eta \in \Sigma$, with Σ the space of sufficiently smooth functions that satisfy (7.2.7).

7.3 Galerkin's method

7.3.1 Introduction

In Section (6.1) we have introduced Ritz's method as a numerical procedure to solve the minimization problem. The idea was based on the approximation of the unknown solution by a finite linear combination of basis functions:

$$u^n(\mathbf{x}) = \sum_{j=1}^{n} a_j \varphi_j(\mathbf{x}), \tag{7.3.1}$$

and to substitute this in the minimization problem. Minimizing over the set of unknown parameters a_j resulted in a system of linear equations to be solved.

Before considering the general convection-diffusion equation, we start with the symmetrical problem (7.1.4-7.1.5). The weak formulation is given in (7.1.2):

$$\int_\Omega \{k(\nabla u \cdot \nabla \eta) - \eta f\}\, d\Omega = 0\,, \forall \eta \in \Sigma. \tag{7.3.2}$$

Substitution of (7.3.1) in (7.3.2) gives

$$\int_\Omega \{k(\nabla u^n \cdot \nabla \eta) - \eta f\}\, d\Omega = 0\,, \forall \eta \in \Sigma. \tag{7.3.3}$$

(7.3.3) contains n unknown parameters a_j. So for a unique solution we need n equations. Since η is in the same space as u it is natural to demand that η is a linear combination of the n basis functions $\varphi_j(\mathbf{x})$:

$$\eta = \sum_{i=1}^{n} b_i \varphi_i(\mathbf{x}). \tag{7.3.4}$$

η is arbitrary, hence a natural choice is to make one of the coefficients b_i equal to 1 and all others to 0. If i runs from 1 to n this results in exactly n linear equations

$$\sum_{j=1}^{n} a_j \int_\Omega \{k(\nabla \varphi_j \cdot \nabla \varphi_i)\, d\Omega = \int_\Omega f \varphi_i\, d\Omega \quad (i = 1, ..., n). \tag{7.3.5}$$

This is identical to using (7.3.4) for each b_i. Why?

Mark that (7.3.5) is precisely the set of Ritz equation corresponding to the PDE (7.1.1). This method, which is in fact a generalization of Ritz is called Galerkin's method.

Summarizing, the method consists of the following steps:

- Derive the weak formulation corresponding to the PDE.
- Approximate the solution by a linear combination of basis functions.
- Replace the test function by each of the basis function separately.

In mathematical terms we may say that we are solving the weak formulation in the function space Σ, which is expanded by an infinite number of basis functions. In Galerkin's method we are looking for a solution in a finite dimensional subspace of Σ.

7.3.2 Galerkin's method applied to the convection-diffusion equation

The extension of Galerkin's method to more general problems like for example the convection-diffusion equation is straightforward. First we have to derive the weak formulation. For the convection-diffusion Equation (7.2.1) with boundary conditions (7.2.2) to (7.2.4), the weak formulation is given in (7.2.8):

$$\int_\Omega \kappa(\nabla T \cdot \nabla \eta) + \rho c_p(\mathbf{u} \cdot \nabla T)\eta + cT\eta\, d\Omega + \int_{\Gamma_3} \sigma T \eta\, d\Gamma = \int_\Omega f\eta\, d\Omega + \int_{\Gamma_2} g_2 \eta\, d\Gamma + \int_{\Gamma_3} g_3 \eta\, d\Gamma. \tag{7.3.6}$$

The next step is to approximate T by T^n:

$$T^n = \sum_{j=1}^{n+n_b} T_j \varphi_j(\mathbf{x}), \tag{7.3.7}$$

where n_b refers to the prescribed (essential) boundary conditions, and to substitute $\eta = \varphi_i(\mathbf{x})$ for i from 1 to n. This yields the following system of equations:

$$\sum_{j=1}^{n+n_b} T_j \{ \int_\Omega \kappa(\nabla\varphi_j \cdot \nabla\varphi_i) + \rho c_p(\mathbf{u}\cdot\nabla\varphi_j)\varphi_i + c\varphi_j\varphi_i \, d\Omega + \int_{\Gamma_3} \sigma\varphi_j\varphi_i \, d\Gamma \} =$$

$$\int_\Omega f\varphi_i \, d\Omega + \int_{\Gamma_2} g_2\varphi_i \, d\Gamma + \int_{\Gamma_3} g_3\varphi_i \, d\Gamma, \quad i = 1, \ldots, n. \tag{7.3.8}$$

In matrix-vector notation this can be written as $\mathbf{ST} = \mathbf{F}$.

Exercise 7.3.1
Give the elements of the matrix $\mathbf{S}$ *and the right-hand side vector* $\mathbf{F}$.

Why do we have to use j *in the summation (7.3.7) and* i *for the test function and not vice versa?* □

7.3.3 The convection-diffusion equation in $\mathbb{R}^1$ by finite elements

Once the Galerkin equations are derived, we can apply the finite element method since the FEM is just a tool to construct basis functions. In this section we shall limit ourselves to the 1D convection-diffusion equation:

$$-\frac{d}{dx}\kappa\frac{dT}{dx} + \rho c_p u\frac{dT}{dx} = f, \tag{7.3.9}$$

with boundary conditions

$$\begin{aligned} T(0) &= T_0, \\ \kappa\tfrac{dT}{dx}(1) &= 0. \end{aligned} \tag{7.3.10}$$

The weak formulation corresponding to Equation (7.3.9) with boundary conditions (7.3.10) is given by

$$\int_0^1 (\kappa\frac{dT}{dx}\frac{d\eta}{dx} + \rho c_p u\frac{dT}{dx}\eta) \, dx = \int_0^1 f\eta \, dx \tag{7.3.11}$$

with $T(0) = T_0$ and $\eta(0) = 0$.

Hence the Galerkin equations corresponding to the weak formulation (7.3.11) are given by

$$\sum_{j=0}^{n} T_j \int_0^1 (\kappa\frac{d\varphi_j}{dx}\frac{d\varphi_i}{dx} + \rho c_p u\frac{d\varphi_j}{dx}\varphi_i) \, dx = \int_0^1 f\varphi_i \, dx, \quad i = 1, \ldots, n. \tag{7.3.12}$$

In order to apply the finite element method we use the linear basis functions defined in (6.2.10). Again we can introduce finite element matrices and vectors to store the contribution for each element.

Example 7.3.1

Let $\rho c_p = 1$ and κ and u be constant.
Then the element matrix corresponding to (7.3.12) is given by

$$S^{e_k} = \frac{\kappa}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \frac{u}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \tag{7.3.13}$$

with h the length of the element.

For constant values of h the large matrix corresponding to (7.3.4) is given by

$$\frac{\kappa}{h}\begin{bmatrix} 2 & -1 & & & & & \\ -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & & \ddots & & & \\ & & & & -1 & 2 & -1 \\ & & & & & -1 & 1 \end{bmatrix} + \frac{u}{2}\begin{bmatrix} 0 & 1 & & & & & \\ -1 & 0 & 1 & & & & \\ & -1 & 0 & 1 & & & \\ & & & \ddots & & & \\ & & & & -1 & 0 & 1 \\ & & & & & -1 & 1 \end{bmatrix}. \tag{7.3.14}$$

Compare this result with a central finite difference scheme. (Choose end point $x = 1$ between two nodes of the grid).

So from (7.3.14) we may conclude that discretization of the convection term by Galerkin finite elements results in a scheme which is very similar to central differences. How to upwind in case of finite elements will be the subject of Section (7.4.2).

Exercise 7.3.2 *Prove Equation (7.3.11)* □

Exercise 7.3.3 *Prove Equation (7.3.12)* □

Exercise 7.3.4 *Prove Equation (7.3.13)* □

Exercise 7.3.5 *Prove Equation (7.3.14)* □

7.3.4 The convection-diffusion equation in $\mathbb{R}^2$ by finite elements

Consider the convection-diffusion equation (7.2.1) with boundary conditions (7.2.2) to (7.2.4). The weak formulation is given in (7.2.8) and the corresponding Galerkin equations in (7.3.8). For the application of the FEM we subdivide the domain in triangles and use the linear basis functions defined in (6.3.8). The boundary integrals in (7.3.8) are approximated by line elements in exactly the same way as in Section 6.3.4. The element vector for the internal element (triangle) is of course the same as in Formula (6.3.17). Only the element matrix for the triangle is different from that in Section 6.3.1.

Exercise 7.3.6
Show that the element matrix S^{e_k} for the triangle corresponding to (7.3.8) is a 3×3 matrix with elements s_{ij} given by

$$s_{ij} = \int_{e_k} \kappa(\nabla\varphi_j \cdot \nabla\varphi_i) + \rho c_p(\mathbf{u} \cdot \nabla\varphi_j)\varphi_i + c\varphi_i\varphi_j \, d\Omega. \tag{7.3.15}$$

□

Exercise 7.3.7 *Let κ and c be constant scalars, ρc_p be equal to 1 and* $\mathbf{u}$ *be a vector with constant components. Express the elements of the element matrix S^{e_k} in terms of Δ (6.3.11) and the coefficients β_i and γ_i given in (6.3.13).*
Hint: use the Newton Cotes formula to approximate the integrals. □

7.4 Petrov-Galerkin

7.4.1 Introduction

In Section 7.3 we have introduced the Galerkin method. Galerkin is based on the discretization of the weak formulation, where the solution space and the test space are identical. This approach is very common in finite elements and it has many advantages. However, it is not necessary that solution space and space of test functions are the same. In the literature one can find for example a method called collocation, where the test functions are in fact delta functions. This means that one satisfies the differential equation point-wise in the nodal points. In that case it is of course necessary to require more smoothness of the approximation to the solution since integration by parts, to reduce the order of the weak formulation, is no longer possible. For that reason this approach has never become very popular. However, sometimes one uses test functions that do not have the same shape as the basis functions, without affecting the continuity requirements. So starting point is the same weak formulation as for Galerkin. Such methods are for example applied to stabilize the numerical solution. Methods in which the test functions and the basis functions for the solution have different shapes are called Petrov-Galerkin methods.

A typical application in which Petrov-Galerkin methods are used, is convection dominated flow. In finite difference methods it is necessary to use upwind schemes to stabilize the solution, in finite elements Petrov-Galerkin plays the same role.

In the remainder of this chapter we shall use SGA for the Standard Galerkin Approach and SUPG for Petrov-Galerkin. The letters SU will be explained in Section 7.4.3.

7.4.2 Upwinding in $\mathbb{R}^1$ by Petrov-Galerkin

In Section 3.3.2.2 we introduced upwind differencing using the artificial model problem:

$$-\varepsilon\frac{d^2c}{dx^2} + u\frac{dc}{dx} = 0, \tag{7.4.1}$$

with boundary conditions

$$c(0) = 0, \quad c(1) = 1. \tag{7.4.2}$$

Figure 3.8 shows the exact solution for $\varepsilon = 0.01$ and $u = 1$. If we use SGA as in Section 7.3.3 we get a scheme that is almost identical to a central difference scheme (see Exercise 7.3.3). So one may expect the same behavior as for central differences. Figure 7.1 shows that this is indeed the case.

The central difference scheme for Equation (7.4.1) with boundary conditions (7.4.2) is given by:

$$-\varepsilon\frac{c_{i+1} - 2c_i + c_{i-1}}{h^2} + u\frac{c_{i+1} - c_{i-1}}{2h} = 0, \quad i = 1, ..., n. \tag{7.4.3}$$

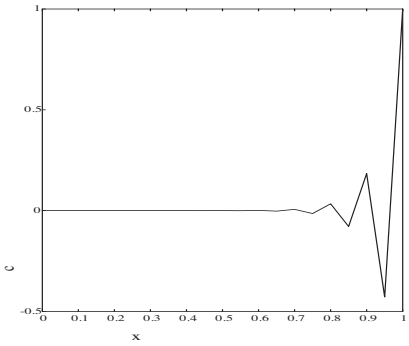


Figure 7.1: Solution of 1d convection-diffusion equation by SGA.

To avoid the unrealistic wiggles we use upwind to approximate the convective terms. The most simple one is first order upwind, where the convection term is discretized by a backward difference scheme for positive velocity u.
The first order upwind scheme for Equation (7.4.1) reads ($u > 0$)

$$-\varepsilon\frac{c_{i+1} - 2c_i + c_{i-1}}{h^2} + u\frac{c_i - c_{i-1}}{h} = 0, \quad i = 1, ..., n. \tag{7.4.4}$$

The local truncation error for Equation (7.4.4) is equal to

$$-\frac{uh}{2}\frac{d^2c}{dx^2} + O(h^2). \tag{7.4.5}$$

Exercise 7.4.1 *Prove (7.4.5)* □

(7.4.5) shows that first order upwind in fact introduces an artificial diffusion of $\varepsilon + \frac{uh}{2}$. There are many other upwind schemes that in fact introduce an artificial diffusion in some clever way.

This observation has inspired FEM researchers to simulate this behavior by a suitable choice of the test functions. To derive the weak formulation for Equation (7.4.1) we multiply by a test function η:

$$\int_0^1 (-\varepsilon\frac{d^2c}{dx^2} + u\frac{dc}{dx})\eta \, dx = 0. \tag{7.4.6}$$

Now the trick is to split $\eta(x)$ into two parts $w(x)$ and $p(x)$ ($\eta(x) = w(x) + p(x)$), where $w(x)$ is the classical test function from the same space as the solution and $p(x)$ is used to take care of the upwind behavior. The $w(x)$ part ensures the consistency of the scheme. This function must be so smooth that integration by parts is allowed. $p(x)$ on the other hand will be defined elementwise, which means that

it may be discontinuous over the element boundaries. In this way Equation (7.4.6) is written as

$$\int_0^1 (\varepsilon \frac{dc}{dx}\frac{dw}{dx} + u\frac{dc}{dx}w)\,dx + \int_0^1 (-\varepsilon\frac{d^2c}{dx^2} + u\frac{dc}{dx})p\,dx = 0. \tag{7.4.7}$$

The last integral is replaced by the sum over the elements, so in fact contributions over the element boundaries are neglected. Hence

$$\int_0^1 (\varepsilon \frac{dc}{dx}\frac{dw}{dx} + u\frac{dc}{dx}w)\,dx + \sum_{elements}\int_{element} (-\varepsilon\frac{d^2c}{dx^2} + u\frac{dc}{dx})p\,dx = 0. \tag{7.4.8}$$

Note that the last term is just a correction to the standard SGA equations. This correction goes to zero if c approaches the exact solution. Of course the choice of p is essential for the behavior of the scheme. Since we are dealing with linear basis functions, the term $\varepsilon\frac{d^2c}{dx^2}$ is zero per element. So the extra term reduces to

$$\sum_{elements}\int_{element} (u\frac{dc}{dx})p\,dx. \tag{7.4.9}$$

Now we choose $p(x)$ such that we get an artificial diffusion of the size $\frac{uh}{2}$.

Exercise 7.4.2
Show that if we choose $p(x) = \frac{h}{2}\frac{d\varphi_i}{dx}$ per element, the artificial diffusion is equal to $\frac{uh}{2}$.

Hint compare Expression (7.4.9) with the discretization of the diffusion term. □

In practice one chooses $p(x) = \frac{h\xi}{2}\frac{d\varphi_i}{dx}$, with ξ some parameter depending on the ratio of u and ε.

ξ equal to sign(u) corresponds to the classical upwind scheme. Popular choices for ξ can be found for example in Brooks and Hughes [6].

7.4.3 SUPG: stream line upwinding in $\mathbb{R}^2$ by Petrov-Galerkin

To extend the upwind Petrov-Galerkin method to 2D one might consider to use the same approach as in Section (7.4.2). An alternative is to apply the upwind technique of Section (7.4.2) to both coordinate directions. However, both approaches have the disadvantage that we have a diffusion term in all directions. The wiggles we get are due to convection, so it makes sense to apply upwind only in the direction of the flow. So a natural choice for upwind in more dimensions is to apply the one-dimensional upwind in the velocity direction. Brooks and Hughes [6] achieved this by replacing the term $p(x) = \frac{h\xi}{2}\frac{d\varphi_i}{dx}$ by $p(\mathbf{x}) = \frac{h\xi}{2}\frac{\nabla\varphi_i\cdot\mathbf{u}}{||\mathbf{u}||}$. This means that the x-derivative of the basis function in the one-dimensional problem is replaced by the directional derivative of the basis function in the direction of the velocity. For h one takes some representative distance in the element, preferably in the direction of $\mathbf{u}$. Since streamlines are always in the direction of the velocity this method is commonly called the Streamline Upwind Petrov Galerkin method (SUPG).

To show the difference between SGA and SUPG we consider the following benchmark problem.

Rotating cone problem

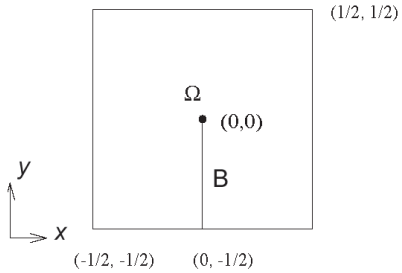


Figure 7.2: Definition region for rotating cone problem.

Consider the region Ω sketched in Figure 7.2.
The region consists of a square with a cut B. In the inner region we suppose that the concentration satisfies the convection-diffusion equation

$$-\varepsilon \triangle c + \mathbf{u} \cdot \nabla c = 0 \,, \tag{7.4.10}$$

where ε is chosen equal to 10^{-6} and the velocity $\mathbf{u}$ is such that the flow rotates counter clockwise. This is achieved by setting $\mathbf{u} = \begin{bmatrix} -y \\ x \end{bmatrix}$. At the outer boundary we use the boundary condition

$$c|_\Gamma = 0 \,. \tag{7.4.11}$$

On the starting curve B the concentration c is set equal to

$$c|_B = \cos(2\pi(y + \frac{1}{4})) \,, \tag{7.4.12}$$

and due to the small diffusion one expects that the concentration at the end curve is nearly the same. The end curve has the same co-ordinates as B but the nodal points differ, which means that the solution may be different from the starting one. Since no boundary condition is given at the outflow curve "B" implicitly the boundary condition

$$\varepsilon \frac{\partial c}{\partial n}|_B = 0 \,, \tag{7.4.13}$$

is prescribed. (Why?)

If we make contour lines (i.e. lines with equal values) of the concentration, we expect concentric circles. However, if we subdivide the region into 20×20 squares, each of which is subdivided into two triangles by drawing the diagonal in arbitrary direction we get a very irregular set of contour lines as can be seen in Figure 7.3.
If we solve the same problem with SUPG the result is much smoother as is shown in Figure 7.4. The fact that circles in this picture are not completely closed is due to the numerical diffusion.

7.5 An example of a system of coupled PDEs

We finish this chapter with the plane stress example of Section 5.4.3. This problem can be formulated in terms of a minimization problem, see Equation (5.4.5), but in this case we shall use the weak formulation.



Figure 7.3: Equi-concentration lines for rotating cone problem computed by SGA.

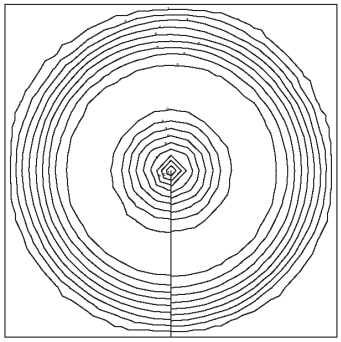


Figure 7.4: Equi-concentration lines for rotating cone problem computed by SUPG.

In Exercise 5.7.3 we have derived that the displacements (u, v) satisfy the equation

$$\begin{aligned} -\frac{\partial \sigma_{xx}}{\partial x} - \frac{\partial \tau_{xy}}{\partial y} &= 0\,, \\ -\frac{\partial \tau_{xy}}{\partial x} - \frac{\partial \sigma_{yy}}{\partial y} &= 0\,. \end{aligned} \tag{7.5.1}$$

Exercise 7.5.1
Show with the information of Section 5.4.3 that the boundary conditions for Equation (7.5.1) are given by:

$u = v = 0$ *on* Γ_1
$\sigma_{xx} n_x + \tau_{xy} n_y = t_1$, $\tau_{xy} n_x + \sigma_{yy} n_y = t_2$ *on* Γ_2
$\sigma_{xx} n_x + \tau_{xy} n_y = 0$, $\tau_{xy} n_x + \sigma_{yy} n_y = 0$ *on all other boundaries,*

with (n_x, n_y) *the normal on the boundary.* □

To derive the weak formulation we multiply the first equation of 7.5.1 by a test function δu and the second equation by δv and integrate over the domain Ω.

Exercise 7.5.2 *Show using the divergence theorem, that the weak formulation corresponding to Equation (7.5.1) with the boundary conditions given in Exercise 7.5.1 can be written*

as:

$$\int_\Omega \{\sigma_{xx}\frac{\partial \delta u}{\partial x} + \tau_{xy}\frac{\partial \delta u}{\partial y}\}\, d\Omega = \int_{\Gamma_2} t_1 \delta u\, d\Gamma, \tag{7.5.2}$$
$$\int_\Omega \{\tau_{xy}\frac{\partial \delta v}{\partial x} + \sigma_{yy}\frac{\partial \delta v}{\partial y}\}\, d\Omega = \int_{\Gamma_2} t_2 \delta v\, d\Gamma.$$

□

The next step is to apply the Galerkin method. We approximate u and v by a linear combination of basis functions

$u^n = \sum_{j=1}^{n} u_j \varphi_j(\mathbf{x}), v^n = \sum_{j=1}^{n} v_j \varphi_j(\mathbf{x}).$
The same basis functions for u and v are used. δu is replaced by $\varphi_i(\mathbf{x})$ for i in Σ_1, the set of non-prescribed u-velocity components and δv by $\varphi_i(\mathbf{x})$ for i in Σ_2, the set of non-prescribed v-velocity components.

In this particular example Σ_1 and Σ_2 are identical.

Exercise 7.5.3 *Show that the system of Galerkin equations corresponding to the weak formulation (7.5.2) is given by*

$$\sum_{j=1}^{n} u_j \int_\Omega \{A\frac{\partial \varphi_j}{\partial x}\frac{\partial \varphi_i}{\partial x} + B\frac{\partial \varphi_j}{\partial y}\frac{\partial \varphi_i}{\partial y}\}\, d\Omega + \sum_{j=1}^{n} v_j \int_\Omega \{\nu A\frac{\partial \varphi_j}{\partial y}\frac{\partial \varphi_i}{\partial x} + B\frac{\partial \varphi_j}{\partial x}\frac{\partial \varphi_i}{\partial y}\}\, d\Omega$$
$$= \int_{\Gamma_2} t_1 \varphi_i\, d\Gamma,\ i \in \Sigma_1 \tag{7.5.3}$$
$$\sum_{j=1}^{n} u_j \int_\Omega \{B\frac{\partial \varphi_j}{\partial y}\frac{\partial \varphi_i}{\partial x} + \nu A\frac{\partial \varphi_j}{\partial x}\frac{\partial \varphi_i}{\partial y}\}\, d\Omega + \sum_{j=1}^{n} v_j \int_\Omega \{B\frac{\partial \varphi_j}{\partial x}\frac{\partial \varphi_i}{\partial x} + A\frac{\partial \varphi_j}{\partial y}\frac{\partial \varphi_i}{\partial y}\}\, d\Omega$$
$$= \int_{\Gamma_2} t_2 \varphi_i\, d\Gamma,\ i \in \Sigma_2$$

□

To apply the FEM, the region Ω is subdivided into triangular elements. The same linear basis functions as in Section 6.3.4 are used. In each point there are two unknowns so the size of the element matrix must be 6×6 and the element vector has 6 entries.

Suppose we order the triangle unknowns u_1, u_2, u_3, v_1, v_2, v_3 with 1, 2 and 3 the local node numbers of the triangle. The rows of the element matrix are of course ordered in the same way. Then the element matrix can be split into 4 submatrices:

$$S^{e_k} = \begin{bmatrix} S^{e_k}_{uu} & S^{e_k}_{uv} \\ S^{e_k}_{vu} & S^{e_k}_{vv} \end{bmatrix}. \tag{7.5.4}$$

Exercise 7.5.4 *Compute the elements of the four subelement matrices under the condition that ν and E are constant.* □

7.6 Mathematical theory

Existence and uniqueness of solutions of the weak formulation are in general much more difficult to prove than in case of a minimization problem. However, for a subclass of problems, it is possible to give some general theory.

Consider a linear PDE of order $2m$ of the form

$$\mathbf{Lu} = \mathbf{f}. \tag{7.6.1}$$

The weak formulation can be written as a bilinear form:

$$a(u, v) = (f, v), \tag{7.6.2}$$

where $a(u, v)$ contains only derivatives of order m. This means that we have removed all higher order derivatives by integration by parts. The solution must be found in some Hilbert space (more precisely a Sobolev space H^m) and the test functions are arbitrary functions in that space. Now we can prove the following theorem

Theorem 7.6.1 *Let V_0 be a real Hilbert space with inner product $(.,.)_0$. Let V_1 be a closed subspace of V_0 with inner product $(.,.)_1$. Let $a(u, v)$ be a positive continuous bilinear form mapping $V_1 \times V_1 \to R$, which means*

$$\begin{aligned}
a(u, v + w) &= a(u, v) + a(u, w), && (7.6.3)\\
a(u + w, v) &= a(u, v) + a(w, v), && (7.6.4)\\
a(\lambda u, v) &= a(u, \lambda v) = \lambda a(u, v), && (7.6.5)\\
|a(u, v)| &\leq K\|u\|_1\|v\|_1, && (7.6.6)\\
a(u, u) &\geq \gamma_1\|u\|_1^2, && (7.6.7)\\
\|u\|_1 &\geq \gamma_0\|u\|_0 && (7.6.8)
\end{aligned}$$

(7.6.3) to (7.6.5) imply linearity, (7.6.6) is continuity and (7.6.7) means positiveness. Let f be an element of V_0. Then the weak formulation

$$\textit{find } u \in V_1 \textit{ such that } a(u, v) = (f, v)_0 \ \forall v \in V_1, \tag{7.6.9}$$

has exactly one solution in V_1.

Proof
The Lax-Milgram theorem (see for example [49], page 92) states that, under conditions (7.6.3) to (7.6.7), for each linear functional $F(v)$ on V_1 there is precisely one element $w \in V_1$ such that

$$a(w, v) = F(v), \ \forall v \in V_1. \tag{7.6.10}$$

According to (7.6.8) for a given $f \in V_0$ the inner product $(f, v)_0$ is a bounded linear functional on V_1, since

$$(f, v)_0 \leq \|f\|_0\|v\|_0 \leq \frac{1}{\gamma_0}\|f\|_0\|v\|_1. \tag{7.6.11}$$

This proves the theorem. □

Remark: if we compare (7.6.3) to (7.6.7) with (5.9.5) to (5.9.7), then we see that the main difference is the symmetry requirement. This is necessary for the minimization problem, but not for the existence of the solution of the weak formulation. (7.6.6), (7.6.7) are necessary to guarantee that $a(u, u)^{\frac{1}{2}}$ is an equivalent norm of

$\|.\|_1$. In (5.9.5) to (5.9.7) $a(.,.)$ created itself V_1, and hence $a(u,u)^{\frac{1}{2}}$ was the norm on V_1. In that case(7.6.6), (7.6.7) are satisfied automatically.

In fact now we require that the symmetrical part of $a(.,.)$ is positive definite.

The next theorem gives an error estimate in terms of the "1"-norm, for our finite dimensional approximation.

Theorem 7.6.2 *Let V_0, V_1 and $a(.,.)$ be defined as in Theorem 7.6.1. Let V_{1h} be a finite dimensional subspace of V_1. Let u be the solution of the weak problem*

$$a(u,v) = (f,v)_0 \ \ \forall v \in V_1, \tag{7.6.12}$$

and let u_h be the solution of the finite dimensional problem

$$a(u_h, v_h) = (f, v_h)_0 \ \ \forall v_h \in V_{1h}. \tag{7.6.13}$$

Then we have the following estimate:

$$\|u - u_h\|_1 \leq \frac{K}{\gamma_1} \min_{v_h \in V_{1h}} \|u - v_h\|_1. \tag{7.6.14}$$

Proof
Since V_{1h} is a subspace of V_1

$$a(u, v_h) = (f, v_h)_0 \ \ \forall v_h \in V_{1h}, \tag{7.6.15}$$

and from (7.6.13)

$$a(u - u_h, v_h) = 0 \ \ \forall v_h \in V_{1h}, \tag{7.6.16}$$

From (7.6.7) and (7.6.16) it follows that

$$\gamma_1 \|u - u_h\|^2 \leq a(u - u_h, u - u_h) = a(u - u_h, u). \tag{7.6.17}$$

Because of (7.6.16) we get

$$\gamma_1 \|u - u_h\|^2 \leq a(u - u_h, u - v_h) \ \ \forall v_h \in V_{1h}, \tag{7.6.18}$$

and due to the continuity of a (7.6.6)

$$\gamma_1 \|u - u_h\|^2 \leq K \|u - u_h\| \|u - v_h\| \ \ \forall v_h \in V_{1h}. \tag{7.6.19}$$

So

$$\|u - u_h\| \leq \frac{K}{\gamma_1} \|u - v_h\| \ \ \forall v_h \in V_{1h}. \tag{7.6.20}$$

which proves the theorem. □

Theorem 7.6.2 shows that the error in the solution in energy norm is smaller than a constant times the interpolation error in energy norm. So the error estimate is of the same type as for minimization problems, although the constant is in general larger than 1. So the estimate is not as sharp as it is for a minimization problem.

7.7 Summary of Chapter 7

Instead of deriving a minimization problem corresponding to a PDE, one may also derive a weak formulation by multiplying the PDE by a test function. Integration by parts may be used to get rid of higher order derivatives. Natural boundary conditions are automatically part of the weak formulation due to this integration

by parts. This method can be applied to general PDEs and is therefore much more applicable than the minimization approach.

To approximate the solution, Galerkin's method is applied, which is a direct generalization of the Ritz method. The solution is approximated by a finite set of basis functions and the test function runs through the same set. Application of the FEM to Galerkin is completely identical to the use of FEM in case of Ritz.

By using test functions that are different from the basis functions one gets the Petrov-Galerkin method. A typical application is SUPG, the finite element equivalent of upwind differencing.

Chapter 8

Extension of the FEM

Objectives

In the previous Chapters 6 and 7 we have introduced the FEM as a technique to construct basis functions for Ritz and Galerkin. Until now we have limited ourselves to linear interpolation polynomials in $\mathbb{R}^1$ and $\mathbb{R}^2$. In $\mathbb{R}^2$ this means automatically that we have to use triangles.

In this chapter we shall extend the theory to higher order polynomials. Furthermore we shall show how quadrilaterals can be handled. In each case we have to check whether these elements satisfy the continuity requirements formulated in Chapter 6. This will lead to the *isoparametric transformations*. Finally we shall show that these requirements, in the case of fourth order PDEs, lead to complicated elements. In practice this is solved by reducing the fourth order problem as a set of two second order problems. In this way the continuity requirements may be reduced.

8.1 (Straight) quadratic triangles

One may expect that the linear interpolations we have used so far, lead to errors of the order h^2, provided the solution is smooth enough. If we want a higher order accuracy, it is necessary to use higher order polynomials. In this section we shall derive the basis functions corresponding to quadratic interpolation. Elements using quadratic interpolation polynomials are usually addressed as *quadratic elements*.

In $\mathbb{R}^1$ the situation is simple. A quadratic interpolation polynomial over an element with vertices $x_1$and x_3 and midpoint x_2 can be written as (compare with (6.2.4) - (6.2.5))

$$u(x) = l_1(x)u_1 + l_2(x)u_2 + l_3(x)u_3, \tag{8.1.1}$$

with $l_i(x)$ the quadratic Lagrangian polynomials defined by

$$l_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}, \tag{8.1.2a}$$

$$l_2(x) = \frac{(x - x_3)(x - x_1)}{(x_2 - x_3)(x_2 - x_1)}, \tag{8.1.2b}$$

$$l_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2).} \tag{8.1.2c}$$

Again these polynomials satisfy

$$l_j(x_i) \;=\; \delta_{ij}. \tag{8.1.3}$$

So the basis functions $\varphi_i(x)$ can be defined by

$$\begin{aligned} &\varphi_i(\mathbf{x}) \quad \text{quadratic,} \\ &\varphi_i(\mathbf{x}_j) = \delta_{ij}, \quad i,j,= 1,2,\cdots,n+1\,. \end{aligned} \tag{8.1.4}$$

Figure 8.1 shows a "vertex" basis function and Figure 8.2 a basis function corresponding to the mid point of an element.

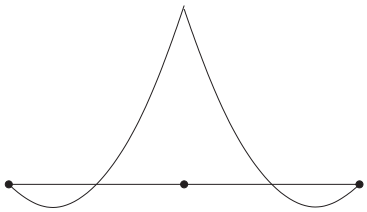


Figure 8.1: Basis function for a vertex.

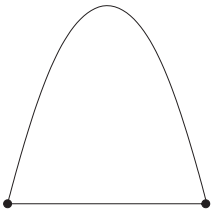


Figure 8.2: Midpoint basis function.

Theorem 8.1.1 *The Newton Cotes integration rule for the quadratic elements in* $\mathbb{R}^1$ *is given by*

$$\int_e f(x)\,dx \;=\; \frac{h}{6}[f(x_1) + 4f(x_2) + f(x_3)], \tag{8.1.5}$$

with h the length of the interval $[x_1, x_3]$ *and* x_2 *the mid point. This is of course Simpson's rule.*

Exercise 8.1.1 *Prove Theorem (8.1.1).* □

Exercise 8.1.2 *Compute the element matrix and element vector for Poisson's equation (6.2.1) using quadratic polynomials. Use the Newton Cotes rule (8.1.5) to compute the integrals.* □

The extension to $\mathbb{R}^2$ is more complex. First we shall limit ourselves to quadratic triangles with straight sides. The extension to curved sides is treated in Section (8.4).

A quadratic polynomial in $\mathbb{R}^2$ is uniquely defined by 6 parameters (why?). In Chapter 6 we have seen that for second order PDEs, it is necessary that the basis functions are continuous. In order to get continuity, it is necessary that the values of the interpolation on the common edge of two adjacent triangles are equal for both triangles. An edge is one dimensional. A quadratic polynomial in $\mathbb{R}^1$ is uniquely defined by 3 parameters, hence we need three nodal points on each edge of the triangle. So it is natural to use vertices and the midside points as nodes of the triangle. See Figure 8.3.

The 6 basis functions on the triangle are implicitly defined by the requirements (8.1.4). If we write the quadratic polynomial $\varphi_i(x)$ as

$$\varphi_i(x) = \alpha_i + \beta_i x + \gamma_i y + \delta_i x^2 + \epsilon_i xy + \eta_i y^2, \tag{8.1.6}$$

the coefficients $\alpha_i, \ldots \eta_i$ can be computed by solving a 6×6 system of linear equations. To do this element by element is relatively expensive. A more subtle approach is to express the basis functions $\varphi_i(\mathbf{x})$ in terms of the linear basis functions

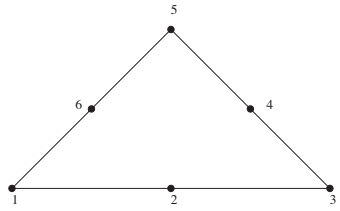


Figure 8.3: Nodes of quadratic triangle.

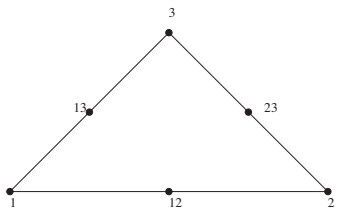


Figure 8.4: Special numbering of nodes.

$\lambda_i(\mathbf{x})$ (6.3.22) corresponding to the vertices of the triangle. In this particular case it is easier to use the local numbering of Figure 8.4 to define the basis functions. The basis functions corresponding to the vertices are denoted by $\psi_i(\mathbf{x})$ and the basis functions corresponding to the midpoints by $\psi_{ij}(\mathbf{x})$. The relation between φ_i and ψ_i is trivial.

First we consider the basis functions $\psi_i(\mathbf{x})$ corresponding to the vertices.

Since $\psi_i(\mathbf{x}_j) = \delta_{ij}$ an obvious choice is to define $\psi_i = \lambda_i v_i$ with v_i a linear function. (Why?).
This function satisfies 3 of the six equations automatically.
From $\psi_i(\mathbf{x}_i) = 1$ it follows that $v_i(\mathbf{x}_i) = 1$.
From $\psi_i(\mathbf{x}_{kl}) = 0$ it follows that $v_i(\mathbf{x}_{kl}) = 0$ for $k = i$ or $l = i$.
Because the points x_{ij} are in the middle of the sides this implies that
$v_i(\mathbf{x}_j) = -1$ and $v_i(\mathbf{x}_k) = -1$.
So v_i can be written as $\lambda_i - \lambda_j - \lambda_k = 2\lambda_i - 1$ (Theorem 6.3.3) $i \neq j,\ i \neq k,\ j \neq k$.

Next consider the mid points $\mathbf{x}_{ij}$.

Since $\psi_{ij}(\mathbf{x}_k) = 0$, $\psi_{ij}(\mathbf{x}_{kl}) = 0$ if $ij \neq kl$ we have $\psi_{ij} = 0$ on the sides not containing point ij. So a natural choice is $\psi_{ij} = \alpha\lambda_i\lambda_j$. From $\psi_{ij}(\mathbf{x}_{ij}) = 1$ it follows that $\alpha = 4$.

In conclusion in each element the quadratic basis functions can be expressed in the linear basis functions by

$$\begin{aligned} \psi_i &= \lambda_i(2\lambda_i - 1), \\ \psi_{ij} &= 4\lambda_i\lambda_j. \end{aligned} \tag{8.1.7}$$

Theorem 8.1.2 *The Newton Cotes formula for the quadratic triangle is given by*

$$\int_e Int(\mathbf{x})\, d\Omega = \frac{|\Delta|}{6}[Int(\mathbf{x}_{12}) + Int(\mathbf{x}_{23}) + Int(\mathbf{x}_{13})]. \tag{8.1.8}$$

Exercise 8.1.3 *Prove Theorem (8.1.2).* □

Exercise 8.1.4 *Compute the element matrix and vector corresponding to the Poisson equation (6.3.1) for a quadratic triangle. Use the Newton Cotes rule (8.1.8).* □

8.2 Linear triangles revisited

In Section 6.3.2 we have introduced linear triangles and computed the basis functions by direct solution of the system of equations (6.3.10). An alternative approach is to map the triangle in (x, y)-space on a so-called standard triangle in (ξ, η)-space, with coordinates (0,0), (1,0) and (0,1). Although there is no need to do so for the

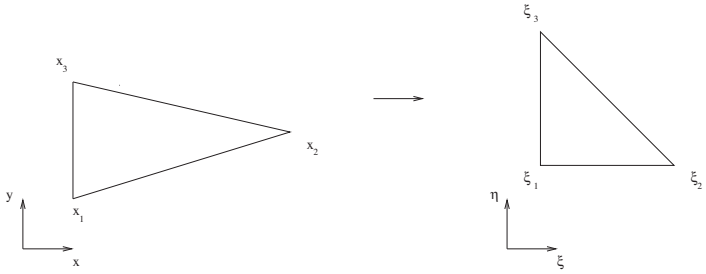


Figure 8.5: General triangle Standard triangle

linear triangle, we shall follow this approach here, since mapping is necessary for a number of elements that will be treated later on.
Consider the standard and general triangle in Figure 8.5. The basis functions on the standard triangle must satisfy $\phi_i(\xi_j, \eta_j) = \delta_{ij}$ and one immediately verifies that they are given by

$$\begin{aligned} \phi_1(\xi,\eta) &= 1-\xi-\eta, \\ \phi_2(\xi,\eta) &= \xi, \\ \phi_3(\xi,\eta) &= \eta. \end{aligned} \tag{8.2.1}$$

The linear transformation from the general triangle to the standard triangle is given by

$$\mathbf{x}_1 \rightarrow (0,0),\ \mathbf{x}_2 \rightarrow (1,0),\ \mathbf{x}_3 \rightarrow (0,1), \tag{8.2.2}$$

hence

$$\begin{aligned} x &= x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta, \\ y &= y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta. \end{aligned} \tag{8.2.3}$$

In order that the transformation is applicable it must be invertible. So the Jacobian of the transformation must be non-singular for each $\mathbf{x}$ in the triangle.

The Jacobian matrix $\mathbf{J}$ is defined by

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix}. \tag{8.2.4}$$

Theorem 8.2.1 *The determinant of* $\mathbf{J}$ *is equal to:*

$$det(\mathbf{J}) = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1), \tag{8.2.5}$$

and this is precisely the parameter Δ in (6.3.11).
Integration of some function $f(x,y)$ over the general triangle can be simplified by transformation to the standard triangle:

$$\int_{e_{xy}} f(x,y)\, d\Omega_{xy} = \int_{e_{\xi\eta}} f(\xi,\eta)|det(J)|\, d\Omega_{\xi\eta}. \tag{8.2.6}$$

Exercise 8.2.1 *Prove that* $|\Delta|$ *is twice the area of the original triangle.*
Hint: use (8.2.6) *with* $f = 1$. □

Since $|\Delta|$ is only zero if the triangle has area zero, the transformation can not be singular.
To show how this transformation can be utilized to compute an element matrix or vector, we consider the simple example of the Poisson equation (6.2.1).
The element matrix has elements $s_{ij} = \int_e \nabla\varphi_i(\mathbf{x}) \cdot \nabla\varphi_j(\mathbf{x})\, d\Omega$. We transform this integral to an integral in the (ξ, η)-plane. Hence:

$$s_{ij} = \int_{e_{xy}} \nabla\varphi_i(\mathbf{x}) \cdot \nabla\varphi_j(\mathbf{x})\, dxdy = \int_{e_{\xi\eta}} \nabla\varphi_i \cdot \nabla\varphi_j |det(J)| d\xi\, d\eta. \tag{8.2.7}$$

Since $|det(J)|$ is constant the integral reduces to

$$s_{ij} = |det(J)| \int_e (\nabla\varphi_i \cdot \nabla\varphi_j) d\xi\, d\eta. \tag{8.2.8}$$

To compute the values of $\nabla\varphi_i$ we express the derivatives to x and y into derivatives of ξ and η:

$$\begin{aligned} \frac{\partial\varphi_k}{\partial x} &= \frac{\partial\varphi_k}{\partial\xi}\frac{\partial\xi}{\partial x} + \frac{\partial\varphi_k}{\partial\eta}\frac{\partial\eta}{\partial x}, \\ \frac{\partial\varphi_k}{\partial y} &= \frac{\partial\varphi_k}{\partial\xi}\frac{\partial\xi}{\partial y} + \frac{\partial\varphi_k}{\partial\eta}\frac{\partial\eta}{\partial y}. \end{aligned} \tag{8.2.9}$$

To compute these derivatives, we need the values of $\frac{\partial\xi}{\partial x}$ and so on.

Theorem 8.2.2 *The matrix*

$$\begin{bmatrix} \frac{\partial\xi}{\partial x} & \frac{\partial\xi}{\partial y} \\ \frac{\partial\eta}{\partial x} & \frac{\partial\eta}{\partial y} \end{bmatrix}, \tag{8.2.10}$$

is the inverse of the matrix

$$\begin{bmatrix} \frac{\partial x}{\partial\xi} & \frac{\partial x}{\partial\eta} \\ \frac{\partial y}{\partial\xi} & \frac{\partial y}{\partial\eta} \end{bmatrix}. \tag{8.2.11}$$

Exercise 8.2.2 *Prove Theorem (8.2.2).* □

Exercise 8.2.3 *Show that*

$$\begin{bmatrix} \frac{\partial\xi}{\partial x} & \frac{\partial\xi}{\partial y} \\ \frac{\partial\eta}{\partial x} & \frac{\partial\eta}{\partial y} \end{bmatrix} = \frac{1}{det(J)} \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix} \tag{8.2.12}$$

□

Exercise 8.2.4 *Show from these formulas that*

$$\nabla\phi_i = \begin{bmatrix} \beta_i \\ \gamma_i \end{bmatrix}, \tag{8.2.13}$$

as defined in (6.3.13). □

Exercise 8.2.5 *Show that* s_{ij} *in* (8.2.8) *is given by* $s_{ij} = \frac{|\Delta|}{2}(\beta_i\beta_j + \gamma_i\gamma_j)$. □

Exercise 8.2.6 *Show that* $\int_{e_{\xi\eta}} \phi_i(\xi, \eta) d\xi d\eta = \frac{1}{6}$ *for* $i = 1, 2, 3$.
Use this result and (8.2.6) *to derive the Newton Cotes formula* (6.3.26). □

Since the jacobian as well as the derivatives for the linear basis functions are constant a lot of this work is superfluous. However, in the next sections we shall use this approach for more complex elements.

8.3 Quadrilaterals

Until now the derivation of the basis functions for triangles was relatively simple. Once we use quadrilaterals, however, things become more complicated. Due to the continuity requirement, it is not trivial what the basis functions look like.

Let us first start with the simple case of a rectangle with all sides in the coordinate directions. Such a rectangle may be considered as the product of two one-dimensional elements in x and y-direction respectively. The most simple element is the one with the 4 vertices as nodes and a bilinear approximation.

Theorem 8.3.1 *Consider the rectangle* $(x_1, x_2) \times (y_1, y_2)$ *and local node numbers 1 to 4, (Figure 8.6).*

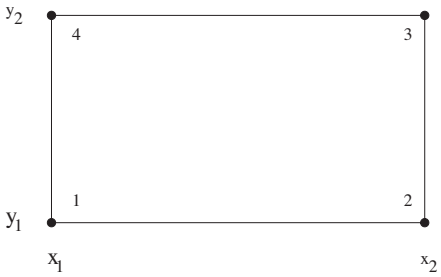


Figure 8.6: Nodes of rectangle.

The four basis functions are defined by:

$$\begin{aligned} \varphi_1(x,y) &= \lambda_1(x)\lambda_1(y), \\ \varphi_2(x,y) &= \lambda_2(x)\lambda_1(y), \\ \varphi_3(x,y) &= \lambda_2(x)\lambda_2(y), \\ \varphi_4(x,y) &= \lambda_1(x)\lambda_2(y), \end{aligned} \tag{8.3.1}$$

with $\lambda_i(x)$ *the one-dimensional basis functions in x-direction and* $\lambda_j(y)$ *in y-direction.*

Exercise 8.3.1 *Prove Theorem (8.3.1).*
Why are these basis functions continuous across element boundaries? □

One easily verifies that the basis functions $\varphi_i(x,y)$ in (8.3.1) have the shape

$$\varphi_i(x,y) = \alpha_i + \beta_i x + \gamma_i y + \delta_i xy. \tag{8.3.2}$$

Unfortunately basis functions of the shape (8.3.2) are not continuous for a general quadrilateral (Why?). Since it is not clear what the general shape of the basis functions must be, we have to use some special construction method.

The standard technique used in the literature is known under the name *isoparametric transformations*. The idea is as follows: one does not know what the basis functions look like for a general quadrilateral but for a square with sides in x and y-direction it is obvious. Therefore one transforms the general quadrilateral element in the x-y-plane with a coordinate transformation $(x,y) \rightarrow (\xi,\eta)$ to a *standard element* (the unit square) in the ξ-η-plane. Such a transformation is called isoparametric if it satisfies the following properties:

1. The nodes $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k$ are transformed to fixed points $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, ..., \boldsymbol{\xi}_k$, i.e. the points in the reference element are always the same.
2. Straight sides in the original element remain straight in the reference element.
3. If the basis functions in the transformed element are given by $\varphi_1(\mathbf{x})$, $\varphi_2(\mathbf{x})$, ... , $\varphi_k(\mathbf{x})$ then the inverse transformation $(\xi, \eta) \rightarrow (x, y)$ is given by

$$\mathbf{x} = \sum_{l=1}^{k} \mathbf{x}_l \varphi_l(\xi, \eta), \tag{8.3.3}$$

and the interpolation by

$$u(\mathbf{x}) = \sum_{l=1}^{k} u_l \varphi_l(\xi, \eta). \tag{8.3.4}$$

In other words we use the same elements for transformation and interpolation.

Note that the basis functions are only known explicitly in the reference element, to compute their values in the original element we have to do a back-transformation. In fact also only the back-transformation is given explicitly.

Figure 8.7 shows the transformation of the quadrilateral element to a unit square in (ξ, η) space.

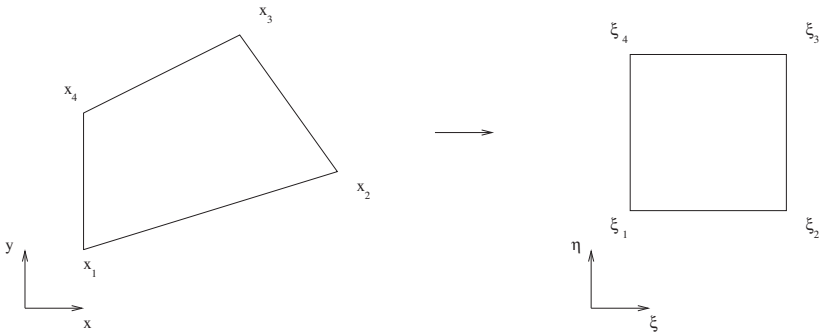


Figure 8.7: Transformation of quadrilateral to unit square.

In this case the isoparametric transformation is a *bilinear transformation*. The nodes $\mathbf{x}_i$ of the quadrilateral are transformed to the vertices of the unit square in the following way:

$$\mathbf{x}_1 \rightarrow (0,0),\ \mathbf{x}_2 \rightarrow (1,0),\ \mathbf{x}_3 \rightarrow (1,1),\ \mathbf{x}_4 \rightarrow (0,1). \tag{8.3.5}$$

The basis functions in the (ξ, η)-plane are bi-linear and defined by

$$\varphi_1 = (1-\xi)(1-\eta),\ \varphi_2 = \xi(1-\eta),\ \varphi_3 = \xi\eta,\ \varphi_4 = (1-\xi)\eta. \tag{8.3.6}$$

Note that the transformation (8.3.3) transforms straight sides of the reference element into straight sides of the quadrilateral in (x, y)-space. Moreover the function $u(\mathbf{x})$ defined by (8.3.4) reduces to a straight line on the sides of the quadrilateral (Why?). Hence continuity of the interpolation is satisfied.

In order that the transformation is applicable it must be invertible, in other words for each $\mathbf{x}$ in the quadrilateral we must have a unique $\boldsymbol{\xi}$. So the Jacobian of the transformation must be non-singular for each $\mathbf{x}$ in the quadrilateral.

Theorem 8.3.2 *The transformation (8.3.3) is given by*

$$\begin{aligned} x &= x_1 + (x_2 - x_1)\xi + (x_4 - x_1)\eta + (x_1 - x_2 + x_3 - x_4)\xi\eta, \\ y &= y_1 + (y_2 - y_1)\xi + (y_4 - y_1)\eta + (y_1 - y_2 + y_3 - y_4)\xi\eta. \end{aligned} \tag{8.3.7}$$

Theorem 8.3.3 *The determinant of* $\mathbf{J}$ *is equal to:*

$$det(\mathbf{J}) = (x_2 - x_1 + A_x\eta)(y_4 - y_1 + A_y\xi) - (x_4 - x_1 + A_x\xi)(y_2 - y_1 + A_y\eta), \tag{8.3.8}$$

with $A_x = x_1 - x_2 + x_3 - x_4$ *and* $A_y = y_1 - y_2 + y_3 - y_4$.

Exercise 8.3.2 *Prove Theorem (8.3.2).* □

Exercise 8.3.3 *Prove Theorem (8.3.3).* □

Theorem 8.3.4 *The transformation (8.3.4) is invertible if and only if all angles of the quadrilateral are less than* π*, i.e. the quadrilateral is convex.*

Proof

Since the terms of second degree cancel, d$et(\mathbf{J})$ is linear. In other words if the sign of d$et(\mathbf{J})$ is the same at each vertex, it is impossible that it is zero inside the element (Why?).

The value of the d$et(\mathbf{J})$ in point (0,0) is equal to $(x_2 - x_1)(y_4 - y_1) - (x_4 - x_1)(y_2 - y_1)$ which is equal to the outer product $(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_4 - \mathbf{x}_1)$.

Since $\mathbf{v}_1 \times \mathbf{v}_2 = ||\mathbf{v}_1||\ ||\mathbf{v}_2||\sin\varphi$, with φ the angle between the two vectors. This is positive if the angle φ (counter-clockwise) is less than π. So in point 1 we must have an angle less than π.

The same holds for all other points and corresponding angles.

In conclusion for a convex quadrilateral the transformation is regular, if one of the angles is larger than π the transformation is singular.

To show how this transformation can be utilized to compute an element matrix or vector, we consider the simple example of the Poisson equation (6.2.1).

The element matrix has elements $s_{ij} = \int_e \nabla\varphi_i(\mathbf{x}) \cdot \nabla\varphi_j(\mathbf{x})\ d\Omega$. Since the basis functions are only known in the reference element we have to transform this integral to an integral in the (ξ, η)-plane as given in (8.2.7). In general it is complicated to compute the integral (8.2.7) exactly so one uses a numerical integration rule.

Theorem 8.3.5 *The Newton Cotes rule corresponding to the reference element 8.3.5 is given by the two-dimensional equivalent of the Trapezoid rule:*

$$\int_0^1 \int_0^1 Int(\xi, \eta)\ d\xi d\eta = \frac{1}{4} \sum_{k=1}^{4} Int(\xi_k, \eta_k). \tag{8.3.9}$$

Exercise 8.3.4 *Prove Theorem (8.3.5).* □

Approximation of Equation (8.2.7) by the Newton Cotes rule (8.3.9) leads to

$$s_{ij} \approx \frac{1}{4} \sum_{k=1}^{4} (\nabla\varphi_i \cdot \nabla\varphi_j |det(J)|)(\xi_k, \eta_k). \tag{8.3.10}$$

The values of $|det(J)|$ in the integration points in the reference element can be computed immediately from Equation (8.3.8). To compute the values of $\nabla\varphi_i$ we have to express the derivatives to x and y into derivatives of ξ and η, since φ_i is only known in the (ξ, η)-plane, see (8.2.9).

