An Introduction to the Finite Element Method (FEM) for Differential Equations in 1D

Mohammad Asadzadeh

June 24, 2015

Contents

1 Introduction **1**

1.1 Ordinary differential equations (ODE) 1

1.2 Partial differential equations (PDE) 2

1.2.1 Exercises 8

2 Polynomial Approximation in 1d **9**

2.1 Overture 9

2.1.1 Basis function in nonuniform partition 14

2.2 Variational formulation for (IVP) 17

2.3 Galerkin finite element method for (2.1.1) 19

2.4 A Galerkin method for (BVP) 21

2.4.1 The nonuniform version 26

2.5 Exercises 27

3 Interpolation, Numerical Integration in 1d **31**

3.1 Preliminaries 31

3.2 Lagrange interpolation 39

3.3 Numerical integration, Quadrature rules 41

3.3.1 Composite rules for uniform partitions 44

3.3.2 Gauss quadrature rule 48

4 Two-point boundary value problems **53**

4.1 A Dirichlet problem 53

4.2 The finite element method (FEM) 58

4.3 Error estimates in the energy norm 59

4.4 FEM for convection–diffusion–absorption BVPs 65

4.5 Exercises 72

5 Scalar Initial Value Problems **81**
5.1 Solution formula and stability 82
5.2 Finite difference methods 83
5.3 Galerkin finite element methods for IVP 86
5.3.1 The continuous Galerkin method 87
5.3.2 The discontinuous Galerkin method 90
5.4 Exercises 92

6 Initial Boundary Value Problems in 1d **95**
6.1 Heat equation in 1d 95
6.1.1 Stability estimates 96
6.1.2 FEM for the heat equation 100
6.1.3 Exercises 104
6.2 The wave equation in 1d 106
6.2.1 Wave equation as a system of PDEs 107
6.2.2 The finite element discretization procedure 108
6.2.3 Exercises 111

A Answers to Exercises **115**

B Algorithms and MATLAB Codes **121**

Table of Symbols and Indices **135**

Preface and acknowledgments. This text is an elementary approach to finite element method used in numerical solution of differential equations in one space dimension. The purpose is to introduce students to piecewise polynomial approximation of solutions using a minimum amount of theory. The presented material in this note should be accessible to students with knowledge of calculus of single- and several-variables and linear algebra. The theory is combined with approximation techniques that are easily implemented by Matlab codes presented at the end.

During several years, many colleagues have been involved in the design, presentation and correction of these notes. I wish to thank Niklas Eriksson and Bengt Svensson who have read the entire material and made many valuable suggestions. Niklas has contributed to a better presentation of the text as well as to simplifications and corrections of many key estimates that has substantially improved the quality of this lecture notes. Bengt has made all *xfig* figures. The final version is further polished by John Bondestam Malmberg and Tobias Gebäck who, in particular, have many useful input in the Matlab codes.

void

Chapter 1

Introduction

In this lecture notes we present an introduction to approximate solutions for differential equations. A differential equation is a relation between a function and its derivatives. In case the derivatives that appear in a differential equation are only with respect to one variable, the differential equation is called ordinary. Otherwise it is called a partial differential equation. For example,

$$\frac{du}{dt} - u(t) = 0, \tag{1.0.1}$$

is an ordinary differetial equation, whereas

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0, \tag{1.0.2}$$

is a partial differential (PDE) equation. In (1.0.2) $\frac{\partial u}{\partial t}$, $\frac{\partial^2 u}{\partial x^2}$ denote the partial derivatives. Here t denotes the time variable and x is the space variable. We shall only study one space dimentional equations that are either stationary (time-independent) or time dependent. Our focus will be on the following equations:

1.1 Ordinary differential equations (ODE)

• An example of population dynamic as in (1.0.1)

$$\frac{du}{dt} - \lambda u(t) = f(t), \tag{1.1.1}$$

where λ is a constant and f is a source function.
• A stationary (time-independent) heat equation as

$$-\frac{d^2u}{dx^2} = f(x), \tag{1.1.2}$$

• A stationary convection-diffusion equation

$$-\frac{d^2u}{dx^2} + \frac{du}{dx} = f(x), \tag{1.1.3}$$

where $f(x)$ is a source function.

1.2 Partial differential equations (PDE)

• The *heat equation*

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = f(x). \tag{1.2.1}$$

• The *wave equation*

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = f(x). \tag{1.2.2}$$

• The time depending *convection-diffusion or reaction-diffusion equation*

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} + \frac{\partial u}{\partial x} = f(x). \tag{1.2.3}$$

Some notation. For convenince we shall use the following notation:

$$\dot{u} = \frac{\partial u}{\partial t}, \quad \ddot{u} = \frac{\partial^2 u}{\partial t^2}, \quad u' = \frac{\partial u}{\partial x}, \quad u'' = \frac{\partial^2 u}{\partial x^2}.$$

Example 1.1 (Initial Conditions). *Consider the simple equation $\dot{u}(t) = t$. Evidently, $u(t) = t^2/2$, is a solution. But, for any constant C, $t^2/2 + C$ is also a solution. In this way we have infintely many solutions (one for each constant C). To determine a unique solution we need to supply the equation with one extra condition. Since the time variable t is always assumed to be $t \geq 0$, if we know the value of $u(t)$, e.g., at the beginning, i.e., the initial value e.g., $u(0) = 3$, then $u(t) = t^2/2 + 3$ is the unique solution to the initial value problem: $\dot{u}(t) = t$, $u(0) = 3$. A differential equation associated with initial conditions is an initial value problem.*

Example 1.2 (Boundary Conditions). *Likewise* $u(x) = -x^2/2$ *is a solution to* $-u''(x) = 1$. *But also all* $u(x) = -x^2/2 + Ax + B$ *are solutions, for all arbitrary constants* A *and* B. *Therefore, to determine a unique solution* $u(x)$ *we need to determine some fixed values for* A *and* B, *hence we need to supply two conditions. Here if e.g.,* x *belongs to a bounded interval, say,* $[0, 1]$, *then given the boundary values* $u(0) = 1$ *and* $u(1) = 0$, *we get from* $u(x) = -x^2/2 + Ax + B$ *that* $B = 1$ *and* $A = -1/2$. *Thus the solution to the the initial boundary value problem:* $-u''(x) = 1$, $u(0) = 1$, $u(1) = 0$ *is:* $u(x) = -x^2/2 - x/2 + 1$. *The generale rule is that one should supply as many conditions as the highest ordre of the derivative in each variable. So, for example, for the hear equation* $\dot{u} - u'' = 0$, *to get a unique solution we nedd to supply one initial condition (there is one time derivative in the equation) and two boundary conditions (there are two derivatives in* x*), whereas for the wave equation* $\ddot{u} - u'' = 0$ *we have to give two conditions in each variable* x *and* t. *A differential equation with supplied boundary conditions is a boundary value problem.*

Objectives: For f being a simple elementary function (a polynomial, a trigonometric, or exponential type function or a combination of them), the equations (1.1.1)-(1.2.3), associated with suitable initial and boundary conditions, have often closed form analytic solutions. But real problems: general two and three dimensional problems, modeled by equations with variable cefficients and in complex geometry, are seldom analytically solvable.

In this note our objective is to introduce numerical methods that approximate solutions for differential equations by polynomials. To check the quality (*reliability and efficiency*) of these numerical methods, we choose to apply them to the equations (1.1.1)-(1.2.3), where we already know their analytic solutions. Below we shall give examples of analytic solutions to ODEs: (1.1.1)-(1.1.3). For examples on analytic solutions for the PDEs: (1.2.1)-(1.2.3), we refer to the *separation of variables technique* introduced in the second part of our course.

Example 1.3. *Determine the solution to the initial value problem*

$$\dot{u}(t) - \lambda u(t) = 0, \qquad u(0) = u_0, \tag{1.2.4}$$

assuming that $u(t) > 0$, *for all* t, $\lambda = 1$ *and* $u_0 = 2$.

Solution. Since $u(t) \neq 0$, for all t, we may divide the equation (1.2.4) by

$u(t)$ and get $\frac{\dot{u}(t)}{u(t)} = \lambda$. Relabeling t by s and integrating over $(0,t)$ we get

$$\int_0^t \frac{\dot{u}(s)}{u(s)}\,ds = \lambda \int_0^t ds \quad \Longrightarrow \quad \Big[\ln u(s)\Big]_0^t = \lambda[s]_0^t. \tag{1.2.5}$$

Hence we have

$$\ln u(t) - \ln u(0) = \lambda t \quad \text{or} \quad \ln \frac{u(t)}{u(0)} = \lambda t. \tag{1.2.6}$$

Thus

$$\frac{u(t)}{u(0)} = e^{\lambda t}, \qquad \text{i.e.} \quad u(t) = u_0 e^{\lambda t}. \tag{1.2.7}$$

Consequently, with $\lambda = 1$ and $u_0 = 2$ we have $u(t) = 2e^t$.

To derive solutions to our examples on a systematic way, we recall the procedure for determining a particular solution u_p to a second order differential equation with constant coefficiets of the form:

$$u''(x) + au'(x) + bu(x) = f(x). \tag{1.2.8}$$

1. If $f(x) =$ a polynomial of degree n. Set
   - i) $u_p(x) = a_0 + a_1 x + \cdots + a_n x^n, \quad$ if $b \neq 0$
   - ii) $u_p(x) = x(a_0 + a_1 x + \cdots + a_n x^n), \quad$ if $b = 0, a \neq 0$
2. If $f(x) =$ (polynom) $\times\, e^{\sigma x}$. Set
   - i) $u_p(x) = z(x)e^{\sigma x}$.
     This gives a new differential equation for z solved by 1).
   - ii) $u_p(x) = Ae^{\sigma x}$, if polynom = constant.
     This works if $\sigma^2 + a\sigma + b \neq 0$: i.e. σ *is not* a root to the characteristic equation.
3. If $f(x) = p\cos(\omega x) + q\sin(\omega x)$. Set
   - i) $u_p(x) = C\cos(\omega x) + D\sin(\omega x)$, for $-\omega^2 + ai\omega + b \neq 0$, i.e., if $i\omega$ *is not* a root to the characteristic equation.
   - ii) $u_p(x) = x(C\cos(\omega x) + D\sin(\omega x))$, if $-\omega^2 + ai\omega + b = 0$.

Example 1.4. *Find all solutions to the differential equation*

$$u''(x) - u(x) = \cos(x). \tag{1.2.9}$$

Solution. Due to the highest number of derivatives (here 2 which is *also called the order of this differential equation*), we shall have solutions depending on two arbitrary constants. As we mentioned earilear a unique solution would require supplying 2 conditions, which we skip in this problem.

We note that the characteristic equation to this differentisal equatios: $r^2 - 1 = 0$ has the roots $\omega = \pm 1$. We split the solution procedure in 3 steps:

Step 1: According to the table above we choose a particular solution $u_p(x)$ of the form

$$u_p(x) = A\cos x + B\sin x. \tag{1.2.10}$$

Differentiating twice and inserting in the equation (1.2.9) yeilds

$$\begin{aligned} u_p'(x) &= -A\sin x + B\cos x \\ u_p''(x) &= -A\cos x - B\sin x \\ u_p''(x) - u_p(x) &= -2A\cos x - 2B\sin x = \cos x \end{aligned}$$

identifying the coefficients yields $A = -\frac{1}{2}, B = 0$. Thus

$$u_p(x) = -\frac{1}{2}\cos x \tag{1.2.11}$$

Step 2: The homogeneous solution is given by the standard ansatz

$$u_h(x) = C_1 e^{r_1 x} + C_2 e^{r_2 x}, \tag{1.2.12}$$

where C_1 and C_2 are arbitrary constants and $r_1 = 1$ and $r_2 = -1$ are the roots of the characteristic equation. Hence

$$u_h(x) = C_1 e^{x} + C_2 e^{-x}. \tag{1.2.13}$$

Step 3: Finally, the general solution is given by adding the particular and homogeneous solutions

$$u(x) = -\frac{1}{2}\cos x + C_1 e^{x} + C_2 e^{-x}. \tag{1.2.14}$$

In the above example we obtained general solutions depending on two constants. Below we shall demonstrate an example where, supplying two boundary conditions, we obtain a unique solution

Example 1.5. *Determine the unique solution of the following boundary value problem*

$$u'' + 2u' + u = 1 + x + 2\sin x, \qquad u(0) = 1, \quad u'(0) = 0. \tag{1.2.15}$$

Homogeneous solution:
The characteristic equation for the differential equation (1.2.15) is given by

$$r^2 + 2r + 1 = 0, \quad \text{and has dubbel root} \quad r_{1,2} = -1. \tag{1.2.16}$$

This gives the homogeneous solutions as

$$u_h = (C_1 + C_2 x)e^{-x}. \tag{1.2.17}$$

Particular solution:
The particula solution can be written as sum of two particular solution to the following equations:

$$u_1'' + 2u_1' + u_1 = 1 + x, \tag{1.2.18}$$

and

$$u_2'' + 2u_2' + u_2 = 2\sin x. \tag{1.2.19}$$

Since the differential equation is *linear*, a concept justified by the relation

$$(au_1 + bu_2)' = au_1' + bu_2' \quad \textit{and} \quad \forall a, b \in \mathbb{R},$$

thus $u = u_1 + u_2$ will be a particular solution for (1.2.15). Using the table of particular solutions, we may insert $u_1(x) = Ax + B$, as particular solution, in (1.2.18) and get

$$2A + Ax + B = 1 + x. \tag{1.2.20}$$

Identifying the coefficients in (1.2.20) gives $A = 1$ and $B = -1$. Hence

$$u_1(x) = x - 1.$$

Once again using the table of particular solutions, we may insert $u_2(x) = A\sin x + B\cos x$, as particular solution, in (1.2.19) and get

$$2A\cos x - 2B\sin x = 2\sin x. \tag{1.2.21}$$

Identifying the coefficients in (1.2.21) gives $A = 0$ and $B = -1$. Hence

$$u_2(x) = -\cos x.$$

Thus the general solution is given by

$$u = u_h + u_1 + u_2 = (C_1 + C_2 x)e^{-x} + (x - 1) - \cos x. \tag{1.2.22}$$

Now we use the boundary conditions and determine the coefficients C_1 and C_2. Observe that

$$u' = C_2 e^{-x} - (C_1 + C_2 x)e^{-x} + 1 + \sin x,$$

and we have that

$$u(0) = 1 \Longrightarrow C_1 - 1 - 1 = 1 \Longrightarrow C_1 = 3.$$

Further

$$u'(0) = 0 \Longrightarrow C_2 - C_1 + 1 = 0 \Longrightarrow C_2 = C_1 - 1 \Longrightarrow C_2 = 2.$$

Thus the final solution is

$$u(x) = x - 1 - \cos x + e^{-x}(3 + 2x).$$

Summary: These examples of ODEs can serve as a sort of warm up. As we mentioned the corresponding analytical solutions for our PDEs is the subject of Fourier analysis that we cover on the second part of this course. The remaing chapters will be devoted to the approximation methods for solution of our ODEs and PDEs. We shall approximate the solutions with, piecewise, polynomials. Such approximations are known as the *Galerkin finite element methods (FEM)*. In its final step, a finite element procedure yields a linear system of equations (LSE) where the unknowns are the approximate values of the solution at certain points. Then, an approximate solution is constructed by adapting, piecewise, polynomials of certain degree to these point values.

The entries of the coefficient matrix and the right hand side of FEM's final linear system of equations consist of integrals which are not always easily computable. Therefore, numerical integration are introduced to approximate such integrals. *Interpolation techniques* are introduced for both accurate polynomial approximations and to derive error estimates necessary in determining qualitative properties of the approximate solutions. That is to show how the approximate solution approaches the exact solution as the number of unknowns increase.

1.2.1 Exercises

Problem 1.1. *Find all solutions to the following homogeneous (their right hand side is zero "0") differential equations*
a) $u'' - 3u' + 2u = 0$ *b)* $u'' + 4u = 0$ *c)* $u'' - 6u' + 9u = 0$

Problem 1.2. *Find all solutions to the following non-homogeneous (their right hande side are non-zero "* $\neq 0$*") differential equations*
a) $u''+2u'+2u = (1+x)^2$ *b)* $u''+u'+2u = \sin x$ *c)* $u''+3u'+2u = e^x$

Problem 1.3. *Find a particular solution to each of the following equations*
a) $u'' - 2u' = x^2$ *b)* $u'' + u = \sin x$ *c)* $u'' + 3u' + 2u = e^x + \sin x$.

Problem 1.4. *Solve the boundary value problem for all* $x \in (0, 1)$,

$$-u'' + u = f(x), \quad u(0) = u(1) = 0,$$

a) for $f(x) = 0$, *b) for* $f(x) = x$, *c) for* $f(x) = \sin(\pi x)$,

Problem 1.5. *Solve the following boundary value problems*
a) $-u'' = x - 1, \quad 0 < x < \pi, \qquad u'(0) = u(\pi) = 0,$
b) $-u'' = x, \quad 0 < x < \pi, \qquad u'(0) = u'(1) = 0.$

Chapter 2

Polynomial Approximation in 1d

Our objective is to present the finite element method (FEM) as an approximation technique for solution of differential equations using piecewise polynomials. This chapter is devoted to some necessary mathematical environments and tools, as well as a motivation for the unifying idea of using finite elements: A numerical strategy arising from the need of changing a continuous problem into a discrete one. The continuous problem will have infinitely many unknowns (if one asks for $u(x)$ at every x), and it cannot be solved exactly on a computer. Therefore it has to be approximated by a discrete problem with a finite number of unknowns. The more unknowns we keep, the better the accuracy of the approximation will be, but at a greater computational expense.

2.1 Overture

Below we shall introduce a few standard examples of classical differential equations and some regularity requirements.

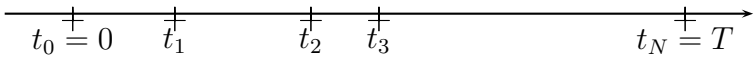
Ordinary differential equations (ODEs)
An *initial value problem* (IVP), for instance a model in population dynamics where $u(t)$ is the size of the population at time t, can be written as

$$\dot{u}(t) = \lambda u(t), \quad 0 < t < T, \qquad u(0) = u_0, \tag{2.1.1}$$

where $\dot{u}(t) = \frac{du}{dt}$ and λ is a positive constant. For $u_0 > 0$ this problem has the increasing analytic solution $u(t) = u_0 e^{\lambda \cdot t}$, which blows up as $t \to \infty$.

• **Numerical solutions of (IVP)**

Example 2.1. *Explicit (forward) Euler method (a finite difference method). We discretize the IVP* (2.1.1) *with the forward Euler method based on a partition of the interval* $[0, T]$ *into* N *subintervals, and an approximation of*



the derivative by a difference quotient at each subinterval $[t_k, t_{k+1}]$ *by* $\dot{u}(t) \approx \frac{u(t_{k+1})-u(t_k)}{t_{k+1}-t_k}$. *Then an approximation of* (2.1.1) *is given by*

$$\frac{u(t_{k+1}) - u(t_k)}{t_{k+1} - t_k} = \lambda \cdot u(t_k), \qquad k = 0, \ldots, N-1, \quad \textit{and} \quad u(0) = u_0, \tag{2.1.2}$$

and thus, letting $\Delta t_k = t_{k+1} - t_k$,

$$u(t_{k+1}) = (1 + \lambda \Delta t_k) u(t_k). \tag{2.1.3}$$

Starting with $k = 0$ *and the data* $u(0) = u_0$, *the solution* $u(t_k)$ *would, iteratively, be computed at the subsequent points:* $t_1,\ t_2, \ldots, t_N = T$.
For a uniform partition, where all subintervals have the same length Δt, (2.1.3) *would be of the form*

$$u(t_{k+1}) = (1 + \lambda \Delta t) u(t_k), \quad k = 0, 1, \ldots, N-1. \tag{2.1.4}$$

Iterating we get

$$u(t_{k+1}) = (1 + \lambda \Delta t) u(t_k) = (1 + \lambda \Delta t)^2 u(t_{k-1}) = \ldots = (1 + \lambda \Delta t)^{k+1} u_0.$$

Other finite difference methods for (2.1.1) are introduced in Chapter 5. There are corresponding finite difference methods for PDE's. Our goal, however, is to study the *Galerkin finite element method*. To this approach we need to introduce some basic tools:

Finite dimensional linear space of polynomials on an interval
Below we give an examples of finite dimensional linear space of polynomials defined on an interval. In our study we shall consider, mainly, polynomials of

degree 1. Higher degree polynomials are studied in some details in Chapter 3: *the polynomial interpolation in 1D*.

We define $\mathcal{P}^{(q)}(a,b) := \{\text{Space of polynomials of degree} \le q,\ a \le x \le b\}$. A possible basis for $\mathcal{P}^{(q)}(a,b)$ would be $\{x^j\}_{j=0}^{q} = \{1, x, x^2, x^3, \ldots, x^q\}$. These are, in general, non-orthogonal polynomials and may be orthogonalized by the Gram-Schmidt procedure. The dimension of $\mathcal{P}^q$ is therefore $q+1$.

Example 2.2. *For linear approximation we shall only need the basis functions* 1 *and* x. *An alternative linear basis function on the interval* $[a,b]$ *is given by two functions* $\lambda_a(x)$ *and* $\lambda_b(x)$ *with the additional property*

$$\lambda_a(x) = \begin{cases} 1, & x = a \\ 0, & x = b \end{cases} \quad \textit{and} \quad \lambda_b(x) = \begin{cases} 1, & x = b \\ 0, & x = 0. \end{cases}$$

Being linear $\lambda_a(x) = Ax + B$. To determine the coefficients A and B we have that

$$\begin{cases} \lambda_a(a) = 1 & \Longrightarrow \quad Aa + B = 1 \\ \lambda_a(b) = 0 & \Longrightarrow \quad Ab + B = 0 \end{cases}$$

Subtracting the two relations above we get $A(b-a) = -1 \implies A = \frac{-1}{b-a}$. Then, from the second relation: $B = -Ab$ we get $B = \frac{b}{b-a}$. Thus,

$$\lambda_a(x) = \frac{b-x}{b-a}. \qquad \text{Likewise} \qquad \lambda_b(x) = \frac{x-a}{b-a}.$$

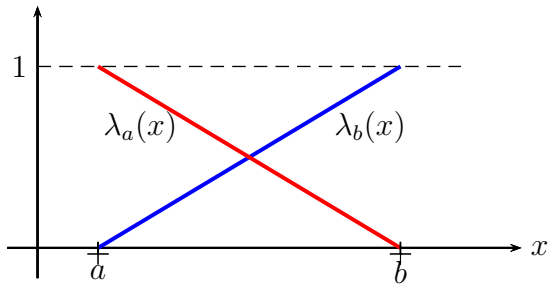


Figure 2.1: Linear basis functions $\lambda_a(x)$ and $\lambda_b(x)$.

Note that

$$\lambda_a(x) + \lambda_b(x) = 1, \quad \text{and} \quad a\lambda_a(x) + b\lambda_b(x) = x.$$

Thus, we get the original basis functions: 1 and x for the linear polynomial functions, as a linear combination of the basis functions $\lambda_a(x)$ and $\lambda_b(x)$. Hence, any linear function $f(x)$ on an interval $[a, b]$ can be written as:

$$f(x) = f(a)\lambda_a(x) + f(b)\lambda_b(x). \tag{2.1.5}$$

This is easily seen by the fact that the right hand side in (2.1.5) yields:

$$f(a)\lambda_a(a) + f(b)\lambda_b(a) = f(a) \times 1 + f(b) \times 0 = f(a),$$

$$f(a)\lambda_a(b) + f(b)\lambda_b(b) = f(a) \times 0 + f(b) \times 1 = f(b).$$

That is the two sides in (2.1.5) agree in two distinct points, therefore, being linear, they represent the same function.

Example 2.3. *Let $[a, b] = [0, 1]$ then $\lambda_0(x) = 1 - x$ and $\lambda_1(x) = x$. Consider the linear function $f(x) = 3x + 5/2$. Then $f(0) = 5/2$, $f(1) = 11/2$ and*

$$f(0)\lambda_0(x) + f(1)\lambda_1(x) = \frac{5}{2}(1 - x) + \frac{11}{2}x = 3x + 5/2 = f(x).$$

Definition 2.1. *Let $f(x)$ be a real valued function definied on $\mathbb{R}$ or on an interval that contains $[a, b]$. A linear interpolant of $f(x)$ on a and b is a linear function $\pi_1 f(x)$ such that $\pi_1 f(a) = f(a)$ and $\pi_1 f(b) = f(b)$.*

As in verification of (2.1.5), we have also $\pi_1 f(x) = f(a)\lambda_a(x) + f(b)\lambda_b(x)$:

$$\pi_1 f(x) = f(a)\frac{b - x}{b - a} + f(b)\frac{x - a}{b - a}.$$

Below, for simplicity, first we shall assume a uniform partition of the interval $[0, 1]$ into $M + 1$ subintervals of the same size h, i.e., we let $x_j = jh$, and consider subintervals $I_j := [x_{j-1}, x_j] = [(j - 1)h, jh]$ for $j = 1, \ldots, M + 1$. Then setting $a = x_{j-1} = (j - 1)h$ and $b = x_j = jh$ we may define

$$\lambda_{j-1}(x) = -\frac{x - jh}{h} \quad \text{and} \quad \lambda_j(x) = \frac{x - (j - 1)h}{h}.$$

We denote the space of all continuous piecewise linear polynomial functions on $\mathcal{T}_h$, by V_h. Let

$$V_h^0 := \{v : v \in V_h, \quad v(0) = v(1) = 0\}.$$

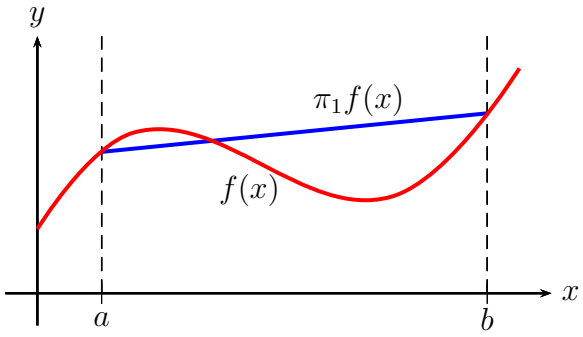


Figure 2.2: The linear interpolant $\pi_1 f(x)$ on a single interval.

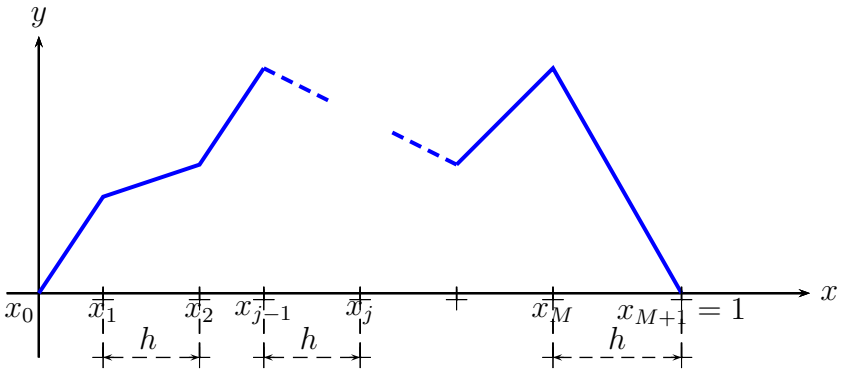


Figure 2.3: An example of a function in V_h^0 with uniform partition.

Applying (2.1.5), on each subinterval I_j, $j = 1, \ldots, M+1$, (using $\lambda_j(x), j = 1, \ldots, M$) we can easily construct the functions belonging V_h^0. To construct a function $v(x) \in V_h$ we shall also need additional basis functions $\lambda_0(x)$ and/or $\lambda_{M+1}(x)$ if $v(0) \neq 0$, and/or $v(1) \neq 0$, corresponding to non-vanishing data in the boundary value problems.

The standard basis for piecewise linears in a uniform partition are given by the so called *hat-functions* $\varphi_j(x)$ with the property that $\varphi_j(x)$ is a piecewise linear function such that $\varphi_j(x_i) = \delta_{ij}$, where

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases} \quad \text{i.e.} \quad \varphi_j(x) = \begin{cases} \frac{x-(j-1)h}{h} & (j-1)h \leq x \leq jh \\ \frac{(j+1)h-x}{h} & jh \leq x \leq (j+1)h \\ 0 & x \notin [(j-1)h, (j+1)h], \end{cases}$$

with obvious modifications for $j = 0$ and $j = M + 1$. The hat function $\varphi_j(x)$ is just a combination of two basis functions $\lambda_j(x)$ of the two adjacent intervals I_j and I_{j+1} (each of these two *adjacent* intervals has its own $\lambda_j(x)$, check this), extended by zero for $x \notin (I_j \cup I_{j+1})$.

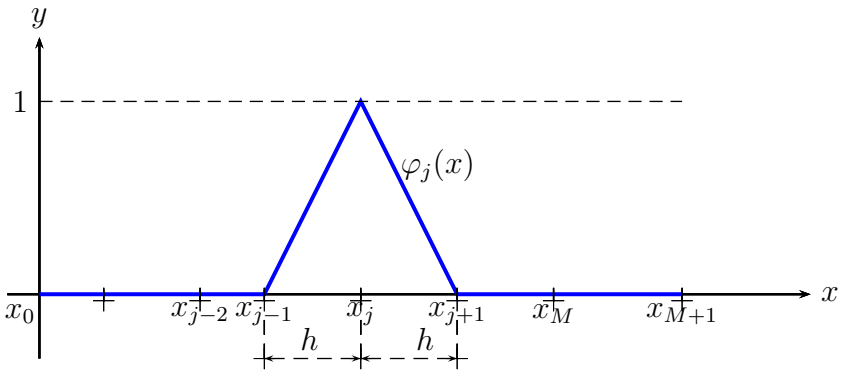


Figure 2.4: A general piecewise linear basis function $\varphi_j(x)$.

2.1.1 Basis function in nonuniform partition

Below we generalize the above procedure to the case of nonuniform partition. Let now $I = [0, 1]$ and define a partition of I into a collection of nonuniform subintervals. For example $\mathcal{T}_h : 0 = x_0 < x_1 < \ldots < x_M < x_{M+1} = 1$, with $h_j = x_j - x_{j-1}$, and $j = 1, \ldots, M + 1$, is a partition of $[0, 1]$ into $M + 1$ subintervals. Here $h := h(x)$, known as *the mesh function*, is a piecewise constant function defined as $h(x) = h_j$ for $x \in I_j = [x_{j-1}, x_j]$. We shall see that $\pi_1 f$ "gets closer to" f, as $\max h(x) \to 0$. Now we may apply the concept of the linear interpolant to a a set of nonuniform subintevals $I_j := [x_{j-1}, x_j]$ of a given interval I, simply by setting $a = x_{j-1}$ and $b = x_j$. Therefore, we define

$$\lambda_{j-1}(x) = \frac{x_j - x}{x_j - x_{j-1}} \quad \text{and} \quad \lambda_j(x) = \frac{x - x_{j-1}}{x_j - x_{j-1}}.$$

The corresponding basis functions for the nonuniform case are given as

$$\varphi_j(x) = \begin{cases} \frac{x - x_{j-1}}{h_j} & x_{j-1} \le x \le x_j \\ \frac{x_{j+1} - x}{h_{j+1}} & x_j \le x \le x_{j+1} \\ 0 & x \notin [x_{j-1}, x_{j+1}]. \end{cases}$$

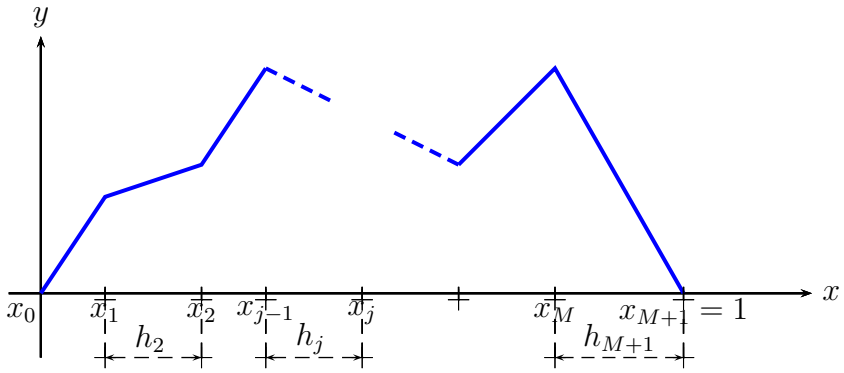


Figure 2.5: An example of a function in V_h^0.

Again with obvious modifications for $j = 0$ and $j = M + 1$.

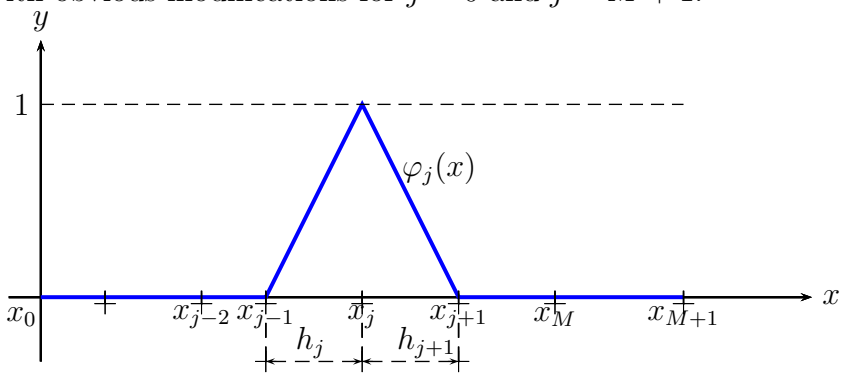


Figure 2.6: A general piecewise linear basis function $\varphi_j(x)$.

Vector spaces

To establish a framework we introduce some basic mathematical concepts:

Definition 2.2. *A set V of functions or vectors is called a linear space, or a vector space, if for all $u, v \in V$ and all $\alpha \in \mathbb{R}$ (real number), we have that*

$$\begin{aligned} &(i) \quad u + v \in V, \qquad (\textit{closed under addition}) \\ &(ii) \quad \alpha u \in V, \qquad (\textit{closed under multiplication by scalars}), \\ &(iii) \quad \exists\,(-u) \in V : \; u + (-u) = 0, \quad (\textit{closed under inverse}), \end{aligned} \tag{2.1.6}$$

where (i) and (ii) obey the usual rules of addition and multiplication by scalars. Observe that $\alpha = 0$ in (ii) (or (iii) and (i), with $v = (-u)$), implies that 0 (zero vector) is an element of every vector space.

Definition 2.3. *A scalar product (inner product) is a real valued operator on $V \times V$, viz $\langle u, v\rangle : V \times V \to \mathbb{R}$ such that for all u, v, $w \in V$ and all $\alpha \in \mathbb{R}$,*

$$\begin{aligned} &(i) \quad \langle u, v\rangle = \langle v, u\rangle, \qquad \textit{(symmetry)} \\ &(ii) \quad \langle u + \alpha v, w\rangle = \langle u, w\rangle + \alpha\langle v, w\rangle, \quad \textit{(bi-linearity)} \\ &(iii) \quad \langle v, v\rangle \geq 0, \quad \forall v \in V, \quad \textit{(positivity)} \\ &(iv) \quad \langle v, v\rangle = 0, \Longleftrightarrow v = 0 \quad \textit{(positive definiteness).} \end{aligned} \tag{2.1.7}$$

Definition 2.4. *A vector space V is called an inner product space if V is associated with a scalar product $\langle \cdot, \cdot \rangle$, defined on $V \times V$.*

Example 2.4. *A usual example of scalar product of two functions u and v defined on an interval $[a, b]$, known as the L_2 scalar product, is defined by*

$$\langle u, v\rangle := \int_a^b u(x)v(x)dx. \tag{2.1.8}$$

Here are examples of some vector spaces that are also linear product spaces associated with the scalar product defined by (2.1.8).

- $C(a, b)$: The space of continuous functions on an interval (a, b),
- $\mathcal{P}^{(q)}[a, b]$: the space of all polynomials of degree $\leq q$ on $C[a, b]$ and
- $V_h(a, d)$ and $V_h^0(a, b)$ defined above.

The reader may easily check that all the properties $(i) - (iv)$, in the definition, for the scalar product are fullfiled for these spaces.

Definition 2.5. *Two (real-valued) functions $u(x)$ and $v(x)$ are called orthogonal if $\langle u, v\rangle = 0$. The orthogonality is also denoted by $u \perp v$.*

Example 2.5. *For the functions $u(x) = 1$ and $v(x) = x$, we have that*

$$\int_{-1}^{1} u(x)v(x)dx = \int_{-1}^{1} 1{\times}x\, dx = 0, \quad \int_0^1 u(x)v(x)dx = \int_0^1 1{\times}x\, dx = 1/2 \neq 0.$$

Thus, 1 and x are orthogonal on the interval $[-1, 1]$, but not on $[0, 1]$.

