Computational Differential Equations

K. Eriksson, D. Estep, P. Hansbo and C. Johnson

May 26, 1997

To Anita, Floris, Ingrid, and Patty.

Contents

Preface **xi**

Part I: Introduction **1**

1: The Vision of Leibniz **3**

2: A Brief History **11**
2.1. A timeline for calculus 12
2.2. A timeline for the computer 15
2.3. An important mathematical controversy 17
2.4. Some important personalities 19

3: A Review of Calculus **25**
3.1. Real numbers. Sequences of real numbers 26
3.2. Functions 29
3.3. Differential equations 38
3.4. The Fundamental Theorem of Calculus 41
3.5. A look ahead 58

4: A Short Review of Linear Algebra **61**
4.1. Linear combinations, linear independency, and basis. . . . 62
4.2. Norms, inner products, and orthogonality. 63
4.3. Linear transformations and matrices 67
4.4. Eigenvalues and eigenvectors 68
4.5. Norms of matrices 70
4.6. Vector spaces of functions 71

5: Polynomial Approximation **75**
5.1. Vector spaces of polynomials 75
5.2. Polynomial interpolation 78

5.3. Vector spaces of piecewise polynomials 87
5.4. Interpolation by piecewise polynomials 90
5.5. Quadrature 95
5.6. The L_2 projection into a space of polynomials 100
5.7. Approximation by trigonometric polynomials 102

6: Galerkin's Method 104
6.1. Galerkin's method with global polynomials 105
6.2. Galerkin's method with piecewise polynomials 113
6.3. Galerkin's method with trigonometric polynomials 122
6.4. Comments on Galerkin's method 126
6.5. Important issues 127

7: Solving Linear Algebraic Systems 129
7.1. Stationary systems of masses and springs 131
7.2. Direct methods 136
7.3. Direct methods for special systems 146
7.4. Iterative methods 151
7.5. Estimating the error of the solution 166

Part II: The archetypes 171

8: Two-Point Boundary Value Problems 173
8.1. The finite element method 178
8.2. Error estimates and adaptive error control 189
8.3. Data and modeling errors 197
8.4. Higher order finite element methods 199
8.5. The elastic beam 202
8.6. A comment on a priori and a posteriori analysis 204

9: Scalar Initial Value Problems 205
9.1. A model in population dynamics 207
9.2. Galerkin finite element methods 209
9.3. A posteriori error analysis and adaptive error control . . . 217
9.4. A priori error analysis 228
9.5. Quadrature errors 234
9.6. The existence of solutions 237

10: Initial Value Problems for Systems **241**
10.1. Model problems 242
10.2. The existence of solutions and Duhamel's formula 249
10.3. Solutions of autonomous problems 251
10.4. Solutions of non-autonomous problems 254
10.5. Stability 255
10.6. Galerkin finite element methods 261
10.7. Error analysis and adaptive error control 263
10.8. The motion of a satellite 271

11: Calculus of Variations **275**
11.1. Minimization problems 276
11.2. Hamilton's principle 280

12: Computational Mathematical Modeling **284**

Part III: Problems in several dimensions **291**

13: Calculus of Several Variables **293**
13.1. Functions 293
13.2. Partial derivatives 295
13.3. Integration 303
13.4. Representation by potentials 312
13.5. A model of heat conduction 313
13.6. Calculus of variations in several dimensions 317
13.7. Mechanical models 318

14: Piecewise Polynomials in Several Dimensions **320**
14.1. Meshes in several dimensions 321
14.2. Vector spaces of piecewise polynomials 324
14.3. Error estimates for piecewise linear interpolation 334
14.4. Quadrature in several dimensions 339

15: The Poisson Equation **343**
15.1. The finite element method for the Poisson equation 354
15.2. Energy norm error estimates 370
15.3. Adaptive error control 374
15.4. Dealing with different boundary conditions 377