Exercise 8.3.5 *Compute the values of $\frac{\partial\xi}{\partial x}$ in the integration points.* □

So the easiest way to approximate the Integrals (8.2.7) in the element matrix is to create a number of tables and combine these tables into the Sum (8.3.9).

In FEM programs the standard sequence to do it is the following:

1. Make a table of the values of ξ and η in the integration points. In this case this table is very simple, but if one uses Gauss integration the numbers are less trivial.

2. Make a table of the weights of the numerical integration. The weights include the factor $|det(J)|$, hence in this particular case we have $w_k = \frac{1}{4}|det(J(\mathbf{x}_k))|$.

3. Make a table of $\frac{\partial x}{\partial\xi}$ and so on in the integration points.

4. Use this table also to create $\nabla\varphi_i$.

Note that only the results of steps 2 and 4 are needed to compute the integrals (8.2.7).

The original weights of the numerical integration ($\frac{1}{4}$) are dimensionless, but due to the multiplication by $|det(J)|$, the weights get the dimension of an area.

Exercise 8.3.6 *Compute the element vector corresponding Poisson's equation (6.3.1), in the case of arbitrary quadrilateral. Use Newton Cotes integration.* □

8.4 Curved quadratic triangles

In Section 8.1 we have shown how basis functions for a straight quadratic triangle can be expressed in terms of linear basis functions. When the boundary of the domain is curved, it is necessary to approximate the boundary of the region by a piecewise quadratic polynomial in order to get the same order of accuracy one expects by using quadratic elements (See Section 8.7). So in practice we may have quadratic elements with a curved boundary.

The derivation of the basis functions for these elements is very similar to that of quadrilaterals. It is hard to find the general expression for the basis functions on the curved triangle and therefore we use an isoparametric transformation to map the curved triangle on a reference triangle with straight sides and vertices (0,0), (1,0) and (0,1). See Figure 8.8.

Let $\varphi_i(\mathbf{x})$ be the basis functions corresponding to the straight triangle (see Formula (8.1.7), with $\lambda_1 = 1 - \xi - \eta$, $\lambda_2 = \xi$ and $\lambda_3 = \eta$). The isoparametric transformation is defined by

$$\mathbf{x} = \sum_{k=1}^{6} \mathbf{x}_k \varphi_k(\xi, \eta). \tag{8.4.1}$$

Due to this transformation the boundaries of the triangle will be polynomials of degree two at most.

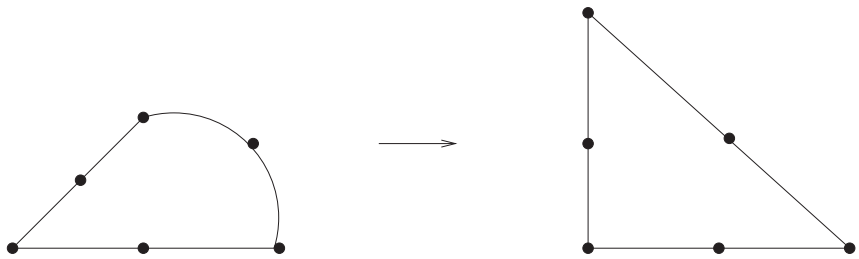


Figure 8.8: Transformation of curved triangle to reference element.

The transformation is non-singular if the determinant of the Jacobian has the same sign on the whole triangle. Unfortunately it is not easy to give a general rule about the restrictions an element has to satisfy in order that the transformation is regular. In general, angles of the triangle should not be too large ($< 135°$) and the "mid"points of the sides must be close to the actual middle of the edge. Furthermore the triangle may not be too curved. Usually it suffices to check the determinant in the integration points. If the sign of the determinant is the same in all integration points, one assumes that the mapping is invertible.

From Equation (8.4.1) it follows that

$$\frac{\partial x}{\partial \xi} = \sum_{k=1}^{6} \mathbf{x}_k \frac{\partial}{\partial \xi} \varphi_k(\xi, \eta) \,, \quad \frac{\partial x}{\partial \eta} = \sum_{k=1}^{6} \mathbf{x}_k \frac{\partial}{\partial \eta} \varphi_k(\xi, \eta) \,. \tag{8.4.2}$$

Exercise 8.4.1 *Show that the Newton Cotes integration rule for the quadratic reference element is given by*

$$\int_e Int(\xi, \eta)\, d\xi\, d\eta = \frac{1}{6}(Int(\frac{1}{2}, 0) + Int(\frac{1}{2}, \frac{1}{2}) + Int(0, \frac{1}{2})). \tag{8.4.3}$$

□

Exercise 8.4.2 *Compute the (ξ, η) derivatives of the basis functions in the Newton Cotes integration points of the reference element.* □

Exercise 8.4.3 *Show how the Jacobian matrix can be computed in the Newton Cotes integration points.* □

Exercise 8.4.4 *Indicate how the weights for the Newton Cotes integration in the original curved element can be computed.* □

Exercise 8.4.5 *Show how the x and y derivatives of the basis functions in the Newton Cotes integration points can be computed.* □

8.5 Application to the Stokes equations

The Stokes equations are derived from the Navier-Stokes equations (2.4.4a,2.4.4b) by removing the convective terms. These equations are only valid in case of small velocities. If the viscosity is a constant, the Stokes equations for incompressible

flow may be formulated as:

$$\begin{aligned} -\mu\Delta u + \frac{\partial p}{\partial x} &= f_x, \\ -\mu\Delta v + \frac{\partial p}{\partial y} &= f_y, \\ \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0. \end{aligned} \tag{8.5.1}$$

$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix}$ is the velocity vector and p the pressure. In order to have a unique solution it is necessary to give boundary conditions for each velocity component on the complete boundary. One can prove that no explicit boundary condition for the pressure is required.

As an example we consider flow in straight channel, see Figure (8.9).

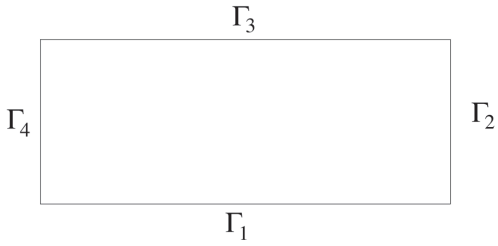


Figure 8.9: Straight channel with boundaries.

Boundary conditions are a given velocity vector on the inflow boundary Γ_4, no-slip boundary conditions on Γ_1 and Γ_3 and *outflow boundary conditions* on Γ_2. The mathematical formulation of the boundary conditions reads

$$\begin{array}{lll} u = 0, & v = 0 & \text{on } \Gamma_1, \\ p - \mu\frac{\partial u}{\partial n} = 0, & v = 0 & \text{on } \Gamma_2, \\ u = 0, & v = 0 & \text{on } \Gamma_3, \\ u = u(y), & v = 0 & \text{on } \Gamma_4. \end{array} \tag{8.5.2}$$

For reasons that go beyond the scope of this book, the polynomials to approximate the pressure are in general one degree lower than those of the velocity components. Both velocity components are approximated in the same way. In order to derive the weak formulation we multiply the first equation in (8.5.1) by a test function δu, the second one by δv and the last one by δp. These test functions belong to the same spaces as u, v and p respectively.

Exercise 8.5.1 *Show that the weak formulation of (8.5.1) with boundary conditions (8.5.2) can be written as*

$$\begin{aligned} \int_\Omega \mu\nabla u \cdot \nabla\delta u \, d\Omega - \int_\Omega p\frac{\partial \delta u}{\partial x} \, d\Omega &= \int_\Omega f_x \delta u \, d\Omega, \\ \int_\Omega \mu\nabla v \cdot \nabla\delta v \, d\Omega - \int_\Omega p\frac{\partial \delta v}{\partial y} \, d\Omega &= \int_\Omega f_y \delta v \, d\Omega, \\ \int_\Omega (\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y})\delta p \, d\Omega &= 0. \end{aligned} \tag{8.5.3}$$

What are the boundary conditions for δu, δv and δp? □

Exercise 8.5.2 *Give the continuity requirements for u v, δu and δv.*

Are there any restrictions for p and δp? □

One of the most simple elements found in the literature for the (Navier-)Stokes equations for incompressible flow is the *bi-linear velocity, constant pressure quadrilateral*. The velocity components are approximated by bi-linear polynomials in the way described in Section (8.3). The pressure is approximated by a constant per element.

Exercise 8.5.3 *Show that the Galerkin equations corresponding to the weak formulation (8.5.3) are given by*

$$\begin{aligned}
\sum_{j=1}^{n} u_j \int_\Omega \mu \nabla\varphi_j \cdot \nabla\varphi_i \, d\Omega - \sum_{j=1}^{m} p_j \int_\Omega \psi_j \frac{\partial\varphi_i}{\partial x} \, d\Omega &= \int_\Omega f_x \varphi_i \, d\Omega \;\; (i = 1, ..., n_u), \\
\sum_{j=1}^{n} v_j \int_\Omega \mu \nabla\varphi_j \cdot \nabla\varphi_i \, d\Omega - \sum_{j=1}^{m} p_j \int_\Omega \psi_j \frac{\partial\varphi_i}{\partial y} \, d\Omega &= \int_\Omega f_y \varphi_i \, d\Omega \;\; (i = 1, ..., n_v), \\
\sum_{j=1}^{n} u_j \int_\Omega \frac{\partial\varphi_j}{\partial x} \psi_i \, d\Omega + \sum_{j=1}^{n} v_j \int_\Omega \frac{\partial\varphi_j}{\partial y} \psi_i \, d\Omega &= 0 \;\; (i = 1, ..., m). \qquad (8.5.4)
\end{aligned}$$

Here n is the number of u or v velocity unknowns and m the number of pressure unknowns. n_u is the number of non-prescribed u velocity unknowns and n_v is the number of non-prescribed v velocity unknowns. □

Exercise 8.5.4 *What is the number of u-velocity unknowns per element? And the number of p unknowns?*

Give an expression for the pressure basis functions ψ_i per element. □

Exercise 8.5.5 *Suppose we order the unknowns per element in the sequence u_1, u_2, ... ,v_1, v_2, ... , p_1, ...*
Then the element matrix can be split into 9 parts according to:

$$\mathbf{S}^{e_k} = \begin{bmatrix} \mathbf{S}_{uu} & \mathbf{S}_{uv} & \mathbf{S}_{up} \\ \mathbf{S}_{vu} & \mathbf{S}_{vv} & \mathbf{S}_{vp} \\ \mathbf{S}_{pu} & \mathbf{S}_{pv} & \mathbf{S}_{pp} \end{bmatrix} \qquad (8.5.5)$$

Give the sizes of the subelement matrices.
Give the formulas of the elements of each subelement matrix in integral form. □

8.6 Circle symmetry

Most real world problems are three-dimensional. But solving 3D problems is laborious and time-consuming and on top of that post-processing, like graphically representing results is much more difficult for 3D- than for 2D-problems. Therefore one often tries to reduce a problem from three to two dimensions by assuming certain symmetries in the solution. One such possibility is the assumption that a solution does not depend on a certain coordinate. (Translation symmetry). Another possibility is, if a region is cylindrically shaped, to assume that the solution has circular symmetry. For that to be possible all data, including boundary conditions, must have circular symmetry. In that case it is possible to reduce the

three-dimensional problem to a two-dimensional one by introducing cylinder coordinates (r, θ, z), defined by

$$\begin{aligned} x &= r\cos\theta, \\ y &= r\sin\theta, \\ z &= z. \end{aligned} \tag{8.6.1}$$

In finite difference methods, the standard approach is to transform the PDE in (x, y, z) to a PDE in (r, z).

Exercise 8.6.1 *Show that Poisson's equation $-\Delta u = f$ in cylinder coordinates (r, z) can be written as:*

$$\frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} + \frac{1}{r}\frac{\partial u}{\partial r} = f. \tag{8.6.2}$$

Suppose that $r = 0$ is part of the region. What is the boundary condition in $r = 0$? Why do we need a boundary condition in $r = 0$? □

Of course in FEM it is also possible to solve the transformed Poisson equation (8.6.2) with boundary condition as defined in Exercise 8.6.1. However, in that case we have to take care of the singularity in $r = 0$. Also we have to take the artificial boundary condition in $r = 0$ into account.

A more natural approach is the following. We derive the weak formulation and Galerkin equations for the original 3D problem. This does not contain any singularity in $r = 0$, nor do we need an artificial boundary condition. Afterwards we take into account that the solution is constant in θ-direction by assuming that the basis functions are constant in that direction. So integration in the θ-direction is trivial. This approach also leads to a 2D formulation, but without special requirements.

Exercise 8.6.2 *Let u satisfy Poisson's equation $-\Delta u = f$ on a 3D circle-symmetric region. Let $u = g$ at the boundary with g independent of θ. Show that the Galerkin equations corresponding to this problem are given by*

$$\sum_{j=1}^{n} u_j \int_{\Omega} \nabla\varphi_i \cdot \nabla\varphi_j \, d\Omega = \int_{\Omega} f\varphi_i \, d\Omega. \tag{8.6.3}$$

□

In order to compute the element matrix and vector on this element we have to transform the Expression (8.6.3) from (x, y, z)-coordinates to (r, z)-coordinates.

Exercise 8.6.3 *Show that the determinant of the Jacobian matrix of the transformation is equal to r.* □

Exercise 8.6.4 *Show that the elements of the element matrix are given by*

$$s_{ij} = 2\pi \int_{e_{rz}} \left(\frac{\partial\varphi_i}{\partial r}\frac{\partial\varphi_j}{\partial r} + \frac{\partial\varphi_i}{\partial z}\frac{\partial\varphi_j}{\partial z}\right) r \, dr dz. \tag{8.6.4}$$

Hint: transform $\frac{\partial\varphi_i}{\partial x}$, $\frac{\partial\varphi_i}{\partial y}$ and $\frac{\partial\varphi_i}{\partial z}$ into $\frac{\partial\varphi_i}{\partial r}$ and $\frac{\partial\varphi_i}{\partial z}$, using the transformation (8.6.1) □

8.7 Theoretical remarks

In the mathematical parts of Chapters 5 to 7 it has been shown that the error made by the finite element method (measured in energy norm), is smaller than a constant C times the error we would have if we interpolated the exact solution by the same type of approximation:

$$\|u - u_h\|_L \leq C\|u - u_I\|_L, \tag{8.7.1}$$

with u the exact solution, u_h the FEM-solution, u_I the interpolation of u and $\| \ \|_L$ the energy norm. In a minimization problem $C = 1$, proving that the FEM solution is the best approximation in the energy norm. In a general weak formulation, the constant C is related to the ratio of the symmetric and the anti-symmetric part of the operator L. For example in a convection-dominated flow C is large, whereas for diffusion dominated problems C is close to 1.

From (8.7.1) it follows that if we want to estimate the FEM error it is necessary to estimate the interpolation error. Suppose we use an approximation by k^{th} degree polynomials. Then one can prove, under certain (geometrical) conditions, that the error in L^2 norm is of the order h^{k+1}. So for a linear approximation, the interpolation error in L^2 norm is $O(h^2)$.

If the order of the differential equation is $2m$ (Poisson $m = 1$, bi-harmonic $m = 2$), then the energy norm contains derivatives of order m. In general, for each derivative, the interpolation error is reduced by an order 1. So the interpolation error in energy norm of an $2m$-th order operator using k-th degrees polynomials is of order h^{k+1-m}. This means that also the FEM error in energy norm is of order h^{k+1-m}. Under certain conditions one can prove that this error in L^2 norm is again of order h^{k+1}, which is comparable to the interpolation error.

In the above text we mentioned "under certain (geometrical) conditions". One can prove that the elements must satisfy some requirements in order that these estimates can be applied. It goes beyond the scope of this book to give exact formulations, but the following rules are generally valid.

- In case of triangles, the largest angle must not be too close to π. A practical bound is that all angles must be smaller than 135°. A large angle gives a bad approximation of the derivatives and thus a large error. Sharp corners, on the other hand, do not pose problems.
- In case of quadrilaterals, it is necessary that the mapping to a standard square is invertible. In fact the size of the Jacobian must be not too large. In practice this also implies that corners must not be much larger than 135°.

All before mentioned estimates are valid in case of exact integration and under the condition that the whole region is completely covered by elements. In practice, however, one uses numerical integration and the boundary will be approximated by polynomials. It is very difficult to make an estimation of the effect of these approximations. Under certain special conditions, one can prove some theorems about this matter ([11]), but these proofs are very complicated.

In general one can state the following:

If we use polynomials of degree k to approximate the solution, it is also necessary to approximate the boundary of the region by polynomials of the same order. Otherwise the order of accuracy is reduced to lower order. So with linear elements, it is sufficient to approximate the boundary of the region by piecewise linear polynomials, i.e. straight lines. But with quadratic elements, it is necessary to use a quadratic approximation of the boundary.

With respect to the numerical integration, the rules are more complicated.

In Cartesian coordinates it is necessary that the numerical integration is exact for polynomials of degree $2k - 2m$, otherwise the accuracy of the global approximation is reduced. Hence for a second order differential equation ($m = 1$), we have the following requirements:

- Linear elements ($k = 1$), the integration must be exact for constant polynomials.
- Quadratic elements ($k = 2$), the integration must be exact for quadratic polynomials.
- Third degree elements ($k = 3$), the integration must be exact for fourth order polynomials.

For that reason Newton Cotes integration can only be applied to linear and quadratic elements.

In other types of coordinate systems, like for example cylindrical coordinates, the situation is more complex. Actually the rules above remain valid if we adapt the integration rules to reflect the type of coordinates. For example if we incorporate a factor r in the integration rules for cylindrical coordinates then the same type of rules apply. If we do not adapt the integration rules, the numerical integration must be exact for polynomials of degree $2k - 2m + 1$. So in that case the Newton Cotes rule can only be applied to linear elements.

8.8 Fourth order problems

Until now we have limited ourselves to second order problems. This is not without a reason, fourth order problems are much more difficult to solve. An extended description of the various methods to handle this kind of problems is beyond the scope of this book. Nevertheless we shall show some basic techniques used in the literature. These methods will be shown using a very simple example, the clamped beam.

8.8.1 The clamped beam

Consider the beam sketched in Figure 8.10, clamped in both ends 0 and l.

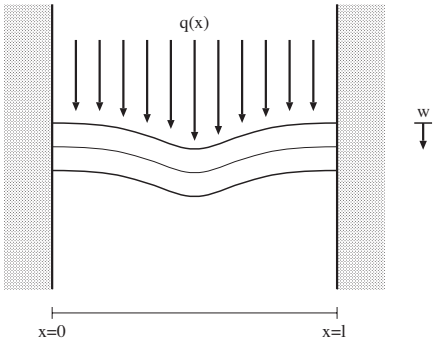


Figure 8.10: Clamped beam with load q.

On the beam we have a given load $q(x)$. The transverse displacement w of the neutral line with respect to the equilibrium satisfies the differential equation:

$$EI\frac{d^4w}{dx^4} = q, \tag{8.8.1}$$

with EI the *flexural rigidity* of the beam. Boundary conditions are

$$w(0) = w(l) = 0, \quad \frac{dw}{dx}(0) = \frac{dw}{dx}(l) = 0. \tag{8.8.2}$$

Note that for this fourth order problem we have to give 2 boundary conditions on the whole boundary.
The transverse displacement w is the solution of the minimization problem

$$\min_w \int_0^l \frac{1}{2}EI(\frac{d^2w}{dx^2})^2 - qw\,dx. \tag{8.8.3}$$

Exercise 8.8.1 *Prove (8.8.3).* □

In order to solve the minimization problem (8.8.3), Ritz's method may be applied and we use the FEM to construct the basis functions $\varphi_i(x)$.
Suppose that we approximate w by a linear combination of basis functions $\varphi_i(x)$

$$w_h(x) = \sum_{j=1}^{n} \alpha_j \varphi_j(x). \tag{8.8.4}$$

Then the Ritz equations corresponding to 8.8.3 are given by

$$\sum_{j=1}^{n} \alpha_j \int_0^l EI\frac{d^2\varphi_j}{dx^2}\frac{d^2\varphi_i}{dx^2}\,dx = \int_0^l q\varphi_i\,dx. \tag{8.8.5}$$

Exercise 8.8.2 *Prove (8.8.5).* □

The basis functions φ_i must satisfy the boundary conditions (8.8.2), but also they have to be continuously differentiable, i.e. $\varphi_i(x) \in C^1(0,l)$. Why?

The simplest element in the FEM, that can be used to construct the basis functions is a 2 node element. In order to ensure the continuity of the derivatives over the element boundaries, *Hermitian interpolation* is applied. In each node i we introduce two unknowns w_i and $(w_x)_i$ and write the interpolation per element as

$$w_h(x) = \sum_{j=1}^{2} w_j\psi_{j0}(x) + \sum_{j=1}^{2}(w_x)_j\psi_{j1}(x), \tag{8.8.6}$$

with $\psi_{j0}(x)$ and $\psi_{j1}(x)$ third degree polynomials, satisfying

$$\psi_{j0}(x_i) = \delta_{ij}, \;\; \frac{d\psi_{j0}}{dx}(x_i) = 0, \;\; \psi_{j1}(x_i) = 0, \;\; \frac{d\psi_{j1}}{dx}(x_i) = \delta_{ij} \tag{8.8.7}$$

So the parameters α_j in (8.8.4) are either w_j or $(w_x)_j$.

Exercise 8.8.3 *Express the basis functions $\psi_{j0}(x)$ and $\psi_{j1}(x)$ in terms of the linear basis functions $\lambda_i(x)$.* □

Exercise 8.8.4 *Compute with the basis functions ψ_{j0} and ψ_{j1} the element matrix corresponding to (8.8.5) for this element.*
What is the size of the element matrix?
Suppose that q is a constant. Compute the element vector. □

In order to get continuity of the first derivatives, we had to introduce the first derivatives of the unknown as parameters. There are in fact alternatives but they are more complicated.

If we extend the example to 2D or even 3D problems, construction of basis functions satisfying continuity of the first derivatives is much more cumbersome. For example if we want a complete polynomial (i.e. a polynomial containing all terms until a certain degree) on a triangle satisfying continuity of the first derivatives, it is necessary to use a fifth degree polynomial with 21 parameters. The reason is that we need continuity both in tangential and normal direction. If we drop the demand for a complete polynomial, the degree may be lowered a bit, but still the element is very complicated.

For that reason one can find many attempts in the literature to get rid of the C^1 requirement. In fact one can find two major solution strategies:

- violate the C^1 continuity requirement and carry out the assembly procedure as if there is no problem, in other words use non-conforming elements.
- Use a mixed formulation

The non-conforming approach is wrong in general, but can be justified if special conditions are met. These conditions are formulated in terms of the *Patch test* of Irons ([21]). It is not simple to apply this condition for general PDEs.

The mixed formulation is easier to generalize and we shall do the beam problem as a simple example.

8.8.2 A simple example of the mixed approach

The idea of the mixed approach is simple. We formulate the minimization problem (8.8.3) in terms of two variables w and β ($=\frac{dw}{dx}$), instead of one (w). These variables are considered to be independent, but we relate them by the constraint

$$\beta - \frac{dw}{dx} = 0. \tag{8.8.8}$$

So we can rewrite the minimization problem as

$$\min_{w,\beta} \int_0^l \frac{1}{2} EI(\frac{d\beta}{dx})^2 - qw \, dx, \tag{8.8.9}$$

under the constraint (8.8.8). A well known technique from the theory of minimization with constraints is the *penalty approach*. We multiply the squared of the constraint by a number $\frac{\alpha}{2}$ and add this to the minimization problem:

$$\min_{w,\beta} \int_0^l \frac{1}{2} EI(\frac{d\beta}{dx})^2 + \frac{\alpha}{2}(\beta - \frac{dw}{dx})^2 - qw \, dx, \tag{8.8.10}$$

If α is large the minimum is reached for $\beta - \frac{dw}{dx}$ small. So the constraint is satisfied approximately. The final step is to solve the penalized minimization problem by the FEM. Both β and w are approximated by linear polynomials per element

$$\beta_h = \sum_{j=1}^{n} \beta_j \varphi_j(x), \quad w_h = \sum_{j=1}^{n} w_j \varphi_j(x). \tag{8.8.11}$$

The boundary conditions can be formulated in terms of w and β.

Exercise 8.8.5 *Show that the Ritz equations corresponding to (8.8.10) using the approximation (8.8.11) are given by*

$$\sum_{j=1}^{n} w_j \int_0^l \alpha \frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx}\,dx - \sum_{j=1}^{n} \beta_j \int_0^l \alpha\varphi_j \frac{d\varphi_i}{dx}\,dx = \int_0^l q\varphi_i\,dx, \quad i = 1,\ldots,n, \tag{8.8.12}$$

$$-\sum_{j=1}^{n} w_j \int_0^l \alpha\varphi_i \frac{d\varphi_j}{dx}\,dx + \sum_{j=1}^{n} \beta_j \int_0^l (EI\frac{d\varphi_i}{dx}\frac{d\varphi_j}{dx} + \alpha\varphi_i\varphi_j)\,dx = 0, \quad i = 1,\ldots,n. \tag{8.8.13}$$

Is the stiffness matrix symmetrical? □

Exercise 8.8.6 *Order the unknowns in the sequence $w_1, w_2, \beta_1, \beta_2$. Compute the element matrix and vector under the condition that E, I, α and q are constant.* □

The formulation given above is only one of the many different mixed formulations one can find in the literature. It is just a demonstration, of how by introducing new variables, one can effectively reduce the order of the equations. In practice, however, one must be careful with this kind of approximations, since in many cases the various unknowns must approximated with different types of polynomial. A discussion of this subject goes beyond the scope of this book.

8.9 Summary of Chapter 8

In Chapters 6 and 7 linear interpolation functions were used. In this chapter it has been demonstrated how the basis functions for higher order elements can be derived.
Also quadrilaterals, which require a special approach to ensure continuity, have been treated. For quadrilaterals as well as curved elements, the standard technique is mapping an arbitrary element onto a standard (reference) element. All integrals are evaluated on this reference element.

Fourth order problems have been introduced for a very simple 1D example, just to show the difficulties and possible solutions.

Chapter 9

Solution of large systems of equations

Objectives

In this chapter we will focus on the solution of systems of linear equations resulting from the discretization of PDE's. The corresponding matrices are generally large and sparse. There are two classes of methods to solve such systems of equations: direct and iterative methods. All direct solvers are variants of Gaussian elimination.
We shall first deal with the direct solvers. Special storage techniques to reduce the amount of memory like band methods and profile methods are treated. Using a special renumbering technique, the size of the matrix can be made semi-optimal.
Direct methods sometimes fall short. This happens mostly, when the number of unknowns becomes excessive like in large 3D problems. Even with optimal numbering the fill-in becomes huge and the L and U matrices no longer fit into memory. That is where iterative methods enter the picture. Historically there has been a trade off between memory and computing time. This certainly is true for the earlier iteration methods like *Jacobi*, *Gauss-Seidel* and *Successive Overrelaxation (SOR))*. But in the early seventies two very different methods became popular fast: the *Conjugate Gradient Method (CG)*, later generalized to *Bi-CGStab* and the *Multigrid Method*. These methods were so successful in fact, that they challenged the direct methods in their own back yard: computation time. The Multigrid method even achieves theoretically the best result possible: the number of computations increases *linearly* with the number of unknowns.

We start with classical iteration methods and derive convergence and stop criteria for them. Then we turn our attention to Krylov space methods like CG and Bi-CGStab. We shall see, that the success of these methods much depends on the choice of *preconditioner* and that classical iteration methods may serve as a preconditioner. Finally we shall describe a very powerful preconditioner: Incomplete LU decomposition.

Finally we shall turn our attention to Multigrid. There too we shall see that the success of the method will depend on various strategic choices. The preconditioners are called *smoothers* in Multigrid.

Often Multigrid and CG like methods are presented as competitors. There is no need for that: Multigrid is an excellent preconditioner for Bi-CGStab.

9.1 Direct methods

9.1.1 Introduction

The discretization of an elliptic PDE leads always to a system of (non-)linear equations. As will be seen in Section 9.7, non-linear systems are solved by a series of linear problems with the same structure. So a fast solution of systems of linear equations is of great importance for the discretization of PDE's.
The matrices resulting from discretization are in general large and sparse. If a suitable numbering is applied, these matrices have also a band structure. Matrices for which only elements within the band are stored are referred to as *band matrices*. Matrices for which only the non-zero elements are stored are known as *compact matrices*. They require extra information about the position of the non-zero elements.

Exercise 9.1.1 *Assume that we discretize the Poisson equation with Dirichlet boundary conditions on a rectangular domain by a central finite difference discretization. Suppose that the number of nodes in each coordinate direction is equal to $n + 2$.*
Show that the size of the discretization matrix is equal to $n \times n$ in $\mathbb{R}^1$, $n^2 \times n^2$ in $\mathbb{R}^2$ and $n^3 \times n^3$ in $\mathbb{R}^3$. □

Exercise 9.1.2 *Suppose that we use a natural numbering.*
Show that the band width of the matrices in Exercise 9.1.1 is equal to 3 in $\mathbb{R}^1$, $2n + 1$ in $\mathbb{R}^2$ and $2n^2 + 1$ in $\mathbb{R}^3$. □

Exercise 9.1.3 *Show that the number of non-zero elements per row for the matrices in Exercise 9.1.1 is equal to 3 in $\mathbb{R}^1$, 5 in $\mathbb{R}^2$ and 7 in $\mathbb{R}^3$.* □

Exercise 9.1.4 *Compute the number of entries that we have to store for the matrices in Exercise 9.1.1 in case of a full matrix, a band matrix and a compact matrix for n = 10, 100 and 1000 respectively. How many bytes is this, if a real takes 8 bytes?* □

The previous exercises show that for $n = 10$ the band matrices in all three dimensions can be stored in the internal memory of the computer, but that for $n = 100$ only the 1D and 2D matrices fit into memory. In the case of $n = 1000$ even the 2D band matrix is too large and the 3D compact matrix fits only in very large 64 bits computers.
In the next sections we shall treat some basis techniques for direct and linear solvers.

9.1.2 Gaussian elimination

As mentioned in Section 9.1.1 all direct solvers are variants of Gaussian elimination. In numerical applications, Gaussian elimination is carried out in the form of a *LU-decomposition*. In case of a *band matrix*, which arises if we apply a discretization technique on a structured (rectangular) grid, all elements outside the band are zero. This property is kept after Gaussian elimination, provided rows and columns are not interchanged. For unstructured meshes, like in FEM, a more sophisticated approach is necessary: the *profile method*. A good numbering of the equations is essential to keep the number of elements to be stored as low as possible.
First we start by explaining the LU-decomposition, then band methods are treated, followed by profile methods. Finally we make a few remarks about automatic optimal renumbering techniques.
For completeness we give a short description of the Gaussian elimination process.

Consider the system of linear equations:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2, \\ \vdots \qquad\qquad & \quad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n, \end{aligned} \tag{9.1.1}$$

or in matrix vector notation

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{9.1.2}$$

Gaussian elimination transforms system (9.1.2) into an upper triangular matrix by elementary row operations.
Consider the matrix $\mathbf{A}^0$ extended with the right-hand side $\mathbf{b}$.

$$\mathbf{A}^{(0)} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} & b_1 \\ a_{21} & a_{22} & \ldots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} & b_n \end{pmatrix}. \tag{9.1.3}$$

Subtract the first row multiplied by a suitable constant from the other rows, such that the first column becomes zero from row two. Define:

$$\begin{aligned} m_{j1} &= \tfrac{a_{j1}}{a_{11}} \qquad\qquad j = 2, 3, \ldots n. \\ a_{jk}^{(1)} &= a_{jk} - m_{j1}a_{1k} \quad k = 1, \ldots, n \\ \text{and} \quad b_j^{(1)} &= b_j - m_{j1}b_1. \end{aligned} \tag{9.1.4}$$

One easily verifies that $a_{j1}^{(1)} = 0$, $j = 2, 3, \ldots, n$. This produces a new extended matrix

$$\mathbf{A}^{(1)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & a_{32}^{(1)} & a_{33}^{(1)} & \ldots & a_{3n}^{(1)} & b_3^{(1)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \ldots & a_{nn}^{(1)} & b_n^{(1)} \end{pmatrix}. \tag{9.1.5}$$

It can be verified easily that the system of equations, described by this new extended matrix has the same solution as the original system 9.1.2. Now subtract the second row times a constant from the next rows such that the second column becomes zero from row number 2. Hence:

$$\begin{aligned} m_{j2} &= \tfrac{a_{j2}^{(1)}}{a_{22}^{(1)}} \qquad\qquad j = 3, 4, \ldots n. \\ a_{jk}^{(2)} &= a_{jk}^{(1)} - m_{j2}a_{2k}^{(1)} \quad k = 2, \ldots, n \\ \text{and} \quad b_j^{(2)} &= b_j^{(1)} - m_{j2}b_2^{(1)}. \end{aligned} \tag{9.1.6}$$

This gives

$$\mathbf{A}^{(2)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} & b_2^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \ldots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \ldots & a_{nn}^{(2)} & b_n^{(2)} \end{pmatrix}. \tag{9.1.7}$$

The i^{th} step of the iteration:

$$\begin{aligned} m_{ji} &= \frac{a_{ji}^{(i-1)}}{a_{ii}^{(i-1)}} && j = i+1, i+2, \ldots n. \\ a_{jk}^{(i)} &= a_{jk}^{(i-1)} - m_{ji} a_{ik}^{(i-1)} && k = i, i+1, \ldots, n \\ \text{and} \quad b_j^{(i)} &= b_j^{(i-1)} - m_{ji} b_i^{(i-1)}. \end{aligned} \tag{9.1.8}$$

If we proceed this process until $i = n-1$, we get the matrix

$$\mathbf{A}^{(n-1)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} & b_1 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} & b_2^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \ldots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & \vdots & 0 & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & a_{nn}^{(n-1)} & b_n^{(n-1)} \end{pmatrix}. \tag{9.1.9}$$

This is an upper triangular system. The solution can be determined immediately by back substitution. The quantities m_{ji} are called *multiplicators* and the quantities a_{ii}^{i-1} *pivots*. Since in the i^{th} step we have to subdivide by the pivot a_{ii}^{i-1} to compute the multiplicators m_{ji}, a necessary and sufficient condition for the application of the Gaussian elimination process is that none of the pivots a_{ii}^{i-1} is zero. Usually one interchanges rows and/or columns of the matrix to avoid zero (or small) pivots. However, in this book we shall not use this interchanging process called *pivoting* .

9.1.3 LU-decomposition

The Gaussian elimination process, transforms the matrix A with elements a_{ij} by elementary row operations into an upper triangular matrix U:

$$U = \begin{pmatrix} a_{11} & a_{12} & \ldots & \ldots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} \\ \vdots & 0 & a_{33}^{(2)} & \ldots & a_{3n}^{(2)} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & a_{nn}^{(n-1)} \end{pmatrix}. \tag{9.1.10}$$

Besides, that we can store the multiplicators m_{ji}, which are used to create the zero lower triangle, into a lower triangular matrix L:

$$L = \begin{pmatrix} 1 & 0 & \ldots & \ldots & 0 \\ m_{21} & 1 & \ddots & & \ldots \\ \vdots & m_{32} & \ddots & \ddots & \ldots \\ \vdots & & \ddots & \ddots & 0 \\ m_{n1} & m_{n2} & \ldots & m_{n,n-1} & 1 \end{pmatrix}. \tag{9.1.11}$$

One easily verifies that

$$A = LU. \tag{9.1.12}$$

Once we have constructed the L and U matrix, the solution of $A\mathbf{x} = \mathbf{b}$ is straight forward. Substitution of (9.1.12) gives

$$LU\mathbf{x} = \mathbf{b}. \tag{9.1.13}$$

Define $U\mathbf{x} = \mathbf{y}$, then the sequence of solving is:

$$L\mathbf{y} = \mathbf{b}, \qquad U\mathbf{x} = \mathbf{y}. \tag{9.1.14}$$

As the form of the matrices L and U is triangular, these equations can be solved immediately.

Exercise 9.1.5 *Let L have elements l_{ij} with*

$$l_{ij} = \begin{cases} 0, & \text{if } i < j, \\ 1, & \text{if } i = j, \\ l_{ij}, & \text{if } i > j. \end{cases} \tag{9.1.15}$$

Show that the solution of $L\mathbf{y} = \mathbf{b}$ is given by

$$y_i = b_i - \sum_{k=1}^{i-1} l_{ik} y_k. \tag{9.1.16}$$

□

Exercise 9.1.6 *Let U have elements u_{ij} with*

$$u_{ij} = \begin{cases} 0, & \text{if } i > j, \\ 1, & \text{if } i = j, \\ u_{ij}, & \text{if } i < j. \end{cases} \tag{9.1.17}$$

Show that the solution of $U\mathbf{x} = \mathbf{y}$ is given by

$$x_i = y_i - \sum_{k=i}^{n} u_{ik} x_k. \tag{9.1.18}$$

In which sequence do we have to compute x_i? □

An alternative way to compute the matrices L and U is by direct substitution. Define the matrices L and U as in Exercises 9.1.5 and 9.1.6. Since $A = LU$ we have

$$a_{ij} = \sum_{k=1}^{n} l_{ik} u_{kj} = \sum_{k=1}^{\min(i,j)} l_{ik} u_{kj}. \quad \text{Why?} \tag{9.1.19}$$

Exercise 9.1.7 *Show that l_{ij} and u_{ij} are defined by the following relations*

$$\begin{aligned} u_{ii} &= a_{ii} - \sum_{k=1}^{i-1} l_{ik} u_{ki}, \\ u_{ij} &= (a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}) / u_{ii}, \\ l_{ij} &= (a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj}) / u_{jj}. \end{aligned} \tag{9.1.20}$$

Give the sequence in which u_{ij} and l_{ij} can be computed. □

From this exercise we see that the LU-decomposition is unique iff the diagonal elements u_{ii} are non-zero. This is equivalent to having non-zero pivots during Gaussian elimination. In case the pivots are small it may be necessary to interchange rows and/or columns. However, in case of discretization methods, it is common practice to avoid pivoting, since this destroys the structure of the matrix. In many cases the discretization matrix has the property that pivoting is not required.

9.1.4 Band method

When we discretize a PDE on a rectangular structured grid, the matrix we get is always a band matrix, provided a natural numbering is chosen. The band width of such a matrix defines the amount of storage needed as well as the amount of work required to solve the system of equations.

Exercise 9.1.8 *Let the band width of the matrix* $\mathbf{A}$ *be equal to* $2b+1$, *i.e.* $a_{ij} = 0$ *if* $|i-j| > b$.
Prove by induction that $l_{ij} = 0$ *if* $i > j+b$ *and* $u_{ij} = 0$ *if* $j > i+b$. □

From Exercise 9.1.8 it follows that $\mathbf{L}$ and $\mathbf{U}$ are zero for elements outside the band. So it is indeed sufficient to store only the elements inside the band. Band matrices are stored column-wise, hence $(2b+1) \times n$ positions for non-symmetrical and $(b+1) \times n$ positions for symmetrical matrices are needed. So the storage of a typical non-symmetrical band matrix looks like:

$$\mathbf{A} = \begin{pmatrix} 0 & \dots & 0 & 0 & a_{11} & a_{12} & a_{13} & \dots & a_{1,1+b} \\ 0 & \dots & 0 & a_{21} & a_{22} & a_{23} & a_{24} & \dots & a_{2,2+b} \\ 0 & \dots & a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & \dots & a_{3,3+b} \\ \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ a_{n,n-b} & \dots & a_{n,n-2} & a_{n,n-1} & a_{nn} & 0 & 0 & \dots & 0 \end{pmatrix} \tag{9.1.21}$$

9.1.5 Profile method

Consider a matrix $\mathbf{A}$, which is the result of discretizing a PDE either by FEM, FVM or FDM. Let i be an arbitrary node in the grid with neighbors $j, k, l, \dots, m$ (Figure 9.1).

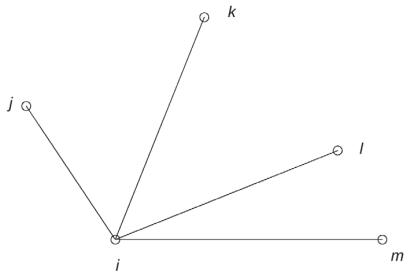


Figure 9.1: Nodal point i with neighbors.

Node i is connected to all of its neighbors, which implies that in general the elements $a_{ij}, a_{ik}, \dots, a_{im}$ are unequal to zero. If node n is not connected to i, then a_{in}=a_{ni}=0. Such elements in the matrix are called *essential zeros*. The fact that elements are essentially zero is a property of the grid and not of the specific PDE. Knowledge of essential zeros can be used to solve the system of equations efficiently. A typical example is the band method treated in Section 9.1.4. Another example is the *profile method*.

By *profile of a matrix* we mean the following:

Consider the i^{th} row. Let a_{ij} be the first essential non-zero element in this row counted from left to right. Hence j is the smallest column number in row i corresponding to an essential non-zero element. Then all elements $a_{ij}, a_{i,j+1}, \dots, a_{ii}$ belong to the *profile* or *envelope* of the matrix.

Consider on the other hand the i^{th} column of $\mathbf{A}$. Let j be the smallest row number in column j corresponding to an essential non-zero element. Then all elements $a_{ji}, a_{j,i+1}, \ldots, a_{ii}$ belong to the profile of the matrix. Mark that essential non-zero elements may be zero by coincidence. However, these elements are still considered to be non-zero. So actually a profile may be seen as a variable band. See Figure 9.2

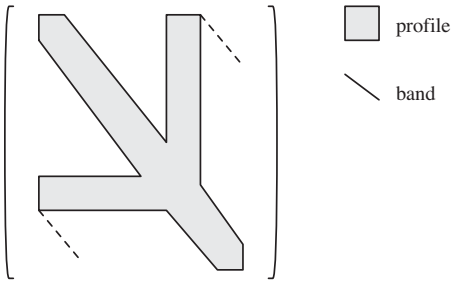


Figure 9.2: Example of a profile.

Exercise 9.1.9 *Show that the profile of a matrix arising from the discretization of a PDE is symmetrical.* □

Exercise 9.1.10 *Show that, if an LU-decomposition is applied, all elements outside the profile remain zero. Elements inside the profile of the L and U matrix will be non-zero in general .* □

From Exercise 9.1.10 it is clear that it is sufficient to store only those elements of the matrix that are *inside the profile*. Of course this requires a special storage scheme, otherwise the amount of memory needed is not better than for a band matrix. A standard storage method due to George [17] is the following one.

Let $\mathcal{L}$ be the lower triangle of the matrix $\mathbf{A}$ (without diagonal), $\mathcal{D}$ the diagonal and $\mathcal{U}$ the upper triangle. Hence

$$\mathbf{A} = \mathcal{L} + \mathcal{D} + \mathcal{U}. \tag{9.1.22}$$

The matrix $\mathbf{A}$ is stored in a one-dimensional array in the sequence

$$a_{11}, a_{21}, a_{22}, a_{12}, a_{31}, a_{32}, a_{33}, a_{23}, a_{13}, \ldots$$

where all the elements outside the profile are skipped. So the storage can be expressed as follows:
Start with diagonal element a_{11}.
Next store all elements of row 2 of $\mathcal{L}$ from left to right, followed by the diagonal element a_{22}, followed by all elements of column 2 of $\mathcal{U}$ from the diagonal to the top.
This process is repeated for all next rows and columns.
So the i^{th} row/column is stored as:

$$a_{i,p_i}, a_{i,p_i+1}, \ldots, a_{ii}, a_{i-1,i}, a_{i-2,i}, \ldots, a_{p_i,i},$$

with p_i the index of the first non-zero element in row i. In case of a symmetrical matrix, of course the upper triangle is not stored.
To keep track of the start of each new row, it is sufficient to store the position of the diagonal elements in the 1D array. This requires one extra integer array of length n (why?).

Exercise 9.1.11 *Show how to find an arbitrary element a_{ij} in the lower triangular matrix $\mathcal{L}$.* □

Exercise 9.1.12 *Show how to find an arbitrary element a_{ij} in the upper triangular matrix $\mathcal{U}$.* □

To perform an LU-decomposition on a profile matrix, it is necessary to apply an adapted method: the *profile method*. Special in this method is the sequence in which the elements of the LU-decomposition are computed. The sequence used is:

$$d_{11}, l_{21}, d_{22}, u_{12}, l_{31}, l_{32}, d_{33}, u_{23}, u_{13}, \ldots$$

So this is precisely the sequence of the matrix storage.

Exercise 9.1.13 *Give the formulas to compute* **L**, **D** *and* **U**, *utilizing the profile structure.* □

In the literature sometimes other names for the profile method are used, like *wave front method* and *frontal solution method*.
A simple example of a profile matrix is created by a one-dimensional problem with periodical boundary conditions as sketched in Figure 9.3. In this problem point i is connected to points $i-1$ and $i+1$ leading to a band width of 3. However, because of the periodical boundary conditions, point n and 1 have the same unknown and

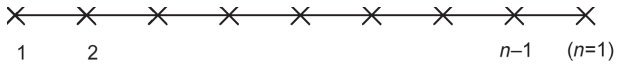


Figure 9.3: One-dimensional mesh, for problem with periodical boundary conditions.

point 1 is connected to both $n-1$ and 2. Point $n-1$ connected to $n-2$ and 1. The corresponding matrix gets the structure as sketched in Figure 9.4. The band width of this matrix is equal to $n-1$, which means that in case of a band storage, the matrix is full. The profile sketched in Figure 9.4b is much smaller.

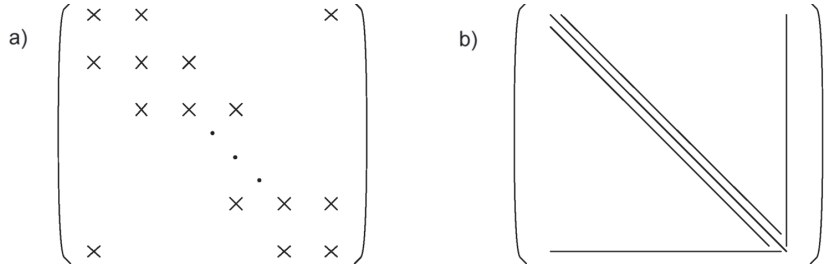


Figure 9.4: a) Non-zero pattern of one-dimensional problem with periodical boundary conditions, b) corresponding profile.

A good numbering may reduce the band width or the profile of the matrix considerably. In the next section we shall deal with a simple but effective renumbering algorithm.

9.1.6 Renumbering techniques

For finite element methods various renumbering algorithms have been constructed. Many of them are variants of the so-called Cuthill-McKee renumbering algorithm. The Cuthill-McKee [14] algorithm is a renumbering technique developed to reduce the band-width or envelope of a matrix. For an extended description see for example George and Liu [17]. Of course it is always possible to compute the optimal storage, but in general computation is so expensive that it takes more time than solving the system of equations. All renumbering techniques are therefore semi-optimal, in the sense that they try to optimize the storage, without performing to many operations.

The idea of Cuthill-McKee is that the local envelope of a matrix is minimal if all neighboring nodes have a number as close as possible to the node itself. Suppose we have a starting node or a set of starting nodes. This starting set has node numbers 1 to n. Then the idea is to give all direct neighbors of this starting set the node numbers from $n + 1$. This process is repeated until the complete set of nodes is exhausted. There are a number of variants of this algorithm all of which try to improve the envelope or profile, but the basic idea is the same.

Of course the difficult part of this process is how to find the starting set of nodes. This may be done automatically, which may be relatively difficult or by hand. In the last case one usually chooses a starting node (for example a corner node), or a starting curve.

Figure 9.5 shows the result of the first 9 steps of Cuthill-McKee in a rectangular grid. In the first step we start with the lower left point,indicated by a black circle. In the next step the three surrounding nodes are added (white circles). Each next set of nodes in the consecutive steps of the Cuthill-McKee algorithm, has been marked with a circle with different fill-in. In step 9 we arrive at a set of vertical nodes, which means that we are almost at an optimal numbering.

In *reversed Cuthill-McKee* we repeat the process in reversed order, starting with the last set of nodes found. In the example this will lead to a natural ordering.

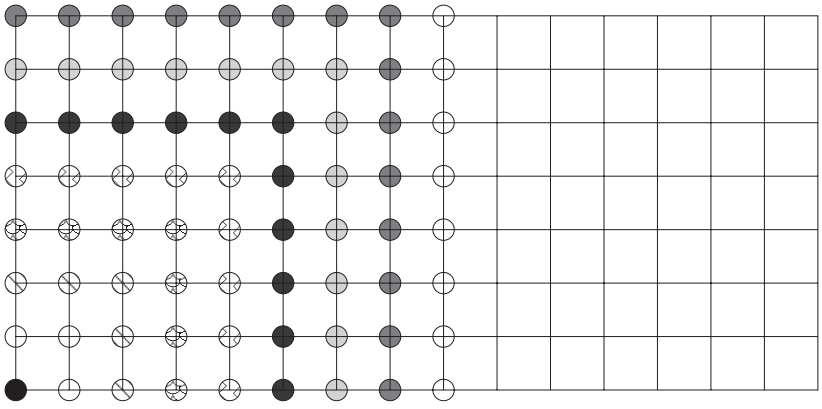


Figure 9.5: Example of the Cuthill-McKee algorithm.

9.2 Generic iterative process.

In this and the subsequent sections we shall consider *iterative methods*. We start out with a system of linear equations we want to solve: $A\mathbf{x} = \mathbf{b}$. Then we choose a *start value* $\mathbf{x}_0$ and we calculate the *residual* $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$. Now we have to improve $\mathbf{x}_0$ in some way, by adding a *correction* vector $\mathbf{c}_0$ to obtain a new estimate $\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{c}_0$. If we have been doing things right, the new residual $\mathbf{r}_1$ will in some way be smaller than $\mathbf{r}_0$. After that the process repeats itself, until the residual has become sufficiently small. Now the central question is of course: how to determine that correction step $\mathbf{c}_k$ to obtain $\mathbf{x}_{k+1}$ from $\mathbf{x}_k$?

Exercise 9.2.1 *Prove that when we take* $\mathbf{c}_k = A^{-1}\mathbf{r}_k$, $\mathbf{x}_{k+1}$ *solves* $A\mathbf{x} = \mathbf{b}$. □

So apparently $\mathbf{c}_k = A^{-1}\mathbf{r}_k$ would be the perfect choice, but unfortunately that is not easier to solve than our original system. However, it sets us on a trail. We may use an *approximation* $P^{-1}\mathbf{r}_k$ to $A^{-1}\mathbf{r}_k$ for $\mathbf{c}_k$. Such a matrix P is called a *preconditioner*. Different preconditioners generate different iterative methods.

9.3 Defect correction

9.3.1 Algorithm

The *defect correction* or *standard iteration* algorithm is the direct implementation of the above idea. It may be summarized in the following few lines of *pseudo code*

Defect correction algorithm

Presets: $\mathbf{x}_0 = 0; \mathbf{r}_0 = \mathbf{b}; k = 0$
while $\|\mathbf{r}_k\|_\infty > \varepsilon\|\mathbf{b}\|_\infty$ **do**
  $\mathbf{c}_k = P^{-1}\mathbf{r}_k$
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{c}_k$
  $\mathbf{r}_{k+1} = \mathbf{r}_k - A\mathbf{c}_k$
  $k = k + 1$
end while

Exercise 9.3.1 *Prove that the expression for* $\mathbf{r}_{k+1}$ *in this algorithm is equivalent to* $\mathbf{r}_{k+1} = \mathbf{b} - A\mathbf{x}_{k+1}$. □

Exercise 9.3.2 *Let* S *be an* $N \times N$ *matrix with* $s_{j,j+1} = 1, j = 1, \ldots, N-1, s_{jk} = 0$ *otherwise and let* I *be the identity matrix. We wish to solve* $A\mathbf{x} = \mathbf{f}$, *with* $A = 2I - S - S^T$ *and* $f_k = 1, k = 1, \ldots, N$. *Use defect correction with* $P^{-1} = \frac{1}{2}I$. *Use Matlab. Compare the number of iterations with N=10, N=100 and N=1000 to arrive at a residual with* $\|\mathbf{r}_k\|_\infty < 10^{-4}$. □

9.3.2 Convergence of defect correction

Let ξ be the solution to $A\mathbf{x} = \mathbf{b}$ and $\varepsilon_k = \xi - \mathbf{x}_k$ the error in the k-th iterate. We may combine various lines of Algorithm (9.3.1) to obtain

$$\mathbf{r}_{k+1} = (I - AP^{-1})\mathbf{r}_k, \tag{9.3.1}$$

$$\varepsilon_{k+1} = (I - P^{-1}A)\varepsilon_k. \tag{9.3.2}$$

Exercise 9.3.3 *Prove the two relations from Equation* (9.3.1) □

Exercise 9.3.4 *Prove that for any two matrices A, B that if*

$$\lim_{k\to\infty}(AB)^k = 0, \tag{9.3.3}$$

then

$$\lim_{k\to\infty}(BA)^k = 0. \tag{9.3.4}$$

□

Exercise 9.3.5 *Prove that* $\mathbf{r}_k = (I - AP^{-1})^k\mathbf{r}_0$. □

Exercise 9.3.6 *Prove that* $\varepsilon_k = (I - P^{-1}A)^k\varepsilon_0$. □

For the algorithm to converge, we must apparently have, that $\varepsilon_k \to 0$ or equivalently $\mathbf{r}_k \to 0$. This is expressed by the following theorem.

Theorem 9.3.1 *Let* $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq \lambda_N$ *be the eigenvalues of* $I - P^{-1}A$. *Then the defect correction method with matrix A and preconditioner* P^{-1} *converges if and only if* $|\lambda_1| < 1$.