Definition 2.6 (Norm)**.** *If $u \in V$ then the norm of u, or the length of u, associated with the scalar product* (2.1.8) *above is defined by:*

$$\|u\| = \sqrt{\langle u, u\rangle} = \langle u, u\rangle^{1/2} = \Big(\int_a^b |u(x)|^2 dx\Big)^{1/2}. \tag{2.1.9}$$

This norm is known as the L_2-norm of $u(x)$. There are other norms that we will introduce later on.

Now we recall one of the most useful inequalities that is frequently used in estimating the integrals of product of two functions.

Lemma 2.1 (The Cauchy-Schwarz inequality)**.** *For all inner products with their corresponding norms We have that*

$$|\langle u, v\rangle| \leq \|u\|\|v\|.$$

In particular for the L_2-norm and scalar product

$$\Big|\int uv\,dx\Big| \leq \Big(\int |u|^2\,dx\Big)^{1/2}\Big(\int |v|^2\,dx\Big)^{1/2}.$$

Proof. A simple proof is given by using

$$\langle u - av, u - av\rangle \geq 0, \qquad \text{with} \qquad a = \langle u, v\rangle/\|v\|^2.$$

Then by the definition of the L_2-norm and the symmetry property of the scalar product we get

$$0 \leq \langle u - av, u - av\rangle = \|u\|^2 - 2a\langle u, v\rangle + a^2\|v\|^2.$$

Setting $a = \langle u, v\rangle/\|v\|^2$ and rearranging the terms we get

$$0 \leq \|u\|^2 - \frac{\langle u, v\rangle^2}{\|v\|^4}\|v\|^2, \quad \text{and consequently} \quad \frac{\langle u, v\rangle^2}{\|v\|^2} \leq \|u\|^2,$$

which yields the desired result. □

Now we shall return to approximate solution for (2.1.1) using polynomials. To this approach we introduce the concept of weak formulation viz,

2.2 Variational formulation for (IVP)

We multiply the initial value problem (2.1.1) with test functions v in a certain vector space V and integrate over $[0, T]$, to get

$$\int_0^T \dot{u}(t)v(t)\,dt = \lambda\int_0^T u(t)v(t)\,dt, \qquad \forall v \in V, \tag{2.2.1}$$

or equivalently

$$\int_0^T (\dot{u}(t) - \lambda\, u(t))v(t)dt = 0, \qquad \forall v(t) \in V, \tag{2.2.2}$$

which, interpreted as inner product, means that

$$(\dot{u}(t) - \lambda\, u(t)) \perp v(t), \quad \forall v(t) \in V. \tag{2.2.3}$$

We refer to (2.2.1) as the *variational problem* for (2.1.1). We shall seek a solution for (2.2.1) in $C(0,T)$, or in

$$V := H^1(0,T) := \Big\{ f : \int_0^T \Big(f(t)^2 + \dot{f}(t)^2 \Big)\, dt < \infty \Big\}.$$

Definition 2.7. *If w is an approximation of u in the variational problem (2.2.1), then $\mathcal{R}(w(t)) := \dot{w}(t) - \lambda w(t)$ is called the residual error of $w(t)$.*

In general for an approximate solution w we have $\dot{w}(t) - \lambda w(t) \neq 0$, otherwise w and u would satisfy the same equation and by uniqueness we would get the exact solution ($w = u$). Our requirement is instead that w should satisfy (2.2.3), i.e. the equation (2.1.1) in average. In other words

$$\mathcal{R}(w(t)) \perp v(t), \quad \forall v(t) \in V. \tag{2.2.4}$$

We look for an *approximate solution* $U(t)$, called a *trial function* for (2.1.1), in the space of polynomials of degree $\leq q$:

$$V^{(q)} := \mathcal{P}^{(q)} = \{U : U(t) = \xi_0 + \xi_1 t + \xi_2 t^2 + \ldots + \xi_q t^q\}. \tag{2.2.5}$$

Hence, to determine $U(t)$ we need to determine the coefficients $\xi_0, \xi_1, \ldots \xi_q$. We refer to $V^{(q)}$ as the *trial space*. Note that $u(0) = u_0$ is given and therefore we may take $U(0) = \xi_0 = u_0$. It remains to find the real numbers $\xi_1, \ldots, \xi_q$. These are coefficients of the q linearly independent monomials t, t^2, ..., t^q. To this approach we define the *test function space*:

$$V_0^{(q)} := \mathcal{P}_0^{(q)} = \{v \in \mathcal{P}^{(q)} : v(0) = 0\}. \tag{2.2.6}$$

Thus, v can be written as $v(t) = c_1 t + c_2 t^2 + \ldots + c_q t^q$. For an approximate solution U, we require its residual $R(U)$ to satisfy the condition (2.2.4):

$$\mathcal{R}(U(t)) \perp v(t), \qquad \forall v(t) \in \mathcal{P}_0^{(q)}.$$

2.3 Galerkin finite element method for (2.1.1)

Given $u(0) = u_0$, find the approximate solution $U \in \mathcal{P}^{(q)}$ of (2.1.1) satisfying

$$\int_0^T \mathcal{R}(U(t))v(t)dt = \int_0^T (\dot{U}(t) - \lambda U(t))v(t)dt = 0,\ \forall v(t) \in \mathcal{P}_0^{(q)}. \quad (2.3.1)$$

Formally, this can be obtained requiring U to satify (2.2.2). Thus, since $U \in \mathcal{P}^{(q)}$, we may write $U(t) = u_0 + \sum_{j=1}^q \xi_j t^j$, then $\dot{U}(t) = \sum_{j=1}^q j\xi_j t^{j-1}$. Further, $\mathcal{P}_0^{(q)}$ is spanned by $v_i(t) = t^i, i = 1, 2, \ldots, q$. Therefore, it suffices to use these t^i:s as test functions. Inserting these representations for U, $\dot{U}$ and $v = v_i$, $i = 1, 2, \ldots, q$ into (2.3.1) we get

$$\int_0^1 \Big(\sum_{j=1}^q j\xi_j t^{j-1} - \lambda u_0 - \lambda \sum_{j=1}^q \xi_j t^j\Big) \cdot t^i dt = 0, \quad i = 1, 2, \ldots, q. \quad (2.3.2)$$

Moving the data to the right hand side, this relation can be rewritten as

$$\int_0^1 \Big(\sum_{j=1}^q (j\xi_j t^{i+j-1} - \lambda\, \xi_j t^{i+j})\Big) dt = \lambda u_0 \int_0^1 t^i dt, \quad i = 1, 2, \ldots, q. \quad (2.3.3)$$

Performing the integration (ξ_j:s are constants independent of t) we get

$$\sum_{j=1}^q \xi_j \Big[j \cdot \frac{t^{i+j}}{i+j} - \lambda \frac{t^{i+j+1}}{i+j+1}\Big]_{t=0}^{t=1} = \Big[\lambda \cdot u_0 \frac{t^{i+1}}{i+1}\Big]_{t=0}^{t=1}, \quad (2.3.4)$$

or equivalently

$$\sum_{j=1}^q \Big(\frac{j}{i+j} - \frac{\lambda}{i+j+1}\Big)\xi_j = \frac{\lambda}{i+1} \cdot u_0 \quad i = 1, 2, \ldots, q, \quad (2.3.5)$$

which is a linear system of equations with q equations and q unknowns $(\xi_1, \xi_2, \ldots, \xi_q)$; in the coordinates form. In the matrix form (2.3.5) reads

$$\mathcal{A}\Xi = \mathbf{b}, \quad \text{with} \quad \mathcal{A} = (a_{ij}), \quad \Xi = (\xi_j)_{j=1}^q, \quad \text{and} \quad \mathbf{b} = (b_i)_{i=1}^q. \quad (2.3.6)$$

But the matrix $\mathcal{A}$ although invertible, is *ill-conditioned*, i.e. difficult to invert numerically with any accuracy. Mainly because $\{t^i\}_{i=1}^q$ does not form an orthogonal basis. For large i and j the last two rows (columns) of $\mathcal{A}$ computed

from $a_{ij} = \dfrac{j}{i+j} - \dfrac{\lambda}{i+j+1}$, are very close to each other resulting in a very small value for the determinant of $\mathcal{A}$.
If we insist to use polynomial basis up to certain order, then instead of monomials, the use of Legendre orthogonal polynomials would yield a diagonal (sparse) coefficient matrix and make the problem well conditioned. This however, is a rather tedious task. A better approach would be through the use of piecewise polynomial approximations (see Chapter 5) on a partition of $[0, T]$ into subintervals, where we use *low order* polynomial approximations on each subinterval.

The L_2-projection onto a space of polynomials

A polynomial πf interpolating a given function $f(x)$ on an interval (a, b) agrees with point values of f at a certain discrete set of points $x_i \in (a, b)$: $\pi f(x_i) = f(x_i),\ i = 1, \ldots, n$, for some integer n. This concept can be generalized to determine a polynomial Pf so that certain averages agree. These could include the usual average of f over $[a, b]$ defined by,

$$\frac{1}{b-a}\int_a^b f(x)\,dx,$$

or a *generalized average* of f with respect to a *weight function* w defined by

$$\langle f, w\rangle = \int_a^b f(x)w(x)\,dx.$$

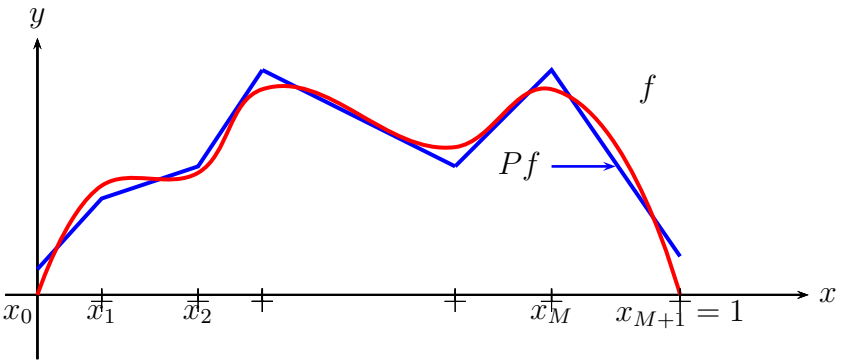


Figure 2.7: An example of a function f and its L_2 projection Pf in $[0, 1]$.

Definition 2.8. *The orthogonal projection, or L_2-projection, of the function f onto $\mathcal{P}^q(a,b)$ is the polynomial $Pf \in \mathcal{P}^q(a,b)$ such that*

$$(f,w) = (Pf,w) \iff (f - Pf, w) = 0 \quad \text{for all } w \in \mathcal{P}^q(a,b). \tag{2.3.7}$$

Thus, (2.3.7) is equivalent to a $(q+1)\times(q+1)$ system of equations.

2.4 A Galerkin method for (BVP)

We consider Galerkin method for the following stationary ($\dot{u} = du/dt = 0$) heat equation in one dimension:

$$-u''(x) = f(x), \quad 0 < x < 1; \qquad u(0) = u(1) = 0. \tag{2.4.1}$$

Let $\mathcal{T}_h : \{jh\}_{j=0}^{M+1}$, $(M+1)h = 1$ be a uniform partition of the interval $[0,1]$ into the subintervals $I_j = ((j-1)h, jh)$, with the same length $|I| = h$, $j = 1, 2, \ldots, M+1$. We define the finite dimensional space V_h^0 by

$$V_h^0 := \{v \in \mathcal{C}(0,1) : v \,\text{is a piecewise linear function on}\, \mathcal{T}_h,\ v(0) = v(1) = 0\},$$

with the basis functions $\{\varphi_j\}_{j=1}^{M}$ defined below (these functions will be used to determine the values of approximate solution at the points x_j, $j = 1, \ldots, M$. Due to the fact that u is known at the boundary points 0 and 1; it is not necessary to supply test functions corresponding to the values at $x_0 = 0$ and $x_{M+1} = 1$. However, in the case of given non-homogeneous boundary data $u(0) = u_0 \neq 0$ and/or $u(1) = u_1 \neq 0$, to represent the trial function, one uses the basis functions to all internal nodes as well as those corresponding to the non-homogeneous data (i.e. at $x = 0$ and/or $x = 1$).

Remark 2.1. *If the Dirichlet boundary condition is given at only one of the boundary points; say $x_0 = 0$ and the other one satisfies, e.g. a Neumann condition as*

$$-u''(x) = f(x), \quad 0 < x < 1; \qquad u(0) = b_0, \quad u'(1) = b_1, \tag{2.4.2}$$

then the function φ_0 (at $x_0 = 0$) will be unnecessary (no matter whether $b_0 = 0$ or $b_0 \neq 0$), whereas one needs to provide the half-base function φ_{M+1} at $x_{M+1} = 1$ (dashed in (2.8) below). Note that, φ_0 participates (as data) in representing the trial function U (see excercises at the end of this chapter).

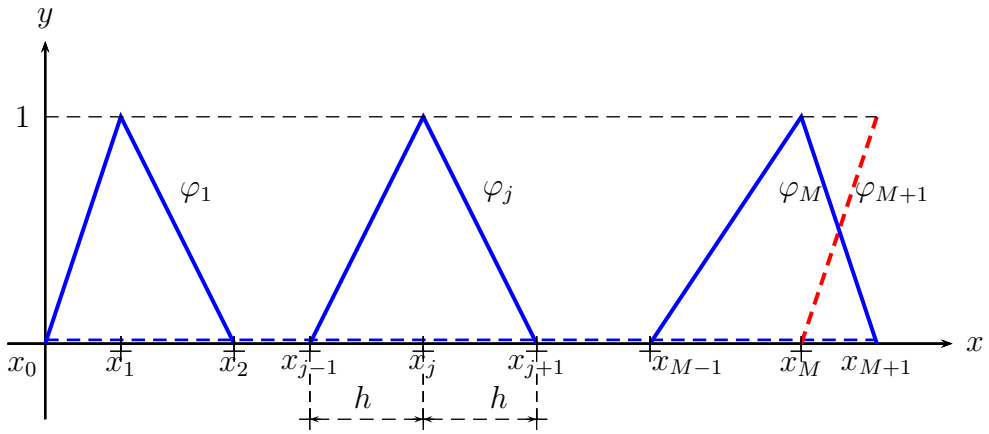


Figure 2.8: Piecewise linear basis functions

Now we define the function space

$$V_0 = H_0^1(0,1) := \left\{ w : \int_0^1 (w(x)^2 + w'(x)^2)\, dx < \infty, \quad w(0) = w(1) = 0 \right\},$$

A variational formulation for problem (2.4.1), is based on multiplying (2.4.1) by a test function $v \in V_0$ and integrating over $[0,1)$:

$$\int_0^1 (-u''(x) - f(x))v(x)dx = 0, \qquad \forall v(x) \in V_0. \tag{2.4.3}$$

Integrating by parts we get

$$-\int_0^1 u''(x)v(x)dx = \int_0^1 u'(x)v'(x)dx - [u'(x)v(x)]_0^1, \tag{2.4.4}$$

and since for $v(x) \in V_0$; $v(0) = v(1) = 0$, we end up with

$$-\int_0^1 u''(x)v(x)dx = \int_0^1 u'(x)v'(x)\, dx. \tag{2.4.5}$$

Thus the *variational formulation* for (2.4.1) is: Find $u \in V_0$ such that

$$\int_0^1 u'(x)v'(x)\, dx = \int_0^1 f(x)v(x)dx, \qquad \forall v \in V_0 \tag{2.4.6}$$

This is a justification for the *finite element formulation*:

The Galerkin finite element method (FEM) for the problem (2.4.1): Find $U(x) \in V_h^0$ such that

$$\int_0^1 U'(x)v'(x)\,dx = \int_0^1 f(x)v(x)dx, \qquad \forall v(x) \in V_h^0. \tag{2.4.7}$$

Thus the Galerkin approximation U is very similar to Pu: The L_2-projection of u. We shall determine $\xi_j = U(x_j)$ which are *the approximate* values of $u(x)$ at the node points $x_j = jh$, $1 \le j \le M$. To this end using basis functions $\varphi_j(x)$, we may write

$$U(x) = \sum_{j=1}^{M} \xi_j\,\varphi_j(x) \quad \text{ which implies that } \quad U'(x) = \sum_{j=1}^{M} \xi_j \varphi_j'(x). \tag{2.4.8}$$

Thus, (2.4.7) can be written as

$$\sum_{j=1}^{M} \xi_j \int_0^1 \varphi_j'(x)\,v'(x)dx = \int_0^1 f(x)v(x)dx, \qquad \forall v(x) \in V_h^0. \tag{2.4.9}$$

Since every $v(x) \in V_h^0$ is a linear combination of the basis functions $\varphi_i(x)$, it suffices to try with $v(x) = \varphi_i(x)$, for $i = 1, 2, \ldots, M$: That is, to find ξ_j (constants), $1 \le j \le M$ such that

$$\sum_{j=1}^{M} \Big(\int_0^1 \varphi_i'(x)\varphi_j'(x)dx \Big)\xi_j = \int_0^1 f(x)\varphi_i(x)dx, \quad i = 1, 2, \ldots, M. \tag{2.4.10}$$

This $M \times M$ system of equations can be written in the matrix form as

$$\mathbf{A}\xi = \mathbf{b}. \tag{2.4.11}$$

Here $\mathbf{A}$ is called the *stiffness matrix* and $\mathbf{b}$ the *load vector*:

$$\mathbf{A} = \{a_{ij}\}_{i,j=1}^{M}, \quad a_{ij} = \int_0^1 \varphi_i'(x)\varphi_j'(x)dx, \tag{2.4.12}$$

$$\mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_M \end{pmatrix}, \text{ with } \quad b_i = \int_0^1 f(x)\varphi_i(x)dx, \text{ and } \xi = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \cdots \\ \xi_M \end{pmatrix}. \tag{2.4.13}$$

To compute the entries a_{ij} of the matrix $\mathbf{A}$, first we need to derive $\varphi_i'(x)$, viz

$$\varphi_i(x) = \begin{cases} \frac{x-(i-1)h}{h} & (i-1)h \le x \le ih \\ \frac{(i+1)h-x}{h} & ih \le x \le (i+1)h \\ 0 & \text{else} \end{cases}$$

$$\varphi_i'(x) = \begin{cases} \frac{1}{h} & (i-1)h < x < ih \\ -\frac{1}{h} & ih < x < (i+1)h \\ 0 & \text{else} \end{cases}$$

Stiffness matrix A:
If $|i-j| > 1$, then φ_i and φ_j have disjoint support, see Figure 2.7, and

$$a_{ij} = \int_0^1 \varphi_i'(x)\varphi_j'(x)dx = 0.$$

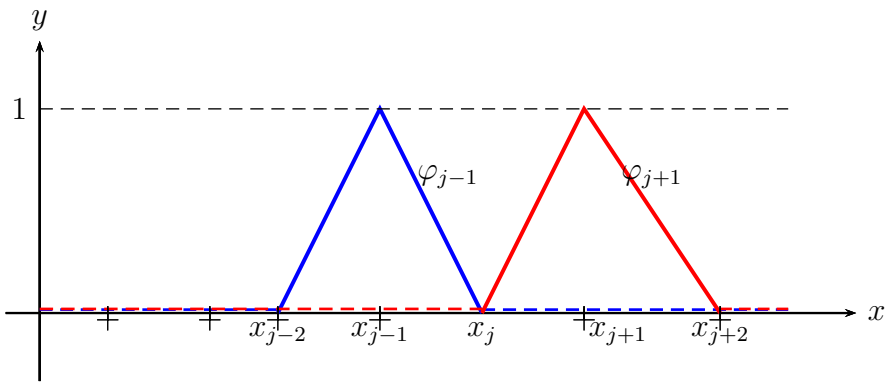


Figure 2.9: φ_{j-1} and φ_{j+1}.

As for $i = j$: we have that

$$a_{ii} = \int_{x_{i-1}}^{x_i} \left(\frac{1}{h}\right)^2 dx + \int_{x_i}^{x_{i+1}} \left(-\frac{1}{h}\right)^2 dx = \frac{\overbrace{x_i - x_{i-1}}^{h}}{h^2} + \frac{\overbrace{x_{i+1} - x_i}^{h}}{h^2} = \frac{1}{h} + \frac{1}{h} = \frac{2}{h}.$$

It remains to compute a_{ij} for the case of (applicable!) $j = i \pm 1$: A straightforward calculation (see the fig below) yields

$$a_{i,i+1} = \int_{x_i}^{x_{i+1}} \Big(-\frac{1}{h} \Big) \cdot \frac{1}{h} dx = -\frac{x_{i+1} - x_i}{h^2} = -\frac{1}{h}. \tag{2.4.14}$$

Obviously $a_{i+1,i} = a_{i,i+1} = -\frac{1}{h}$. To *summarize*, we have

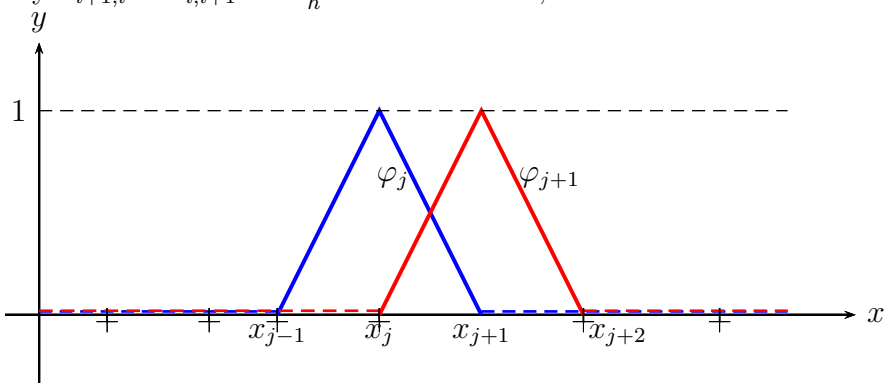


Figure 2.10: φ_j and φ_{j+1}.

$$\begin{cases} a_{ij} = 0, & \text{if } |i-j| > 1, \\ a_{ii} = \frac{2}{h}, & i = 1, 2, \ldots, M, \\ a_{i-1,i} = a_{i,i-1} = -\frac{1}{h}, & i = 2, 3, \ldots, M. \end{cases} \tag{2.4.15}$$

By symmetry $a_{ij} = a_{ji}$, and we finally have the stiffness matrix for approximating the stationary heat conduction by piecewise linear polynomials in a uniform mesh, as:

$$\mathbf{A}_{unif} = \frac{1}{h} \cdot \begin{bmatrix} 2 & -1 & 0 & \ldots & \ldots & 0 \\ -1 & 2 & -1 & 0 & \ldots & \ldots \\ 0 & -1 & 2 & -1 & 0 & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & 0 \\ \ldots & \ldots & 0 & -1 & 2 & -1 \\ 0 & \ldots & \ldots & 0 & -1 & 2 \end{bmatrix}. \tag{2.4.16}$$

As for the components of the load vector **b** we have

$$b_i = \int_0^1 f(x)\varphi_i(x)\,dx = \int_{x_{i-1}}^{x_i} f(x)\frac{x - x_{i-1}}{h}\,dx + \int_{x_i}^{x_{i+1}} f(x)\frac{x_{i+1} - x}{h}\,dx.$$

2.4.1 The nonuniform version

Now let $\widetilde{\mathcal{T}}_h : 0 = x_0 < x_1 < \ldots < x_M < x_{M+1} = 1$ be a partition of the interval $(0,1)$ into nonuniform subintervals $I_j = (x_{j-1}, x_j)$, with lengths $|I_j| = h_j = x_j - x_{j-1}$, $j = 1, 2, \ldots, M+1$. We define the finite dimensional space V_h^0 by

$$V_h^0 := \{v \in \mathcal{C}(0,1) : v \,\text{is a piecewise linear function on}\, \widetilde{\mathcal{T}}_h,\ v(0) = v(1) = 0\},$$

with the nonuniform basis functions $\{\varphi_j\}_{j=1}^M$. To compute the entries a_{ij} of the coefficient matrix **A**, first we need to derive $\varphi_i'(x)$ for the nonuniform basis functions: i.e.,

$$\varphi_i(x) = \begin{cases} \frac{x - x_{i-1}}{h_i} & x_{i-1} \le x \le x_i \\ \frac{x_{i+1} - x}{h_{i+1}} & x_i \le x \le x_{i+1} \\ 0 & \text{else} \end{cases} \implies$$

$$\varphi_i'(x) = \begin{cases} \frac{1}{h_i} & x_{i-1} < x < x_i \\ -\frac{1}{h_{i+1}} & x_i < x < x_{i+1} \\ 0 & \text{else} \end{cases}$$

Nonuniform stiffness matrix A:
If $|i - j| > 1$, then φ_i and φ_j have disjoint support, see Figure 2.9, and

$$a_{ij} = \int_0^1 \varphi_i'(x)\varphi_j'(x)dx = 0.$$

As for $i = j$: we have that

$$a_{ii} = \int_{x_{i-1}}^{x_i} \left(\frac{1}{h_i}\right)^2 dx + \int_{x_i}^{x_{i+1}} \left(-\frac{1}{h_{i+1}}\right)^2 dx = \frac{\overbrace{x_i - x_{i-1}}^{h_i}}{h_i^2} + \frac{\overbrace{x_{i+1} - x_i}^{h_{i+1}}}{h_{i+1}^2} = \frac{1}{h_i} + \frac{1}{h_{i+1}}.$$

For the case of (applicable!) $j = i \pm 1$:

$$a_{i,i+1} = \int_{x_i}^{x_{i+1}} \Big(-\frac{1}{h_{i+1}} \Big) \cdot \frac{1}{h_{i+1}} dx = -\frac{x_{i+1} - x_i}{h_{i+1}^2} = -\frac{1}{h_{i+1}}. \tag{2.4.17}$$

Obviously $a_{i+1,i} = a_{i,i+1} = -\frac{1}{h_{i+1}}$. Thus in nonuniform case we have that

$$\begin{cases} a_{ij} = 0, & \text{if } |i-j| > 1, \\ a_{ii} = \frac{1}{h_i} + \frac{1}{h_{i+1}}, & i = 1, 2, \ldots, M, \\ a_{i-1,i} = a_{i,i-1} = -\frac{1}{h_i}, & i = 2, 3, \ldots, M. \end{cases} \tag{2.4.18}$$

By symmetry $a_{ij} = a_{ji}$, and we finally have the stiffness matrix in nonuniform mesh, for the stationary heat conduction as:

$$\mathbf{A} = \begin{bmatrix} \frac{1}{h_1} + \frac{1}{h_2} & -\frac{1}{h_2} & 0 & \ldots & 0 \\ -\frac{1}{h_2} & \frac{1}{h_2} + \frac{1}{h_3} & -\frac{1}{h_3} & 0 & 0 \\ 0 & \ldots & \ldots & \ldots & 0 \\ \ldots & 0 & \ldots & \ldots & -\frac{1}{h_M} \\ 0 & \ldots & 0 & -\frac{1}{h_M} & \frac{1}{h_M} + \frac{1}{h_{M+1}} \end{bmatrix}. \tag{2.4.19}$$

With a *uniform mesh*, i.e. $h_i = h$ we get that $\mathbf{A} = \mathbf{A}_{unif}$.

Remark 2.2. *Unlike the matrix $\mathcal{A}$ for polynomial approximation of IVP in* (2.3.5)*,* $\mathbf{A}$ *has a more desirable structure, e.g.* $\mathbf{A}$ *is a sparse, tridiagonal and symmetric matrix. This is due to the fact that the basis functions $\{\varphi_j\}_{j=1}^M$ are nearly orthogonal.*

2.5 Exercises

Problem 2.1. *Prove that $V_0^{(q)} := \{v \in P^{(q)}(0,1) : v(0) = 0\}$, is a subspace of $\mathcal{P}^{(q)}(0,1)$.*

Problem 2.2. *Consider the ODE:* $\dot{u}(t) = u(t), \quad 0 < t < 1; \qquad u(0) = 1.$ *Compute its Galerkin approximation in $\mathcal{P}^{(q)}(0,1)$, for $q = 1, 2, 3,$ and 4.*

Problem 2.3. *Consider the ODE:* $\dot{u}(t) = u(t), \quad 0 < t < 1; \qquad u(0) = 1.$ *Compute the* $L_2(0,1)$ *projection of the exact solution* u *into* $\mathcal{P}^3(0,1)$.

Problem 2.4. *Compute the stiffness matrix and load vector in a finite element approximation of the boundary value problem*

$$-u''(x) = f(x), \quad 0 < x < 1, \quad u(0) = u(1) = 0,$$

with $f(x) = x$ *and* $h = 1/4$.

Problem 2.5. *We want to find a solution approximation* $U(x)$ *to*

$$-u''(x) = 1, \quad 0 < x < 1, \quad u(0) = u(1) = 0,$$

using the ansatz $U(x) = A \sin \pi x + B \sin 2\pi x$.

a. *Calculate the exact solution* $u(x)$.

b. *Write down the residual* $R(x) = -U''(x) - 1$

c. *Use the orthogonality condition*

$$\int_0^1 R(x) \sin \pi n x \, dx = 0, \quad n = 1, 2,$$

to determine the constants A *and* B.

d. *Plot the error* $e(x) = u(x) - U(x)$.

Problem 2.6. *Consider the boundary value problem*

$$-u''(x) + u(x) = x, \quad 0 < x < 1, \quad u(0) = u(1) = 0.$$

a. *Verify that the exact solution of the problem is given by*

$$u(x) = x - \frac{\sinh x}{\sinh 1}.$$

b. *Let* $U(x)$ *be a solution approximation defined by*

$$U(x) = A \sin \pi x + B \sin 2\pi x + C \sin 3\pi x,$$

where A, B, *and* C *are unknown constants. Compute the residual function*

$$R(x) = -U''(x) + U(x) - x.$$

c. Use the orthogonality condition

$$\int_0^1 R(x) \sin \pi n x \, dx = 0, \quad n = 1, 2, 3,$$

to determine the constants A, B, and C.

Problem 2.7. *Let $U(x) = \xi_0 \phi_0(x) + \xi_1 \phi_1(x)$ be a solution approximation to*

$$-u''(x) = x - 1, \quad 0 < x < \pi, \quad u'(0) = u(\pi) = 0,$$

where ξ_i, $i = 0, 1$, are unknown coefficients and

$$\phi_0(x) = \cos \frac{x}{2}, \qquad \phi_1(x) = \cos \frac{3x}{2}.$$

a. Find the analytical solution $u(x)$.

b. Define the approximate solution residual $R(x)$.

c. Compute the constants ξ_i using the orthogonality condition

$$\int_0^\pi R(x) \, \phi_i(x) \, dx = 0, \quad i = 0, 1,$$

i.e., by approximating $u(x)$ as a linear combination of $\phi_0(x)$ and $\phi_1(x)$

Problem 2.8. *Use the projection technique of the previous exercises to solve*

$$-u''(x) = 0, \quad 0 < x < \pi, \quad u(0) = 0, \quad u(\pi) = 2,$$

assuming that $U(x) = A \sin x + B \sin 2x + C \sin 3x + \frac{2}{\pi^2} x^2$.

Problem 2.9. *Show that $(f - P_h f, v) = 0$, $\forall v \in V_h$, if and only if $(f - P_h f, \varphi_i) = 0$, $i = 0, \ldots, N$; where $\{\varphi_i\}_{i=1}^N \subset V_h$ is the basis of hat-functions.*

Chapter 3

Interpolation, Numerical Integration in 1d

3.1 Preliminaries

Definition 3.1. *A polynomial interpolant $\pi_q f$ of a function f, defined on an interval $I = [a, b]$, is a polynomial of degree $\leq q$ having the nodal values at $q + 1$ distinct points $x_j \in [a, b]$, $j = 0, 1, \ldots, q$, coinciding with those of f, i.e., $\pi_q f \in \mathcal{P}^q(a, b)$ and $\pi_q f(x_j) = f(x_j),\ j = 0, \ldots, q$.*

Below we illustrate this definition through a simple and familiar example.

Example 3.1. Linear interpolation on an interval. *We start with the unit interval $I := [0, 1]$ and a continuous function $f : I \to \mathbb{R}$. We let $q = 1$ and seek the linear interpolant of f on I, i.e. the linear function $\pi_1 f \in \mathcal{P}^1$, such that $\pi_1 f(0) = f(0)$ and $\pi_1 f(1) = f(1)$. Thus we seek the constants C_0 and C_1 in the following representation of $\pi_1 f \in \mathcal{P}^1$,*

$$\pi_1 f(x) = C_0 + C_1 x, \qquad x \in I, \tag{3.1.1}$$

where

$$\begin{array}{lll} \pi_1 f(0) = f(0) & \Longrightarrow & C_0 = f(0), \quad \textit{and} \\ \pi_1 f(1) = f(1) & \Longrightarrow & C_0 + C_1 = f(1) \Longrightarrow C_1 = f(1) - f(0). \end{array} \tag{3.1.2}$$

Inserting C_0 and C_1 into (3.1.1) *it follows that*

$$\pi_1 f(x) = f(0) + (f(1) - f(0))x = f(0)(1 - x) + f(1)x := f(0)\lambda_0(x) + f(1)\lambda_1(x).$$

In other words $\pi_1 f(x)$ *is represented in two different bases:*

$\pi_1 f(x) = C_0 \cdot 1 + C_1 \cdot x$, *with* $\{1, x\}$ *as the set of basis functions and*

$\pi_1 f(x) = f(0)(1-x)+f(1)x$, *with* $\{1-x, x\}$ *as the set of basis functions. The functions* $\lambda_0(x) = 1 - x$ *and* $\lambda_1(x) = x$ *are linearly independent, since if*

$$0 = \alpha_0(1 - x) + \alpha_1 x = \alpha_0 + (\alpha_1 - \alpha_0)x, \quad \text{for all } x \in I, \tag{3.1.3}$$

then

$$\left.\begin{array}{rcl} x = 0 & \Longrightarrow & \alpha_0 = 0 \\ x = 1 & \Longrightarrow & \alpha_1 = 0 \end{array}\right\} \Longrightarrow \alpha_0 = \alpha_1 = 0. \tag{3.1.4}$$

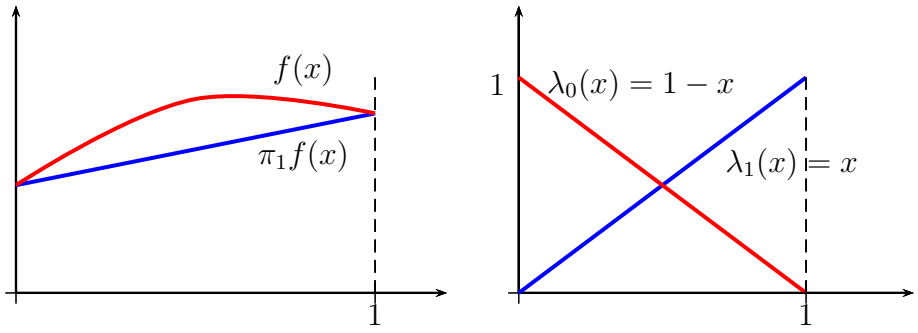


Figure 3.1: Linear interpolation and basis functions for $q = 1$.

Remark 3.1. *Note that if we define a scalar product on* $\mathcal{P}^k(a, b)$ *by*

$$(p, q) = \int_a^b p(x)q(x)\,dx, \qquad \forall p, q \in \mathcal{P}^k(a, b), \tag{3.1.5}$$

then we can easily verify that neither $\{1, x\}$ *nor* $\{1 - x, x\}$ *is an orthogonal basis for* $\mathcal{P}^1(0, 1)$*, since* $(1, x) := \int_0^1 1 \cdot x\,dx = [\frac{x^2}{2}] = \frac{1}{2} \neq 0$ *and* $(1 - x, x) := \int_0^1 (1 - x)x\,dx = \frac{1}{6} \neq 0$.