15.5. Error estimates in the L_2 norm 386

16: The Heat Equation 395
16.1. Maxwell's equations 395
16.2. The basic structure of solutions of the heat equation . . . 397
16.3. Stability 404
16.4. A finite element method for the heat equation 406
16.5. Error estimates and adaptive error control 411
16.6. Proofs of the error estimates 414

17: The Wave Equation 422
17.1. Transport in one dimension 423
17.2. The wave equation in one dimension 426
17.3. The wave equation in higher dimensions 434
17.4. A finite element method 439
17.5. Error estimates and adaptive error control 443

18: Stationary Convection-Diffusion Problems 452
18.1. A basic model 453
18.2. The stationary convection-diffusion problem 455
18.3. The streamline diffusion method 460
18.4. A framework for an error analysis 464
18.5. A posteriori error analysis in one dimension 468
18.6. Error analysis in two dimensions 471
18.7. 79 A.D. 474

19: Time Dependent Convection-Diffusion Problems 476
19.1. Euler and Lagrange coordinates 478
19.2. The characteristic Galerkin method 482
19.3. The streamline diffusion method on an Euler mesh 487
19.4. Error analysis 491

20: The Eigenvalue Problem for an Elliptic Operator 497
20.1. Computation of the smallest eigenvalue 501
20.2. On computing larger eigenvalues 502
20.3. The Schrödinger equation for the hydrogen atom 505
20.4. The special functions of mathematical physics 508

21: The Power of Abstraction **510**
21.1. The abstract formulation 511
21.2. The Lax-Milgram theorem 513
21.3. The abstract Galerkin method 515
21.4. Applications 516
21.5. A strong stability estimate for Poisson's equation 525

Part I

Introduction

This first part has two main purposes. The first is to review some mathematical prerequisites needed for the numerical solution of differential equations, including material from calculus, linear algebra, numerical linear algebra, and approximation of functions by (piecewise) polynomials. The second purpose is to introduce the basic issues in the numerical solution of differential equations by discussing some concrete examples. We start by proving the Fundamental Theorem of Calculus by proving the convergence of a numerical method for computing an integral. We then introduce Galerkin's method for the numerical solution of differential equations in the context of two basic model problems from population dynamics and stationary heat conduction.

6

Galerkin's Method

It is necessary to solve differential equations. (Newton)

Ideally, I'd like to be the eternal novice, for then only, the surprises would be endless. (Keith Jarret)

In Chapters 3 and 5, we discussed the numerical solution of the simple initial value problem $u'(x) = f(x)$ for $a < x \leq b$ and $u(a) = u_0$, using piecewise polynomial approximation. In this chapter, we introduce *Galerkin's method* for solving a general differential equation, which is based on seeking an (approximate) solution in a (finite-dimensional) space spanned by a set of basis functions which are easy to differentiate and integrate, together with an orthogonality condition determining the coefficients or coordinates in the given basis. With a finite number of basis functions, Galerkin's method leads to a system of equations with finitely many unknowns which may be solved using a computer, and which produces an approximate solution. Increasing the number of basis functions improves the approximation so that in the limit the exact solution may be expressed as an infinite series. In this book, we normally use Galerkin's method in the computational form with a finite number of basis functions. The basis functions may be global polynomials, piecewise polynomials, trigonometric polynomials or other functions. The finite element method in basic form is Galerkin's method with piecewise polynomial approximation. In this chapter, we apply Galerkin's method to two examples with a variety of basis functions. The first example is an initial value problem that models population growth and we use a global polynomial approximation. The second example is a boundary

value problem that models the flow of heat in a wire and we use piecewise polynomial approximation, more precisely piecewise linear approximation. This is a classic example of the *finite element method*. For the second example, we also discuss the *spectral method* which is Galerkin's method with trigonometric polynomials.

The idea of seeking a solution of a differential equation as a linear combination of simpler basis functions, is old. Newton and Lagrange used power series with global polynomials and Fourier and Riemann used Fourier series based on trigonometric polynomials. These approaches work for certain differential equations posed on domains with simple geometry and may give valuable qualitative information, but cannot be used for most of the problems arising in applications. The finite element method based on piecewise polynomials opens the possibility of solving general differential equations in general geometry using a computer. For some problems, combinations of trigonometric and piecewise polynomials may be used.