Proof
We prove the theorem for non defect matrices. Assume $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N$ are the N linear independent eigenvectors of $I - P^{-1}A$, belonging to $\lambda_1, \lambda_2, \ldots, \lambda_N$. Now ε_0 can be written as a linear combination of those eigenvectors:

$$\boldsymbol{\varepsilon_0 = \sum_{j=1}^{N} \alpha_j \mathbf{v}_j,} \tag{9.3.5}$$

and since every multiplication with $I - P^{-1}A$ multiplies $\mathbf{v}_j$ with α_j we have:

$$\boldsymbol{\varepsilon_k = (I - P^{-1}A)^k\varepsilon_0 = \sum_{j=1}^{N} \alpha_j \lambda_j^k \mathbf{v}_j.} \tag{9.3.6}$$

Now since λ_1 is the largest eigenvalue in absolute value and $|\lambda_1| < 1$, each term in this sum will vanish eventually. Alternatively, if $\lambda_1 \geq 1$ the first term in the sum will never vanish. □

The value $|\lambda_1|$ is also called the *spectral radius* of the matrix $I - P^{-1}A$ and denoted $\rho(I - P^{-1}A)$.

9.3.3 Error estimate for defect correction

A closer look at Equation (9.3.6) reveals, that at the end of the day there will be only one term left in that sum: the first. Let us assume that λ_1 is an eigenvalue with multiplicity 1 and let us also assume there is no eigenvalue $-\lambda_1$. Then in the long run

$$\boldsymbol{\varepsilon_{k+1} \approx \lambda_1 \varepsilon_k.} \tag{9.3.7}$$

This enables us to estimate the error for a defect correction process.

Theorem 9.3.2 $\varepsilon_{k+1} \approx \frac{\lambda_1}{1-\lambda_1}(\mathbf{x}_{k+1} - \mathbf{x}_k)$.

Proof We subtract $\lambda_1\varepsilon_{k+1}$ from both sides of Equation (9.3.7) and note that $\varepsilon_k - \varepsilon_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k$ to obtain

$$\boldsymbol{(1 - \lambda_1)\varepsilon_{k+1} \approx \lambda_1(\mathbf{x}_{k+1} - \mathbf{x}_k),} \tag{9.3.8}$$

and dividing both sides by $1 - \lambda_1$ gives the result.

The miracle is, that we can estimate the error in terms of things we know, viz $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$. Things we know?? What about λ_1 then? A little patience, we are coming to that.

9.3.4 Estimate of the spectral radius

We subtract Equation (9.3.7) with index k from that with index $k+1$ to obtain

$$\boldsymbol{\varepsilon_{k+1} - \varepsilon_k \approx \lambda_1(\varepsilon_k - \varepsilon_{k-1}).} \tag{9.3.9}$$

But $\varepsilon_k - \varepsilon_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k$ and the above expression transforms into

$$\mathbf{x}_{k+1} - \mathbf{x}_k = \lambda_1(\mathbf{x}_k - \mathbf{x}_{k-1}). \tag{9.3.10}$$

Or letting $\mathbf{d}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$, we find by applying standard least squares technique:

$$\lambda_1 \approx \frac{(\mathbf{d}_{k+1}, \mathbf{d}_k)}{(\mathbf{d}_k, \mathbf{d}_k)}. \tag{9.3.11}$$

Exercise 9.3.7 *Show that for large k*

$$\varepsilon_{k+N} = \lambda_1^N \varepsilon_k. \tag{9.3.12}$$

Infer from this that to gain one decimal digit you need

$$N = -\frac{1}{^{10}\log \lambda_1} \tag{9.3.13}$$

iterations. □

9.3.5 M-matrices

An important class of matrices for which preconditioners can be constructed such that the defect correction iteration converges are *M-matrices*. These often occur in the context of partial differential equations.

Definition 9.3.1 *A matrix A is called an M-matrix if*

1. $a_{jk} \leq 0, \quad$ *if* $j \neq k$,
2. $A^{-1} \geq 0$.

Exercise 9.3.8 *Show that a diagonally dominant Z-matrix is an M-matrix. Use the discrete maximum principle of Chapter 3.* □

Exercise 9.3.9 *Show that an M-matrix has nonnegative diagonal elements. Use a contradiction argument. Assume $a_{kk} < 0$ and consider $A\mathbf{e}_k$, with $\mathbf{e}_k$ the k-th unit vector.* □

Exercise 9.3.10 *Show that an upper triangular Z-matrix is an M-matrix if the diagonal elements are positive.* □

If $A^{-1} \geq 0$ (so in particular if A is an M matrix) we can construct a convergent preconditioner as follows. Split A into

$$A = P - Q, \tag{9.3.14}$$

in which P^{-1} and Q have only nonnegative entries, also denoted by $P^{-1} \geq 0, Q \geq 0$. Such a splitting is called *regular*. A famous theorem by Varga [42] guarantees that $\rho(P^{-1}Q) < 1$ for regular splittings. In other words, the defect correction process converges with P as preconditioner.

Exercise 9.3.11 *Let the matrix $B \geq 0$. Show that for any eigenvalue λ of B with corresponding eigenvector* $\mathbf{v}$

$$(I - B)|\mathbf{v}| \leq (1 - |\lambda|)|\mathbf{v}|, \tag{9.3.15}$$

with $|\mathbf{v}|_i = |v_i|$. Show, that if in addition $(I - B)^{-1} \geq 0$ that

$$|\mathbf{v}| \leq (1 - |\lambda|)(I - B)^{-1}|\mathbf{v}|, \tag{9.3.16}$$

and hence that $|\lambda| < 1$. □

Exercise 9.3.12 *Prove Varga's theorem. Use the result of the previous exercise.* □

9.4 Classical preconditioners

In this section we shall introduce various classical preconditioners that have been in wide use. Today they are not used much any longer in a context of defect correction. The reason for that we will see in our chapter on Krylov space methods and Multi Grid methods.

9.4.1 Jacobi

The oldest and arguably simplest iterative method known to man is that of Jacobi. Let's write $A = D - L - U$ in which D is the diagonal, L the lower triangular part and U the upper triangular part of the matrix A. For *Jacobi's method* we take $P = D$. The iteration matrix becomes:

$$M = I - P^{-1}A = I - D^{-1}(D - L - U) = D^{-1}(L + U). \tag{9.4.1}$$

Exercise 9.4.1 *Show, that Jacobi's method may be written as*

$$D\mathbf{x}_{k+1} = (L + U)\mathbf{x}_k + \mathbf{b}. \tag{9.4.2}$$

□

Exercise 9.4.2 *Apply Jacobi's method to the problem of Exercise 9.3.2. Estimate λ_1 and show from the numerical results that $\lambda_1 = 1 - kh^2$, with k a positive constant.* □

Exercise 9.4.3 *Prove using Gershgorin's Theorem that Jacobi's method converges if A is diagonally dominant.* □

Exercise 9.4.4 *Prove that Jacobi's method converges if A is an M-matrix.* □

9.4.2 Gauss-Seidel

With the same notation as in the previous section we take $P = D - L$ to obtain the method of Gauss-Seidel. Note that we in general do not calculate P^{-1} itself, since the defect correction algorithm only requires us to solve the set $P\mathbf{c}_k = \mathbf{r}_k$. That is easy in this case, because it only requires backsubstitution.

Exercise 9.4.5 *Prove that Gauss-Seidel's method can be written as*

$$(D - L)\mathbf{x}_{k+1} = U\mathbf{x}_k + \mathbf{b}. \tag{9.4.3}$$

□

Exercise 9.4.6 *Prove that Gauss-Seidel's iteration matrix is given by*

$$M = I - P^{-1}A = I - (D-L)^{-1}(D-L-U) = (D-L)^{-1}U. \tag{9.4.4}$$

□

Exercise 9.4.7 *Prove that Gauss-Seidel converges if A is an M-matrix.* □

Gauss-Seidel's method converges for an important class of practical problems, the positive definite problems. Roughly all those that come from a minimization problem.

Theorem 9.4.1 *Let $A = D - L - L^T$ be positive definite. Then Gauss-Seidel's method converges.*

Proof
We have to show that all eigenvalues of $M = (D-L)^{-1}L^T$ are in absolute value less than 1. Let λ be an eigenvalue of M, with corresponding eigenvector $\mathbf{v}$. We have:

$$(D-L)^{-1}L^T\mathbf{v} = \lambda\mathbf{v}, \tag{9.4.5}$$

or equivalently

$$L^T\mathbf{v} = \lambda(D-L)\mathbf{v}. \tag{9.4.6}$$

This eigenvalue λ may be *complex* and the corresponding eigenvector will also be complex in that case. The conjugate complex quantities $\bar{\lambda}$ and $\bar{\mathbf{v}}$ will be eigenvalue and eigenvector too, since A is a real matrix. We define the ordinary inner product on complex spaces:

$$(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} \bar{x}_k y_k. \tag{9.4.7}$$

(Observe that $(\mathbf{v}, \mathbf{v}) > 0$ unless $\mathbf{v} = 0$ and that $(\mathbf{v}, A\mathbf{v}) > 0$ unless $\mathbf{v} = 0$.) Consider

$$\begin{aligned}
(\mathbf{v}, A\mathbf{v}) &= (\mathbf{v}, (D-L-L^T)\mathbf{v}), && (9.4.8)\\
&= (\mathbf{v}, (D-L)\mathbf{v}) - \lambda(\mathbf{v}, (D-L)\mathbf{v}), \text{by Equation (9.4.6)}, && (9.4.9)\\
&= (1-\lambda)(\mathbf{v}, (D-L)\mathbf{v}). && (9.4.10)
\end{aligned}$$

We also have:

$$\begin{aligned}
(\mathbf{v}, A\mathbf{v}) &= (\mathbf{v}, (D-L-L^T)\mathbf{v}), && (9.4.11)\\
&= (\mathbf{v}, (D-L^T)\mathbf{v}) - (\mathbf{v}, L\mathbf{v}), && (9.4.12)\\
&= (\mathbf{v}, (D-L^T)\mathbf{v}) - (L^T\mathbf{v}, \mathbf{v}), && (9.4.13)\\
&= (\mathbf{v}, (D-L^T)\mathbf{v}) - (\lambda(D-L)\mathbf{v}, \mathbf{v}), && (9.4.14)\\
&= (\mathbf{v}, (D-L^T)\mathbf{v}) - \bar{\lambda}(\mathbf{v}, (D-L^T)\mathbf{v}), && (9.4.15)\\
&= (1-\bar{\lambda})(\mathbf{v}, (D-L^T)\mathbf{v}). && (9.4.16)
\end{aligned}$$

Because A is positive definite λ cannot be equal to 1. Hence

$$\begin{aligned}
\left(\frac{1}{1-\lambda} - \frac{1}{1-\bar{\lambda}}\right)(\mathbf{v}, A\mathbf{v}) &= (\mathbf{v}, (2D-L-L^T)\mathbf{v}), && (9.4.17)\\
&= (\mathbf{v}, A\mathbf{v}) + (\mathbf{v}, D\mathbf{v}), && (9.4.18)
\end{aligned}$$

and therefore

$$\left(\frac{1}{1-\lambda} - \frac{1}{1-\bar{\lambda}} - 1\right)(\mathbf{v}, A\mathbf{v}) = (\mathbf{v}, D\mathbf{v}). \tag{9.4.19}$$

Because A is positive definite, so is D (why?) and we find

$$\frac{1}{1-\lambda} - \frac{1}{1-\bar{\lambda}} - 1 > 0, \tag{9.4.20}$$

$$\frac{1}{1-\lambda} - \frac{1}{1-\bar{\lambda}} > 1, \tag{9.4.21}$$

$$\frac{1-\bar{\lambda}+1-\lambda}{(1-\lambda)(1-\bar{\lambda})} > 1, \tag{9.4.22}$$

$$\frac{2-2\Re(\lambda)}{1-2\Re(\lambda)+|\lambda|^2} > 1, \tag{9.4.23}$$

in which $\Re(\lambda)$ denotes the real part of λ. The denominator in Inequality (9.4.23) is always positive (why?) hence

$$2-2\Re(\lambda) > 1-2\Re(\lambda)+|\lambda|^2, \tag{9.4.24}$$

$$|\lambda|^2 < 1. \tag{9.4.25}$$

□

9.4.3 Successive Overrelaxation SOR

Successive overrelaxation (SOR) has been devised as an improvement on Gauss-Seidel's method. The preconditioner of choice is $P = \frac{1}{\omega}(D - \omega L)$ in which $\omega > 0$ is a parameter that still has to be chosen. It comes down to multiplying the Gauss-Seidel correction *in each point* by ω before applying it. Since the back substitution in Gauss-Seidel uses already updated points this works recursively in a fairly complex way. The term *overrelaxation* really applies only for values $\omega > 1$, for $\omega < 1$ you have *underrelaxation.*

Exercise 9.4.8 *Show that the SOR process can be expressed as*

$$(D - \omega L)\mathbf{x}_{k+1} = ((1-\omega)D + \omega U)\mathbf{x}_k + \omega\mathbf{b}. \tag{9.4.26}$$

□

Exercise 9.4.9 *The* iteration matrix *M for a defect correction method is $I - P^{-1}A$. Show for the SOR iteration matrix M_ω:*

$$M_\omega = (D - \omega L)^{-1}((1-\omega)D + U). \tag{9.4.27}$$

□

Theorem 9.4.2 *If A is positive definite and $0 < \omega < 2$, then SOR converges.*

Exercise 9.4.10 *Prove Theorem 9.4.2 in the same way as Theorem 9.4.1* □

Apparently it is important how to choose ω. There are a number of theoretical results on that, that are valid for matrices A of a special structure: *diagonally block tridiagonal*. That is, the matrix consists of *blocks* and only the *diagonal*, *superdiagonal* and *subdiagonal* blocks may be nonzero. Moreover, the diagonal blocks must be diagonal matrices themselves.

Exercise 9.4.11 *Let V be a rectangle with a regular grid. A* checker board numbering *of the nodes is constructed as follows. The points are painted white and black alternately in the same pattern as the squares of a checkerboard. Now first the black points are numbered and after that the white points. Show that for the 5-point Laplace molecule the resulting matrix is a 2×2 block matrix with diagonal block matrices in the diagonal blocks.* □

Exercise 9.4.12 *Let V be a rectangle with a regular grid that has been obliquely numbered. Show that for the 5-point Laplace molecule the resulting matrix is diagonally block tridiagonal.* □

For diagonally blocktridiagonal matrices there is a functional relationship between the eigenvalues $\lambda_{\omega,k}$ of M_ω, the iteration matrix of SOR and the corresponding eigenvalues $\lambda_{1,k}$ of M_1 the iteration matrix of Gauss-Seidel.

$$\left(\frac{\lambda_\omega - 1 + \omega}{\omega}\right)^2 = \lambda_\omega \lambda_1. \tag{9.4.28}$$

For a proof see [5].

Now let $|\lambda_{1,1}| = \rho(M_1)$ the spectral radius of the Gauss-Seidel iteration matrix. For diagonally block tridiagonal matrices A the following expression for the optimal value of ω holds:

$$\omega_{\text{opt}} = \frac{2}{1 + \sqrt{1 - \lambda_{1,1}}}. \tag{9.4.29}$$

So to estimate the optimum value of ω we have to know the value of $\lambda_{1,1}$. It is possible though, to estimate this value during the SOAR process using Equation (9.3.11) to estimate $\lambda_{\omega,1}$ and subsequently use Equation (9.4.28) to estimate $\lambda_{1,1}$ Care has to be taken however, that the eigenvalue belonging to the spectral radius does not become complex, because in that case the use of estimate (9.3.11) is no longer justified.

Exercise 9.4.13 *Let A be a real matrix with complex eigenvalues λ_1 and $\bar{\lambda}_1$ such that $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \ldots$. Show that* two *components survive in the error:*

$$\varepsilon_k \approx a_1 \lambda_1^k \mathbf{v}_1 + \bar{a}_1 \bar{\lambda}^k \bar{\mathbf{v}_1}. \tag{9.4.30}$$

□

In the remaining exercises of this section you may assume that A is a positive definite diagonally blocktridiagonal matrix.

Exercise 9.4.14 *Show from Equation* (9.4.28) *that $\lambda_{\omega,k}$ is complex if*

$$\lambda_{1,k} < \frac{4(\omega - 1)}{\omega^2}. \tag{9.4.31}$$

Show from Equation (9.4.28) *that if $\lambda_{\omega,k}$ is complex, then $|\lambda_{\omega,k}| = |\omega - 1|$. (Hint: What is the product of the roots of a quadratic equation)* □

Exercise 9.4.15 *Show that when $\omega = \omega_{\text{opt}}$ that*

$$\lambda_{1,1} = \frac{4(\omega - 1)}{\omega^2}. \tag{9.4.32}$$

Show using Exercise 9.4.14 that when $\omega > \omega_{\text{opt}}$ all eigenvalues of M_ω lie on a circle in the complex plane with radius $\omega - 1$. □

Exercise 9.4.16 *Suppose $\lambda_1 = 0.9999$. Estimate the number of iterations to gain one decimal digit using Gauss-Seidel. (Use Exercise 9.3.7) Calculate $\lambda_{\omega,1} = 1 - \omega_{\text{opt}}$ and estimate the number of iterations to gain one decimal using optimal SOAR.* □

9.4.4 Block variations

Block variations of Jacobi, Gauss Seidel and SOAR can be used if the matrix A in the system of equations $A\mathbf{x} = \mathbf{b}$ is a block matrix and the *diagonal* blocks can be solved easily, for example if they are tridiagonal. The convergence properties of block variations are a bit better than those of the standard methods.

9.4.5 Operation count

In numerical approximations of PDE's the number of unknowns per equation is fixed. In that case it is easy to estimate the operation count *per iteration*. Let N be the number of unknowns, then clearly the number of operations for a matrix vector multiplication takes kN operations (multiplication + addition), with k the number of unknowns per equation. Solving the preconditioner equation takes mN operations, $m < k$. If we take the two matrix additions into account and the calculation of $\|r_k\|$ for the stop criterion we end up with $(k + m + 3)N$ operations per iteration.

How many iterations do we need? Basically this depends on the accuracy we demand, but a good measure for that is the number of iterations n to gain one extra accurate decimal. This number n is given by (see Exercise 9.3.7)

$$n = -\frac{1}{^{10}\log \rho}, \tag{9.4.33}$$

or using natural logarithms

$$n = -\frac{\ln 10}{\ln \rho} = -\frac{2.3}{\ln \rho}, \tag{9.4.34}$$

with ρ the spectral radius of the iteration matrix $I - P^{-1}A$. This spectral radius is for PDE matrices and the Jacobi and Gauss-Seidel methods $1 - k_p h^2$. Here k_p is a constant depending on the problem, the form of the region and the method.

Exercise 9.4.17 *Show, using Equation* (9.4.29) *and Exercise 9.4.15 that for diagonal block-tridiagonal matrices the spectral radius of the SOAR iteration matrix for optimal ω is $\rho = 1 - k'_p h$.* □

Since roughly $h^{-1} = N^{\frac{1}{2}}$ for two dimensional problems and $h^{-1} = N^{\frac{1}{3}}$ for three dimensional problems we find

$$n = \frac{2.3}{k_p h^2} = K_p N, \tag{9.4.35}$$

for 2D Jacobi and Gauss Seidel and

$$n = \frac{2.3}{k_p h} = K_p N^{\frac{1}{2}}, \tag{9.4.36}$$

for 2D optimal SOAR. Since these are the number of iterations and the operation count per iteration is $O(N)$ the total operation count to gain one digit (2D) is $nO(N) = KN^2$ for Gauss Seidel and Jacobi and $nO(N) = KN^{\frac{3}{2}}$ for optimal SOAR. K depends on the method. For 3D these numbers are: $KN^{\frac{5}{3}}$ and $KN^{\frac{4}{3}}$ respectively.

9.5 Krylov Space Methods

Our treatment of Krylov space methods has to be superficial. We shall only consider CG in detail and out of the numerous other possibilities we shall only present BiCG-Stab in algorithmic form. The reader who wants to have a more thorough understanding of the subject should consult [41].

9.5.1 Introduction

We first consider simple standard iteration without preconditioning on $A\mathbf{x} = \mathbf{b}$.

Let us consider the form of the error after $n+1$ iterations (see Exercises 9.3.5 and 9.3.6):

$$\mathbf{r}_{n+1} = (I - A)^{n+1}\mathbf{r}_0 = P_{n+1}(A)\mathbf{r}_0, \tag{9.5.1}$$

$$\varepsilon_{n+1} = (I - A)^{n+1}\varepsilon_0 = P_{n+1}(A)\varepsilon_0. \tag{9.5.2}$$

$P_{n+1}(A)$ is an $n+1$-st degree matrix polynomial. Let us assume, that A has eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$ and corresponding eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$. We may take $\mathbf{x}_0 = 0$ and $\mathbf{r}_0 = \mathbf{b}$ without loss of generality and express $\mathbf{r}_0$ as a linear combination of eigenvectors:

$$\mathbf{r}_0 = \sum_{k=1}^{N} \alpha_k \mathbf{v}_k, \tag{9.5.3}$$

and because $A\mathbf{v}_k = \lambda_k \mathbf{v}_k$ we have

$$\mathbf{r}_n = (I - A)^n r_0 = \sum_{k=1}^{N} \alpha_k (1 - \lambda_k)^n \mathbf{v}_k, \tag{9.5.4}$$

and in general for any matrix polynomial $P_n(A)$:

$$P_n(A)\mathbf{r}_0 = \sum_{k=1}^{N} \alpha_k P_n(\lambda_k)\mathbf{v}_k. \tag{9.5.5}$$

Immediately we deduce that for convergence of the standard iteration all λ_k must be within a circle in the complex plane with radius 1 and real midpoint 1 (see Figure 9.6) The polynomial $(1-\lambda)^n$ is not specifically chosen to make the residual as small as possible. On the contrary, let us draw a picture of $(1-x)^{10}$ on the interval $(0, 2)$.
You can see, that for eigenvectors with eigenvalues between 0.4 and 1.6 there is a very good convergence, but for eigenvalues close to 0 and close to 2 the convergence is rather bad.

There are two things that could be done about that. First you could try to find a better polynomial than $(1-\lambda)^n$. That is what *Krylov (sub)space methods* do. Secondly you could try to treat the remaining parts of the spectrum differently. That is what *Multigrid methods* do.

9.5.2 The Krylov Space

The *Krylov subspace of dimension k* is spanned by the vectors $\mathbf{r}_0, A\mathbf{r}_0, A^2\mathbf{r}_0, \ldots, A^{k-1}\mathbf{r}_0$ and denoted by $\mathcal{K}^k(A; \mathbf{r}_0)$.

Exercise 9.5.1 *Show that in standard iteration* $\mathbf{r}_k \in \mathcal{K}^{k+1}(A; \mathbf{r}_0)$. *Infer from this that* $\mathbf{x}_{k+1} \in \mathcal{K}^{k+1}(A; \mathbf{r}_0)$. □

Krylov subspace methods all try to optimize the approximate solution in the k-dimensional subspace $\mathcal{K}^k(A; \mathbf{r}_0)$. This can be done in various ways of which we will consider only two: Conjugate Gradients and Bi-CGStab.

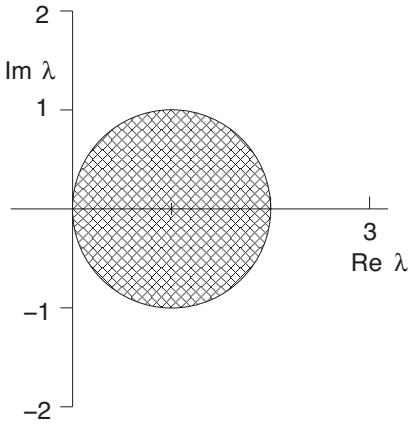


Figure 9.6: Region of convergence.

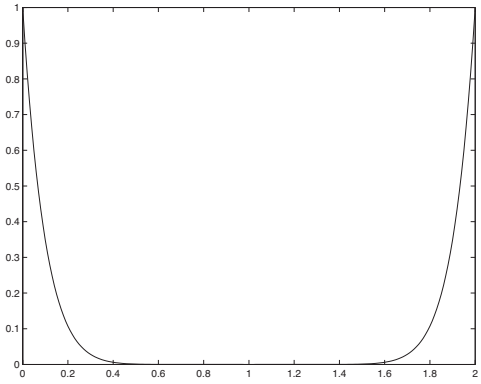


Figure 9.7: Graph of $(1-x)^{10}$.

9.5.3 Conjugate Gradients

The Conjugate Gradient method can best be explained for *A positive definite*. In that case solving $A\mathbf{x} = \mathbf{b}$ is equivalent to a minimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x}. \tag{9.5.6}$$

Exercise 9.5.2 *Show that for positive definite matrices A these formulations are equivalent. Use the same type of argument as in Chapter 5.* □

Starting from $\mathbf{x}_0 = 0$, $\mathbf{r}_0 = \mathbf{b}$ we now seek $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k$, with $\mathbf{s}_j \in \mathcal{K}^j(A;\mathbf{b})$ and solve the minimization problem in the Krylov subspace $\mathcal{K}^k$

$$\min_{\mathbf{x}\in\mathcal{K}^k} \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x}. \tag{9.5.7}$$

Exercise 9.5.3 *Let $A\xi = \mathbf{b}$, in other words ξ is the solution to the linear system. Show that Problem 9.5.7 is equivalent to*

$$\min_{\mathbf{x}\in\mathcal{K}^k} \frac{1}{2}(\mathbf{x}-\xi)^T A(\mathbf{x}-\xi). \tag{9.5.8}$$

□

Exercise 9.5.4 *Let $A\xi = \mathbf{b}$, in other words ξ is the solution to the linear system. Show that Problem 9.5.7 is equivalent to*

$$\min_{\mathbf{x}\in\mathcal{K}^k} \frac{1}{2}\mathbf{r}^T A^{-1}\mathbf{r}, \tag{9.5.9}$$

in which $\mathbf{r} = \mathbf{b} - A\mathbf{x}$. □

So in a way we do the best we can, but you may ask yourself the question if this is really an easier problem than the original one.

Let us write $\mathbf{x}_k = \sum_{j=0}^{k-1} \alpha_j \mathbf{s}_j$. Let us solve minimization Problem 9.5.7. This is just Ritz's method, so what we get is a $k \times k$ set of equations of the form

$$\sum_{j=0}^{k-1} \sigma_{mj}\alpha_j = \beta_m, \quad m = 0, 1, \ldots, k-1, \tag{9.5.10}$$

in which

$$\sigma_{mj} = (\mathbf{s}_m, A\mathbf{s}_j), \tag{9.5.11a}$$
$$\beta_m = (\mathbf{s}_m, \mathbf{b}), \tag{9.5.11b}$$

or in matrix vector notation $\Sigma\alpha = \beta$.

Exercise 9.5.5 *Explain that Equations* (9.5.11) *are an analogue for Galerkin's method.* □

Exercise 9.5.6 *Prove that $\mathbf{r}_k$ is orthogonal to span $\mathbf{s}_0, \mathbf{s}_2, ..., \mathbf{s}_{k-1}$, hence orthogonal to $\mathcal{K}^k(A; \mathbf{b})$.* □

The remark could be made that this is not much of an iterative method. That is right, so far it is not. But we have some freedom left in the choice of $\mathbf{s}_j$ which will make it one. If you consider Equation (9.5.10) you will see, that in every new step *all α_j will change*, unless you make the matrix Σ *diagonal*. In that case the addition of a new dimension to the Krylov subspace will not touch already calculated α_j's . And *that* makes it a truly iterative method. So we have to make sure that $(\mathbf{s}_k, A\mathbf{s}_j) = 0, k \neq j$.

9.5.4 CG algorithm

We summarize all this in the following algorithm:
Conjugate Gradient Algorithm (CG)
Require: A positive definite
1: Presets: $\mathbf{x}_0 = 0$, $\mathbf{r}_0 = \mathbf{b}$, $\mathbf{s}_0 = \mathbf{r}_0$, $k = 0$
2: **while** $\|\mathbf{r}_k\| > \varepsilon\|\mathbf{b}\|$ **do**
3: $\alpha_k = (\mathbf{r}_k, \mathbf{r}_k)/(\mathbf{s}_k, A\mathbf{s}_k)$ {See Exercise 9.5.8}
4: $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k\mathbf{s}_k$
5: $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{s}_k$
6: $\beta_k = (\mathbf{r}_{k+1}, \mathbf{r}_{k+1})/(\mathbf{r}_k, \mathbf{r}_k)$
7: $\mathbf{s}_{k+1} = \mathbf{r}_{k+1} + \beta_k\mathbf{s}_k$
8: $k = k + 1$
9: **end while**

The algorithm generates residual vectors $\mathbf{r}_k$ in subsequent Krylov spaces and they are *mutually orthogonal*. This follows in fact from Galerkin's condition (see Exercise 9.5.6). Moreover, the search directions $\mathbf{s}_k$ are mutually $A-$orthogonal ensuring the diagonality of the Galerkin matrix Σ. The positive definiteness of A guarantees that the algorithm will not crash: the denominator of α_k cannot vanish.

The proof of these claims is the subject of the subsequent exercises.

Exercise 9.5.7 *Show from lines 5 and 7, that if* $\mathbf{r}_k, \mathbf{s}_k \in \mathcal{K}^m(A;\mathbf{b})$ *then* $\mathbf{r}_{k+1}, \mathbf{s}_{k+1} \in \mathcal{K}^{m+1}(A;\mathbf{b})$*, independent of the values of* α_k *and* β_k*. Infer by induction that* $\mathbf{r}_k, \mathbf{s}_k \in \mathcal{K}^{k+1}(A;\mathbf{b})$ *for all* k*.*

Explain that if

$$\mathcal{K}^{k+1}(A;\mathbf{b}) \subset \text{span}\,\{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_k\}, \tag{9.5.12}$$

then

$$\mathcal{K}^{k+2}(A;\mathbf{b}) \subset \text{span}\,\{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{k+1}\} \text{ if } \alpha_k \neq 0. \tag{9.5.13}$$

(Hint: it is sufficient to show that $A^{k+1}\mathbf{b} \in \text{span}\,\{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{k+1}\}$*. Use lines 5 and 7 of the algorithm.)* □

Exercise 9.5.8 *Assume*

$$(\mathbf{r}_k, \mathbf{v}) = 0,\ \forall \mathbf{v} \in \mathcal{K}^k(A;\mathbf{b}) \quad \textit{(Galerkin's condition)} \tag{9.5.14}$$

and

$$(\mathbf{s}_k, A\mathbf{v}) = 0,\ \forall \mathbf{v} \in \mathcal{K}^k(A;\mathbf{b}) \quad \textit{(diagonality condition)}. \tag{9.5.15}$$

1. *Show that by line 5 of the algorithm* $(\mathbf{r}_{k+1}, \mathbf{v}) = 0,\ \forall \mathbf{v} \in \mathcal{K}^k(A;\mathbf{b})$ independent of α_k*. Infer from this and the previous exercise, that to satisfy Galerkin's condition for* $\mathcal{K}^{k+1}$ *it is sufficient to ensure that* $(\mathbf{r}_{k+1}, \mathbf{r}_k) = 0$*. Show that line 3 of the algorithm does just that. Use line 7 and the diagonality condition.*

2. *Show that by line 7 of the algorithm* $(\mathbf{s}_{k+1}, A\mathbf{v}) = 0,\ \forall \mathbf{v} \in \mathcal{K}^k$ independent of β_k*. Use the diagonality condition and the fact that if* $\mathbf{v} \in \mathcal{K}^k$ *then* $A\mathbf{v} \in \mathcal{K}^{k+1}$*. Infer from this and the previous exercise that to satisfy the diagonality condition for* $\mathcal{K}^{k+1}$ *it is sufficient to ensure that* $(\mathbf{s}_{k+1}, A\mathbf{s}_k) = 0$*. Show that line 6 of the algorithm does just that. Use lines 5 and 3.*

□

Exercise 9.5.9 *From Equation* (9.5.11) *you would expect*

$$\alpha_k = (\mathbf{s}_k, \mathbf{b})/(\mathbf{s}_k, A\mathbf{s}_k) \tag{9.5.16}$$

on line 3. Show that

$$(\mathbf{s}_k, \mathbf{b}) = (\mathbf{s}_k, \mathbf{r}_k). \tag{9.5.17}$$

(Use the diagonality condition). Next show that

$$(\mathbf{s}_k, \mathbf{r}_k) = (\mathbf{r}_k, \mathbf{r}_k). \tag{9.5.18}$$

(Use line 7 of the algorithm) □

Exercise 9.5.10 *Various inner products in the CG algorithm are used multiple times. It is a waste to calculate them each time anew. The same is true for the matrix vector product* $A\mathbf{s}_k$*. Reformulate the algorithm in such a way that each iteration only needs two inner products and one matrix vector multiplication.* □

9.5.5 Preconditioning

In practice CG is always used with a preconditioner P, but we have to be careful. It is not a good idea to apply the algorithm to the preconditioned system $P^{-1}A\mathbf{x} = P^{-1}\mathbf{b}$, because in general $P^{-1}A$ will no longer be symmetric even if P is. If we have a *factorization* of P:

$$P = LL^T, \tag{9.5.19}$$

we can construct a symmetric preconditioned system as follows:

$$L^{-1}AL^{-1^T}\mathbf{y} = L^{-1}\mathbf{b}. \tag{9.5.20}$$

This has various drawbacks, most notably, that the preconditioner must be available in factored form and that the solution must be backtransformed later on. There is a better way to go about this. We have defined CG with respect to the classical inner product $(\mathbf{x}, \mathbf{y}) = \sum x_k y_k$ but in fact every positive definite matrix B generates an inner product $(\mathbf{x}, \mathbf{y})_B = (\mathbf{x}, B\mathbf{y})$.

Exercise 9.5.11 *Show that $(.,.)_B$ is a proper inner product.* □

Exercise 9.5.12 *Show that with P and A symmetric positive definite $(P^{-1}A\mathbf{x}, \mathbf{y})_P = (\mathbf{x}, P^{-1}A\mathbf{y})_P$.* □

Using this inner product we may formulate the *preconditioned CG algorithm*:

Require: A, P positive definite
Presets: $\mathbf{x}_0 = 0$, $\mathbf{r}_0 = \mathbf{b}$, $\mathbf{t}_0 = \mathbf{s}_0 = P^{-1}\mathbf{r}_0$, $k = 0$
while $(\mathbf{r}_k, \mathbf{t}_k) > \varepsilon^2(\mathbf{r}_0, \mathbf{t}_0)$ **do**
  $\alpha_k = (\mathbf{r}_k, \mathbf{t}_k)/(\mathbf{s}_k, A\mathbf{s}_k)$
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$
  $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{s}_k$
  Solve $P\mathbf{t}_{k+1} = \mathbf{r}_{k+1}$
  $\beta_k = (\mathbf{r}_{k+1}, \mathbf{t}_{k+1})/(\mathbf{r}_k, \mathbf{t}_k)$
  $\mathbf{s}_{k+1} = \mathbf{t}_{k+1} + \beta_k \mathbf{s}_k$
  $k = k + 1$
end while

The algorithm uses ordinary inner products, but careful analysis will show that it is in fact CG applied to $P^{-1}A\mathbf{x} = P^{-1}\mathbf{b}$, with the $(.,.)_P$ inner product.

Exercise 9.5.13 *Show that minimizing $(\varepsilon, P^{-1}A\varepsilon)_P$ is the same as minimizing $(\varepsilon, A\varepsilon)$.* □

Exercise 9.5.14 *Show that the preconditioned CG as described above minimizes over the Krylov space $\mathcal{K}^k(P^{-1}A; P^{-1}\mathbf{b})$.* □

Exercise 9.5.15 *(Symmetric Gauss Seidel preconditioner) Let $A = D - L - L^T$ be positive definite, with D diagonal and L lower triangular. Show that $P = (D - L)D^{-1}(D - L^T)$ is positive definite and symmetric.Show that $P\mathbf{t} = \mathbf{r}$ can be solved in three easy steps, of which the first is: solve $(D - L)\mathbf{y} = \mathbf{r}$.* □

Exercise 9.5.16 *Show that the symmetric Gauss Seidel preconditioner generates a regular splitting if A is an M-matrix.* □

Exercise 9.5.17 *Do Exercise 9.5.10 for the preconditioned CG algorithm.* □

9.5.6 Convergence

In theory CG is a finite algorithm (why?) but that is not the reason for its usefulness. Also as an iteration process it has very good properties. In order to understand that we take a closer look at approximations in the Krylov space. Elements of the Krylov space $\mathcal{K}^{k+1}(A;\mathbf{b})$ can be written as

$$\mathbf{y} = \sum_{j=0}^{k} a_j A^j \mathbf{b} = Q_k(A)\mathbf{b}, \tag{9.5.21}$$

in which Q_k is a k^{th} degree polynomial. In CG approximations $a_0 = 1$ hence $Q(0) = 1$ and in fact, the i^th iteration can be written as

$$\varepsilon_i = Q_i(A)\varepsilon_0, \tag{9.5.22}$$

in which $\varepsilon_i = \mathbf{x} - \mathbf{x}_i$ is the error in the i^{th} iteration step. Since (see Exercise 9.5.3) CG minimizes $(\varepsilon_i, A\varepsilon_i)$, or equivalently $(\varepsilon_i, \varepsilon_i)_A$. Apparently the expression

$$(Q_i(A)\varepsilon_0, Q_i(A)\varepsilon_0)_A \tag{9.5.23}$$

is minimized over all possible polynomials Q_i of degree i with $Q_i(0) = 1$. Let us call this optimal polynomial P_i. Let $\lambda_1, \lambda_2, \ldots, \lambda_N$ be the eigenvalues of A in increasing order, with corresponding eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N$. We have:

$$\mathbf{b} = \sum_{j=1}^{N} b_j \mathbf{v}_j, \tag{9.5.24}$$

$$\mathbf{r}_i = P_i(A)\mathbf{r}_0 = \sum_{j=1}^{N} P_i(\lambda_j) b_j \mathbf{v}_j, \tag{9.5.25}$$

$$\varepsilon_i = A^{-1}\mathbf{r}_i = \sum_{j=1}^{N} P_i(\lambda_j)/\lambda_j b_j \mathbf{v}_j, \tag{9.5.26}$$

$$(\varepsilon_i, \varepsilon_i)_A = \sum_{j=1}^{N} P_i^2(\lambda_j)/\lambda_j b_j^2. \tag{9.5.27}$$

By comparing P_i with specific polynomials we may obtain an error estimate.

$$(\varepsilon_i, \varepsilon_i)_A = \sum_{j=1}^{N} P_i^2(\lambda_j)/\lambda_j b_j^2 \leq \sum_{j=1}^{N} Q_i^2(\lambda_j)/\lambda_j b_j^2, \tag{9.5.28}$$

$$\leq \max_{\lambda_j} Q_i^2(\lambda_j) \sum_{j=1}^{N} b_j^2/\lambda_j = \max_{\lambda_j} Q_i^2(\lambda_j)(\varepsilon_0, \varepsilon_0)_A. \tag{9.5.29}$$

Q_i is an arbitrary polynomial with $Q_i(0) = 1$.

Exercise 9.5.18 *If A has an eigenvalue with multiplicity $N-1$ then CG needs two iterations to find the exact solution. Explain why.* □

Exercise 9.5.19 *Explain why it is a good thing to have clusters of eigenvalues and a bad thing to have the eigenvalues evenly spread out over the spectrum.* □

We now derive a famous upperbound for the error in the $\|.\|_A$ norm by using scaled, shifted and mirrored Chebyshev polynomials. Chebyshev polynomials $T_n(x)$ are connected to $\cos(n\phi)$. You can expand $\cos(n\phi)$ into an n-th degree polynomial in $\cos\phi$. See exercise 9.5.20

Exercise 9.5.20 *Show that* $\cos n\phi$ *satisfies the following recurrence relation:*

$$\cos(n+1)\phi = 2\cos\phi\cos n\phi - \cos(n-1)\phi, \quad n \geq 2. \tag{9.5.30}$$

Infer from this and $\cos 0 = 1$, *that* $\cos n\phi$ *is a polynomial in* $\cos\phi$ □

$T_n(x)$ is now obtained by substituting $\cos\phi = x$ into that polynomial. That explains its behavior for $-1 \leq x \leq 1$: $|T_n(x)| \leq 1, \quad -1 \leq x \leq 1$. And for values outside that interval? The general solution for the recurrence relation:

$$T_{n+1} = 2xT_n - T_{n-1}, \quad T_0 = 1, T_1 = x, \tag{9.5.31}$$

is given by

$$T_n(x) = \frac{1}{2}((x + \sqrt{x^2-1})^n + (x + \sqrt{x^2-1})^{-n}). \tag{9.5.32}$$

Exercise 9.5.21 *Show that the general solution* u_n *of the recurrence relation*

$$u_{n+1} = 2xu_n - u_{n-1} \tag{9.5.33}$$

satisfies $u_n = A\rho_1^n + B\rho_2^n$ *with* A *and* B *arbitrary and* ρ_1 *and* ρ_2 *the solutions of the quadratic equation* $\rho^2 - 2x\rho + 1 = 0$. *Calculate from this the expression in Equation* (9.5.32).
Clearly this expression is valid for $x \geq 1$. *Is it also valid for* $x < 1$? □

We now cleverly take as our comparison polynomial:

$$Q_i(\lambda) = \frac{T_i(\frac{2\lambda-(\lambda_1+\lambda_N)}{\lambda_1-\lambda_N})}{T_i(-\frac{\lambda_1+\lambda_N}{\lambda_1-\lambda_N})}, \tag{9.5.34}$$

in which λ_1 is the minimum and λ_N the maximum eigenvalue of A. Observe, that we have scaled in such a way that $Q_i(0) = 1$. Let us introduce the condition number $K = \lambda_N/\lambda_1$ and the quantity $B = (K+1)/(K-1)$. By inequality (9.5.29) we have

$$(\varepsilon_i, \varepsilon_i)_A \leq \max_{\lambda_j} Q_i^2(\lambda_j)(\varepsilon_0, \varepsilon_0)_A, \tag{9.5.35}$$

$$\leq \frac{1}{T_i^2(B)}(\varepsilon_0, \varepsilon_0)_A, \tag{9.5.36}$$

$$\leq \frac{2}{(B + \sqrt{B^2-1})^i}(\varepsilon_0, \varepsilon_0)_A, \tag{9.5.37}$$

$$\leq 2(B - \sqrt{B^2-1})^i(\varepsilon_0, \varepsilon_0)_A, \tag{9.5.38}$$

$$\leq 2\left(\frac{\sqrt{K}-1}{\sqrt{K}+1}\right)^i(\varepsilon_0, \varepsilon_0)_A. \tag{9.5.39}$$

This famous error estimate is on the pessimistic side, because as the process continues the *effective* condition number will get gradually smaller, because the extremal eigenvalues of the spectrum will have been sufficiently well approximated by the polynomial P_i. Compared to the standard iteration methods CG performs at least as well as SOAR, but is applicable to more general matrices. In fact if we use a preconditioner like Incomplete LU (See section 9.5.8.2) CG will outperform SOAR by a considerable margin.

9.5.7 Krylov space methods for non symmetric matrices.

9.5.7.1 Bi-CG and CGS

The Bi-CG method can be viewed ([41], pg 98) as an application of the preconditioned CG method on the problem

$$\begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{x}} \\ \mathbf{x} \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ \hat{\mathbf{b}} \end{pmatrix}, \tag{9.5.40}$$

with preconditioner $P = \left(\begin{smallmatrix} 0 & I \\ I & 0 \end{smallmatrix}\right)$. $\hat{\mathbf{b}}$ must be some suitably chosen vector. The big problem with that is, that neither the system matrix $B = \left(\begin{smallmatrix} 0 & A \\ A^T & 0 \end{smallmatrix}\right)$nor the preconditioner P is positive definite, so the inner product $(.,.)_P$ is not a proper inner product and the algorithm may break down.

The CGS method tries to improve the convergence speed of Bi-CG by a clever trick on the matrix polynomials $P_i(A)$. For a derivation we refer once more to [41]. If both algorithms converge, CGS converges about twice as fast for the same operation count per iteration. CGS unfortunately has a rather irregular convergence behavior.

9.5.7.2 BiCG-stab

BiCG-stab stabilizes the convergence behavior of CGS and maintains the improved convergence of this method. For the derivation of this method, which is well beyond the scope of this book see [41]. We shall just present the algorithm.

BiCG-Stab without preconditioning
Presets: $\mathbf{x}_0$, $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$, $\tilde{\mathbf{r}} \neq 0$, $k = 0$, $\beta_0 = 0$, $\mathbf{p}_0 = 0$, $\omega_0 = 1$, $\mathbf{v}_0 = 0$.
$\rho_0 = (\tilde{\mathbf{r}}, \mathbf{r}_0)$
while not converged **do**
  if $\rho_k = 0$ or $\omega_k = 0$ **then**
    Break {Failure}
  end if
  $\mathbf{p}_{k+1} = \mathbf{r}_k + \beta_k(\mathbf{p}_k - \omega_k \mathbf{v}_k)$
  $\mathbf{v}_{k+1} = A\mathbf{p}_{k+1}$
  $\alpha_{k+1} = \rho_k / (\tilde{\mathbf{r}}, \mathbf{v}_{k+1})$
  $\mathbf{s} = \mathbf{r}_k - \alpha_{k+1}\mathbf{v}_{k+1}$
  $\mathbf{t} = A\mathbf{s}$
  $\omega_{k+1} = (\mathbf{t}, \mathbf{s}) / (\mathbf{t}, \mathbf{t})$
  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_{k+1}\mathbf{p}_{k+1} + \omega_{k+1}\mathbf{s}$
  $\mathbf{r}_{k+1} = \mathbf{s} - \omega_{k+1}\mathbf{t}$
  $\rho_{k+1} = (\tilde{\mathbf{r}}, \mathbf{r}_{k+1})$
  $\beta_{k+1} = (\rho_{k+1}/\rho_k)(\alpha_{k+1}/\omega_{k+1})$
  $k = k + 1$
end while

9.5.8 Preconditioners

The success of CG and BiCG-stab largely depends on the application of preconditioners. The easy preconditioners are based on Jacobi and Gauss-Seidel, but the most powerful preconditioners are *incomplete factorizations*.

9.5.8.1 Jacob and Gauss-Seidel

Let A be given by $D - L - U$ in which D is diagonal, L is lower triangular and U is upper triangular. Now the Jacob preconditioner is just given by $P = D$. In this particular case it is also possible (provided A is symmetric and positive definite) to do simple CG on a modified system:

$$TAT\mathbf{y} = T\mathbf{b}, \quad \mathbf{x} = T\mathbf{y}, \tag{9.5.41}$$

with $T = \sqrt{D^{-1}} = \text{diag}(d_{kk}^{-\frac{1}{2}})$.

The Jacob-preconditioner is useful when diagonal elements of A differ by several orders of magnitude. If all diagonal elements are roughly the same size this preconditioner has very little effect.

A simple Gauss-Seidel preconditioner is given by $P = D - L$. A factorized variation, based on symmetric GS to use with simple CG is given by

$$(D - L)^{-1}SAS(D - L^T)^{-1}\mathbf{y} = (D - L)^{-1}S\mathbf{b}, \tag{9.5.42}$$

with $S = \sqrt{D}$.

9.5.8.2 Incomplete LU factorization

An LU factorization (see previous chapter) of a large sparse matrix A can be prohibitively expensive because of the fill in, but a very powerful preconditioner can be constructed by making an *approximate* factorization that uses the sparsity structure of the matrix A or allows a very limited fill in only. It works like this. First of all we define a set of matrix coefficients that we are going to use in our incomplete factorization. Usually this set includes all coefficients of A that are non zero. In any case all *diagonal* elements of A must belong to this set. The complement of this set the *neglected set* is denoted by $\mathcal{S}$. Now an incomplete factorization follows the same procedure as a normal decomposition (see Section 9.1.2), with some important exceptions. The matrix A^0 we start the decomposition with *has zeros for indices in the neglected set* and otherwise is equal to A. In actual practice this often will not make any difference, because usually we will choose the neglected set in such a way that $A^0 = A$. Now consider the equations for the update (9.1.8). In incomplete decompositions we leave $a_{jk}^{(i)}$ unchanged if either $(j,k) \in \mathcal{S}$, $(j,i) \in \mathcal{S}$ or $(i,k) \in \mathcal{S}$. In other words, all coefficients in the update equation must belong to the complement of the neglected set, otherwise there is no update.

Exercise 9.5.22 *Show that in incomplete decomposition*

1. $\ell_{ij} = 0$, if $(i,j) \in S$,
2. $u_{i,j} = 0$, if $(i,j) \in S$.

□

The first question is under which circumstances such factorizations are useful. It turns out that for M-matrices A an incomplete factorization will create a regular splitting. This will guarantee the convergence of CG or BiCG-stab using a preconditioner like that (why?).

We first show, that if A is an M matrix, both factors L and U in the LU-factorization will be M-matrices too. We shall show this in steps. Consider the first step in the LU factorization of A:

$$A^{(1)} = L^{(1)}A, \tag{9.5.43}$$

with $L^{(1)}$ lower triangular, with ones on the diagonal and the multiplicators

$$m_{j1} = -a_{j1}/a_{11}, j = 2, \ldots, N$$

in the first column. Observe that $L^{(1)}$ is nonnegative.

Theorem 9.5.1 *If A is an M-matrix, so is $A^{(1)}$.*

Proof.
The elements of $A^{(1)}$ are given by $a_{jk}^{(1)} = a_{jk} + m_{j1}a_{1k}, j = 2, \ldots, N$ and the first row is unchanged. Apparently all off-diagonal elements of $A^{(1)}$ remain non-positive, since m_{j1} is nonnegative and a_{1k} is nonpositive for $k > 1$. So $A^{(1)}$ is still a Z-matrix.

To show that ${A^{(1)}}^{-1}$ is nonnegative we consider the solution of

$$A^{(1)}\mathbf{x}^{(k)} = \mathbf{e}_k, \tag{9.5.44}$$

where $\mathbf{e}_k$ is the k-th unit vector hence $\mathbf{x}^{(k)}$ is just the k-th column of the updated inverse. For $k = 1$ we have that

$$A^{(1)}\mathbf{x}^{(1)} = \mathbf{e}_1 \tag{9.5.45}$$

has solution $\mathbf{x}^{(1)T} = (1/a_{11}, 0, \ldots, 0)$, which is clearly nonnegative. For all other $\mathbf{e}_k$ the solution of

$$A^{(1)}\mathbf{x}^{(k)} = \mathbf{e}_k \tag{9.5.46}$$

is the same as that of

$$A\mathbf{x}^{(k)} = \mathbf{e}_k, \tag{9.5.47}$$

since the right hand side is unaffected by the Gauss step. □

Exercise 9.5.23 *Show from the proof above, that extending an M-matrix on the left with a zero column and at the top with a row with positive diagonal element and nonpositive off-diagonal generates another M-matrix. Infer from this result, that at each state of the Gaussian elimination the intermediate $A^{(k)}$ is an M-matrix if A is an M-matrix.* □

Making a diagonal element *larger* or an off diagonal element *less negative* leaves the M-property untouched, as may be readily shown.

Exercise 9.5.24 *Let $A = (a_{jk})$ be an M-matrix. Let $b_{11} = a_{11} + \alpha$ with $\alpha > 0$, and $b_{jk} = a_{jk}$ otherwise. Show that $B = (b_{jk})$ is also an M-matrix.* □

Exercise 9.5.25 *Let $A = (a_{jk})$ be an M-matrix. Let $b_{12} = a_{12} + \alpha$ with $0 \leq \alpha \leq |a_{12}|$, and $b_{jk} = a_{jk}$ otherwise. Show that $B = (b_{jk})$ is also an M-matrix.* □

This important property gives us the possibility to ignore fill-in in the elimination process, while the intermediate matrix is still an M-matrix.

Exercise 9.5.26 *Show that ${L^{(k)}}^{-1}$ is obtained by multiplying all off diagonal elements of L^k by -1. Show that ${L^{(k)}}^{-1}$is an M-matrix.* □

After $N - 1$ steps of Gaussian elimination we have:

$$U = L^{(N-1)}L^{(N-2)} \ldots L^{(1)}A, \tag{9.5.48}$$

from which we see, that $L^{-1} = L^{(N-1)}L^{(N-2)} \ldots L^{(1)}$ in the decomposition $A = LU$.

Exercise 9.5.27 *Show that if A is an M-matrix both U and L are M-matrices. Show that this remains true if part of the fill in is neglected.* □

Let us formalize this result in a theorem. We denote the index set of neglected fill in by $\mathcal{S}$. $\mathcal{S}$ cannot include diagonal elements.

Theorem 9.5.2 *Let A be an M-matrix. There exists for every index set $\mathcal{S}$ a lower triangular $\tilde{L}$ with unit diagonal. upper triangular $\tilde{U}$ and remainder N such that*

1. $\ell_{kj} = 0, u_{kj} = 0$ *if* $(k,j) \in \mathcal{S}$,
2. $n_{kj} = 0$ *if* $(k,j) \notin \mathcal{S}$,

such that $A = \tilde{L}\tilde{U} - N$. The factors $\tilde{U}$ and $\tilde{L}$ are completely determined by $\mathcal{S}$. $\tilde{L}$ and $\tilde{U}$ are both M-matrices hence $P = \tilde{L}\tilde{U}$ has nonnegative inverse. Since $N \geq 0$ the splitting is regular and generates a convergent iteration process.

A common strategy for choosing $\mathcal{S}$ is to take for the LU decomposition the same sparsity pattern as A has:

$$\mathcal{S} = \{(k,j) |\, a_{k,j} = 0\}. \tag{9.5.49}$$

Exercise 9.5.28 *Show that $\mathcal{S} = \{(k,j) \mid k \neq j\}$ leads to the Jacob preconditioner.* □

We conclude this section with an algorithmic description of the incomplete LU factorization.

ILU algorithm

Require: A is M-matrix
  Presets: $\mathcal{S}$ set of neglected updates
  for $k = 1..N-1$ **do**
    for $j = k+1..N$ **do**
      if $(j,k) \notin S$ **then**
        $\ell_{jk} = a_{jk}/a_{kk}$ {Store the multiplicator in L}
        for $m = k+1..N$ **do**
          if $(j,m) \notin S$ and $(k,m) \notin S$ **then**
            $a_{jm} = a_{jm} - \ell_{jk}a_{km}$
          end if
        end for{m}
      end if
    end for{j}
  end for{k}

Apart from the two tests whether the update should be performed at all, the ILU algorithm is the same as a normal LU decomposition algorithm.