With such background, it is natural to pose the following question:

Question 3.1. *How well does $\pi_q f$ approximate f? In other words how large/small will the error be in approximating $f(x)$ by $\pi_q f(x)$?*

To answer this question we need to estimate the difference between $f(x)$ and $\pi_q f(x)$. For instance for $q = 1$, geometrically, the deviation of $f(x)$ from $\pi_1 f(x)$ (from being linear) depends on the *curvature* of $f(x)$, i.e. on how *curved* $f(x)$ is. In other words, on how *large* $f''(x)$ is, say, on an interval (a, b). To quantify the relationship between the size of the error $f - \pi_1 f$ and the size of f'', we need to introduce some measuring instrument for vectors and functions:

Definition 3.2. *Let $\mathbf{x} = (x_1, \ldots, x_n)^T$ and $\mathbf{y} = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$ be two column vectors (T stands for transpose). We define the scalar product of $\mathbf{x}$ and $\mathbf{y}$ by*

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y} = x_1 y_1 + \cdots + x_n y_n,$$

and the vector norm for $\mathbf{x}$ as the Euclidean length of $\mathbf{x}$:

$$\|\mathbf{x}\| := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{x_1^2 + \cdots + x_n^2}.$$

$L_p(a, b)$-norm: *Assume that f is a real valued function defined on the interval (a, b). Then we define the L_p-norm ($1 \le p \le \infty$) of f by*

$$L_p\textbf{-norm} \qquad \|f\|_{L_p(a,b)} := \Big(\int_a^b |f(x)|^p dx \Big)^{1/p}, \quad 1 \le p < \infty,$$

$$L_\infty\textbf{-norm} \qquad \|f\|_{L_\infty(a,b)} := \max_{x \in [a,b]} |f(x)|.$$

For $1 \le p \le \infty$ we define the $L_p(a, b)$-space by

$$L_p(a, b) := \{f : \|f\|_{L_p(a,b)} < \infty\}.$$

Below we shall answer Question 3.1, first in the L_∞-norm, and then in the L_p-norm (mainly for $p = 1, 2$.)

Theorem 3.1. *(L_∞-error estimates for linear interpolation in an interval) Assume that $f'' \in L_\infty(a, b)$. Then, for $q = 1$, i.e. only 2 interpolation nodes (e.g. end-points of the interval), there are interpolation constants, C_i, $i = 1, 2, 3.$, independent of the function f and the size of the interval $[a, b]$, such that*

(1) $\|\pi_1 f - f\|_{L_\infty(a,b)} \le C_1 (b - a)^2 \|f''\|_{L_\infty(a,b)}$

(2) $\|\pi_1 f - f\|_{L_\infty(a,b)} \le C_2(b-a)\|f'\|_{L_\infty(a,b)}$

(3) $\|(\pi_1 f)' - f'\|_{L_\infty(a,b)} \le C_3(b-a)\|f''\|_{L_\infty(a,b)}$.

Proof. Note that every linear function, $p(x)$ on $[a,b]$ can be written as a linear combination of the basis functions $\lambda_a(x)$ and $\lambda_b(x)$ where

$$\lambda_a(x) = \frac{b-x}{b-a} \quad \text{and} \quad \lambda_b(x) = \frac{x-a}{b-a}: \tag{3.1.6}$$

$$p(x) = p(a)\lambda_a(x) + p(b)\lambda_b(x). \tag{3.1.7}$$

Recall that linear combinations of $\lambda_a(x)$ and $\lambda_b(x)$ give the basis functions $\{1, x\}$ for $\mathcal{P}^1$:

$$\lambda_a(x) + \lambda_b(x) = 1, \qquad a\lambda_a(x) + b\lambda_b(x) = x. \tag{3.1.8}$$

Here, $\pi_1 f(x)$ being a linear function connecting the two points $(a, f(a))$ and $(b, f(b))$, is represented by

$$\pi_1 f(x) = f(a)\lambda_a(x) + f(b)\lambda_b(x). \tag{3.1.9}$$

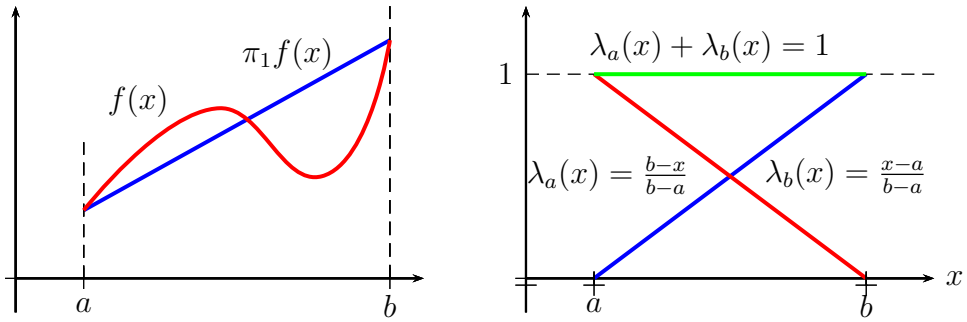


Figure 3.2: Linear Lagrange basis functions for $q = 1$.

By the Taylor expansion for $f(a)$ and $f(b)$ about $x \in (a,b)$ we can write

$$\begin{cases} f(a) = f(x) + (a-x)f'(x) + \dfrac{1}{2}(a-x)^2 f''(\eta_a), & \eta_a \in [a,x] \\ f(b) = f(x) + (b-x)f'(x) + \dfrac{1}{2}(b-x)^2 f''(\eta_b), & \eta_b \in [x,b]. \end{cases} \tag{3.1.10}$$

Inserting $f(a)$ and $f(b)$ from (3.1.10) into (3.1.9), it follows that

$$\begin{aligned}\pi_1 f(x) =&[f(x) + (a-x)f'(x) + \frac{1}{2}(a-x)^2 f''(\eta_a)]\lambda_a(x) + \\ &+[f(x) + (b-x)f'(x) + \frac{1}{2}(b-x)^2 f''(\eta_b)]\lambda_b(x).\end{aligned}$$

Rearranging the terms, using (3.1.8) and the identity (which also follows from (3.1.8)) $(a-x)\lambda_a(x) + (b-x)\lambda_b(x) = 0$ we get

$$\begin{aligned}\pi_1 f(x) &= f(x)[\lambda_a(x) + \lambda_b(x)] + f'(x)[(a-x)\lambda_a(x) + (b-x)\lambda_b(x)] + \\ &+ \frac{1}{2}(a-x)^2 f''(\eta_a)\lambda_a(x) + \frac{1}{2}(b-x)^2 f''(\eta_b)\lambda_b(x) = \\ &= f(x) + \frac{1}{2}(a-x)^2 f''(\eta_a)\lambda_a(x) + \frac{1}{2}(b-x)^2 f''(\eta_b)\lambda_b(x).\end{aligned}$$

Consequently

$$|\pi_1 f(x) - f(x)| = \Big|\frac{1}{2}(a-x)^2 f''(\eta_a)\lambda_a(x) + \frac{1}{2}(b-x)^2 f''(\eta_b)\lambda_b(x)\Big|. \quad (3.1.11)$$

To proceed, we note that for $a \le x \le b$ both $(a-x)^2 \le (a-b)^2$ and $(b-x)^2 \le (a-b)^2$, furthermore $\lambda_a(x) \le 1$ and $\lambda_b(x) \le 1$, $\forall\, x \in (a,b)$. Moreover, by the definition of the maximum norm both $|f''(\eta_a)| \le \|f''\|_{L_\infty(a,b)}$, and $|f''(\eta_b)| \le \|f''\|_{L_\infty(a,b)}$. Thus we may estimate (3.1.11) as

$$|\pi_1 f(x) - f(x)| \le \frac{1}{2}(a-b)^2 \cdot 1 \cdot \|f''\|_{L_\infty(a,b)} + \frac{1}{2}(a-b)^2 \cdot 1 \cdot \|f''\|_{L_\infty(a,b)}, \quad (3.1.12)$$

and hence

$$|\pi_1 f(x) - f(x)| \le (a-b)^2 \|f''\|_{L_\infty(a,b)} \quad \text{corresponding to} \quad c_i = 1. \quad (3.1.13)$$

The other two estimates (2) and (3) are proved similarly. □

Remark 3.2. *We can show that the optimal value of* $C_1 = \frac{1}{8}$ *(cf Problem 3.10), i.e. the constant* $C_1 = 1$ *of the proof above is not the optimal one.*

An analogue to Theorem 3.1 can be proved in the L_p-norm, $p = 1,\ 2$. This general version (concisely stated below as Theorem 3.2) is the frequently used L_p-interpolation error estimate.

Theorem 3.2. *Let $\pi_1 v(x)$ be the linear interpolant of the function $v(x)$ on (a,b). Then, assuming that v is twice differentiable ($v \in \mathcal{C}^2(a,b)$), there are interpolation constants c_i, $i = 1,2,3$ such that for $p = 1,\ 2,\ \infty$,*

$$\|\pi_1 v - v\|_{L_p(a,b)} \leq c_1 (b-a)^2 \|v''\|_{L_p(a,b)}, \tag{3.1.14}$$

$$\|(\pi_1 v)' - v'\|_{L_p(a,b)} \leq c_2 (b-a) \|v''\|_{L_p(a,b)}, \tag{3.1.15}$$

$$\|\pi_1 v - v\|_{L_p(a,b)} \leq c_3 (b-a) \|v'\|_{L_p(a,b)}. \tag{3.1.16}$$

For $p = \infty$ this is just the previous Theorem 3.1.

Proof. For $p = 1$ and $p = 2$, the proof uses the integral form of the Taylor expansion and is left as an exercise. □

Below we review a simple piecewise linear interpolation procedure on a partition of an interval:

Vector space of piecewise linear functions on an interval. Given $I = [a,b]$, let $\mathcal{T}_h : a = x_0 < x_1 < x_2 < \ldots < x_{N-1} < x_N = b$ be a partition of I into subintervals $I_j = [x_{j-1}, x_j]$ of length $h_j = |I_j| := x_j - x_{j-1}$; $j = 1,2,\ldots,N$. Let

$$V_h := \{v | v \text{ is a continuous, piecewise linear function on } \mathcal{T}_h\}, \tag{3.1.17}$$

then V_h is a vector space with the previously introduced *hat functions*: $\{\varphi_j\}_{j=0}^N$ as basis functions. Note that $\varphi_0(x)$ and $\varphi_N(x)$ are left and right *half-hat* functions, respectively. We now show that every function in V_h is a linear combination of φ_j:s.

Lemma 3.1. *We have that*

$$\forall v \in V_h; \qquad v(x) = \sum_{j=0}^{N} v(x_j)\varphi_j(x). \tag{3.1.18}$$

Proof. Both the left and right hand side are continuous piecewise linear functions. Thus it suffices to show that they have the same nodal values: Let $x = x_j$, then since $\varphi_i(x_j) = \delta_{ij}$,

$$\begin{aligned} RHS|_{x_j} =& v(x_0)\varphi_0(x_j) + v(x_1)\varphi_1(x_j) + \ldots + v(x_{j-1})\varphi_{j-1}(x_j) \\ &+ v(x_j)\varphi_j(x_j) + v(x_{j+1})\varphi_{j+1}(x_j) + \ldots + v(x_N)\varphi_N(x_j) \\ =& v(x_j) = LHS|_{x_j}. \end{aligned} \tag{3.1.19}$$

□

Definition 3.3. *For a partition* $\mathcal{T}_h : a = x_0 < x_1 < x_2 < \ldots < x_N = b$ *of the interval* $[a, b]$ *we define the mesh function* $h(x)$ *as the piecewise constant function* $h(x) := h_j = x_j - x_{j-1}$ *for* $x \in I_j = (x_{j-1}, x_j),\ j = 1, 2, \ldots, N$.

Definition 3.4. *Assume that* f *is a continuous function in* $[a, b]$. *Then the continuous piecewise linear interpolant of* f *is defined by*

$$\pi_h f(x) = \sum_{j=0}^{N} f(x_j)\varphi_j(x), \qquad x \in [a, b].$$

Here the sub-index h *refers to the mesh function* $h(x)$.

Hence

$$\pi_h f(x_j) = f(x_j), \qquad j = 0, 1, \ldots, N. \tag{3.1.20}$$

Remark 3.3. *Note that we denote the linear interpolant, defined for a single interval* $[a, b]$, *by* $\pi_1 f$ *which is a polynomial of degree 1, whereas the piecewise linear interpolant* $\pi_h f$ *is defined for a partition* $\mathcal{T}_h$ *of* $[a, b]$ *and is a piecewise linear function. For the piecewise polynomial interpolants of (higher) degree* q *we shall use the notation for Cardinal functions of Lagrange interpolation (see Section 3.2).*

Note that for each interval $I_j,\ j = 1, \ldots, N$, we have that

(i) $\pi_h f(x)$ is linear on $I_j \implies \quad \pi_h f(x) = c_0 + c_1 x$ for $x \in I_j$.

(ii) $\pi_h f(x_{j-1}) = f(x_{j-1})$ and $\pi_h f(x_j) = f(x_j)$.

Combining (i) and (ii) we get

$$\begin{cases} \pi_h f(x_{j-1}) = c_0 + c_1 x_{j-1} = f(x_{j-1}) \\ \pi_h f(x_j) = c_0 + c_1 x_j = f(x_j) \end{cases} \implies \begin{cases} c_1 = \frac{f(x_j) - f(x_{j-1})}{x_j - x_{j-1}} \\ c_0 = \frac{-x_{j-1} f(x_j) + x_j f(x_{j-1})}{x_j - x_{j-1}}. \end{cases}$$

Thus, we may write

$$\begin{cases} c_0 = f(x_{j-1})\frac{x_j}{x_j - x_{j-1}} + f(x_j)\frac{-x_{j-1}}{x_j - x_{j-1}} \\ c_1 x = f(x_{j-1})\frac{-x}{x_j - x_{j-1}} + f(x_j)\frac{x}{x_j - x_{j-1}}. \end{cases} \tag{3.1.21}$$

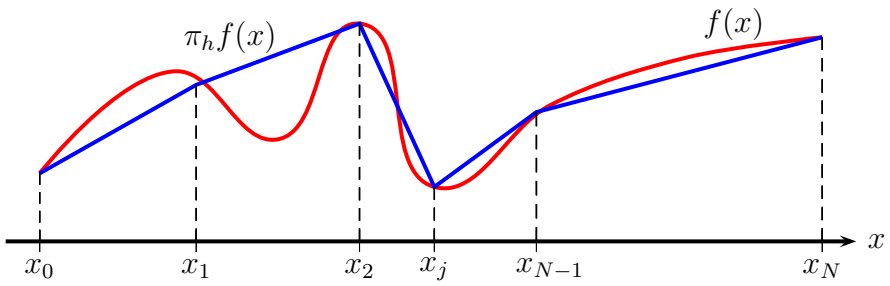


Figure 3.3: Piecewise linear interpolant $\pi_h f(x)$ of $f(x)$.

For $x \in [x_{j-1}, x_j]$, $j = 1, 2, \ldots, N$, adding up the equations in (3.1.21) yields

$$\begin{aligned}\pi_h f(x) = c_0 + c_1 x &= f(x_{j-1})\frac{x_j - x}{x_j - x_{j-1}} + f(x_j)\frac{x - x_{j-1}}{x_j - x_{j-1}} \\ &= f(x_{j-1})\lambda_{j-1}(x) + f(x_j)\lambda_j(x),\end{aligned}$$

where $\lambda_{j-1}(x)$ and $\lambda_j(x)$ are the restrictions of the piecewise linear basis functions $\varphi_{j-1}(x)$ and $\varphi_j(x)$ to I_j.

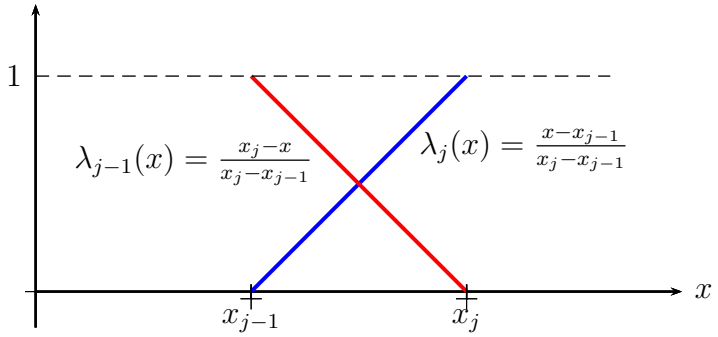


Figure 3.4: Linear Lagrange basis functions for $q = 1$ on the subinterval I_j.

In the next section we shall generalize the above procedure and introduce Lagrange interpolation basis functions.

The main result of this section can be stated as follows:

Theorem 3.3. *Let $\pi_h v(x)$ be the piecewise linear interpolant of the function $v(x)$ on the partition $\mathcal{T}_h$ of $[a,b]$. Then assuming that v is sufficiently regular ($v \in \mathcal{C}^2(a,b)$), there are interpolation constants c_i, $i = 1,2,3$, such that for $p = 1,\ 2,\ \infty$,*

$$\|\pi_h v - v\|_{L_p(a,b)} \leq c_1 \|h^2 v''\|_{L_p(a,b)}, \tag{3.1.22}$$
$$\|(\pi_h v)' - v'\|_{L_p(a,b)} \leq c_2 \|h v''\|_{L_p(a,b)}, \tag{3.1.23}$$
$$\|\pi_h v - v\|_{L_p(a,b)} \leq c_3 \|h v'\|_{L_p(a,b)}. \tag{3.1.24}$$

Proof. Recalling the definition of the partition $\mathcal{T}_h$, we may write

$$\begin{aligned} \|\pi_h v - v\|^p_{L_p(a,b)} &= \sum_{j=1}^{N} \|\pi_h v - v\|^p_{L_p(I_j)} \leq \sum_{j=1}^{N} c_1^p \|h_j^2 v''\|^p_{L_p(I_j)} \\ &\leq c_1^p \|h^2 v''\|^p_{L_p(a,b)}, \end{aligned} \tag{3.1.25}$$

where in the first inequality we apply Theorem 3.2 to an arbitrary partition interval I_j and them sum over j. The other two estimates are proved similarly. □

3.2 Lagrange interpolation

Consider $\mathcal{P}^q(a,b)$; the vector space of all polynomials of degree $\leq q$ on the interval (a,b), with the basis functions $1, x, x^2, \ldots, x^q$. We have seen, in Chapter 2, that this is a non-orthogonal basis (with respect to scalar product (3.1.5) with, e.g. $a = 0$ and $b = 1$) that leads to ill-conditioned coefficient matrices. We will now introduce a new set of basis functions, which being *almost orthogonal* have some useful properties.

Definition 3.5 (Cardinal functions)**.** *Lagrange basis is the set of polynomials $\{\lambda_i\}_{i=0}^q \subset P^q(a,b)$ associated with the $(q+1)$ distinct points, $a = x_0 < x_1 < \ldots < x_q = b$ in $[a,b]$ and determined by the requirement that: at the nodes, $\lambda_i(x_j) = 1$ for $i = j$, and 0 otherwise ($\lambda_i(x_j) = 0$ for $i \neq j$), i.e. for $x \in [a,b]$,*

$$\lambda_i(x) = \frac{(x-x_0)(x-x_1)\ldots(x-x_{i-1}) \downarrow (x-x_{i+1})\ldots(x-x_q)}{(x_i-x_0)(x_i-x_1)\ldots(x_i-x_{i-1}) \uparrow (x_i-x_{i+1})\ldots(x_i-x_q)}. \tag{3.2.1}$$

By the arrows $\downarrow, \uparrow$ *in* (3.2.1) *we want to emphasize that* $\lambda_i(x) = \prod_{j \neq i} \left(\frac{x - x_j}{x_i - x_j} \right)$ *does not contain the singular factor* $\frac{x - x_i}{x_i - x_i}$. *Hence*

$$\lambda_i(x_j) = \frac{(x_j - x_0)(x_j - x_1) \dots (x_j - x_{i-1})(x_j - x_{i+1}) \dots (x_j - x_q)}{(x_i - x_0)(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_q)} = \delta_{ij},$$

and $\lambda_i(x)$, $i = 0, 1, \dots, q$, *is a polynomial of degree* q *on* (a, b) *with*

$$\lambda_i(x_j) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j. \end{cases} \tag{3.2.2}$$

Example 3.2. *Let* $q = 2$, *then we have* $a = x_0 < x_1 < x_2 = b$, *where*

$$i = 1, j = 2 \Rightarrow \lambda_1(x_2) = \frac{(x_2 - x_0)(x_2 - x_2)}{(x_1 - x_0)(x_1 - x_2)} = 0$$

$$i = j = 1 \Rightarrow \lambda_1(x_1) = \frac{(x_1 - x_0)(x_1 - x_2)}{(x_1 - x_0)(x_1 - x_2)} = 1.$$

A polynomial $P(x) \in \mathcal{P}^q(a, b)$ with the values $p_i = P(x_i)$ at the nodes x_i, $i = 0, 1, \dots, q$, can be expressed in terms of the above Lagrange basis as

$$P(x) = p_0\lambda_0(x) + p_1\lambda_1(x) + \dots + p_q\lambda_q(x). \tag{3.2.3}$$

Using (3.2.2), $P(x_i) = p_0\lambda_0(x_i) + p_1\lambda_1(x_i) + \dots + p_i\lambda_i(x_i) + \dots + p_q\lambda_q(x_i) = p_i$.

Recalling definition 3.1, if we choose $a \leq \xi_0 < \xi_1 < \dots < \xi_q \leq b$, as $q + 1$ distinct interpolation nodes on $[a, b]$, then the interpolating polynomial $\pi_q f \in \mathcal{P}^q(a, b)$ satisfies

$$\pi_q f(\xi_i) = f(\xi_i), \qquad i = 0, 1, \dots, q \tag{3.2.4}$$

and the Lagrange formula (3.2.3) for $\pi_q f(x)$ reads as

$$\pi_q f(x) = f(\xi_0)\lambda_0(x) + f(\xi_1)\lambda_1(x) + \dots + f(\xi_q)\lambda_q(x), \quad a \leq x \leq b.$$

Example 3.3. *For* $q = 1$, *we have only the nodes* a *and* b. *Recall that* $\lambda_a(x) = \frac{b - x}{b - a}$ *and* $\lambda_b(x) = \frac{x - a}{b - a}$, *thus as in the introduction in this chapter*

$$\pi_1 f(x) = f(a)\lambda_a(x) + f(b)\lambda_b(x). \tag{3.2.5}$$

Example 3.4. *To interpolate* $f(x) = x^3 + 1$ *by piecewise polynomials of degree 2, in the partition* $x_0 = 0$, $x_1 = 1$, $x_2 = 2$ *of the interval* $[0, 2]$, *we have*

$$\pi_2 f(x) = f(0)\lambda_0(x) + f(1)\lambda_1(x) + f(2)\lambda_2(x),$$

where $f(0) = 1$, $f(1) = 2$, $f(2) = 9$, *and we may compute Lagrange basis as*

$$\lambda_0(x) = \frac{1}{2}(x-1)(x-2), \quad \lambda_1(x) = -x(x-2), \quad \lambda_2(x) = \frac{1}{2}x(x-1).$$

This yields

$$\pi_2 f(x) = 1 \cdot \frac{1}{2}(x-1)(x-2) - 2 \cdot x(x-2) + 9 \cdot \frac{1}{2}x(x-1) = 3x^2 - 2x + 1.$$

3.3 Numerical integration, Quadrature rules

In the finite element approximation procedure of solving differential equations, with a given source term (data) $f(x)$, we need to evaluate integrals of the form $\int f(x)\varphi_i(x)\,dx$, with $\varphi_i(x)$ being a finite element basis function. Such integrals are not easily computable for higher order approximations (e.g. with φ_i:s being Lagrange basis of high order) and more involved data. Further, we encounter matrices with entries being the integrals of products of these, higher order, basis functions and their derivatives. Except some special cases (see calculations for $\mathbf{A}$ and $\mathbf{A}_{unif}$ in the previous chapter), such integrations are usually performed approximately by using numerical methods. Below we briefly review some of these numerical integration techniques.

We approximate the integral $I = \int_a^b f(x)dx$ using a partition of the interval $[a, b]$ into subintervals, where on each subinterval f is approximated by polynomials of a certain degree d. We shall denote the approximate value of the integral I by I_d. To proceed we assume, without loss of generality, that $f(x) > 0$ on $[a, b]$ and that f is continuous on (a, b). Then the integral $I = \int_a^b f(x)dx$ is interpreted as the area of the domain under the curve $y = f(x)$; limited by the x-axis and the lines $x = a$ and $x = b$. We shall approximate this area using the values of f at certain points as follows.

We start by approximating the integral over a single interval $[a, b]$. These rules are referred to as *simple rules*.

i) Simple midpoint rule uses the value of f at the midpoint $\bar{x} := \frac{a+b}{2}$ of $[a, b]$, i.e. $f\left(\frac{a+b}{2}\right)$. This means that f is approximated by the constant function

(polynomial of degree 0) $P_0(x) = f\Big(\frac{a+b}{2}\Big)$ and the area under the curve $y = f(x)$ by

$$I = \int_a^b f(x)dx \approx (b-a)f\Big(\frac{a+b}{2}\Big). \tag{3.3.1}$$

To prepare for generalizations, if we let $x_0 = a$ and $x_1 = b$ and assume that the length of the interval is h, then

$$I \approx I_0 = hf\Big(a + \frac{h}{2}\Big) = hf(\bar{x}) \tag{3.3.2}$$

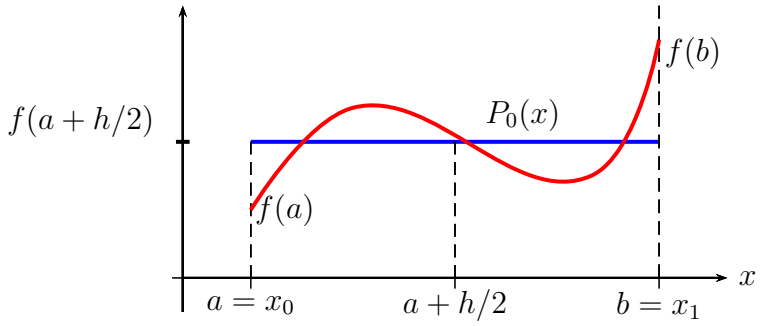


Figure 3.5: Midpoint approximation I_0 of the integral $I = \int_{x_0}^{x_1} f(x)dx$.

ii) Simple trapezoidal rule uses the values of f at two endpoints a and b, i.e. $f(a)$ and $f(b)$. Here f is approximated by the linear function (polynomial of degree 1) $P_1(x)$ passing through the two points $\Big(a, f(a)\Big)$ and $\Big(b, f(b)\Big)$. Consequently, the area under the curve $y = f(x)$ is approximated as

$$I = \int_a^b f(x)dx \approx (b-a)\frac{f(a)+f(b)}{2}. \tag{3.3.3}$$

This is the area of the trapezoidal between the lines $y = 0$, $x = a$ and $x = b$ and under the graph of $P_1(x)$, and therefore is referred to as the *simple trapezoidal rule*. Once again, for the purpose of generalization, we let $x_0 = a$, $x_1 = b$ and assume that the length of the interval is h, then (3.3.3) can be

written as

$$\begin{aligned} I \approx I_1 =& hf(a) + \frac{h[f(a+h) - f(a)]}{2} = h\frac{f(a) + f(a+h)}{2} \\ \equiv& \frac{h}{2}[f(x_0) + f(x_1)]. \end{aligned} \tag{3.3.4}$$

iii) Simple Simpson's rule uses the values of f at the two endpoints a and b,

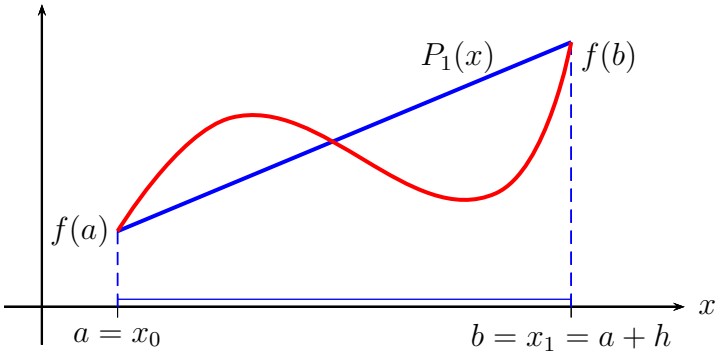


Figure 3.6: Trapezoidal approximation I_1 of the integral $I = \int_{x_0}^{x_1} f(x)dx$.

and the midpoint $\frac{a+b}{2}$ of the interval $[a, b]$, i.e. $f(a)$, $f(b)$, and $f\left(\frac{a+b}{2}\right)$. In this case the area under $y = f(x)$ is approximated by the area under the graph of the second degree polynomial $P_2(x)$; with $P_2(a) = f(a)$, $P_2\left(\frac{a+b}{2}\right) = f\left(\frac{a+b}{2}\right)$, and $P_2(b) = f(b)$. To determine $P_2(x)$ we may use Lagrange interpolation for $q = 2$: let $x_0 = a$, $x_1 = (a + b)/2$ and $x_2 = b$, then

$$P_2(x) = f(x_0)\lambda_0(x) + f(x_1)\lambda_1(x) + f(x_2)\lambda_2(x), \tag{3.3.5}$$

where

$$\begin{cases} \lambda_0(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}, \\ \lambda_1(x) = \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}, \\ \lambda_2(x) = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}. \end{cases} \tag{3.3.6}$$

Thus

$$I = \int_a^b f(x)dx \approx \int_a^b P_2(x)\,dx = \sum_{i=0}^{2} f(x_i) \int_a^b \lambda_i(x)\,dx. \tag{3.3.7}$$

Now we can easily compute the integrals

$$\int_a^b \lambda_0(x)\,dx = \int_a^b \lambda_2(x)\,dx = \frac{b-a}{6}, \quad \int_a^b \lambda_1(x)\,dx = \frac{4(b-a)}{6}. \tag{3.3.8}$$

Hence

$$I = \int_a^b f(x)dx \approx I_2 = \frac{b-a}{6}[f(x_0) + 4f(x_1) + f(x_2)]. \tag{3.3.9}$$

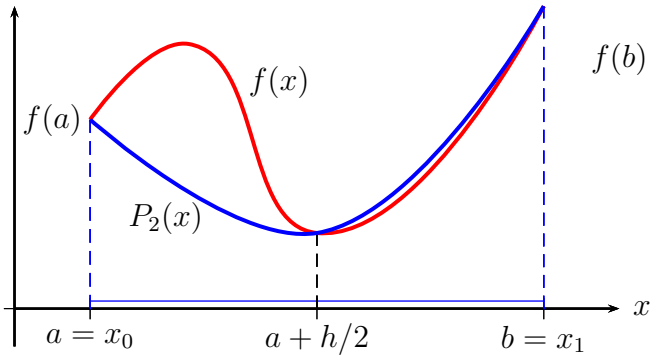


Figure 3.7: Simpson's rule approximation I_2 of the integral $I = \int_{x_0}^{x_1} f(x)dx$.

Obviously these approximations are less accurate for large intervals, $[a, b]$ and/or oscillatory functions f. Following Riemann's idea we can use these rules, instead of on the whole interval $[a, b]$, for the subintervals in an appropriate partition of $[a, b]$. Then we get the following generalized versions.

3.3.1 Composite rules for uniform partitions

We shall use the following *General algorithm* to approximate the integral

$$I = \int_a^b f(x)dx.$$

(1) Divide the interval $[a, b]$, uniformly, into N subintervals

$$a = x_0 < x_1 < x_2 < \ldots < x_{N-1} < x_N = b. \tag{3.3.10}$$

(2) Write the integral as

$$\int_a^b f(x)dx = \int_{x_0}^{x_1} f(x)\,dx + \ldots + \int_{x_{N-1}}^{x_N} f(x)\,dx = \sum_{k=1}^{N} \int_{x_{k-1}}^{x_k} f(x)\,dx. \tag{3.3.11}$$

(3) For each subinterval $I_k := [x_{k-1}, x_k]$, $k = 1, 2, \ldots, N$, apply the *same* integration rule $(i) - (iii)$. Then we get the following generalizations.

(M) *Composite midpoint rule:* approximates f by constants (the values of f at the midpoint of the subinterval) on each subinterval. Let

$$h = |I_k| = \frac{b-a}{N}, \qquad \text{and} \quad \bar{x}_k = \frac{x_{k-1} + x_k}{2}, \quad k = 1, 2, \ldots, N.$$

Then, using the simple midpoint rule for the interval $I_k := [x_{k-1}, x_k]$,

$$\int_{x_{k-1}}^{x_k} f(x)\,dx \approx \int_{x_{k-1}}^{x_k} f(\bar{x}_k)\,dx = hf(\bar{x}_k). \tag{3.3.12}$$

Summing over k, we get the Composite midpoint rule as:

$$\int_a^b f(x)dx \approx \sum_{k=1}^{N} hf(\bar{x}_k) = h[f(\bar{x}_1) + \ldots + f(\bar{x}_N)] := M_N. \tag{3.3.13}$$

(T) *Composite trapezoidal rule*: approximates f by simple trapezoidal rule on each subinterval I_k,

$$\int_{x_{k-1}}^{x_k} f(x)\,dx \approx \frac{h}{2}[f(x_{k-1}) + f(x_k)]. \tag{3.3.14}$$

Summing over k yields the composite trapezoidal rule

$$\begin{aligned} \int_a^b f(x)dx &\approx \sum_{k=1}^{N} \frac{h}{2}[f(x_{k-1}) + f(x_k)] \\ &= \frac{h}{2}[f(x_0) + 2f(x_1) + \ldots + 2f(x_{N-1}) + f(x_N)] := T_N. \end{aligned} \tag{3.3.15}$$

(S) *Composite Simpson's rule*: approximates f by simple Simpson's rule on each subinterval I_k,

$$\int_{x_{k-1}}^{x_k} f(x)\,dx \approx \frac{h}{6}\Big[f(x_{k-1}) + 4f\Big(\frac{x_{k-1}+x_k}{2}\Big) + f(x_k)\Big]. \tag{3.3.16}$$

To simplify, we introduce the following identification on each I_k:

$$z_{2k-2} = x_{k-1}, \quad z_{2k-1} = \frac{x_{k-1}+x_k}{2} := \bar{x}_k, \quad z_{2k} = x_k, \quad h_z = \frac{h}{2}. \tag{3.3.17}$$

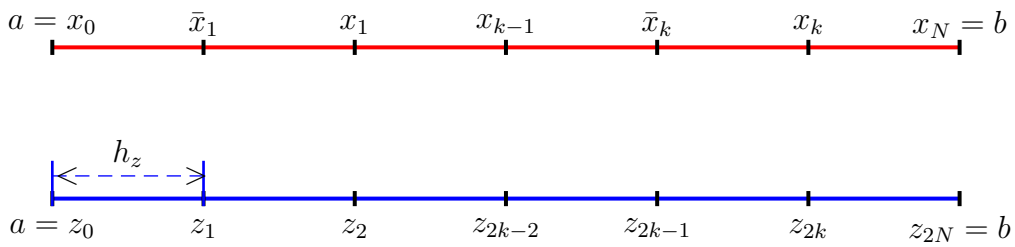


Figure 3.8: Identification of subintervals for composite Simpson's rule

Then, summing (3.3.16) over k and using the above identification, we obtain the composite Simpson's rule viz,

$$\begin{aligned}
\int_a^b f(x)dx &\approx \sum_{k=1}^{N} \frac{h}{6}\Big[f(x_{k-1}) + 4f\Big(\frac{x_{k-1}+x_k}{2}\Big) + f(x_k)\Big] \\
&= \sum_{k=1}^{N} \frac{h_z}{3}\Big[f(z_{2k-2}) + 4f(z_{2k-1}) + f(z_{2k})\Big] \\
&= \frac{h_z}{3}\Big[f(z_0) + 4f(z_1) + 2f(z_2) + 4f(z_3) + 2f(z_4) \\
&\quad + \ldots + 2f(z_{2N-2}) + 4f(z_{2N-1}) + f(z_{2N})\Big] := S_N.
\end{aligned} \tag{3.3.18}$$

The figure below illustrates the starting procedure for the composite Simpson's rule. The numbers in the brackets indicate the actual coefficients on each subinterval. For instance the end of the first interval: $x_1 = z_2$, coincides with the start of the second interval, ending to the add-up $[1] + [1] = 2$ as the coefficient of $f(z_2)$. This is the case for each interior node x_k, i.e. z_{2k}:s; $k = 1, \ldots, N-1$.