6.1. Galerkin's method with global polynomials

6.1.1. A population model

In the simplest model for the growth of a population, like the population of rabbits in West Virginia, the rate of growth of the population is proportional to the population itself. In this model we ignore the effects of predators, overcrowding, and migration, for example, which might be okay for a short time provided the population of rabbits is relatively small in the beginning. We assume that the time unit is chosen so that the model is valid on the time interval $[0, 1]$. We will consider more realistic models valid for longer intervals later in the book. If $u(t)$ denotes the population at time t then the differential equation expressing the simple model is $\dot{u}(t) = \lambda u(t)$, where λ is a positive real constant and $\dot{u} = du/dt$. This equation is usually posed together with an initial condition $u(0) = u_0$ at time zero, in the form of an initial value problem:

$$\begin{cases} \dot{u}(t) = \lambda u(t) & \text{for } 0 < t \leq 1, \\ u(0) = u_0. \end{cases} \tag{6.1}$$

The solution of (6.1), $u(t) = u_0 \exp(\lambda t)$, is a smooth increasing function when $\lambda > 0$.

6.1.2. Galerkin's method

We now show how to compute a polynomial approximation U of u in the set of polynomials $V^{(q)} = \mathcal{P}^q(0,1)$ on $[0,1]$ of degree at most q using Galerkin's method. We know there are good approximations of the solution u in this set, for example the Taylor polynomial and interpolating polynomials, but these require knowledge of u or derivatives of u at certain points in $[0,1]$. The goal here is to compute a polynomial approximation of u using only the information that u solves a specified differential equation and has a specified value at one point. We shall see that this is precisely what Galerkin's method achieves. Since we already know the analytic solution in this model case, we can use this knowledge to evaluate the accuracy of the approximations.

Because $\{t^j\}_{j=0}^q$ is a basis for $V^{(q)}$, we can write $U(t) = \sum_{j=0}^q \xi_j t^j$ where the coefficients $\xi_j \in \mathbb{R}$ are to be determined. It is natural to require that $U(0) = u_0$, that is $\xi_0 = u_0$, so we may write

$$U(t) = u_0 + \sum_{j=1}^{q} \xi_j t^j,$$

where the "unknown part" of U, namely $\sum_{j=1}^q \xi_j t^j$, is in the subspace $V_0^{(q)}$ of $V^{(q)}$ consisting of the functions in $V^{(q)}$ that are zero at $t = 0$, i.e. in $V_0^{(q)} = \{v : v \in V^{(q)}, v(0) = 0\}$.

Problem 6.1. Prove that $V_0^{(q)}$ is a subspace of $V^{(q)}$.

We determine the coefficients by requiring U to satisfy the differential equation in (6.1) in a suitable "average" sense. Of course U can't satisfy the differential equation at every point because the exact solution is not a polynomial. In Chapter 4, we gave a concrete meaning to the notion that a function be zero on average by requiring the function to be orthogonal to a chosen subspace of functions. The Galerkin method is based on this idea. We define the *residual error* of a function v for the equation (6.1) by

$$R(v(t)) = \dot{v}(t) - \lambda v(t).$$

The residual error $R(v(t))$ is a function of t once v is specified. $R(v(t))$ measures how well v satisfies the differential equation at time t. If the residual is identically zero, that is $R(v(t)) \equiv 0$ for all $0 \le t \le 1$, then the equation is satisfied and v is the solution. Since the exact solution u is not a polynomial, the residual error of a function in $V^{(q)}$ that satisfies the initial condition is never identically zero, though it can be zero at distinct points.