9.6 The multigrid algorithm

The multigrid algorithm (MG) has become one of the most successful iterative techniques in the past 30 years. Its main strength is, that the operation count increases *linearly* with the number of unknowns N or in other words that the number of iterations is independent of the stepsize. Its main weakness is, that it usually needs a lot of fine-tuning before this is realized in practice. We can only give the briefest of introductions into this very interesting subject. For further information we refer the reader to [48], [18] and [40].

9.6.1 A one-dimensional example

Although MG is never applied to one-dimensional problems, the ideas behind it can perfectly well be illustrated with a one-dimensional example. We consider the one-dimensional boundary problem on the interval $(0,1)$:

$$-\frac{d^2u}{dx^2} = f, \quad u(0) = 0, u(1) = 0. \tag{9.6.1}$$

We discretize this on a grid of $N = 2^p + 1$ points to obtain a familiar set of equations:

$$\begin{aligned}
2u_1 - u_2 &= h^2 f_1 && \text{(9.6.2a)}\\
-u1 + 2u_2 + u_3 &= h^2 f_2 && \text{(9.6.2b)}\\
\vdots \qquad\qquad & \vdots \qquad\qquad\\
-u_{k-1} + 2u_k - u_{k+1} &= h^2 f_k && \text{(9.6.2c)}\\
\vdots \qquad\qquad & \vdots \qquad\qquad\\
-u_{N-3} + 2u_{N-2} - u_{N-1} &= h^2 f_{N-2} && \text{(9.6.2d)}\\
-u_{N-2} + 2u_{N-1} &= h^2 f_{N-1} && \text{(9.6.2e)}
\end{aligned}$$

or $A\mathbf{u} = h^2\mathbf{f}$ in which A is an $(N-1)\times(N-1)$ tridiagonal matrix with 2's on the diagonal and -1's on subdiagonal and super diagonal. The eigenvalues and eigen vectors of such a matrix can be calculated exactly.

Theorem 9.6.1 *Let the $(N-1)\times(N-1)$ matrix A be defined as above, The eigenvalues λ_k are given by*

$$\lambda_k = 4\sin^2\frac{k\pi}{2N}, \quad k = 1, 2, \ldots, N-1, \tag{9.6.3}$$

and the components v_{kj} of the corresponding eigenvectors $\mathbf{v}_k$ by

$$v_{kj} = \sqrt{\frac{2}{N}}\sin\frac{kj\pi}{N}. \tag{9.6.4}$$

Proof

Consider the three term recurrence relation

$$-u_{j-1} + (2-\lambda)u_j - u_{j+1} = 0, \quad u_0 = 0, u_N = 0. \tag{9.6.5}$$

From the theory of linear recurrence relations we know, that the general solution is of the form $u_k = a\rho_1^k + b\rho_2^k$, where $\rho_{1,2}$ are the solutions of the quadratic:

$$-\rho^2 + (2-\lambda)\rho - 1 = 0. \tag{9.6.6}$$

Because A is symmetric, the eigenvalues are real and from Gershgorin's theorem they should lie in the interval $(0,4)$. Therefore the discriminant of Equation (9.6.6) is negative and the roots are conjugate complex. Since $\rho_1\rho_2 = 1$ (why?) we set $\rho_1 = e^{i\phi}$ and $\rho_2 = e^{-i\phi}$. This substituted in Equation (9.6.6) gives

$$\begin{aligned}
2 - \lambda &= e^{i\phi} + e^{-i\phi},\\
\lambda &= 2 - 2\cos\phi,\\
&= 4\sin^2\frac{1}{2}\phi.
\end{aligned} \tag{9.6.7}$$

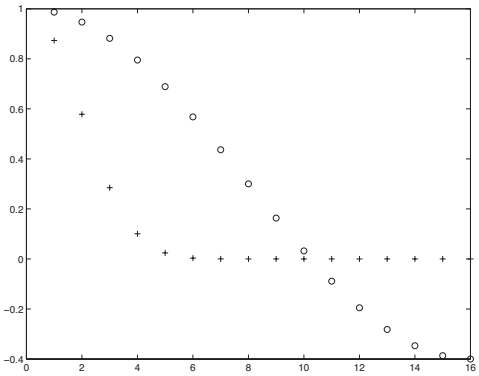


Figure 9.8: Eigenvalues 'o' and 10-th powers of eigenvalues '+'.

The general solution therefore is $u_j = ae^{ij\phi} + be^{-ij\phi}$ and since $u_0 = 0$ we get $a = -b$ or $u_j = c \sin j\phi$. Because $u_N = 0$, $\sin N\phi = 0$ or $\phi = \dfrac{k\pi}{N}$.

$c = \sqrt{2/N}$ follows from a normalization argument. See Exercise 9.6.1. □

Exercise 9.6.1 *Show that*

$$\sum_{j=1}^{N} \sin^2 \frac{jk\pi}{N} = \frac{1}{2}N. \tag{9.6.8}$$

(Hint: write $\sin\phi = (e^{i\phi} - e^{-i\phi})/(2i)$*.)* □

Exercise 9.6.2 *Show that*

$$\sum_{j=1}^{N-1} \sin \frac{jk\pi}{N} \sin \frac{j\ell\pi}{N} = 0, \quad \text{if } k \neq \ell. \tag{9.6.9}$$

□

Exercise 9.6.3 *Show that for large N the* smallest *eigenvalue of A*

$$\lambda_1 \approx \frac{\pi^2}{N^2} = \pi^2 h^2 \tag{9.6.10}$$

□

9.6.2 Smooth and rough part of the spectrum

For this example let us look at a classic iteration process called *damped Jacob.* The preconditioner is given by $P = \alpha^{-1}D$, $\alpha < 1$ and the iteration process by

$$\mathbf{c}^k = D^{-1}\mathbf{r}_k \tag{9.6.11a}$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha\mathbf{c}^k \tag{9.6.11b}$$

$$\mathbf{r}^{k+1} = \mathbf{r}^k - \alpha A\mathbf{c}^k \tag{9.6.11c}$$

The eigenvalues μ_k of the iterationmatrix $M = I - P^{-1}A$ are given by $\mu_k = 1 - \frac{1}{2}\alpha\lambda_k$ in which λ_k the eigenvalues of A. They have been pictured in Figure 9.8.

In the same figure the eigenvalues to the power 10 have been plotted. This is what remains of the components of the error after 10 iterations. As you can

see, all components have disappeared except for those belonging to the smallest eigenvalues λ_k of the matrix A corresponding to those closest to 1 in the iteration matrix M. These eigenvalues are called the *smooth part* of the spectrum and the eigenvalues that are damped out the *rough part* for reasons that become clear when you look at Figure 9.9.

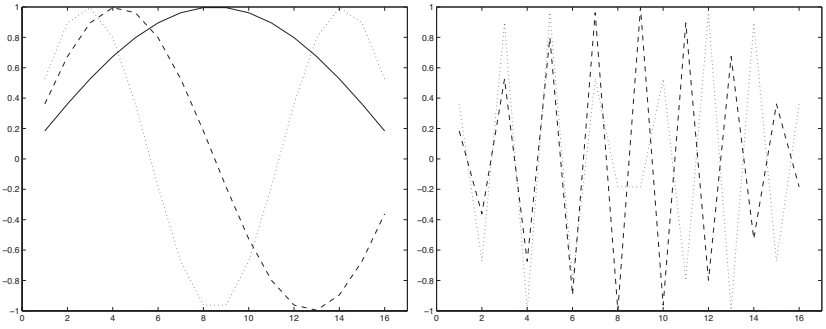


Figure 9.9: Smooth ($\mathbf{v}_{1,2,3}$) and rough ($\mathbf{v}_{15,16}$) eigenvectors.

The contribution to the error belonging to the rough part of the spectrum is annihilated very soon by the preconditioner that for that reason is called a *smoother*. The contribution belonging to the smooth part, however, is not annihilated at all and is the reason the classic iteration process converges so slowly.

9.6.3 Two grid algorithm

The central idea of the MG algorithm is to obtain the smooth parts of the solution in a different way, notably from a solution to a related problem *on a coarser grid*. Suppose we somehow had the solution in the even numbered points $(x_0, x_2, .., x_{16})$, $(u_0, u_2, ...u_{16})$. We could obtained an initial estimate by linear interpolation: $u^0_{2k+1} = (u_{2k} + u_{2k+2})/2$. Since the error in this estimate is 0 in the even points and has the same sign in the odd points as the second derivative it looks something like Figure 9.10.

Apparently this error has a large component in the rough part of the spectrum, exactly where our smoother is most effective. This is the central idea of the *two grid algorithm*. We define a *coarse grid* $\mathcal{G}_H : x_0, x_2, x_4, x_{2k} \ldots x_N$ and a fine grid $\mathcal{G}_h : x_0, x_1, \ldots, x_N$. Our original problem $A_h\mathbf{u}_h = \mathbf{f}_h$ lives on the fine grid. On the coarse grid we can calculate the solution to a related problem $A_H\mathbf{u}_H = \mathbf{f}_H$.

Apparently there must be a mapping from fine grid to coarse grid $R_{Hh}\mathbf{f}_h = \mathbf{f}_H$. This mapping is called the *restriction* in MG speak. And the interpolation operator to get from $\mathbf{u}_H$ to $\mathbf{u}_h$ is called the *prolongation* $P_{hH}\mathbf{u}_H = \mathbf{u}_h$. The prolongation operator is easy in this case. It is the matrix

$$\begin{pmatrix} \frac{1}{2} & 0 & \ldots & \ldots & \ldots & \ldots \\ 1 & 0 & \ldots & \ldots & \ldots & \ldots \\ \frac{1}{2} & \frac{1}{2} & \ldots & \ldots & \ldots & \ldots \\ 0 & 1 & 0 & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & 0 & 1 & 0 \\ \ldots & \ldots & \ldots & \ldots & \frac{1}{2} & \frac{1}{2} \\ \ldots & \ldots & \ldots & \ldots & 0 & 1 \\ \ldots & \ldots & \ldots & \ldots & 0 & \frac{1}{2} \end{pmatrix} \tag{9.6.12}$$

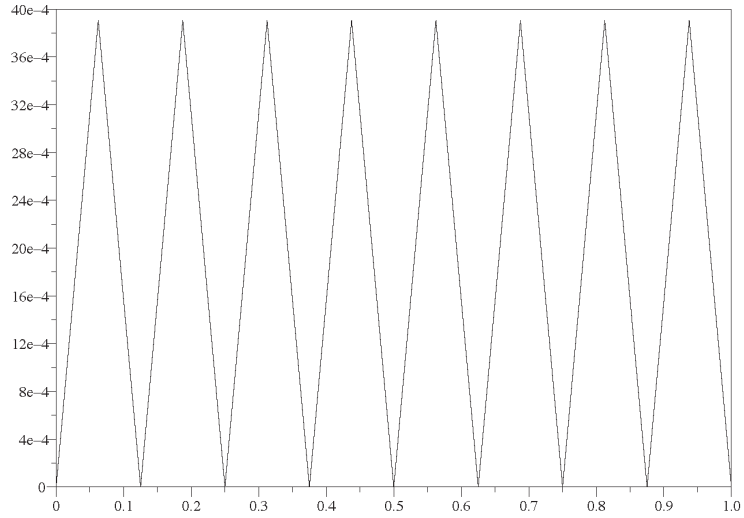


Figure 9.10: Typical error by interpolation.

Since we basically try to solve a set of equations in a space of reduced dimension an obvious strategy is to apply the Galerkin method, in other words take $R = P^T$. Other choices are possible too, and come down to Petrov Galerkin.

Exercise 9.6.4 *Show, that $A_H = P_{hH}^T A_h P_{hH}$ is a tridiagonal matrix with 1's on the diagonal and $-\frac{1}{2}$'s on the sub and super diagonal. Show that $f_{H,k} = \frac{1}{2} f_{h,2k-1} + f_{h,2k} + \frac{1}{2} f_{h,2k+1}$.* □

We present the two grid algorithm in algorithmic form:
Two grid algorithm

1: Presets: $\mathbf{u}_h^0$, $\mathbf{r}_h^0 = \mathbf{f}_h - A\mathbf{u}_h^0$
2: $\mathbf{u}_h^{\text{prs}} = S(\mathbf{u}_h^0, \mathbf{b}, A, n_0)$ {Presmoothing}
3: $\mathbf{r}_H = R_{Hh}\mathbf{r}_h$
4: Solve $A_H \mathbf{c}_H = \mathbf{r}_H$
5: $\mathbf{u}_h^{\text{cgc}} = \mathbf{u}_h^p + P_{hH}\mathbf{c}_H$ {Coarse Grid Correction}
6: $\mathbf{u}_h^{\text{pos}} = S(\mathbf{u}_h^{\text{cgc}}, \mathbf{b}, A, n_1)$ {Postsmoothing}

So the two grid algorithm consist of a pre-smoothing stage, a coarse grid correction and a post-smoothing stage. The notation $S(\mathbf{u}_h^0, \mathbf{b}, A, n)$ means: do n steps with the preferred smoother, with initial estimate $\mathbf{u}_h^0$, right hand side $\mathbf{b}$ and matrix A.

To get an impression how effective the two grid algorithm is, we look at the contributions of the rough and smooth spectra to the original error and the error after coarse grid correction. Only odd modes are shown, because the even modes happen to vanish in this specific example ($f(x) = 1$). See Table 9.11. As you can see,

	smooth				rough			
mode	1	3	5	7	9	11	13	15
initial $\times 10^{-4}$	2580	96	21	7	3	2	1	0
after corr $\times 10^{-4}$	25	9	6	5	5	6	9	25

Figure 9.11: Error reduction by coarse grid correction.

the coarse grid correction action is complementary to that of the smoother. The

contribution of the smooth part of the spectrum is reduced very rapidly, whereas the rough part is increased. Presmoothing is not very efficient in this particular case, since the contribution of the rough component to the initial error is almost negligible.

Exercise 9.6.5 *Explain why we use* damped *Jacob as smoother and not original vintage Jacob. (Hint: consider Figure 9.7)* □

9.6.4 From two grid to multigrid

The *Multigrid* algorithm consists of applying the two grid algorithm recursively to step 4. To solve the coarse grid problem $A_H\mathbf{c}_H = \mathbf{r}_H$, we define an even coarser grid
$x_0, x_4, x_8, x_{12}, x_{16}$, and because we run out of fonts in which to print H to express the fact that this is an even coarser grid it is maybe a good idea to let the notation reflect the *coarsening level*. Let our finest grid be coarsening level 0 and let each application coarsening increase the level by 1. So in level 0 we have $2^p + 1$ points, in level 1 $2^{p-1} + 1$ points, in level k $2^{p-k} + 1$ points. We denote the matrix and vectors on level ℓ with A_ℓ and $\mathbf{v}_\ell$ respectively. R_ℓ and P_ℓ operate from this level to the next higher and lower levels respectively. If we have three points left in our grid we must stop, because we only have one unknown left. The other two are boundaries. Hence we should stop at level $p-1$ or earlier. At level $p-1$ we can solve the problem $A_{p-1}\mathbf{c}_{p-1} = \mathbf{r}_{p-1}$ directly. This gives us the following algorithm:
MGRecursive $(A_\ell, \mathbf{r}_\ell, \mathbf{c}_\ell, \ell)$
  if $\ell < p-1$ **then**
    $\mathbf{c}_\ell = S(\mathbf{0}, \mathbf{r}_\ell, A_\ell, n_0)$ {Presmoothing}
    $\mathbf{r}_{\ell+1} = R_\ell(\mathbf{r}_\ell - A_\ell\mathbf{c}_\ell)$ {Calculate coarse grid residual}
    $A_{\ell+1} = R_\ell A_\ell P_{\ell+1}$ {Calculate coarse grid matrix}
    call MGRecursive $(A_{\ell+1}, \mathbf{r}_{\ell+1}, \mathbf{c}_{\ell+1}, \ell+1)$
    $\mathbf{c}_\ell = \mathbf{c}_\ell + P_{\ell+1}\mathbf{c}_{\ell+1}$ {Coarse grid correction}
    $\mathbf{c}_\ell = S(\mathbf{c}_\ell, \mathbf{r}_\ell, A_\ell, n_1)$ {Postsmoothing}
  else
    Solve $A_{p-1}\mathbf{c}_{p-1} = \mathbf{r}_{p-1}$ {Direct solution on coarsest level}
  end if

For clearness of presentation the calculation of the coarse grid matrix has been put into the algorithm. This is definitely not a good idea in practice, because the algorithm will be used several times and these coarse grid operators do not change. It is a better idea, to do a preliminary stage in which the restriction, prolongation and matrix on all levels are calculated and stored.

The analysis of the multigrid algorithm is far more involved than that of the two grid algorithm, but the analysis remains qualitatively valid.

Exercise 9.6.6 *Explain, that if $n_0 = 0$ (no presmoothing), the right hand side on the coarsest level is given by*

$$\mathbf{r}_{p-1} = R_{p-2}R_{p-3}\dots R_1R_0\mathbf{r}_0. \tag{9.6.13}$$

Exercise 9.6.7 *Calculate the right hand side on the coarsest grid for $p = 3$ and $f(x) = 1$. (No presmoothing). Calculate the matrix on the coarsest grid for the model problem.*

9.6.5 Convergence of the two grid algorithm

For ease of presentation we consider the two grid algorithm with post smoothing only. It consists of two steps, the coarse grid correction and the postsmoothing. We

will first study the effect coarse grid correction. Let us call the coarse grid space Σ_H and the fine grid space Σ_h. The example matrix A is positive definite so the problem to solve in Σ_h is:

$$\min_{\mathbf{c}_h \in \Sigma_h} \frac{1}{2}(\mathbf{c}_h, A\mathbf{c}_h) - (\mathbf{c}_h, \mathbf{r}) \tag{9.6.14}$$

but instead we solved:

$$\min_{\mathbf{c}_H \in \Sigma_H} \frac{1}{2}(P_{hH}\mathbf{c}_H, AP_{hH}\mathbf{c}_H) - (P_{hH}\mathbf{c}_H, \mathbf{r}) \tag{9.6.15}$$

and after that put $\tilde{\mathbf{c}}_h = P_{hH}\mathbf{c}_H$. So we approximated the solution on a subspace but on that subspace we obtained the best possible solution in the sense that it minimizes the quadratic form (9.6.15). In particular, if $\hat{\mathbf{c}}_H$ is the exact solution in the coarse grid points you would still have, from Equation (9.6.15) that

$$\frac{1}{2}(\tilde{\mathbf{c}}, A\tilde{\mathbf{c}}) - (\tilde{\mathbf{c}}, \mathbf{r}) \leq \frac{1}{2}(P_{hH}\hat{\mathbf{c}}_H, AP_{hH}\hat{\mathbf{c}}_H) - (P_{hH}\hat{\mathbf{c}}_H, \mathbf{r}). \tag{9.6.16}$$

And since $\mathbf{r} = A\hat{\mathbf{c}}$ this transforms, after adding $\frac{1}{2}(\hat{\mathbf{c}}, A\hat{\mathbf{c}})$ to both sides into

$$\frac{1}{2}(\tilde{\mathbf{c}} - \hat{\mathbf{c}}, A(\tilde{\mathbf{c}} - \hat{\mathbf{c}})) \leq \frac{1}{2}(P_{hH}\hat{\mathbf{c}} - \hat{\mathbf{c}}, AP_{hH}(\hat{\mathbf{c}} - \hat{\mathbf{c}})). \tag{9.6.17}$$

Hence

$$\|\tilde{\mathbf{c}} - \hat{\mathbf{c}}\|_A \leq \|P_{hH}\hat{\mathbf{c}} - \hat{\mathbf{c}}\|_A. \tag{9.6.18}$$

This very interesting result (analogous to the Finite Element error estimate) tells us, that the error after the coarse grid correction is no worse than the error in linear interpolation of the exact solution *measured in the* $\|.\|_A$ *norm.*

Let us have a look at the residual after coarse grid correction:

$$\tilde{\mathbf{r}} = \mathbf{r} - A_h P_{hH} A_H^{-1} R_{Hh}\mathbf{r}, \tag{9.6.19}$$

$$= (I - A_h P_{hH} A_H^{-1} R_{Hh})\mathbf{r} = Q\mathbf{r}. \tag{9.6.20}$$

Exercise 9.6.8 *Show that* $Range(Q) = Ke(R_{Hh})$. *Infer from this that repeated coarse grid corrections are useless without intermediate smoothing.* □

Finally we shall look in detail to the reduction of the various modes in the spectrum. We shall show that the smooth part of the spectrum is reduced by the coarse grid correction by a fixed amount, independent of the stepsize h. We already know that the rough part of the spectrum is reduced by a fixed amount, irrespective of the stepsize. Together we conclude that one twogrid iteration reduces the error by a fixed amount, irrespective of the stepsize. This means that we need only a fixed number of iterations to get to a certain accuracy *and that the total number of operations is only dependent on the number of operations per iteration*.

The analysis we are about to perform is made easy by the fact, that the smooth part of the eigenvectors of the coarse grid corrector Q and the damped Jacob smoother S are the same: $v_{kj} = \sin jk\pi/N$. Usually this is not the case and this makes convergence analysis a lot harder. For a more general treatment of convergence properties of the multigrid algorithm we refer the reader to [48], [18] and [40].

We perform the analysis in several steps in a number of exercises.

Exercise 9.6.9 *Let A_h be an $N_h - 1 \times N_h - 1$ matrix, $N_h = 2^{p+1}$, with diagonal $2N_h$ and sub and superdiagonals $-N_h$. Show that A_H is an $N_H - 1 \times N_H - 1$ matrix with $N_H = 2^p$ with diagonal $2N_H$ and sub and super diagonals $-N_H$.*

Calculate the eigenvalues of A_H and A_h using Theorem 9.6.1. Calculate the eigenvectors too. □

Exercise 9.6.10 *Show that the smooth part of the eigenvectors of A_h (i.e. $k < N_H$) is completely represented in the eigenvectors of A_H, if you realize that $\mathbf{v}_{Hk}$ contain only the even vectorcomponents of $\mathbf{v}_{hk}$. For instance $\mathbf{v}_{h1}$ has components $\sin j\pi/N_h$ and the even components of that vector just make up $\mathbf{v}_{H1}$ with components $\sin j\pi/N_H = \sin 2j\pi/N_h$.* □

The rough part of the eigenvectors of A_h (i.e. $k \geq N_H$) is not easily represented by eigenvectors with $k_H = k_h \mod N_H$. For instance $\mathbf{v}_{hN_H+1}$ has components $\sin((N_H+1)j\pi/N_h)$ and the even components of that vector just make up

$$\begin{aligned}\sin((N_H+1)2j\pi/N_h) &= \sin((N_H+1)j\pi/N_H),\\ &= \sin(j\pi + j\pi/N_H),\\ &= (-1)^j \sin j\pi/N_H.\end{aligned} \tag{9.6.21}$$

Exercise 9.6.11 *Show that for eigenvectors $\mathbf{v}_{hk}$ of A_h belonging to the smooth spectrum*

$$R_{Hh}\mathbf{v}_{hk} = (1 + cosk\pi/N_h)\mathbf{v}_{Hk}. \tag{9.6.22}$$

□

Exercise 9.6.12 *Show, that for eigenvectors $\mathbf{v}_{Hk}$ of A_H:*

$$P_{hH}\mathbf{v}_{Hk} = \mathbf{v}_{hk} + \text{rough part}, \tag{9.6.23}$$

in which the rough part has even components 0 and odd components

$$v_{k,2j-1} = (1 - \cos k\pi/N_h)\sin k(2j-1)\pi/N_h \tag{9.6.24}$$

□

From Exercises 9.6.9, 9.6.10 9.6.11 and 9.6.12, we conclude, that $Q\mathbf{v}_{hk}$ with k belonging to the smooth part of the spectrum has a smooth spectrum component of:

$$Q\mathbf{v}_{hk} = \left(1 - (1 + \cos k\pi/N)\frac{\lambda_{hk}}{\lambda_{Hk}}\right)\mathbf{v}_{hk} \tag{9.6.25}$$

Now $\lambda_{hk} = 4N_h sin^2 k\pi/(2N_h)$ and $\lambda_{Hk} = 4N_H sin^2 k\pi/(2N_H)$ and after short manipulation:

$$\begin{aligned}2(1 + cos\phi)\frac{\sin^2\phi/2}{\sin^2\phi} &= 2(2 - 2sin^2\phi/2)\frac{\sin^2\phi/2}{\sin^2\phi}\\ &= \frac{4\cos^2\phi/2\sin^2\phi/2}{sin^2\phi}\\ &= 1\end{aligned} \tag{9.6.26}$$

in other words, the smooth part of the spectrum is completely annihilated by the coarse grid correction. That is not entirely true, by the way, because there is some crossover between rough and smooth components.

9.6.6 Restriction and prolongation in two dimensions

The one-dimensional example is in a way special, because many aspects of the algorithm can be calculated exactly. Nevertheless, the main components of the algorithm remain true in more dimensions. The only thing that needs special attention are the restriction and prolongation operators. A straightforward generalization to 2 and 3 dimensions would be bi- or trilinear interpolation and that would fit the bill just fine, but for a small problem.

Let us assume we are working on a rectangular region with 2^p cells in x-direction and 2^q cells in y-direction. We define the bilinear restrictions and prolongations as follows: $P = P_x P_y$ in which P_x is the one-dimensional prolongation operator in x-direction applied to each single row of the region and P_y is is the one-dimensional prolongation operator in y-direction applied to each single column.

Exercise 9.6.13 *Express P_x and P_y in matrix form. Show that P_x and P_y commute.* □

We take again $R = P^T$. However, if you calculate the coarse grid operator A_H from a fine grid operator A_h coming from the five point Laplace molecule, you will see, that the foot print increases from a five point to a nine point molecule. Fortunately, on going to even coarser grids the foot print does not increase.

Exercise 9.6.14 *Show this. Show also that in 3 dimensions the foot print increases from a 7-point molecule to a 27-point molecule.* □

9.6.7 Concluding remarks about MG

There is a vast amount of literature on the subject of MG algorithms. [48] and [18] are classics that are recently reprinted, [40] is recent. All contain pointers to publications that may be of further interest.

A couple of remarks is in order:

1. Damped Jacob is a good smoother, but not the only one. Gauss Seidel is also good and the incomplete *LU* factorizations are very good.
2. The use of powers of 2 as number of cells is widely spread, but not really necessary. The algorithm has been successfully applied for any number of grid points.
3. There are other variations that we have not treated in this short exposition. Most notably other interpolation strategies (cell centered versus vertex centered) and recursion strategies (F-, V- and W cycles) have not been covered. We only have shown a simple vertex centered V-cycle.
4. One multigrid cycle is really a *preconditioner* that can be used with defect-correction or Bi-CGStab. (It is hard to use with CG because of the symmetry requirement). The latter choice is by far the best.
5. There are still unsolved problems with MG in unstructured grids or rapidly changing coefficients. Also applications to 3D problems have still some unchartered waters.

9.7 Non-linear equations

The discretization of non-linear PDEs leads to non-linear algebraic equations. Although many methods to solve non-linear algebraic system are available in the

mathematical literature, we will only treat two classical iterative processes: *Picard iteration* and *Newton iteration*. These two methods usually respectively exhibit linear and quadratic convergence.

9.7.1 Picard iteration

First we consider a class of problems that are small perturbations of linear problems. For instance

$$-\text{div grad}\, u + f(u) = 0, \quad \text{on } \Omega, \tag{9.7.1}$$

and $u = 0$ on Γ. If you discretize this the standard way, you end up with a set of equations of the form

$$A\mathbf{u} + \mathbf{f}(\mathbf{u}) = 0, \tag{9.7.2}$$

in which $f_k(\mathbf{u}) = f(u_k)$. To approximate the solution of the above equation, we generate an array $\mathbf{u}^n$ with the goal that $\mathbf{u}^n \to \mathbf{u}$ as $n \to \infty$. The estimates $\mathbf{u}^n$ are obtained by solving a *linear* system of equations. Since we are only able to solve linear problems as $A\mathbf{u} = \mathbf{b}$, a natural way to go about this is to start out with an initial estimate $\mathbf{u}^0$ and solve the following iteratively:

$$A\mathbf{u}^{n+1} = -\mathbf{f}(\mathbf{u}^n). \tag{9.7.3}$$

Such an iterative process is known as *Picard iteration*.

Exercise 9.7.1 *Show that if* $\mathbf{u}$ *is the solution of* (9.7.2) *and* $\varepsilon^n = \mathbf{u} - \mathbf{u}^n$, *with* $\mathbf{u}^n$ *solution of* (9.7.3) *that*

$$A\varepsilon^{n+1} = D(\mathbf{u})\varepsilon^n + O(\|\varepsilon^n\|^2), \tag{9.7.4}$$

in which D *is a diagonal matrix with* $d_{kk}(\mathbf{u}) = -f'(u_k)$. *Show that this process cannot converge if at least one eigenvalue of* $A^{-1}D$ *is larger than 1 in absolute value.* □

An other example concerns the case of an elliptic equation in which the coefficients depend on the solution u. Let us consider the following equation

$$-\text{div}\ (D(u)\text{grad}\, u) = f(\mathbf{x}). \tag{9.7.5}$$

If $D(u)$ is not a constant, for instance $D(u) = u$, then the above equation is nonlinear. To solve the above equation, we generate a sequence of approximations u^n as in the previous example. Here the above equation is solved by iterating

$$-\text{div}\ \left(D(u^n)\text{grad}\, u^{n+1}\right) = f(\mathbf{x}). \tag{9.7.6}$$

After construction of an appropriate discretization, a linear system to obtain u^{n+1} has to be solved. In general if one wants to solve a nonlinear problem using Picard's method, convergence is not always guaranteed. One needs to use commonsense to solve the problem.

So a natural way to obtain an iterative process to a non-linear set of equations $\mathbf{f}(\mathbf{x}) = 0$ is to reform it to a *fixed point form* $\mathbf{x} = G(\mathbf{x})$ with the same solution. On this fixed point form you graft an iterative process:

$$\mathbf{x}^{k+1} = G(\mathbf{x}^k) \tag{9.7.7}$$

There is a famous convergence result due to Banach on such processes.

Theorem 9.7.1 *Let* $\mathcal{D}$ *be a closed subset of* $\mathbb{R}^n$ *and let* G *be a mapping* $\mathbb{R}^n \to \mathbb{R}^n$ *such that*

1. *if* $\mathbf{x} \in \mathcal{D}$ *then* $G(\mathbf{x}) \in \mathcal{D}$
2. $\|G(\mathbf{x}) - G(\mathbf{y})\| \leq \alpha\|\mathbf{x} - \mathbf{y}\|$ *for all* $\mathbf{x}, \mathbf{y}, \in \mathcal{D}$, $\alpha < 1$

then $\mathcal{D}$ *contains precisely one fixed point of* G.

Proof
Choose $\mathbf{x}^0 \in \mathcal{D}$. By elementary induction it will be clear, that the whole sequence generated by $\mathbf{x}^{k+1} = G\mathbf{x}^k$ is in $\mathcal{D}$. Apparently $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| = \|G(\mathbf{x}^k) - G(\mathbf{x}^{k-1})\| \leq \alpha\|\mathbf{x}^k - \mathbf{x}^{k-1}\|... \leq \alpha^k\|\mathbf{x}^1 - \mathbf{x}^0\|$. Hence the sequence converges to a limit which lies in $\mathcal{D}$ because $\mathcal{D}$ is closed.

There cannot be two different fixed points ξ and η. If there were, $\|G(\xi) - G(\eta)\| \leq \alpha\|\xi - \eta\|$, which is clearly impossible, since $G(\xi) = \xi$ and $G(\eta) = \eta$. □

A mapping that satisfies the conditions of Theorem 9.7.1 is called a *contraction* or a *contractive mapping*.

Exercise 9.7.2 *Let* $\mathbf{x}^{k+1} = G(\mathbf{x}^k)$ *be an iterative process with limit* ξ. *G has continuous partial derivatives in a neighborhood D of* ξ *and* $\|G'(\mathbf{x}))\| < 1, \mathbf{x} \in D$. G' *is the matrix with*

$$g'_{kj} = \frac{\partial g_k}{\partial x_j}. \tag{9.7.8}$$

Show that D contains a subset on which G is a contraction. □

9.7.2 Newton's method in more dimensions

In order to find a faster converging solution process to the set of non linear equations

$$\mathbf{f}(\mathbf{x}) = 0, \quad \mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n, \mathbf{x} \in \mathbb{R}^n \tag{9.7.9}$$

we try to find an analogue to Newton's method for functions of one variable:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)}. \tag{9.7.10}$$

In the neighborhood of the root ξ we have by Taylors theorem:

$$0 = f(\xi) = f(x) + (\xi - x)f'(x) + O((\xi - x)^2), \tag{9.7.11}$$

for functions of one variable. We arrive at Newton's formula by neglecting the second order term. We try something similar in n dimensions. In the neighborhood of the root ξ we have:

$$0 = f_1(\xi) = f_1(\mathbf{x}) + \sum_{j=1}^{n}(\xi_j - x_j)\frac{\partial f_1}{\partial x_j}(\mathbf{x}) + O(\|\xi - \mathbf{x}\|^2), \tag{9.7.12a}$$

$$0 = f_2(\xi) = f_2(\mathbf{x}) + \sum_{j=1}^{n}(\xi_j - x_j)\frac{\partial f_2}{\partial x_j}(\mathbf{x}) + O(\|\xi - \mathbf{x}\|^2), \tag{9.7.12b}$$

$$\vdots$$

$$0 = f_n(\xi) = f_n(\mathbf{x}) + \sum_{j=1}^{n}(\xi_j - x_j)\frac{\partial f_n}{\partial x_j}(\mathbf{x}) + O(\|\xi - \mathbf{x}\|^2). \tag{9.7.12c}$$

Neglecting the second order term Equations (9.7.12) we arrive at an iteration process that is analogous to (9.7.10):

$$f_1(\mathbf{x}^k) + \sum_{j=1}^{n} (x_j^{k+1} - x_j^k) \frac{\partial f_1}{\partial x_j}(\mathbf{x}^k) = 0, \tag{9.7.13a}$$

$$f_2(\mathbf{x}^k) + \sum_{j=1}^{n} (x_j^{k+1} - x_j^k) \frac{\partial f_2}{\partial x_j}(\mathbf{x}^k) = 0, \tag{9.7.13b}$$

$$\vdots$$

$$f_n(\mathbf{x}^k) + \sum_{j=1}^{n} (x_j^{k+1} - x_j^k) \frac{\partial f_n}{\partial x_j}(\mathbf{x}^k) = 0. \tag{9.7.13c}$$

We can put this into vector notation:

$$f'(\mathbf{x}^k)(\mathbf{x}^{k+1} - \mathbf{x}^k) = -\mathbf{f}(\mathbf{x}^k), \tag{9.7.14}$$

where $f'(\mathbf{x})$ is the Jacobian matrix

$$f'(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} (\mathbf{x}). \tag{9.7.15}$$

We now present the algorithmic form.

Newton's method for multivariate functions

1: Presets: $\mathbf{x}^0$ {initial estimate}, $\mathbf{r}^0 = \mathbf{f}(\mathbf{x}^0)$, $k = 0$
2: **while** $\|\mathbf{r}^k\| > \varepsilon$ **do**
3: Solve $f'(\mathbf{x}^k)\mathbf{c}^k = -\mathbf{r}^k$
4: $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{c}^k$
5: $\mathbf{r}^{k+1} = \mathbf{f}(\mathbf{x}^{k+1})$
6: $k = k + 1$
7: **end while**

The calculation of the Jacobian is often very time consuming and various schemes have been proposed to improve on that. For the solution of the linear system on line 3 we can use any type of solver. The structure of the Jacobian often has the same sparsity pattern as the corresponding linearization of the PDE.

Example 9.7.1 *We consider the following differential equation in one spatial dimension, with boundary conditions:*

$$u(1-u)\frac{d^2u}{dx^2} + x = 0, \qquad u(0) = u(1) = 0. \tag{9.7.16}$$

A finite difference discretization, with equidistant grid-spacing h and n unknowns ($h = 1/(n+1)$), gives

$$f_i(\mathbf{u}) = u_i(1-u_i)\frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} + x_i = 0, \text{ for } i \in \{1, \ldots, n\}. \tag{9.7.17}$$

Note that for $i = 1$ and $i = n$, the boundary conditions are used. This system of n equations with n unknowns is seen as a system of non-linear equations. Using the Picard fixed point or Newton method requires an initial guess for the solution. This initial guess could be chosen by solving the linearized system or by choosing a vector that reflects the

values at a Dirichlet boundary (if there is any). Let $\mathbf{u}^k$ *represent the solution at the k-th iterate, then, one way of using the Picard fixed point method is the following:*

$$u_i^k(1-u_i^k)\frac{u_{i-1}^{k+1}-2u_i^{k+1}+u_{i+1}^{k+1}}{h^2}+x_i=0, \text{ for } i\in\{1,\dots,n\}. \tag{9.7.18}$$

This requires the solution of a system of linear equation at each iterate.

If one prefers to use the Newton method, then, the calculation of the Jacobian matrix is necessary. Considering the i-th row of the Jacobian matrix, all entries are zero, except the one on and the ones adjacent to the main diagonal, that is

$$\begin{aligned}
\frac{\partial f_i}{\partial u_{i-1}}(\mathbf{u}^k) &= \frac{u_i^k(1-u_i^k)}{h^2},\\
\frac{\partial f_i}{\partial u_i}(\mathbf{u}^k) &= \frac{2u_i^k(1-u_i^k)}{h^2}+(1-2u_i^k)\frac{u_{i-1}^k-2u_i^k+u_{i+1}^k}{h^2},\\
\frac{\partial f_i}{\partial u_{i+1}}(\mathbf{u}^k) &= \frac{u_i^k(1-u_i^k)}{h^2}.
\end{aligned} \tag{9.7.19}$$

The rest of the procedure is straightforward. □

Exercise 9.7.3 *Consider the discretization of*

$$-\operatorname{div}\operatorname{grad} u+\mathrm{e}^u=0, \tag{9.7.20}$$

on the square $(0,1)\times(0,1)$. *Calculate* $f'(\mathbf{u})$. *Compare the structure of the Jacobian to the matrix generated by the discretization of the Laplacian.* □

Exercise 9.7.4 *Consider the discretization of*

$$\operatorname{div}\left(\frac{\operatorname{grad} u}{\sqrt{1+u_x^2+u_y^2}}\right)=0, \tag{9.7.21}$$

on the square $(0,1)\times(0,1)$ *by finite volume method. What is the sparsity structure of* $f'(\mathbf{x})$*?* □

9.7.3 Starting values

Although Newton's method converges quadratically in a neighborhood of the root, convergence is often very sensitive to good initial estimates. These are suggested sometimes by the technical context, but if obtaining an initial estimate appears to be a problem the following trick, known as *homotopy method* may be applied.

Suppose the solution to some other problem, say $\mathbf{g}(\mathbf{x})=0$ is known (e.g. a linearization of the original). Consider the following set of problems:

$$(1-\lambda)\mathbf{g}(\mathbf{x})+\lambda\mathbf{f}(\mathbf{x})=0,\quad \lambda\in(0,1). \tag{9.7.22}$$

For $\lambda=1$ we have our original problem, for $\lambda=0$ we have our auxiliary problem. Now the idea is to proceed in small steps h from $\lambda_0=0,\lambda_1=h,\lambda_2=2h$ to $\lambda_N=Nh=1$, using Newton's method as solver and always taking the solution to the problem with λ_k as initial estimate to the problem with λ_{k+1}. This is an expensive method but somewhat more robust than simple Newton.

9.8 Summary of Chapter 9

In this chapter we have studied methods to solve linear and non-linear sets of equations. Direct methods are important particularly for not too large two dimensional problems. In general an LU-decomposition is used. For a structured grid a band method is optimal, but for unstructured girds, profile methods, generally require less memory and computing time.

Renumbering techniques, like Cuthill-McKee reduce the size of the matrix to an almost optimal one.

Iterative methods become important for large problems, where direct methods may be too expensive or do not fit into memory. We first looked at *defect correction* or *standard iteration methods* like Jacob, Gauss Seidel and Successive Overrelaxation. After that we met *Krylov space methods* like Conjugate Gradients (CG) and BiCG-stab. We found that the standard methods could be used as *preconditioner*. We also met a more powerful preconditioner *incomplete LU factorization*.

We learned about the Multigrid algorithm, how its convergence is independent of the stepsize of the approximation, by using a *coarse grid correction* to get rid of the smooth part of the spectrum.

We briefly looked at non linear problems and met simple Picard iteration and a generalization of *Newton's method* to $\mathbb{R}^n$. The *homotopy* method can be used to find a starting value if all other inspiration fails.

Chapter 10

The heat- or diffusion equation

Objectives

In this chapter several numerical methods to solve the heat equation are considered. Since this equation also describes diffusion, the equation is referred to as the diffusion equation. The equation describes very common processes in physics and engineering and we would like our numerical models to inherit certain properties of the physics. The most important aspect - and typical for diffusion equations - is the property that the solution tends to an equilibrium solution as time proceeds. If the coefficients in the heat equation and the boundary conditions do not depend on time, there exists exactly one equilibrium solution, and the solution of the heat equation tends to this equilibrium solution independent of the initial condition.

10.1 A fundamental inequality

The next theorem states this result more precisely.

Theorem 10.1.1 *Let Ω be a bounded domain in $\mathbb{R}^2$, let Δ be given by*

$$\Delta = \text{div grad } = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}. \tag{10.1.1}$$

Let $u_E(\mathbf{x})$ be the solution of

$$\Delta u + f(\mathbf{x}) = 0, \tag{10.1.2}$$

with boundary conditions

$$u(\mathbf{x}) = g_1(\mathbf{x}), \quad \mathbf{x} \in \Gamma_1 \tag{10.1.3}$$

$$\frac{\partial u}{\partial n}(\mathbf{x}) = g_2(\mathbf{x}), \quad \mathbf{x} \in \Gamma_2 \tag{10.1.4}$$

$$(\sigma u)(\mathbf{x}) + \frac{\partial u}{\partial n}(\mathbf{x}) = g_3(\mathbf{x}), \quad \mathbf{x} \in \Gamma_3 \tag{10.1.5}$$

Further, let $u(\mathbf{x}, t)$ be the solution of the initial value problem

$$\frac{\partial u}{\partial t} = \Delta u + f(\mathbf{x}), \tag{10.1.6}$$

with initial condition $u(\mathbf{x}, t_0) = u_0(\mathbf{x})$ and boundary conditions (10.1.3)–(10.1.5). Let $R(t)$ be the quadratic residual, which is

$$R(t) = \int_\Omega \left(u(\mathbf{x}, t) - u_E(\mathbf{x})\right)^2 d\Omega, \tag{10.1.7}$$

then, there is a $\gamma > 0$ *such that*

$$R(t) < R(t_0)e^{-\gamma(t-t_0)}, \quad \forall t > t_0. \tag{10.1.8}$$

Proof
Apparently, u_E is a solution of (10.1.6) with $\partial u_E/\partial t = 0$. The difference $v = u_E - u$ satisfies:

$$\frac{\partial v}{\partial t} = \Delta v, \tag{10.1.9}$$

with initial condition $v(\mathbf{x}, t_0) = v_0 = u_E - u_0$ and boundary conditions

$$v(\mathbf{x}) = 0, \quad \mathbf{x} \in \Gamma_1 \tag{10.1.10}$$

$$\frac{\partial v}{\partial n}(\mathbf{x}) = 0, \quad \mathbf{x} \in \Gamma_2 \tag{10.1.11}$$

$$(\sigma v)(\mathbf{x}) + \frac{\partial v}{\partial n}(\mathbf{x}) = 0, \quad \mathbf{x} \in \Gamma_3. \tag{10.1.12}$$

Multiplication of equation (10.1.9) by v and subsequent integration over Ω, gives

$$\int_\Omega v\frac{\partial v}{\partial t}\, d\Omega = \int_\Omega v\Delta v\, d\Omega \tag{10.1.13}$$

$$\int_\Omega \frac{1}{2}\frac{\partial v^2}{\partial t}\, d\Omega = -\int_\Omega \|\text{grad}\, v\|^2\, d\Omega + \int_\Gamma v\frac{\partial v}{\partial n}\, d\Gamma. \tag{10.1.14}$$

Here the right-hand side follows from Green's Theorem 1.3.12. We interchange the order of integration over Ω, differentiate with respect to time and apply the boundary conditions to get:

$$\frac{1}{2}\frac{dR}{dt} = -\int_\Omega \|\text{grad}\, v\|^2\, d\Omega - \int_{\Gamma_3} \sigma v^2\, d\Gamma. \tag{10.1.15}$$

According to Poincaré's Lemma [1], (provided $\Gamma \neq \Gamma_2$), there exists a $\gamma_0 > 0$ such that

$$\int_\Omega \|\text{grad}\, v\|^2\, d\Omega > \gamma_0 \int_\Omega v^2\, d\Omega = \gamma_0 R. \tag{10.1.16}$$

Letting $\gamma = 2\gamma_0$, we obtain:

$$\frac{dR}{dt} < -\gamma R, \tag{10.1.17}$$

hence

$$\frac{dR}{dt} + \gamma R < 0. \tag{10.1.18}$$

This inequality holds for all $t > t_0$. We multiply this equation by $e^{\gamma t}$ to get

$$e^{\gamma t}\left(\frac{dR}{dt} + \gamma R\right) = \frac{d\left(e^{\gamma t}R\right)}{dt} < 0. \tag{10.1.19}$$

After integration from t_0 to t this yields

$$e^{\gamma t}R(t) - e^{\gamma t_0}R(t_0) < 0, \tag{10.1.20}$$

hence

$$R(t) < e^{-\gamma(t-t_0)}R(t_0). \tag{10.1.21}$$

This proves the theorem. □

Remarks

1. The quadratic residual tends to zero exponentially, hence the time dependent solution tends to the equilibrium solution exponentially.
2. If a *Neumann*-boundary condition is given on the *entire* boundary, a compatibility condition (which?) has to be satisfied in order that a physical equilibrium is possible. For this particular case the conditions of the theorem have to be adapted. If the compatibility condition condition is not satisfied, the solution of the time dependent problem is unbounded. Depending on the sign of the net heat production temperature goes to $\pm\infty$.
3. This theorem, proved for the Laplace operator, also holds for the general elliptic operator
$$L = \sum_{\alpha}^{n}\sum_{\beta}^{n} \frac{\partial}{\partial x_\alpha} K_{\alpha\beta} \frac{\partial}{\partial x_\beta},$$
with K positive definite.
4. In a similar way, it is possible to establish *analytical stability* for this problem, i.e. one can demonstrate well-posedness in relation to the initial conditions: Given two solutions u and v with initial conditions u_0 and $u_0 + \epsilon_0$ respectively, then, for $\epsilon(\mathbf{x}, t) = (v - u)(\mathbf{x}, t)$, we have
$$\left(\int_\Omega \epsilon^2 \, d\Omega \right)(t) < e^{-\gamma(t-t_0)} \int_\Omega \epsilon_0^2 \, d\Omega. \tag{10.1.22}$$
Hence, for this problem, we have absolute (asymptotic) stability, because the error tends to zero as $t \to \infty$.

□

Exercise 10.1.1 *Prove Theorem (10.1.1) for the general elliptic operator*
$$L = \sum_{\alpha}^{n}\sum_{\beta}^{n} \frac{\partial}{\partial x_\alpha} K_{\alpha\beta} \frac{\partial}{\partial x_\beta}, \quad K \text{ positive definite.}$$
(Hint: For any symmetric matrix K, $(\mathbf{x}, K\mathbf{x}) \geq \lambda_0(\mathbf{x}, \mathbf{x})$, $\mathbf{x} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$, where λ_0 represents the smallest eigenvalue of K.)

□

Exercise 10.1.2 *Demonstrate the analytic absolute stability of (10.1.6).* □

10.2 Method of lines

A very general method to solve time dependent problems is the *method of lines*. In this method we start with the *spatial* discretization of the problem
$$\frac{\partial u}{\partial t} = \Delta u + f. \tag{10.2.1}$$
This spatial discretization can be based on Finite Differences, Finite Volumes or on Finite Elements. The spatial discretization results in a system of ordinary differential equations the size of which is determined by the number of parameters used to approximate u. Formally, this system can be written as
$$M \frac{d\mathbf{u}_h}{dt} = S\mathbf{u}_h + \mathbf{f}_h. \tag{10.2.2}$$

The quantities with index $_h$ represent the discrete approximations of the continuous quantities. Note the matrix M, the *mass matrix*, in the left-hand side. It is the identity matrix in Finite Differences, but has different structure in Finite Volumes or Finite Elements. M represents the scaling of the equations in the discretization. The matrix S is a (possibly scaled) discrete representation of the elliptic operator L and for the FEM it is the same as the stiffness matrix of the corresponding elliptic problem. We illustrate the method with a few examples.

10.2.1 One dimensional examples

In this section we consider the following equation with one space coordinate:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f(x,t), \quad x \in [0,1], \tag{10.2.3}$$

with initial condition $u(x,t_0) = u_0(x)$. We look at two different discretization methods.

Example 10.2.1 FDM, Dirichlet
We use as boundary conditions: $u(0) = u(1) = 0$. Similarly as in Chapter 3, the interval $(0,1)$ is divided into sub-intervals of size h, such that $Nh = 1$. The second order derivative is discretized using the second divided difference in each gridnode. In each gridnode x_j, $j = 0 \ldots N$, there is a u_j, which, of course, also depends on time. From the boundary conditions, it follows that $u_0 = 0 = u_N$, hence the remaining unknowns are $u_1, \ldots, u_{N-1}$. After elimination of u_0 and u_N we obtain the following system of ordinary differential equations:

$$\frac{d\mathbf{u}_h}{dt} = S\mathbf{u}_h + \mathbf{f}_h, \tag{10.2.4}$$

with

$$S = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & 0 & \ldots & \ldots & 0 \\ 1 & -2 & 1 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & 1 & -2 & 1 \\ 0 & \ldots & \ldots & 0 & 1 & -2 \end{pmatrix}, \tag{10.2.5}$$

$$\mathbf{u}_h = \begin{pmatrix} u_1 \\ \vdots \\ u_{N-1} \end{pmatrix} \quad \text{and} \quad \mathbf{f}_h = \begin{pmatrix} f_1 \\ \vdots \\ f_{N-1} \end{pmatrix}. \tag{10.2.6}$$

in which $\mathbf{u}_h$ and $\mathbf{f}_h$ both depend on t. □

Example 10.2.2 FVM, right-hand boundary point Neumann
We take as boundary conditions $u(0) = 0$, $u'(1) = 0$. Further, a non-equidistant grid is used with N grid nodes, and $h_i = x_{i+1} - x_i$. As a control volume around the node x_i, the interval $V_i = (x_i - 1/2h_{i-1}, x_i + 1/2h_i)$ is used. Subsequently, we integrate the differential equation over the control volume. This gives:

$$\int_{x_i - 1/2h_{i-1}}^{x_i + 1/2h_i} \frac{\partial u}{\partial t}\, dx = \int_{x_i - 1/2h_{i-1}}^{x_i + 1/2h_i} \frac{\partial^2 u}{\partial x^2} + f\, dx, \tag{10.2.7}$$

hence

$$\frac{\partial}{\partial t}\int_{x_i-1/2h_{i-1}}^{x_i+1/2h_i} u\,dx = \frac{\partial u}{\partial x}\Big|_{x_{i+1/2h}} - \frac{\partial u}{\partial x}\Big|_{x_{i-1/2h}} + \int_{x_i-1/2h_{i-1}}^{x_i+1/2h_i} f\,dx. \tag{10.2.8}$$

For the integrals

$$\int_{x_i-1/2h_{i-1}}^{x_i+1/2h_i} u\,dx \quad \text{and} \quad \int_{x_i-1/2h_{i-1}}^{x_i+1/2h_i} f\,dx,$$

the mid-point rule will be used. □

Exercise 10.2.1 *Give the mass matrix and stiffness matrix for this problem, so that the discretization can be written as*

$$M\frac{d\mathbf{u}_h}{dt} = S\mathbf{u}_h + \mathbf{f}_h. \tag{10.2.9}$$

□

10.2.2 Two-dimensional example

In this section we consider the following equation in two spatial coordinates:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + f(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \tag{10.2.10}$$

with initial condition $u(\mathbf{x}, t_0) = u_0(\mathbf{x})$.

Example 10.2.3 FEM, Neumann, Robin
Take Ω bounded, $\partial u/\partial n = 0$ on Γ_1, $\partial u/\partial n + \sigma u = 0$ on Γ_2, with $\Gamma_1 \cup \Gamma_2 = \Gamma$. We distribute Ω into triangles, multiply (10.2.10) by ϕ_k, integrate by parts and obtain:

$$\begin{aligned}\frac{d}{dt}\sum_{i=1}^{N} u_i \int_\Omega \phi_i\phi_k\,d\Omega = &-\sum_{i=1}^{N} u_i \int_\Omega (\operatorname{grad}\phi_i, \operatorname{grad}\phi_k)\,d\Omega \\ &+ \int_\Gamma \phi_k \frac{\partial u}{\partial n}\,d\Gamma + \int_\Omega f\phi_k\,d\Omega.\end{aligned} \tag{10.2.11}$$

After taking the boundary conditions into account, one obtains:

$$\begin{aligned}\frac{d}{dt}\sum_{i=1}^{N} u_i \int_\Omega \phi_i\phi_k\,d\Omega = &-\sum_{i=1}^{N} u_i \int_\Omega (\operatorname{grad}\phi_i, \operatorname{grad}\phi_k)\,d\Omega \\ &- \sum_{i=1}^{N} u_i \int_{\Gamma_2} \sigma\phi_k\phi_i\,d\Gamma + \int_\Omega f\phi_k\,d\Omega.\end{aligned} \tag{10.2.12}$$

This gives a system of ordinary differential equations of the form

$$M\frac{d\mathbf{u}}{dt} = S\mathbf{u} + \mathbf{f}. \tag{10.2.13}$$

with

$$m_{ki} = \int_{\Omega} \phi_k \phi_i \, d\Omega, \tag{10.2.14}$$

$$s_{ki} = -\int_{\Omega} (\operatorname{grad} \phi_k, \operatorname{grad} \phi_i) \, d\Omega - \int_{\Gamma_2} \sigma \phi_k \phi_i \, d\Gamma, \tag{10.2.15}$$

$$f_k = \int_{\Omega} f \phi_k \, d\Omega. \tag{10.2.16}$$

□

We note that if Newton-Cotes integration is applied to the coefficients of the mass matrix, the mass matrix becomes diagonal. This process is called *lumping*.