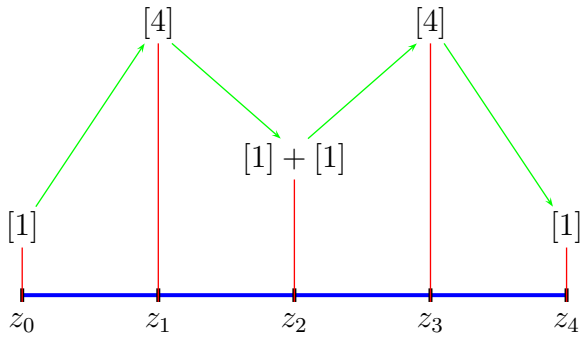


Figure 3.9: Coefficients for composite Simpson's rule

Remark 3.4. *One can verify that the errors of these integration rules are depending on the regularity of the function and the size of interval (in simple rules) and the mesh size (in the composite rules). These error estimates, for both simple and composite quadrature rules, can be found in any elementary text book in numerical linear algebra and/or numerical analysis are read as follows:*

Eroor in simple Midpoint rule

$$|\int_{x_{k-1}}^{x_k} f(x)\,dx - hf(\bar{x}_k)| = \frac{h^3}{24}|f''(\eta)|, \qquad \eta \in (x_{k-1}, x_k).$$

Error in composite Midpoint rule

$$|\int_a^b f(x)\,dx - M_N| = \frac{h^2(b-a)}{24}|f''(\xi)|, \qquad \xi \in (a, b).$$

Eroor in simple trapezoidal rule

$$|\int_{x_{k-1}}^{x_k} f(x)\,dx - \frac{h}{2}[f(\bar{x}_{k-1} + f(x_k)]| = \frac{h^3}{12}|f''(\eta)|, \qquad \eta \in (x_{k-1}, x_k).$$

Error in composite trapezoidal rule

$$|\int_a^b f(x)\,dx - T_N| = \frac{h^2(b-a)}{12}|f''(\xi)|, \qquad \xi \in (a,b).$$

Eroor in simple Simpson's rule

$$|\int_a^b f(x)\,dx - \frac{b-a}{6}[f(a)+4f((a+b)/2)+f(b)]| = \frac{1}{90}\Big(\frac{b-a}{2}\Big)^5|f^{(4)}(\eta)|, \qquad \eta \in (a,b).$$

Error in composite Simpson's rule

$$|\int_a^b f(x)\,dx - S_N| = \frac{h^4(b-a)}{180}\max_{\xi\in[a,b]}|f^4(\xi)|, \quad h=(b-a)/N.$$

Remark 3.5. *The rules (M), (T) and (S) use values of the function at equally spaced points. These are not always the best approximation methods. Below we introduce a general and more optimal approach.*

3.3.2 Gauss quadrature rule

This is an approximate integration rule aimed to choose the points of evaluation of an integrand f in an *optimal* manner, not necessarily at equally spaced points. Here, we illustrate this rule by an example:

Problem: Choose the nodes $x_i \in [a,b]$, and coefficients c_i, $1 \le i \le n$ such that, for an arbitrary integrable function f, the following error is minimal:

$$\int_a^b f(x)dx - \sum_{i=1}^n c_i f(x_i). \tag{3.3.19}$$

Solution. The relation (3.3.19) contains $2n$ unknowns consisting of n nodes x_i and n coefficients c_i. Therefore we need $2n$ equations. Thus if we replace f by a polynomial, then an optimal choice of these $2n$ parameters yields a quadrature rule (3.3.19) which is *exact* for *polynomials*, f, of *degree* $\le 2n-1$.

Example 3.5. *Let $n=2$ and $[a,b]=[-1,1]$. Then the coefficients are c_1 and c_2 and the nodes are x_1 and x_2. Thus optimal choice of these 4 parameters should yield that the approximation*

$$\int_{-1}^1 f(x)dx \approx c_1 f(x_1) + c_2 f(x_2), \tag{3.3.20}$$

is indeed exact for $f(x)$ replaced by any polynomial of degree ≤ 3. So, we replace f by a polynomial of the form $f(x) = Ax^3+Bx^2+Cx+D$ and require equality in (3.3.20)*. Thus, to determine the coefficients c_1, c_2 and the nodes x_1, x_2, in an optimal way, it suffices to change the above approximation to equality when f is replaced by the basis functions for polynomials of degree ≤ 3: i.e., $1, x, x^2$ and x^3. Consequently we get the equation system*

$$\begin{aligned}
\int_{-1}^{1} 1dx = c_1 + c_2 \qquad &\Longrightarrow [x]_{-1}^{1} = 2 = c_1 + c_2 \\
\int_{-1}^{1} xdx = c_1 \cdot x_1 + c_2 \cdot x_2 &\Longrightarrow \Big[\frac{x^2}{2}\Big]_{-1}^{1} = 0 = c_1 \cdot x_1 + c_2 \cdot x_2 \\
\int_{-1}^{1} x^2dx = c_1 \cdot x_1^2 + c_2 \cdot x_2^2 &\Longrightarrow \Big[\frac{x^3}{3}\Big]_{-1}^{1} = \frac{2}{3} = c_1 \cdot x_1^2 + c_2 \cdot x_2^2 \\
\int_{-1}^{1} x^3dx = c_1 \cdot x_1^3 + c_2 \cdot x_2^3 &\Longrightarrow \Big[\frac{x^4}{4}\Big]_{-1}^{1} = 0 = c_1 \cdot x_1^3 + c_2 \cdot x_2^3,
\end{aligned} \tag{3.3.21}$$

which, although nonlinear, has the unique solution presented below:

$$\begin{cases} c_1 + c_2 = 2 \\ c_1x_1 + c_2x_2 = 0 \\ c_1x_1^2 + c_2x_2^2 = \frac{2}{3} \\ c_1x_1^3 + c_2x_2^3 = 0 \end{cases} \Longrightarrow \begin{cases} c_1 = 1 \\ c_2 = 1 \\ x_1 = -\frac{\sqrt{3}}{3} \\ x_2 = \frac{\sqrt{3}}{3}. \end{cases} \tag{3.3.22}$$

Hence, the approximation

$$\int_{-1}^{1} f(x)dx \approx c_1f(x_1) + c_2f(x_2) = f\Big(-\frac{\sqrt{3}}{3}\Big) + f\Big(\frac{\sqrt{3}}{3}\Big), \tag{3.3.23}$$

is exact for all polynomials of degree ≤ 3.

Example 3.6. *Let $f(x) = 3x^2 + 2x + 1$. Then $\int_{-1}^{1}(3x^2 + 2x + 1)dx = [x^3 + x^2 + x]_{-1}^{1} = 4$, and we can easily check that $f(-\sqrt{3}/3) + f(\sqrt{3}/3) = 4$.*

Exercises

Problem 3.1. *Use the expressions $\lambda_a(x) = \frac{b-x}{b-a}$ and $\lambda_b(x) = \frac{x-a}{b-a}$ to show*

$$\lambda_a(x) + \lambda_b(x) = 1, \qquad \text{and} \quad a\lambda_a(x) + b\lambda_b(x) = x.$$

Give a geometric interpretation by plotting, $\lambda_a(x)$, $\lambda_b(x)$, $\lambda_a(x) + \lambda_b(x)$, $a\lambda_a(x)$, $b\lambda_b(x)$ *and* $a\lambda_a(x) + b\lambda_b(x)$.

Problem 3.2. *Determine the linear interpolant* $\pi_1 f \in \mathcal{P}^1(0,1)$ *and plot* f *and* $\pi_1 f$ *in the same figure, when*
(a) $f(x) = x^2$, *(b)* $f(x) = \sin(\pi x)$.

Problem 3.3. *Determine the linear interpolation of the function*

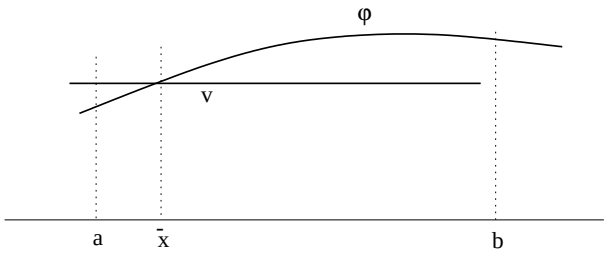
$$f(x) = \frac{1}{\pi^2}(x-\pi)^2 - \cos^2(x - \frac{\pi}{2}), \qquad -\pi \le x \le \pi.$$

where the interval $[-\pi, \pi]$ *is divided into 4 equal subintervals.*

Problem 3.4. *Assume that* $w' \in L_1(I)$. *Let* $x, \bar{x} \in I = [a,b]$ *and* $w(\bar{x}) = 0$. *Show that*

$$|w(x)| \le \int_I |w'| dx. \tag{3.3.24}$$

Problem 3.5. *Let now* $v(t)$ *be the constant interpolant of* φ *on* I.



Show that

$$\int_I h^{-1}|\varphi - v|\, dx \le \int_I |\varphi'|\, dx. \tag{3.3.25}$$

Problem 3.6. *Show that* $\mathcal{P}^q(a,b) = \{$*the set of polynomials of degree* $\le q\}$, *is a vector space but,* $P^q(a,b) := \{p(x) | p(x)$ *is a polynomial of degree* $= q\}$, *is not a vector space.*

Problem 3.7. *Compute formulas for the linear interpolant of a continuous function* f *through the points* a *and* $(b+a)/2$. *Plot the corresponding Lagrange basis functions.*

Problem 3.8. *Prove the following interpolation error estimate:*

$$||\pi_1 f - f||_{L_\infty(a,b)} \leq \frac{1}{8}(b-a)^2||f''||_{L_\infty(a,b)}.$$

Problem 3.9. *Prove that any value of f on the sub-intervals, in a partition of (a,b), can be used to define $\pi_h f$ satisfying the error bound*

$$||f - \pi_h f||_{L_\infty(a,b)} \leq \max_{1\leq i\leq m+1} h_i||f'||_{L_\infty(I_i)} = ||hf'||_{L_\infty(a,b)}.$$

Prove that choosing the midpoint improves the bound by an extra factor $1/2$.

Problem 3.10. *Compute and graph $\pi_4\Big(e^{-8x^2}\Big)$ on $[-2,2]$, which interpolates e^{-8x^2} at 5 equally spaced points in $[-2,2]$.*

Problem 3.11. *Write down a basis for the set of piecewise quadratic polynomials $W_h^{(2)}$ on a partition $a = x_0 < x_1 < x_2 < \ldots < x_{m+1} = b$ of (a,b) into subintervals $I_i = (x_{i-1}, x_i)$, where*

$$W_h^{(q)} = \{v : v|_{I_i} \in \mathcal{P}^q(I_i), i = 1, \ldots, m+1\}.$$

Note that, a function $v \in W_h^{(2)}$ is not necessarily continuous.

Problem 3.12. *Determine a set of basis functions for the space of continuous piecewise quadratic functions $V_h^{(2)}$ on $I = (a,b)$, where*

$$V_h^{(q)} = \{v \in W_h^{(q)} : v \text{ is continuous on } I\}.$$

Problem 3.13. *Prove that*

$$\int_{x_0}^{x_1} f'\Big(\frac{x_1+x_0}{2}\Big)\Big(x - \frac{x_1+x_0}{2}\Big)\,dx = 0.$$

Problem 3.14. *Prove that*

$$\Big|\int_{x_0}^{x_1} f(x)\,dx - f\Big(\frac{x_1+x_0}{2}\Big)(x_1-x_0)\Big|$$
$$\leq \frac{1}{2}\max_{[x_0,x_1]}|f''|\int_{x_0}^{x_1}\Big(x - \frac{x_1+x_0}{2}\Big)^2 dx \leq \frac{1}{24}(x_1-x_0)^3\max_{[x_0,x_1]}|f''|.$$

Hint: Use Taylor expansion of f about $x = \frac{x_1+x_0}{2}$.

Chapter 4

Two-point boundary value problems

In this chapter we focus on finite element approximation procedure for two-point boundary value problems (BVPs). For each problem we formulate a corresponding variational formulation (VF) and a minimization problem (MP) and prove that the solution to either of BVP, its VF and MP satisfies the other two as well, i.e,

$$(BVP) \text{ " } \Longleftrightarrow \text{ " } (VF) \Longleftrightarrow (MP).$$

The $\Longleftarrow$ in the equivalence " $\Longleftrightarrow$ " is subject to a regularity requirement on the solution up to the order of the underlying PDE.

4.1 A Dirichlet problem

Assume that a horizontal elastic bar which occupies the interval $I := [0, 1]$, is fixed at the end-points. Let $u(x)$ denote the displacement of the bar at a point $x \in I$, $a(x)$ be the *modulus of elasticity*, and $f(x)$ a given *load function*, then one can show that u satisfies the following boundary value problem

$$(BVP) \qquad \begin{cases} -\Big(a(x)u'(x)\Big)' = f(x), \quad 0 < x < 1, \\ u(0) = u(1) = 0. \end{cases} \tag{4.1.1}$$

Equation (4.1.1) is of Poisson's type modelling also the stationary heat flux.

We shall assume that $a(x)$ is piecewise continuous function in $(0, 1)$, bounded for $0 \leq x \leq 1$ and $a(x) > 0$ for $0 \leq x \leq 1$.

Let $v(x)$ and its derivative $v'(x), x \in I$, be square integrable functions, that is: v, $v' \in L_2(0,1)$, and define the L_2-based Sobolev space by

$$H_0^1(0,1) := \Big\{ v(x) : \int_0^1 (v(x)^2 + v'(x)^2)dx < \infty, \quad v(0) = v(1) = 0 \Big\}. \quad (4.1.2)$$

The variational formulation (VF). We multiply the equation in (BVP) by a so called test function $v(x) \in H_0^1(0,1)$ and integrate over $(0,1)$ to obtain

$$-\int_0^1 (a(x)u'(x))'v(x)dx = \int_0^1 f(x)v(x)dx. \quad (4.1.3)$$

Using integration by parts we get

$$-\Big[a(x)u'(x)v(x)\Big]_0^1 + \int_0^1 a(x)u'(x)v'(x)dx = \int_0^1 f(x)v(x)dx. \quad (4.1.4)$$

Now since $v(0) = v(1) = 0$ we have thus obtained the *variational formulation* for the problem (4.1.1) as follows: find $u(x) \in H_0^1$ such that

$$\text{(VF)} \qquad \int_0^1 a(x)u'(x)v'(x)dx = \int_0^1 f(x)v(x)dx, \quad \forall v(x) \in H_0^1. \quad (4.1.5)$$

In other words we have shown that if u satisfies (BVP), then u also satisfies the (VF) above. We write this as (BVP) $\Longrightarrow$ (VF). Now the question is whether the reverse implication is true, i.e. under which conditions can we deduce the implication (VF) $\Longrightarrow$ (BVP)? It appears that this question has an affirmative answer, *provided that the solution u to* (VF) *is twice differentiable*. Then, modulo this regularity requirement, the two problems are indeed equivalent. We prove this in the following theorem.

Theorem 4.1. *The following two properties are equivalent*

i) u satisfies (BVP)

ii) u is twice differentiable and satisfies (VF).

Proof. We have already shown that (BVP) $\Longrightarrow$ (VF).

It remains to prove that (VF) $\Longrightarrow$ (BVP). Integrating by parts on the left hand side in (4.1.5), *assuming that u is twice differentiable*, $f \in C(0,1)$, $a \in C^1(0,1)$, and using $v(0) = v(1) = 0$ we return to the relation (4.1.3):

$$-\int_0^1 (a(x)u'(x))'v(x)dx = \int_0^1 f(x)\ v(x)dx, \qquad \forall v(x) \in H_0^1 \quad (4.1.6)$$

which can be rewritten as

$$\int_0^1 \Big\{ -\Big(a(x)u'(x)\Big)' - f(x)\Big\} v(x)dx = 0, \qquad \forall v(x) \in H_0^1. \tag{4.1.7}$$

To show that u satisfies BVP is equivalent to *claim* that (4.1.7) implies

$$-\Big(a(x)u'(x)\Big)' - f(x) \equiv 0, \qquad \forall x \in (0,1). \tag{4.1.8}$$

Suppose not. Then there exists at least one point $\xi \in (0,1)$, such that

$$-\Big(a(\xi)u'(\xi)\Big)' - f(\xi) \neq 0, \tag{4.1.9}$$

where we may assume, without loss of generality, that

$$-\Big(a(\xi)u'(\xi)\Big)' - f(\xi) > 0 \quad (\text{or } < 0). \tag{4.1.10}$$

Thus, by continuity $\exists \delta > 0$ such that

$$g(x) := -\Big(a(x)u'(x)\Big)' - f(x) > 0, \quad \text{for all } x \in I_\delta := (\xi-\delta, \xi+\delta). \tag{4.1.11}$$

Now, take the test function $v(x)$ in (4.1.7) as the *hat-function* $v^*(x) > 0$,

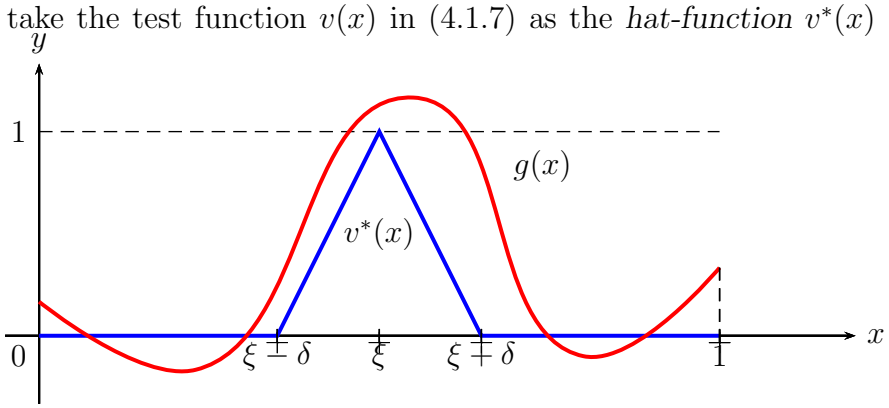


Figure 4.1: The hat function $v^*(x)$ over the interval $(\xi - \delta, \xi + \delta)$.

with $v^*(\xi) = 1$ and the support I_δ, see Fig 4.1. Then $v^*(x) \in H_0^1$ and

$$\int_0^1 \Big\{ -\Big(a(x)u'(x)\Big)' - f(x)\Big\} v^*(x)dx = \int_{I_\delta} \underbrace{g(x)}_{>0}\underbrace{v^*(x)}_{>0} dx > 0.$$

This contradicts (4.1.7). Thus our claim is true. Note further that in (VF) $u \in H_0^1$ implies that $u(0) = u(1) = 0$ and hence we have also the boundary conditions and the proof is complete. □

Corollary 4.1. *(i) If $f(x)$ is continuous and $a(x)$ is continuously differentiable: $f \in C(0,1)$ and $a \in C^1(0,1)$, then (BVP), (VF) have the same solution.*
(ii) If $a(x)$ is discontinuous and $f \in L_2$, then (BVP) is not always well-defined but (VF) has still a meaning. Therefore (VF) covers a larger set of data than (BVP).
(iii) More important: in (VF), $u \in C^1(0,1)$,while (BVP) is formulated for u having two derivatives, i.e. $u \in C^2(0,1)$.

The minimization problem (MP). For the problem (4.1.1), we may formulate yet another equivalent problem, viz:

Find $u \in H_0^1$ such that $F(u) \leq F(w)$, $\forall w \in H_0^1$, where $F(w)$ is the total potential energy of the displacement $w(x)$, given by

$$\text{(MP)} \qquad F(w) = \underbrace{\frac{1}{2}\int_0^1 a(w')^2 dx}_{\text{Internal (elastic) energy}} - \underbrace{\int_0^1 fw dx}_{\text{Load potential}}. \tag{4.1.12}$$

This means that the solution u minimizes the energy functional $F(w)$. Below we show that the above minimization problem is equivalent to the variational formulation (VF) and hence also to the boundary value problem (BVP).

Theorem 4.2. *The following two properties are equivalent*

a) u satisfies the variational formulation (VF)

b) u is the solution for the minimization problem (MP)

i.e.

$$\int_0^1 au'v'dx = \int_0^1 fvdx, \quad \forall v \in H_0^1 \Longleftrightarrow F(u) \leq F(w), \forall w \in H_0^1. \tag{4.1.13}$$

Proof. ($\Longrightarrow$): First we show that the variational formulation (VF) implies the minimization problem (MP). To this end, for $w \in H_0^1$ we let $v = w - u$,

then, since H_0^1 is a vector space and $u \in H_0^1$, hence $v \in H_0^1$ and

$$F(w) = F(u+v) = \frac{1}{2}\int_0^1 a\Big((u+v)'\Big)^2 dx - \int_0^1 f(u+v)dx =$$
$$= \underbrace{\frac{1}{2}\int_0^1 2au'v'dx}_{(i)} + \underbrace{\frac{1}{2}\int_0^1 a(u')^2 dx}_{(ii)} + \frac{1}{2}\int_0^1 a(v')^2 dx$$
$$- \underbrace{\int_0^1 fudx}_{(iii)} - \underbrace{\int_0^1 fvdx}_{(iv)}.$$

Now using (VF) we have $(i) - (iv) = 0$. Further by the definition of the functional F, $(ii) - (iii) = F(u)$. Thus

$$F(w) = F(u) + \frac{1}{2}\int_0^1 a(x)(v'(x))^2 dx, \tag{4.1.14}$$

and since $a(x) > 0$ we get $F(w) \geq F(u)$, thus we have proved " $\Longrightarrow$ " part. ($\Longleftarrow$): Next we show that the minimization problem (MP) implies the variational formulation (VF). To this end, assume that $F(u) \leq F(w)\ \forall w \in H_0^1$, and for an arbitrary function $v \in H_0^1$, set $g_v(\varepsilon) = F(u+\varepsilon v)$, then by (MP), g (as a function of ε) has a *minimum* at $\varepsilon = 0$. In other words $\frac{\partial}{\partial \varepsilon} g_v(\varepsilon)\Big|_{\varepsilon=0} = 0$. We have that

$$g_v(\varepsilon) = F(u+\varepsilon v) = \frac{1}{2}\int_0^1 a\Big((u+\varepsilon v)'\Big)^2 dx - \int_0^1 f(u+\varepsilon v)dx =$$
$$= \frac{1}{2}\int_0^1 \{a(u')^2 + a\varepsilon^2(v')^2 + 2a\varepsilon u'v'\}dx - \int_0^1 fudx - \varepsilon\int_0^1 fvdx.$$

The derivative $\frac{\partial g_v}{\partial \varepsilon}(\varepsilon)$, of $g(\varepsilon, v)$ is

$$\frac{\partial g_v}{\partial \varepsilon}(\varepsilon) = \frac{1}{2}\int_0^1 \{2a\varepsilon(v')^2 + 2au'v'\}dx - \int_0^1 fvdx, \tag{4.1.15}$$

where $\frac{\partial g_v}{\partial \varepsilon}\Big|_{(\varepsilon=0)} = 0$, yields

$$\int_0^1 au'v'dx - \int_0^1 fvdx = 0, \tag{4.1.16}$$

which is our desired variational formulation (VF). Hence, we conclude that $F(u) \leq F(w)$, $\forall w \in H_0^1 \Longrightarrow$ (VF), and the proof is complete. □

We summarize the two theorems in short as

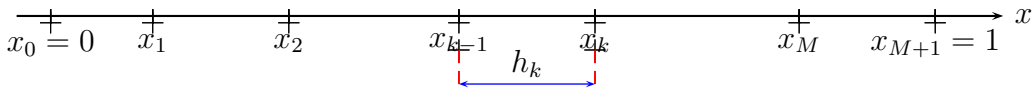
Corollary 4.2.

$$(BVP) \text{ " } \Longleftrightarrow \text{ " } (VF) \Longleftrightarrow (MP).$$

Recall that " $\Longleftrightarrow$ " is a conditional equivalence, requiring u to be twice differentiable, for the reverse implication.

4.2 The finite element method (FEM)

We now formulate the finite element procedure for boundary value problems. To do so we let $\mathcal{T}_h = \{0 = x_0 < x_1 < \ldots < x_M < x_{M+1} = 1\}$ be a partition of the interval $I = [0, 1]$ into subintervals $I_k = [x_{k-1}, x_k]$ and set $h_k = x_k - x_{k-1}$. Define the piecewise *constant* function $h(x) := x_k - x_{k-1} = h_k$ for $x \in I_k$.



Let $\mathcal{C}(I, P_1(I_k))$ denote the set of all continuous piecewise linear functions on $\mathcal{T}_h$ (continuous in the whole interval I, linear on each subinterval I_k), and define

$$V_h^0 = \{v : v \in \mathcal{C}(I, P_1(I_k)), \quad v(0) = v(1) = 0\}. \tag{4.2.1}$$

Note that V_h^0 is a finite dimensional ($dim V_h^0 = M$) subspace of

$$H_0^1 = \Big\{v(x) : \int_0^1 (v(x)^2 + v'(x)^2)dx < \infty, \quad \text{and} \quad v(0) = v(1) = 0\Big\}. \tag{4.2.2}$$

Continuous Galerkin of degree 1, cG(1). A finite element formulation for our Dirichlet boundary value problem (BVP) is given by: find $u_h \in V_h^0$ such that the following discrete variational formulation holds true

$$\text{(FEM)} \qquad \int_0^1 a(x)u_h'(x)v'(x)dx = \int_0^1 f(x)v(x)dx, \quad \forall v \in V_h^0. \tag{4.2.3}$$

The finite element method (FEM) is a finite dimensional version of the variational formulation (VF), where the test functions are in a finite dimensional subspace V_h^0, of H_0^1, spanned by the hat-functions, $\varphi_j(x)$, $j = 1, \ldots, M$.

Thus, if in VF we restrict v to V_h^0 (rather that H_0^1) and subtract FEM from it, we get the *Galerkin orthogonality*:

$$\int_0^1 a(x)(u'(x) - u_h'(x))v'(x)dx = 0, \quad \forall v \in V_h^0. \tag{4.2.4}$$

Now the purpose is to *estimate the error* arising in approximating the solution for BVP by functions in V_h^0. To this approach we need some measuring environment for the error. We recall the definition of L_p-norms:

L_p-norm $\quad \|v\|_{L_p} = \Big(\int_0^1 |v(x)|^p dx \Big)^{1/p}, \quad 1 \le p < \infty$

L_∞-norm $\quad \|v\|_{L_\infty} = \max\limits_{x\in[0,1]} |v(x)|,$

and also introduce:

Weighted L_2-norm $\quad \|v\|_a = \Big(\int_0^1 a(x)|v(x)|^2 dx \Big)^{1/2}, \quad a(x) > 0$

Energy-norm $\quad \|v\|_E = \Big(\int_0^1 a(x)|v'(x)|^2 dx \Big)^{1/2},$

Note that $\quad \|v\|_E = \|v'\|_a.$

4.3 Error estimates in the energy norm

We shall study an *a priori error estimate*; where a certain norm of the error is estimated by some norm of the *exact solution* u. Here, the error analysis gives information about the size of the error, depending on the (unknown) exact solution u, before any computational steps. An *a posteriori error estimate*; where the error is estimated by some norm of the *residual* of the approximate solution is also included.

Below, first we shall prove a qualitative result which states that the finite element solution is the best approximate solution to the Dirichlet problem in the energy norm.

Theorem 4.3. *Let $u(x)$ be the solution to the Dirichlet boundary value problem* (4.1.1) *and $u_h(x)$ its finite element approximation given by* (4.2.3)*, then*

$$\|u - u_h\|_E \le \|u - v\|_E, \qquad \forall v \in V_h^0. \tag{4.3.1}$$

This means that the finite element solution $u_h \in V_h^0$ is the best approximation of the solution u, in the energy norm, by functions in V_h^0.

Proof. We take an arbitrary $v \in V_h^0$, then using the energy norm

$$\begin{aligned}\|u-u_h\|_E^2 &= \int_0^1 a(x)(u'(x)-u_h'(x))^2\,dx \\ &= \int_0^1 a(x)(u'(x)-u_h'(x))(u'(x)-v'(x)+v'(x)-u_h'(x))dx \\ &= \int_0^1 a(x)(u'(x)-u_h'(x))(u'(x)-v'(x))dx \\ &\quad + \int_0^1 a(x)(u'(x)-u_h'(x))(v'(x)-u_h'(x))\,dx.\end{aligned} \tag{4.3.2}$$

Since $v-u_h \in V_h^0$, by Galerkin orthogonality (4.2.4), the last integral is zero. Thus,

$$\begin{aligned}\|u-u_h\|_E^2 &= \int_0^1 a(x)(u'(x)-u_h'(x))(u'(x)-v'(x))dx \\ &= \int_0^1 a(x)^{\frac{1}{2}}(u'(x)-u_h'(x))a(x)^{\frac{1}{2}}(u'(x)-v'(x))dx \\ &\le \Big(\int_0^1 a(x)(u'(x)-u_h'(x))^2dx\Big)^{\frac{1}{2}}\Big(\int_0^1 a(x)(u'(x)-v'(x))^2dx\Big)^{\frac{1}{2}} \\ &= \|u-u_h\|_E\cdot\|u-v\|_E,\end{aligned} \tag{4.3.3}$$

where, in the last estimate, we used Cauchy-Schwarz inequality. Thus

$$\|u-u_h\|_E \le \|u-v\|_E, \qquad \forall v \in V_h^0, \tag{4.3.4}$$

and the proof is complete. □

The next step is to show that there exists a function $v(x) \in V_h^0$ such that $\|u-v\|_E$ is not *too large*. The function that we have in mind is $\pi_h u(x)$: the *piecewise linear interpolant of* $u(x)$, introduced in Chapter 3.

Theorem 4.4. *[An a priori error estimate] Let u and u_h be the solutions of the Dirichlet problem (BVP) and the finite element problem (FEM), respectively. Then there exists an interpolation constant C_i, depending only on $a(x)$, such that*

$$\|u-u_h\|_E \le C_i\|hu''\|_a. \tag{4.3.5}$$

Proof. Since $\pi_h u(x) \in V_h^0$, we may take $v = \pi_h u(x)$ in (4.3.1) and use, e.g. the second estimate in the interpolation Theorem 3.3 (slightly generalized to the weigthed norm $\|\cdot\|_a$, see remark below) to get

$$\begin{aligned} \|u - u_h\|_E &\le \|u - \pi_h u\|_E = \|u' - (\pi_h u)'\|_a \\ &\le C_i \|hu''\|_a = C_i \Big(\int_0^1 a(x) h^2(x) u''(x)^2 \, dx \Big)^{1/2}, \end{aligned} \tag{4.3.6}$$

which is the desired result and the proof is complete. □

Remark 4.1. *The interpolation theorem is not stated in the weighted norm. The $a(x)$ dependence of the interpolation constant C_i can be shown as follows*

$$\begin{aligned} \|u' - (\pi_h u)'\|_a &= \Big(\int_0^1 a(x)(u'(x) - (\pi_h u)'(x))^2 \, dx \Big)^{1/2} \\ &\le \Big(\max_{x\in[0,1]} a(x)^{1/2} \Big) \cdot \|u' - (\pi_h u)'\|_{L_2} \le c_i \Big(\max_{x\in[0,1]} a(x)^{1/2} \Big) \|hu''\|_{L_2} \\ &= c_i \Big(\max_{x\in[0,1]} a(x)^{1/2} \Big) \Big(\int_0^1 h(x)^2 u''(x)^2 \, dx \Big)^{1/2} \\ &\le c_i \frac{(\max_{x\in[0,1]} a(x)^{1/2})}{(\min_{x\in[0,1]} a(x)^{1/2})} \cdot \Big(\int_0^1 a(x) h(x)^2 u''(x)^2 \, dx \Big)^{1/2}. \end{aligned}$$

Thus

$$C_i = c_i \frac{(\max_{x\in[0,1]} a(x)^{1/2})}{(\min_{x\in[0,1]} a(x)^{1/2})}, \tag{4.3.7}$$

where $c_i = c_2$ is the interpolation constant in the second estimate in Theorem 3.3.

Remark 4.2. *If the objective is to divide $[0,1]$ into a finite number of subintervals, then one can use the result of Theorem 4.4: to obtain an optimal partition of $[0,1]$, where whenever $a(x)u''(x)^2$ gets large we compensate by making $h(x)$ smaller. This, however, "requires that the exact solution $u(x)$ is known" [1]. Now we state the a posteriori error estimate, which instead of the unknown solution $u(x)$, uses the residual of the computed solution $u_h(x)$.*

Theorem 4.5 (An a posteriori error estimate)**.** *There is an interpolation constant c_i depending only on $a(x)$ such that the error in the finite element*

[1] Note that when a is a given constant then, $-u''(x) = (1/a)f(x)$ is known.

approximation of the Dirichlet boundary value problem (4.1.10)*, satisfies*

$$\|u-u_h\|_E \le c_i\Big(\int_0^1 \frac{1}{a(x)}h^2(x)R^2(u_h(x))dx\Big)^{1/2}, \tag{4.3.8}$$

where $R(u_h(x)) = f + (a(x)u_h'(x))'$ *is the residual, and* $u(x)-u_h(x) \in H_0^1$.

Proof. By the definition of the *energy norm* we have

$$\begin{aligned}\|e(x)\|_E^2 &= \int_0^1 a(x)(e'(x))^2dx = \int_0^1 a(x)(u'(x)-u_h'(x))e'(x)dx \\ &= \int_0^1 a(x)u'(x)e'(x)dx - \int_0^1 a(x)u_h'(x)e'(x)dx\end{aligned} \tag{4.3.9}$$

Since $e \in H_0^1$ the variational formulation (VF) gives that

$$\int_0^1 a(x)u'(x)e'(x)dx = \int_0^1 f(x)e(x)dx. \tag{4.3.10}$$

Hence, we can write

$$\|e(x)\|_E^2 = \int_0^1 f(x)e(x)dx - \int_0^1 a(x)u_h'(x)e'(x)dx. \tag{4.3.11}$$

Adding and subtracting the interpolant $\pi_h e(x)$ and its derivative $(\pi_h e)'(x)$ to e and e' in the integrands above yields

$$\begin{aligned}\|e(x)\|_E^2 = &\int_0^1 f(x)(e(x)-\pi_h e(x))dx + \underbrace{\int_0^1 f(x)\pi_h e(x)dx}_{(i)} \\ &- \int_0^1 a(x)u_h'(x)(e'(x)-(\pi_h e)'(x))dx - \underbrace{\int_0^1 a(x)u_h'(x)(\pi_h e)'(x)dx}_{(ii)}.\end{aligned}$$

Since $u_h(x)$ is the solution of the (FEM) given by (4.2.3) and $\pi_h e(x) \in V_h^0$ we have that $-(ii)+(i)=0$. Hence

$$\begin{aligned}\|e(x)\|_E^2 &= \int_0^1 f(x)(e(x)-\pi_h e(x))dx - \int_0^1 a(x)u_h'(x)(e'(x)-(\pi_h e)'(x))dx \\ &= \int_0^1 f(x)(e(x)-\pi_h e(x))dx - \sum_{k=1}^{M+1}\int_{x_{k-1}}^{x_k} a(x)u_h'(x)(e'(x)-(\pi_h e)'(x))dx.\end{aligned}$$

To continue we integrate by parts in the integrals in the summation above

$$
\begin{aligned}
&-\int_{x_{k-1}}^{x_k} a(x)u_h'(x)(e'(x)-(\pi_h e)'(x))dx \\
&= -\Big[a(x)u_h'(x)(e(x)-\pi_h e(x))\Big]_{x_{k-1}}^{x_k} + \int_{x_{k-1}}^{x_k} (a(x)u_h'(x))'(e(x)-\pi_h e(x))\,dx.
\end{aligned}
$$

Now, using $e(x_k) = \pi_h e(x_k)$, $k = 0, 1 \ldots, M+1$, where the x_k:s are the interpolation nodes, the boundary terms vanish and thus we end up with

$$
-\int_{x_{k-1}}^{x_k} a(x)u_h'(x)(e'(x)-(\pi_h e)'(x))dx = \int_{x_{k-1}}^{x_k} (a(x)u_h'(x))'(e(x)-\pi_h e(x))dx.
$$

Thus, summing over k, we have

$$
-\int_0^1 a(x)u_h'(x)(e'(x)-(\pi_h e(x))'dx = \int_0^1 (a(x)u_h'(x))'(e(x)-\pi_h e(x))dx,
$$

where $(a(x)u_h'(x))'$ should be interpreted locally on each subinterval $[x_{k-1}, x_k]$. (Since $u_h'(x)$ in general is discontinuous, $u_h''(x)$ does not exist globally on $[0,1]$.) Therefore

$$
\begin{aligned}
\|e(x)\|_E^2 &= \int_0^1 f(x)(e(x)-\pi_h e(x))dx + \int_0^1 (a(x)u_h'(x))'(e(x)-\pi_h e(x))dx \\
&= \int_0^1 \{f(x)+(a(x)u_h'(x))'\}(e(x)-\pi_h e(x))dx.
\end{aligned}
$$

Now let $R(u_h(x)) = f(x) + (a(x)u_h'(x))'$, i.e. $R(u_h(x))$ is the *residual error*, which is a well-defined function except in the set $\{x_k\}$, $k = 1, \ldots, M$; where $(a(x_k)u_h'(x_k))'$ is not defined. Then, using Cauchy-Schwarz' inequality we get the following estimate

$$
\begin{aligned}
\|e(x)\|_E^2 &= \int_0^1 R(u_h(x))(e(x)-\pi_h e(x))dx = \\
&= \int_0^1 \frac{1}{\sqrt{a(x)}} h(x) R(u_h(x)) \cdot \sqrt{a(x)} \Big(\frac{e(x)-\pi_h e(x)}{h(x)}\Big)\,dx \\
&\le \Big(\int_0^1 \frac{1}{a(x)} h^2(x) R^2(u_h(x))dx\Big)^{1/2} \Big(\int_0^1 a(x)\Big(\frac{e(x)-\pi_h e(x)}{h(x)}\Big)^2 dx\Big)^{1/2}.
\end{aligned}
$$

Further, by the definition of the weighted L_2-norm we have,

$$\Big\|\frac{e(x)-\pi_h e(x)}{h(x)}\Big\|_a = \Big(\int_0^1 a(x)\Big(\frac{e(x)-\pi_h e(x)}{h(x)}\Big)^2 dx\Big)^{1/2}. \tag{4.3.12}$$

To estimate (4.3.12) we can use the third interpolation estimate (in Theorem 5.5) for $e(x)$ in each subinterval and get

$$\Big\|\frac{e(x)-\pi_h e(x)}{h(x)}\Big\|_a \le C_i\|e'(x)\|_a = C_i\|e(x)\|_E, \tag{4.3.13}$$

where C_i as before depends on $a(x)$. Thus

$$\|e(x)\|_E^2 \le \Big(\int_0^1 \frac{1}{a(x)}h^2(x)R^2(u_h(x))dx\Big)^{1/2}\cdot C_i\|e(x)\|_E, \tag{4.3.14}$$

and the proof is complete. □

Adaptivity

Below we briefly outline the adaptivity procedure based on the a posteriori error estimate which uses the *approximate* solution and which can be used for mesh-refinements. Loosely speaking, the estimate (4.3.8) predicts local mesh refinement, i.e. indicates the regions (subintervals) which should be subdivided further. More specifically the idea is as follows: assume that one seeks an error less than a given error tolerance TOL > 0:

$$\|e\|_E \le \text{TOL}, \qquad e(x) := u(x) - u_h(x). \tag{4.3.15}$$

Then, one may use the following steps as a mesh refinement strategy:

(i) Make an initial partition of the interval

(ii) Compute the corresponding FEM solution $u_h(x)$ and residual $R(u_h(x))$.