The Galerkin approximation U is the function in $V^{(q)}$ satisfying $U(0) = u_0$ such that its residual error $R(U(t))$ is orthogonal to all functions in $V_0^{(q)}$, i.e.,

$$\int_0^1 R(U(t))v(t)\,dt = \int_0^1 (\dot{U}(t) - \lambda U(t))v(t)\,dt = 0 \quad \text{for all } v \in V_0^{(q)}. \tag{6.2}$$

This is the *Galerkin orthogonality* property of U, or rather of the residual $R(U(t))$. Since the coefficient of U with respect to the basis function 1 for $V^{(q)}$ is already known ($\xi_0 = u_0$), we require (6.2) to hold only for functions v in $V_0^{(q)}$. By way of comparison, note that the true solution satisfies a stronger orthogonality condition, namely

$$\int_0^1 (\dot{u} - \lambda u)v\,dt = 0 \quad \text{for } \textit{all} \text{ functions } v. \tag{6.3}$$

We refer to the set of functions where we seek the Galerkin solution U, in this case the space $V^{(q)}$ of polynomials w satisfying $w(0) = u_0$, as the *trial space* and the space of the functions used for the orthogonality condition, which is $V_0^{(q)}$, as the *test space*. In this case, the trial and test space are different because of the *non-homogeneous* initial condition $w(0) = u_0$ (assuming $u_0 \neq 0$), satisfied by the trial functions and the homogeneous boundary condition $v(0) = 0$ satisfied by the test functions $v \in V_0^{(q)}$. In general, different methods are obtained choosing the trial and test spaces in different ways.

6.1.3. The discrete system of equations

We now show that (6.2) gives an invertible system of linear algebraic equations for the coefficients of U. Substituting the expansion for U

into (6.2) gives

$$\int_0^1 \left(\sum_{j=1}^q j\xi_j t^{j-1} - \lambda u_0 - \lambda \sum_{j=1}^q \xi_j t^j \right) v(t)\, dt = 0 \quad \text{for all } v \in V_0^{(q)}.$$

It suffices to insure that this equation holds for every basis function for $V_0^{(q)}$, yielding the set of equations:

$$\sum_{j=1}^q j\xi_j \int_0^1 t^{j+i-1}\, dt - \lambda \sum_{j=1}^q \xi_j \int_0^1 t^{j+i}\, dt = \lambda u_0 \int_0^1 t^i\, dt, \quad i = 1, ..., q,$$

where we have moved the terms involving the initial data to the right-hand side. Computing the integrals gives

$$\sum_{j=1}^q \left(\frac{j}{j+i} - \frac{\lambda}{j+i+1} \right) \xi_j = \frac{\lambda}{i+1} u_0, \quad i = 1, ..., q. \tag{6.4}$$

This is a $q \times q$ system of equations that has a unique solution if the matrix $A = (a_{ij})$ with coefficients

$$a_{ij} = \frac{j}{j+i} - \frac{\lambda}{j+i+1}, \quad i, j = 1, ..., q,$$

is invertible. It is possible to prove that this is the case, though it is rather tedious and we skip the details. In the specific case $u_0 = \lambda = 1$ and $q = 3$, the approximation is

$$U(t) \approx 1 + 1.03448t + .38793t^2 + .301724t^3,$$

which we obtain solving a 3×3 system.

Problem 6.2. Compute the Galerkin approximation for $q = 1, 2, 3$, and 4 assuming that $u_0 = \lambda = 1$.

Plotting the solution and the approximation for $q = 3$ in Fig. 6.1, we see that the two essentially coincide.

Since we know the exact solution u in this case, it is natural to compare the accuracy of U to other approximations of u in $V^{(q)}$. In Fig. 6.2, we plot the errors of U, the third degree polynomial interpolating u at $0, 1/3, 2/3$, and 1, and the third degree Taylor polynomial

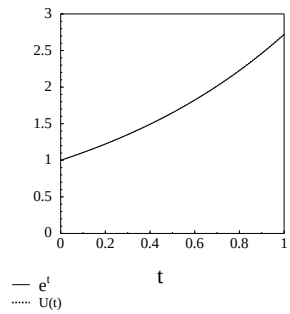

Figure 6.1: The solution of $\dot{u} = u$ and the third degree Galerkin approximation.