10.3 Consistency of the spatial discretization

In Chapter 3 consistency of a discretization of a differential operator was treated. For the FVM and FEM discretization of the diffusion equation, it is necessary to include the scaling of the mass matrix M. This means that consistency of the discretization implies that $M^{-1}Sy$ tends to Ly as h tends to zero. In practical situations this can be hard to verify. In order to determine the order of consistency, it suffices to multiply each equation from a FVM discretization by the area of the control volume. For a FEM discretization it is cumbersome to determine *the order* of the consistency of the approximation of the differential operator. However, a conforming FEM approach is always consistent. Each classical definition is pessimistic about the order of the accuracy (if one uses the rule of thumb: order of consistency = accuracy of the numerical solution). Roughly speaking, the accuracy of the numerical solution is $O(h^{p+1})$ for interpolation polynomials of the order p. For convenience this order of the accuracy of the solution is used as the 'definition' of the consistency of the solution.

We will demonstrate that the truncation error of the spatial discretization, of the system ordinary differential equations, causes an error of the same order for the time dependent PDE. We suppose that the *exact* solution of the heat equation, can be substituted into the *discrete* approximation, to obtain:

$$M\frac{d\mathbf{y}}{dt} = S\mathbf{y} + \mathbf{f} + M\mathbf{E}(t), \tag{10.3.1}$$

where $E_k(t) = O(h^p)$ is the error of the kth equation, which, of course, depends on t. The generic discretization parameter (for instance the diameter of the largest element) is denoted by h and p represents the order of the consistency. In the remaining part of this section, the following properties of S and M will be used:

- M and S are symmetric.
- M is positive definite, S is negative definite (i.e. $(\mathbf{x}, S\mathbf{x}) < 0$, for $\mathbf{x} \neq 0$).
- There is a $\gamma_0 > 0$ such that

$$\frac{(\mathbf{x}, S\mathbf{x})}{(\mathbf{x}, M\mathbf{x})} < -\gamma_0. \tag{10.3.2}$$

Now we will show that the difference between the exact solution of the heat equation and the solution of the system of ordinary differential equations is bounded by the error $\mathbf{E}(t)$. Since M is a positive matrix the expression $\|\mathbf{x}\|_M$ defined by $\|\mathbf{x}\|_M = (\mathbf{x}, M\mathbf{x})^{\frac{1}{2}}$ is a proper vector norm. We formulate our result in this norm.

Theorem 10.3.1 *The difference $\epsilon = \mathbf{y} - \mathbf{u}$ between the exact solution of the heat equation and the solution of the system of ordinary differential equations (10.3.1), satisfies the following estimate:*

$$\|\epsilon\|_M < \frac{1}{\gamma_0} \sup_{t>t_0} \|\mathbf{E}(t)\|_M. \tag{10.3.3}$$

Proof
The proof is similar to the proof of the fundamental inequality of Theorem 10.1.1. We subtract the solution of

$$M\frac{d\mathbf{u}}{dt} = S\mathbf{u} + \mathbf{f}, \tag{10.3.4}$$

from (10.3.1), to obtain:

$$M\frac{d\epsilon}{dt} = S\epsilon + M\mathbf{E}. \tag{10.3.5}$$

Since $\mathbf{y}$ and $\mathbf{u}$ have the same initial condition, we have $\epsilon(t_0) = 0$. Taking the inner product of the above equation with ϵ we get:

$$\frac{1}{2}\frac{d(\epsilon, M\epsilon)}{dt} = (\epsilon, S\mathbf{ffl}) + (\epsilon, M\mathbf{E}), \text{ or} \tag{10.3.6}$$

$$\|\varepsilon\|_M \frac{d\|\varepsilon\|_M}{dt} = (\epsilon, S\epsilon) + (\epsilon, M\mathbf{E}). \tag{10.3.7}$$

With $(\varepsilon, S\varepsilon) < -\gamma_0(\varepsilon, M\varepsilon)$ and Schwartz's inequality $(\varepsilon, M\mathbf{E}) \leq \|\varepsilon\|_M \|\mathbf{E}\|_M$ this transforms into

$$\frac{d\|\varepsilon\|_M}{dt} < -\gamma_0\|\varepsilon\|_M + \|\mathbf{E}\|_M, \tag{10.3.8}$$

and hence

$$\frac{d}{dt}\left(e^{\gamma_0 t}\|\varepsilon\|_M\right) < e^{\gamma_0 t}\|\mathbf{E}\|_M. \tag{10.3.9}$$

We integrate this expression and use $\epsilon_0 = 0$ to obtain

$$e^{\gamma_0 t}\|\varepsilon\|_M < \int_{t_0}^{t} e^{\gamma_0 \tau}\|\mathbf{E}\|_M \, d\tau. \tag{10.3.10}$$

Hence

$$\|\varepsilon\|_M < \frac{1}{\gamma_0}(1 - e^{-\gamma_0(t-t_0)}) \sup_{t>t_0} \|\mathbf{E}\|_M, \tag{10.3.11}$$

and the theorem follows. □

Remark
If $\tilde{y} = \sum_{i=1}^{N} y_i \phi_i$, (i.e. the interpolated value using the exact solution), and if $\tilde{u}$ represents the FEM approximation, then

$$\int_{\Omega} (\tilde{y} - \tilde{u})^2 \, d\Omega = (\epsilon, M\epsilon), \tag{10.3.12}$$

which is straightforward to show. Something similar holds for the FVM approach.

Exercise 10.3.1 *Prove inequality (10.3.2).*
Hint: Consider

$$\sup_{\mathbf{x}} \frac{(\mathbf{x}, S\mathbf{x})}{(\mathbf{x}, \mathbf{x})} \frac{(\mathbf{x}, \mathbf{x})}{(\mathbf{x}, M\mathbf{x})} < \sup_{\mathbf{x}} \frac{(\mathbf{x}, S\mathbf{x})}{(\mathbf{x}, \mathbf{x})} \sup_{\mathbf{y}} \frac{(\mathbf{y}, \mathbf{y})}{(\mathbf{y}, M\mathbf{y})}$$

□

Exercise 10.3.2 *Positive definiteness of M implies that for all α, β and vectors $\mathbf{x}$ and $\mathbf{y}$*

$$(\alpha\mathbf{x} + \beta\mathbf{y}, M(\alpha\mathbf{x} + \beta\mathbf{y})) = \alpha^2 \|\mathbf{x}\|_M{}^2 + 2\alpha\beta(\mathbf{x}, M\mathbf{y}) + \beta^2 \|\mathbf{y}\|_M{}^2 > 0 \qquad (10.3.13)$$

Use this to prove Schwartz's inequality. □

Exercise 10.3.3 *Prove the fundamental inequality of Theorem 10.1.1 for the solution of*

$$M\frac{d\mathbf{u}}{dt} = S\mathbf{u} + \mathbf{f}. \qquad (10.3.14)$$

□

10.4 Time integration

The next step we have to take is to integrate in time our system of ordinary differential equations, that we obtained by the method of lines. To this end we use well known methods for numerical integration of initial value problems, like Euler, improved Euler, Runge-Kutta or the trapezoidal rule.

Example 10.4.1 *Application of Euler's method gives:*

$$M\frac{\mathbf{u}^{n+1}}{\Delta t} = M\frac{\mathbf{u}^n}{\Delta t} + S\mathbf{u}^n + \mathbf{f}^n, \qquad (10.4.1)$$

in which $\mathbf{u}^{n+1}$ and $\mathbf{u}^n$ represent the solutions on t_{n+1} and t_n respectively, with $t_n = t_0 + n\Delta t$. □

So unless M is diagonal we have to solve a system of equations *in each time step* even when we use an explicit integration scheme. In FDM or FVM M is diagonal, but in FEM M has the complexity of the Laplacian operator. If you nevertheless really want to use an explicit integration method (there are several reasons why you would not) you can diagonalize M by a technique known as *lumping*. See Exercise 10.4.1. This technique can be used only for linear basis functions. If you do not lump the mass matrix you have to solve a system with the complexity of the Laplacian in each time step.

Exercise 10.4.1 *Calculate the element mass matrix for linear basis functions*

$$m^e_{ij} = \int_e \lambda_i \lambda_j \, de \qquad (10.4.2)$$

using Newton Cotes' integration rule. Show that the element mass matrix is diagonal and explain that the large mass matrix has to be diagonal too. □

Exercise 10.4.2 *Formulate the implicit method of Euler (backward) for the system of ordinary differential equations as obtained from the method of lines.* □

Exercise 10.4.3 *Formulate the improved Euler method for this system.* □

Example 10.4.2 *The method of Crank-Nicholson or* the trapezoid rule *for our system of ordinary differential equations is given by:*

$$(\frac{M}{\Delta t} - \frac{1}{2}S)\mathbf{u}^{n+1} = (\frac{M}{\Delta t} + \frac{1}{2}S)\mathbf{u}^n + \frac{1}{2}\left(\mathbf{f}^n + \mathbf{f}^{n+1}\right). \qquad (10.4.3)$$

□

Example 10.4.3 *The* θ-method *for the system of ordinary differential equations is given by:*

$$(\frac{M}{\Delta t} - \theta S)\mathbf{u}^{n+1} = (\frac{M}{\Delta t} + (1-\theta)S)\mathbf{u}^n + (1-\theta)\mathbf{f}^n + \theta\mathbf{f}^{n+1}, \tag{10.4.4}$$

where θ *is a real number in the closed interval between zero and one. Note that* $\theta = 0$, $\theta = 1$ *and* $\theta = \frac{1}{2}$ *correspond to the Forward, Backward Euler and the Crank-Nicholson method respectively.* □

For the θ-method it can be shown that the global error in the time integration is of second order if $\theta = \frac{1}{2}$ and else the order of the error is of first order.

10.5 Stability of the numerical integration

In section 10.1 we demonstrated that the heat equation is *absolutely* stable with respect to the initial conditions. This means that if two solutions have different initial conditions, the difference between these two solutions vanishes as $t \to \infty$. This property also holds for the system of ordinary differential equations obtained by the method of lines (see Exercise 10.5.1). We want to make sure that the numerical time integration inherits this property, so that the numerical time integration is absolutely stable as well. Stability of numerical integration methods in time is treated more extensively in [7]. We state the most important results. The stability of the system of ordinary differential equations:

$$\frac{d\mathbf{u}}{dt} = A\mathbf{u} + \mathbf{f}, \tag{10.5.1}$$

is determined by the 'error-equation'

$$\boldsymbol{\frac{d\epsilon}{dt} = A\epsilon.} \tag{10.5.2}$$

1. The system is absolutely stable if and only if the real part of the eigenvalues λ_k of the matrix A is negative, i.e. $\mathrm{Re}(\lambda_k) < 0$.

2. Each numerical solution procedure has an *amplification matrix* $G(\Delta tA)$, given by the numerical solution of (10.5.2):

   $$\boldsymbol{\epsilon^{n+1} = G(\Delta tA)\epsilon^n.} \tag{10.5.3}$$

   If the error equation is *scalar* (i.e. the system reduces to of one equation only: $\epsilon' = \lambda\epsilon$), the matrix reduces to an *amplification factor* , which is denoted by $C(\Delta t\lambda)$.

3. A numerical solution method is absolutely stable if all eigenvalues μ_k of $G(\Delta tA)$ have the property $|\mu_k| < 1$.

4. The eigenvalues μ_k of $G(\Delta tA)$ can be obtained by substitution of the eigenvalues λ_k of the matrix A into the amplification factor:

   $$\mu_k = C(\Delta t\lambda_k). \tag{10.5.4}$$

   Hence, for stability we need $|C(\Delta t\lambda_k)| < 1$.

Exercise 10.5.1 *The amplification matrices for forward Euler, improved Euler, backward Euler, Crank-Nicholson and the θ-method are given by*

$$\begin{aligned} I + \Delta t A, \\ I + \Delta t A + \frac{1}{2}(\Delta t A)^2, \\ (I - \Delta t A)^{-1}, \\ (I - \frac{1}{2}\Delta t A)^{-1}(I + \frac{1}{2}\Delta t A), \\ (I - \theta \Delta t A)^{-1}(I + (1-\theta)\Delta t A). \end{aligned}$$

Show this. What are the corresponding amplification factors? □

If the mass matrix is not I we have $A = M^{-1}S$, hence, in order to investigate the stability of the numerical time integration, the eigenvalues of $M^{-1}S$ have to be estimated. We note that the eigenvalues of $M^{-1}S$ are the same as the eigenvalues of the generalized eigenvalue problem:
Determine λ and $\mathbf{x} \neq \mathbf{0}$, such that

$$S\mathbf{x} = \lambda M\mathbf{x}. \tag{10.5.5}$$

All eigenvalues of the above generalized eigenvalue are real-valued and negative, since S is negative definite and M is positive definite. (See [36]). For real-valued eigenvalues, the following criterion for stability holds

$$\Delta t < \frac{c}{|\lambda_{max}|}, \tag{10.5.6}$$

with $c = 2$ for Euler and improved Euler and $c = 2.8$ for Runge-Kutta (see [7]). Hence we have to estimate the maximal eigenvalue of the generalized eigenvalue problem. This is treated in the next section.

10.5.1 Gershgorin's circle theorem

The most interesting case to consider is that M is diagonal. We may formulate Gershgorin's Theorem for this case to estimate the lie of the eigenvalues.

Theorem 10.5.1 (Gershgorin)
Let M be diagonal, then, for all eigenvalues λ of $M^{-1}S$ holds:

$$|m_{kk}\lambda - s_{kk}| \leq \sum_{i=1_{i\neq k}}^{N} |s_{ki}|. \tag{10.5.7}$$

Remark:
Eigenvalues may be complex valued in general and for complex eigenvalues $\lambda = \mu + i\nu$, the absolute value is the *modulus:* $|\lambda| = \sqrt{\mu^2 + \nu^2}$. So the eigenvalues are located within a circle in the complex plane and that is the reason why the theorem is also often referred to as Gershgorin's *circle* theorem. But for symmetric M and S the eigenvalues of $M^{-1}S$ are real-valued.

Proof
Let λ be an eigenvalue of the generalized eigenvalue problem with corresponding eigenvector $\mathbf{v}$, then,

$$\sum_i s_{pi}v_i = \lambda m_{pp}v_p, \quad p = 1, \ldots, N. \tag{10.5.8}$$

Let v_k be the component of $\mathbf{v}$ with the largest modulus. For this index l we have

$$\lambda m_{kk} - s_{kk} = \sum_i \sum_{i \neq k} s_{ki} \frac{v_i}{v_k}, \tag{10.5.9}$$

and because $|v_i/v_k| \leq 1$, we get

$$|\lambda m_{kk} - s_{kk}| \leq \sum_i \sum_{i \neq k} |s_{ki}|. \tag{10.5.10}$$

This proves the theorem. □

Example 10.5.1 *For the heat equation in one spatial dimension (see example 10.2.1) the Finite Difference Method gives $M = I$ and hence*

$$|\lambda_{max}| < \frac{4}{h^2}. \tag{10.5.11}$$

From this we obtain a stability criterion for the Forward Euler method:

$$\Delta t < \frac{2h^2}{4} = \frac{1}{2}h^2. \tag{10.5.12}$$

Application of a two dimensional Finite Difference Method (see example 10.4.1) with two spatial coordinates, gives in a similar way:

$$|\lambda_{max}| < \frac{4}{(\Delta x)^2} + \frac{4}{(\Delta y)^2}, \tag{10.5.13}$$

and a stability criterion of the form

$$\Delta t < \frac{\beta^2}{2(1+\beta^2)}(\Delta x)^2, \tag{10.5.14}$$

in which $\Delta y = \beta \Delta x$. □

Example 10.5.2 *Lumping the mass matrix in Example 10.2.2 gives $m_{ii} = \frac{1}{2}(h_{i-1} + h_i)$. Gershgorin's Theorem results in the following estimate:*

$$|\lambda_{max}| < \sup_i \frac{2}{h_{i-1} + h_i} \left(\frac{2}{h_{i-1}} + \frac{2}{h_i} \right) \tag{10.5.15}$$

$$= \sup_i \frac{4}{h_{i-1} h_i}, \tag{10.5.16}$$

and a stability criterion of the form

$$\Delta t < \frac{1}{2} \inf_i (h_{i-1} h_i). \tag{10.5.17}$$

□

In all the examples the time step has to be smaller than the product of a factor times the square of the grid spacing. In practical situations, this could imply that the time step has to be very small. For that reason explicit time integration methods are not popular for the heat equation. Implicit methods such as the Crank-Nicholson method or the implicit Euler (backward) method are usually preferred. *This always implies the solution of a problem with the complexity of the Laplacian in each time step*. In one space dimension, this amounts to the solution of a tridiagonal system of equations in each time step, which is no big deal. Two and more space dimensions however lead to the same type of problems as the Laplacian. For iterative methods the solution on the previous time level is of course an excellent starting value.

For regions with simple geometries some special implicit methods for the heat equation are available. This will be addressed later.

Exercise 10.5.2 *Prove that Euler backward and Crank-Nicholson are absolutely stable for each value of the step-size Δt if $Re(\lambda_k) < 0$.* □

Exercise 10.5.3 *Prove that the θ-method is absolutely stable for all time steps if $\theta \geq \frac{1}{2}$. Derive a condition for stability for the case that $\theta < \frac{1}{2}$.* □

As an illustration of the stability of the numerical solution to the heat problem we consider a Finite Element solution in the square $\Omega = [0,1] \times [0,1]$, on which

$$\frac{\partial u}{\partial t} = 0.5 \Delta u. \tag{10.5.18}$$

We take as initial condition and boundary condition at all the boundaries Γ:

$$\begin{aligned} u(x,y,0) &= \sin(x)\sin(y), (x,y) \in \Omega \\ u(x,y,t) &= \sin(x)\sin(y), (x,y) \in \Gamma. \end{aligned} \tag{10.5.19}$$

Exercise 10.5.4 *Prove that the analytical solution to the above problem is given by:*

$$u(x,y,t) = e^{-t}\sin(x)\sin(y). \tag{10.5.20}$$

□

In Figure 10.1 we show the numerical solution to the above problem as computed by the use of the Forward Euler method with $\Delta t = 0.1$. For this case the stability criterion is violated and hence the solution exhibits unphysical behavior. In Figure 10.2 we show the solution that has been obtained for the same data by the backward Euler method. Now the solution exhibits the expected physical behavior. The contourlines are nice and smooth and are similar to the ones of the analytical solution.

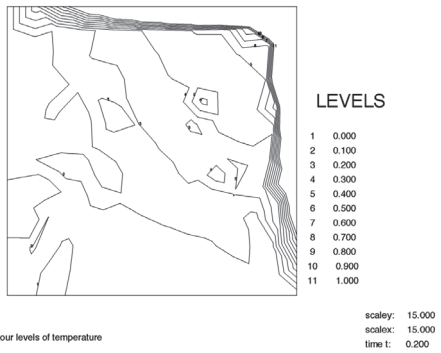


Figure 10.1: Contourlines of the numerical solution to the heat equation with $\Delta t = 0.1$ as obtained by the use of the *Forward* (explicit) Euler method (unstable solution).

10.5.2 Stability analysis of Von Neumann

As an alternative method to estimate the eigenvalues of the matrix $M^{-1}S$ we present a method due to the American mathematician John Von Neumann. This method

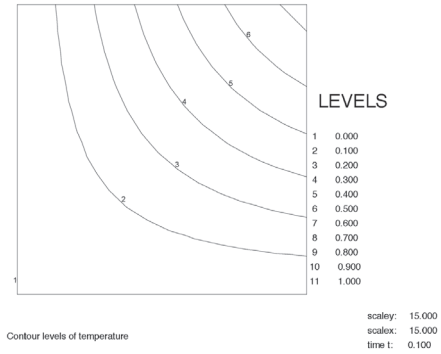


Figure 10.2: Contourlines of the numerical solution to the heat equation with $\Delta t = 0.1$ as obtained by the use of the *Backward* (implicit) Euler method.

has gained much popularity. For equations with *constant coefficients* and *equidistant grids* it can be shown that eigenvectors of $M^{-1}S$ can be written as

$$v_k = e^{i\rho kh} \tag{10.5.21}$$

in one and

$$v_{kl} = e^{i(\rho k\Delta x + \sigma l\Delta y)} \tag{10.5.22}$$

in two space dimensions. The region must be rectangular in 2 D. The numbers ρ and σ depend on the type of boundary conditions. In order to estimate an upper bound for the eigenvalues it is sufficient to substitute these expressions in one single equation of the generalized eigenvalue problem.

Example 10.5.3 *As an example we consider the heat equation with an equidistant grid in one space dimension*

$$\lambda e^{i\rho kh} = \frac{1}{h^2}(e^{i\rho(k-1)h} - 2e^{i\rho kh} + e^{i\rho(k+1)h}). \tag{10.5.23}$$

We divide the left and right-hand sides of this equation by $e^{i\rho kh}$ and obtain using the relation $1/2(e^{i\phi} + e^{-i\phi}) = \cos\phi$:

$$\lambda = \frac{2(\cos(\rho h) - 1)}{h^2} = -4\frac{\sin^2 \rho h/2}{h^2}. \tag{10.5.24}$$

From this we find for the eigenvalues the estimate:

$$|\lambda| \leq \frac{4}{h^2} \tag{10.5.25}$$

and the stability criterion

$$\Delta t < \frac{1}{2}h^2, \tag{10.5.26}$$

for the forward Euler time-integration. □

Remark

1. Von Neumann's original method also uses time dependence and calculates *amplification factors* directly. Our presentation is more in line with the method of lines.

2. The domain of computation, in which the Von Neumann analysis is applied, does not necessarily have to be rectangular. In that case the analysis gives a rough upper bound for the eigenvalues, which in fact holds for the smallest rectangle that encloses the domain of computation. The coefficients in the PDE have to be constant. Furthermore the discretization has to be equidistant, otherwise the analysis is not valid. If both Gershgorin's Theorem and the Von Neumann analysis can be applied, these methods give the same stability criterion. Gershgorin's Theorem can also be applied for non-constant coefficients and non-equidistant grids. But the mass matrix has to be diagonal in that case.

10.6 The accuracy of the time integration

When we use a numerical method for time integration we make an error at each time step. These errors accumulate in general, and you might ask if this accumulation could be disastrous. From [7] we know that in a bounded time interval $(t_0, T]$ a local truncation error of the order $O(h^m)$ gives a global error of the same order. The forward and backward methods of Euler have $m = 1$, whereas the improved Euler method and the method of Crank-Nicholson have $m = 2$. Absolutely stable systems like the heat equation have even better properties. If the numerical integration is stable, the global error is uniformly bounded on the interval (t_0, ∞).

Theorem 10.6.1 *Let $\mathbf{y}(t)$ be the solution of the absolutely stable system*

$$\frac{d\mathbf{y}}{dt} = A\mathbf{y} + \mathbf{f}, \quad \mathbf{y}(t_0) = y_0. \tag{10.6.1}$$

Further, let $\mathbf{u}^n$ be the solution of the numerical method

$$\mathbf{u}^{n+1} = G(\Delta t A)\mathbf{u}^n + I_n(\mathbf{f}), \quad \mathbf{u}^0 = \mathbf{y}_0, \tag{10.6.2}$$

where $I_n(f)$ represents an approximation of

$$\int_{t_n}^{t_{n+1}} \mathbf{f} dt, \tag{10.6.3}$$

so that

1. $$\mathbf{y}(t_{n+1}) = G(\Delta t A)\mathbf{y}(t_n) + I_n(\mathbf{f}) + (\Delta t)^{m+1}\mathbf{p}^n. \tag{10.6.4}$$
   Here $\|p^n\|$ is uniformly bounded for all n and Δt.
2. $\lim_{n\to\infty} G(\Delta t A)^n \to 0, \quad \forall \Delta t < \tau,$

then, the following holds

$$\|\mathbf{y}(t_n) - \mathbf{u}^n\| = O((\Delta t)^m). \tag{10.6.5}$$

In other words: if the local truncation error in time is of order m (after division of equation (10.6.4) by Δt), the global error is also of order m provided the integration is stable.

Proof
We define $\epsilon^n = y(t_n) - u^n$and subtract Equation (10.6.2) from Equation (10.6.4) to get:

$$\boldsymbol{\epsilon^{n+1} = G(\Delta t A)\epsilon^n + (\Delta t)^{m+1}\mathbf{p}^n.} \tag{10.6.6}$$

Now $\epsilon_0 = 0$ and we shall show by induction that

$$\boldsymbol{\epsilon^n = (\Delta t)^{m+1} \sum_{k=0}^{n-1} G(\Delta t A)^{n-k-1} \mathbf{p}^k.} \tag{10.6.7}$$

Equation (10.6.7) holds for $n = 0$. Assume Equation (10.6.7) holds until n. We obtain

$$\epsilon^{n+1} = G(\Delta t A)\epsilon^n + (\Delta t)^{m+1}\mathbf{p}^n \tag{10.6.8}$$

$$= (\Delta t)^{m+1} G(\Delta t A) \sum_{k=0}^{n-1} G(\Delta t A)^{n-k-1}\mathbf{p}^k + (\Delta t)^{m+1}\mathbf{p}^n \tag{10.6.9}$$

$$= (\Delta t)^{m+1} \sum_{k=0}^{n} G(\Delta t A)^{n-k}\mathbf{p}^k. \tag{10.6.10}$$

From this we conclude that (10.6.7) holds for all n. $\|p^n\|$ is uniformly bounded, so there exists a vector $\mathbf{p}_{max}$ with $\|\mathbf{p}^n\| < \|\mathbf{p}_{max}\|$ for all n. Putting this into (10.6.10) we obtain

$$\boldsymbol{\|\epsilon^n\| \leq (\Delta t)^{m+1} \sum_{k=0}^{n-1} \|G(\Delta t A)^{n-k-1}\| \|\mathbf{p}_{max}\|.} \tag{10.6.11}$$

We use the diagonalization of $G(\Delta t A)$:

$$G(\Delta t A) = Q^{-1} M Q, \tag{10.6.12}$$

where Q is a matrix with the eigenvectors of $G(\Delta t A)$ as columns and M is a diagonal matrix with the eigenvalues of $G(\Delta t A)$. For $\|G(\Delta t A)^k\|$ we have

$$G(\Delta t A)^k = Q^{-1} M^k Q \tag{10.6.13}$$

$$\|G(\Delta t A)^k\| \leq |\mu_1^k| \, \|Q^{-1}\| \, \|Q\|. \tag{10.6.14}$$

μ_1 is the eigenvalue of $G(\Delta t A)$ with the largest modulus. This gives

$$\boldsymbol{\|\epsilon^n\| \leq (\Delta t)^{m+1} \frac{1 + |\mu_1^n|}{1 - |\mu_1|} \|Q^{-1}\| \, \|Q\| \, \|\mathbf{p}_{max}\|.} \tag{10.6.15}$$

Since $\mu_1 = C(\lambda_1 \Delta t) = 1 + \lambda_1 \Delta t + O(\Delta t^2)$, we have $1 - \mu_1 = \lambda_1 \Delta t + O(\Delta t^2)$ and we finally obtain

$$\boldsymbol{\|\epsilon^n\| \leq K(\Delta t)^m,} \tag{10.6.16}$$

which proves the theorem. □

10.7 Conclusions for the method of lines

We summarize the results of the methods of lines for the heat/diffusion equation.

- Using the method of lines, the PDE is written as a system of ordinary differential equations by the spatial discretization of the elliptic operator.
- The global error of the *analytic* solution of this system of ordinary differential equations (compared to the solution of the solution of the PDE) is of the same order as the consistency of the FDM and FVM and an order higher than the degree of the interpolation polynomials of the FEM.

- The *numerical* solution of this system has an additional error due to the numerical time integration. This global error is of the order of $K\Delta t^m$, if the local truncation error is of the order $O(\Delta t^m)$ This constant does not depend on time t and this estimate holds at the *entire* time interval (t_0, ∞).
- Explicit (and some implicit) methods have a stability criterion of the form

$$\Delta t < c\Delta x^2 \tag{10.7.1}$$

  and hence these methods are less suitable for the heat equation.

10.8 Special difference methods for the heat equation

The method of lines is a general method, which is applicable to one, two or three spatial dimensions. At each time step, the implicit methods give a problem to be solved with the same complexity as the Poisson problem. Therefore, one has searched for methods that are *stable* but have a simpler complexity than the Poisson problem. We present one example of such a method: The ADI method. This method can only be used with regular grids with a five-point molecule for the elliptic operator. Unfortunately, the ADI method cannot be used if the elliptic operator is discretized using a general Finite Element Method. First we sketch the principle of the ADI method and subsequently a formal description of the ADI method is given.

10.8.1 The principle of the ADI method

The abbreviation ADI means *Alternating Direction Implicit*. This is a fairly accurate description of the working of the method. Suppose that we have to solve the heat equation on a rectangle with length l_x and width l_y and we use a discretization with stepsize Δx and Δy respectively, such that $N_x\Delta x = l_x$ and $N_y\Delta y = l_y$. For convenience we apply Dirichlet boundary conditions at all the boundaries of the domain of computation, where we set $u = 0$. For the time integration of t_n up to t_{n+1} the ADI method uses two steps. The idea is as follows: first we use a half time step with an intermediate auxiliary quantity u^*. To compute u^* we use the implicit Euler time integration method for the derivative with respect to x and the explicit Euler time integration for the derivative with respect to y. In the next half time step, we reverse this process. Hence: The first step, a so-called half time step, computes an auxiliary-quantity u^*_{ij} according to:

$$\begin{aligned} u^*_{ij} = u^n_{ij} + \frac{\Delta t}{2\Delta x^2}(u^*_{i+1,j} - 2u^*_{i,j} + u^*_{i-1,j}) + \\ \frac{\Delta t}{2\Delta y^2}(u^n_{i,j+1} - 2u^n_{i,j} + u^n_{i,j-1}) + \Delta t f^*_{ij}, \\ i = 1, \ldots N_x - 1, j = 1 \ldots N_y - 1, \end{aligned} \tag{10.8.1}$$

where f^*_{ij} denotes $f(i\Delta x, j\Delta y, t_0 + (n + \frac{1}{2})\Delta t)$. Subsequently u^{n+1} is calculated according to:

$$\begin{aligned} u^{n+1}_{ij} = u^*_{ij} + \frac{\Delta t}{2\Delta x^2}(u^*_{i+1,j} - 2u^*_{i,j} + u^*_{i-1,j}) + \\ \frac{\Delta t}{2\Delta y^2}(u^{n+1}_{i,j+1} - 2u^{n+1}_{i,j} + u^{n+1}_{i,j-1}) + \Delta t f^*_{ij}, \\ i = 1, \ldots N_x - 1, j = 1 \ldots N_y - 1. \end{aligned} \tag{10.8.2}$$

In Equation (10.8.1) for a fixed index j a tridiagonal system of equations has to be solved in $\mathbf{u}_j^*$, with

$$\mathbf{u}_j^* = \begin{pmatrix} u_{1j}^* \\ u_{2j}^* \\ \vdots \\ u_{N_x-1,j}^* \end{pmatrix}. \tag{10.8.3}$$

In total there are $N_y - 1$ systems like this one to be solved in order to determine all the values of $\mathbf{u}_j^*$. Similarly, one has to solve in Equation (10.8.2) for a fixed index i a tridiagonal system of equations in $\mathbf{u}_i^{n+1}$, with

$$\mathbf{u}_i^{n+1} = \begin{pmatrix} u_{i1}^{n+1} \\ u_{i2}^{n+1} \\ \vdots \\ u_{i,N_y-1}^{n+1} \end{pmatrix}. \tag{10.8.4}$$

This is exactly in the other direction, which explains the name of the method. In total we are faced with $N_x - 1$ of such systems. Hence to integrate the heat equation from t_n up to t_{n+1} one has to

- solve $N_y - 1$ tridiagonal systems of size $N_x - 1$
- solve $N_x - 1$ tridiagonal systems of size $N_y - 1$

Exercise 10.8.1 *Verify that the amount of computational effort per time step for the ADI method is proportional to the total number of gridpoints. (Hint: How many operations does it take to solve a $N \times N$ tridiagonal system of equations?) Further, verify that the direct solution of the problem with the method of lines using for instance the method of Crank-Nicholson with a profile-method takes a computational effort which is proportional to $(N_x - 1)^2(N_y - 1)$ of $(N_x - 1)(N_y - 1)^2$, depending on the used numbering of the unknowns.* □

Indeed the computational complexity of the ADI method is better than that of the method of lines. However, the question remains whether this benefit is not at the expense of the accuracy or the stability of the method. To scrutinize this, a formal description of the ADI method is presented in the next section.

10.8.2 Formal description of the ADI method

The ADI method can be seen as a special way to integrate the system of ordinary differential equations

$$\frac{d\mathbf{u}}{dt} = (A_x + A_y)\mathbf{u} + \mathbf{f}, \tag{10.8.5}$$

which arises from a PDE using the method of lines. The ADI method of this system is given by:

$$\mathbf{u}^* = \mathbf{u}^n + \frac{1}{2}\Delta t(A_x\mathbf{u}^* + A_y\mathbf{u}^n + \mathbf{f}^*) \tag{10.8.6}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^* + \frac{1}{2}\Delta t(A_x\mathbf{u}^* + A_y\mathbf{u}^{n+1} + \mathbf{f}^*). \tag{10.8.7}$$

From this the intermediate step $\mathbf{u}^*$ can be eliminated:

$$(I - \frac{1}{2}\Delta t A_x)\mathbf{u}^* = (I + \frac{1}{2}\Delta t A_y)\mathbf{u}^n + \frac{1}{2}\Delta t \mathbf{f}^* \tag{10.8.8}$$

$$(I - \frac{1}{2}\Delta t A_y)\mathbf{u}^{n+1} = (I + \frac{1}{2}\Delta t A_x)\mathbf{u}^* + \frac{1}{2}\Delta t \mathbf{f}^* \tag{10.8.9}$$

and from multiplication of the top part of this expression by $I + \frac{1}{2}\Delta t A_x$ and the bottom part with $I - \frac{1}{2}\Delta t A_x$ and from the fact that these matrices commute, one obtains:

$$(I - \frac{1}{2}\Delta t A_x)(I - \frac{1}{2}\Delta t A_y)\mathbf{u}^{n+1} = (I + \frac{1}{2}\Delta t A_x)(I + \frac{1}{2}\Delta t A_y)\mathbf{u}^n + \Delta t \mathbf{f}^*. \tag{10.8.10}$$

Equation (10.8.10) is the basis of our investigations. First, we make a statement about the accuracy.

Theorem 10.8.1 *Equation (10.8.10) differs from Crank-Nicholson's method applied in (10.8.5) by a term of the order of* $O(\Delta t^3)$.

Proof
Crank-Nicholson applied on 10.8.5 gives

$$(I - \frac{1}{2}\Delta t A_x - \frac{1}{2}\Delta t A_y)\mathbf{u}^{n+1} = (I + \frac{1}{2}\Delta t A_x + \frac{1}{2}\Delta t A_y)\mathbf{u}^n + \frac{1}{2}\Delta t(\mathbf{f}^n + \mathbf{f}^{n+1}). \tag{10.8.11}$$

Elaboration of 10.8.10 gives:

$$(I - \frac{1}{2}\Delta t A_x - \frac{1}{2}\Delta t A_y)\mathbf{u}^{n+1} = (I + \frac{1}{2}\Delta t A_x + \frac{1}{2}\Delta t A_y)\mathbf{u}^n + \frac{1}{4}\Delta t^2 A_x A_y(\mathbf{u}^n - \mathbf{u}^{n+1}) + \Delta t \mathbf{f}^*. \tag{10.8.12}$$

Now the theorem immediately follows by noting that $\mathbf{u}^n - \mathbf{u}^{n+1}$ is of order $O(\Delta t)$ and that $\mathbf{f}^* = \frac{1}{2}(\mathbf{f}^n + \mathbf{f}^{n+1}) + O(\Delta t^2)$. Hence the ADI method has the same accuracy as the method of Crank Nicholson, which is $O(\Delta t^2)$. □

It is hard to investigate the stability of the ADI method theoretically. In practical situations, it turns out that the ADI method does not require a stringent stability criterion. In a special case, there is a theoretical justification for the unconditional stability of the ADI method:

Theorem 10.8.2 *If A_x and A_y are commuting matrices (i.e. $A_x A_y = A_y A_x$), then, the ADI method is unconditionally stable.*

Proof
We have to calculate the eigenvalues of

$$(I - \frac{1}{2}\Delta t A_y)^{-1}(I - \frac{1}{2}\Delta t A_x)^{-1}(I + \frac{1}{2}\Delta t A_x)(I + \frac{1}{2}\Delta t A_y)$$

but under the conditions of the conditions of the theorem all these matrices commute. Then, the eigenvalues of these matrices are given by the product of the separate matrices

$$(I - \frac{1}{2}\Delta t A_x)^{-1}(I + \frac{1}{2}\Delta t A_x) \text{ and } (I - \frac{1}{2}\Delta t A_y)^{-1}(I + \frac{1}{2}\Delta t A_y).$$

These eigenvalues are

$$\frac{1 + \frac{1}{2}\Delta t \lambda_x}{1 - \frac{1}{2}\Delta t \lambda_x} \text{ and } \frac{1 + \frac{1}{2}\Delta t \lambda_y}{1 - \frac{1}{2}\Delta t \lambda_y}.$$

Since λ is real-valued is negative, the moduli of all these eigenvalues are less than one. □

Exercise 10.8.2 *Show that the operators A_x and A_y commute on the problem of the rectangle with Dirichlet conditions.* □

Extension of the ADI method to three spatial dimensions is not straightforward. The most straightforward way (three steps, subsequently for the x-,y- and z coordinate) is no longer unconditionally stable. Further, its global error is of the order $O(\Delta t)$. There exist adequate ADI methods for three spatial coordinates, see [26].

10.9 Summary of Chapter 10

In this chapter we paid attention to the numerical solution of the *heat* or *diffusion* equation. We have shown, that with one exception this equation has an equilibrium solution and that independent of the initial values the transient solution tends to this equilibrium solution exponentially fast.

We introduced the *method of lines* for the numerical solution which transforms the PDE into a set of ODE's by discretizing first the spatial differential operators. We estimated the effect of the truncation error of the spatial discretization on the solution of this system of ODE's. We proved that this effect is uniformly bounded. Finite Volume and Finite Element methods generate a *mass matrix*. The mass matrix of the FEM has the same complexity as the Laplacian operator. For that reason even for explicit time integration methods a system of equations of that complexity has to be solved in each time step. This is not necessary if the mass matrix is diagonal and therefore one often lumps the mass matrix, transforming it into a diagonal matrix. This procedure is only possible for linear approximation.

We briefly considered the stability of the explicit integration schemes for which we had to estimate the lie of the eigenvalues of the system matrix. To this end we could use *Gershgorin's circle theorem* or *Von Neumann's stability analysis.*

Finally we considered the *ADI-method*, an unconditionally stable method of considerable less complexity than Crank-Nicolson's method, but with the same accuracy.

Chapter 11

The wave equation

Objectives

In this chapter we shall look at various methods for the time integration of the wave equation. This equation is crucial in applications dealing with electromagnetic radiation, wave propagation, acoustics and seismics (used for oil finding for instance). Before we do this, a conservation principle for the solution of the wave equation is derived. The numerical solution should satisfy this principle as well. Stability in terms of decay and growth of the numerical solution as a function of time is investigated for several methods. Furthermore, the concepts *dispersion* and *dissipation* will be introduced and an illustration of these concepts will be given. Finally a procedure to derive the CFL-criterion, a criterion for the numerical solution to represent the exact solution, will be given by use of the characteristic curves in the x, t-plane.

11.1 A fundamental equality

Consider the wave equation on a domain Ω

$$\frac{\partial^2 u}{\partial t^2} = c^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) =: c^2 \Delta u. \tag{11.1.1}$$

In Equation (11.1.1) no *internal* energy source term is taken into account. Further, homogeneous boundary conditions are imposed on the boundaries Γ_1, Γ_2 and Γ_3 of the domain Ω, i.e.

$$\begin{aligned} u &= 0, && (x,y) \in \Gamma_1, \\ \frac{\partial u}{\partial n} &= 0, && (x,y) \in \Gamma_2, \\ \sigma u + \frac{\partial u}{\partial n} &= 0, && (x,y) \in \Gamma_3. \end{aligned} \tag{11.1.2}$$

Hence there is no transport of energy through the boundaries. Therefore the PDE (11.1.1) with boundary conditions (11.1.2) is homogeneous. As initial conditions, we have that u and $\dfrac{\partial u}{\partial t}$ are given at $t = 0$ at all points in the domain of computation. Now we will show that the 'energy' of this equation is preserved in time.

Theorem 11.1.1 *The homogeneous wave equation* (11.1.1) *with homogeneous boundary*

conditions (11.1.2) *satisfies the following conservation principle*

$$\frac{1}{2}\int_{\Omega}\left\{\left(\frac{\partial u}{\partial t}\right)^2 + c^2||\text{grad}\, u||^2\right\}\, d\Omega + \frac{1}{2}\int_{\Gamma_3}\sigma c^2 u^2 d\Gamma = \textit{Constant}. \tag{11.1.3}$$

Proof: We multiply both sides of the equality of Equation (11.1.1) by $\frac{\partial u}{\partial t}$ and integrate the results over the domain Ω to obtain

$$\int_{\Omega}\frac{1}{2}\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right)^2 d\Omega = \int_{\Omega} c^2\frac{\partial u}{\partial t}\Delta u\, d\Omega = \int_{\Omega} c^2\frac{\partial u}{\partial t}\text{div grad}\, u\, d\Omega. \tag{11.1.4}$$

Assuming that all derivatives are continuous and using the product rule for differentiation, the integrand of the right-hand side can be written as

$$\text{div}\left(\text{grad}\,(u)\frac{\partial u}{\partial t}\right) - \text{grad}\,(u)\cdot\text{grad}\left(\frac{\partial u}{\partial t}\right). \tag{11.1.5}$$

This yields

$$\int_{\Omega}\frac{1}{2}\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right)^2 d\Omega = \int_{\Omega} c^2\text{div}\left(\text{grad}\, u\frac{\partial u}{\partial t}\right)\, d\Omega - \int_{\Omega} c^2\text{grad}\,(u)\cdot\text{grad}\left(\frac{\partial u}{\partial t}\right)\, d\Omega. \tag{11.1.6}$$

We apply the Divergence Theorem to the first term on the right-hand side and use the product rule for differentiation on the second term of the right-hand side to get

$$\int_{\Omega}\frac{1}{2}\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right)^2 d\Omega = \int_{\Gamma_1\cup\Gamma_2\cup\Gamma_3} c^2\frac{\partial u}{\partial n}\frac{\partial u}{\partial t}\, d\Gamma - \frac{1}{2}\int_{\Omega} c^2\frac{\partial}{\partial t}\left(\text{grad}\,(u)\cdot\text{grad}\,(u)\right)\, d\Omega. \tag{11.1.7}$$

The boundary integral on the right-hand side vanishes on Γ_1 and Γ_2 due to the boundary conditions. Application of the boundary condition on Γ_3 then transforms Equation (11.1.7) into

$$\int_{\Omega}\frac{1}{2}\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right)^2 d\Omega = -\int_{\Gamma_3} c^2\sigma u\frac{\partial u}{\partial t}\, d\Gamma - \frac{1}{2}\int_{\Omega} c^2\frac{\partial}{\partial t}\left(\text{grad}\,(u)\cdot\text{grad}\,(u)\right)\, d\Omega. \tag{11.1.8}$$

Finally using a standard differentiation property we get

$$\int_{\Omega}\frac{1}{2}\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right)^2 d\Omega = -\int_{\Gamma_3}\frac{1}{2}c^2\sigma\frac{\partial u^2}{\partial t}d\Gamma - \frac{1}{2}\int_{\Omega} c^2\frac{\partial}{\partial t}\left(\text{grad}\,(u)\cdot\text{grad}\,(u)\right)\, d\Omega. \tag{11.1.9}$$

Interchanging the differentiation and integration operations in the above expression and subsequent integration over time t proves the theorem. □

Remarks

1. Consider the wave equation with a source term

$$\frac{\partial^2 u}{\partial t^2} = c^2\Delta u + f(\mathbf{x}, t). \tag{11.1.10}$$

The difference between two solutions of Equation (11.1.10) with the same source term f and the same boundary conditions satisfies Equation (11.1.3) and homogeneous boundary conditions (11.1.2).

2. The first term in Equation (11.1.3) gives the kinetic energy of the vibrating medium, whereas the second and a third term involve the potential energy. Therefore, Equation (11.1.3) is commonly referred to as the energy norm.
3. The total amount of energy is entirely defined by the two initial conditions $u(x, y, t_0)$ and $\frac{\partial u}{\partial t}(x, y, t_0)$.
4. The difference in this 'energy-norm', between two solutions with the same boundary conditions and different initial conditions is constant at all stages.

Exercise 11.1.1 *Prove remarks 1 and 4.* □

Exercise 11.1.2 *The solution of the heat equation in the previous chapter tends to an equilibrium solution (i.e. a steady-state) as t tends to infinity. Does the solution of the wave equation tend to a steady state as t tends to infinity?* □

From remark 4 it follows that the solution of the wave equation is neutrally stable, that is an error made in the initial conditions will neither decrease nor increase and hence it persists. This property must also hold for our numerical methods. Otherwise the numerical solution would not exhibit the same physical characteristics as the analytical solution

11.2 The method of lines

In a similar way as we did for parabolic equations we may first discretize only the spatial part of the wave equation. The difference with the previous chapter is that we now have to deal with a second order system with respect to time. After the discretization of Equation (11.1.10), we obtain:

$$M\frac{d^2\mathbf{u}}{dt^2} = c^2 S\mathbf{u} + \mathbf{f}, \quad \mathbf{u}(t_0) = \mathbf{u}_0, \quad \frac{d\mathbf{u}}{dt}(t_0) = \mathbf{v}_0. \tag{11.2.1}$$

Here M and S are the mass and stiffness matrices respectively, just like in the previous chapter. Next, we establish that equation (11.2.1) also conserves the energy if $\mathbf{f} = \mathbf{0}$.

Theorem 11.2.1 *Let* $\mathbf{f} = \mathbf{0}$, *then*

$$\frac{1}{2}\left(\frac{d\mathbf{u}}{dt}, M\frac{d\mathbf{u}}{dt}\right) - \frac{1}{2}c^2(\mathbf{u}, S\mathbf{u}) = \textit{constant}. \tag{11.2.2}$$

Exercise 11.2.1 *Prove this theorem. Hint: Use the symmetry of M and S.* □

11.2.1 The error in the solution of the system

Application of the method of lines generates a truncation error $\mathbf{E}$ in the spatial discretization. This may be defined by

$$M\frac{d^2\mathbf{y}}{dt^2} = c^2 S\mathbf{y} + \mathbf{f} + M\mathbf{E}, \tag{11.2.3}$$

where $\mathbf{y}$ denotes the exact solution to the wave equation. This definition holds for Finite Difference and Finite Volume methods. For the Finite Element Method, the order of the truncation error depends on the approximation properties of the basis functions. Under fairly general assumptions it can be shown that this truncation error is equal to the truncation error of the polynomial interpolation of the basis

functions. This truncation error causes an error in the solution of (11.2.1) of the form Ch^p, where h denotes a generic discretization parameter (such as the diameter of the largest element used in the discretization) and p represents the order of consistency (i.e. for Finite Elements it is the order of the polynomial degree plus one). For the heat equation it was possible to find a constant C, valid for the entire interval of integration (t_0, ∞). For the wave equation this is not possible. The constant C depends linearly on the length of the integration interval (t_0, T). A complete analysis of the error is beyond the scope of the book., but qualitatively the phenomenon is explained as follows: An eigenvibration of (11.1.1) is given by a function of the form of $e^{i\lambda ct}U(x, y)$, where U satisfies the *homogeneous* boundary conditions (note that the boundary conditions can be of several types). Substitution into equation (11.1.1) yields

$$-\lambda^2 c^2 U = c^2 \Delta U. \tag{11.2.4}$$

This is just the eigenvalue problem for the Laplace operator, which has an infinite number of solutions in terms of eigenpairs λ_k and U_k. λ_k is the eigenfrequency of the vibration and U_k the eigenfunction. These quantities depend on the domain of computation Ω. Generally speaking the wavelength of the eigenfunction (i.e. the number of peaks) decreases as the eigenfrequency increases.

Consider the discrete version of Equation (11.1.1), which is given by system (11.2.1). We obtain:

$$-\lambda_h^2 c^2 M\mathbf{U} = c^2 S\mathbf{U}. \tag{11.2.5}$$

The subscript h indicates that eigenvalues of the discretized problem are considered. The discretized system only has a finite number of eigenvalues, or put it differently: the resolution is finite on the discrete grid. The shortest wave that can be represented on a grid has wave length $O(2h)$. For eigenfunctions that can be represented well on the grid we have

$$|\lambda - \lambda_h| = O(h^p) \text{ and } ||\mathbf{U} - \mathbf{U}_h|| = O(h^p). \tag{11.2.6}$$

Since the eigenfrequencies of numerical and exact solution differ, the difference between the numerical solution and the exact solution increases as the simulation proceeds. This results in a *phase-shift error*. Moreover, this phase-shift error differs for the different eigenvibrations. This phenomenon is called *dispersion*. Since each solution can be written as a linear combination of eigenvibrations, there will be dispersion in the solution of equation (11.2.1) in relation to the solution of equation (11.1.1). This dispersion even exists for the eigenfunctions, which are represented well on the grid (i.e. eigenfunctions with a large wavelength, i.e. a small frequency). Therefore, the difference between the solution of (11.2.1) and the exact solution of the wave equation (11.1.1) increases as the interval of the time integration increases. Since the error is of the form $C(T - t_0)h^p$, one has to use a more accurate spatial discretization as T increases if the same absolute accuracy is to be maintained for the final stages of time interval as for the initial stages of the computation process.

As an example, we consider

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}, \text{ for } 0 < x < 1, \tag{11.2.7}$$

subject to boundary conditions $u(0, t) = 0 = u(1, t)$ and some initial condition. It can be shown that the eigenvalues and eigenfunctions of the spatial differential operator with the given boundary conditions are respectively given by

$$\lambda_k = k\pi \text{ and } U_k = \sin k\pi x, \quad k = 1, 2, \ldots. \tag{11.2.8}$$

Note that $-\lambda_k^2$ are the actual eigenvalues of the spatial differential operator. Once a finite difference method with an equidistant grid for which $h = \frac{1}{N}$ (where h represents the step size) has been used, it follows that the eigenvalues and eigenvectors of the discretized problem are respectively given by

$$\lambda_{hk} = \frac{2\sin\left(\frac{1}{2}k\pi h\right)}{h}, \text{ and } \mathbf{U}_k = \begin{pmatrix} \sin k\pi h \\ \sin 2k\pi h \\ \dots \\ \sin (N-1)k\pi h \end{pmatrix}. \tag{11.2.9}$$

Note that $-\lambda_{hk}^2$ are the actual eigenvalues of the discretized problem. Note that the eigenvectors are exact. It can be demonstrated that $|\lambda_1 - \lambda_{h1}| = O(h^2)$ and that for $k = \frac{N}{2}$ the phase shift error is already significant. In the following exercise, the claims that we made in this paragraph are sustained by a motivation.

Exercise 11.2.2 *Consider the initial boundary value problem in Equation* (11.2.7)*.*

- *Verify by substitution that the eigenfunctions and eigenvalues are respectively given by*
$$U_k = \sin k\pi x \text{ and } \lambda_k = k\pi, \quad k = 1, 2, \dots. \tag{11.2.10}$$
Note that the eigenvalues of the Laplacian operator are given by $-\lambda_k^2$*.*
- *Use the Finite Difference method to create an equidistant discretization for which* $h = \frac{1}{N}$*, with h representing the stepsize.*
- *Verify by substitution that the eigenfunctions and eigenvectors of the discretized problem are respectively given by*
$$\mathbf{U}_k = \begin{pmatrix} \sin k\pi h \\ \sin 2k\pi h \\ \dots \\ \sin (N-1)k\pi h \end{pmatrix}, \text{ and } \lambda_{hk} = \frac{2\sin\left(\frac{1}{2}k\pi h\right)}{h}. \tag{11.2.11}$$
Note that the eigenvectors are exact. Show, further that $|\lambda_1 - \lambda_{h1}| = O(h^2)$ *and that for* $k = \dfrac{N}{2}$ *the phase-shift error is already significant.*

□

11.3 Numerical time integration

One possibility to integrate equation (11.2.1) numerically is to write it as a system of first order differential equations with respect to time:

$$\begin{aligned} \frac{d\mathbf{u}}{dt} &= \mathbf{v}, \\ M\frac{d\mathbf{v}}{dt} &= c^2 S\mathbf{u} + \mathbf{f}, \end{aligned} \tag{11.3.1}$$

with initial conditions $\mathbf{u}(t_0) = \mathbf{u}_0$ and $\mathbf{v}(t_0) = \mathbf{v}_0$. For this system the ordinary numerical methods for initial value problems can be used.