(iii) If $\|e\|_E >$ TOL, refine the mesh in the places where $\dfrac{1}{a(x)}R^2(u_h(x))$ is large and perform the steps (ii) and (iii) again.

4.4 FEM for convection–diffusion–absorption BVPs

We now return to the Galerkin approximation of a solution to boundary value problems and give a framework for the cG(1) (*continuous Galerkin of degree 1*) finite element procedure leading to a linear system of equations of the form $A\xi = \mathbf{b}$. More specifically, we shall extend the approach in Chapter 2, for the stationary heat equation, to cases involving *absorption* and/or *convection* terms. We also consider non-homogeneous Dirichlet boundary conditions. We illustrate this procedure through the following two examples.

Example 4.1. *Determine the coefficient matrix and load vector for the cG(1) finite element approximation of the boundary value problem*

$$-u''(x) + 4u(x) = 0, \quad 0 < x < 1; \qquad u(0) = \alpha \neq 0, \quad u(1) = \beta \neq 0,$$

on a uniform partition $\mathcal{T}_h$ of the interval $[0,\, 1]$ into $n+1$ subintervals.
Solution: The objective is to construct an approximate solution u_h in a finite dimensional space spanned by the piecewise linear basis functions (hat-functions) $\varphi_j(x)$, $j = 0, 1, \ldots, n+1$ on the partition $\mathcal{T}_h$. This results in a discrete problem represented by a linear system of equations $A\xi = \mathbf{b}$, for the unknown $\xi = \{c_j\}_{j=1}^{n}$, ($c_0 = \alpha$ and $c_{n+1} = \beta$ are given in boundary data.)
The continuous solution is assumed to be in the Hilbert space

$$H^1 = \left\{ w : \int_0^1 \left(w(x)^2 + w'(x)^2 \right) dx < \infty \right\}.$$

Since $u(0) = \alpha$ and $u(1) = \beta$ are given, we need to take the trial functions in

$$V := \{ w : w \in H^1, \quad w(0) = \alpha,\ w(1) = \beta \},$$

and the test functions in

$$V^0 := H_0^1 = \{ w : w \in H^1, \quad w(0) = w(1) = 0 \}.$$

We multiply the PDE by a test function $v \in V^0$ and integrate over $(0, 1)$. Integrating by parts we get

$$-u'(1)v(1) + u'(0)v(0) + \int_0^1 u'v'\, dx + 4\int_0^1 uv\, dx = 0 \quad \Longleftrightarrow$$

$$(VF): \quad \textit{Find} \quad u \in V \quad \textit{so that} \quad \int_0^1 u'v'\, dx + 4\int_0^1 uv\, dx = 0, \qquad \forall v \in V^0.$$

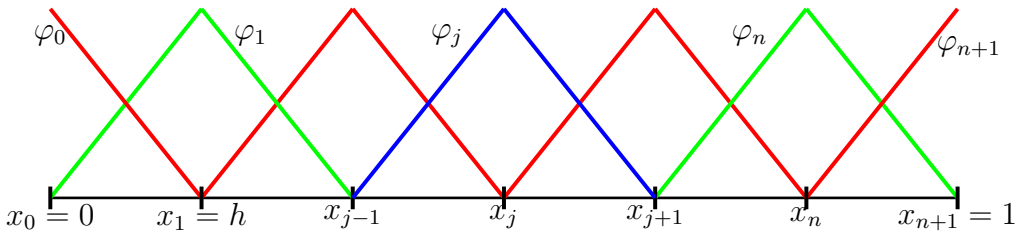
The partition $\mathcal{T}_h$, *of* $[0,1]$ *into* $n+1$ *uniform subintervals* $I_1 = [0,h]$, $I_2 = [h,2h]$, ..., *and* $I_{n+1} = [nh,(n+1)h]$, *is also described by the nodes* $x_0 = 0, x_1 = h, \ldots, x_n = nh$ *and* $x_{n+1} = (n+1)h = 1$. *The corresponding discrete function spaces are (varying with* h *and hence with* n*),*

$$V_h := \{w_h : w_h \text{ is piecewise linear, continuous on } \mathcal{T}_h,\ w_h(0) = \alpha,\ w_h(1) = \beta\},$$

and

$$V_h^0 := \{v_h : v_h \text{ is piecewise linear and continuous on } \mathcal{T}_h,\ \ v_h(0) = v_h(1) = 0\}.$$

Note that here, the basis functions needed to represent functions in V_h *are the hat-functions* $\varphi_j, j = 0, \ldots, n+1$ *(including the two* half-hat-functions *φ_0 and φ_{n+1}), whereas the basis functions describing* V_h^0 *are* φ_i*:s for* $i = 1, \ldots, n$, *i.e. all* full-hat-functions *but not* φ_0 *and* φ_{n+1}. *This is due to the fact that the values* $u(0) = \alpha$ *och* $u(1) = \beta$ *are given and therefore we do not need to determine those two nodal values approximately.*



Now the finite element formulation (the discrete variational formulation) is: find $u_h \in V_h$ *such that*

$$(FEM) \qquad \int_0^1 u_h' v'\,dx + 4\int_0^1 u_h v\,dx = 0, \quad \forall v \in V_h^0.$$

We have that $u_h(x) = c_0\varphi_0(x) + \sum_{j=1}^n c_j\varphi_j(x) + c_{n+1}\varphi_{n+1}(x)$, *where* $c_0 = \alpha$, $c_{n+1} = \beta$ *and*

$$\varphi_0(x) = \frac{1}{h}\begin{cases} h - x & 0 \le x \le h \\ 0, & \text{else} \end{cases}, \quad \varphi_j(x) = \frac{1}{h}\begin{cases} x - x_{j-1}, & x_{j-1} \le x \le x_j \\ x_{j+1} - x & x_j \le x \le x_{j+1} \\ 0 & x \notin [x_{j-1}, x_{j+1}]. \end{cases}$$

and

$$\varphi_{n+1}(x) = \frac{1}{h}\begin{cases} x - x_n & nh \le x \le (n+1)h \\ 0, & \textit{else.} \end{cases}.$$

Inserting u_h into (FEM), and choosing $v = \varphi_i(x)$, $i = 1, \dots, n$ we get

$$\begin{aligned}\sum_{j=1}^{n}\Big(\int_0^1 \varphi_j'(x)\varphi_i'(x)\,dx + 4\int_0^1 \varphi_j(x)\varphi_i(x)\,dx\Big)c_j \\ = -\Big(\int_0^1 \varphi_0'(x)\varphi_i'(x)\,dx + 4\int_0^1 \varphi_0(x)\varphi_i(x)\,dx\Big)c_0 \\ - \Big(\int_0^1 \varphi_{n+1}'(x)\varphi_i'(x)\,dx + 4\int_0^1 \varphi_{n+1}(x)\varphi_i(x)\,dx\Big)c_{n+1}.\end{aligned}$$

In matrix form this corresponds to $A\xi = \mathbf{b}$ with $A = S+4M$, where $S = A_{unif}$ is the, previously computed, stiffness matrix:

$$S = \frac{1}{h}\begin{bmatrix} 2 & -1 & 0 & 0 & \dots & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ 0 & -1 & 2 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & -1 & 2 & -1 \\ 0 & \dots & \dots & \dots & -1 & 2 \end{bmatrix}, \tag{4.4.1}$$

and M is the mass-matrix given by

$$M = \begin{bmatrix} \int_0^1 \varphi_1\varphi_1 & \int_0^1 \varphi_2\varphi_1 & \dots & \int_0^1 \varphi_n\varphi_1 \\ \int_0^1 \varphi_1\varphi_2 & \int_0^1 \varphi_2\varphi_2 & \dots & \int_0^1 \varphi_n\varphi_2 \\ \dots & \dots & \dots & \dots \\ \int_0^1 \varphi_1\varphi_n & \int_0^1 \varphi_2\varphi_n & \dots & \int_0^1 \varphi_n\varphi_n \end{bmatrix}. \tag{4.4.2}$$

Note the index locations in the matrices S and M:

$$s_{ij} = \int_0^1 \varphi_j'(x)\varphi_i'(x)\,dx, \qquad m_{ij} = \int_0^1 \varphi_j(x)\varphi_i(x)\,dx.$$

This, however, does not make any difference in the current example, since, as seen, both S and M are symmetric. To compute the entries of M, we follow the same procedure as in Chapter 2, and notice that, as S, also M is symmetric and its elements m_{ij} are

$$m_{ij} = m_{ji} = \begin{cases} \int_0^1 \varphi_i \varphi_j \, dx = 0, & \forall i,j \quad \textit{with} \quad |i-j| > 1 \\ \int_0^1 \varphi_j^2(x) \, dx, & \textit{for} \quad i = j \\ \int_0^1 \varphi_j(x) \varphi_{j+1}(x), & \textit{for} \quad i = j+1. \end{cases} \tag{4.4.3}$$

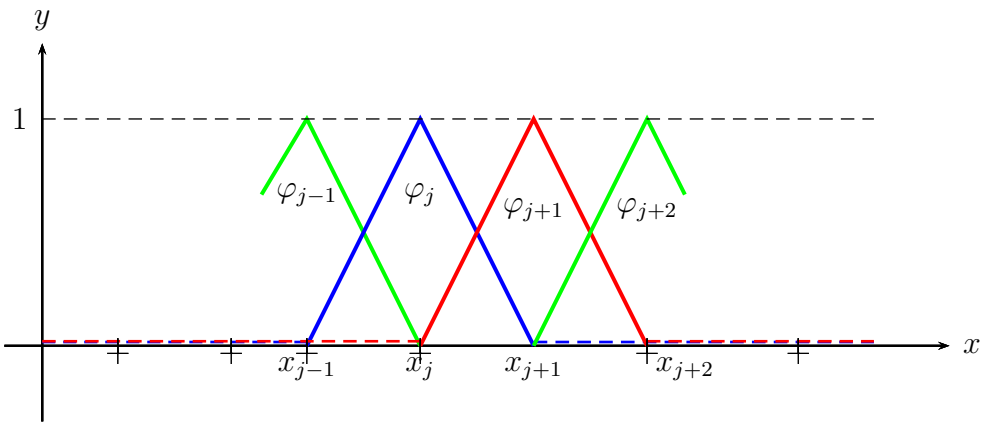


Figure 4.2: φ_j and φ_{j+1}.

The diagonal elements are

$$\begin{aligned} m_{jj} &= \int_0^1 \varphi_j(x)^2 \, dx = \frac{1}{h^2}\Big(\int_{x_{j-1}}^{x_j} (x - x_{j-1})^2 \, dx + \int_{x_j}^{x_{j+1}} (x_{j+1} - x)^2 \Big) \\ &= \frac{1}{h^2}\Big[\frac{(x - x_{j-1})^3}{3} \Big]_{x_{j-1}}^{x_j} - \frac{1}{h^2}\Big[\frac{(x_{j+1} - x)^3}{3} \Big]_{x_j}^{x_{j+1}} \\ &= \frac{1}{h^2} \cdot \frac{h^3}{3} + \frac{1}{h^2} \cdot \frac{h^3}{3} = \frac{2}{3} h, \quad j = 1, \ldots, n, \end{aligned} \tag{4.4.4}$$

and the two super- and sub-diagonals can be computed as

$$\begin{aligned} m_{j,j+1} = m_{j+1,j} &= \int_0^1 \varphi_j \varphi_{j+1}\, dx = \frac{1}{h^2}\int_{x_j}^{x_{j+1}} (x_{j+1}-x)(x-x_j) = [PI] \\ &= \frac{1}{h^2}\Big[(x_{j+1}-x)\frac{(x-x_j)^2}{2}\Big]_{x_j}^{x_{j+1}} - \frac{1}{h^2}\int_{x_j}^{x_{j+1}} -\frac{(x-x_j)^2}{2}\, dx \\ &= \frac{1}{h^2}\Big[\frac{(x-x_j)^3}{6}\Big]_{x_j}^{x_{j+1}} = \frac{1}{6}h, \quad j = 1,\ldots,n-1. \end{aligned}$$

Thus the mass matrix in this case is

$$M = h\begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & \ldots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \ldots & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ 0 & \ldots & \ldots & \ldots & \frac{1}{6} & \frac{2}{3} \end{bmatrix} = \frac{h}{6}\begin{bmatrix} 4 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 4 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 4 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & 1 & 4 & 1 \\ 0 & \ldots & \ldots & \ldots & 1 & 4 \end{bmatrix}.$$

Hence, for $i,j = 1,\ldots,n$, the coefficient matrix $A = S + 4M$ is given by

$$[A]_{ij} = \int_0^1 \varphi_i'\varphi_j'\, dx + 4\int_0^1 \varphi_i\varphi_j(x)\, dx = \begin{cases} \frac{2}{h} + \frac{8h}{3}, & i = j, \\ -\frac{1}{h} + \frac{2h}{3}, & |i-j| = 1, \\ 0 & \textit{else.} \end{cases}$$

Finally, with $c_0 = \alpha$ och $c_{n+1} = \beta$, we get the load vector viz,

$$\begin{aligned} b_1 &= -(-\frac{1}{h} + \frac{2h}{3})c_0 = \alpha(\frac{1}{h} - \frac{2h}{3}), \\ b_2 &= \ldots = b_{n-1} = 0, \\ b_n &= -(-\frac{1}{h} + \frac{2h}{3})c_{n+1} = \beta(\frac{1}{h} - \frac{2h}{3}). \end{aligned}$$

Now, for each particular choice of h (i.e. n), α and β we may solve $A\xi = \mathbf{b}$ to obtain the nodal values of the approximate solution u_h at the inner nodes x_j, $j = 1,\ldots,n$. That is: $\xi = (c_1,\ldots,c_n)^T := (u_h(x_1),\ldots,u_h(x_n))^T$. Connecting the points $(x_j, u_h(x_j))$, $j = 0,\ldots,n+1$ by straight lines we obtain the desired continuous piecewise linear approximation of the solution.

Remark 4.3. *An easier way to compute the above integrals $m_{j,j+1}$ (as well as m_{jj}) is through Simpson's rule, which is exact for polynomials of degree ≤ 2. Since $\varphi_j(x)\varphi_{j+1}(x) = 0$ at $x = x_j$ and $x = x_{j+1}$, we need to evaluate only the midterm of the Simpson's formula, i.e.*

$$\int_0^1 \varphi_j\varphi_{j+1}\,dx = 4\frac{h}{6}\varphi_j\Big(\frac{x_j + x_{j+1}}{2}\Big)\cdot\varphi_{j+1}\Big(\frac{x_j + x_{j+1}}{2}\Big)\cdot = 4\cdot\frac{h}{6}\cdot\frac{1}{2}\cdot\frac{1}{2} = \frac{h}{6}.$$

For a uniform partition one may use $\varphi_0 = 1 - x/h$ and $\varphi_1 = x/h$ on $(0, h)$:

$$\int_0^1 \varphi_0\varphi_1\,dx = \int_0^h (1 - \frac{x}{h})\frac{x}{h}\,dx = \Big[(1 - \frac{x}{h})\frac{x^2}{2h}\Big]_0^h - \int_0^h \frac{(-1)}{h}\cdot\frac{x^2}{2h}\,dx = \frac{h}{6}.$$

Example 4.2. *Below we consider a convection-diffusion problem:*

$$-\varepsilon u''(x) + pu'(x) = r, \quad 0 < x < 1; \qquad u(0) = 0, \quad u'(1) = \beta \neq 0,$$

where ε and p are positive real numbers and $r \in \mathbf{R}$. Here $-\varepsilon u''$ is the diffusion term, pu' corresponds to convection, and r is a given (here for simplicity a constant) source ($r > 0$) or sink ($r < 0$). We would like to answer the same question as in the previous example. This time with $c_0 = u(0) = 0$. Then, the test function at $x = 0$; φ_0 will not be necessary. But since $u(1)$ is not given, we shall need the test function at $x = 1$: φ_{n+1}. The function space for the continuous solution: the trial function space, and the test function space are both the same:

$$V := \left\{ w : \int_0^1 \Big(w(x)^2 + w'(x)^2\Big)\,dx < \infty, \text{ and } w(0) = 0 \right\}.$$

We multiply the PDE by a test function $v \in V$ and integrate over $(0, 1)$. Then, integration by parts yields

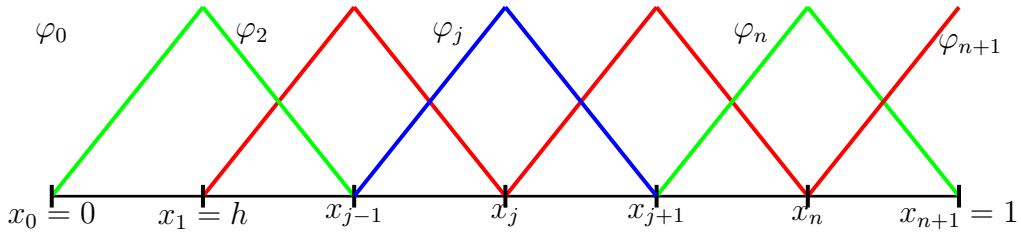
$$-\varepsilon u'(1)v(1) + \varepsilon u'(0)v(0) + \varepsilon\int_0^1 u'v'dx + p\int_0^1 u'vdx = r\int_0^1 v\,dx.$$

Hence, we end up with the variational formulation: find $u \in V$ such that

$$\text{(VF)} \qquad \varepsilon\int_0^1 u'v'\,dx + p\int_0^1 u'v\,dx = r\int_0^1 v\,dx + \varepsilon\beta v(1), \qquad \forall v \in V.$$

The corresponding discrete test and trial function space is

$$V_h^0 := \{w_h : w_h \text{ is piecewise linear and continuous on } \mathcal{T}_h, \text{ and } w_h(0) = 0\}.$$



Thus, the basis functions for V_h^0 *are the hat-functions* $\varphi_j, j = 1, \ldots, n+1$ *(including the half-hat-function* φ_{n+1}*), and hence* $\dim(V_h^0) = n+1$.

Now the finite element formulation reads as follows: find $u_h \in V_h^0$ *such that*

$$(FEM) \qquad \varepsilon \int_0^1 u_h' v'\, dx + p \int_0^1 u_h' v\, dx = r \int_0^1 v\, dx + \varepsilon\beta v(1), \quad \forall v \in V_h^0.$$

Inserting the ansatz $u_h(x) = \sum_{j=1}^{n+1} \xi_j \varphi_j(x)$ *into (FEM), and choosing* $v = \varphi_i(x)$, $i = 1, \ldots, n+1$, *we get*

$$\sum_{j=1}^{n+1} \Big(\varepsilon \int_0^1 \varphi_j'(x)\varphi_i'(x)\, dx + p \int_0^1 \varphi_j'(x)\varphi_i(x)\, dx \Big) \xi_j = r \int_0^1 \varphi_i(x)\, dx + \varepsilon\beta\varphi_i(1),$$

In matrix form this corresponds to the linear system of equations $A\xi = \mathbf{b}$ *with* $A = \varepsilon\, \tilde{S} + p\, C$, *where* $\tilde{S}$ *is computed as* A_{unif} *and is the* $(n+1) \times (n+1)$*-stiffness matrix with its last diagonal element* $\tilde{s}_{n+1,n+1} = \int_0^1 \varphi_{n+1}' \varphi_{n+1}'\, dx = 1/h$, *and* C *is the convection matrix with the elements*

$$c_{ij} = \int_0^1 \varphi_j'(x)\varphi_i(x)\, dx.$$

Hence we have, evidently,

$$\tilde{S} = \frac{1}{h} \begin{bmatrix} 2 & -1 & 0 & 0 & \ldots & 0 \\ -1 & 2 & -1 & 0 & \ldots & 0 \\ 0 & -1 & 2 & -1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & -1 & 2 & -1 \\ 0 & \ldots & \ldots & \ldots & -1 & 1 \end{bmatrix}.$$

To compute the entries for C, we note that like, S, M and $\tilde{S}$, also C is a tridiagonal matrix. But C is anti-symmetric. Its entries are

$$\begin{cases} c_{ij} = 0, & \text{for} \quad |i-j| > 1 \\ c_{ii} = \int_0^1 \varphi_i(x)\varphi_i'(x)\,dx = 0, & \text{for} \quad i = 1, \ldots, n \\ c_{n+1,n+1} = \int_0^1 \varphi_{n+1}(x)\varphi_{n+1}'(x)\,dx = 1/2, & \\ c_{i,i+1} = \int_0^1 \varphi_i(x)\varphi_{i+1}'(x)\,dx = 1/2, & \text{for} \quad i = 1, \ldots, n \\ c_{i+1,i} = \int_0^1 \varphi_{i+1}(x)\varphi_i'(x)\,dx = -1/2, & \text{for} \quad i = 1, \ldots, n. \end{cases} \tag{4.4.5}$$

Finally, we have the entries b_i of the load vector $\mathbf{b}$ as

$$b_1 = \ldots = b_n = rh, \qquad b_{n+1} = rh/2 + \varepsilon\beta.$$

Thus,

$$C = \frac{1}{2}\begin{bmatrix} 0 & 1 & 0 & 0 & \ldots & 0 \\ -1 & 0 & 1 & 0 & \ldots & 0 \\ 0 & -1 & 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & -1 & 0 & 1 \\ 0 & \ldots & \ldots & \ldots & -1 & 1 \end{bmatrix}, \quad \mathbf{b} = rh \begin{bmatrix} 1 \\ 1 \\ 1 \\ . \\ 1 \\ 1/2 \end{bmatrix} + \varepsilon\beta \begin{bmatrix} 0 \\ 0 \\ 0 \\ . \\ 0 \\ 1 \end{bmatrix}.$$

Remark 4.4. *In the convection dominated case $\frac{\varepsilon}{p} << 1$ this standard FEM will not work. Spurious oscillations in the approximate solution will appear. The standard FEM has to be modified in this case.*

4.5 Exercises

Problem 4.1. *Consider the two-point boundary value problem*

$$-u'' = f, \quad 0 < x < 1; \qquad u(0) = u(1) = 0. \tag{4.5.1}$$

Let $V = \{v : \|v\| + \|v'\| < \infty, \quad v(0) = v(1) = 0\}$, $\|\cdot\|$ denotes the L_2-norm.

a. Use V to derive a variational formulation of (4.5.1).

b. Discuss why V is valid as a vector space of test functions.

c. Classify which of the following functions are admissible test functions

$$\sin \pi x, \qquad x^2, \qquad x \ln x, \qquad e^x - 1, \qquad x(1-x).$$

Problem 4.2. *Assume that $u(0) = u(1) = 0$, and that u satisfies*

$$\int_0^1 u'v' \, dx = \int_0^1 fv \, dx,$$

for all $v \in V = \{v : \|v\| + \|v'\| < \infty, \quad v(0) = v(1) = 0\}$.

a. Show that u minimizes the functional

$$F(v) = \frac{1}{2} \int_0^1 (v')^2 \, dx - \int_0^1 fv \, dx. \tag{4.5.2}$$

Hint: $F(v) = F(u + w) = F(u) + \ldots \geq F(u)$.

b. Prove that the above minimization problem is equivalent to

$$-u'' = f, \quad 0 < x < 1; \qquad u(0) = u(1) = 0.$$

Problem 4.3. *Consider the two-point boundary value problem*

$$-u'' = 1, \quad 0 < x < 1; \qquad u(0) = u(1) = 0. \tag{4.5.3}$$

Let $\mathcal{T}_h : x_j = \frac{j}{4}$, $j = 0, 1, \ldots, 4$, denote a partition of the interval $0 < x < 1$ into four subintervals of equal length $h = 1/4$ and let V_h be the corresponding space of continuous piecewise linear functions vanishing at $x = 0$ and $x = 1$.

a. Compute a finite element approximation $U \in V_h$ to (4.5.3).

b. Prove that $U \in V_h$ is unique.

Problem 4.4. *Consider once again the two-point boundary value problem*

$$-u'' = f, \quad 0 < x < 1; \qquad u(0) = u(1) = 0.$$

a. Prove that the finite element approximation $U \in V_h$ to u satisfies

$$\|(u - U)'\| \leq \|(u - v)'\|, \qquad \text{for all } v \in V_h.$$

b. Use this result and interpolation estimate to deduce that

$$\|(u - U)'\| \leq C\|hu''\|, \tag{4.5.4}$$

where C depends on the interpolation constant.

Problem 4.5. *Consider the two-point boundary value problem*

$$\begin{aligned} -(au')' &= f, & 0 < x < 1, \\ u(0) &= 0, & a(1)u'(1) = g_1, \end{aligned} \tag{4.5.5}$$

where $a > 0$ is a positive function and g_1 is a constant.

a. Derive the variational formulation of (4.5.5).

b. Discuss how the boundary conditions are implemented.

Problem 4.6. *Consider the two-point boundary value problem*

$$-u'' = 0, \quad x \in I := (0,1); \qquad u(0) = 0, \quad u'(1) = 7. \tag{4.5.6}$$

Divide I into two subintervals of length $h = \frac{1}{2}$ and let V_h be the corresponding space of continuous piecewise linear functions vanishing at $x = 0$.

a. Formulate a finite element method for (4.5.6).

b. Calculate by hand the finite element approximation $U \in V_h$ to (4.5.6).

c. Study how the boundary condition at $x = 1$ is approximated.

Problem 4.7. *Consider the two-point boundary value problem*

$$-u'' = 0, \quad 0 < x < 1; \qquad u'(0) = 5, \quad u(1) = 0. \tag{4.5.7}$$

Let $\mathcal{T}_h : x_j = jh$, $j = 0, 1, \ldots, N$, $h = 1/N$ be a uniform partition of the interval $0 < x < 1$ into N subintervals and let V_h be the corresponding space of continuous piecewise linear functions.

a. Use V_h, with $N = 3$, and formulate a finite element method for (4.5.7).

b. Compute the finite element approximation $U \in V_h$ assuming $N = 3$.

Problem 4.8. *Consider the problem of finding a solution approximation to*

$$-u'' = 1, \quad 0 < x < 1; \qquad u'(0) = u'(1) = 0. \tag{4.5.8}$$

Let $\mathcal{T}_h$ be a partition of the interval $0 < x < 1$ into two subintervals of equal length $h = \frac{1}{2}$ and let V_h be the corresponding space of continuous piecewise linear functions.

a. Find the exact solution to (4.5.8) *by integrating twice.*

b. Compute a finite element approximation $U \in V_h$ to u if possible.

Problem 4.9. *Consider the two-point boundary value problem*

$$-((1+x)u')' = 0, \quad 0 < x < 1; \qquad u(0) = 0, \quad u'(1) = 1. \tag{4.5.9}$$

Divide the interval $0 < x < 1$ *into 3 subintervals of equal length* $h = \frac{1}{3}$ *and let* V_h *be the corresponding space of continuous piecewise linear functions vanishing at* $x = 0$.

a. Use V_h *to formulate a finite element method for* (4.5.9).

b. Verify that the stiffness matrix $\mathbf{A}$ *and the load vector* $\mathbf{b}$ *are given by*

$$\mathbf{A} = \frac{1}{2}\begin{bmatrix} 16 & -9 & 0 \\ -9 & 20 & -11 \\ 0 & -11 & 11 \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

c. Show that $\mathbf{A}$ *is symmetric tridiagonal, and positive definite.*

d. Derive a simple way to compute the energy norm $\|U\|_E^2$, *defined by*

$$\|U\|_E^2 = \int_0^1 (1+x)U'(x)^2\,dx,$$

where $U \in V_h$ *is the finite element solution approximation.*

Problem 4.10. *Consider the two-point boundary value problem*

$$-u'' = 0, \quad 0 < x < 1; \qquad u(0) = 0, \quad u'(1) = k(u(1) - 1). \tag{4.5.10}$$

Let $\mathcal{T}_h : 0 = x_0 < x_1 < x_2 < x_3 = 1$, *where* $x_1 = \frac{1}{3}$ *and* $x_2 = \frac{2}{3}$ *be a partition of the interval* $0 \le x \le 1$ *and let* V_h *be the corresponding space of continuous piecewise linear functions, which vanish at* $x = 0$.

a. Compute a solution approximation $U \in V_h$ *to* (4.5.10) *assuming* $k = 1$.

b. Discuss how the parameter k *influence the boundary condition at* $x = 1$. *In particular when* $k \to \infty$ *and* $k \to 0$.

Problem 4.11. *Consider the finite element method applied to*

$$-u'' = 0, \quad 0 < x < 1; \qquad u(0) = \alpha, \quad u'(1) = \beta,$$

where α and β are given constants. Assume that the interval $0 \le x \le 1$ is divided into three subintervals of equal length $h = 1/3$ and that $\{\varphi_j\}_0^3$ is a nodal basis of V_h, the corresponding space of continuous piecewise linear functions.

a. Verify that the ansatz

$$U(x) = \alpha\varphi_0(x) + \xi_1\varphi_1(x) + \xi_2\varphi_2(x) + \xi_3\varphi_3(x),$$

yields the following system of equations

$$\frac{1}{h}\begin{bmatrix} -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} \alpha \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \beta \end{bmatrix}. \tag{4.5.11}$$

b. If $\alpha = 2$ and $\beta = 3$ show that (4.5.11) can be reduced to

$$\frac{1}{h}\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} = \begin{bmatrix} -2h^{-1} \\ 0 \\ 3 \end{bmatrix}.$$

c. Solve the above system of equations to find $U(x)$.

Problem 4.12. *Compute a finite element solution approximation to*

$$-u'' + u = 1; \qquad 0 \le x \le 1, \qquad u(0) = u(1) = 0, \tag{4.5.12}$$

using the continuous piecewise linear ansatz $U = \xi_1\varphi_1(x) + \xi_2\varphi_2(x)$ where

$$\varphi_1(x) = \begin{cases} 3x, & 0 < x < \frac{1}{3} \\ 2 - 3x, & \frac{1}{3} < x < \frac{2}{3}, \\ 0, & \frac{2}{3} < x < 1 \end{cases} \qquad \varphi_2(x) = \begin{cases} 0, & 0 < x < \frac{1}{3} \\ 3x - 1, & \frac{1}{3} < x < \frac{2}{3}. \\ 3 - 3x, & \frac{2}{3} < x < 1 \end{cases}$$

Problem 4.13. *Consider the following eigenvalue problem*

$$-au'' + bu = 0; \qquad 0 \le x \le 1, \qquad u(0) = u'(1) = 0, \tag{4.5.13}$$

where a, $b > 0$ are constants. Let $\mathcal{T}_h : 0 = x_0 < x_1 < \ldots < x_N = 1$, be a non-uniform partition of the interval $0 \le x \le 1$ into N intervals of length $h_i = x_i - x_{i-1}$, $i = 1, 2, \ldots, N$ and let V_h be the corresponding space of continuous piecewise linear functions. Compute the stiffness and mass matrices.

Problem 4.14. *Show that the FEM with the mesh size h for the problem:*

$$\begin{cases} -u'' = 1 & 0 < x < 1 \\ u(0) = 1 & u'(1) = 0, \end{cases} \tag{4.5.14}$$

with

$$U(x) = 7\varphi_0(x) + U_1\varphi_1(x) + \ldots + U_m\varphi_m(x). \tag{4.5.15}$$

leads to the linear system of equations: $\tilde{A} \cdot \tilde{U} = \tilde{b}$, where

$$\underset{m \times (m+1)}{\tilde{A} = \frac{1}{h}\begin{bmatrix} -1 & 2 & -1 & & 0 \\ 0 & -1 & 2 & -1 & \ldots \\ \ldots & \ldots & \ldots & & \ldots \\ 0 & \ldots & 0 & & \ldots \end{bmatrix}}, \quad \underset{(m+1) \times 1}{\tilde{U} = \begin{bmatrix} 7 \\ U_1 \\ \ldots \\ U_m \end{bmatrix}}, \quad \underset{m \times 1}{\tilde{b} = \begin{bmatrix} h \\ \ldots \\ h \\ h/2 \end{bmatrix}},$$

. which is reduced to $AU = b$, with

$$A = \frac{1}{h}\begin{bmatrix} 2 & -1 & 0 & \ldots & 0 \\ -1 & 2 & -1 & 0 & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{bmatrix}, \quad U = \begin{bmatrix} U_1 \\ U_2 \\ \ldots \\ U_m \end{bmatrix}, \quad b = \begin{bmatrix} h + \frac{7}{h} \\ h \\ \ldots \\ h \\ h/2 \end{bmatrix}.$$

Problem 4.15. *Prove an a priori error estimate for the $cG(1)$ finite element method for the problem*

$$-u'' + \alpha u = f, \quad \text{in } I = (0,1), \qquad u(0) = u(1) = 0,$$

where the coefficient $\alpha = \alpha(x)$ is a bounded positive function on I, ($0 \leq \alpha(x) \leq K,\ x \in I$).

Problem 4.16. *a) Formulate a $cG(1)$ method for the problem*

$$\begin{cases} (a(x)u'(x))' = 0, & 0 < x < 1, \\ a(0)u'(0) = u_0, & u(1) = 0. \end{cases}$$

and give an a priori error estimate.

b) Let $u_0 = 3$ and compute the approximate solution in a) for a uniform partition of $I = [0,1]$ into 4 intervals and

$$a(x) = \begin{cases} 1/4, & x < 1/2, \\ 1/2, & x > 1/2. \end{cases}$$

c) Show that, with these special choices, the computed solution is equal to the exact one, i.e. the error is equal to 0.

Problem 4.17. *Prove an a priori error estimate for the finite element method for the problem*

$$-u''(x) + u'(x) = f(x), \quad 0 < x < 1, \qquad u(0) = u(1) = 0.$$

Problem 4.18. *(a) Prove an a priori error estimate for the $cG(1)$ approximation of the boundary value problem*

$$-u'' + cu' + u = f \quad \text{in } I = (0,1), \qquad u(0) = u(1) = 0,$$

where $c \geq 0$ is constant.

(b) For which value of c is the a priori error estimate optimal?

Problem 4.19. *Let U be the piecewise linear finite element approximation for*

$$-u''(x) + 2xu'(x) + 2u(x) = f(x), \quad x \in (0,1), \qquad u(0) = u(1) = 0,$$

in a partition $\mathcal{T}_h$ of the interval $[0,1]$. Set $e = u - U$ and derive a priori error estimates in the energy-norm:

$$||e||_E^2 = ||e'||^2 + ||e||^2, \quad \textit{where} \quad ||w||^2 = \int_0^1 w(x)^2\,dx.$$

Chapter 5

Scalar Initial Value Problems

This chapter is devoted to numerical methods for time discretizations. Here, we shall consider problems depending on the time variable, only. The approximation techniques developed in this chapter, combined with those of the previous chapter for boundary value problems, can be used for the numerical study of initial boundary value problems; such as, e.g. the heat and wave equations.