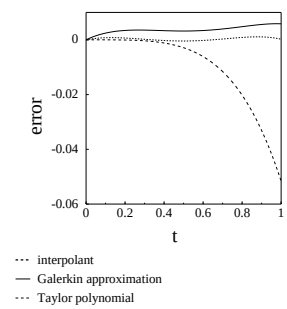

Figure 6.2: The errors of the third degree Galerkin approximation, a third degree interpolant of the solution, and the third degree Taylor polynomial of the solution.

of u computed at $t = 0$. The error of U compares favorably with the error of the interpolant of U and both of these are more accurate than the Taylor polynomial of u in the region near $t = 1$ as we would expect. We emphasize that the Galerkin approximation U attains this accuracy without any specific knowledge of the solution u except the initial data at the expense of solving a linear system of equations.

Problem 6.3. Compute the $L_2(0,1)$ projection into $\mathcal{P}^3(0,1)$ of the exact solution u and compare to U.

Problem 6.4. Determine the discrete equations if the test space is changed to $V^{(q-1)}$.

6.1.4. A surprise: ill-conditioning

Stimulated by the accuracy achieved with $q = 3$, we compute the approximation with $q = 9$. We solve the linear algebraic system in two ways: first exactly using a symbolic manipulation package and then approximately using Gaussian elimination on a computer that uses roughly 16 digits. In general, the systems that come from the discretization of a differential equation are too large to be solved exactly and we are forced to solve them numerically with Gaussian elimination for example.

We obtain the following coefficients ξ_i in the two computations:

exact coefficients

$$\begin{pmatrix} .14068... \\ .48864... \\ .71125... \\ .86937... \\ .98878... \\ 1.0827... \\ 1.1588... \\ 1.2219... \\ 1.2751... \end{pmatrix}$$

approximate coefficients

$$\begin{pmatrix} 152.72... \\ -3432.6... \\ 32163.2... \\ -157267.8... \\ 441485.8... \\ -737459.5... \\ 723830.3... \\ -385203.7... \\ 85733.4... \end{pmatrix}$$

We notice the huge difference, which makes the approximately computed U worthless. We shall now see that the difficulty is related to the fact that the system of equations (6.4) is *ill-conditioned* and this problem is exacerbated by using the standard polynomial basis $\{t^i\}_{i=0}^q$.

Problem 6.5. If access to a symbolic manipulation program and to numerical software for solving linear algebraic systems is handy, then compare the coefficients of U computed exactly and approximately for $q = 1, 2, ...$ until significant differences are found.

It is not so surprising that solving a system of equations $A\xi = b$, which is theoretically equivalent to inverting A, is sensitive to errors in the coefficients of A and b. The errors result from the fact that the computer stores only a finite number of digits of real numbers. This sensitivity is easily demonstrated in the solution of the 1×1 "system" of equations $ax = 1$ corresponding to computing the inverse $x = 1/a$ of a given real number $a \neq 0$. In Fig. 6.3, we plot the inverses of two numbers a_1 and a_2 computed from two approximations $\tilde{a}_1$ and $\tilde{a}_2$ of the same accuracy. We see that the corresponding errors in the approximations $\tilde{x} = 1/\tilde{a}_i$ of the exact values $x = 1/a_i$ vary greatly in the two cases, since

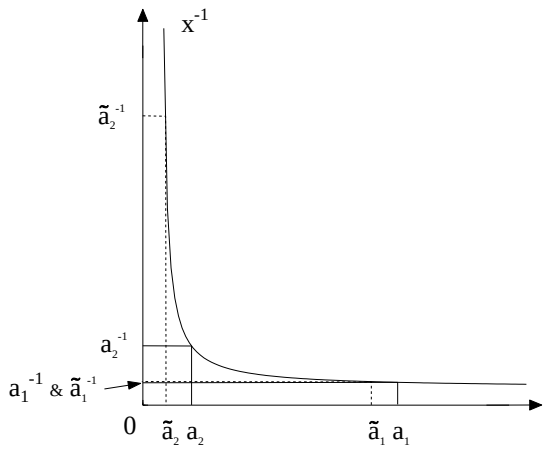

Figure 6.3: The sensitivity of the solution of $ax = 1$ to errors in a.