Example 11.3.1 *Forward Euler applied to System* (11.3.1)*, gives*

$$\begin{aligned} \frac{\mathbf{u}^{n+1}}{\Delta t} &= \frac{\mathbf{u}^n}{\Delta t} + \mathbf{v}^n, \\ M\frac{\mathbf{v}^{n+1}}{\Delta t} &= M\frac{\mathbf{v}^n}{\Delta t} + c^2 S\mathbf{u}^n + \mathbf{f}^n. \end{aligned} \tag{11.3.2}$$

Exercise 11.3.1 *Give the equations for* $\mathbf{u}$ *and* $\mathbf{v}$ *when a Crank-Nicholson time integration of System* (11.3.1) *is applied.* □

11.4 Stability of the numerical integration

From the conservation of energy of the solutions of both the wave equation and the discretization based on the method of lines, it follows that asymptotic stability does not make much sense here. A perturbation of the initial conditions will never vanish. A *fundamental solution* of the form $\mathbf{u}(t) = e^{\lambda t}\mathbf{u}, \mathbf{v}(t) = e^{\lambda t}\mathbf{v}$ of system (11.3.1) with $\mathbf{f} = 0$ has a purely imaginary λ as is shown in the next theorem.

Theorem 11.4.1 *Consider system* (11.3.1) *and let* λ *be an eigenvalue of the generalized eigenvalue problem*

$$\begin{aligned} \lambda\mathbf{u} &= \mathbf{v}, \\ \lambda M\mathbf{v} &= S\mathbf{u}. \end{aligned} \tag{11.4.1}$$

If M *is symmetric positive definite and if* S *is symmetric negative definite, then, the eigenvalues of the above generalized eigenvalue problem are purely imaginary.*

Proof: We substitute the upper equation into the bottom equation, to obtain:

$$\lambda^2 M\mathbf{u} = S\mathbf{u}, \tag{11.4.2}$$

which is the generalized eigenvalue problem for M and S. Next we show that the eigenvalues of the above generalized eigenvalue problem are real-valued and negative.

This amounts to establishing that the eigenvalues of $M^{-1}S$ are negative, real-valued. Since M is symmetric positive definite, the matrix $M^{-1/2}$ exists and $M^{-1}S$ is similar to $M^{1/2}M^{-1}SM^{-1/2} = M^{-1/2}SM^{-1/2}$. This matrix is symmetric and hence all the eigenvalues of $M^{-1}S$ are real-valued (we used the fact that matrices that are similar have the same eigenvalues). Furthermore, S is symmetric negative definite, i.e. $(S\mathbf{x}, \mathbf{x}) < 0$ for all $\mathbf{x} \neq \mathbf{0}$. Hence for $M^{-1/2}SM^{-1/2}$, we have $(M^{-1/2}SM^{-1/2}\mathbf{x}, \mathbf{x}) = (SM^{-1/2}\mathbf{x}, M^{-1/2}\mathbf{x}) < 0$ for all $\mathbf{x} \neq \mathbf{0}$. This implies that the eigenvalues of $M^{-1/2}SM^{-1/2}$ are negative and from similarity the eigenvalues of $M^{-1}S$ are negative as well. Combining this fact with the knowledge that the eigenvalues of $M^{-1}S$ are real-valued, it follows that λ^2 is negative and hence the eigenvalue λ is purely imaginary. This completes the proof. □

With the purely imaginary eigenvalues of the above generalized eigenvalue problem (11.4.1), it follows that the solution of system (11.3.1) is neutrally stable. An absolutely stable time integration method decays the error of the solution and also the solution itself as $t \to \infty$. Whereas an unstable time integration method blows up the error and the solution. This implies that with neither of these time integration methods, the wave equation can be integrated numerically up to any large time t. Hence we have to define an *end time* T and choose the time step Δt accordingly small. If $T = n\Delta t$ and $\lim_{\Delta t \to 0} |C(\lambda \Delta t)|^n = 1$ for a particular method, then, the wave equation can be integrated up to this bounded time T. Note that $n \to \infty$ as $\Delta t \to 0$.

11.5 Total dissipation and dispersion

Since the eigenvalues of (11.4.1) are purely imaginary the solution of (11.3.1) can be written as a linear combination of products of eigenvectors and undamped vi-

brations. Hence it is sufficient to consider a single differential equation of the form

$$\frac{dw}{dt} = i\mu w, \text{ subject to } w(t_0) = w_0. \tag{11.5.1}$$

The behavior of this differential equation qualitatively reflects the behavior of the total system (11.3.1). The exact solution is

$$w(t) = w_0 \mathrm{e}^{i\mu(t-t_0)}. \tag{11.5.2}$$

For the solution at $t^{n+1} = t_0 + (n+1)\Delta t$ we note that

$$w(t^{n+1}) = w(t^n)\mathrm{e}^{i\mu\Delta t}. \tag{11.5.3}$$

Hence the *amplification factor* of the exact solution is given by

$$C(i\mu\Delta t) = \mathrm{e}^{i\mu\Delta t} \Rightarrow |C(i\mu\Delta t)| = 1 \text{ and } \arg(C(i\mu\Delta t)) = \mu\Delta t. \tag{11.5.4}$$

The argument of the amplification factor, $\arg(C(i\mu\Delta t))$, is referred to as the *phase shift*. Hence in each time step there is a phase shift in the exact solution, whereas the modulus of the exact solution does not change.

Exercise 11.5.1 *Show that the complex differential equation* (11.5.1) *is equivalent to the system*

$$\begin{aligned} \frac{du}{dt} &= -\mu v \\ \frac{dv}{dt} &= \mu u, \end{aligned} \tag{11.5.5}$$

where $u = Re\{w\}$ and $v = Im\{w\}$. Show that $w(t) = Constant$ is equivalent to conservation of energy. □

For the numerical method, the following relation holds

$$w^{n+1} = C(i\mu\Delta t)w^n. \tag{11.5.6}$$

If the modulus of the amplification factor is larger than one, the energy increases in each time step. This is called *amplification*. Conversely, if the amplification factor is smaller than one the energy decreases. This is called *dissipation*.

Example 11.5.1 *The modulus of the amplification factor of Euler's method is*

$$|C(i\mu\Delta t)| = \sqrt{1 + (\mu\Delta t)^2}. \tag{11.5.7}$$

So the amplification of the method is $O(\mu^2\Delta t^2)$ accurate.

The phase shift per time step of a numerical method is defined by the argument of the amplification factor, i.e.

$$\Delta\Phi = \arg(C(i\mu\Delta t)) = \arctan\left(\frac{\mathrm{Im}\{C\}}{\mathrm{Re}\{C\}}\right). \tag{11.5.8}$$

Example 11.5.2 *The phase shift of the improved Euler method is given by*

$$\Delta\Phi = \arctan\left(\frac{\mu\Delta t}{1 - \frac{1}{2}(\mu\Delta t)^2}\right). \tag{11.5.9}$$

The phase error or *dispersion* is the difference between the exact and numerical phase shifts. This is referred to as *dispersion* because the phase shifts differ for the different values of μ_k in equation (11.3.1).

Exercise 11.5.2 *Show that the phase error of the improved Euler method per time step is* $O((\mu\Delta t)^3)$. □

The *total dissipation*, $D_n(i\mu\Delta t)$, is the product of the dissipations of all the time steps from t_0 up to the end time T. The *total dispersion*, $\Delta\Phi_n(i\mu\Delta t)$, is the sum over the phase errors of all the time steps. Note that we have $n\Delta t = T - t_0$. The total dissipation and the total dispersion are measures of the error in the numerical solution. As $\Delta t \to 0$ the total dissipation should tend to 1 and the total dispersion should tend to 0.

Exercise 11.5.3 *Why do we need*

$$\lim_{\Delta t \to 0} D_n(i\mu\Delta t) = 1? \tag{11.5.10}$$

□

As an illustration we calculate the total dissipation and total dispersion for the forward Euler method:

$$D_n = (C(i\mu\Delta t))^n = (1 + (\mu\Delta t)^2)^{\frac{T-t_0}{2\Delta t}}. \tag{11.5.11}$$

From a Taylor series of the exponential, we see that

$$1 \leq D_n \leq \left(\mathrm{e}^{(\mu\Delta t)^2}\right)^{\frac{T-t_0}{2\Delta t}}. \tag{11.5.12}$$

Subsequently, from a linearization of the exponential, we get

$$\mathrm{e}^{(\mu\Delta t)^2 \frac{T-t_0}{2\Delta t}} = 1 + O(\mu^2\Delta t). \tag{11.5.13}$$

So the condition $\lim_{\Delta t \to 0} D_n(i\mu\Delta t) = 1$ is satisfied. For the total dispersion we have

$$\begin{aligned} &\Delta\Phi_n(i\mu\Delta t) = n(\mu\Delta t - \Delta\Phi) = n(\mu\Delta t - \arctan(\mu\Delta t)) = \\ &n(\mu\Delta t - (\mu\Delta t + O((\mu\Delta t)^3))) = nO((\mu\Delta t)^3)) = O(\mu^3\Delta t^2). \end{aligned} \tag{11.5.14}$$

Note that $n\Delta t = T - t_0$ and that the exact phase shift is $\mu\Delta t$. This has been used in this expression. It is clear from the expression that the total dispersion tends to zero as the time step tends to zero. In Figures 11.1 and 11.2 the total dissipation and dispersion are plotted as a function of the time-step Δt.

This total dispersion and total dissipation can be investigated for other time integration methods as well. We leave this as an exercise to the reader.

11.6 Direct time integration of the second order system

In principle it is not necessary to write equation (11.3.1) as a system of two first order differential equations. A lot of methods are available to integrate a second order differential equation of the form

$$\frac{d^2\mathbf{y}}{dt^2} = f(\mathbf{y}, t) \tag{11.6.1}$$

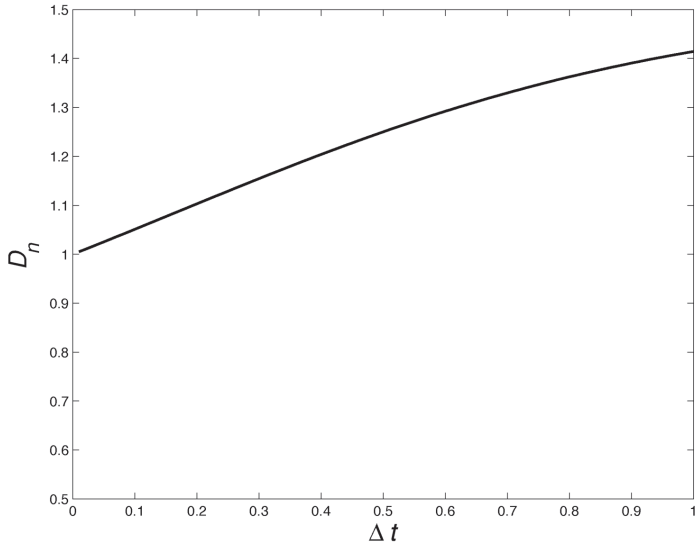


Figure 11.1: Dissipation of the forward Euler method for $\mu = 1$ and $T - t_0 = 1$.

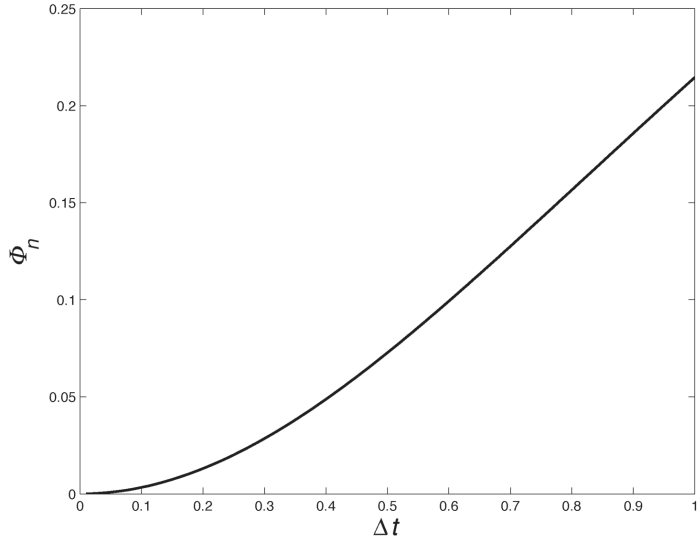


Figure 11.2: Dispersion of the forward Euler method for $\mu = 1$ and $T - t_0 = 1$.

directly. For a comprehensive survey of numerical methods to solve this system of second order differential equations, we refer to [23]. In this course we will treat two example schemes, applied to (11.2.1):

1. Explicitly
$$M\mathbf{u}^{n+1} - 2M\mathbf{u}^n + M\mathbf{u}^{n-1} = \Delta t^2 \left(c^2 S\mathbf{u}^n + \mathbf{f}^n \right). \tag{11.6.2}$$

2. Implicitly
$$\begin{aligned} M\mathbf{u}^{n+1} - 2M\mathbf{u}^n + M\mathbf{u}^{n-1} = \\ \frac{\Delta t^2}{4} \left(c^2 (S\mathbf{u}^{n+1} + 2S\mathbf{u}^n + S\mathbf{u}^{n-1}) + \mathbf{f}^{n+1} + 2\mathbf{f}^n + \mathbf{f}^{n-1} \right). \end{aligned} \tag{11.6.3}$$

Both methods are consistent of $O(\Delta t^2)$ in time. These methods are referred to as *three step* schemes, which implies that before one starts using these schemes, one first has to use an other method, such as Euler explicit:

$$\mathbf{u}_1 = \mathbf{u}_0 + \Delta t \mathbf{v}_0. \tag{11.6.4}$$

Using the explicit Euler method for the first step is satisfactory, since its error for the first step is $O(\Delta t^2)$.

The Equations (11.6.2) and (11.6.3) are special cases of the popular Newmark-(α, β) scheme. This scheme is usually written in a three-level form based on displacement $\mathbf{u}$, velocity $\mathbf{v}$ and acceleration $\mathbf{a}$. It uses a Taylor expansion, where the higher order terms are averaged.
Newmark reads:

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \mathbf{v}^n + \frac{\Delta t^2}{2}((1-2\beta)\mathbf{a}^n + 2\beta \mathbf{a}^{n+1}), \tag{11.6.5}$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \Delta t((1-\gamma)\mathbf{a}^n + \gamma \mathbf{a}^{n+1}), \tag{11.6.6}$$

$$M\mathbf{a}^{n+1} + c^2 S\mathbf{u}^{n+1} = \mathbf{f}^{n+1}. \tag{11.6.7}$$

At $t = t_0$ we solve $\mathbf{a}^0$ from the equation of motion (11.6.7). In the following steps we substitute (11.6.5) in (11.6.7) to get an equation for $\mathbf{a}^{n+1}$. Finally (11.6.5) and (11.6.6) are used to compute $\mathbf{u}^{n+1}$ and $\mathbf{v}^{n+1}$.
It is possible to rewrite Newmark as a three-step scheme:

$$\begin{aligned} (M + \beta c^2 \Delta t^2 S)\mathbf{u}^{n+1} - (2M - (\frac{1}{2} + \gamma - 2\beta)c^2 \Delta t^2 S)\mathbf{u}^n + \\ (M + (\frac{1}{2} - \gamma + \beta)c^2 \Delta t^2 S)\mathbf{u}^{n-1} = \Delta t^2 \mathbf{F}, \end{aligned} \tag{11.6.8}$$

with

$$\mathbf{F} = (\frac{1}{2} - \gamma + \beta)\mathbf{f}^{n-1} + (\frac{1}{2} + \gamma - 2\beta)\mathbf{f}^n + \beta \mathbf{f}^{n+1}. \tag{11.6.9}$$

Remark
(11.6.7) can not be used to compute $\mathbf{a}$ at boundaries with prescribed displacements at $t = t_0$. Why not? In practice one often takes $\mathbf{a} = 0$ in that case.
An alternative is to use a Taylor series expansion at $t = t_0 + \Delta t$ and to express $\mathbf{a}^0$ in $\mathbf{u}^0$, $\mathbf{v}^0$, and $\mathbf{u}^1$ at that boundary.

Exercise 11.6.1 *Prove that (11.6.8), (11.6.9) follows from (11.6.5)-(11.6.7).*
Hint: Substitute (11.6.6) in (11.6.5) to eliminate $\mathbf{v}^n$*, and write the equation for the previous timestep to get an expression of the form:*

$$\mathbf{u}^{n-1} = \mathbf{u}^n - \mathbf{v}^n \Delta t + \frac{\Delta t^2}{2}((1 - 2(\gamma - \beta))\mathbf{a}^{n-1} + 2(\gamma - \beta)\mathbf{a}^n). \tag{11.6.10}$$

Add (11.6.5) and (11.6.10) to get an expression for $\mathbf{u}^{n+1}$ *and* $\mathbf{a}^{n+1}$.
Then use the equation of motion (11.6.7). □

Exercise 11.6.2 *Show that the Newmark scheme reduces to the explicit central difference scheme ((11.6.2)) if* $\beta = 0$ *and* $\gamma = \frac{1}{2}$. □

Exercise 11.6.3 *Show that the Newmark scheme reduces to the implicit central difference scheme ((11.6.2)) if* $\beta = \frac{1}{4}$ *and* $\gamma = \frac{1}{2}$. □

Exercise 11.6.4 *Show that the three step implicit scheme (11.6.3) is identical to Crank-Nicholson's method for* (11.3.1)*. (Hint: write out the steps for* n *and* $n + 1$ *and eliminate all the* $\mathbf{v}$*'s.) Note that the first step of the three step method should be taken with Crank-Nicholson's method instead of the previously mentioned Euler explicit method.* □

11.7 The CFL criterion

From the section about the numerical time integration, it is clear that the time step plays an important role in the numerical integration. In general the time step Δt and step size Δx cannot be chosen independently. This was already observed for Euler's method. In 1928 Courant, Friedrichs and Lewy formulated a condition for the time step for the numerical solution to be a representation of the exact solution. Their condition was obtained by using a physical argument. Commonly one refers to it as the CFL criterion. Often this CFL condition is used in relation with stability of a numerical method. Strictly, this is not true since the CFL criterion represents a condition for convergence. In the following text an intuitive justification of the CFL criterion will be given. It is possible though to derive the CFL criterion in full mathematical rigor.

The solution of the wave equation can be represented by a superposition of linear waves, which all have a velocity c. Consider the solution at any node x_i at time t^j, then, within a time interval Δt, this *point source* influences the solution within the distance $c\Delta t$ from position x_i. Within a time interval Δt, the solution at locations with distance larger than $c\Delta t$ from x_i is not influenced by the solution at x_i on t_j. Usually this is referred to as the *region of influence* of $u(x_i, t^j)$. Vice versa, $u(x_i, t^j)$ is determined by the *point sources* of $u(x, t^{j+1} - \tau)$, with $|x - x_i| < c\tau$ for $\tau < \Delta t$. This region of influence is indicated by the dashed part in Figure 11.3. For the explicit time integration of the wave equation, the spatial discretization is done at time t^j. For the finite differences solution with one spatial coordinate at x_i on t^j, one uses $u(x_i, t^j)$, $u(x_{i-1}, t^j)$ and $u(x_{i+1}, t^j)$, i.e.

$$\left.\frac{d^2u}{dx^2}\right|_{t=t^j} = \frac{u(x_{i-1}, t^j) - 2u(x_i, t^j) + u(x_{i+1}, t^j)}{\Delta x^2}. \tag{11.7.1}$$

The CFL criterion of an explicit scheme for the wave equation is as follows: *The region of influence of* $u(x, t^{j+1})$ *with* $|x - x_i| < c\tau$ *for* $\tau < \Delta t$ *(hence around* x_i*), may not contain any locations at* t^j *outside the interval of the grid nodes* $(x_{i-1}, x_i, x_{i+1}$ *here), which are used for the discretization of the second order partial derivatives of the wave equation.*

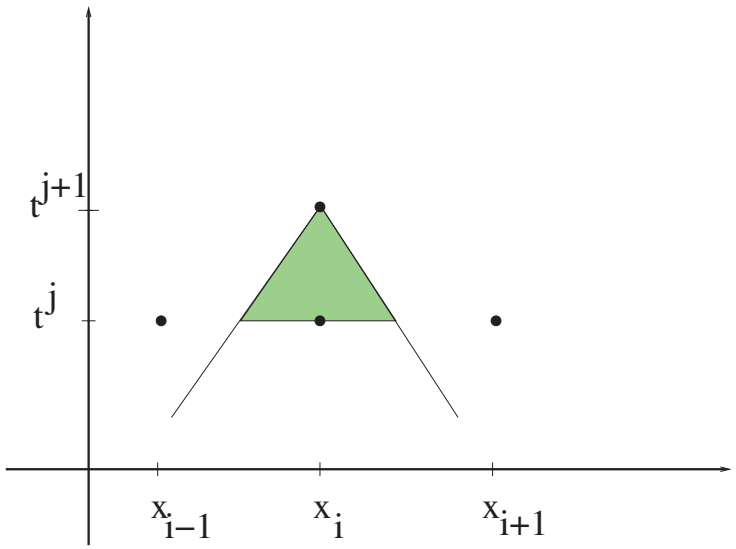


Figure 11.3: The solution at x_i on t^{j+1} is determined by u(x,$t^{j+1} - \tau$) where x $\in$ (x_j − cτ,x_j + cτ) for $\tau < \Delta$t. The region of influence is indicated by the grey region. This situation satisfies the CFL criterion.

The CFL criterion guarantees that the numerical solution is determined only by all the point sources that physically have an influence on this solution. In the case of Figure 11.3, it turns out that the region of influence, for $\tau < \Delta t$, only contains locations within the interval (x_{i-1}, x_{i+1}) and hence for this Δt the CFL criterion is satisfied and convergence of the numerical solution is to be expected. Before an example is treated, the following important aspects should be noted:

Remarks

1. For a *three step* scheme it is sufficient to check the CFL criterion for the final two steps. By induction it follows that the criterion is satisfied for all steps.
2. For an implicit scheme the CFL criterion is irrelevant. Since, then, the entire previous time step determines the solution on the present time step.

An example of the derivation of the CFL criterion is treated:

Example 11.7.1 *Consider the explicit time integration of the wave equation in one dimension with equidistant nodes:*

$$u_i^{n+1} - 2u_i^n + u_i^{n-1} = \left(\frac{c\Delta t}{\Delta x}\right)^2 (u_{i+1}^n - 2u_i^n - u_{i-1}^n). \tag{11.7.2}$$

The region of determination of u_i^{n+1} is then given by the interval $(x_i - c\tau, x_i + c\tau)$ for $\tau < \Delta t$. The nodes for the spatial discretization are x_{i-1}, x_i and x_{i+1}. The interval that is defined by these nodes (hence (x_{i-1}, x_{i+1})) must contain the region of influence $(x_i - c\tau, x_i + c\tau)$ for $\tau < \Delta t$. Hence, the CFL criterion for this case is given by $x_i - c\tau > x_{i-1}$ and $x_i + c\tau < x_{i+1}$ for $\tau < \Delta t$. Since the nodes are equidistant and $\Delta x = x_{i+1} - x_i$, this implies the following CFL-criterion:

$$\frac{c\Delta t}{\Delta x} \leq 1. \tag{11.7.3}$$

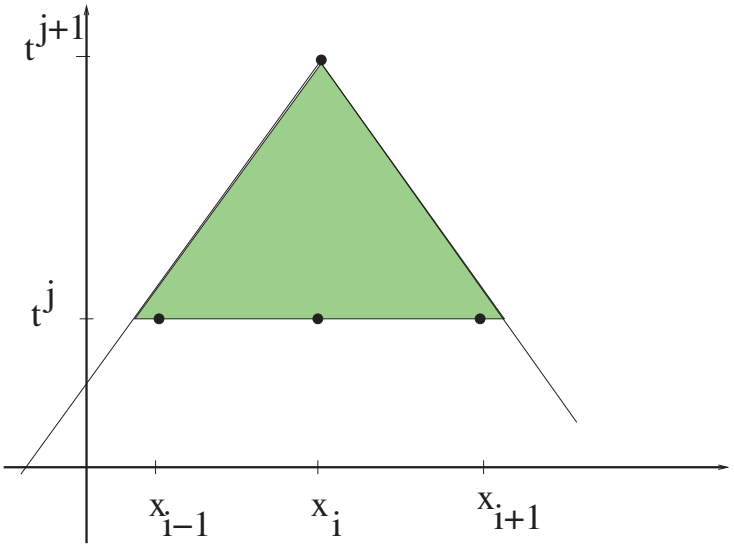


Figure 11.4: The solution at x_i on t^{j+1} is determined by u(x,$t^{j+1}-\tau$) where x $\in$ (x_j $-$ cτ,x_j + cτ) for $\tau < \Delta$t. The region of influence is indicated by the grey region and some part of the region of influence is outside the interval (x_{i-1},x_{i+1}). Hence this situation violates the CFL criterion.

An example of a region of influence for a time step that does not satisfy the CFL criterion is shown in Figure 11.4.

Exercise 11.7.1 *Check that for the wave equation with one spatial coordinate, the Euler forward method by*

$$\begin{aligned} u_i^{n+1} &= u_i^n + \Delta t v_i^n \\ v_i^{n+1} &= v_i^n + \frac{c^2 \Delta t}{\Delta x^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n), \end{aligned} \tag{11.7.4}$$

cannot satisfy the CFL criterion. If the first equation is replaced with

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{4}(v_{i-1}^n + 2v_i^n + v_{i+1}^n), \tag{11.7.5}$$

there is a CFL criterion. Give the CFL criterion for this case. □

11.8 Summary of Chapter 11

This chapter has dealt with numerical methods for the solution of the (hyperbolic) wave equation. The hyperbolic nature of the wave equation is important for the nature of the numerical solutions. To solve the PDE the method of lines has been used. It first deals with the spatial derivatives and considers time integration of the resulting system of ODE's as a separate problem.

A direct time integration scheme for the second derivative of the time has also been presented. The numerical amplification factor for the dissipation and the phase shift of the numerical solution have been defined and analyzed. Finally,

the derivation of the CFL-criterion, using the concept of the region of influence in the x, t plane, has been given. This CFL criterion is necessary for the numerical solution to be a representation of the exact solution.

Chapter 12

The transport equation

Objectives

The transport equation is fundamental in modeling (multi-phase) flow in porous media, such as underground oil, gas and water reservoirs. Some engineering disciplines, where the transport equation plays an important role as well, are geosciences, aerospace engineering and naval engineering. The transport equation is also referred to as a first order hyperbolic conservation law and an important application in aerospace engineering involves modeling of air flow around aircraft. The backbone for understanding the nature of the solutions and boundary conditions of the transport equation lies in the analysis of the characteristics of the solutions. Further, some classical numerical methods to solve hyperbolic conservation laws will be presented. The presentation is given for configurations with one spatial coordinate only. Finally, some mathematical theory for the transport equation will be presented. *Traveling wave solutions* for Burgers equation are analyzed and subsequently the nature of the solutions of the Buckley-Leverett equation is discussed.

12.1 Introduction

The transport equation describes transport of one or various components in n dimensions. In this chapter we limit ourselves to transport in one spatial dimension. The most general form of a transport equation in *conservative form* is:

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = \mathbf{g}(\mathbf{u}, x, t). \tag{12.1.1}$$

with $\mathbf{u} = (u_1, \ldots, u_m)^T$ denoting the vector with the transported quantities and $\mathbf{f} = (f_1, \ldots, f_m)^T$ the *flux-vector*. If the vector $\mathbf{g}$ does not depend on x and t, then the problem is called *autonomous*. In many transport problems the right-hand side involves a chemical reaction, whose rate often only depends on the solution. Hence, many transport problems are autonomous.

In the literature the transport problem is often referred to as hyperbolic, e.g. a hyperbolic conservation law. Although the equation certainly does not satisfy the standards for hyperbolicity as in Chapter 2, this classification does make sense, since the solutions of Equation (12.1.1) can be represented by waves, just like those of genuine hyperbolic partial differential equations. This is justified in the following exercise:

Exercise 12.1.1 *Show that the transport equation of two components*

$$\begin{cases} \dfrac{\partial u}{\partial t} + c\dfrac{\partial v}{\partial x} = 0 \\ \dfrac{\partial v}{\partial t} + c\dfrac{\partial u}{\partial x} = 0 \end{cases} \tag{12.1.2}$$

is equivalent to the wave equation $\dfrac{\partial^2 u}{\partial t^2} = c^2 \dfrac{\partial^2 u}{\partial x^2}$.

In non-conservative form, Equation (12.1.1) has the following shape:

$$\frac{\partial \mathbf{u}}{\partial t} + A(\mathbf{u})\frac{\partial \mathbf{u}}{\partial x} = \mathbf{g}, \tag{12.1.3}$$

with $a_{ij} = \frac{\partial f_i}{\partial u_j}$. Hence A is the Jacobian of the flux-vector. For regular transport A must have real eigenvalues. In the literature 12.1.1 and 12.1.3 are commonly referred to as hyperbolic if the Jacobian A has real eigenvalues (i.e. regular transport). Imposing initial and boundary conditions for (12.1.1) and (12.1.3) is usually not trivial. This will be illustrated by use of the characteristics for the transport of one component.

12.2 Characteristics

Let us consider the equation

$$\frac{\partial u}{\partial t} + a(u)\frac{\partial u}{\partial x} = g(u), \tag{12.2.1}$$

which describes transport of one component in non-conservative form. We consider a curve in the (x,t) plane, parameterized by s with the property

$$\frac{dx}{ds} = \rho(s)a(u) \text{ and } \frac{dt}{ds} = \rho(s), \tag{12.2.2}$$

then along this curve we obtain from the total derivative of u with respect to s:

$$\frac{du}{ds} = \frac{\partial u}{\partial x}\frac{dx}{ds} + \frac{\partial u}{\partial t}\frac{dt}{ds} = \rho(s)g(u). \tag{12.2.3}$$

This means that if the value of u is known at a certain point, i.e. $u(x_0,t_0) = u_0$, then, the value over a curve is defined for $(x(s),t(s),u(x(s),t(s))$, is the solution of the coupled system of ordinary differential equations:

$$\frac{dx}{ds} = \rho(s)a(u), \quad \frac{dt}{ds} = \rho(s), \quad \frac{du}{ds} = \rho(s)g(u). \tag{12.2.4}$$

with initial conditions $x(0) = x_0$, $t(0) = t_0$ and $u(0) = u_0$. The (x,t) curve of Equation (12.2.2) is referred to as a *characteristic*, the system (12.2.2) is called the characteristic equation and Equation (12.2.3) is referred to as the characteristic relation. The Equations (12.2.2) and (12.2.3) give the solution of the partial differential equation in the form of a system of coupled ordinary differential equations. One also expresses this property as follows: along the characteristics the PDE changes into an ODE. Note that if $g = 0$, then the solution u is constant along a characteristic and the quantity u is transported along the characteristics. The choice of ρ is arbitrary, it influences the parameterization and not the solution itself. If g does not depend on u then the characteristics can be determined by just solving Equation (12.2.2). However, for cases in which a depends on u, the complete Equation (12.2.4) must be solved. Then, one obtains the characteristic and the solution at the same time.

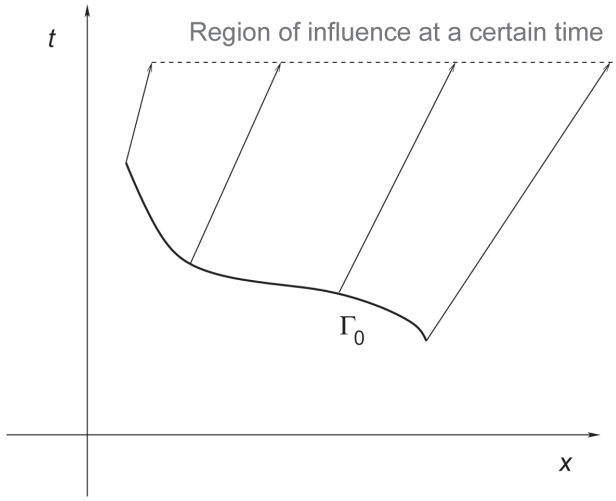


Figure 12.1: The region of influence of Γ_0 at a certain time. This region is indicated by the dashed line. The arrows indicate the characteristics of the solution.

Exercise 12.2.1 *Show that the characteristic equation corresponding to the PDE*

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0. \tag{12.2.5}$$

is given by

$$\frac{dx}{ds} = \rho c \text{ and } \frac{dt}{ds} = \rho. \tag{12.2.6}$$

Show that this gives $\dfrac{dx}{dt} = c$. *Hence all lines of the form* $x - ct =$ *constant are characteristics. Suppose that* u *is given at* $t = 0$ *on the interval* $(0,1)$, *show that then the solution at time* t *is determined on the interval* $x \in (ct, 1+ct)$ *and show that it is given by* $u(x,t) = u_0(x-ct)$.

We formulate the initial value problem as follows:

Let Γ_0 *be a curve in the* (x,t)*-plane, such that each characteristic intersects* Γ_0 *only once. Let* u *be given on* Γ_0. *Then, the solution is determined on a strip* Σ, *which is constructed by the union of all the characteristics that intersect* Γ_0. *The solution on each characteristic is determined by the system of ordinary differential equations* (12.2.4) *with as the initial condition the values of the solution at* Γ_0. *The situation has been pictured in Figure 12.1.*

The strip Σ is called the *region of influence* of Γ_0.

Exercise 12.2.2 *Why is a characteristic not allowed to intersect* Γ_0 *twice for a general initial condition on* Γ_0? *What condition should be satisfied if the characteristic intersects the curve* Γ_0 *twice?*

Exercise 12.2.3 *Given the differential equation*

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = -u, \tag{12.2.7}$$

with initial condition $u(x,0) = u_0(x)$ *on the interval* $0 \le x \le 1$.

1. *What is the equation for the characteristics?*
2. *What is the characteristic relation?*
3. *What is the region of influence of the interval* $0 \leq x \leq 1$*?*

Exercise 12.2.4 *Given the differential equation*

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 1, \tag{12.2.8}$$

with initial conditions on $\Gamma_0 = \Gamma_1 \cup \Gamma_2$ *with*

1. $\Gamma_1 = \{(x,t) : t = 0, 0 \leq x \leq 1\}$
   $\Gamma_2 = \{(x,t) : x = 0, 0 \leq t < \infty\}$.
2. $\Gamma_1 = \{(x,t) : t = 0, 0 \leq x \leq 1\}$
   $\Gamma_2 = \{(x,t) : x = 1, 0 \leq t < \infty\}$.

In which of these two cases is the problem well-posed for a general initial condition? What is the region of influence of u?

Exercise 12.2.5 *Give for the differential equation*

$$\frac{\partial u}{\partial t} + t\frac{\partial u}{\partial x} = f, \tag{12.2.9}$$

the region of influence of the start curve: $\Gamma_0 = \{(x,t) : -1 \leq t \leq 1, x = 0\}$. *Are all choices for u allowed on the curve* Γ_0*?*

In case of smooth solutions of u with respect to x and t, it is necessary that two characteristics, which originate from different locations on Γ_0 with different initial values, do not intersect.

12.3 Some classical numerical procedures

In the past many numerical methods to solve the transport equation were based on the characteristics of the solution. However, the popularity of these methods decreased and therefore they are not treated in this book. These methods were gradually replaced by the *fixed grid* methods. In this section first the classical methods of central and upwind discretization are analyzed. Subsequently, the Lax-Wendroff scheme for smooth solutions and the use of fluxlimiters for discontinuous solutions are introduced as higher order methods to solve the transport equation.

12.3.1 Central discretization and upwind discretization

We consider again the transport equation

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \quad \Longleftrightarrow \quad \frac{\partial u}{\partial t} = -\frac{\partial f(u)}{\partial x}. \tag{12.3.1}$$

In this text an equidistant distribution of the gridnodes is considered.
We use a Finite Volume Method, hence we integrate over a control volume (see Figure 12.2), which gives on time-step t_j:

$$\int_{\Omega_i} \frac{\partial u}{\partial t}(x,t_j)dx = -\int_{\Omega_i} \frac{\partial f(u(x,t_j))}{\partial x}dx =$$

$$= -\left[f(u(x_{i+\frac{1}{2}},t_j)) - f(u(x_{i-\frac{1}{2}},t_j))\right].$$

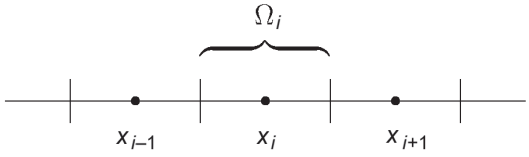


Figure 12.2: A gridcell used for the Finite Volume Discretization.

We define $f^j_{i-\frac{1}{2}} := f(u(x_{i-\frac{1}{2}}, t_j))$ and $f^j_{i+\frac{1}{2}} := f(u(x_{i+\frac{1}{2}}, t_j))$, as the flux on both boundaries of the gridcell Ω_i at time step $j\Delta t$. The flux entering Ω_{i+1} and the flux leaving Ω_i are balanced with the accumulation in Ω_i. The integral on the left-hand side of the above equation is approximated as follows:

$$\int_{\Omega_i} \frac{\partial u}{\partial t}(x, t_j)dx \approx \frac{\partial u}{\partial t}(x_i, t_j)\Delta x \approx \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\Delta t}\Delta x.$$

Define $u_i^{j+1} = u(x_i, t_{j+1})$ and $u_i^j = u(x_i, t_j)$, in this way the following discretization of (12.3.1) is obtained:

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} + \frac{f^j_{i+\frac{1}{2}} - f^j_{i-\frac{1}{2}}}{\Delta x} = 0. \tag{12.3.2}$$

We analyze difference equation (12.3.2). We consider the simple example $f(u) = u$. We discretize the above equation by the use of the second order *central discretization*. The fluxes on the boundaries of Ω_i are approximated

$$f^j_{i+\frac{1}{2}} := f(u^j_{i+\frac{1}{2}}) = u^j_{i+\frac{1}{2}} \approx \frac{u^j_{i+1} + u^j_i}{2},$$

$$f^j_{i-\frac{1}{2}} := f(u^j_{i-\frac{1}{2}}) = u^j_{i-\frac{1}{2}} \approx \frac{u^j_{i-1} + u^j_i}{2}.$$

With these approximations, a central discretization results from Equation (12.3.2):

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} + \frac{u^j_{i+1} - u^j_{i-1}}{2\Delta x} = 0, \tag{12.3.3}$$

In Chapter 3 it is derived that the truncation error in the spatial discretization is of second order. It turns out that the above discretization is prone to unphysical oscillations. For the stationary convection-diffusion equation this is motivated in Chapter 3. This will be analyzed later in this section by the use of the Von Neumann stability analysis.

Due to this oscillatory behavior it is preferable to use an alternative method. We derive this discretization method by the use of characteristics. The points (x_{i-1}, t^j), (x_i, t^j) and (x_i, t^{j+1}) in the (x,t)-plane are sketched in Figure 12.3. Over each characteristic the value of u is constant. Since, $\frac{dx}{dt} = 1$, the characteristics are parallel to the line $x = t$ in the (x,t)-plane. So following the characteristic through (x_i, t^{j+1}) we end up at the point $(x_i - \Delta t, t^j)$ at the line $t = t^j$ (see Figure 12.3). Hence, we have $u_i^{j+1} = u^j(x_i - \Delta t)$. Since $u^j(x_i - \Delta t)$ is not a value on a node, its value is

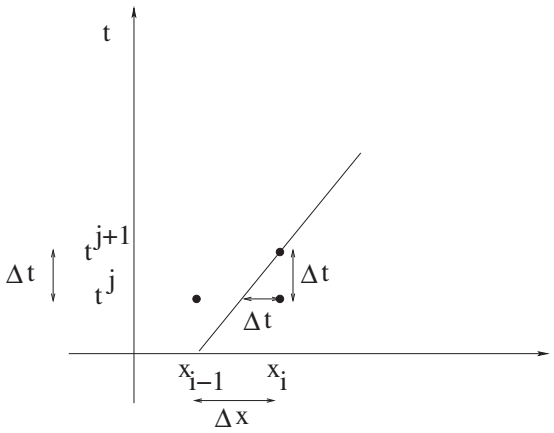


Figure 12.3: Derivation of the first order upwind discretization.

computed by linear interpolation between the values u_{i-1}^j and u_i^j, to obtain

$$u^j(x_i - \Delta t) = u_{i-1}^j \frac{\Delta t}{\Delta x} + u_i^j \frac{\Delta x - \Delta t}{\Delta x} = u_i^j + \frac{\Delta t}{\Delta x}(u_{i-1}^j - u_i^j). \tag{12.3.4}$$

Keeping in mind that $u_i^{j+1} = u^j(x_i - \Delta t)$, the above equation is written as

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} + \frac{u_i^j - u_{i-1}^j}{\Delta x} = 0. \tag{12.3.5}$$

The above equation is commonly referred to as the first order *upwind* discretization.

In the coming text the accuracy and stability issues are investigated for these two discretization for the linear transport equation, where $f(u) = u$.

The Taylor Series around $x = x_i$ and $t = t^j$ for u_i^{j+1} and u_{i-1}^j are given by

$$\begin{aligned} u_{i-1}^j &= u_i^j - \Delta x \cdot \frac{\partial u}{\partial x} + \frac{(\Delta x)^2}{2}\frac{\partial^2 u}{\partial x^2} + \cdots, \\ u_i^{j+1} &= u_i^j + \Delta t \cdot \frac{\partial u}{\partial t} + \frac{(\Delta t)^2}{2}\frac{\partial^2 u}{\partial t^2} + \cdots. \end{aligned}$$

Substitution of the above equations into Equation (12.3.5) gives

$$\frac{u_i^j + \Delta t \cdot \frac{\partial u}{\partial t} + \frac{(\Delta t)^2}{2}\frac{\partial^2 u}{\partial t^2} + \cdots - u_i^j}{\Delta t} +$$

$$+ \frac{u_i^j - u_i^j + \Delta x \cdot \frac{\partial u}{\partial x} - \frac{(\Delta x)^2}{2}\frac{\partial^2 u}{\partial x^2} + \cdots}{\Delta x} = 0.$$

Then, after neglecting higher order terms one obtains:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \frac{\Delta x}{2}\frac{\partial^2 u}{\partial x^2} - \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}. \tag{12.3.6}$$

We remark that $\frac{\partial u}{\partial t} = -\frac{\partial u}{\partial x} \Longrightarrow \frac{\partial^2 u}{\partial t^2} = -\frac{\partial^2 u}{\partial t \partial x} = -\frac{\partial^2 u}{\partial x \partial t} = -\frac{\partial}{\partial x}(-\frac{\partial u}{\partial x}) = \frac{\partial^2 u}{\partial x^2}$ (provided the second order partial derivatives are continuous). Substitution into (12.3.6)

gives the following equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \left(\frac{\Delta x}{2} - \frac{\Delta t}{2}\right)\frac{\partial^2 u}{\partial x^2} = \mathbb{D}\frac{\partial^2 u}{\partial x^2}, \tag{12.3.7}$$

with $\mathbb{D} = \dfrac{\Delta x}{2} - \dfrac{\Delta t}{2}$. This equation is a convection-diffusion equation. From the above equation it is clear that the discretization error for this upwind discretization is first order for the time step and the grid spacing. Therefore, this formula is referred to as first order upwind. The dispersion term, $\mathbb{D}\dfrac{\partial^2 u}{\partial x^2}$, is called the numerical diffusion. We know that the convection-diffusion equation has a stable solution if and only if $\mathbb{D} \geq 0$. So stability is guaranteed if

$$0 \leq \mathbb{D} = \frac{\Delta x}{2} - \frac{\Delta t}{2} \quad \Longleftrightarrow \quad \frac{\Delta t}{\Delta x} \leq 1. \tag{12.3.8}$$

Inequality (12.3.8) is commonly called the *CFL-condition* after Courant-Friedrichs-Lewy. The values of Δt and Δx have to satisfy the CFL-condition. Note that if $\mathbb{D} = 0$ (or $\dfrac{\Delta t}{\Delta x} = 1$), then Equation (12.3.7) reduces to (12.3.1) with $f(u) = u$. For this case there is no numerical diffusion. Then, the error consists of higher order terms only. In most practical situations with variable coefficients or non-linearities, it is impossible to tune Δt and Δx such that $\mathbb{D} = 0$.

Note that we use an explicit time-integration, which has a time-step restriction for stability. For an implicit time integration one obtains:

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = -\frac{u_i^{j+1} - u_{i-1}^{j+1}}{\Delta x}.$$

Using a similar procedure with Taylor-expansion as for the explicit scheme, one obtains

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \frac{1}{2}(\Delta t + \Delta x)\frac{\partial^2 u}{\partial x^2},$$

hence $\mathbb{D} := \frac{1}{2}(\Delta t + \Delta x) > 0$ for all $\Delta t, \Delta x > 0$. Above equation always has a stable solution, however, errors due to space discretization and time integration do not tend to cancel each other. Therefore, an implicit time integration method is not widely used.

12.3.1.1 Von Neumann stability analysis

Next, the issue of stability is treated. In the present section the method of Von Neumann is used to analyze stability. This method is based on the estimation of the eigenvalues of the discretization matrix. The procedure can be used for PDE's with constant coefficients and equidistant grids only. The method involves the use of a discrete Fourier series and is formally applicable to rectangular geometries with periodical boundary conditions. In this section only one spatial co-ordinate is considered. Let us consider the function $\hat{u}$ and the domain $x \in [0, 1]$, then, a Fourier series of $\hat{u}$ is given by

$$\hat{u}(x, t) = \sum_{\alpha=1}^{\infty} \rho_\alpha(t) \mathrm{e}^{-2\pi\alpha x i}. \tag{12.3.9}$$

The functions $\rho_\alpha(t)$ are referred to as Fourier coefficients. Let N denote the number of grid nodes. Then, the above relation is written for the function $\hat{u}$ on the grid, i.e.

$\hat{u}_k^n = \hat{u}(k\Delta x, n\Delta t)$:

$$\hat{u}_k^n = \sum_{\alpha=1}^{N-1} \rho_\alpha^n \mathrm{e}^{-2\pi\alpha k\Delta x i}, \tag{12.3.10}$$

where we define $\rho_\alpha^n = \rho_\alpha(n\Delta t)$. The above relation can represent a Discrete Fourier Transform of the perturbed solution. For stability we require that ρ_α^n stays bounded as $n \to \infty$. Consider the following example

$$\frac{u_k^{n+1} - u_k^n}{\Delta t} + \frac{u_{k+1}^n - u_{k-1}^n}{2\Delta x} = 0, \tag{12.3.11}$$

i.e. a central discretization with Euler forward (explicit) time integration. Then, substitution of (12.3.10) into (12.3.11) yields

$$\sum_{\alpha=1}^{N-1} (\rho_\alpha^{n+1} - \rho_\alpha^n)\mathrm{e}^{-2\pi\alpha k\Delta x i} = \frac{\Delta t}{2\Delta x} \sum_{\alpha=1}^{N-1} \rho_\alpha^n \left(\mathrm{e}^{-2\pi\alpha(k-1)\Delta x i} - \mathrm{e}^{-2\pi\alpha(k+1)\Delta x i}\right). \tag{12.3.12}$$

Collecting all terms for a fixed value of α, ρ_α^{n+1} and ρ_α^n, and division by $\mathrm{e}^{-2\pi\alpha k\Delta x i}$, gives

$$\rho_\alpha^{n+1} = \rho_\alpha^n \left[1 + \frac{\Delta t}{2\Delta x}\left(\mathrm{e}^{2\pi\alpha\Delta x i} - \mathrm{e}^{-2\pi\alpha\Delta x i}\right)\right]. \tag{12.3.13}$$

Using $\sin(\theta) = \dfrac{\mathrm{e}^{\theta i} - \mathrm{e}^{-\theta i}}{2i}$, gives

$$\rho_\alpha^{n+1} = \rho_\alpha^n \left[1 + i\frac{\Delta t}{\Delta x}\sin(2\pi\alpha\Delta x)\right]. \tag{12.3.14}$$

The ratio $\dfrac{\rho_\alpha^{n+1}}{\rho_\alpha^n}$ represents an amplification factor. A condition for stability is

$$\left|\frac{\rho_\alpha^{n+1}}{\rho_\alpha^n}\right| \leq 1, \tag{12.3.15}$$

i.e. the modulus of the ratio between the Fourier coefficients ρ_α^n at consecutive time-steps may not be larger than one. Note that the Fourier coefficients are not real, then, for the central explicit discretization follows

$$\left|\frac{\rho_\alpha^{n+1}}{\rho_\alpha^n}\right|^2 = 1 + \left(\frac{\Delta t}{\Delta x}\right)^2 \sin^2\left(\frac{2\pi\alpha}{N}\right) > 1, \tag{12.3.16}$$

and hence the central discretization with Euler-forward time integration is always unstable, regardless the value of Δt and Δx.

An other example with conditional stability is considered in the following exercise, where an explicit time-integration is considered for an upwind discretization scheme.

Exercise 12.3.1 *Consider the discretization method with Euler forward time integration and a first order upwind spatial discretization:*

$$\frac{u_k^{n+1} - u_k^n}{\Delta t} + \frac{u_k^n - u_{k-1}^n}{\Delta x} = 0. \tag{12.3.17}$$

Use the Von Neumann stability of this section to show that the above mentioned formula gives stability if $\dfrac{\Delta t}{\Delta x} < 1$.

Bear in mind that the Fourier coefficients are not real in general.

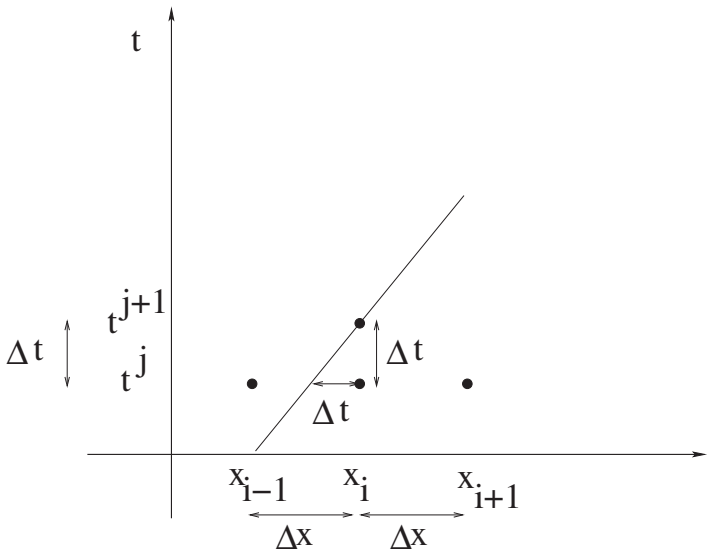


Figure 12.4: Derivation of the Lax-Wendroff scheme.

An alternative and more general method for the analysis of stability is the so-called matrix-method, where the eigenvalues of the discretization matrix, where the time integration is incorporated, are estimated by use of Gershgorin's Theorem. This has been treated in Chapter 10 for the heat equation.

12.3.1.2 The Lax-Wendroff scheme

When solutions are smooth the Lax Wendroff scheme is suitable for hyperbolic conservation Laws. We present the Lax-Wendroff scheme for the first order linear advection equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0. \tag{12.3.18}$$

To derive the Lax-Wendroff scheme for one spatial coordinate, characteristics are used. The procedure is similar to the derivation of the first-order upwind method. Consider the (x,t)-plane as in Figure 12.4, the points (x_{i-1}, t^j), (x_i, t^j), (x_{i+1}, t^j) and (x_i, t^{j+1}) are indicated. We consider the characteristic that passes through (x_i, t^{j+1}). Note that the characteristics are straight lines for which $\dfrac{dx}{dt} = 1$. Since the solution is constant over the characteristics, we have $u_i^{j+1} = u^j(x_i - \Delta t)$ (see Figure 12.4). Since $x_i - \Delta t$ generally does not co-coincide with a node, quadratic interpolation between u_{i-1}^j, u_i^j and u_{i+1}^j is used to approximate the value $u^j(x_i - \Delta t)$ (see Figure 12.4). By use of $u_i^{j+1} = u^j(x_i - \Delta t)$, the solution at the new time step is determined. Let the CFL-number be given by σ, i.e. $\sigma = \dfrac{\Delta t}{\Delta x}$, then, this procedure gives

$$u_i^{j+1} = \frac{\sigma}{2}(\sigma + 1)u_{i-1}^j - (\sigma^2 - 1)u_i^j + \frac{\sigma}{2}(\sigma - 1)u_{i+1}^j. \tag{12.3.19}$$

Exercise 12.3.2 *Use quadratic interpolation to approximate $u^j(x_i - \Delta t)$ using the values u_{i-1}^j, u_i^j and u_{i+1}^j. Finally, show that Equation* (12.3.19) *follows.*

This scheme due to Lax-Wendroff is more accurate than the previously treated first-order upwind and second-order central discretization methods. However, due to the higher order interpolation, it is suitable for hyperbolic PDE's with smooth solutions only. For shock solutions, it has been proved (see Leveque [25]) that spurious oscillations are introduced by this method and hence for these cases the Lax-Wendroff scheme is not popular. Then, one relies on alternative methods which are suitable for shock capturing. These schemes are based on flux limiters or slope limiters.