As a model problem we shall consider the classical example of population dynamics described by the following ordinary differential equation (ODE)

$$\begin{array}{ll} (\mathrm{DE}) & \left\{ \begin{array}{rc} \dot{u}(t) + a(t)u(t) = f(t), & 0 < t \leq T, \\ u(0) = u_0, & \end{array} \right. \\ (\mathrm{IV}) & \end{array} \tag{5.0.1}$$

where $f(t)$ is the source term and $\dot{u}(t) = \dfrac{du}{dt}$. The coefficient $a(t)$ is a bounded function. If $a(t) \geq 0$ the problem $(5.0.1)$ is called $parabolic$, while $a(t) \geq \alpha > 0$ yields a $dissipative\ problem$, in the sense that, with increasing t, perturbations of solutions to $(5.0.1)$, e.g. introduced by numerical discretization, will decay. In general, in numerical approximations for $(5.0.1)$, the error accumulates when advancing in time, i.e. the error of previous time steps adds up to the error of the present time step. The different types of error accumulation/perturbation growth are referred to as stability properties of the initial value problem.

5.1 Solution formula and stability

Theorem 5.1. *The solution of the problem* (5.0.1) *is given by*

$$u(t) = u_0 \cdot e^{-A(t)} + \int_0^t e^{-(A(t)-A(s))} f(s) ds, \tag{5.1.1}$$

where $A(t) = \int_0^t a(s)ds$ *and* $e^{A(t)}$ *is the integrating factor.*

Proof. Multiplying the (DE) by the integrating factor $e^{A(t)}$ we have

$$\dot{u}(t)e^{A(t)} + \dot{A}(t)e^{A(t)}u(t) = e^{A(t)}f(t), \tag{5.1.2}$$

where we used that $a(t) = \dot{A}(t)$. Equation (5.1.2) can be rewritten as

$$\frac{d}{dt}\Big(u(t)e^{A(t)}\Big) = e^{A(t)}f(t).$$

We denote the variable by s and integrate from 0 to t to get

$$\int_0^t \frac{d}{ds}\Big(u(s)e^{A(s)}\Big)ds = \int_0^t e^{A(s)}f(s)ds,$$

i.e.

$$u(t)e^{A(t)} - u(0)e^{A(0)} = \int_0^t e^{A(s)}f(s)ds.$$

Now since $A(0) = 0$ and $u(0) = u_0$ we get the desired result

$$u(t) = u_0 e^{-A(t)} + \int_0^t e^{-(A(t)-A(s))} f(s) ds. \tag{5.1.3}$$

□

This representation of u is known as the *Variation of constants formula*.

Theorem 5.2 (Stability estimates)**.** *Using the solution formula, we can derive the following stability estimates:*

(i) If $a(t) \geq \alpha > 0$*, then* $|u(t)| \leq e^{-\alpha t}|u_0| + \dfrac{1}{\alpha}(1 - e^{-\alpha t}) \max\limits_{0\leq s\leq t} |f(s)|$,

(ii) If $a(t) \geq 0$ *(i.e.* $\alpha = 0$*; the parabolic case), then*

$$|u(t)| \leq |u_0| + \int_0^t |f(s)|ds \quad \text{or} \quad |u(t)| \leq |u_0| + \|f\|_{L_1(0,t)}. \tag{5.1.4}$$

Proof. (i) For $a(t) \geq \alpha > 0$ we have that $A(t) = \int_0^t a(s)ds$ is an increasing function of t, $A(t) \geq \alpha t$ and

$$A(t) - A(s) = \int_0^t a(r)\,dr - \int_0^s a(r)\,dr = \int_s^t a(r)\,dr \geq \alpha(t-s). \quad (5.1.5)$$

Thus $e^{-A(t)} \leq e^{-\alpha t}$ and $e^{-(A(t)-A(s))} \leq e^{-\alpha(t-s)}$. Hence, using (5.1.3) we get

$$|u(t)| \leq |u_0|e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)}|f(s)|ds, \quad (5.1.6)$$

which yields

$$|u(t)| \leq e^{-\alpha t}|u_0| + \max_{0\leq s\leq t}|f(s)|\Big[\frac{1}{\alpha}e^{-\alpha(t-s)}\Big]_{s=0}^{s=t}, \quad \text{i.e.}$$
$$|u(t)| \leq e^{-\alpha t}|u_0| + \frac{1}{\alpha}(1-e^{-\alpha t})\max_{0\leq s\leq t}|f(s)|.$$

(ii) Let $\alpha = 0$ in (5.1.6) (which is true also in this case: for $\alpha = 0$), then $|u(t)| \leq |u_0| + \int_0^t |f(s)|ds$, and the proof is complete. □

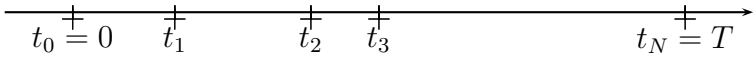
Remark 5.1. *(i) expresses that the effect of the initial data u_0 decays exponentially with time, and that the effect of the source term f on the right hand side does not depend on the length of the time interval, only on the maximum value of f, and on the value of α. In case (ii), the influence of u_0 remains bounded in time, and the integral of f indicates an accumulation in time.*

5.2 Finite difference methods

Let us first continue as in Example 2.1 and give the other two, very common, finite difference approaches, for numerical solution of (5.0.1): Let

$$\mathcal{T}_k := \{0 = x_0 < x_1 <, \ldots < x_{N-1} < x_N = T\},$$

be a partition of the time interval $[0, T]$ into the subintervals $I_k := [x_{k-1}, x_k],\ k = 1, \ldots N$ as in the Example 2.1:



Example 5.1. *Now we discretize the IVP* (5.0.1)*, for* $a(t)$ *being positive constant, with the backward Euler method, on the partition* $\mathcal{T}_k$*, approximating the derivative* $\dot{u}(t)$ *by a backward-difference quotient at each subinterval* $I_k = (t_{k-1}, t_k]$ *by* $\dot{u}(t) \approx \frac{u(t_k)-u(t_{k-1})}{t_k - t_{k-1}}$*. Then an approximation of* (5.0.1)*, with* $f(t) \equiv 0$ *is given by*

$$\frac{u(t_k) - u(t_{k-1})}{t_k - t_{k-1}} = -a \cdot u(t_k), \qquad k = 1, \ldots, N, \quad \text{and} \quad u(0) = u_0. \tag{5.2.1}$$

(Note that in forward Euler, we would have $-au(t_{k-1})$ *on the right hand side of* (5.2.1)*). Letting* $\Delta t_k = t_k - t_{k-1}$*,* (5.2.1) *yields*

$$(1 + a\Delta t_k)u(t_k) = u(t_{k-1}). \tag{5.2.2}$$

Starting with $k = 1$ *and the data* $u(0) = u_0$*, the solution* $u(t_k)$ *would, iteratively, be computed at the subsequent points:* $t_1,\ t_2, \ldots, t_N = T$*.*
For a uniform partition, where all subintervals have the same length Δt*, and since for* $a > 0$*,* $1 + a\Delta t > 0\,(\neq 0)$*,* (5.2.2) *can be written as*

$$u(t_k) = (1 + a\Delta t)^{-1} u(t_{k-1}), \quad k = 1, 2, \ldots, N. \tag{5.2.3}$$

Iterating we get the Backward or Implicit Euler method for (5.0.1)*:*

$$u(t_k) = (1 + a\Delta t)^{-1} u(t_{k-1}) = (1 + a\Delta t)^{-2} u(t_{k-2}) = \ldots = (1 + a\Delta t)^{-k} u_0.$$

Remark 5.2. *Note that, for the problem* (5.0.1)*, with* $f(t) \equiv 0$*, in the Example 2.1 we just replace* λ *with* $-a$*, and get the forward (explicit) Euler method:*

$$u(t_k) = (1 - a\Delta t)^k u_0. \tag{5.2.4}$$

Example 5.2. *Now we introduce the Crank-Nicolson method for the finite difference approximation of* (5.0.1)*. Here, first we integrate the equation* (5.0.1) *over* $I_k = [t_{k-1}, t_k]$ *to get*

$$u(t_k) - u(t_{k-1}) + a \int_{t_{k-1}}^{t_k} u(t)\, dt = \int_{t_{k-1}}^{t_k} f(t)\, dt. \tag{5.2.5}$$

Then, approximate the integral term by the simple trapezoidal rule we get

$$u(t_k) - u(t_{k-1}) + a\frac{\Delta t_k}{2}(u(t_k) + u(t_{k-1})) = \int_{t_{k-1}}^{t_k} f(t)\,dt. \tag{5.2.6}$$

Rearranging the terms yields

$$\Big(1 + a\frac{\Delta t_k}{2}\Big)u(t_k) = \Big(1 - a\frac{\Delta t_k}{2}\Big)u(t_{k-1}) + \int_{t_{k-1}}^{t_k} f(t)\,dt.$$

Or, equivalently,

$$u(t_k) = \frac{1 - a\Delta t_k/2}{1 + a\Delta t_k/2}u(t_{k-1}) + \frac{1}{1 + a\Delta t_k/2}\int_{t_{k-1}}^{t_k} f(t)\,dt.$$

Let us assume a zero source term ($f = 0$), and uniform partition, i.e. $\Delta t_k = \Delta t$ for $k = 1, 2, \ldots, N$, then we have the following Crank-Nicolson method:

$$u(t_k) = \Big(\frac{1 - a\Delta t/2}{1 + a\Delta t/2}\Big)^k u_0. \tag{5.2.7}$$

Example 5.3. *Consider the initial value problem:*

$$\dot{u}(t) + au(t) = 0, \quad t > 0, \quad u(0) = 1.$$

a) Let $a = 40$, and the time step $k = 0.1$. Draw the graph of $U_n := U(nk)$, $k = 1, 2, \ldots,$ approximating u using (i) explicit (forward) Euler, (ii) implicit (Backward) Euler, and (iii) Crank-Nicolson methods.

b) Consider the case $a = i$, $(i^2 = -1)$, having the complex solution $u(t) = e^{-it}$ with $|u(t)| = 1$ for all t. Show that this property is preserved in Crank-Nicolson approximation, (i.e. $|U_n| = 1$), but NOT in any of the Euler approximations.

Solution: *a) With $a = 40$ and $k = 0.1$ we get the explicit Euler:*

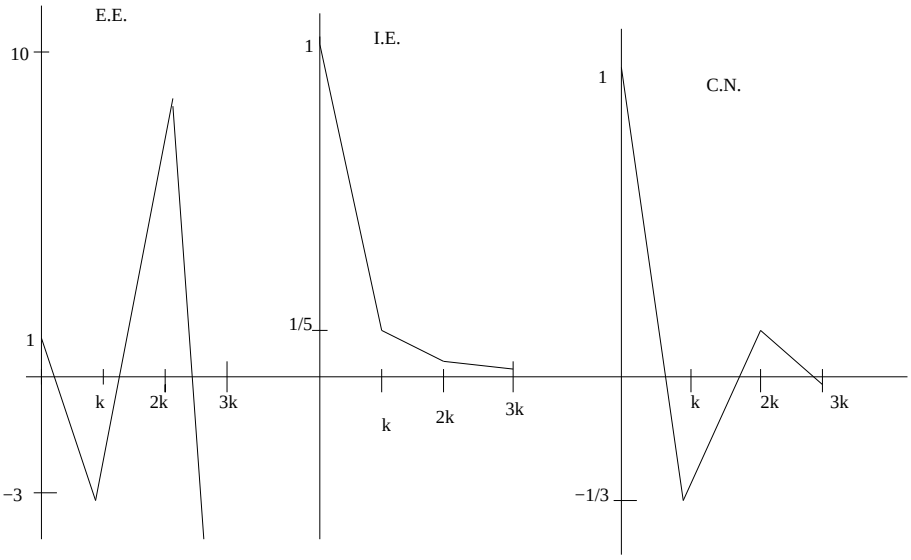
$$\begin{cases} U_n - U_{n-1} + 40 \times (0.1)U_{n-1} = 0, \\ U_0 = 1. \end{cases} \Longrightarrow \begin{cases} U_n = -3U_{n-1}, \quad n = 1, 2, \ldots, \\ U_0 = 1. \end{cases}$$

Implicit Euler:

$$\begin{cases} U_n = \frac{1}{1+40\times(0.1)}U_{n-1} = \frac{1}{5}U_{n-1}, \quad n = 1, 2, 3, \ldots, \\ U_0 = 1. \end{cases}$$

Cranck-Nicolson:

$$\begin{cases} U_n = \frac{1-\frac{1}{2}\times 40\times(0.1)}{1+\frac{1}{2}\times 40\times(0.1)} U_{n-1} = -\frac{1}{3}U_{n-1}, \quad n = 1,\, 2,\, 3, \ldots, \\ U_0 = 1. \end{cases}$$



b) With $a = i$ we get
Explicit Euler

$$|U_n| = |1 - (0.1) \times i||U_{n-1}| = \sqrt{1 + 0.01}|U_{n-1}| \Longrightarrow |U_n| \geq |U_{n-1}|.$$

Implicit Euler

$$|U_n| = \Big|\frac{1}{1 + (0.1) \times i}\Big||U_{n-1}| = \frac{1}{\sqrt{1 + 0.01}}|U_{n-1}| \leq |U_{n-1}|.$$

Crank-Nicolson

$$|U_n| = \Big|\frac{1 - \frac{1}{2}(0.1) \times i}{1 + \frac{1}{2}(0.1) \times i}\Big||U_{n-1}| = |U_{n-1}|.$$

5.3 Galerkin finite element methods for IVP

The polynomial approximation procedure introduced in Chapter 2, along with (2.2.5)-(2.3.5), for the initial value problem (2.1.1), or (5.0.1), being

over the whole time interval $(0, T)$ is referred as the *global Galerkin method*. In this section, first we introduce two versions of the global Galerkin method and then extend them to partitions of the interval $(0, T)$ using piecewise polynomial test (multiplier) and trial (solution) functions. Here, we shall focus on two simple, low degree polynomial, approximation cases.

• *The continuous Galerkin method of degree 1; cG(1).* In this case the trial functions are piecewise linear and continuous while the test functions are piecewise constant and discontinuous, i.e. *unlike the cG(1) for BVP*, here the trial and test functions belong to different polynomial spaces.

• *The discontinuous Galerkin method of degree 0; dG(0).* Here both the trial and test functions are chosen to be piecewise constant and discontinuous.

5.3.1 The continuous Galerkin method

Recall the global Galerkin method of degree q; (2.3.1), for the initial value problem (5.0.1): find $U \in \mathcal{P}^q(0, T)$, with $U(0) = u_0$ such that

$$\int_0^T (\dot{U} + aU)v dt = \int_0^T f v \, dt, \quad \forall v \in \mathcal{P}^q(0, T), \text{ with } v(0) = 0. \tag{5.3.1}$$

We formulate the following alternative formulation: Find $U \in \mathcal{P}^q(0, T)$ with $U(0) = u_0$ such that

$$\int_0^T (\dot{U} + aU)v dt = \int_0^T f v dt, \quad \forall v \in \mathcal{P}^{q-1}(0, T), \tag{5.3.2}$$

Note that in (5.3.1) we have that $v \in span\{t, t^2, \ldots, t^q\}$, whereas in (5.3.2) the test functions $v \in span\{1, t, t^2, \ldots, t^{q-1}\}$. Hence, the difference between these two formulations lies in the choice of their test function spaces. We shall focus on (5.3.2), due to the fact that, actually, this method yields a more accurate approximation of degree q than the original method (5.3.1). The following example illustrates the phenomenon

Example 5.4. *Consider the IVP*

$$\begin{cases} \dot{u}(t) + u(t) = 0, & 0 \le t \le 1, \\ u(0) = 1. \end{cases} \tag{5.3.3}$$

The exact solution is given by $u(t) = e^{-t}$. *The continuous piecewise linear approximation, with the ansatz* $U(t) = 1 + \xi_1 t$ *in,*

$$\int_0^1 (\dot{U}(t) + U(t))v(t)\,dt = 0, \tag{5.3.4}$$

and $v(t) = t$ *(i.e.* (5.3.1)*) yields*

$$\int_0^1 (\xi_1 + 1 + \xi_1 t)t\,dt = 0 \Longrightarrow \left[(\xi_1 + 1)\frac{t^2}{2} + \xi_1\frac{t^3}{3}\right]_0^1 = 0 \Longrightarrow \xi_= -3/5.$$

Hence in this case the approximate solution, that we denote by U_1 *is given by* $U_1(t) = 1 - (3/5)t$. *Whereas* (5.3.2) *for this problem means* $v(t) = 1$ *and gives*

$$\int_0^1 (\xi_1 + 1 + \xi_1 t)\,dt = 0 \Longrightarrow \left[(\xi_1 + 1)t + \xi_1\frac{t^2}{2}\right]_0^1 = 0 \Longrightarrow \xi_= -2/3.$$

In this case the approximate solution that we denote by U_2 *is given as* $U_2(t) = 1 - (2/3)t$. *As we can see in the figure below* U_2 *is a better approximation for* e^{-t} *than* U_1.

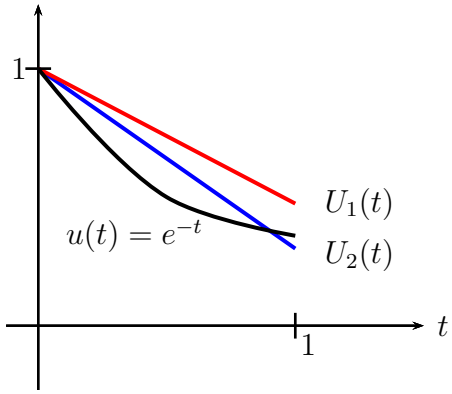


Figure 5.1: Two continuous linear Galerkin approximations of e^{-t}.

Before generalizing (5.3.2) to piecewise polynomial approximation, which is the $cG(q)$ method, we consider a canonical example of(5.3.2).

Example 5.5. *Consider* (5.3.2) *with* $q = 1$, *then choosing* $v \equiv 1$, *yields*

$$\int_0^T (\dot{U} + aU)v dt = \int_0^T (\dot{U} + aU) dt = U(T) - U(0) + \int_0^T aU(t) dt \quad (5.3.5)$$

Here $U(t)$, *as a linear function, is given by*

$$U(t) = U(0)\frac{T-t}{T} + U(T)\frac{t}{T}. \quad (5.3.6)$$

Inserting (5.3.6) *into* (5.3.5) *we get*

$$U(T) - U(0) + \int_0^T a\Big(U(0)\frac{T-t}{T} + U(T)\frac{t}{T}\Big)\, dt = \int_0^T f\, dt, \quad (5.3.7)$$

which is an equation for the unknown quantity $U(T)$. *Thus, using* (5.3.6) *with a given* $U(0)$, *we get the linear approximation* $U(t)$ *for all* $t \in [0, T]$. *Below we generalize this example to piecewise linear approximation and demonstrate the iteration procedure for the cG(1) scheme.*

The cG(1) Algorithm
For a partition $\mathcal{T}_k$ *of the interval* $[0, T]$ *into subintervals* $I_k = (t_{k-1}, t_k]$, *we perform the following steps:*

(1) Given $U(0) = U_0 = u_0$ *and a source term* f, *apply (5.3.7) to the first subinterval* $(0, t_1]$ *and compute* $U(t_1)$. *Then, using* (5.3.6) *one gets* $U(t), \forall t \in [0, t_1]$.

(2) Assume that we have computed $U_{k-1} := U(t_{k-1})$. *Hence,* U_{k-1} *and* f *are now considered as data. Now we consider the problem on the subintervals* $I_k = (t_{k-1}, t_k]$, *and compute the unknown* $U_k := U(t_k)$ *from the local version of* (5.3.7),

$$U_k - U_{k-1} + \int_{t_{k-1}}^{t_k} a\Big(\frac{t_k - t}{t_k - t_{k-1}}U_{k-1} + \frac{t - t_{k-1}}{t_k - t_{k-1}}U_k\Big) dt = \int_{t_{k-1}}^{t_k} f dt.$$

Having both U_{k-1} *and* U_k, *by linearity, we get* $U(t)$ *for* $t \in I_k$. *To get the continuous piecewise linear approximation in the whole interval* $[0, t_N]$, *step (2) is performed in successive subintervals* I_k, $k = 2, \ldots, N$.

The cG(q) method
The Global continuous Galerkin method of degree q, formulated on a partition $\mathcal{T}_k,\ 0 = t_0 < t_1 < \ldots < t_N = T$ of the interval $(0, T)$, is referred to as the $cG(q)$ method and reads as: find $U(t) \in V_k^{(q)}$, such that $U(0) = u_0$, and

$$\int_0^{t_N} (\dot{U} + aU)w dt = \int_0^{t_N} f w dt, \qquad \forall w \in W_k^{(q-1)}, \tag{5.3.8}$$

where

$$V_k^{(q)} = \{v : v \text{ continuous, piecewise polynomial of degree } \leq q \text{ on } \mathcal{T}_k\},$$
$$W_k^{(q-1)} = \{w : w \text{ discontinuous, piecewise polynomial, deg } w \leq q-1 \text{ on } \mathcal{T}_k\}.$$

So, the difference between the global continuous Galerkin method and cG(q) is that now we have *piecewise polynomials* on a partition of $[0, T]$ rather than global polynomials in the whole interval $[0, T]$.

5.3.2 The discontinuous Galerkin method

We start presenting the *global discontinuous Galerkin method of degree q*: find $U(t) \in \mathcal{P}^q(0, T)$ such that

$$\int_0^T (\dot{U} + aU)v dt + (U(0) - u(0))v(0) = \int_0^T f v dt, \quad \forall v \in \mathcal{P}^q(0, T). \tag{5.3.9}$$

This approach gives up the requirement that $U(t)$ satisfies the initial condition. Instead, the initial condition is imposed in a variational sense by the term $(U(0) - u(0))v(0)$. As in the cG(q) case, to derive the *discontinuous Galerkin method of degree q*: $dG(q)$ scheme, the above strategy can be formulated for the subintervals in a partition $\mathcal{T}_k$. To this end, we recall the notation for the right/left limits: $v_n^{\pm} = \lim_{s \to 0^+} v(t_n \pm s)$ and the corresponding *jump* term $[v_n] = v_n^+ - v_n^-$ at time level $t = t_n$. Then, the $dG(q)$ method for (5.0.1) reads as follows: for $n = 1, \ldots, N$; find $U(t) \in \mathcal{P}^q(t_{n-1}, t_n)$ such that

$$\int_{t_{n-1}}^{t_n} (\dot{U} + aU)v dt + U_{n-1}^+ v_{n-1}^+ = \int_{t_{n-1}}^{t_n} f v dt + U_{n-1}^- v_{n-1}^+, \ \ \forall v \in \mathcal{P}^q(t_{n-1}, t_n). \tag{5.3.10}$$

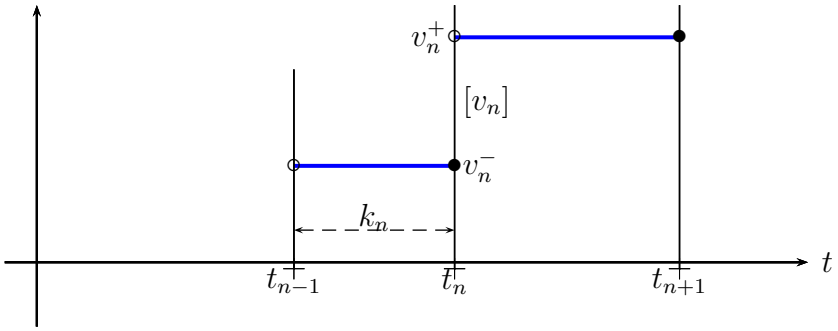


Figure 5.2: The jump $[v_n]$ and the right and left limits $v_n^{\pm}$

Example 5.6 (dG(0))**.** *Let $q = 0$, then v is constant generated by the single basis function: $v \equiv 1$. Further, we have $U(t) = U_n = U_{n-1}^+ = U_n^-$ on $I_n = (t_{n-1}, t_n]$, and $\dot{U} \equiv 0$. Thus, for $q = 0$ (5.3.10) yields the following dG(0) formulation: for $n = 1, \ldots, N$; find piecewise constants U_n such that*

$$\int_{t_{n-1}}^{t_n} aU_n dt + U_n = \int_{t_{n-1}}^{t_n} f dt + U_{n-1}. \tag{5.3.11}$$

Summing over n in (5.3.10)*, we get the following general dG(q) formulation: Find $U(t) \in W_k^{(q)}$, with $U_0^- = u_0$ such that*

$$\sum_{n=1}^{N} \int_{t_{n-1}}^{t_n} (\dot{U} + aU) w dt + \sum_{n=1}^{N} [U_{n-1}] w_{n-1}^+ = \int_0^{t_N} f w dt, \quad \forall w \in W_k^{(q)}. \tag{5.3.12}$$

Remark 5.3. *One can show that $cG(1)$ converges faster than $dG(0)$, whereas $dG(0)$ has better stability properties than $cG(1)$: More specifically, in the parabolic case when $a > 0$ is constant and ($f \equiv 0$) we can easily verify that (see Exercise 6.10 at the end of this chapter) the $dG(0)$ solution U_n corresponds to the* Backward Euler *scheme*

$$U_n = \Big(\frac{1}{1 + ak}\Big)^n u_0,$$

and the $cG(1)$ solution $\widetilde{U}_n$ is given by the Crank-Nicolson scheme:

$$\widetilde{U}_n = \Big(\frac{1 - \frac{1}{2}ak}{1 + \frac{1}{2}ak}\Big)^n u_0,$$

where k is the constant time step.

5.4 Exercises

Problem 5.1. *(a) Derive the stiffness matrix and load vector in piecewise polynomial (of degree q) approximation for the ODE of population dynamics:*

$$\dot{u}(t) = \lambda u(t), \quad \text{for } 0 < t \leq 1, \qquad u(0) = u_0.$$

(b) Let $\lambda = 1$ and $u_0 = 1$ and determine the approximate solution $U(t)$, for $q = 1$ and $q = 2$.

Problem 5.2. *Consider the initial value problem*

$$\dot{u}(t) + a(t)u(t) = f(t), \quad 0 < t \leq T, \qquad u(0) = u_0.$$

Show that if $a(t) = 1$, $f(t) = 2\sin(t)$, then we have
$u(t) = \sin(t) - \cos(t) = \sqrt{2}\sin(t - \pi/2)$.

Problem 5.3. *Compute the solution for*

$$\dot{u}(t) + a(t)u(t) = t^2, \quad 0 < t \leq T, \qquad u(0) = 1,$$

corresponding to
(a) $a(t) = 4$, *(b)* $a(t) = -t$.

Problem 5.4. *Compute the cG(1) approximation for the differential equations in the above problem. In each case, determine the condition on the step size that guarantees that U exists.*

Problem 5.5. *Without using the solution Theorem 5.1, prove that if $a(t) \geq 0$ then, a continuously differentiable solution of* (5.0.1) *is unique.*

Problem 5.6. *Consider the initial value problem*

$$\dot{u}(t) + a(t)u(t) = f(t), \quad 0 < t \leq T, \qquad u(0) = u_0.$$

Show that for $a(t) > 0$, and for $N = 1, 2, \ldots$, the piecewise linear approximate solution U for this problem satisfies the error estimate

$$|u(t_N) - U_N| \leq \max_{[0,t_N]} |k(\dot{U} + aU - f)|, \quad k = k_n, \text{ for } t_{n-1} < t \leq t_n.$$

Problem 5.7. *Consider the initial value problem*

$$\dot{u}(t) + au(t) = 0, \quad t > 0, \quad u(0) = u_0, \quad (a = \text{ constant}).$$

Assume a constant time step k and verify the iterative formulas for $dG(0)$ and $cG(1)$ approximations U and $\tilde{U}$, respectively: i.e.

$$U_n = \Big(\frac{1}{1+ak}\Big)^n u_0, \qquad \tilde{U}_n = \Big(\frac{1-ak/2}{1+ak/2}\Big)^n u_0.$$

Problem 5.8. *Assume that*

$$\int_{I_j} f(s)\,ds = 0, \qquad \text{for } \ j = 1, 2, \ldots,$$

where $I_j = (t_{j-1}, t_j)$, $t_j = jk$ with k being a positive constant. Prove that if $a(t) \geq 0$, then the solution for (5.0.1) *satisfies*

$$|u(t)| \leq e^{-A(t)}|u_0| + \max_{0\leq s\leq t} |kf(s)|.$$

Problem 5.9. *Formulate a continuous Galerkin method using piecewise polynomials based on the original global Galerkin method.*

Problem 5.10. *Formulate the dG(1) method for the differential equations specified in Problem 5.3.*

Problem 5.11. *Write out the a priori error estimates for the equations specified in Problem 5.3.*

Problem 5.12. *Use the a priori error bound to show that the residual of the dG(0) approximation satisfies $\mathcal{R}(U) = \mathcal{O}(1)$.*

Problem 5.13. *Prove the following stability estimate for the dG(0) method described by* (5.3.12),

$$|U_N|^2 + \sum_{n=0}^{N-1} |[U_n]|^2 \leq |u_0|^2.$$

Chapter 6

Initial Boundary Value Problems in 1d

A large class of phenomena in nature, science and technology, such as seasonal periods, heat distribution, wave propagation, etc, are varying both in space and time. To describe these phenomena in a physical domain requires the knowledge of their initial status, as well as information on the boundary of the domain, or asymptotic behavior in the case of unbounded domains. Problems that model such properties are called initial boundary value problems. In this chapter we shall study the two most important equations of this type: namely, the heat equation and the wave equation in one space dimension. We also address (briefly) the one-space dimensional time-dependent convection-diffusion problem.

6.1 Heat equation in 1d

In this section we focus on some basic L_2-stability and finite element error estimates for the, time-dependent, one-space dimensional heat equation. Here, to illustrate, we consider an example of an initial boundary value problem (IBVP) for the one-dimensional heat flux, viz

$$\begin{cases} \dot{u} - u'' = f(x,t), & 0 < x < 1, \quad t > 0, \\ u(x,0) = u_0(x), & 0 < x < 1, \\ u(0,t) = u_x(1,t) = 0, & t > 0. \end{cases} \tag{6.1.1}$$

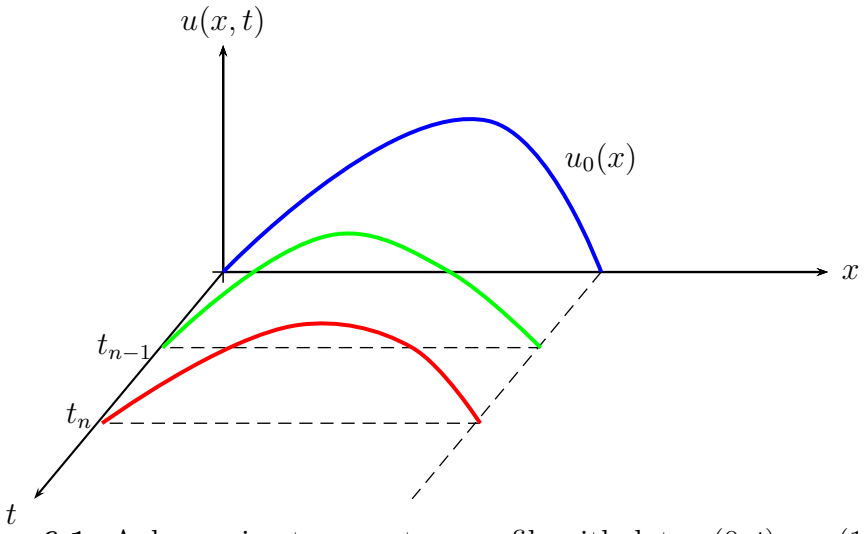


Figure 6.1: A decreasing temperature profile with data $u(0,t) = u(1,t) = 0$.

Example 6.1. *Describe the physical meaning of the functions and parameters in the problem* (6.1.1)*, when* $f = 20 - u$*.*

Answer: The problem is an example of heat conduction where

$u(x,t)$,	means the temperature at the point x and time t.
$u(x,0) = u_0(x)$,	is the initial temperature at time $t = 0$.
$u(0,t) = 0$,	means fixed temperature at the boundary point $x = 0$.
$u'(1,t) = 0$,	means isolated boundary at the boundary point $x = 1$ (where no heat flux occurs).
$f = 20 - u$,	is the heat source, in this case a control system to force $u \to 20$.

Remark 6.1. *Observe that it is possible to generalize* (6.1.1) *to a* u *dependent source term* f*, e.g. as in the above example where* $f = 20 - u$*.*

6.1.1 Stability estimates

We shall derive a general stability estimate for the mixed (Dirichlet at one end point and Neumann in the other) initial boundary value problem above,

prove a one-dimensional version of the *Poincare inequality* and finally derive some stability estimates in the homogeneous ($f \equiv 0$) case.

Theorem 6.1. *The IBVP* (6.1.1) *satisfies the stability estimates*

$$||u(\cdot,t)|| \leq ||u_0|| + \int_0^t ||f(\cdot,s)||\, ds, \tag{6.1.2}$$

$$||u'(\cdot,t)||^2 \leq ||u_0'||^2 + \int_0^t ||f(\cdot,s)||^2\, ds, \tag{6.1.3}$$

where u_0 and u_0' are assumed to be $L_2(I)$ functions with $I = (0,1)$. Note further that, here $||\bullet(\cdot,t)||$ is the time dependent L_2 norm:

$$||w(\cdot,s)|| := ||w(\cdot,s)||_{L_2(0,1)} = \Big(\int_0^1 ||w(x,s)||^2\, dx\Big)^{1/2}.$$

Proof. Multiply the equation in (6.1.1) by u and integrate over $(0,1)$ to get

$$\int_0^1 \dot{u}u\, dx - \int_0^1 u''u\, dx = \int_0^1 fu\, dx. \tag{6.1.4}$$

Note that $\dot{u}u = \frac{1}{2}\frac{d}{dt}u^2$. Hence, integration by parts in the second integral yields

$$\frac{1}{2}\frac{d}{dt}\int_0^1 u^2\, dx + \int_0^1 (u')^2\, dx - u'(1,t)u(1,t) + u'(0,t)u(0,t) = \int_0^1 fu\, dx.$$

Then, using boundary conditions and Cauchy-Schwarz' inequality yields

$$||u||\frac{d}{dt}||u|| + ||u'||^2 = \int_0^1 fu\, dx \leq ||f||\,||u||. \tag{6.1.5}$$

Now since $||u'||^2 \geq 0$, consequently $||u||\frac{d}{dt}||u|| \leq ||f||\,||u||$, and thus

$$\frac{d}{dt}||u|| \leq ||f||. \tag{6.1.6}$$

Relabeling the variable from t to s, and integrating over time we end up with

$$||u(\cdot,t)|| - ||u(\cdot,0)|| \leq \int_0^t ||f(\cdot,s)||\, ds, \tag{6.1.7}$$

which yields the first assertion (6.1.2) of the theorem. To prove (6.1.3) we multiply the differential equation by $\dot{u}$, integrate over $(0,1)$, and use integration by parts so that we have on the left hand side

$$\int_0^1 (\dot{u})^2\,dx - \int_0^1 u''\dot{u}\,dx = ||\dot{u}||^2 + \int_0^1 u'\dot{u}'\,dx - u'(1,t)\dot{u}(1,t) + u'(0,t)\dot{u}(0,t).$$

Then, since $u(0,t) = 0 \Longrightarrow \dot{u}(0,t) = 0$, we have

$$||\dot{u}||^2 + \frac{1}{2}\frac{d}{dt}||u'||^2 = \int_0^1 f\dot{u}\,dx \le ||f||\,||\dot{u}|| \le \frac{1}{2}\Big(||f||^2 + ||\dot{u}||^2\Big), \qquad (6.1.8)$$

where in the last step we used Cauchy-Schwarz' inequality. Hence,

$$\frac{1}{2}||\dot{u}||^2 + \frac{1}{2}\frac{d}{dt}||u'||^2 \le \frac{1}{2}||f||^2, \qquad (6.1.9)$$

and therefore, evidently,

$$\frac{d}{dt}||u'||^2 \le ||f||^2. \qquad (6.1.10)$$

Finally, integrating over $(0,t)$ we get the second assertion of the theorem:

$$||u'(\cdot,t)||^2 - ||u'(\cdot,0)||^2 \le \int_0^t ||f(\cdot,s)||^2\,ds, \qquad (6.1.11)$$

and the proof is complete. □

To proceed we give a proof of the *Poincare inequality* (in 1d) which is one of the most useful inequalities in PDE and analysis.