$1/a_i - 1/\tilde{a}_i = (\tilde{a}_i - a_i)/(a_i\tilde{a}_i)$. The closer a_i is to zero the more sensitive is the solution $1/a_i$ to errors in a_i. This expresses that computing $1/a$ is *ill-conditioned* when a is close to zero.

In general, the solution of $Ax = b$ is sensitive to errors in the entries of A when A is "close" to being non-invertible. Recall that a matrix is non-invertible if one row (or column) is a linear combination of the other rows (or columns). In the example of computing the coefficients of the Galerkin approximation with $q = 9$ above, we can see that there might be a problem if we look at the coefficient matrix A:

$$\begin{pmatrix} 0.167 & 0.417 & 0.550 & 0.633 & 0.690 & 0.732 & 0.764 & 0.789 & 0.809 \\ 0.0833 & 0.300 & 0.433 & 0.524 & 0.589 & 0.639 & 0.678 & 0.709 & 0.735 \\ 0.0500 & 0.233 & 0.357 & 0.446 & 0.514 & 0.567 & 0.609 & 0.644 & 0.673 \\ 0.0333 & 0.190 & 0.304 & 0.389 & 0.456 & 0.509 & 0.553 & 0.590 & 0.621 \\ 0.0238 & 0.161 & 0.264 & 0.344 & 0.409 & 0.462 & 0.506 & 0.544 & 0.576 \\ 0.0179 & 0.139 & 0.233 & 0.309 & 0.371 & 0.423 & 0.467 & 0.505 & 0.538 \\ 0.0139 & 0.122 & 0.209 & 0.280 & 0.340 & 0.390 & 0.433 & 0.471 & 0.504 \\ 0.0111 & 0.109 & 0.190 & 0.256 & 0.313 & 0.360 & 0.404 & 0.441 & 0.474 \\ 0.00909 & 0.0985 & 0.173 & 0.236 & 0.290 & 0.338 & 0.379 & 0.415 & 0.447 \end{pmatrix}$$

which is nearly singular since the entries in some rows and columns are quite close. On reflection, this is not surprising because the last two rows are given by $\int_0^1 R(U,t)t^8\,dt$ and $\int_0^1 R(U,t)t^9\,dt$, respectively, and t^8 and t^9 look very similar on $[0,1]$. We plot the two basis functions in Fig. 6.4.

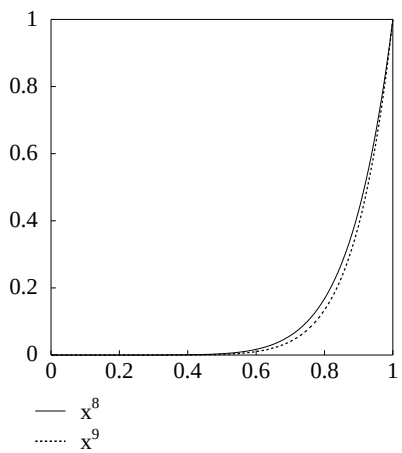

Figure 6.4: The basis functions t^8 and t^9.

In general, linear systems of algebraic equations obtained from the discretization of a differential equation tend to become ill-conditioned as the discretization is refined. This is understandable because refining the discretization and increasing the accuracy of the approximation makes it more likely that computing the residual error is influenced by the finite precision of the computer, for example. However, the degree of ill conditioning is influenced greatly by the differential equation and the choice of trial and test spaces, and even the choice of basis functions for these spaces. The standard monomial basis used above leads to an ill-conditioned system because the different monomials become very similar as the degree increases. This is related to the fact that the monomials are not an orthogonal basis. In general, the best results with respect to reducing the effects of ill-conditioning are obtained by using an orthogonal bases for the trial and test spaces. As an example, the Legendre polynomials, $\{\varphi_i(x)\}$, with $\varphi_0 \equiv 1$ and

$$\varphi_i(x) = (-1)^i \frac{\sqrt{2i+1}}{i!} \frac{d^i}{dx^i}\big(x^i(1-x)^i\big), \quad 1 \le i \le q,$$

form an orthonormal basis for $\mathcal{P}^q(0,1)$ with respect to the L_2 inner product. It becomes more complicated to formulate the discrete equations using this basis, but the effects of finite precision are greatly reduced.