12.3.1.3 Flux limiters

From the preceding section it is clear that the upwind scheme causes numerical diffusion as an undesired side effect. The error is of the order of Δx. However, the solution is physical in the sense that no spurious oscillations are introduced. In this section we will consider some higher order methods. We point out that these higher order methods are useful in those parts of the domain of computation where the solution behaves smoothly. At positions where no smoothness is attained we will fall back on the first order upwind method. We turn back to the original first order hyperbolic transport equation with one spatial coordinate

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0. \tag{12.3.20}$$

For the sake of simplicity only equidistant grids for the computational domain are considered. Application of the Finite Volume method over the volume Ω_i around grid node i and division by Δx gives:

$$\int_{\Omega_i} \frac{\partial u}{\partial t} dx = f(u_{i-1/2}) - f(u_{i+1/2}). \tag{12.3.21}$$

Let w_i approximate the mean value of the solution within Ω_i, then we get

$$\frac{w_i^{j+1} - w_i^j}{\Delta t} = \frac{f_{i-1/2} - f_{i+1/2}}{\Delta x}. \tag{12.3.22}$$

For most cases one takes u linear or constant between two consecutive gridnodes and then this average over Ω_i equals the value of u at the particular gridnode, i.e. $w_i = u_i$. The first order upwind approximation gives

$$f_{i+1/2} = f(w_i), \quad f_{i-1/2} = f(w_{i-1}), \tag{12.3.23}$$

and a second order central scheme gives

$$\begin{aligned} f_{i+1/2} &= f(\frac{w_{i+1} + w_i)}{2}) = f(w_i + 1/2(w_{i+1} - w_i)) \\ f_{i-1/2} &= f(\frac{w_{i-1} + w_i}{2}) = f(w_{i-1} + 1/2(w_i - w_{i-1})) \end{aligned} \tag{12.3.24}$$

For the first order upwind scheme, it is known that the numerical solution exhibits no unphysical oscillations. However, it tends to smear out discontinuities due to numerical diffusion. A higher order scheme, such as Lax-Wendroff's scheme, is more accurate but initial conditions with discontinuities can develop into numerical solutions with spurious oscillations. Hence, near shocks one avoids the use of methods which are prone to unphysical oscillations and one does not insist on the higher order accuracy of the solution near discontinuities. Therefore, one tries

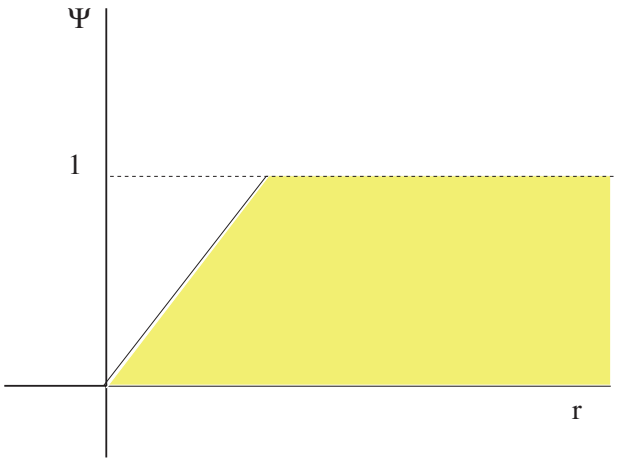


Figure 12.5: The conditions for ψ to avoid unphysical oscillation. This area is indicated by the grey region.

to combine the advantages of both methods: near discontinuities one uses a first order upwind discretization and away from a discontinuity, where the solution is smooth, one uses a higher order method. Van Leer proposes to generalize the relations of $f_{i+1/2}$ to

$$f_{i+1/2} = f(w_i + \psi(r_i)(w_{i+1} - w_i)). \tag{12.3.25}$$

The idea is to let ψ be dependent on the smoothness of the solution and therefore one uses the ratio $r_i := \frac{w_i - w_{i-1}}{w_{i+1} - w_i}$, which is defined by the ratio of the subsequent differences of values of w over neighboring gridnodes. The function ψ is commonly called a *limiter function*. Note that whenever $\psi = \frac{1}{2}$ for both $f_{i-1/2}$ and $f_{i+1/2}$, then, a central scheme is recovered. Whereas, if $\psi = 0$ for both $f_{i-1/2}$ and $f_{i+1/2}$, then, the first order upwind scheme is obtained. Further, if $\psi = 1$ for both $f_{i-1/2}$ and $f_{i+1/2}$, then, a *downwind* discretization is used. The limiter function ψ should be chosen such that first order upwind is obtained near shocks and that the order of discretization is maximal when the solution is smooth. Further, unphysical oscillations are not allowed. Using the concept of conservation of monotonicity and the decrease of total variation (see the appendix of this chapter or Leveque [25], sections 6.7 and 6.12 for the interested reader), one can show that

$$\begin{aligned} 0 \leq \psi(r_i) \leq r_i \\ 0 \leq \psi(r_i) \leq 1, \end{aligned} \tag{12.3.26}$$

provide sufficient conditions to avoid spurious oscillations. This regime is plotted in Figure 12.5 by the colored area.

To make the function ψ more look like a higher order method, a wide variety of functions for ψ has been proposed and investigated. Sweby [38] presents a comparison of the properties of the various choices for ψ. Not aiming at being complete we only mention the limiter due to Van Leer [25]

$$\psi(r) = \frac{r + |r|}{2(1 + |r|)}, \tag{12.3.27}$$

and the 'minmax' limiter due to Kooren [25]

$$\psi(r) = \max\left(0, \min\left(1, \tfrac{1}{3} + \tfrac{1}{6}r, \mu r\right)\right), \quad \text{(Kooren limiter function)}, \quad \mu > 0. \tag{12.3.28}$$

One sees immediately that the second limiter function satisfies all the requirements. It is shown in following exercise that the first limiter satisfies the desired properties as well. Therefore, it is commonly used.

Exercise 12.3.3 *Show that* $\psi(r) = 0$ *for all* $r \leq 0$, $\psi(1) = \frac{1}{2}$ *and that* $\lim_{r\to\infty} \psi(r) = 1$ *and* $\lim_{r\to 0^+} \psi'(r) = 1$. *Further show that* $\psi(r)$ *is monotonic.*

Note that when the profile is almost linear, then, $r \approx 1$. This implies that almost the central discretization is used. Further, when there is a shock between x_{i-1} and x_i, an upwind scheme is obtained. We remark that the situation where both $\psi(r_{i-1}) = 1 = \psi(r_i)$ does not occur for one dimensional geometries. This is nice since this particular case would reflect a *downwind* discretization. Many other limiters are specified by the use of if-statements which is at the expense of computation time. The advantage of these limiters, however, is a slight increase of accuracy.

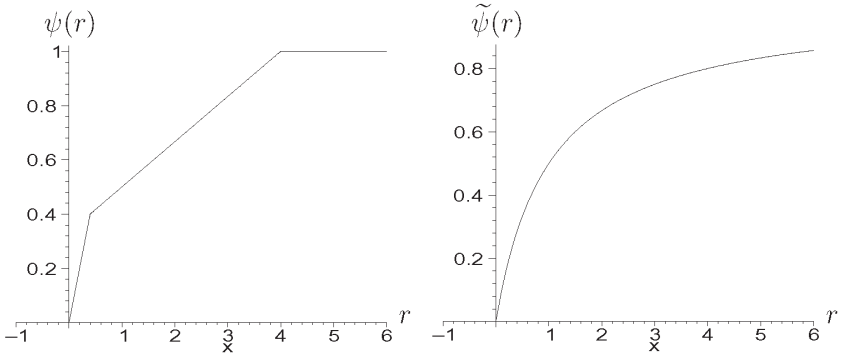


Figure 12.6: Left the Kooren limiter function (12.3.28) and right the Van Leer limiter function.

Further, one can use a predictor-corrector method to improve the accuracy of the solution.

12.4 Mathematical theory for the transport equation

In this section some mathematical background of first order hyperbolic conservation laws is given. This background is commonly used to check the results obtained from numerical simulations. The present treatment is for a scalar conservation law. For systems of hyperbolic PDE's some mathematical theory is presented in Smoller [33].

Since Burgers equation is the simplest case of a non-linear transport equation, first traveling wave solutions for Burgers equation are analyzed. Subsequently, smooth solutions and discontinuous solutions for the Buckley-Leverett with a convex flux function are discussed. Finally, the convexity condition for the Buckley-Leverett equation is relaxed and the construction of analytical solutions is shown.

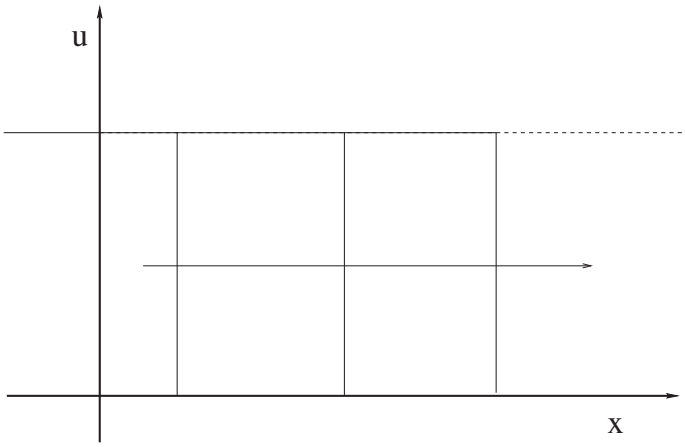


Figure 12.7: Sketch of a shock solution to Burgers equation with $\nu \to 0$ at several times. The shock moves to the right.

12.4.1 Burgers equation

Burgers equation appears in some models from hydrodynamics and aerodynamics. It derivation follows from the conservation of momentum, i.e. a special case of momentum equations due to Euler, in one spatial dimension. This equation is the following:

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}u^2\right) = 0. \tag{12.4.1}$$

Exercise 12.4.1 *Given the above Burgers equation, on the unbounded domain* $\mathbb{R}$*, subject to the initial condition*

$$u(x,0) = \begin{cases} 1, & x < 0, \\ 0, & x > 0. \end{cases} \tag{12.4.2}$$

Show by use of the characteristics that the solution of this problem develops into a shock (see Figure 12.7).

One can show that the general hyperbolic conservation Law for smooth $f(u)$ can be transformed into the above Burgers equation, this also motivates that Burgers equation is an important model for which some qualitative properties will be derived. We will see that solutions of Burgers equation are discontinuous (see Figure 12.7 with shocks) or continuous (see Figure 12.8), depending on the initial / boundary conditions. First we consider the equation with the incorporation of an extra diffusive term, which allows us to have smooth solutions of which the derivatives exist:

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}u^2\right) = \nu\frac{\partial^2 u}{\partial x^2}, \tag{12.4.3}$$

where $\nu > 0$ denotes the viscosity. To get some insight into the structure of the solutions of this equation, we consider the existence of *traveling wave solutions* of the above equation, which are given by $u(x,t) = f(\eta)$ with $\eta = x - st$ (s is to be determined). The solutions that we consider in this section are on the unbounded domain, i.e. $x \in \mathbb{R}$ and $t > 0$. Further, we consider bounded solutions with horizontal asymptotes only, i.e. there exists values u_L and u_R such that

$$\lim_{x\to-\infty} u(x,t) = u_L \text{ and } \lim_{x\to\infty} u(x,t) = u_R. \tag{12.4.4}$$

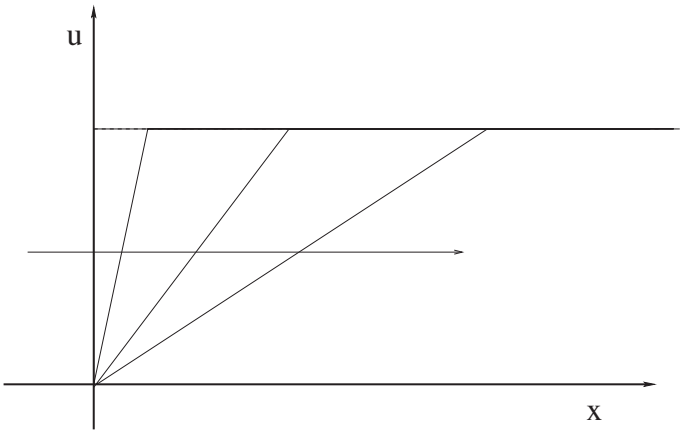


Figure 12.8: Sketch of a continuous solution to Burgers equation with $\nu \to 0$ at several times.

Then Equation (12.4.3) changes into

$$-sf' + \left(\frac{1}{2}f^2\right)' = \nu f''. \tag{12.4.5}$$

After integration of the above equation we obtain

$$-sf + \frac{1}{2}f^2 = \nu f' + A, \tag{12.4.6}$$

where A is a constant of integration. The solution f is assumed to be smooth. Hence, for $|\eta| \to \infty$ we must have $\lim_{|\eta|\to\infty} f'(\eta) = 0$. The numbers s and A are determined from the use of the boundary conditions and $\lim_{|\eta|\to\infty} f'(\eta) = 0$, to obtain

$$-su_L + \frac{1}{2}u_L^2 = A; \tag{12.4.7}$$

$$-su_R + \frac{1}{2}u_R^2 = A. \tag{12.4.8}$$

We solve these equations for s and A to get

$$s = \frac{1}{2}(u_R + u_L) \tag{12.4.9}$$

$$A = \frac{1}{2}u_R u_L. \tag{12.4.10}$$

Note that s defines the velocity of the traveling wave. Substitution of equation (12.4.10) into Equation (12.4.6) gives

$$2\nu f' = (f - u_R)(f - u_L) < 0, \tag{12.4.11}$$

where the inequality holds for f between u_L and u_R. Hence, the solution f should be within the interval between u_L and u_R otherwise the boundary conditions cannot be satisfied unless the solution contains a discontinuity. This implies that if $u = u(x,t)$ satisfies the traveling wave behavior then it is a decreasing function with respect to x. This also implies that $u_L \geq u_R$ only admits traveling wave

solutions , since $u_L < u_R$ requires an increasing function, i.e. $f'(\eta) > 0$, which contradicts Equation (12.4.11). For $u_L \geq u_R$ Equation (12.4.11) is solved by the use of separation of variables to get

$$f(\eta) = u_R + \frac{u_L - u_R}{1 + \exp\left(\frac{u_L - u_R}{2\nu}\eta\right)}, \tag{12.4.12}$$

hence

$$u(x,t) = u_R + \frac{u_L - u_R}{1 + \exp\left(\frac{u_L - u_R}{2\nu}(x - st)\right)}, \tag{12.4.13}$$

with

$$s = \frac{u_L + u_R}{2}. \tag{12.4.14}$$

Note that if $\nu \to 0$ then the solution tends to a shock behavior:

$$\lim_{\nu \to 0} f(\eta) = \begin{cases} u_R, \text{ for } \eta > 0, \\ u_L, \text{ for } \eta < 0. \end{cases} \tag{12.4.15}$$

For the case that $u_R > u_L$ we saw no traveling wave solutions, i.e. $u(x,t) = f(\eta)$ with $\eta = x - st$, does not exist. However, then a solution of a different structure exists. This will be analyzed in the presentation of the Buckley-Leverett equation. The most important conclusions of this section are that the solution of Burgers equation tends to be discontinuous under $u_L > u_R$ as $\nu \to 0$ and that traveling wave solutions exists provided $u_L \geq u_R$.

12.4.2 The Buckley-Leverett equation

The Buckley-Leverett equation plays a crucial role in the flow of two phases in porous media. Its derivation is based on the concept of *relative permeabilities*. For a derivation, we refer to the book of Bear [4]. We consider the Buckley-Leverett equation with (piecewise) smooth solutions, such that the derivatives of the solutions exist. Further, it is assumed that $f(u)$ is a smooth function too. This equation reads as

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \Rightarrow \frac{\partial u}{\partial t} + f'(u)\frac{\partial u}{\partial x} = 0. \tag{12.4.16}$$

We will see that the character of the solution of the above equation depends on the flux function $f(u)$. Consider a point (x_0, t_0) in the x,t-plane and the initial value problem for the characteristics

$$\frac{dx}{ds} = f'(u), \quad \frac{dt}{ds} = 1, \text{ where } \rho(s) = 1, \tag{12.4.17}$$

then we consider characteristics of Equation (12.4.16) in terms of $x(t)$ since

$$\frac{dx}{dt} = f'(u), x(t_0) = x_0.$$

At this curve the following holds after combination with Equation (12.4.17)

$$\frac{d}{dt}u(x(t),t) = \frac{\partial u}{\partial x}(x(t),t)x'(t) + \frac{\partial u}{\partial t}(x(t),t) = \frac{\partial u}{\partial x}f'(u) + \frac{\partial u}{\partial t} = 0. \tag{12.4.18}$$

The first equality follows the Chain Rule for differentiation. Note that for all the differentiations it is necessary that the derivatives exist. We only consider (piecewise) smooth solutions. The method of characteristics is used to study qualitative

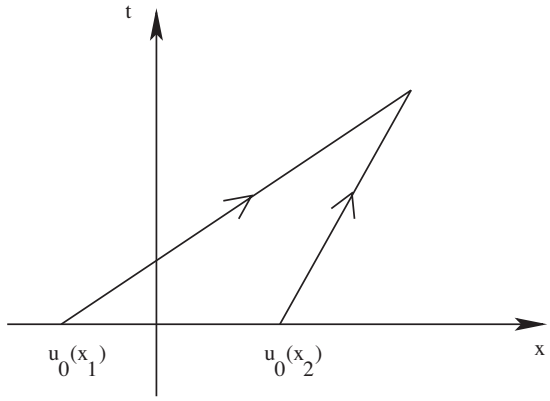


Figure 12.9: Formation of a discontinuity.

aspects of the solution of the Buckley-Leverett equation. As an example we treat the case that $f(u)$, $f'(u)$, $f''(u) > 0$. Suppose that $u(x,0) = u_0(x)$ is given as the initial condition. Now we show by a contradiction argument that no smooth solutions for $u(x,t)$ can exist if $u_0(x)$ is a decreasing function. We recall the characteristic equation (12.4.17) and assume that $u_0(x)$ is a decreasing function. We take two points (x_1, t_0) and (x_2, t_0) with $x_2 > x_1$ (see Figure 12.9. Then, $u(x_2, t_0) < u(x_1, t_0)$ and since $f''(u) > 0$, we obtain

$$\frac{dx_1}{dt} = f'(u(x_1(t), t)) > f'(u(x_2(t), t)) = \frac{dx_2}{dt}. \tag{12.4.19}$$

The last equation is justified since $\frac{d}{dt}u(x(t), t) = 0$ ($u(x,t)$ is constant along its characteristics and hence the characteristics are straight lines). Relation (12.4.19) implies that characteristics intersect and hence the solution becomes multi-valued and the model breaks down. At this point it is possible to assign a large class of solutions to the model problem. Later, in this section, it will turn out that only one solution is physically relevant since it conserves mass. This is a solution with a discontinuity.

As an other example we consider the case $f''(u) > 0$ with the initial condition

$$u(x,0) = u_0(x) = \begin{cases} 0, & \text{for } x < 0, \\ 1, & \text{for } x \geq 0. \end{cases} \tag{12.4.20}$$

Use of the method of characteristics gives with $x_1 \to 0^-$ and $x_2 = 0$ and hence

$$x_1'(t) = f'(u(x_1(t), t)) < f'(u(x_2(t), t)) = x_2'(t). \tag{12.4.21}$$

This implies that the characteristics diverge (see Figure 12.10). For this case we will have a continuous solution. We see that the method of characteristics gives insight into whether or not smooth solutions are possible or whether a discontinuous initial condition stays a shock or develops into a smooth solution. The nature of the continuous and discontinuous solutions will be discussed in the next two sections.

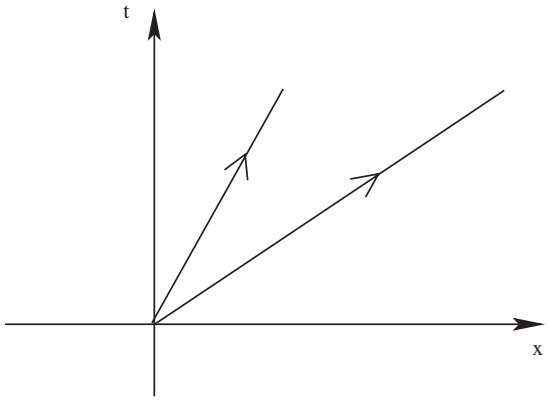


Figure 12.10: Formation of the rarefaction or expansion wave.

12.4.2.1 The smooth solution

For $u_0(x) = \begin{cases} 0, \text{ for } x < 0 \\ 1, \text{ for } x \geq 0 \end{cases}$ and $f''(u) > 0$, then, from the characteristics we saw that the solution loses its discontinuous behavior. Since in a numerical method for a PDE the initial drop occurs over at least one grid-spacing, the numerical initial condition is continuous. The method of characteristics then implies that the solution stays continuous. Further, for $f''(u) < 0$ we will have convergence to a discontinuous solution as time proceeds. This will be the topic in the next section. Now we examine the continuous solution when $f''(u) > 0$. Therefore, we set $u(x,t) = g(\eta)$, $\eta = \dfrac{x}{t}$ with $f'(0) < \eta < f'(1)$, then substitution into the Buckley-Leverett equation (12.4.16) gives with use of the Chain Rule for differentiation

$$-\frac{x}{t^2}g'(\eta) + \frac{1}{t}\frac{d}{d\eta}f(g(\eta)) = 0. \tag{12.4.22}$$

This gives

$$\eta g'(\eta) = f'(g(\eta))g'(\eta). \tag{12.4.23}$$

Hence $g'(\eta) = 0$ (constant state solution) or

$$f'(g(\eta)) = \eta \Rightarrow g(\eta) = (f')^{-1}(\eta). \tag{12.4.24}$$

In the implication we assume that $f'(u)$ is invertible on the domain of consideration, the inverse of $f'(u)$ is denoted by $(f')^{-1}$. For a bounded solution $u \in [0,1]$ we have that

$$u(x,t) = \begin{cases} 0, & \text{for } x < f'(0)t, \\ (f')^{-1}(\frac{x}{t}), & \text{for } f'(0)t < x < f'(1)t, \\ 1, & \text{for } x > f'(1)t. \end{cases} \tag{12.4.25}$$

A solution with the structure of Equation (12.4.25) is called a *rare faction* or an *expansion wave*. Physically, this often amounts a mixing behavior of two phases or *viscous fingering* (as a Saffmann-Taylor instability), see Bear [4].

Example 12.4.1 *For Burgers equation we have* $f(u) = \frac{u^2}{2}$, *hence* $f'(u) = u$. *Therefore*

$(f')^{-1}(\eta) = \eta$ *and the solution becomes with* $f'(0) = 0$ *and* $f'(1) = 1$*:*

$$u(x,t) = \begin{cases} 0, & \text{for } x < 0, \\ \dfrac{x}{t}, & \text{for } 0 \le x \le t, \\ 1, & \text{for } x > t. \end{cases} \tag{12.4.26}$$

12.4.2.2 The discontinuous solution

We consider the case $f'' > 0$ ($f(u)$ is convex), and $u(x,0) = u_0(x) = \begin{cases} 1 \text{ for } x < 0 \\ 0 \text{ for } x \ge 0 \end{cases}$, for the Buckley-Leverett equation. We saw for Burgers equation that the shock speed is given by

$$s = \frac{u_L + u_R}{2}. \tag{12.4.27}$$

We are going to calculate the shock speed by the use of a mass conservation argument for the Buckley-Leverett equation in general. An alternative formal derivation can be given by a consideration of weak solutions and compact support. This is beyond the scope of this book and we refer the interested reader to the book of Smoller [33]. Consider integration over $x \in [a,b]$ of the Buckley-Leverett equation, where a and b are chosen such that the interval contains the shock position:

$$\int_a^b \frac{\partial u}{\partial t} = -\int_a^b \frac{\partial f(u)}{\partial x} dx = f(u(a,t)) - f(u(b,t)). \tag{12.4.28}$$

Since the bounds in the above integral do not depend on t, we may interchange the order of differentiation with respect to time and integration over the fixed interval $[a,b]$. Further, the quantity $\int_a^b u dx$ depends on t only, hence the partial differentiation with respect to t can be written as an ordinary derivative with respect to t, to give

$$\frac{d}{dt}\int_a^b u dx = f(u(a,t)) - f(u(b,t)). \tag{12.4.29}$$

Now suppose that the solution u is discontinuous at a curve $s(t)$ where $a < s(t) < b$, then by the use of Leibniz' Rule and the Chain Rule for differentiation follows

$$\begin{aligned} \frac{d}{dt}\int_a^b u dx &= \frac{d}{dt}\left[\int_a^{s(t)} u dx + \int_{s(t)}^b u dx\right] \\ &= \int_a^{s(t)} \frac{\partial u}{\partial t} dx + u(s^-(t),t)s'(t) + \int_{s(t)}^b \frac{\partial u}{\partial t} dx - u(s^+(t),t)s'(t). \end{aligned} \tag{12.4.30}$$

Here we define $s^-(t)$ and $s^+(t)$ as the positions adjacent to the left and the right side of the shock position respectively. Use of the Buckley-Leverett equation (12.4.16) gives

$$\begin{aligned} \frac{d}{dt}\int_a^b u dx = f(u(a,t)) - f(u(s^-(t),t)) + u(s^-(t),t)s'(t) + \\ f(u(s^+(t),t)) - f(u(b,t)) - u(s^+(t),t)s'(t). \end{aligned} \tag{12.4.31}$$

Since Equation (12.4.29) holds, it follows from Equation (12.4.31) that

$$\big(u(s^-(t),t) - u(s^+(t),t)\big)\, s'(t) = f(u(s^-(t),t)) - f(u(s^+(t),t)). \tag{12.4.32}$$

Let $s'(t)$ be the shockspeed, then if $u(s^-(t),t) - u(s^+(t),t) \neq 0$, then

$$s'(t) = \frac{f(u(s^-(t),t)) - f(u(s^+(t),t))}{u(s^-(t),t) - u(s^+(t),t)} =: \frac{[f(u)]}{[u]}. \tag{12.4.33}$$

This equation is known as the Rankine-Hugoniot condition. Note that if $u(s^+(t),t)$ tends to $u(s^-(t),t)$ (i.e. the continuous case) then the speed of a characteristic is recovered (see exercise 19.1).

Exercise 12.4.2 *Show that, for $u(s^+(t),t)$ tending to $u(s^-(t),t)$, the speed of a characteristic is recovered.*

The case of $f''(u) > 0$ has been examined now, the case of $f''(u) < 0$ can be addressed likewise. This is left as an exercise.

Exercise 12.4.3 *Given the Buckley-Leverett equation with $f''(u) < 0$ with $u(x,0) = \begin{cases} u_L, x < 0 \\ u_R, x \geq 0 \end{cases}$, $u_L \neq u_R$. Under which conditions will be the shock be stable and under which conditions will a rarefaction develop? Motivate this by the use of characteristics.*

Exercise 12.4.4 *Given Burgers equation*

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{2}\right) = 0, \text{ with } u = \begin{cases} 1, x < 0 \\ 0, x \geq 0 \end{cases}. \tag{12.4.34}$$

Describe the solution for $x \in \mathbb{R}$ and $t > 0$.

12.4.2.3 The non-convex case

We abandon the convexity condition for $f(u)$ and consider

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \text{ with } f(u) = \frac{u^2}{u^2 + (1-u)^2}. \tag{12.4.35}$$

This choice of $f(u)$ arises in many applications of two-phase flow in porous media where influences of gravity are neglected. The following initial condition is used

$$u(x,0) = u_0(x) = \begin{cases} 1, & x < 0, \\ 1 - \frac{x}{\epsilon}, & 0 \leq x \leq \epsilon, \\ 0, & x > \epsilon \end{cases} \quad \text{for some } \epsilon > 0. \tag{12.4.36}$$

To gain insight into the qualitative behavior of the solution, characteristics are used:

$$\frac{dt}{ds} = 1, \quad \frac{dx}{ds} = f'(u) \text{ hence } \frac{dx}{dt} = f'(u). \tag{12.4.37}$$

Since $f'(u) = 0$ for $u = 0$ and $u = 1$, it is clear that characteristics, originating from the negative part of the x-axis ($x < 0$) and from $x > \epsilon$, move vertically upward. Further, since $f''(u) = 0$ at $u = \frac{1}{2}$ (corresponding with $x = \frac{\epsilon}{2}$ at $t = 0$) the slope of the characteristics tends to be less vertical as x increases within the interval $0 < x < \frac{\epsilon}{2}$ (at $t = 0$). In the interval $\frac{\epsilon}{2} < x < \epsilon$ at $t = 0$, the characteristics tend to be more vertical again. This is illustrated in Figure 12.11. For the characteristics originating from the x-axis, the following interesting features are observed:

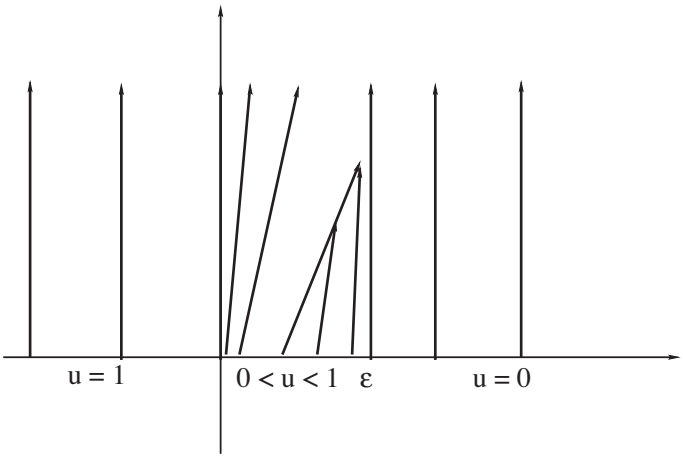


Figure 12.11: A sketch of the characteristics that originate from the x-axis.

- Characteristics originating from the interval $0 < x < \frac{\epsilon}{2}$ diverge and hence a rare faction develops.
- Characteristics originating from the interval $\frac{\epsilon}{2} < x < \epsilon$ converge and intersect and hence a shock develops.

In the remainder of this section some mathematical background for the construction of analytical solutions is presented. For this purpose a traveling wave argument is given for the Buckley-Leverett equation with an added 'viscosity term':

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, \quad \nu > 0. \tag{12.4.38}$$

We only consider solutions for which

$$u(-\infty, t) = u_L, \qquad u(\infty, t) = u_R. \tag{12.4.39}$$

For the traveling wave structure, we set

$$u(x,t) = v(\eta), \text{ with } \eta = \frac{x - ct}{\nu}. \tag{12.4.40}$$

This equation transforms with the 'boundary conditions' into

$$\begin{array}{l} -cv' + (f(v))' = v'' \\ v(-\infty) = u_L \text{ and } v(\infty) = u_R. \end{array} \tag{12.4.41}$$

The above differential equation is integrated to obtain

$$-cv + f(v) = v' + A. \tag{12.4.42}$$

In the above equation c and A are determined from the 'boundary conditions' at $\pm\infty$. Since v has horizontal asymptotes at $|\eta| \to \infty$, it is necessary that $v'(\eta) \to 0$ as $|\eta| \to \infty$. Hence the 'conditions' at $|\eta| \to \infty$ imply

$$\left.\begin{array}{l} -cu_L + f(u_L) = A \\ -cu_R + f(u_R) = A \end{array}\right\} \Rightarrow c = \frac{f(u_R) - f(u_L)}{u_R - u_L}, \quad A = f(u_L) - cu_L. \tag{12.4.43}$$

Hence Equation (12.4.42) changes into

$$v' = f(v) - f(u_L) + c(u_L - v). \tag{12.4.44}$$

Suppose that there exists a $\hat{v}$ between u_L and u_R for which

$$f(\hat{v}) - f(u_L) + c(u_L - \hat{v}) = 0 \Rightarrow v' = 0 \text{ at } v = \hat{v}. \tag{12.4.45}$$

Then we are at an equilibrium point of Equation (12.4.44) and hence $v = \hat{v}$ for $\eta \in \mathbb{R}$ and herewith a contradiction with the boundary conditions is obtained. This implies that v is a strictly monotonic function of η and hence a traveling wave is strictly monotonic. Therewith

$$\begin{array}{l} u_L < u_R \Rightarrow v' > 0 \Rightarrow v > u_L \\ u_L > u_R \Rightarrow v' < 0 \Rightarrow v < u_L. \end{array} \tag{12.4.46}$$

Division of Equation (12.4.44) by $u_L - v$ and use of the above observations gives after some rearrangement

$$c = \frac{f(u_L) - f(v)}{u_L - v} + \frac{v'}{u_L - v} < \frac{f(u_L) - f(v)}{u_L - v}, \tag{12.4.47}$$

since $\frac{v'}{u_L - v} < 0$. The above equation is formulated as

$$c = \frac{f(u_R) - f(u_L)}{u_R - u_L} < \frac{f(u_R) - f(v)}{u_R - v}, \tag{12.4.48}$$

for all v between u_L and u_R. The above inequality is a sufficient and necessary condition for the existence of a traveling wave solution. It is clear that relation (12.4.48) is a consequence of the above arguments and hence (12.4.48) poses a necessary condition for the existence of a traveling wave solution. Next we show that condition (12.4.48) is sufficient for the existence of a traveling wave solution, i.e. (12.4.48) guarantees the existence of a traveling wave, therefore we integrate Equation (12.4.44) to obtain

$$\int_{\frac{u_L + u_R}{2}}^{v(\eta)} \frac{ds}{f(s) - f(u_L) + c(u_L - s)} = \eta. \tag{12.4.49}$$

The traveling wave solution exists if the above integral exists. Condition (12.4.48) implies that the denominator of the above integrand is non-zero for v between u_L and u_R. Hence, the integrand is bounded and therefore the above integral exists. This guarantees the existence of a traveling wave. Now, suppose that $f(u)$ is convex-concave and that $f(0) = 0$ and $f(1) = 1$, we investigate the possibility for traveling waves. See Figure 12.12 for a sketch of $f(u)$. We distinguish the following cases:

- $u_L > u_R = 0$, then $v' < 0$ and hence from Equation (12.4.44) follows

  $$f(v) < f(u_L) - c(u_L - v) = f(u_L) + \frac{f(u_L)}{u_L}(v - u_L), \text{ note that } u_R = 0 = f(u_R). \tag{12.4.50}$$

  The graph of the right-hand side of the above inequality is indicated by the dotted line in Figure 12.12. The position u_1 is the intersection of the dotted line and $f(v)$ and hence the above inequality no longer holds if $v \geq u_1$ and thus traveling waves exists for values of $0 = u_R \leq u_L < u_1$.

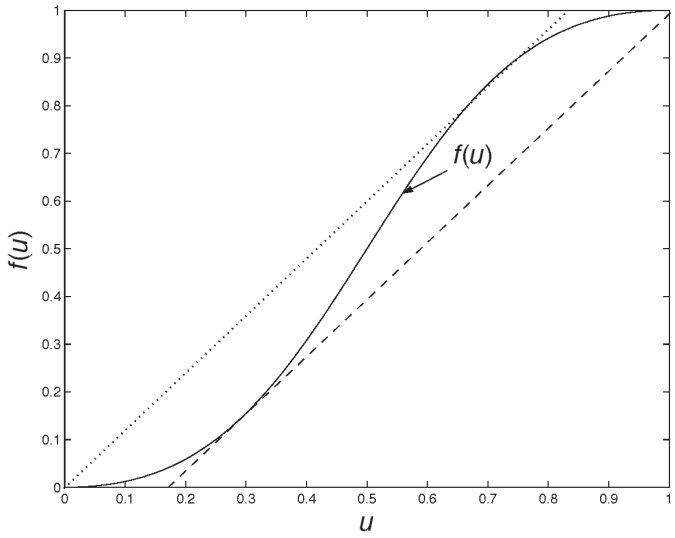


Figure 12.12: A sketch of the function f.

- $u_L < u_R = 1$, then $v' > 0$ and hence from Equation (12.4.44) follows

$$f(v) > f(u_L) + \frac{f(u_L) - 1}{u_L - 1}(v - u_L), \text{ note that } u_R = 1 = f(u_R). \quad (12.4.51)$$

  The graph of the right-hand side of the above inequality is indicated by the dashed line in Figure 12.12. The position u_2 is the intersection of the dashed line and the function $f(v)$ and hence the above inequality no longer holds if $v \leq u_2$ and thus traveling waves exist for values of $u_2 < u_L \leq u_2 = 1$.

Since condition (12.4.48) holds for any $\nu > 0$ (hence also for $\nu \to 0$), it is used as an additional 'entropy condition' for traveling waves. As $\nu \to 0$ this traveling wave becomes a shock, due to the intersection of the characteristics, with velocity

$$c = \frac{f(u) - f(u_L)}{u - u_L} \geq \frac{f(u_R) - f(u_L)}{u_R - u_L} \text{ for } u \text{ between } u_L \text{ and } u_R. \quad (12.4.52)$$

Solutions possibly consist of a combination of a rare faction and a shock.

12.4.2.4 Construction of solutions

Let $f(u)$ have a continuous second order derivative on the interval $[0, 1]$, and let f satisfy the following requirements:

$$\begin{cases} f(s) > 0, f'(s) > 0 \text{ for } s \in (0,1) \\ f(0) = 0, f(1) = 1 \\ f''(s) > 0 \text{ for } s \in [0, \hat{s}) \\ f''(s) < 0 \text{ for } s \in (\hat{s}, 1] \end{cases} \quad (12.4.53)$$

for some $\hat{s} \in (0,1)$. Further, $u(x,t)$ satisfies

$$\begin{cases} \dfrac{\partial u}{\partial t} + \dfrac{\partial f(u)}{\partial x} = 0 \\ u(x,0) = u_0(x) = \begin{cases} 1, & x < 0 \\ 0, & x > 0 \end{cases} \end{cases} \quad (12.4.54)$$

We will construct a solution which consists of a rarefaction and a shock. We see that $u_R = 0$. For the traveling wave part one obtains that a traveling wave is allowed for $0 = u_R \leq u_L < u_1 < 1$, where u_1 is the value at which

$$f'(u_1) = \frac{f(u_1) - f(u_R)}{u_1 - u_R} \text{ and } f(u_1) = \frac{f(u_1) - f(u_R)}{u_1 - u_R} u_1, (\text{ note that } u_R = 0). \tag{12.4.55}$$

This follows from the arguments of the preceding section. The right-hand sides of the two equations are consecutively the derivative of the straight line and the line itself at $u = u_1$. Both expressions imply that u_1 satisfies

$$u_1 - \frac{f(u_1)}{u_1} = 0. \tag{12.4.56}$$

This equation can be solved for u_1 by the use of a zero-point method. There is a shock over $u_R = 0$ and $u_L = u_1$ which travels at the constant speed $c = \frac{f(u_1)}{u_1}$. For the part $u \in [u_1, 1]$ no traveling wave exists, there a rarefaction is obtained, i.e. $u = g(\eta)$, $\eta = \frac{x}{t}$, for $u_L = 1$ and $u_R = u_1$ as the respective left- and right state. Substitution of this rarefaction behavior, gives

$$g'(\eta) = 0 \text{ (constant state) or } \eta = f'(g(\eta)). \tag{12.4.57}$$

This implies that

$$g(\eta) = \begin{cases} 1, \text{ for } 0 < \eta < f'(1) \\ (f')^{-1}(\eta) \text{ for } f'(1) < \eta < f'(u_1) \\ 0 \text{ for } \eta > f'(u_1) \end{cases} . \tag{12.4.58}$$

Hence the solution is constructed by

$$u(x,t) = \begin{cases} 1, \text{ for } 0 < x < f'(1)t \\ (f')^{-1}(\frac{x}{t}) \text{ for } f'(1)t < x < f'(u_1)t \\ 0 \text{ for } x > f'(u_1)t \end{cases} . \tag{12.4.59}$$

Exercise 12.4.5 *Construct the analytical solution of the same problem except for the initial condition*

$$u(x,t) = u_0(x) = \begin{cases} 0, \textit{ for } x < 0, \\ 1, \textit{ for } x > 0, \end{cases} \qquad (u_R = 1), \tag{12.4.60}$$

by use of the arguments of the preceding section.

12.5 Summary of Chapter 12

This chapter treats an introduction into the transport equation, which is also commonly referred to as a first order hyperbolic conservation law. To gain insight into the behavior of the solutions and the nature of the boundary and initial conditions, characteristics are treated. Formerly, many numerical techniques were based on a direct solution of the characteristic relation. Further, the most classical numerical methods for the solution of the transport equation are described. Finally, some mathematical aspects of the structure of the solution are presented. First, the flux-function $f(u)$ is assumed to be convex. Subsequently, the convexity condition for $f(u)$ is dropped. This case is crucial when considering two-phase flow in porous media without gravity.

12.6 Appendix: requirements on flux-limiters

In this appendix we comment on the requirements on the limiter function. The requirements

$$\begin{aligned} 0 \leq \psi(r_i) &\leq r_i \\ 0 \leq \psi(r_i) &\leq 1, \end{aligned} \tag{12.6.1}$$

are demonstrated in this section for the linear hyperbolic partial differential equation only, i.e.

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0. \tag{12.6.2}$$

Before we continue the motivation, the concept of the Total Variation (TV) is introduced:

$$TV(w) := \sum_{i=-\infty}^{i=\infty} |w_i - w_{i-1}|. \tag{12.6.3}$$

The total variation is high for oscillatory functions and zero for functions with a constant value. The only attractive numerical methods contain a decreasing total variation. Because these methods will give a damping of unphysical oscillations. Let j denote the time-step, then, it is required that:

$$TV(w^{j+1}) \leq TV(w^j), \tag{12.6.4}$$

then the method is called Total Variation Diminishing (or briefly TVD).

We shall show now that Euler forward time integration is TVD, if conditions (12.6.1) are satisfied. Euler forward time integration gives

$$w_i^{j+1} = w_i^j - \Delta t \frac{f_{i+1/2}^j - f_{i-1/2}^j}{\Delta x}, \tag{12.6.5}$$

with

$$f_{i+1/2}^j = w_i^j + \psi(r_i)(w_{i+1}^j - w_i^j). \tag{12.6.6}$$

Here r_i is a measure of the smoothness of the data, defined by

$$r_i := \frac{w_i^j - w_{i-1}^j}{w_{i+1}^j - w_i^j}. \tag{12.6.7}$$

We write this as

$$w_i^{j+1} = w_i^j - \frac{\Delta t}{\Delta x}(w_{i+1}^j - w_i^j)\left(1 + \frac{\psi(r_i)}{r_i} - \psi(r_i)\right), \tag{12.6.8}$$

where the Definition (12.6.7) for r_i was used. Harten shows that a sufficient condition for TVD in the sense of relation (12.6.4) is given by

$$0 \leq 1 + \frac{\psi(r_i)}{r_i} - \psi(r_{i-1}) \leq 1. \tag{12.6.9}$$

This is motivated as follows. From Equation (12.6.8) follows for w_i^{j+1}:

$$w_{i-1}^{j+1} = w_{i-1}^j - \frac{\Delta t}{\Delta x}(w_i^j - w_{i-1}^j)\left(1 + \frac{\psi(r_{i-1})}{r_{i-1}} - \psi(r_{i-1})\right), \tag{12.6.10}$$

Subtraction of Equation (12.6.8) from (12.6.10), gives

$$w_i^{j+1} - w_{i-1}^{j+1} = (w_i^j - w_{i-1}^j)\left(1 - \frac{\Delta t}{\Delta x}\left(1 + \frac{\psi(r_i)}{r_i} - \psi(r_{i-1})\right)\right) + \\ + \frac{\Delta t}{\Delta x}\left(1 + \frac{\psi(r_{i-1})}{r_{i-1}} - \psi(r_{i-2})\right)(w_{i-1}^j - w_{i-2}^j). \tag{12.6.11}$$

Subsequently we take the absolute value and sum over all indices i and use the triangle identity to obtain:

$$\sum_{i=-\infty}^{i=\infty} |w_i^{j+1} - w_{i-1}^{j+1}| \leq \sum_{i=-\infty}^{i=\infty} |w_i^j - w_{i-1}^j| \left(1 - \frac{\Delta t}{\Delta x}\left(1 + \frac{\psi(r_i)}{r_i} - \psi(r_{i-1})\right)\right) \\ + \sum_{i=-\infty}^{i=\infty} \frac{\Delta t}{\Delta x}\left(1 + \frac{\psi(r_{i-1})}{r_{i-1}} - \psi(r_{i-2})\right)|w_{i-1}^j - w_{i-2}^j| \tag{12.6.12}$$

Next we require condition (12.6.1) to hold and shift the index of the second summation in the right-hand side so that most terms cancel. Hence we are left with

$$\sum_{i=-\infty}^{i=\infty} |w_i^{j+1} - w_{i-1}^{j+1}| \leq \sum_{i=-\infty}^{i=\infty} |w_i^j - w_{i-1}^j|. \tag{12.6.13}$$

Hence the discretization is TVD if condition (12.6.1) is satisfied.

Chapter 13

Moving boundary problems

Objectives

In previous chapters several numerical methods have been presented and applied to model problems: (Navier) Stokes and Euler equations, Transport in porous media, Diffusion problems and Wave equations. In industrial applications a large amount of completely different problems arise. An important class of problems is that of free and moving boundaries. In free-surface problems the boundary (or interface) is not known a-priori but it is a part of the solution. In case of moving boundaries the boundary changes in time. Here an example of a moving boundary problem, the so-called *Stefan problem*, is given. This describes, for example, melting of ice or solidification of liquid metals.

Free/moving boundary problems arise in phase transitions (such as solidification), flow problems, crystal growth, steam injection in oil and gas reservoirs and in bubbly flow. Free boundary problems also occur in finance, where they are solved to determine the price of a call option (the right to purchase shares). In this chapter a moving boundary problem with heat diffusion (Fourier), which is referred to as the classical *Stefan problem*, is presented. The model is applied to freezing of water. Several well-known numerical solution procedures to solve Stefan problems are presented. The advantages and disadvantages of particular methods will be described. For a qualitative picture of the solution a reference is given to exact solutions that hold whenever the domain is of infinite size.

13.1 The formulation of a classical Stefan problem: ice and water

The scientist J. Stefan studied the melting and freezing of the ice-caps near the North pole of the earth [34]. Using his experiments, he formulated a model to describe the area of the ice-caps as a function of time and the classical Stefan problem was born. Weber [47] was, as far as known the first to study the Stefan problem mathematically and he found a so-called 'self-similar' solution. For more historical and mathematical background on the classical Stefan problem and its relating mathematics, we refer to the work of Vuik [46, 45]. In this chapter we consider freezing of water. For more physical background we refer to the textbook of Carslaw and Jaeger [9] . For the sake of illustration, we consider an open rectangular domain $\Omega := \{(x,y) \in \mathbb{R}^2 : 0 < x < 1, 0 < y < 1\}$ that is initially filled with water at temperature T_0. The boundary of Ω is given by $\partial\Omega$, which is divided into $\partial\Omega_1$, $\partial\Omega_2$, $\partial\Omega_3$ and $\partial\Omega_4$. The areas, occupied by ice and water, are respectively

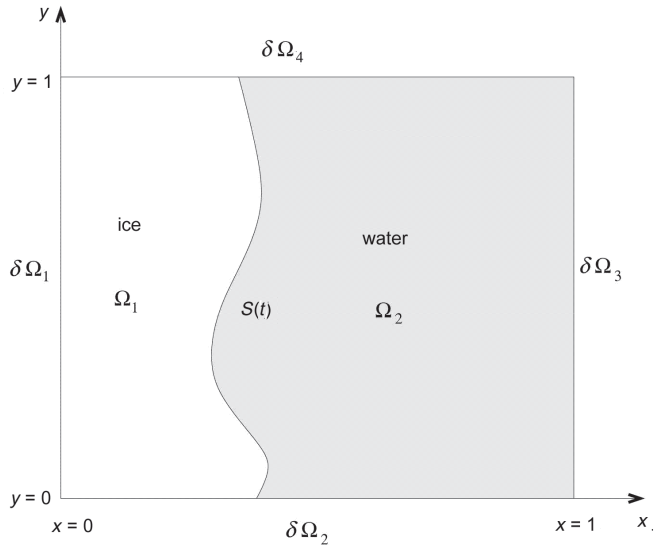


Figure 13.1: Geometry of the domain.

given by the non-overlapping subdomains $\Omega_1(t)$ and $\Omega_2(t)$, where the boundary between water and ice is given by the line $S = S(t)$. This boundary $S(t)$ is also referred to as the (moving) interface. The geometry is shown in Figure 13.1. The temperature in Ω_1 and Ω_2 is denoted by respectively T_1 and T_2, the temperature in Ω is denoted by $T(x, y, t)$.

At the interface, $(x, y) \in S(t)$, between water and ice the temperature is $T = T_s$ (freezing temperature) at all times. At the initial stage we have a temperature T_0 and we assume that the whole domain is filled with water: $\Omega_2(0) = \Omega$. At $x = 0$, the temperature is prescribed: $T = T_*$ for $(x, y) \in \partial\Omega_1$, $T_* < T_s < T_0$. Since T_* is below the freezing temperature, the water freezes as time proceeds. The moving interface starts at $x = 0$ and moves to the right. We assume that heat transport in ice and water only takes place by conduction, i.e. heat diffusion. A further assumption is that there is no heat flux across the other boundaries $\partial\Omega_2$ to $\partial\Omega_4$. The normal velocity of the interface is denoted by v_n. Summarizing we have the following mathematical problem where in both ice and water a heat equation is satisfied. At the interface an amount of heat is produced by the freezing of the water (*latent heat*). The differential equation describing the process in the interior of the domain is given by:

$$\rho c \frac{\partial T}{\partial t} = \lambda \Delta T, \qquad \mathbf{x} \in \Omega, \tag{13.1.1}$$

with initial condition

$$T = T_0, \qquad \mathbf{x} \in \Omega. \tag{13.1.2}$$

The boundary conditions are

$$T = T_*, \qquad \mathbf{x} \in \partial\Omega_1, \tag{13.1.3}$$

$$\frac{\partial T}{\partial n} = 0, \qquad \mathbf{x} \in \bigcup_{k=2}^{4} \partial\Omega_k \tag{13.1.4}$$

On the interface we need the following two conditions

$$T = T_s, \qquad \mathbf{x} \in S(t), \tag{13.1.5}$$

and (latent heat)

$$\rho_1 L v_n = \lambda_1 \frac{\partial T_1}{\partial n} - \lambda_2 \frac{\partial T_2}{\partial n}, \qquad \mathbf{x} \in S(t). \tag{13.1.6}$$

We need two interface conditions to calculate the position of the interface. This problem is the classical Stefan problem in a rectangular domain. Equation (13.1.6) gives the rate of the interface. The unknowns are the temperature T and the position of the interface S between both phases. The densities of water and ice are respectively given by ρ_1 and ρ_2. The parameter L represents the latent heat of solidification. The parameters c_1, c_2 and λ_1, λ_2 denote the heat capacities and heat conductivities of respectively ice and water. Existence and uniqueness of a solution pair T and S has been established in Cannon [8] and Vuik [45]. Before treating some numerical solution techniques, the exact solution for a simple one-dimensional case will be given. This solution, also called a self-similar solution, shows the qualitative behavior of this kind of problem.

13.2 An exact (self-similar) solution for an unbounded region

With a *self-similar* solution we mean a solution for the temperature that depends on a pair of x and t, e.g. $T = T(\frac{x}{\sqrt{t}})$. To get some quick insight into the behavior of the solution of the Stefan problem, we present a self-similar solution of the Stefan problem for an unbounded interval: $x \in (0, \infty)$. The solution is for a one-dimensional case and mimics the actual behavior of the solution of the Stefan problem (13.1.1)-(13.1.6) especially in the early stages when the temperature at $\partial\Omega_3$ has not been effected by the freezing front yet. Hence it serves as a test-problem, which can be used to check the behavior of the results from numerical solutions. We require that the solution for the temperature, $T = T(x,t)$ is bounded, continuous and monotone in x and t and hence its derivative with respect to x vanishes as $x \to \infty$. For this purpose we search bounded solutions of the following problem, with the same symbols as in the previous section:

$$\rho c \frac{\partial T}{\partial t} = \lambda \frac{\partial^2 T}{\partial x^2} \tag{13.2.1}$$

$$T(x,0) = T_0 \tag{13.2.2}$$

$$T(0,t) = T_* \tag{13.2.3}$$

$$S(0) = 0 \tag{13.2.4}$$

$$\rho_1 L \frac{dS}{dt} = \lambda_1 \frac{\partial T_1}{\partial x} - \lambda_2 \frac{\partial T_2}{\partial x}, \qquad x = S(t) \tag{13.2.5}$$

$$T(S(t),t) = T_s \tag{13.2.6}$$

The above problem, equations (13.2.1)-(13.2.6) admits self-similar solutions in the form $T = T(\frac{x}{\sqrt{t}})$ and $S = k\sqrt{t}$. Explicit formulas for T, S, k can be determined using procedures given in the text-books [9, 13] and the very early paper of Neumann [47].

In Figure 13.2 the temperature profile during freezing of water at consecutive times is shown. The initial water temperature is $T_0 = 2^o = 275K$, freezing temperature, $T_s = 0^o = 273K$. The temperature at $x = 0$ is maintained at $T_* = -17^o = 250K$ at all stages. Further physical data are given in Table 13.1. Figure 13.3 displays the interface position (ice thickness) as a function of time for various temperatures at $x = 0$ (i.e. T_*).

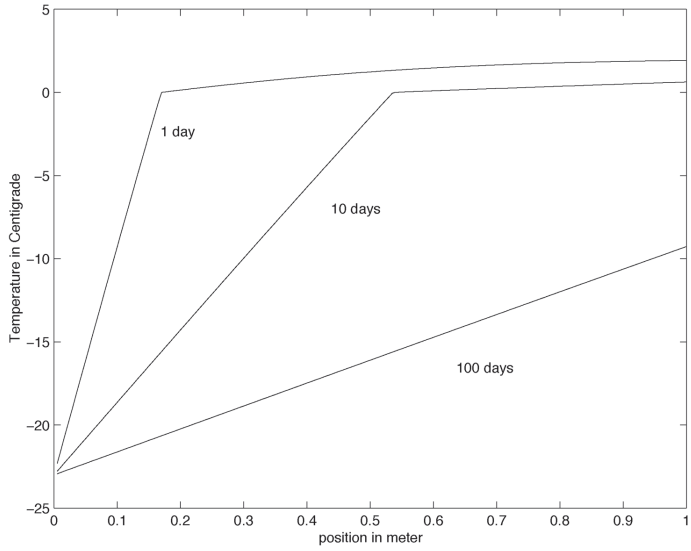


Figure 13.2: The temperature profile of freezing water at consecutive times for T_* = −23 °C. The rate factor k is obtained from the numerical solution, see [9].

Table 13.1. Input data.

Physical quantity	Value	Si-Unit
T_*	250	K
T_s	273	K
T_0	275	K
λ_1	2.2	W/(mK)
λ_2	0.55	W/(mK)
L	33400	J/kg
ρ_1	920	kg/m^3
ρ_2	1000	kg/m^3

13.3 Numerical methods

Various numerical techniques are known to solve Stefan problems. For an overview we refer for instance to the book of Crank [13], where roughly the following methods are distinguished: Front tracking and Fixed domain methods. The main feature of the Fixed domain methods is that the front is defined implicitly and the discretization mesh does not move. Whereas the front is followed explicitly and the mesh moves with the interface in the Front tracking methods. Besides fixed domain and front tracking methods there exists also a hybrid form where a fixed basis grid is used, which in each time step is locally adapted to the front. After the time step the local adaptation is removed.