Theorem 6.2 (The Poincare inequality in $1-d$)**.** *Assume that u and u' are square integrable functions on an interval $[0,L]$. Then, there exists a constant C_L, independent of u, but dependent on L, such that if $u(0) = 0$,*

$$\int_0^L u(x)^2\,dx \le C_L \int_0^L u'(x)^2\,dx, \quad \textit{i.e.} \quad ||u|| \le \sqrt{C_L}||u'||. \qquad (6.1.12)$$

Proof. For $x \in [0,L]$ we may write

$$\begin{aligned} u(x) &= \int_0^x u'(y)\,dy \le \int_0^x |u'(y)|\,dy = \int_0^x |u'(y)|\cdot 1\,dy \\ &\le \Big(\int_0^x |u'(y)|^2\,dy\Big)^{1/2}\cdot\Big(\int_0^x 1^2 dy\Big)^{1/2} \\ &\le \Big(\int_0^L |u'(y)|^2\,dy\Big)^{1/2}\cdot\Big(\int_0^L 1^2 dy\Big)^{1/2} = \sqrt{L}\Big(\int_0^L |u'(y)|^2\,dy\Big)^{1/2}, \end{aligned}$$

where in the last step we used the Cauchy-Schwarz inequality. Thus, squaring both sides and integrating, we get

$$\int_0^L u(x)^2\,dx \le \int_0^L L\Big(\int_0^L |u'(y)|^2\,dy\Big)\,dx = L^2\int_0^L |u'(y)|^2\,dy, \qquad (6.1.13)$$

and hence

$$||u|| \le L||u'||. \qquad (6.1.14)$$

□

Remark 6.2. *The constant $C_L = L^2$ indicates that the Poincare inequality is valid for arbitrary bounded intervals, but not for unbounded intervals. If $u(0) \neq 0$ and, for simplicity $L = 1$, then by a similar argument as above we get the following version of the one-dimensional Poincare inequality:*

$$||u||^2_{L_2(0,1)} \le 2\Big(u(0)^2 + ||u'||^2_{L_2(0,1)}\Big). \qquad (6.1.15)$$

Theorem 6.3 (Stability of the homogeneous heat equation)**.** *The initial boundary value problem for the heat equation*

$$\begin{cases} \dot{u} - u'' = 0, & 0 < x < 1, \quad t > 0 \\ u(0,t) = u_x(1,t) = 0, & \qquad\quad t > 0 \\ u(x,0) = u_0(x), & 0 < x < 1, \end{cases} \qquad (6.1.16)$$

satisfies the following stability estimates

$$a) \quad \frac{d}{dt}||u||^2 + 2||u'||^2 = 0, \qquad b) \quad ||u(\cdot,t)|| \le e^{-t}||u_0||.$$

Proof. a) Multiply the equation by u and integrate over $x \in (0,1)$, to get

$$0 = \int_0^1 (\dot{u} - u'')u\,dx = \int_0^1 \dot{u}u\,dx + \int_0^1 (u')^2\,dx - u'(1,t)u(1,t) + u'(0,t)u(0,t),$$

where we used integration by parts. Using the boundary data we then have

$$\frac{1}{2}\frac{d}{dt}\int_0^1 u^2\,dx + \int_0^1 (u')^2\,dx = \frac{1}{2}\frac{d}{dt}||u||^2 + ||u'||^2 = 0.$$

This gives the proof of a). As for the proof of b), using a) and the Poincare inequality, with $L = 1$, i.e., $||u|| \leq ||u'||$ we get

$$\frac{d}{dt}||u||^2 + 2||u||^2 \leq 0. \tag{6.1.17}$$

Multiplying both sides of (6.1.17) by the integrating factor e^{2t} yields

$$\frac{d}{dt}\Big(||u||^2 e^{2t}\Big) = \Big(\frac{d}{dt}||u||^2 + 2||u||^2\Big)e^{2t} \leq 0. \tag{6.1.18}$$

We replace t by s and integrate with respect to s, over $(0, t)$, to obtain

$$\int_0^t \frac{d}{ds}\Big(||u||^2 e^{2s}\Big)\, ds = ||u(\cdot, t)||^2 e^{2t} - ||u(\cdot, 0)||^2 \leq 0. \tag{6.1.19}$$

This yields

$$||u(\cdot, t)||^2 \leq e^{-2t}||u_0||^2 \Longrightarrow ||u(\cdot, t)|| \leq e^{-t}||u_0||, \tag{6.1.20}$$

and completes the proof. □

6.1.2 FEM for the heat equation

We consider the one-dimensional heat equation with Dirichlet boundary data:

$$\begin{cases} \dot{u} - u'' = f, & 0 < x < 1, \quad t > 0, \\ u(0, t) = u(1, t) = 0, & \quad t > 0, \\ u(x, 0) = u_0(x), & 0 < x < 1. \end{cases} \tag{6.1.21}$$

The *Variational formulation* for this problem reads as follows: For every time interval $I_n = (t_{n-1}, t_n]$, find $u(x, t),\ x \in (0, 1),\ t \in I_n$, such that

$$\int_{I_n}\int_0^1 (\dot{u}v + u'v')dxdt = \int_{I_n}\int_0^1 fvdxdt, \quad \forall v: \ \ v(0, t) = v(1, t) = 0. \ \ \text{(VF)}$$

A piecewise linear Galerkin finite element method: $cG(1) - cG(1)$ is then formulated as: for each time interval $I_n := (t_{n-1}, t_n]$, with $t_n - t_{n-1} = k_n$, let

$$U(x, t) = U_{n-1}(x)\Psi_{n-1}(t) + U_n(x)\Psi_n(t), \tag{6.1.22}$$

where

$$\Psi_n(t) = \frac{t - t_{n-1}}{k_n}, \qquad \Psi_{n-1}(t) = \frac{t_n - t}{k_n}, \tag{6.1.23}$$

and

$$U_{\tilde{n}}(x) = U_{\tilde{n},1}\varphi_1(x) + U_{\tilde{n},2}\varphi_2(x) + \ldots + U_{\tilde{n},m}\varphi_m(x), \quad \tilde{n} = n-1,\ n \tag{6.1.24}$$

with φ_j being the usual continuous, piecewise linear finite element basis functions (hat-functions) corresponding to a partition of $\Omega = (0,1)$, with $0 = x_0 < \cdots < x_\ell < x_{\ell+1} < \cdots < x_{m+1} = 1$, and $\varphi_j(x_i) := \delta_{ij}$. Now the Galerkin method (FEM) is to determine the unknown coefficients $U_{n,\ell}$ in the above representation for U (U is a continuous, piecewise linear function both in space and time variables) that satisfies the following discrete variational formulation: Find $U(x,t)$ given by (6.1.22) such that

$$\int_{I_n}\int_0^1 (\dot{U}\varphi_i + U'\varphi_i')\,dxdt = \int_{I_n}\int_0^1 f\varphi_i\,dxdt, \qquad i = 1,2,\ldots,m. \tag{6.1.25}$$

Note that, on $I_n = (t_{n-1}, t_n]$ and with $U_n(x) := U(x,t_n)$ and $U_{n-1}(x) := U(x,t_{n-1})$,

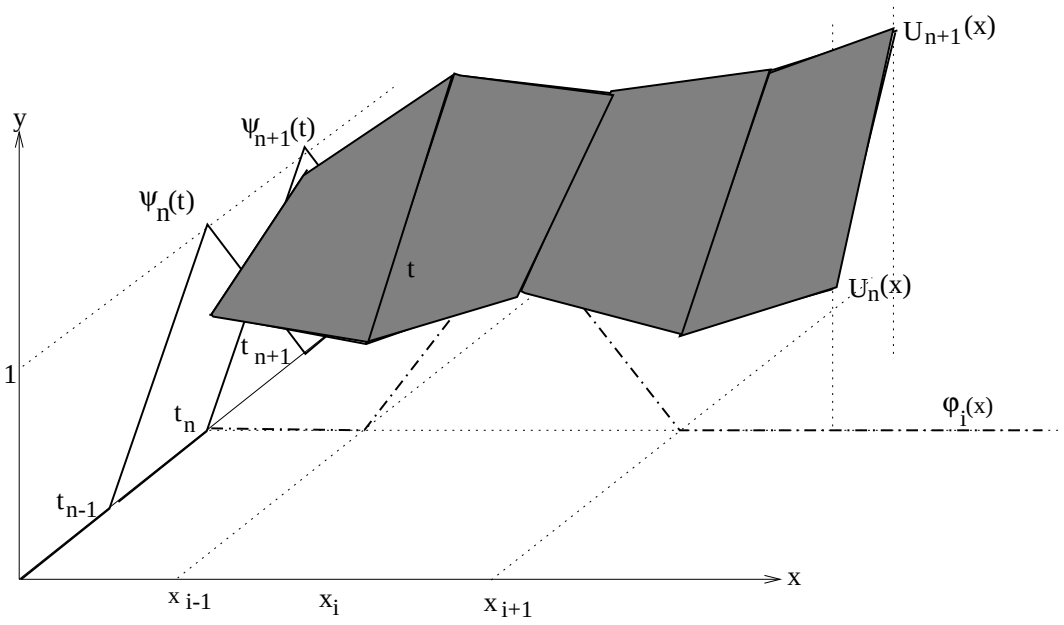
$$\dot{U}(x,t) = U_{n-1}(x)\dot{\Psi}_{n-1}(t) + U_n(x)\dot{\Psi}_n(t) = \frac{U_n - U_{n-1}}{k_n}. \tag{6.1.26}$$

Further differentiating (6.1.22) with respect to x we have

$$U'(x,t) = U_{n-1}'(x)\Psi_{n-1}(t) + U_n'(x)\Psi_n(t). \tag{6.1.27}$$

Inserting (6.1.26) and (6.1.27) into (6.1.25) we get using the identities, $\int_{I_n} dt = k_n$ and $\int_{I_n}\Psi_n dt = \int_{I_n}\Psi_{n-1}dt = k_n/2$ that,

$$\begin{aligned}\underbrace{\int_0^1 U_n\varphi_i dx}_{M\cdot U_n} - \underbrace{\int_0^1 U_{n-1}\varphi_i\,dx}_{M\cdot U_{n-1}} + \underbrace{\int_{I_n}\Psi_{n-1}\,dt}_{k_n/2}\underbrace{\int_0^1 U_{n-1}'\varphi_i'\,dx}_{S\cdot U_{n-1}} \\ + \underbrace{\int_{I_n}\Psi_n\,dt}_{k_n/2}\underbrace{\int_0^1 U_n'\varphi_i'\,dx}_{S\cdot U_n} = \underbrace{\int_{I_n}\int_0^1 f\varphi_i\,dxdt}_{F_n}.\end{aligned} \tag{6.1.28}$$



This can be written in a compact form as the *Crank-Nicolson system*

$$\Big(M + \frac{k_n}{2}S\Big)U_n = \Big(M - \frac{k_n}{2}S\Big)U_{n-1} + F_n, \qquad \text{(CNS)}$$

with the solution U_n given by the data U_{n-1} and F, viz

$$U_n = \underbrace{\Big(M + \frac{k_n}{2}S\Big)^{-1}}_{B^{-1}}\underbrace{\Big(M - \frac{k_n}{2}S\Big)}_{A}U_{n-1} + \underbrace{\Big(M + \frac{k_n}{2}S\Big)^{-1}}_{B^{-1}}F_n, \qquad (6.1.29)$$

where M and S (computed below) are known as the *mass-matrix* and *stiffness-matrix*, respectively, and

$$U_n = \begin{bmatrix} U_{n,1} \\ U_{n,2} \\ \dots \\ U_{n,m} \end{bmatrix}, \qquad F = \begin{bmatrix} F_{n,1} \\ F_{n,2} \\ \dots \\ F_{n,m} \end{bmatrix}, \qquad F_{n,i} = \int_{I_n}\int_0^1 f\varphi_i \, dx\, dt. \quad (6.1.30)$$

Thus, given the source term f we can determine the vector F_n and then, for each $n = 1, \dots N$, given the vector U_{n-1} (the initial value is given by

$U_{0,j} := u_0(x_j)$) we may use the CNS to compute $U_{n,\ell}$, $\ell = 1, 2, \ldots m$ (m nodal values of U at the x_j : s, and at the time level t_n).

We now return to the computation of the matrix entries for M and S, for a uniform partition (all subintervals are of the same length) of the interval $I = (0, 1)$. Note that differentiating (6.1.24) with respect to x, yields

$$U_n'(x) = U_{n,1}\varphi_1'(x) + U_{n,2}\varphi_2'(x) + \ldots + U_{n,m}\varphi_m'(x). \tag{6.1.31}$$

Hence, for $i = 1, \ldots, m$, the rows in the system of equations are given by

$$\int_0^1 U_n'\varphi_i' = \Big(\int_0^1 \varphi_i'\varphi_1'\Big)U_{n,1} + \Big(\int_0^1 \varphi_i'\varphi_2'\Big)U_{n,2} + \ldots + \Big(\int_0^1 \varphi_i'\varphi_m'\Big)U_{n,m},$$

which can be written in matrix form as

$$SU_n = \begin{bmatrix} \int_0^1 \varphi_1'\varphi_1' & \int_0^1 \varphi_1'\varphi_2' & \ldots & \int_0^1 \varphi_1'\varphi_m' \\ \int_0^1 \varphi_2'\varphi_1' & \int_0^1 \varphi_2'\varphi_2' & \ldots & \int_0^1 \varphi_2'\varphi_m' \\ \ldots & \ldots & \ldots & \ldots \\ \int_0^1 \varphi_m'\varphi_1' & \int_0^1 \varphi_m'\varphi_2' & \ldots & \int_0^1 \varphi_m'\varphi_m' \end{bmatrix} \begin{bmatrix} U_{n,1} \\ U_{n,2} \\ \ldots \\ U_{n,m} \end{bmatrix}. \tag{6.1.32}$$

Thus, S is just the stiffness matrix $\mathbf{A}_{unif}$ computed in Chapter 2:

$$S = \frac{1}{h}\begin{bmatrix} 2 & -1 & 0 & 0 & \ldots & 0 \\ -1 & 2 & -1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & -1 & 2 & -1 \\ 0 & \ldots & \ldots & \ldots & -1 & 2 \end{bmatrix}. \tag{6.1.33}$$

A non-uniform partition yields a matrix of the form $\mathbf{A}$ in Chapter 2.
Similarly, recalling the notation for the mass matrix M i in (6.1.28), we have

$$[MU_n]_i = \int_0^1 U_n\varphi_i, \quad i = 1, \ldots, m. \tag{6.1.34}$$

Hence, to compute the mass matrix M one should drop all derivatives from the general form of the matrix for S given by (6.1.32). In other words unlike

the form $[SU_n]_i = \int_0^1 U_n'\varphi_i'$, MU_n does not involve any derivatives, neither in U_n nor in φ_i. Consequently

$$M = \begin{bmatrix} \int_0^1 \varphi_1\varphi_1 & \int_0^1 \varphi_1\varphi_2 & \dots & \int_0^1 \varphi_1\varphi_m \\ \int_0^1 \varphi_2\varphi_1 & \int_0^1 \varphi_2\varphi_2 & \dots & \int_0^1 \varphi_2\varphi_m \\ \dots & \dots & \dots & \dots \\ \int_0^1 \varphi_m\varphi_1 & \int_0^1 \varphi_m\varphi_2 & \dots & \int_0^1 \varphi_m\varphi_m \end{bmatrix}. \tag{6.1.35}$$

For a uniform partition, we have computed this mass matrix in Chapter 4:

$$M = h\begin{bmatrix} \frac{2}{3} & \frac{1}{6} & 0 & 0 & \dots & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ 0 & \dots & \dots & \dots & \frac{1}{6} & \frac{2}{3} \end{bmatrix} = \frac{h}{6}\begin{bmatrix} 4 & 1 & 0 & 0 & \dots & 0 \\ 1 & 4 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 1 & 4 & 1 \\ 0 & \dots & \dots & \dots & 1 & 4 \end{bmatrix}.$$

6.1.3 Exercises

Problem 6.1. *Derive a system of equations, as* (6.1.29)*, for $cG(1) - dG(0)$: with the discontinuous Galerkin approximation $dG(0)$ in time with piecewise constants.*

Problem 6.2. *Let $\|\cdot\|$ denote the $L_2(0,1)$-norm. Consider the problem*

$$\begin{cases} -u'' = f, & 0 < x < 1, \\ u'(0) = v_0, & u(1) = 0. \end{cases}$$

a) Show that $|u(0)| \le \|u'\|$ and $\|u\| \le \|u'\|$.

b) Use a) to show that $\|u'\| \le \|f\| + |v_0|$.

Problem 6.3. *Assume that $u = u(x)$ satisfies*

$$\int_0^1 u'v'dx = \int_0^1 fv\,dx, \quad \text{for all } v(x) \text{ such that } v(0) = 0. \tag{6.1.36}$$

Show that $-u'' = f$ for $0 < x < 1$ and $u'(1) = 0$.
Hint: See previous chapters.

Problem 6.4 (Generalized Poincare)**.** *Show that for a continuously differentiable function v defined on $(0,1)$ we have that*

$$||v||^2 \leq v(0)^2 + v(1)^2 + ||v'||^2.$$

Hint: Use partial integration for $\int_0^{1/2} v(x)^2\,dx$ and $\int_{1/2}^1 v(x)^2\,dx$ and note that $(x-1/2)$ has the derivative 1.

Problem 6.5. *Let $\|\cdot\|$ denote the $L_2(0,1)$-norm. Consider the following heat equation*

$$\begin{cases} \dot{u} - u'' = 0, & 0 < x < 1, \quad t > 0, \\ u(0,t) = u_x(1,t) = 0, & t > 0, \\ u(x,0) = u_0(x), & 0 < x < 1. \end{cases}$$

a) Show that the norms: $||u(\cdot,t)||$ and $||u'(\cdot,t)||$ are non-increasing in time. $||u|| = \left(\int_0^1 u(x)^2\,dx\right)^{1/2}$.

b) Show that $||u'(\cdot,t)|| \to 0$, as $t \to \infty$.

c) Give a physical interpretation for a) and b).

Problem 6.6. *Consider the inhomogeneous problem:*

$$\begin{cases} \dot{u} - \varepsilon u'' = f, & 0 < x < 1, \quad t > 0, \\ u(0,t) = u_x(1,t) = 0, & t > 0, \\ u(x,0) = u_0(x), & 0 < x < 1. \end{cases}$$

where $f = f(x,t)$.
a) Show the stability estimate

$$||u(\cdot,t)|| \leq \int_0^t ||f(\cdot,s)||\,ds.$$

b) Show that for the corresponding stationary ($\dot{u} \equiv 0$) problem we have

$$||u'|| \leq \frac{1}{\varepsilon}||f||.$$

Problem 6.7. *Give an a priori error estimate for the following problem:*

$$(au'')'' = f, \quad 0 < x < 1, \qquad u(0) = u'(0) = u(1) = u'(1) = 0,$$

where $a(x) > 0$ on the interval $I = (0,1)$.

6.2 The wave equation in 1d

The theoretical study of the wave equation has some basic differences compared to that of the heat equation. Some important aspects in this regard are given in extended version of these notes. In our study here, the finite element procedure for the wave equation is, mainly, the same as for that of the heat equation outlined in the previous section. We start with an example of the homogeneous wave equation, as an initial-boundary value problem:

$$\begin{cases} \ddot{u} - u'' = 0, & 0 < x < 1 \quad t > 0 \quad (DE) \\ u(0,t) = 0, \quad u(1,t) = 0 & t > 0 \quad (BC) \\ u(x,0) = u_0(x), \quad \dot{u}(x,0) = v_0(x), & 0 < x < 1. \quad (IC) \end{cases} \tag{6.2.1}$$

Theorem 6.4 (conservation of energy)**.** *For the equation* (6.2.1) *we have*

$$\frac{1}{2}||\dot{u}||^2 + \frac{1}{2}||u'||^2 = \frac{1}{2}||v_0||^2 + \frac{1}{2}||u_0'||^2 = \textit{Constant}, \tag{6.2.2}$$

where

$$||w||^2 = ||w(\cdot,t)||^2 = \int_0^1 |w(x,t)|^2\, dx. \tag{6.2.3}$$

Proof. We multiply the equation by $\dot{u}$ and integrate over $I = (0,1)$ to get

$$\int_0^1 \ddot{u}\,\dot{u} dx - \int_0^1 u''\,\dot{u}\, dx = 0. \tag{6.2.4}$$

Using integration by parts and the boundary data we obtain

$$\begin{aligned} \int_0^1 \frac{1}{2}\frac{\partial}{\partial t}\Big(\dot{u}\Big)^2 dx + \int_0^1 u'\,(\dot{u})'\, dx - \Big[u'(x,t)\dot{u}(x,t)\Big]_0^1 \\ = \int_0^1 \frac{1}{2}\frac{\partial}{\partial t}\Big(\dot{u}\Big)^2 dx + \int_0^1 \frac{1}{2}\frac{\partial}{\partial t}\Big(u'\Big)^2 dx \\ = \frac{1}{2}\frac{d}{dt}\Big(||\dot{u}||^2 + ||u'||^2\Big) = 0. \end{aligned} \tag{6.2.5}$$

Thus, we have that the quantity

$$\frac{1}{2}||\dot{u}||^2 + \frac{1}{2}||u'||^2 = \text{Constant, independent of } t. \tag{6.2.6}$$

Therefore the total energy is conserved. We recall that $\frac{1}{2}||\dot{u}||^2$ is the kinetic energy, and $\frac{1}{2}||u'||^2$ is the potential (elastic) energy.

□

Problem 6.8. *Show that* $\|(\dot{u})'\|^2 + \|u''\|^2 =$ *constant, independent of* t. *Hint: Differentiate the equation with respect to* x *and multiply by* $\dot{u}, \ldots$. *Alternatively: Multiply (DE):* $\ddot{u} - u'' = 0$, *by* $-(\dot{u})''$ *and integrate over* I.

Problem 6.9. *Derive a total conservation of energy relation using the Robin type boundary condition:* $u' + u = 0$.

6.2.1 Wave equation as a system of PDEs

We rewrite the wave equation as a system of differential equations. To this approach, we consider solving

$$\begin{cases} \ddot{u} - u'' = 0, & 0 < x < 1, & t > 0, \\ u(0,t) = 0, & u'(1,t) = g(t), & t > 0, \\ u(x,0) = u_0(x), & \dot{u}(x,0) = v_0(x), & 0 < x < 1, \end{cases} \tag{6.2.7}$$

where we let $\dot{u} = v$, and reformulate the problem as:

$$\begin{cases} \dot{u} - v = 0, & \text{(Convection)} \\ \dot{v} - u'' = 0, & \text{(Diffusion)}. \end{cases} \tag{6.2.8}$$

We may now set $w = (u, v)^t$ and rewrite the system (6.2.8) as $\dot{w} + Aw = 0$:

$$\dot{w} + Aw = \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} + \begin{pmatrix} 0 & -1 \\ -\frac{\partial^2}{\partial x^2} & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{6.2.9}$$

In other words, the matrix differential operator is given by

$$A = \begin{pmatrix} 0 & -1 \\ -\frac{\partial^2}{\partial x^2} & 0 \end{pmatrix}.$$

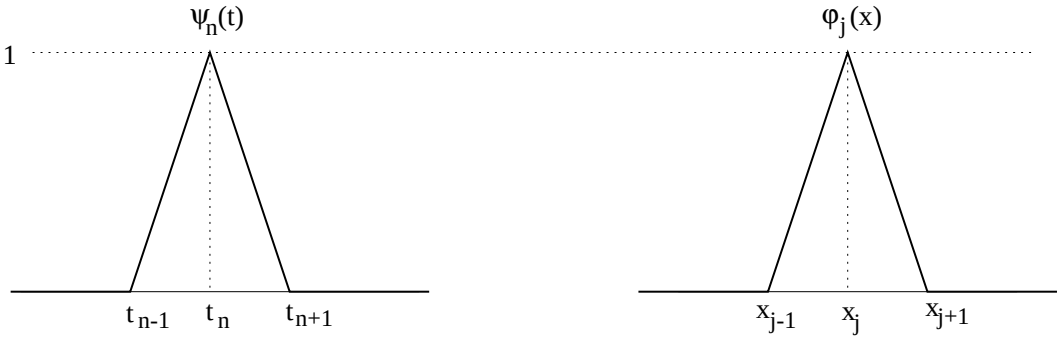
6.2.2 The finite element discretization procedure

We follow the same procedure as in the case of the heat equation, and let $S_n = \Omega \times I_n$, $n = 1, 2, \ldots, N$, with $I_n = (t_{n-1}, t_n]$. Then, for each n we define, on S_n, the piecewise linear approximations

$$\begin{cases} U(x,t) = U_{n-1}(x)\Psi_{n-1}(t) + U_n(x)\Psi_n(t), \\ V(x,t) = V_{n-1}(x)\Psi_{n-1}(t) + V_n(x)\Psi_n(t), \end{cases} \qquad 0 < x < 1, \quad t \in I_n, \tag{6.2.10}$$

where, e.g.

$$\begin{cases} U_{\tilde{n}}(x) = U_{\tilde{n},1}\varphi_1(x) + \ldots + U_{\tilde{n},m}\varphi_m(x), \;\; \tilde{n} = n-1,\, n \\ V_{\tilde{n}}(x) = V_{\tilde{n},1}\varphi_1(x) + \ldots + V_{\tilde{n},m}\varphi_m(x), \;\; \tilde{n} = n-1,\, n. \end{cases} \tag{6.2.11}$$



For $\dot{u} - v = 0$ and $t \in I_n$ we write the general variational formulation

$$\int_{I_n}\int_0^1 \dot{u}\varphi \, dxdt - \int_{I_n}\int_0^1 v\varphi \, dxdt = 0, \qquad \text{for all} \;\; \varphi(x,t). \tag{6.2.12}$$

Likewise, $\dot{v} - u'' = 0$ yields a variational formulation, viz

$$\int_{I_n}\int_0^1 \dot{v}\varphi \, dxdt - \int_{I_n}\int_0^1 u''\varphi \, dxdt = 0. \tag{6.2.13}$$

Integrating by parts in x, in the second term, and using the boundary condition $u'(1,t) = g(t)$ we get

$$\int_0^1 u''\varphi dx = [u'\varphi]_0^1 - \int_0^1 u'\varphi' \, dx = g(t)\varphi(1,t) - u'(0,t)\varphi(0,t) - \int_0^1 u'\varphi' \, dx.$$

Inserting the right hand side in (6.2.13) we get for all φ with $\varphi(0,t)=0$:

$$\int_{I_n}\int_0^1 \dot{v}\varphi\,dxdt+\int_{I_n}\int_0^1 u'\varphi'\,dxdt=\int_{I_n} g(t)\varphi(1,t)\,dt. \tag{6.2.14}$$

The corresponding $cG(1)cG(1)$ finite element method reads as follows: For each n, $n=1,2,\dots,N$, find continuous piecewise linear functions $U(x,t)$ and $V(x,t)$, in a partition, $0=x_0<x_1<\cdots<x_m=1$ of $\Omega=(0,1)$, such that

$$\begin{aligned}\int_{I_n}\int_0^1 &\frac{U_n(x)-U_{n-1}(x)}{k_n}\varphi_j(x)\,dxdt\\ &-\int_{I_n}\int_0^1\Big(V_{n-1}(x)\Psi_{n-1}(t)+V_n(x)\Psi_n(t)\Big)\varphi_j(x)\,dxdt=0,\\ &\qquad\qquad\qquad\qquad\text{for } j=1,2,\dots,m,\end{aligned} \tag{6.2.15}$$

and

$$\begin{aligned}\int_{I_n}\int_0^1 &\frac{V_n(x)-V_{n-1}(x)}{k_n}\varphi_j(x)\,dxdt\\ &+\int_{I_n}\int_0^1\Big(U'_{n-1}(x)\Psi_{n-1}(t)+U'_n(x)\Psi_n(t)\Big)\varphi'_j(x)\,dxdt\\ &\qquad=\int_{I_n} g(t)\varphi_j(1)\,dt,\qquad\text{for } j=1,2,\dots,m,\end{aligned} \tag{6.2.16}$$

where $\dot{U}$, U', $\dot{V}$, and V' are computed using (6.2.10) with

$$\psi_{n-1}(t)=\frac{t_n-t}{k_n},\qquad \psi_n(t)=\frac{t-t_{n-1}}{k_n},\qquad k_n=t_n-t_{n-1}.$$

Thus, the equations (6.2.15) and (6.2.16) are reduced to the *iterative forms*:

$$\begin{aligned}&\underbrace{\int_0^1 U_n(x)\varphi_j(x)dx}_{MU_n}-\frac{k_n}{2}\underbrace{\int_0^1 V_n(x)\varphi_j(x)dx}_{MV_n}\\ &\quad=\underbrace{\int_0^1 U_{n-1}(x)\varphi_j(x)dx}_{MU_{n-1}}+\frac{k_n}{2}\underbrace{\int_0^1 V_{n-1}(x)\varphi_j(x)\,dx}_{MV_{n-1}},\quad j=1,2,\dots,m,\end{aligned}$$

and

$$\underbrace{\int_0^1 V_n(x)\varphi_j(x)dx}_{MV_n} + \frac{k_n}{2}\underbrace{\int_0^1 U_n'(x)\varphi_j'(x)\,dx}_{SU_n}$$
$$= \underbrace{\int_0^1 V_{n-1}(x)\varphi_j(x)\,dx}_{MV_{n-1}} - \frac{k_n}{2}\underbrace{\int_0^1 U_{n-1}'(x)\varphi_j'(x)\,dx}_{SU_{n-1}} + g_n, \quad j = 1, 2, \ldots, m,$$

respectively, where we used (6.2.11) and as we computed earlier

$$S = \frac{1}{h}\begin{bmatrix} 2 & -1 & \ldots & 0 \\ -1 & 2 & -1 & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad M = \frac{h}{6}\begin{bmatrix} 4 & 1 & \ldots & 0 \\ 1 & 4 & 1 & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & 1 & 4 & 1 \\ 0 & \ldots & 1 & 2 \end{bmatrix},$$

where

$$g_n = (0, \ldots, 0, g_{n,m})^T, \quad \text{where} \quad g_{n,m} = \int_{I_n} g(t)\,dt.$$

In compact form the vectors U_n and V_n are determined by solving the linear system of equations:

$$\begin{cases} MU_n - \frac{k_n}{2}MV_n = MU_{n-1} + \frac{k_n}{2}MV_{n-1} \\ \frac{k_n}{2}SU_n + MV_n = -\frac{k_n}{2}SU_{n-1} + MV_{n-1} + g_n, \end{cases} \tag{6.2.17}$$

which is a system of $2m$ equations with $2m$ unknowns:

$$\underbrace{\begin{bmatrix} M & -\frac{k_n}{2}M \\ \frac{k_n}{2}S & M \end{bmatrix}}_{A}\underbrace{\begin{bmatrix} U_n \\ V_n \end{bmatrix}}_{W} = \underbrace{\begin{bmatrix} M & \frac{k_n}{2}M \\ -\frac{k_n}{2}S & M \end{bmatrix}\begin{bmatrix} U_{n-1} \\ V_{n-1} \end{bmatrix} + \begin{bmatrix} 0 \\ g_n \end{bmatrix}}_{b},$$

with $W = A^{-1}b$, $U_n = W(1:m)$ and $V_n = W(m+1:2m)$.

6.2.3 Exercises

Problem 6.10. *Derive the corresponding linear system of equations in the case of time discretization with dG(0).*

Problem 6.11 (discrete conservation of energy)**.** *Show that cG(1)-cG(1) for the wave equation in system form with $g(t) = 0$, conserves energy: i.e.*

$$\|U_n'\|^2 + \|V_n\|^2 = \|U_{n-1}'\|^2 + \|V_{n-1}\|^2. \tag{6.2.18}$$

Hint: Multiply the first equation by $(U_{n-1} + U_n)^t SM^{-1}$ and the second equation by $(V_{n-1}+V_n)^t$ and add up. Use then, e.g., the fact that $U_n^t S U_n = \|U_n'\|^2$, where

$$U_n = \begin{pmatrix} U_{n,1} \\ U_{n,2} \\ \dots \\ U_{n,m} \end{pmatrix}, \text{ and } U_n = U_n(x) = U_{n,1}(x)\varphi_1(x) + \ldots + U_{n,m}(x)\varphi_m(x).$$

Problem 6.12. *Consider the wave equation*

$$\begin{cases} \ddot{u} - u'' = 0, & x \in R, \quad t > 0, \\ u(x,0) = u_0(x), & x \in R, \\ \dot{u}(x,0) = v_0(x), & x \in R. \end{cases} \tag{6.2.19}$$

Plot the graph of $u(x, 2)$ in the following cases.
a) $v_0 = 0$ and

$$u_0(x) = \begin{cases} 1, & x < 0, \\ 0, & x > 0. \end{cases}$$

b) $u_0 = 0$, and

$$v_0(x) = \begin{cases} -1, & -1 < x < 0, \\ 1, & 0 < x < 1, \\ 0, & |x| > 0. \end{cases}$$

Problem 6.13. *Compute the solution for the wave equation*

$$\begin{cases} \ddot{u} - 4u'' = 0, & x > 0, & t > 0, \\ u(0,t) = 0, & & t > 0, \\ u(x,0) = u_0(x), & \dot{u}(x,0) = 0, & x > 0. \end{cases} \tag{6.2.20}$$

Plot the solutions for the three cases $t = 0.5$, $t = 1$, $t = 2$, and with

$$u_0(x) = \begin{cases} 1, & x \in [2,3] \\ 0, & else \end{cases} \tag{6.2.21}$$

Problem 6.14. *Apply cG(1) time discretization directly to the wave equation by letting*

$$U(x,t) = U_{n-1}\Psi_{n-1}(t) + U_n(x)\Psi_n(t), \qquad t \in I_n. \tag{6.2.22}$$

Note that $\dot{U}$ is piecewise constant in time and comment on:

$$\underbrace{\int_{I_n}\int_0^1 \ddot{U}\varphi_j \, dxdt}_{?} + \underbrace{\int_{I_n}\int_0^1 u'\varphi_j' \, dxdt}_{\frac{k}{2}S(U_{n-1}+U_n)} = \underbrace{\int_{I_n} g(t)\varphi_j(1)dt}_{g_n}, \quad j = 1, 2, \ldots, m.$$

Problem 6.15. *Construct a FEM for the problem*

$$\begin{cases} \ddot{u} + \dot{u} - u'' = f, & 0 < x < 1, & t > 0, \\ u(0,t) = 0, & u'(1,t) = 0, & t > 0, \\ u(x,0) = 0, & \dot{u}(x,0) = 0, & 0 < x < 1. \end{cases} \tag{6.2.23}$$

Problem 6.16. *Determine the solution for the wave equation*

$$\begin{cases} \ddot{u} - c^2u'' = f, & x > 0, & t > 0, \\ u(x,0) = u_0(x), & u_t(x,0) = v_0(x), & x > 0, \\ u_x(1,t) = 0, & u(0,t) = 0 & t > 0, \end{cases}$$

in the following cases:

a) $f = 0$.

b) $f = 1, \quad u_0 = 0, \quad v_0 = 0$.

Problem 6.17. *Prove that the solution u of the convection-diffusion problem*

$$-u_{xx} + u_x + u = f, \quad \text{in } I = (0,1), \quad u(0) = u(1) = 0,$$

satisfies the following estimate

$$\Big(\int_I u^2 \phi \, dx \Big)^{1/2} \leq \Big(\int_I f^2 \phi \, dx \Big)^{1/2}.$$

where $\phi(x)$ is a positive weight function defined on $(0,1)$ satisfying $\phi'(x) \leq 0$ and $-\phi'(x) \leq \phi(x)$ for $0 \leq x \leq 1$.

Problem 6.18. *Let ϕ be a solution of the problem*

$$-\varepsilon\phi'' - 3\phi' + 2\phi = e, \qquad \phi'(0) = \phi(1) = 0.$$

Let $\|\cdot\|$ denote the L_2-norm on I. Show that there is a constant C such that

$$|\phi(0)| \leq C\|e\|, \qquad \|\varepsilon\phi''\| \leq C\|e\|.$$

Problem 6.19. *Use relevant interpolation theory estimates and prove an a priori error estimate for the $cG(1)$ finite element method for the problem*

$$-u'' + u' = f, \quad \text{in } I = (0,1), \quad u(0) = u(1) = 0.$$

Problem 6.20. *Prove an a priori error estimate for the $cG(1)$ finite element method for the problem*

$$-u'' + u' + u = f, \quad \text{in } I = (0,1), \quad u(0) = u(1) = 0.$$

Problem 6.21. *Consider the problem*

$$-\varepsilon u'' + xu' + u = f, \quad \text{in } I = (0,1), \qquad u(0) = u'(1) = 0,$$

where ε is a positive constant, and $f \in L_2(I)$. Prove that

$$||\varepsilon u''|| \leq ||f||.$$

Problem 6.22. *We modify the problem 6.21 above according to*

$$-\varepsilon u'' + c(x)u' + u = f(x) \quad 0 < x < 1, \qquad u(0) = u'(1) = 0,$$

where ε is a positive constant, the function c satisfies $c(x) \geq 0$, $c'(x) \leq 0$, and $f \in L_2(I)$. Prove that there are positive constants C_1, C_2 and C_3 such that

$$\sqrt{\varepsilon}||u'|| \leq C_1||f||, \quad ||cu'|| \leq C_2||f||, \qquad \text{and} \quad \varepsilon||u''|| \leq C_3||f||,$$

where $||\cdot||$ is the $L_2(I)$-norm.