Another possibility, which we take up in the second section, is to use piecewise polynomials. In this case, the basis functions are "nearly orthogonal".

Problem 6.6. (a) Show that φ_3 and φ_4 are orthogonal.

6.2. Galerkin's method with piecewise polynomials

We start by deriving the basic model of stationary heat conduction and then formulate a finite element method based on piecewise linear approximation.

6.2.1. A model for stationary heat conduction

We model heat conduction a thin heat-conducting wire occupying the interval $[0, 1]$ that is heated by a *heat source* of intensity $f(x)$, see Fig. 6.5. We are interested in the stationary distribution of the temperature $u(x)$

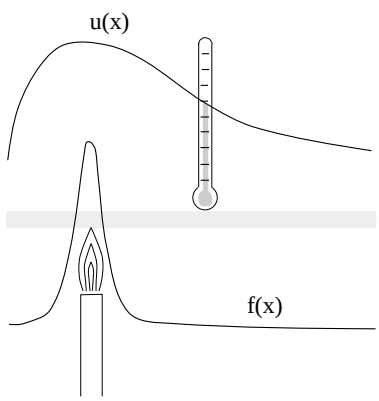

Figure 6.5: A heat conducting wire with a source $f(x)$.

in the wire. We let $q(x)$ denote the heat flux in the direction of the positive x-axis in the wire at $0 < x < 1$. Conservation of energy in a stationary case requires that the net heat flux through the endpoints of an arbitrary sub-interval (x_1, x_2) of $(0, 1)$ be equal to the heat produced in (x_1, x_2) per unit time:

$$q(x_2) - q(x_1) = \int_{x_1}^{x_2} f(x)\, dx.$$

By the Fundamental Theorem of Calculus,

$$q(x_2) - q(x_1) = \int_{x_1}^{x_2} q'(x)\,dx,$$

from which we conclude that

$$\int_{x_1}^{x_2} q'(x)\,dx = \int_{x_1}^{x_2} f(x)\,dx.$$

Since x_1 and x_2 are arbitrary, assuming that the integrands are continuous, we conclude that

$$q'(x) = f(x) \quad \text{for } 0 < x < 1, \tag{6.5}$$

which expresses conservation of energy in differential equation form. We need an additional equation that relates the heat flux q to the temperature gradient (derivative) u' called a *constitutive equation*. The simplest constitutive equation for heat flow is *Fourier's law*:

$$q(x) = -a(x)u'(x), \tag{6.6}$$

which states that heat flows from warm regions to cold regions at a rate proportional to the temperature gradient $u'(x)$. The constant of proportionality is the *coefficient of heat conductivity* $a(x)$, which we assume to be a positive function in $[0,1]$. Combining (6.5) and (6.6) gives the *stationary heat equation* in one dimension:

$$-\left(a(x)u'(x)\right)' = f(x) \quad \text{for } 0 < x < 1. \tag{6.7}$$

To define a solution u uniquely, the differential equation is complemented by *boundary conditions* imposed at the boundaries $x = 0$ and $x = 1$. A common example is the homogeneous *Dirichlet* conditions $u(0) = u(1) = 0$, corresponding to keeping the temperature zero at the endpoints of the wire. The result is a *two-point boundary value problem*:

$$\begin{cases} -(au')' = f & \text{in } (0,1), \\ u(0) = u(1) = 0. \end{cases} \tag{6.8}$$

The boundary condition $u(0) = 0$ may be replaced by $-a(0)u'(0) = q(0) = 0$, corresponding to prescribing zero heat flux, or insulating the wire, at $x = 0$. Later, we aslo consider non-homogeneous boundary conditions of the from $u(0) = u_0$ or $q(0) = g$ where u_0 and g may be different from zero.