13.3.1 Moving grid methods

A Front tracking method explicitly tracks the position of the interface. The equations for the temperature are solved using a discretization method in both subdomains and the discrete temperature gradients are substituted into the rate equa-

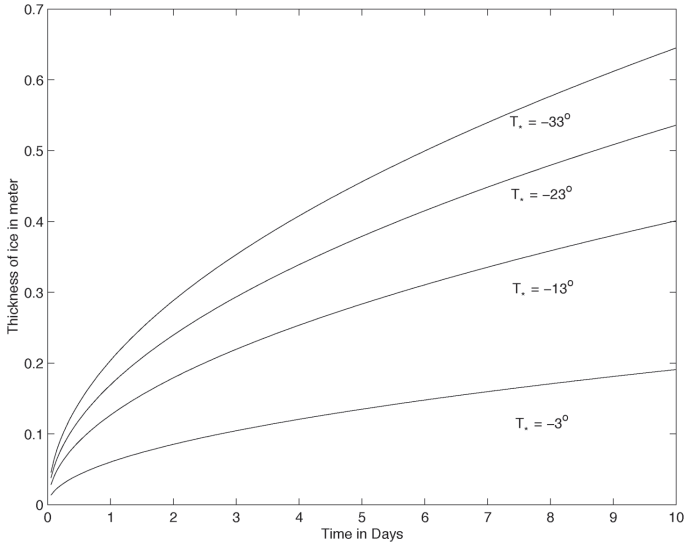


Figure 13.3: The thickness of ice as a function of time for various values of T_*.

tion (see Equation (13.1.6) or (13.2.5)). First we treat a one-dimensional case to illustrate some of the numerical problems that arise in Front tracking methods. Subsequently, we briefly treat a two-dimensional example.

A one-dimensional example

We consider the same situation as in Equations (13.2.1)-(13.2.6) but now the interval in x is bounded, say $x \in [0, 1]$. We again apply a no-flux boundary condition at $x = 1$, i.e.:

$$\lambda_2 \frac{\partial T}{\partial x} = 0, \qquad x = 1. \tag{13.3.1}$$

For now, we assume that the interface position $S(t)$ is given within the interval $0 < S(t) < 1$, and hence Ω_1 and Ω_2 exist and are nonempty sets. Furthermore, we assume that the temperature profile is given at $t = 0$. We use the method of lines to solve the problem. First we deal with the spatial discretization, where we divide Ω_1 and Ω_2 into N grid nodes, i.e.

$$\Delta x_1 = \frac{S(t)}{N}, \qquad \Delta x_2 = \frac{1 - S(t)}{N}. \tag{13.3.2}$$

Note that the grid spacings Δx_1 and Δx_2 depend on time. The index i refers to the grid node:

$$T_{1,i} := T_1(i\,\Delta x_1), \tag{13.3.3}$$

$$T_{2,i} := T_2(S(t) + i\,\Delta x_2). \tag{13.3.4}$$

Further, we use the boundary conditions for $t > 0$

$$T_{1,0} = T_*, \qquad T_{1,N} = T_s, \qquad \text{for } \Omega_1, \tag{13.3.5}$$

$$T_{2,0} = T_s, \quad \lambda_2 \frac{T_{2,N+1} - T_{2,N-1}}{2\,\Delta x_2} = 0, \quad \text{for } \Omega_2. \tag{13.3.6}$$

Note that we add an extra grid point $N + 1$ for Ω_2 to maintain an accuracy of $O(\Delta x_2^2)$ for the global discretization over Ω_2.

Exercise 13.3.1 *Write down the equations from the discretization of the diffusion equation in both subdomains (ice and water) where we use an implicit Euler time integration.*

Since the interface and mesh movement are not incorporated yet into the discretization, we write tildes above the unknowns.

Exercise 13.3.2 *To guarantee a second order spatial accuracy globally, we introduce ghostpoints near the moving interface $S(t)$. Derive expressions for the ghostpoints $\tilde{T}^{j+1}_{1,N+1}$ and $T^{j+1}_{2,-1}$. Hint: treat the discretization of the interface like an internal point in the domain.*

The interface S moves, and the speed is approximated by

$$\rho_1 L \frac{dS}{dt} \approx \rho_1 L \frac{S^{j+1} - S^j}{\Delta t} \approx \lambda_1 \frac{\tilde{T}^{j+1}_{1,N+1} - \tilde{T}^{j+1}_{1,N-1}}{2\,\Delta x_1} - \lambda_2 \frac{\tilde{T}^{j+1}_{2,1} - \tilde{T}^{j+1}_{2,-1}}{2\,\Delta x_2}, \tag{13.3.7}$$

where we use the expressions that were obtained in Exercise 13.3.2. From above expression S^{j+1} is easily computed. We update the grid by computing $\Delta x_1 = \dfrac{S^{j+1}}{N}$ and $\Delta x_2 = \dfrac{1 - S^{j+1}}{N}$, further note that at all stages we have

$$T^{j+1}_{1,N} = T_S = T^{j+1}_{2,0}. \tag{13.3.8}$$

The temperature at the new grid nodes are obtained using linear interpolation. Let $x^j_{k,i}$ denote the position of the i^{th} grid point in Ω_k at time step j, i.e.

$$x^j_{k,i} = \begin{cases} i\,\Delta x_1, \text{ for } k = 1 \\ \\ S^j + i\,\Delta x_2, \text{ for } k = 2 \end{cases}, \tag{13.3.9}$$

and the tildes represent the temperatures that have been computed from the discretization of the heat equation in both regions, then

$$T^{j+1}_{k,i} = \tilde{T}^{j+1}_{k,i} + \frac{\partial \tilde{T}^{j+1}_k}{\partial x}|_i (x^{j+1}_{k,i} - x^j_{k,i}) \approx \tilde{T}^{j+1}_{k,i} + \frac{\tilde{T}^{j+1}_{k,i+1} - \tilde{T}^{j+1}_{k,i-1}}{2\Delta x_k}(x^{j+1}_{k,i} - x^j_{k,i}) \tag{13.3.10}$$

Now the temperature has been updated correctly. Using this interpolation, the time integration can be rewritten, after division by Δt of Equation (13.3.10), by

$$\frac{T^{j+1}_{k,i} - \tilde{T}^{j+1}_{k,i}}{\Delta t} - \frac{x^{j+1}_{k,i} - x^j_{k,i}}{\Delta t} \frac{\tilde{T}^{j+1}_{k,i+1} - \tilde{T}^{j+1}_{k,i-1}}{2\Delta x_k} = 0, \qquad k \in \{1, 2\}. \tag{13.3.11}$$

Equation (13.3.11) represents a discrete convection equation, with mesh velocity

$$v_{\text{mesh}} = \frac{x^{j+1}_{k,i} - x^j_{k,i}}{\Delta t} \tag{13.3.12}$$

that is solved explicitly and using a central discretization. Since the temperature is smooth within Ω_1 and Ω_2, central discretization does not produce any unphysical wiggles (see Chapter 1). However, explicit time integration gives conditional stability: $\dfrac{u_{\text{mesh}} \Delta t}{\Delta x} < 1$ (see Chapter 1). For most practical situations, this is not so limiting since u_{mesh} is usually small. We end the one-dimensional description with some general remarks:

1. We have described the moving-grid method for cases when both Ω_1 and Ω_2 exist. In practice one assumes at $t = 0$ that ice is already present in a very thin layer. When the ice layer is small, one can use fewer grid nodes within Ω_1 and similarly fewer grid nodes within Ω_2 when Ω_2 almost vanishes. This implies that at certain times the grid needs to be regenerated.

2. In the above presentation of the numerical method, the determination of the position of the interface is rather inaccurate. One can use an iterative Trapezium method to improve the accuracy of the position of the interface. We omit the treatment here and refer the interested reader to [43].

A two-dimensional example
For the illustration of the two-dimensional solution of a Stefan problem, we consider an 'ice-disc' in a rectangular domain filled with water. For the sake of illustration we assume that the temperature in the 'ice-disc' is constant in space. Due to a relatively high water temperature the ice starts to melt and the circumference of the 'ice-circle' decreases. This gives a change of topology for the elements attached to the moving boundary. Mathematically we deal with the following problem:

$$\rho_2 c_2 \frac{\partial T}{\partial t} = \lambda_2 \triangle T, \qquad \mathbf{x} \in \Omega \tag{13.3.13}$$

$$T = T_0, \qquad \mathbf{x} \in \Omega, \tag{13.3.14}$$

$$T = T_s, \qquad \mathbf{x} \in S(t), \tag{13.3.15}$$

$$T = T_*, \qquad \mathbf{x} \in \Omega, \tag{13.3.16}$$

$$\frac{\partial T}{\partial n} = 0, \qquad \mathbf{x} \in \bigcup_{k=2}^{4} \partial\Omega_k \tag{13.3.17}$$

$$\rho_1 L v_n = -\lambda_2 \frac{\partial T}{\partial n}, \qquad \mathbf{x} \in S(t). \tag{13.3.18}$$

The equations in (13.3.13)-(13.3.18) are solved using a Finite Element method in both subdomains and the discrete temperature gradients are substituted into the rate Equation (13.3.18) to obtain the speed and position of the grid nodes at the interface as a function of time. An unstructured grid is used for the Finite Element discretization.

Exercise 13.3.3 *Derive a Finite Element formulation for the differential equation for the temperature with boundary and initial conditions in the two-dimensional heat diffusion problem* (13.3.13)-(13.3.17)*, where we have as essential condition $T = T_s$ at the interface S.*

Figure 13.4 shows an example of the initial mesh in subdomain Ω_2. In this subdomain moves the circular inclusion Ω_1. The interface is approximated by a spline. Frequently the number of the nodes at the interface and in the subdomains is kept constant. However, this is not necessary. Once the interface has been moved, the position of the mesh points inside the subdomains are adapted. The value of the temperature at the new positions of the grid nodes are unknown. To obtain these values, one can either use interpolation or a correction for the displacement. Since interpolation is rather expensive, a correction, taking into account the velocity of the grid nodes, is recommended. When we compute the time-derivative between the old and new points, a material derivative, as in fluid mechanics, is used:

$$\frac{dT}{dt} = \frac{\partial T}{\partial t} + \mathbf{u}_{\text{mesh}} \cdot \nabla T, \tag{13.3.19}$$

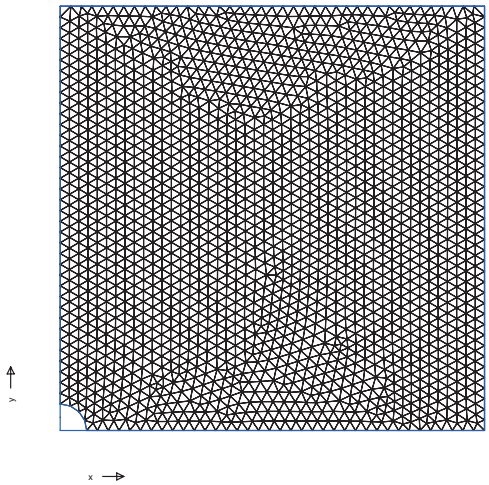


Figure 13.4: The initial mesh in Ω_2.

with the mesh-velocity $\mathbf{u}_{\text{mesh}} = \dfrac{d}{dt}\mathbf{x}$.

The temperature T is determined from

$$\frac{dT}{dt} - \mathbf{u}_{\text{mesh}} \cdot \nabla T = \lambda \Delta T, \text{ for all interior mesh points.} \tag{13.3.20}$$

Above treatment is known as the Arbitrary Lagrangian Eulerian (ALE) method and is very common in fluid dynamics. For a complete description of the Front tracking method for a one-dimensional case we refer to Murray and Landis [28], for two dimensions we refer to Segal et al [19]. The moving mesh is shown in Figure 13.5. During the adaptation of the mesh the quality of the mesh must be checked. As the length of the interface may change, the angles at the mesh-points change as well especially near the moving interface. Further due to the interface movement elements within both subdomains become either stretched or contracted. To avoid ill-shaped elements, remeshing may be necessary. Remeshing is expensive since the values of the temperature at the new mesh points have to be determined using interpolation and a new mesh must be generated.

Figure 13.6 shows an example where the mesh has not been checked at the boundary. Here the elements at the interface have become stretched and hence ill-shaped. An example of remeshing is presented in Figure 13.7, where we display the mesh at a point in time when the subdomain Ω_1 has grown significantly. It can be seen, also from Figure 13.4, that initially there were 5 grid nodes at the moving boundary. For larger values of time, when the subdomain Ω_1 has grown, the number of grid nodes at the interface has increased, see Figure 13.7. Let β be the angle of the elements in the domain, then remeshing has been applied on the basis of the following criterion:

$$\beta_{\min} \leq \beta \leq \beta_{\max},$$

where $\beta_{\min} = 10^o$ and $\beta_{\max} = 120^o$. Whenever some element in the domain does not satisfy this criterion, the domain is remeshed.

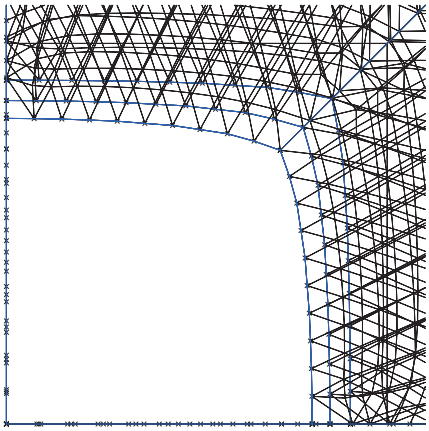


Figure 13.5: The moving mesh of Ω_2. The blue lines represent the moving boundary after t = 0.

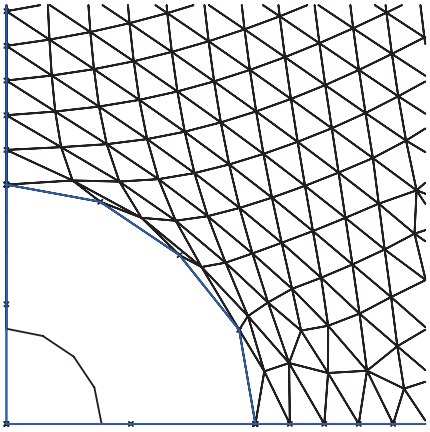


Figure 13.6: The mesh after significant growth of subdomain Ω_1 with the original mesh topology. The blue and black lines respectively represent the moving interface for $t > 0$ and t = 0.

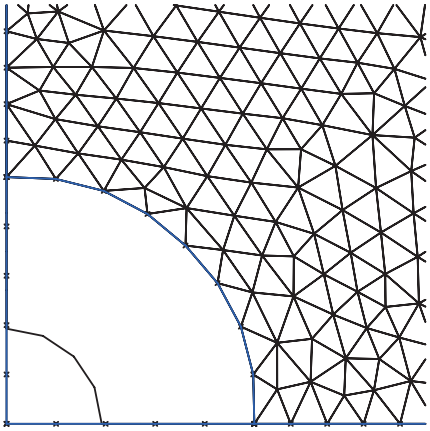


Figure 13.7: The mesh after significant growth of subdomain Ω_1 with mesh topology after remeshing. The blue and black lines respectively represent the moving interface for t $>$ 0 and t = 0. Remeshing is applied whenever one of the angles of the elements falls outside of the range [10,120] degrees.

It will be clear that moving grid methods are rather expensive due to remesh-ing and difficult to program, especially in three dimensional problems. A great advantage, however, is that the interface is part of the boundary of the elements and therefore interface conditions can be satisfied easily.

13.3.2 A fixed domain method: the level set method

As an example of a Fixed grid method we consider the level set method. The method does not track the interface explicitly. The method is conceptually less self-evident than the moving grid method. This is mainly because of the intro-duction of the *level set* function, which is sometimes also referred to as a 'pseudo-temperature'. The method is very powerful especially for three dimensions. The level set method was first introduced by Osher and Sethian [29]. First we outline the level set method in a general way and subsequently we describe an application to a 2D-Stefan problem. The application is for solidification and melting as studied by Chen et al [10]. The idea behind the level set function is as follows: We define an extra unknown, the *pseudo-temperature*. This unknown is only meant to define the interface implicitly. The sign of the pseudo-temperature determines in which phase or subdomain a node is at particular instant of time. Furthermore, the interface position coincides with the zero level of the pseudo-temperature and hence this position is tracked implicitly. Since the interface is convected by some velocity, one can derive a convection equation for the pseudo-temperature, which is solved together with the original PDE. The principles and the concept pseudo-temperature are outlined in the coming subsections.

Application to the Stefan problem
Consider, as before, the heat equation in a solid and a liquid phase Ω_1 and Ω_2 with interface $S(t)$:

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot (\lambda \nabla T), \qquad \mathbf{x} \in \Omega, \tag{13.3.21}$$

$$\rho_1 L v_n = \lambda_1 \frac{\partial T_1}{\partial n} - \lambda_2 \frac{\partial T_2}{\partial n}, \qquad \mathbf{x} \in S(t), \tag{13.3.22}$$

with similar initial and boundary conditions as in Equations (13.1.1)-(13.1.6). The complete region is used as computational domain. We define the *level set* function, which is negative in one phase and positive in the other phase, ϕ. At time $t = 0$, ϕ is defined as

$$\phi = \begin{cases} -d(\mathbf{x}), & \text{for } \mathbf{x} \in \Omega_1(0) \\ 0, & \text{for } \mathbf{x} \in S(0) \\ +d(\mathbf{x}), & \text{for } \mathbf{x} \in \Omega_2(0) \end{cases}, \tag{13.3.23}$$

where the most important feature of ϕ is $\phi(\mathbf{x}, 0) = 0$ on $S(0)$ and that $\phi < 0$ on Ω_1 and $\phi > 0$ on Ω_2. Hence the level set function ϕ is used to indicate in which phase a specific gridnode is. The level set function ϕ is also sometimes referred to as a 'pseudo-temperature'. We prescribe the level set function ϕ at $t = 0$ as a signed distance function: the function $d \geq 0$ denotes the minimal distance between a certain point $\mathbf{x} \in \Omega$ and the boundary $S(t)$. Also for $t > 0$ we require $\phi = 0$ at $S(t)$, $\phi < 0$ in Ω_1 and $\phi > 0$ in Ω_2. From the definition of the function ϕ follows that the interface can be determined for each given ϕ:

$$S(t) = \{(x, y) \in D : \phi(x, y, t) = 0\} \text{ for } t \geq 0. \tag{13.3.24}$$

Since $\phi(x(t), y(t), t) = 0$ at the interface, it follows that the total derivative with respect to time vanishes at the moving boundary

$$\frac{D}{Dt}\phi = \phi_t + \nabla\phi \cdot \mathbf{r}' = 0 \text{ for } \mathbf{r} \in S(t).$$

Here $\mathbf{r}'(t)$ represents the speed of a point $\mathbf{r}(t) \in S(t)$ at the interface, where $\phi = 0$. This point coincides with the moving boundary and hence has the same speed as the moving boundary, so

$$\mathbf{r}'(t) = [\lambda \nabla T] := \lambda_1 \nabla T_1 - \lambda_2 \nabla T_2, \text{ for } \mathbf{x} \in S(t). \tag{13.3.25}$$

This implies that the total derivative for points on the interface with respect to time t (where $\phi = 0$) can be written as

$$\phi_t + \nabla\phi \cdot [\lambda \nabla T] = 0 \text{ for } \mathbf{x} \in S(t). \tag{13.3.26}$$

Above equation only holds at the moving interface. In order to have ϕ as a signed continuous function on the whole domain Ω, it must be prescribed in other positions of Ω as well. We do this by the use of the following PDE

$$\phi_t + \mathbf{u} \cdot \nabla\phi = 0 \text{ for } \mathbf{x} \in \Omega, \tag{13.3.27}$$

where the vector function $\mathbf{u} : \mathbb{R}^2 \times R_0^+ \to \mathbb{R}^2$, will be defined as a continuous extension of $[\lambda \nabla T]$ over Ω, i.e.

$$\mathbf{u} \in U := \{\mathbf{u} \in C(\Omega) : \mathbf{u} = [\lambda \nabla T] \text{ for } \mathbf{x} \in S(t)\} \text{ for } t > 0. \tag{13.3.28}$$

We proceed to construct a continuous extension for $\mathbf{u}$ near $S(t)$, by which we mean that $\mathbf{u}$ is continuous at points near the interface position. This is based on the principles which are described in [10].

For the moment we assume that we have an initial value for $\mathbf{u}$ for points not on S. Furthermore, $\mathbf{u}$ is prescribed on S by the initial value of T. This implies that if the components of $\mathbf{u} = (u_1, u_2)$ satisfy a first order hyperbolic (convection) equation, then $\mathbf{u}$ is continuous near $S(t)$. This convection problem is well-posed

as long as the prescribed value is upwind from points away from $S(t)$ where a 'boundary condition' is imposed. Therefore, we set

$$\frac{\partial u_1}{\partial \tau} + sign(\phi\phi_x)\frac{\partial u_1}{\partial x} = 0, \tag{13.3.29}$$

$$\frac{\partial u_2}{\partial \tau} + sign(\phi\phi_y)\frac{\partial u_2}{\partial y} = 0, \tag{13.3.30}$$

$$\text{subject to } u_1 = \left[\lambda\frac{\partial T}{\partial x}\right], \quad u_2 = \left[\lambda\frac{\partial T}{\partial y}\right], \quad \text{for } \mathbf{x} \in S(t), \tag{13.3.31}$$

then $\mathbf{u}$ points away from $S(t)$ and a well-posed definition of $\mathbf{u}$ is obtained. In the above equations τ is a pseudotime, since the reason for the use of the above equation is just to extend the velocity continuously near the interface. Further, for $(x, y) \in S(t)$, implying $\phi = 0$ and hence $sign(\phi\phi_x) = 0$ and $sign(\phi\phi_y) = 0$, we have

$$\frac{\partial}{\partial \tau}\mathbf{u} = 0 \Rightarrow \mathbf{u} = [\lambda \nabla T] \text{ for } \mathbf{x} \in S(t). \tag{13.3.32}$$

In principle we have given a full system of PDE's to solve the Stefan problem using the level set method. The level set function ϕ defines the position of the interface. Further, defining it as a signed distance function, it satisfies nice monotonicity properties and the closer to zero a particular gridnode value is, the closer the corresponding grid node is to the interface. It is particularly important to have information whether a grid node is in subdomain Ω_1 or Ω_2 when the coefficients in the heat equation are determined. Therefore, the sign of the level set function is crucial. We want to have this information without having to track the position of the moving interface explicitly like in the moving grid method. Furthermore, for boundary conditions at the moving interface it is important to know whether a gridnode is a neighbor of the interface.

One-dimensional implementation

As an example of the application of the level set method, we consider the 1D equation

$$\rho c\frac{\partial T}{\partial t} = \lambda\frac{\partial^2 T}{\partial x^2}, \quad x \in \Omega \tag{13.3.33}$$

$$\rho_1 L\frac{dS}{dt} = \lambda_1\frac{\partial T_1}{\partial x} - \lambda_2\frac{\partial T_2}{\partial x}, \quad x = S(t), \tag{13.3.34}$$

$$T = T_*, \quad x = 0, \tag{13.3.35}$$

$$T = T_s, \quad x = S(t) \tag{13.3.36}$$

$$\frac{\partial T}{\partial x} = 0, \quad t > 0, \tag{13.3.37}$$

$$T = T_0, S = 0, \quad t = 0. \tag{13.3.38}$$

We describe the solution procedure at each time-step. We suppose here that ϕ^j, T^j, u^j are known. Let j be the time-step index, $t_j = j\Delta t$ and $h = \frac{1}{N}$ be the grid-spacing. At each time-step we first solve the temperature field in both subdomains using Finite Differences (see Figure 13.8), to obtain

$$\rho_2 c_2\frac{T_i^{j+1} - T_i^j}{\Delta t} = \lambda_2\frac{T_{i+1}^{j+1} - 2T_i^{j+1} + T_{i-1}^{j+1}}{h^2}, \quad i \in \{1, \ldots, p-2\}. \tag{13.3.39}$$

$$\rho_1 c_1\frac{T_i^{j+1} - T_i^j}{\Delta t} = \lambda_1\frac{T_{i+1}^{j+1} - 2T_i^{j+1} + T_{i-1}^{j+1}}{h^2}, \quad i \in \{p+1, \ldots, n\}, \tag{13.3.40}$$

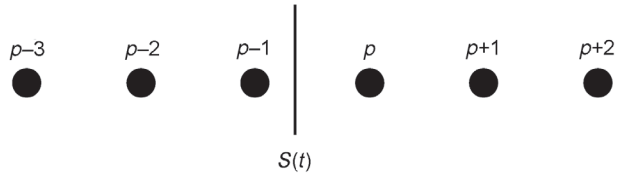


Figure 13.8: The gridpoints in the neighborhood of the moving interface.

For the gridpoints near the interface, we use the fact that ϕ represents a signed distance function. To guarantee a $O(h^2)$ accuracy we use ghost points at both sides of the interface. Let $\hat{T}_p$ and $\hat{T}_{p-1}$ respectively represent ghost points belonging to $\Omega_1 := \{x \in \mathbb{R} : 0 < x < S(t)\}$ and $\Omega_2 := \{x \in \mathbb{R} : S(t) < x < 1\}$. We treat the boundary condition at the right-hand side of the interface, in Ω_2, and the treatment for the boundary condition at the side of the subdomain Ω_1 is left for the reader. For $i = p$, we then have

$$\rho_2 c_2 \frac{T_p^{j+1} - T_p^j}{\Delta t} = \lambda_2 \frac{T_{p+1}^{j+1} - 2T_p^{j+1} + \hat{T}_{p-1}^{j+1}}{h^2} + O(h^2). \tag{13.3.41}$$

The value for $\hat{T}_{p-1}$is computed from a Taylor expansion around $\phi = 0$, i.e. the interface S:

$$\hat{T}_{p-1} = T_S + (x_{p-1} - S)\frac{\partial T}{\partial x}|_S + O(h^2). \tag{13.3.42}$$

The position of the interface, S, is obtained as the zero level of the function ϕ. Having two values of ϕ with opposite sign, it is clear that the interface is between the two corresponding positions. The interface position is then obtained by interpolation using these points. The derivative of T at S is obtained from a Taylor expansion around ϕ_p:

$$\frac{\partial T}{\partial x}|_S = \frac{\partial T}{\partial x}|_{x_p} - (x_p - S)\frac{\partial^2 T}{\partial x^2}|_{\phi_p} + O(h^2) \tag{13.3.43}$$

$$= \frac{T_{p+1} - \hat{T}_{p-1}}{2h} - (x_p - S)\frac{T_{p+1} - 2T_p + \hat{T}_{p-1}}{h^2} + O(h^2) \tag{13.3.44}$$

Equation (13.3.44) is substituted into Equation (13.3.42) where an expression for $\hat{T}_{p-1}$ is obtained (after dropping the $O(h^2)$-terms) and subsequently substituted into equation (13.3.41). A similar procedure is done for the left-hand side of the interface, and the matrix equation is subsequently solved. We need the gradients of the temperature at the interface for the velocity of the interface (see the equation (17.49). When we use for the discretization of the gradient the one-sided discretization formulas

$$\frac{\partial T_2}{\partial x}(S(t), t) = \frac{T_p - T_S}{x_p - S} \tag{13.3.45}$$

$$\frac{\partial T_1}{\partial x}(S(t), t) = \frac{T_S - T_{p-1}}{S - x_{p-1}}, \tag{13.3.46}$$

for the determination of the gradients, it can be seen that once $S \to x_p$ or $S \to x_{p-1}$ division by zero results. This causes the jumps in the velocity of the interface as shown in Figure 13.11. Therefore, we use for the gradients at the interface at both

sides:

$$\frac{\partial T_2}{\partial x} = \frac{T_p - T_S}{x_p - S}\frac{x_p - S}{h} + (1 - \frac{x_p - S}{h})\frac{T_{p+1} - T_p}{h} \tag{13.3.47}$$

$$= \frac{T_p - T_S}{h} + (1 - \frac{x_p - S}{h})\frac{T_{p+1} - T_p}{h} \tag{13.3.48}$$

$$\frac{\partial T_1}{\partial x} = \frac{T_S - T_{p-1}}{S - x_{p-1}}\frac{S - x_{p-1}}{h} + (1 - \frac{S - x_{p-1}}{h})\frac{T_{p-1} - T_{p-2}}{h} \tag{13.3.49}$$

$$= \frac{T_S - T_{p-1}}{h} + (1 - \frac{S - x_{p-1}}{h})\frac{T_{p-1} - T_{p-2}}{h} \tag{13.3.50}$$

This computation of gradients is called the *weighted gradients approach* . The result using the weighted gradient approach is represented by the blue curve in Figure 13.11. Using both gradients, we determine the velocity of the interface. Given the interface velocity, we solve the following equation to obtain the velocity field over the whole domain, with the known level set function ϕ:

$$\frac{\partial u}{\partial \tau} + sign(\phi\phi_x)\frac{\partial u}{\partial x} = 0, \quad x \in (0,1) \tag{13.3.51}$$

$$u(S) = \lambda_1 \frac{\partial T_1}{\partial x}(S) - \lambda_2 \frac{\partial T_2}{\partial x}(S) \tag{13.3.52}$$

Here τ represents a pseudotime since the above equation just artificially extends the velocity continuously. After the update of u we compute the update of the signed distance function from solution of

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} = 0. \tag{13.3.53}$$

Solution of Equations (13.3.53) and (13.3.51), (13.3.52) is done using upwind differences. Note that in the solution of Equation (13.3.53) the discretization of $\frac{\partial \phi}{\partial x}$ depends on the sign of u. The function ϕ was initially chosen to be a signed distance function. However, at the course of the iteration process, the function ϕ looses this property. This is not so bad in general, since only smoothness of ϕ is necessary in all the steps taken until now. Hence, if ϕ is a signed distance function at all time steps, then ϕ is continuous and then all the operations until now are allowed. Therefore, one often requires ϕ to be a distance function although this is not necessary. However, often it is desirable to have ϕ as a signed distance function if local curvatures of the interface are used. This is often done in relation to surface tension. Furthermore, having ϕ as a distance function, guarantees that ϕ is continuous, which is a necessary requirement. Therefore, we iterate

$$\frac{\partial \phi}{\partial \tau} = sign(\phi_0)(1 - |\frac{\partial \phi}{\partial x}|). \tag{13.3.54}$$

In order to get a signed distance function it is necessary that $|\frac{\partial \phi}{\partial x}| = 1$. To satisfy this we consider Φ as the stationary solution of (13.3.54). This equation is solved using the time-step as a kind of convergence parameter. Furthermore, τ is a pseudotime, which is artificial for the convergence towards a stationary solution. We show some results for 100 gridpoints for the freezing of water in Figures 13.9, 13.10. Initially, we have an ice-layer of 3 gridpoints (0.03 m) and we apply a temperature $T_* = -23°C$. The temperature-profile after 1 and 3 days is shown in Figure 13.9. The position of the interface is shown in Figure 13.10. A good correspondence is observed with the analytical solution (see Figure 13.2 and 13.3). However, since the water area is bounded $(S(t), 1)$ for the numerical solution, the ice-thickness

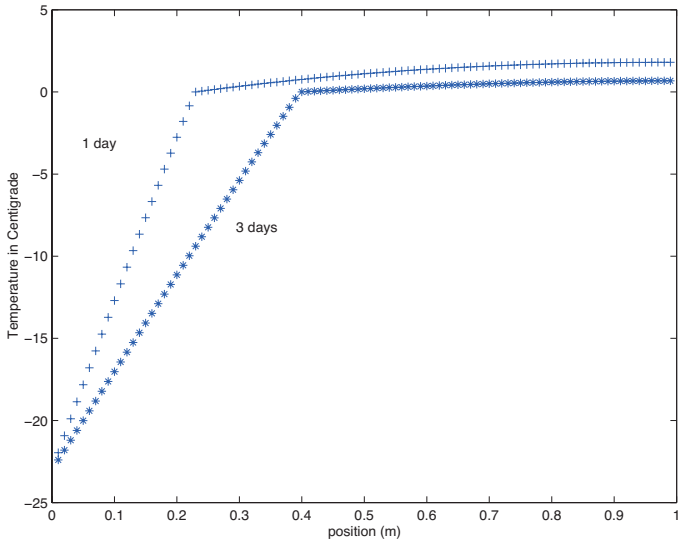


Figure 13.9: Temperature profiles of freezing water at subsequent times. The data have been taken from Table 13.1. The temperature at x = 0 is −23 °C.

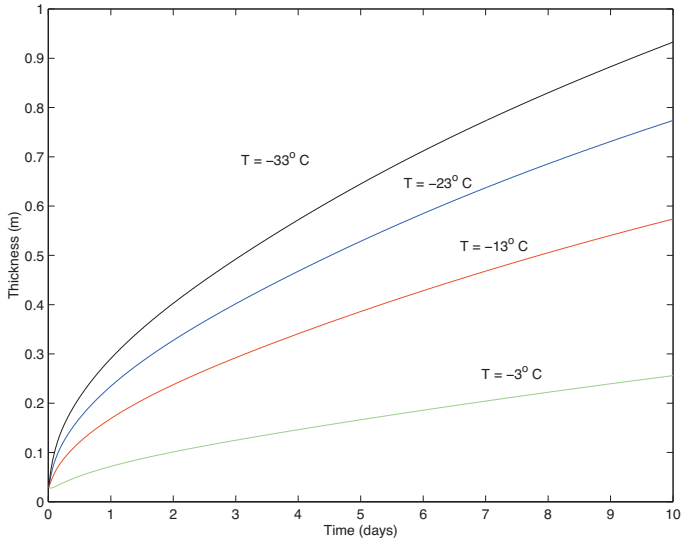


Figure 13.10: The ice-thickness as a function of time for different temperatures at x = 0. Further data have been taken from Table 13.1.

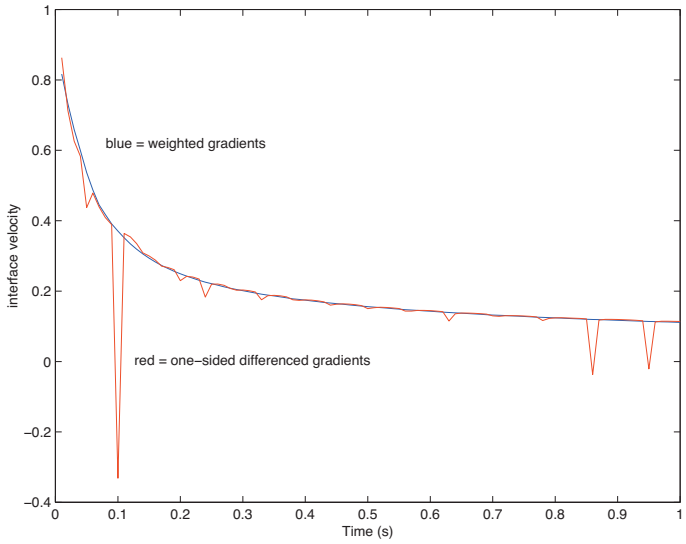


Figure 13.11: The velocity of the interface as a function of time for T = −23 °C at x = 0. Further data are from Table 13.1. The red curve represents the calculation for the one-sided gradients.

evolves faster than for the analytical solution where the water area is unbounded $(S(t), \infty)$. Furthermore, we show the velocity of the interface as a function of time in Figure 13.11. The smooth curve represents the use of the weighted gradients.

13.3.3 Other applications of Stefan problems

Stefan problems occur also, among others, during solidification of metals or in the solid state. Here typical problems of phase transformations involving different lattices (crystals) take place. Some examples are the dissolution or growth of particles in ferrous or non-ferrous alloys or the phase-transformation of ferrite to austenite in steels. Both processes occur during production and optimization of high quality metals and alloys. Models can be found in the book of Visintin [44]. Furthermore, similar numerical techniques are used to solve free boundary problems. We mention the seepage of water through a porous dam as an example of a free boundary problem. This free boundary problem has been described in Crank [13]. Furthermore, it should be noted that Stefan problems are also solved by different methods, such as the phase-field approach, enthalpy method and the method of variational inequalities. These methods are beyond the scope of the book and their principles can be found in textbooks as [44], [13].

13.4 Summary of Chapter 13

In this chapter some examples of Stefan problems are formulated. A moving grid method to solve the Stefan problem is described. This method is conceptually simple, however for cases where several moving boundaries merge, the method fails. Further, the interpolation that has to be applied can be expensive. Next, a fixed grid method, the level set method, is described as a conceptually less obvious method. Here the interface was taken into account in an implicit way and hence the

determination of the exact position of the interface may be less accurate. However, the method is successfully used in cases where several interfaces merge or come close together. Both methods have its benefits and disadvantages. Further, the class of so-called self-similarity solutions have been referred to as a tool to validate the behavior of solutions obtained from numerical methods in a qualitative way.

Bibliography

[1] R.A. Adams. *Sobolev Spaces*. Academic Press, New York, 1975.

[2] Robert A. Adams. *Calculus, a complete course. Fifth Edition.* Addison Wesley Longman, Toronto, 2003.

[3] R. Aris. *Vectors, Tensors and the Basic Equations of Fluid Mechanics*. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962. Reprinted, Dover, New York, 1989.

[4] J. Bear. *Dynamics of Fluids in Porous Media*. American Elsevier publishing company, New York, 1972.

[5] E.K. Blum. *Numerical Analysis and Computation: Theory and Practice*. Addison-Wesley Publishing Company, Reading, Mass, 1972.

[6] A.N. Brooks and T.J.R. Hughes. Stream-line upwind/Petrov Galerkin formulstion for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equation. *Comp. Meth. Appl. Mech. Eng.*, 32:199–259, 1982.

[7] R.L. Burden and J.D. Faires. *Numerical analysis*. Brooks/Cole, Pacific Grove, 2001.

[8] J.R. Cannon. *The one-dimensional heat equation*. Addison-Wesley Publishing company, Menlo park, California, U.S.A., 1984.

[9] H.S. Carslaw and J.C. Jæger. *Conduction of heat in solids*, volume 2. Clarendon Press, Oxford, 1988.

[10] S. Chen, B. Merriman, S. Osher, and P. Smereka. A simple level-set method for solving stefan problems. *J. Comp. Phys.*, 135:8–29, 1997.

[11] Ph.G. Ciarlet. *The Finite Element Method for Elliptic Problems.* North-Holland, Amsterdam, 1978.

[12] R. Courant and D. Hilbert. *Methods of Mathematical Physics, Vol. 2. Partial Differential Equations.* Interscience, New York, 1989.

[13] J. Crank. *Free and Moving Boundary Problems*. Clarendon Press, Oxford, 1984.

[14] E. Cuthill and J. McKee. Reducing the bandwidth of sparse symmetric matrices. In *Proc. 24th Nat. Conf. Assoc. Comput. Mech.*, pages 1–69, New York, 1969. ACM publ.

[15] C. Cuvelier, A. Segal, and A.A. van Steenhoven. *Finite Element Methods and Navier-Stokes Equations*. Reidel Publishing Company, Dordrecht, Holland, 1986.

[16] L.C. Evans. *Partial Differential Equations*. American Mathematical Society, 2010.

[17] A. George and J.W.H. Liu. *Computer Solution of Large Sparse Positive Definite Systems*. Prentice-Hall, Engelwood Cliffs, New Jersey, (USA), 1981.

[18] Wolfgang Hackbusch. *Multi-grid methods and applications*. Springer, Berlin, 2003 (1986).

[19] C.W. Hirt, A.A. Amsden, and J.L. Cook. An Arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14:227–253, 1974.

[20] I. Holand and K. Bell eds. *Finite Element Methods in Stress Analysis*. Tapir, 1969, Trondheim, Norway, 1969.

[21] B.M. Irons and A. Razazaque. Expericience with the patch test for convergence of finite elements. In A.K. Aziz, editor, *The mathemattical foundations of the finite element method with applications to partial differential equations*, pages 557–587, New-York, 1972. Academic Press.

[22] E. Kreyszig. *Introductory Functional Analysis with Applications*. Wiley, New York, 1989.

[23] J.D. Lambert. *Numerical methods in ordinary differential equations*. John Wiley, Englewood Cliffs, 1991.

[24] David C. Lay. *Linear Algebra and its applications*. Addison Wesley, New York, 1993.

[25] R.J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser, Basel, 1992.

[26] A.R. Mitchell and D.F. Griffiths. *The Finite Difference Method in Partial Differential Equations*. Wiley, Chichester, 1994.

[27] A.R. Mitchell and R. Wait. *The finite element method in partial differential equations*. Wiley, Chichester, 1977.

[28] W.D. Murray and F. Landis. Numerical and machine solutions of transient heat-conduction problems involving melting or freezing. *Trans. ASME (C) J. Heat Transfer*, 81:106–112, 1959.

[29] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[30] M.H. Protter and H.F. Weinberger. *Maximum Principles in Differential Equations*. Prentice-Hall, Englewood Cliffs, 1967.

[31] J.N. Reddy. *An introduction to the finite element method*. McGraw-Hill, New York, 1984.

[32] P.P. Silvester and R.L. Ferrari. *Finite elements for electrical engineers*. Cambridge University Press, Cambridge, 1983.

[33] J. Smoller. *Shock Waves and Reaction-Diffusion Equations*. Springer, New York, 1983.

[34] J. Stefan. Über die Theorie der Eisbildung, insbesondere über die Eisbildung im Polarmeere. *Annalen der Physik und Chemie*, 42:269–286, 1891.

[35] James Stewart. *Calculus. Fifth Edition*. Brooks/Cole, New York, 2002.

[36] G. Strang. *Linear Algebra and its Applications, (third edition)*. Harcourt Brace Jovanovich, San Diego, 1988.

[37] G. Strang and G.J. Fix. *An Analysis of the Finite Element Method*. Prentice Hall Inc., Englewood Clifts, New Jersey, 1973.

[38] P.K. Sweby. High resolution schemes using flux-limiters for hyperbolic conservation laws. *SIAM J. Num. Anal.*, 21:995–1011, 1984.

[39] R. Temam. *Navier-Stokes Equations*. North-Holland, Amsterdam, 1985.

[40] U. Trottenberg, C.W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, 2001.

[41] H.A. van der Vorst. *Iterative Krylov Methods for Large Linear Systems*. Cambridge University Press, Cambridge, UK, 2003.

[42] R.S. Varga. *Matrix Iterative Analysis*. Prentice-Hall, Englewood Cliffs, N.J., 1962.

[43] F. Vermolen and K. Vuik. A numerical method to compute the dissolution of second phases in ternary alloys. *J. Comp. Appl. Math.*, 93:123–143, 1998.

[44] A. Visintin. *Models of phase transitions, Progress in nonlinear differential equations and their applications*, volume 28. Birkhäuser, Boston, 1996.

[45] C. Vuik. *The Solution of a One-Dimensional Stefan Problem*. CWI-tract 90. CWI, Amsterdam, 1993.

[46] C. Vuik. Some historical notes about the stefan problem. Reports of the Faculty of Technical Mathematics and Informatics 93–07, Delft, 1993.

[47] H. Weber. *Die partiellen Differential-Gleichungen der Mathematischen Physik*. Vieweg, Braunschweig, 1901.

[48] Pieter Wesseling. *An introduction to Multigrid Methods*. John Wiley & Sons, Ltd, 1992. Corrected reprint. R.T. Edwards Inc., Philadelphia, 2004.

[49] K. Yosida. *Functional Analysis*. Springer Verlag, Berlin, 1971.

[50] O.C. Zienkewicz. *Finite element method in Engineering Science*. Mc Graw-Hill, 1971.

Index

absolutely stable, 203
ADI method, 210
amplification factor, 203
amplification matrix, 203
approximate factorization, 178
arc length, 4
artificial diffusion, 127
assembly of the large matrix, 104

band matrices, 154
band matrix, 154, 158
basis functions, 91, 92
biharmonic equation, 24
bilinear transformation, 139
block matrix, 42
boundary cells, 56
boundary condition, 53, 101
boundary conditions, 15, 20, 24, 64, 80
boundary element, 113
boundary fitted coordinates, 48
boundary layer, 36
Boyle's law, 22
Buckley-Leverett equation, 243

cable equation, 27
cell vertices, 55
central divided difference, 28, 52
CFL criterion, 225
characteristic, 230
characteristic equation, 230
characteristic relation, 230
checker board numbering, 167
Cholesky decomposition, 29
circle symmetry, 145
clamped beam, 147
clamped boundary, 24
clamped plate, 78
classical solution, 69
classification, 13
coarse grid, 183
coarsening level, 185
collocation, 126
compact matrices, 154
compatibility condition, 16
conforming element, 116
conservation, 1, 2, 51
conservation law, 6
conservation laws, 69
conservative form, 229
conservative scheme, 53
consistency, 200
constant coefficients, 14
constitutive equation, 78
continuous eigenvalue problem, 94
contraction, 190
contractive mapping, 190
control volume, 52, 55
convection diffusion, 36
convection term, 56
convection-diffusion equation, 56, 122, 124
convergence of Ritz's method, 95
conversion formula, 41
correction, 162
Crank-Nicholson, 202
Cuthill-McKee renumbering, 161

damped Jacob, 182
Darcy's Law, 3
defect correction, 162
delta function, 115, 126
diagonally block tridiagonal matrix, 167
diagonally dominant, 45
diffusion equation, 195
direct substitution, 157
directional derivative, 4
Dirichlet boundary condition, 16, 27
Dirichlet boundary conditions, 17, 43
discrete maximum principle, 45
discrete transformation, 58
dispersion, 222
displacement, 24
dissipation, 221
divergence, 4, 51
divergence theorem, 5, 73

eigenfunctions, 94
eigenvibrations, 218
elastic string, 7, 69
elasticity modulus, 78

element matrix, 104
element vector, 104
elliptic, 14, 15
elliptic operator, 20
energy norm, 87, 217
energy product, 87
envelope, 158
equilibrium, 14, 21
equilibrium solution, 195
error analysis, 57, 61
error estimate, 48, 117
error in FEM, 146
error in the boundary condition, 48
error in the fluxes, 54
essential boundary condition, 71, 80, 113
essential zeros, 158
Euler, 70
Euler-Lagrange equation, 73, 74
evolution, 14
exact solution, 31
existence, 17, 20

Fick's Law, 3
finite difference methods, 1, 27
finite element method, 91
finite element methods, 1
finite element packages, 107
finite volume methods, 1
fixed domain, 258
fixed point form, 189
flexural rigidity, 148
flux limiters, 238
flux vector, 6
Fourier expansion, 94
Fourier's law, 3
fourth order problems, 147
free boundary, 25
freely supported boundary, 25
front tracking, 258
frontal solution method, 160

Galerkin's method, 123
Gauss, 5
Gaussian elimination, 154
Gaussian rules, 100
general curvilinear coordinates, 48
general minimization in 1-d, 72
generalized formulation, 120
generalized solution, 69
Gershgorin, 9
Gershgorin's theorem, 204
ghost point, 65, 267
global error estimate, 44
gradient, 2
Gramm matrix, 96
Green, 74

half cell control volume, 60
heat conduction coefficient, 2
heat equation, 14, 15, 195
heat flow, 7, 20
Hermitian interpolation, 149
higher order polynomials, 135
Hilbert matrix, 94
homogeneous boundary conditions, 40
homotopy method, 192
Hooke's Law, 78
horizontal numbering, 41
hyperbolic, 14, 15

incomplete factorizations, 177
incompressibility condition, 22, 66
inflow, 17
initial conditions, 15, 17
integration by parts, 71
interior molecule, 55
interpolation error, 47
irrotational, 3
isoparametric transformations, 138
isotherms, 2
iteration matrix, 167
iterative methods, 162

Jacobi's method, 165
Jacobian, 58, 139

kinetic energy, 217
Krylov space, 1, 170

Lagrangian polynomial, 99
Laplace operator, 45
Laplace's equation, 46
Laplacian, 15, 17
Laplacian equation, 40, 54
Laplacian in general coordinates, 58
large matrix, 104
large right-hand side, 104
Lax Wendroff scheme, 237
Lax-Milgram theorem, 11, 132
Lemma of Dubois-Reymond, 71, 73
level set, 264
limiter function, 239
line element, 113
linear basis function in $\mathbb{R}^2$, 109
linear interpolation, 46, 47
loaded plate, 78
lower triangular matrix, 156
LU-decomposition, 29, 154
lumping, 200, 202

M-matrix, 164
mass matrix, 198
material derivative, 22
matrix vector form, 41
maximum principle, 18
mesh generation, 107
mesh Péclet condition, 39
mesh Péclet number, 38
method of lines, 197, 220
methods of lines, 209
midpoint rule, 100
minimal potential energy, 76
minimal surface problem, 75
minimization, 1
minimization with constraints, 150
mixed approach, simple example, 149
modulus of elasticity, 24
moving interface, 256
multi-grid methods, 1
Multigrid, 185
Multigrid methods, 170
multiplicators, 156

nabla, 2, 4
natural boundary, 120
natural boundary condition, 65, 71, 80, 113
natural boundary conditions, 17, 43
Navier-Stokes equations, 21
nearly orthogonal, 95
neglected set, 178
Neumann boundary condition, 16
neutrally stable, 217
Newmark, 224
Newton, 190
Newton iteration, 189
Newton-Cotes, 113
Newton-Cotes rule, 101
Newtonian fluid, 22
nodal points, 40
node point, 28
nodes, 40, 55
non equidistant grids, 51
non rectangular region, 43
non-conforming element, 116
non-homogeneous boundary conditions, 84
non-homogeneous essential boundary condition, 121
non-symmetric problem, 122
normal vector, 79
numerical integration, 100
numerical integration in R^n, 112

oblique numbering, 42
order of the error, 31
overrelaxation, 167

Péclet number, 36
parabolic, 14, 15
parameterization, 4
partial differential equations, 1
penalty approach, 150
perturbation, 57, 62
Petrov-Galerkin method, 126
Petrov-Galerkin upwinding, 126
Picard iteration, 189
piecewise linear, 98
piecewise polynomial, 98
pivoting, 156
pivots, 156
planar stress, 62
plane stress, 23, 77
Poincaré, 10
Poisson's constant, 24
Poisson's equation, 14, 17, 20, 40, 97, 108, 145
Poisson's ratio, 78
porous media, 229
positive definite, 29, 88, 171
positivity, 83
postprocessing, 107
potential, 20
potential energy, 7, 69, 217
potentials, 3
preconditioned CG algorithm, 174
preconditioner, 162, 188
preprocessing, 107
profile, 158
profile method, 154, 158, 160
prolongation, 183
pseudo-temperature, 264

quadratic elements, 135
quadratic interpolation, 135
quadratic triangles, 136
quadratic triangles, curved , 142
quadratic triangles, straight, 135
quasi-linear PDE, 15

radiation boundary conditions, 60
radiation coefficient, 55
reference pressure, 68
reference temperature, 55, 62
region of determination, 226
region of influence, 227, 231
regular splitting, 164
remeshing, 262

residual, 162
restriction, 183
reversed Cuthill-McKee, 161
Riesz' representation theorem, 88
Ritz's method, 1, 91
Robin, 16
rotating cone, 130
rough part, 183

second divided difference, 28
self-adjoint, 88
SGA, 126
Simpson's rule, 100, 136
singularly perturbed problems, 36
smooth part, 183
smoother, 183
Sobolev space, 89
solenoidal, 3
solution space, 120
spectral radius, 163
square integrable, 80
staggered grid, 63
standard iteration, 162
start value, 162
steady state, 217
Stefan problem, 255
Stokes equations, 66, 143
strain, 24
strain-displacement relation, 78
stream line upwinding, 128
stress tensor, 22, 63
strong solution, 89
strongly elliptic, 82
subdivision into triangles, 109
subinterval, 28
super solution, 47
SUPG, 126
symmetry, 83

target space, 91
Taylors formula, 28
test function, 80, 120
test space, 126
the transport equation, 229
time-dependent problems, 17
transformation matrices, 58
transient behavior, 14
transversal vibrations, 17
trapezoid rule, 100
truncation error, 28
two component field, 62
two grid algorithm, 183

unconditional stability, 212
underrelaxation, 167
unique solution, 15
uniqueness, 17
upper triangular matrix, 156
upwind differencing, 38

varying coefficients, 14
vector field, 4
vector space, 88
vertical numbering, 42
Von Neumann, 30, 233

wave equation, 14, 15, 215
wave front method, 160
weak formulation, 69, 80, 119
weak solution, 89
weighted gradients approach, 268
well-posedness, 197
wiggles, 37

Z-matrix, 45

Numerical Methods in Scientific Computing

Jos van Kan, Guus Segal, Fred Vermolen

This is a book about numerically solving partial differential equations occurring in technical and physical contexts and the authors have set themselves a more ambitious target than to just talk about the numerics. Their aim is to show the place of numerical solutions in the general modeling process and this must inevitably lead to considerations about modeling itself. Partial differential equations usually are a consequence of applying first principles to a technical or physical problem at hand. That means, that most of the time the physics also have to be taken into account especially for validation of the numerical solution obtained. This book aims especially at engineers and scientists who have 'real world' problems. It will concern itself less with pesky mathematical detail. For the interested reader though, we have included sections on mathematical theory to provide the necessary mathematical background. Since this treatment had to be on the superficial side we have provided further reference to the literature where necessary.





Jos van Kan, Retired professor Delft University of Technology, Delft Institute of Applied Mathematics

Guus Segal, Retired professor Delft University of Technology, Delft Institute of Applied Mathematics

Fred Vermolen, University of Hasselt, Department of Mathematics and Statistics, Computational Mathematics Group

© 2023 TU Delft OPEN Publising
ISBN 978-94-6366-740-1
DOI https://doi.org/10.59490/t.2023.009

textbooks.open.tudelft.nl

Cover image:
TU Delft OPEN Publishing.
No further use allowed.