Problem 6.23. *Consider the convection-diffusion-absorption problem*

$$-\varepsilon u'' + u' + u = f, \quad \text{in} \quad I = (0,1), \quad u(0) = 0, \qquad \sqrt{\varepsilon}u'(1) + u(1) = 0,$$

where ε is a positive constant, and $f \in L_2(I)$. Prove the following stability estimates for the solution u

$$\|\sqrt{\varepsilon}u'\| + \|u\| + |u(1)| \leq C\|f\|,$$

$$\|u'\| + \|\varepsilon u''\| \leq C\|f\|,$$

where $\|\cdot\|$ denotes the $L_2(I)$-norm, $I = (0,1)$, and C is an appropriate constant.

Appendix A

Answers to Exercises

Chapter 1

1.1 a) $u(x) = C_1 e^x + C_2 e^{2x}$ b) $u(x) = C_1 \cos 2x + C_2 \sin 2t$ c) $u(x) = (C_1 + C_2 x)e^{3x}$

1.2 a) $u(x) = x^2/2 + e^{-x}(A \cos x + B \sin x)$

b) $u(x) = \frac{1}{2}(\sin x - \cos x) + e^{-x/2}(\cos(\sqrt{7}/2)x + \sin(\sqrt{7}/2)x)$

c) $u(x) = C_1 e^{-x} + C_2 e^{-2x} + \frac{1}{6}e^x$.

1.3 a) $u(x) = -\frac{1}{6}x^3 - \frac{1}{4}x^2 - \frac{1}{4}x$ b) $u(x) = -\frac{1}{2}x \cos x$

c) $u(x) = \frac{1}{6}e^x + \frac{1}{10}(\sin x - 3\cos x)$.

1.5 b) No solution.

Chapter 2.

2.2 $q = 1, \quad U(t) = 1 + 3t. \qquad q = 2, \quad U(t) = 1 + \frac{8}{11}t + \frac{10}{11}t^2.$

$q = 3, \quad U(t) = 1 + \frac{30}{29}t + \frac{45}{116}t^2 + \frac{35}{116}t^3.$

$q = 4, \quad U(t) \approx 1 + 0.9971t + 0.5161t^2 + 0.1311t^3 + 0.0737t^4.$

2.3

$$Pu(t) \approx 0.9991 + 1.083t + 0.4212t^2 + 0.2786t^3.$$

2.4
$$A = \begin{bmatrix} 8 & -4 & 0 & \\ -4 & 8 & & -4 \\ 0 & -4 & 8 & \end{bmatrix}, \qquad \mathbf{b} = (b_i)_{i=1}^3, \quad b_i = \frac{i}{16}.$$

2.5 a. $u(x) = \frac{1}{2}x(1-x)$

b. $R(x) = \pi^2 A \sin \pi x + 4\pi^2 B \sin 2\pi x - 1$

c. $A = 4/\pi^3$ and $B = 0$.

2.6 a.

b. $R(x) = (\pi^2+1)A \sin \pi x + (4\pi^2+1)B \sin 2\pi x + (9\pi^2+1)C \sin 3\pi x - x$

c. $A = \dfrac{2}{\pi(\pi^2+1)}$, $B = -\dfrac{1}{\pi(4\pi^2+1)}$ and $C = \dfrac{2}{3\pi(9\pi^2+1)}$.

2.7 a. $u(x) = \frac{1}{6}(\pi^3 - x^3) + \frac{1}{2}(x^2 - \pi^2)$

b. $R(x) = -U''(x) - x + 1 = \frac{1}{4}\xi_0 \cos\frac{x}{2} + \frac{9}{4}\xi_1 \cos\frac{3x}{2}$

c. $\xi_0 = 8(2\pi - 6)/\pi$ and $\xi_1 = \frac{8}{9}(\frac{2}{9} - \frac{2}{3}\pi)/\pi$.

2.8 $U(x) = (16 \sin x + \frac{16}{27} \sin 3x)/\pi^3 + 2x^2/\pi^2$.

Chapter 3.

3.2 (a) x, (b) 0.

3.3
$$\Pi_1 f(x) = \begin{cases} 4 - 11(x+\pi)/(2\pi), & -\pi \le x \le -\frac{\pi}{2}, \\ 5/4 - (x+\frac{\pi}{2})/(2\pi), & -\frac{\pi}{2} \le x \le 0, \\ 1 - 7x/(2\pi), & 0 \le x \le \frac{\pi}{2}, \\ 3(x-\pi)/(2\pi), & \frac{\pi}{2} \le x \le \pi. \end{cases}$$

3.6 Check the conditions required for a Vector space.

3.7
$$\Pi_1 f(x) = f(a)\frac{2x-a-b}{a-b} + f(\frac{a+b}{2})\frac{2(x-a)}{b-a}.$$

3.8 Hint: Use the procedure in the proof of Theorem 3.1, with somewhat careful estimates at the end.

3.10
$$\pi_4\left(e^{-8x^2}\right) \approx 0.25x^4 - 1.25x^2 + 1.$$

3.11 For example we may choose the following basis:

$$\varphi_{i,j}(x) = \begin{cases} 0, & x \in [x_{i-1}, x_i], \\ \lambda_{i,j}(x), & i = 1, \ldots, m+1, \quad j = 0, 1, 2. \end{cases}$$

$$\lambda_{i,0}(x) = \frac{(x-\xi_i)(x-x_i)}{(x_{i-1}-\xi_i)(x_{i-1}-x_i)}, \quad \lambda_{i,1}(x) = \frac{(x-x_{i-1})(x-x_i)}{(\xi_i-x_{i-1})(\xi_i-x_i)},$$

$$\lambda_{i,2}(x) = \frac{(x-x_{i-1})(x-\xi_i)}{(x_i-x_{i-1})(x_i-\xi_i)}, \qquad \xi_i \in (x_{i-1}, x_i).$$

3.12 This is a special case of problem 2.13.

3.13 This is "trivial".

3.14 Hint: Use Taylor expansion of f about $x = \frac{x_1+x_2}{2}$.

Chapter 4.

4.1 c) $\sin \pi x$, $x \ln x$ and $x(1-x)$ are test functions of this problem. x^2 and $e^x - 1$ are not test functions.

4.3 a) U is the solution for

$$AU = f \Longleftrightarrow 1/h \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} = h \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

with $h = 1/4$.

b) A is invertible, therefore U is unique.

4.6 a) ξ is the solution for

$$2\begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 7 \end{pmatrix}$$

b) $(\xi_1, \xi_2) = 7(1/2, 1)$ and $U(x) = 7x$ (same as the exact solution).

4.7 a) In case of $N = 3$, ξ is the solution for

$$A\xi = f \Longleftrightarrow 1/h\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}\begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} -5 \\ 0 \\ 0 \end{pmatrix}$$

with $h = 1/3$. That is: $(\xi_0, \xi_1, \xi_2) = -\frac{1}{3}(15, 10, 5)$.

b) $U(x) = 5x - 5$ (same as the exact solution).

4.8 a) No solution!

b) Trying to get a finite element approximation ends up with the matrix equation

$$A\xi = f \Longleftrightarrow \begin{pmatrix} 2 & -2 & 0 \\ -2 & 4 & -2 \\ 0 & -2 & 2 \end{pmatrix}\begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$

where the coefficient matrix is singular ($detA = 0$). There is no finite element solution.

4.9 d) $||U||_E^2 = \xi^T A\xi$ (check spectral theorem, linear algebra!)

4.10 For an $M + 1$ partition (here $M = 2$) we get $a_{ii} = 2/h$, $a_{i,i+1} = -1/h$ except $a_{M+1,M+1} = 1/h - 1$, $b_i = 0,\ i = 1, \ldots, M$ and $b_{M+1} = -1$:

a) $U = (0, 1/2, 1, 3/2)$.

b) e.g, $U_3 = U(1) \to 1$, as $k \to \infty$.

4.11 c) Set $\alpha = 2$ and $\beta = 3$ in the general FEM solution:
$\xi = \frac{\alpha}{3}(-1,1,1)^T + \beta(0,0,2)^T$:

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} = 2/3 \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix} + 3 \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}.$$

4.12

$$3 \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} + \frac{1}{18} \begin{bmatrix} 4 & 1 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\Longleftrightarrow \text{(MATLAB)} \quad \xi_1 = \xi_2 = 0.102.$$

4.13 Just follow the procedure in the theory.

4.15 a priori: $||e||_E \leq ||u - \pi_h u||_E$.

4.16 a) $||e'||_a \leq C_i ||h(aU')'||_{1/a}$.

b) The matrix equation:

$$\begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 3 & -2 \\ 0 & 0 & 2 & 4 \end{pmatrix} \begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix} = \begin{pmatrix} -3 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

which yields the approximate solution $U = -3(1/2, 1, 2, 3)^T$.

c) Since a is constant and U is linear on each subinterval we have that

$$(aU')' = a'U' + aU'' = 0.$$

By the a posteriori error estimate we have that $||e'||_a = 0$, i.e. $e' = 0$. Combining with the fact that $e(x)$ is continuous and $e(1) = 0$, we get that $e \equiv 0$, which means that the finite, in this case, coincides with the exact solution.

4.17 a priori: $||e||_{H^1} \leq C_i\Big(||hu''|| + ||h^2u''||\Big)$.

4.18 a) a priori: $||e||_E \leq ||u - v||_E(1 + c)$, and a posteriori: $||e||_E \leq C_i||hR(U)||_{L_2(I)}$.

b) Since $c \geq 0$, the a priori error estimate in a) yields optimality for $c \equiv 0$, i.e. in the case of no convection (does this tell anything to you?).

4.19 a priori: $||e||_{H^1} \leq C_i\Big(||hu''|| + 4||h^2u''||\Big)$.

Chapter 5.

5.1 a) $a_{ij} = \frac{j}{j+i} - \frac{1}{j+i+1}, \qquad b_i = \frac{1}{i+1}, \qquad i, j = 1, 2, \ldots,$

b) $q = 1: \quad U(t) = 1 + 3t. \qquad q = 2: \quad U(t) = 1 + \frac{8}{11}t + \frac{10}{11}t^2.$

5.3 a) $u(t) = e^{-4t} + \frac{1}{32}(8t^2 - 4t + 1)$.

b) $u(t) = e^{\frac{1}{2}t^2} - t + \frac{\sqrt{\pi}}{\sqrt{2}}e^{\frac{1}{2}t^2}erf(\frac{t}{\sqrt{2}}), \quad erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-y^2}\,dy.$

5.4 a) $U_i(x_i) = \frac{[(x_i^3 - x_{i-1}^3)/3] - U_i(x_{i-1})\cdot(2(x_i - x_{i-1}) - 1)}{1 + 2(x_i - x_{i-1})}$

Chapter 6.

6.8 $||e|| \leq ||h^2u_{xx}||$

6.14

$$u(x,t) = \begin{cases} \frac{1}{2}(u_0(x + 2t) + u_0(x - 2t)), & x \geq 2t \\ \frac{1}{2}(u_0(2t + x) + u_0(2t - x)), & x < 2t \end{cases}$$

6.16 a) $u(x,t) = \frac{1}{2}[u_0(x + ct) + u_0(ct - x)] + \frac{1}{2c}\Big(\int_0^{x+ct} v_0 + \int_0^{ct-x} v_0\Big)$.

b) $u(x,t) = \frac{1}{2c}\int_0^t 2c(t - s)\,ds = t^2/2$.

6.19 a priori: $||e||_{H^1} \leq C_i\Big(||hu''|| + ||h^2u''||\Big)$.

6.20 a priori: $||e||_E \leq C_i\Big(||hu''|| + ||h^2u''||\Big)$.

Appendix B

Algorithms and MATLAB Codes

To streamline the computational aspects, we have gathered suggestions for some algorithms and Matlab codes that can be used in implementations. These are simple specific Matlab codes on the concepts such as

- The L_2-projection.
- Numerical integration rules: Midpoint, Trapezoidal, Simpson.
- Finite difference Methods: Forward Euler, Backward Euler, Crank-Nicolson.
- Matrices/vectors: Stiffness- Mass-, and Convection Matrices. Load vector.

The Matlab codes are not optimized for speed, but rather intended to be easy to read.

An algorithm for L_2-projection:

1. Choose a partition $\mathcal{T}_h$ of the interval I into N sub-intervals, $N+1$ nodes, and define the corresponding space of piece-wise linear functions V_h.
2. Compute the $(N+1) \times (N+1)$ mass matrix M and the $(N+1) \times 1$ load vector $\mathbf{b}$, viz
$$m_i j = \int_I \varphi_j \varphi_i \, dx, \qquad b_i = \int_I f \varphi_i \, dx.$$
3. Solve the linear system of equations
$$M\xi = \mathbf{b}.$$
4. Set
$$P_h f = \sum_{j=0}^{N} \xi_j \varphi_j.$$

Below are two versions of Matlab codes for computing the mass matrix M:

```
function M = MassMatrix(p, phi0, phiN)

%-------------------------------------------------------------------
% Syntax:   M = MassMatrix(p, phi0, phiN)
% Purpose:  To compute mass matrix M of partition p of an interval
% Data:     p -    vector containing nodes in the partition
%           phi0 - if 1: include basis function at the left endpoint
%                  if 0: do not include a basis function
%           phiN - if 1: include basis function at the right endpoint
%                  if 0: do not include a basis function
%-------------------------------------------------------------------

N = length(p);    % number of rows and columns in M
M = zeros(N, N);  % initiate the matrix M

% Assemble the full matrix (including basis functions at endpoints)
```

```
for i = 1:length(p)-1
    h = p(i + 1) - p(i); % length of the current interval
    M(i, i)         = M(i, i)         + h/3;
    M(i, i + 1)     = M(i, i + 1)     + h/6;
    M(i + 1, i)     = M(i + 1, i)     + h/6;
    M(i + 1, i + 1) = M(i + 1, i + 1) + h/3;
end

% Remove unnecessary elements for basis functions not included
if ~phi0
    M = M(2:end, 2:end);
end
if ~phiN
    M = M(1:end-1, 1:end-1);
end
```

A Matlab code to compute the mass matrix M for a non-uniform mesh: Since now the mesh is not uniform (the sub-intervals have different lengths), we compute the mass matrix assembling the local mass matrix computation for each sub-interval. To this end we can easily compute the mass matrix for the *standard interval* $I_1 = [0, h]$ with the basis functions $\varphi_0 = (h - x)/h$ and $\varphi_1 = x/h$: Then,

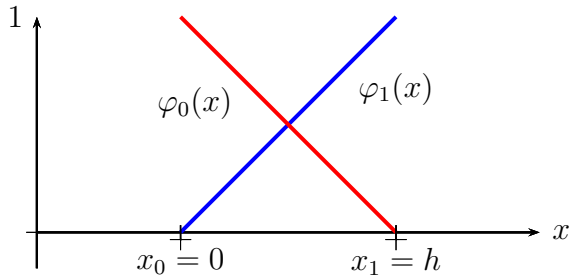


Figure B.1: Standard basis functions $\varphi_0 = (h - x)/h$ and $\varphi_1 = x/h$.

the *standard mass matrix* is given by

$$M^{I_1} = \begin{bmatrix} \int_{I_1} \varphi_0\varphi_0 & \int_{I_1} \varphi_0\varphi_1 \\ \int_{I_1} \varphi_1\varphi_0 & \int_{I_1} \varphi_1\varphi_1 \end{bmatrix}.$$

Inserting for $\varphi_0 = (h-x)/h$ and $\varphi_1 = x/h$ we compute M^{I_1} as

$$M^{I_1}\begin{bmatrix} \int_0^h (h-x)^2/h^2\,dx & \int_0^h (h-x)x/h^2\,dx \\ \int_0^h x(h-x)/h^2\,dx & \int_0^h x^2/h^2\,dx \end{bmatrix} = \frac{h}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}. \quad \text{(B.0.1)}$$

Thus, for an arbitrary sub-interval $I_k := [x_{k-1}, x_k]$ with the mesh size h_k, and basis functions φ_k and φ_{k-1} (see Fig. 3.4.), the *local mass matrix* is given by

$$M^{I_k} = \begin{bmatrix} \int_{I_k} \varphi_{k-1}\varphi_{k-1} & \int_{I_k} \varphi_{k-1}\varphi_k \\ \int_{I_k} \varphi_k\varphi_{k-1} & \int_{I_k} \varphi_{k1}\varphi_k \end{bmatrix} = \frac{h_k}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad \text{(B.0.2)}$$

where h_k is the length of the interval I_k. Note that, assembling, the diagonal elements in the *Global mass matrix* will be multiplied by 2 (see Example 4.1). These elements are corresponding to the interior nodes and are the result of adding their contribution for the intervals in their left and right. The assembling is through the following Matlab routine:

A Matlab routine to compute the load vector b:

To solve the problem of the L_2-projection, it remains to compute/assemble the load vector **b**. To this end we note that **b** depends on the unknown function f, and therefore will be computed by some of numerical integration rules (midpoint, trapezoidal, Simpson or general quadrature). Below we shall introduce Matlab routines for these numerical integration methods.

```
function b = LoadVector(f, p, phi0, phiN)

%--------------------------------------------------------------------
% Syntax:   b = LoadVector(f, p, phi0, phiN)
% Purpose:  To compute load vector b of load f over partition p
%           of an interval
% Data:     f -    right hand side funcion of one variable
%           p -    vector containing nodes in the partition
%           phi0 - if 1: include basis function at the left endpoint
%                  if 0: do not include a basis function
%           phiN - if 1: include basis function at the right endpoint
%                  if 0: do not include a basis function
%--------------------------------------------------------------------

N = length(p);     % number of rows in b
b = zeros(N, 1);   % initiate the matrix S
```

```
% Assemble the load vector (including basis functions at both endpoints)
for i = 1:length(p)-1
    h = p(i + 1) - p(i); % length of the current interval
    b(i)     = b(i)     + .5*h*f(p(i));
    b(i + 1) = b(i + 1) + .5*h*f(p(i + 1));
end

% Remove unnecessary elements for basis functions not included
if ~phi0
    b = b(2:end);
end
if ~phiN
    b = b(1:end-1);
end
```

The data function f can be either inserted as `f=@(x)` followed by some expression in the variable `x`, or more systematically through a separate routine, here called "Myfunction" as in the following example

Example B.1 (Calling a data function $f(x) = x^2$ of the load vector)**.**

```
function y= Myfunction (p)

y=x.^2


\vskip 0.3cm
Then, we assemble the corresponding load vector, viz:

\begin{verbatim}
b = LoadVector (@Myfunction, p, 1, 1)
```

Alternatively we may write

```
f=@(x)x.^2
b = LoadVector(f, p, 1, 1)
```

Now we are prepared to write a Matlab routine "My1DL2Projection" for computing the L_2-projection.

Matlab routine to compute the L_2-projection:

```
function  pf = L2Projection(p, f)

M = MassMatrix(p, 1, 1);       % assemble mass matrix
b = LoadVector(f, p, 1, 1);    % assemle load vector
pf = M\b;                      % solve linear system
plot(p, pf)                    % plot the L2-projection
```

The above routine for assembling the load vector uses the *Composite trapezoidal rule* of numerical integration. Below we gather examples of the numerical integration routines:

A Matlab routine for the composite midpoint rule

```
function M = midpoint(f,a,b,N)

h=(b-a)/N
x=a+h/2:h:b-h/2;
M=0;
for i=1:N
  M = M + f(x(i));
end
M=h*M;
```

A Matlab routine for the composite trapezoidal rule

```
function T=trapezoid(f,a,b,N)

h=(b-a)/N;
x=a:h:b;

T = f(a);
for k=2:N
    T = T + 2*f(x(k));
end
T = T + f(b);
T = T * h/2;
```

A Matlab routine for the composite Simpson's rule

```
function S = simpson(a,b,N,f)

h=(b-a)/(2*N);
x = a:h:b;
p = 0;
q = 0;

for i = 2:2:2*N     % Define the terms to be multiplied by 4
    p = p + f(x(i));
end

for i = 3:2:2*N-1   % Define the terms to be multiplied by 2
    q = q + f(x(i));
end

S = (h/3)*(f(a) + 2*q + 4*p + f(b));  % Calculate final output
```

The precomputations for standard and local stiffness and convection matrices:

$$S^{I_1} = \begin{bmatrix} \int_{I_1} \varphi_0'\varphi_0' & \int_{I_1} \varphi_0'\varphi_1' \\ \int_{I_1} \varphi_1'\varphi_0' & \int_{I_1} \varphi_1\prime\varphi_1' \end{bmatrix} = \begin{bmatrix} \int_{I_1} \frac{-1}{h}\frac{-1}{h} & \int_{I_1} \frac{-1}{h}\frac{1}{h} \\ \int_{I_1} \frac{1}{h}\frac{-1}{h} & \int_{I_1} \frac{1}{h}\frac{1}{h} \end{bmatrix} = \frac{1}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

As in the assembling of the mass-matrix, even here, for the global stiffness matrix, each interior node has contributions from both intervals that the node belongs. Consequently, assembling we have $2/h$ as the interior diagonal elements in the stiffness matrix (rather than $1/h$ in the single interval computes above). For the convection matrix C, however, because of the skew-symmetry the contributions from the *two adjacent interior intervals* will cancel out:

$$C^{I_1} = \begin{bmatrix} \int_{I_1} \varphi_0'\varphi_0 & \int_{I_1} \varphi_0\varphi_1' \\ \int_{I_1} \varphi_1\varphi_0' & \int_{I_1} \varphi_1\prime\varphi_1 \end{bmatrix} = \begin{bmatrix} \int_{I_1} \frac{-1}{h}\frac{h-x}{h} & \int_{I_1} \frac{h-x}{h}\frac{1}{h} \\ \int_{I_1} \frac{x}{h}\frac{-1}{h} & \int_{I_1} \frac{x}{h}\frac{1}{h} \end{bmatrix}$$
$$= \frac{1}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}.$$

A thorough computation of all matrix elements, for both interior and boundary nodes, in the case of continuous piece-wise linear approximation, for Mass-, stiffness- and convection-matrices are demonstrated in Examples 4.1 and 4.2.

A Matlab routine assembling the stiffness matrix:

```
function S = StiffnessMatrix(p, phi0, phiN)

%-------------------------------------------------------------------------
% Syntax:   S = StiffnessMatrix(p, phi0, phiN)
% Purpose:  To compute the stiffness matrix S of a partition p of an
%           interval
% Data:     p -    vector containing nodes in the partition
%           phi0 - if 1: include basis function at the left endpoint
%                  if 0: do not include a basis function
%           phiN - if 1: include basis function at the right endpoint
%                  if 0: do not include a basis function
%-------------------------------------------------------------------------

N = length(p);    % number of rows and columns in S
S = zeros(N, N);  % initiate the matrix S

% Assemble the full matrix (including basis functions at endpoints)
for i = 1:length(p)-1
    h = p(i + 1) - p(i); % length of the current interval
    S(i, i)         = S(i, i)         + 1/h;
    S(i, i + 1)     = S(i, i + 1)     - 1/h;
    S(i + 1, i)     = S(i + 1, i)     - 1/h;
    S(i + 1, i + 1) = S(i + 1, i + 1) + 1/h;
end

% Remove unnecessary elements for basis functions not included
if ~phi0
    S = S(2:end, 2:end);
end
if ~phiN
    S = S(1:end-1, 1:end-1);
end
```

A Matlab routine to assemble the convection matrix:

```
function C = ConvectionMatrix(p, phi0, phiN)

%-------------------------------------------------------------------------
% Syntax:   C = ConvectionMatrix(p, phi0, phiN)
% Purpose:  To compute the convection matrix C of a partition p of an
%           interval
% Data:     p -    vector containing nodes in the partition
%           phi0 - if 1: include a basis function at the left endpoint
%                  if 0: do not include a basis function
%           phiN - if 1: include a basis function at the right endpoint
%                  if 0: do not include a basis function
%-------------------------------------------------------------------------

N = length(p);    % number of rows and columns in C
C = zeros(N, N);  % initiate the matrix C

% Assemble the full matrix (including basis functions at both endpoints)
for i = 1:length(p)-1
    C(i, i)         = C(i, i)         - 1/2;
    C(i, i + 1)     = C(i, i + 1)     + 1/2;
    C(i + 1, i)     = C(i + 1, i)     - 1/2;
    C(i + 1, i + 1) = C(i + 1, i + 1) + 1/2;
end

% Remove unnecessary elementC for basis functions not included
if ~phi0
    C = C(2:end, 2:end);
end
if ~phiN
    C = C(1:end-1, 1:end-1);
end
```

Finally, below we gather the Matlab routines for finite difference approximations (also cG(1) and dG(0)) for the time discretizations.

Matlab routine for Forward-, Backward-Euler and Crank-Nicolson:

```
function [] = three_methods(u0, T, dt, a, f,  exactexists, u)

% Solves the equation du/dt + a(t)*u = f(t)
% u0: initial value;  T: final time;  dt: time step size
% exactexists = 1 <=> exact solution is known
% exactexists = 0 <=> exact solution is unknown

timevector = [0];      % we build up a vector of
                       % the discrete time levels

U_explicit_E = [u0];   % vector which will contain the
                       % solution obtained using "Forward Euler"

U_implicit_E = [u0];   % vector which will contain the
                       % solution with "Backward Euler"

U_CN = [u0];           % vector which will contain the
                       % solution using "Crank-Nicolson"

n = 1;                 % current time interval

t_l = 0;               % left end point of the current
                       % time interval, i.e. t_{n-1}

while t_l < T

  t_r = n*dt;          % right end point of the current
                       % time interval, i.e. t_{n}

  % Forward Euler:
  U_v = U_explicit_E(n);                % U_v = U_{n-1}
  U_h = (1-dt*a(t_l))*U_v+dt*f(t_l);    % U_h = U_{n};
  U_explicit_E(n+1) = U_h;

  % Backward Euler:
  U_v = U_implicit_E(n);                      % U_v = U_{n-1}
```

```
  U_h = (U_v + dt*f(t_r))/(1 + dt*a(t_r));   % U_h = U_{n}
  U_implicit_E(n+1) = U_h;

  % Crank-Nicolson:
  U_v = U_CN(n);    % U_v = U_{n-1}
  U_h = ((1 - dt/2*a(t_l))*U_v + dt/2*(f(t_l)+f(t_r))) ...
          / (1 + dt/2*a(t_r));    % U_h = U_{n}
  U_CN(n+1) = U_h;


  timevector(n+1) = t_r;
  t_l = t_r;  % right end-point in the current time interval
              % becomes the left end-point in the next time interval.

  n = n + 1;

end

% plot (real part (in case the solutions become complex))

figure(1)

plot(timevector, real(U_explicit_E), ':')
hold on
plot(timevector, real(U_implicit_E), '--')
plot(timevector, real(U_CN), '-.')

if (exactexists)
   % if known,  plot also the exact solution
  u_exact = u(timevector);
  plot(timevector, real(u_exact), 'g')
end

xlabel('t')
legend('Explicit Euler', 'Implicit Euler', 'Crank-Nicolson', 0)
hold off


if (exactexists)
```

```
  % if the exact solution is known, then plot the error:
  figure(2)

  plot(timevector, real(u_exact - U_explicit_E), ':')
  hold on
  plot(timevector, real(u_exact - U_implicit_E), '--')
  plot(timevector, real(u_exact - U_CN), '-.')
  legend('Explicit Euler', 'Implicit Euler', 'Crank-Nicolson', 0)
  title('Error')
  xlabel('t')
  hold off

end


return
```

Example B.2. *Solving* $u'(t) + u(t) = 0$ *with* `three_methods`

```
a= @(t) 1;
f= @(t) 0;
u= @(t) exp(-t)
u_0=1;
T= 1;
dt=0.01;
three_methods (u_0, T, dt, a, f, 1, u)
```

Table of Symbols

Symbol	reads	Definition
$\forall$	for all, for every	$\forall x, \quad \cos^2 x + \sin^2 x = 1$
$\exists$	There exists	see below
:	such that	$\exists x : \ x > 3$
$\vee$	or	$x \vee y \quad (x \text{ or } y)$
$\wedge$ or &	and	$x \wedge y \quad (x \text{ and } y)$ also $\quad x \ \& \ y$
$\in$	belongs	$\sqrt{2} \in \mathbb{R}$ ($\sqrt{2}$ is a real numbers $\mathbb{R}$)
$\notin$	not belongs	$\sqrt{2} \notin \mathbb{Q}$ ($\sqrt{2}$ is not a rational number)
$\perp$	orthogonal to	$u \perp v$ (u and v are orthogonal)
:=	defines as	$I := \int_a^b f(x)\,dx$ (I defines as integralen in RHS)
=:	defines	$\int_a^b f(x)\,dx =: I$ (The integral in LHS defines I)
$\approx$	approximates	$A \approx B$ (A approximates B) or A is approximately equal B.
$\Longrightarrow$	implies	$A \Longrightarrow B$ (A implies B.)
$\Longleftrightarrow$	equivalent	$A \Longleftrightarrow B$ (A is equivalent to B.)
ODE		Ordinary Differential Equation
PDE		Partial Differential Equation
IVP		Initial Value Problem
BVP		Boundary Value Problem
VF		Variational Formulation
MP		Minimization Problem
$\mathcal{P}^q(I)$	$p \in \mathcal{P}^q(I)$	$p(x)$ is a polynomial of degree $\leq q$ for $x \in I$.
$H^1(I)$	$v \in H^1(I)$	if $\int_a^b \left(v(x)^2 + v'(x)^2\right) dx < \infty, \quad I = [a, b]$.
$V_h(I)$	$v \in V_h(I)$	the space of piecewise linear functions on a partition of I.
$V_h^0(I)$	$v \in V_h^0(I)$	$v \in V_h(I)$ and v is 0 at both or one of the boundary points.

Symbol	reads	Exempel/Definition
$\|\|f\|\|_p$, $\|\|f\|\|_{L_p(I)}$	L_p-norm of f on I	$\|\|f\|\|_p := \begin{cases} \left(\int_I \|f(x)\|^p\,dx\right)^{1/p}, & 1 \le p < \infty \\ \max_{x\in I}\|f(x)\|, & p = \infty \end{cases}$
$L_p(I)$	L_p-space	$f \in L_p(I)$ iff $\|\|f\|\|_p < \infty$
$\|\|v\|\|_a$	weighted L_2-norm	$\|\|v\|\|_a := \left(\int_I a(x)\|v(x)\|^2\,dx\right)^{1/2}, \quad a(x) > 0$
$\|\|v\|\|_E$	the energy norm	$\|\|v\|\|_E := \left(\int_I a(x)\|v(x)'\|^2\,dx\right)^{1/2}, \quad \|\|v\|\|_E = \|\|v'\|\|_a.$
$\prod$	product	$\prod_{i=1}^{N} i = 1\cdot 2\cdot 3\cdot \ldots \cdot N =: N!$
$\sum$	sum	$\sum_{i=1}^{N} i = 1+2+3+\ldots+N =: N(N+1)/2$
(u,v) or $\langle u,v\rangle$	skalar/inner product	$(u,v) := u_1v_1 + u_2v_2 + \ldots + u_Nv_N, \quad u,v \in \mathbb{R}^N$ $(u,v) := \int_I u(x)v(x)\,dx$ for $u,v \in L_2(I)$.
P_hf	L_2-projection	$(f,w) = (P_hf,w), \qquad \forall w \in \mathcal{P}^q(a,b).$
$\mathcal{T}_h(I)$	a partition of I	$\mathcal{T}_h[a,b] : a = x_0 < x_1 < \ldots < x_N = b.$
π_hf	interpolant of f	$\pi_hf(x_i) = f(x_i)$ in a partition $\mathcal{T}_h$ of $I = [a,b]$.
FDM		Finite Difference Method
FE	Forward Euler	Forward Euler FDM
BE	Backward Euler	Backward Euler FDM $\Longleftrightarrow dG(0)$
CN	Crank-Nicolson	Crank-Nicolson FDM $\Longleftrightarrow cG(1)$
FEM		Finite Element Method/Galerkin Method
$cG(1)$	continuous Galerkin	continuous, piecewise linear Galerkin approx
$dG(0)$	discont. Galerkin	discontinuous, piecewise constant Galerkin
$cG(1)cG(1)$	continuous Galerkin	space time continuous, piecewise linear Galerkin
C_i		Interpolation Constant
C_s		Stability constant
TOL		Error TOLerance

Index

A
Adaptivity 64,

B
boundary condition 3, 5, 7, 55, 74, 75, 97, 107, 108
  Dirichlet 21, 53, 58, 59, 60, 61, 96, 100
  Neumann 21, 96
boundary value problem 3, 6, 8, 13, 28, 29, 53, 56, 58, 59, 61, 64, 65, 72-7, 78, 81, 95, 96, 99, 106
Two point bvp 53, 72-75

C
Cauchy-Schwarz 17, 61, 63, 97-99
Conservation of energy 106, 107, 111,
Convection 2, 64, 71, 72, 95, 120, 121, 127-129
Convection-diffusion 2, 64, 70, 107, 113, 114

Convection matrix 70, 127, 129
Crank-Nicolson 84-86, 91, 102, 121, 130-132

D
differential equation V, 1-8, 41, 92, 93, 98, 107,
ordinary differential equation 1, 9, 81,
partial differential equation 1-3, 50, 5, 7
Diffusion 2, 64, 70, 95, 107, 113, 114

E
Error estimates 7, 33, 59, 95
  a priori error estimates 79, 93
  Interpolation error 35, 51,

F
Finite dimensional spaces 21, 26, 65,
Finite Element Method V, 7, 9, 10, 19, 23, 58, 74, 75, 78, 86, 100, 109, 113
  Continuous Galerkin 58, 64, 87, 90, 93, 104,
  disontinuous Galerkin 87, 90, 104,

G
Galerkin Method BVP 21,
Gauss quadrature 48

H
hat function 13, 14, 30, 36, 55, 59, 66, 70, 101,

I
Interpolation
  Lagrange interpolation 37-39, 43,
  linear interpolation 31, 33, 36,
  polynomial interpolation 11,

Initial Boundary Value Problem (IBVP) 3, 81, 95, 96, 99, 106,
Initial value problem 2, 3, 9, 17, 81, (IBVP) 85-87, 92, 93

J, K

L
Lagrange basis 34,38-41, 50,
linear space 10, 15,
L_2-projection 20, 21, 23, 121, 122, 124-126

M
Mass Matrix 67, 69, 102-104, 122-124, 126, 127
Minimization problem 53, 56, 57, 73,
Mixed bvp 96,

N
Neumann problem/data 21, 96,
Numerical integration 7, 31, 41, 121, 124, 126,
Composite midpoint 45
Composite trapezoidal 45
Composite Simpson's 46
Simple midpoint 41, 45, 47
Simple trapezoidal 42, 45, 47, 85
Simple Simpson's 43, 46, 48

Norm 16
L_2-norm 16, 17, 63, 72, 97, 104, 105, 113, 114
L_p-norm 33, 35
vector norm 33
maximum norm 33, 35

energy norm 59, 60, 62, 75, 79
O
Ordinary Differential Equations (ODE) 1, 7, 9, 28, 81 , 92
Orthogonality 16, 29, 30, 59, 60

P
Partial Differential Equations (PDE) 1-3, 7, 10, 53, 65, 70, 98, 107,
heat equation 2, 21, 64, 95, 99, 100, 105, 106, 108,
wave equation 2, 3, 81, 95, 106, 107, 111, 112
partition 10, 12-14, 20, 21, 26, 36, 37, 39, 41, 44, 51, 58, 61, 64, 65, 70, 73-75, 77-79, 83-85.
Poincare inequality 97-100, 105

Q

R
Residual 18, 29, 30, 59, 61-64, 93

S
Scalar initial value problem, 81
Scalar product 16, 17, 32, 33, 39
stability 81, 82, 91, 93, 95-97, 99, 105, 115
Stiffness matrix, 23-26, 28, 67, 71, 75, 92, 103, 127, 128

T
test function 17-19, 21, 22, 54, 55, 58, 65, 70, 73, 87, 117
trial function 18, 21, 65, 70, 87,

U

V/W
Variational formulation 17, 22, 53.

XYZ