Problem 6.7. Determine the solution u of (6.9) with $a(x) = 1$ by symbolic computation by hand in the case $f(x) = 1$ and $f(x) = x$.

We want to determine the temperature u in the wire by solving the heat equation (6.8) with given f, boundary conditions, and heat conductivity $a(x)$. To compute the solution numerically we use the Galerkin finite element method.

6.2.2. The Galerkin finite element method

We consider the problem (6.8) in the case $a \equiv 1$, that is

$$\begin{cases} -u'' = f & \text{in } (0,1), \\ u(0) = u(1) = 0, \end{cases} \tag{6.9}$$

and formulate the simplest finite element method for (6.9) based on continuous piecewise linear approximation.

We let $\mathcal{T}_h : 0 = x_0 < x_1 < ... < x_{M+1} = 1$, be a *partition* or (*triangulation*) of $I = (0,1)$ into sub-intervals $I_j = (x_{j-1}, x_j)$ of length $h_j = x_j - x_{j-1}$ and let $V_h = V_h^{(1)}$ denote the set of continuous piecewise linear functions on $\mathcal{T}_h$ that are zero at $x = 0$ and $x = 1$. We show an example of such a function in Fig. 6.6. In Chapter 4, we saw that V_h is

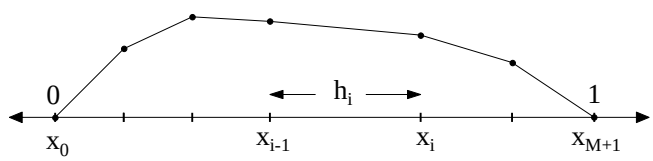

Figure 6.6: A continuous piecewise linear function in V_h.

a finite dimensional vector space of dimension M with a basis consisting of the hat functions $\{\varphi_j\}_{j=1}^M$ illustrated in Fig. 5.11. The coordinates of a function v in V_h in this basis are the values $v(x_j)$ at the interior nodes x_j, $j = 1, ..., M$, and a function $v \in V_h$ can be written

$$v(x) = \sum_{j=1}^{M} v(x_j)\varphi_j(x).$$

Note that because $v \in V_h$ is zero at 0 and 1, we do not include φ_0 and φ_{M+1} in the set of basis functions for V_h.

As in the previous example, Galerkin's method is based on stating the differential equation $-u'' = f$ in the form

$$\int_0^1 (-u'' - f)v\,dx = 0 \quad \text{for all functions} v, \tag{6.10}$$

corresponding to the residual $-u'' - f$ being orthogonal to the test functions v, cf. (6.3). However, since the functions in V_h do not have second derivatives, we can't simply plug a candidate for an approximation of u in the space V_h directly into this equation. To get around this technical difficulty, we use integration by parts to move one derivative from u'' onto v assuming v is differentiable and $v(0) = v(1) = 0$:

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

where we used the boundary conditions on v. We are thus led to the following *variational formulation* of (6.9): find the function u with $u(0) = u(1) = 0$ such that

$$\int_0^1 u'v'\,dx = \int_0^1 fv\,dx, \tag{6.11}$$

for all functions v such that $v(0) = v(1) = 0$. We refer to (6.11) as a *weak form* of (6.10).

The Galerkin finite element method for (6.9) is the following finite-dimensional analog of (6.11): find $U \in V_h$ such that

$$\int_0^1 U'v'\,dx = \int_0^1 fv\,dx \quad \text{for all } v \in V_h. \tag{6.12}$$

We note that the derivatives U' and v' of the functions U and $v \in V_h$ are piecewise constant functions of the form depicted in Fig. 6.7 and are not defined at the nodes x_i. However, the integral with integrand $U'v'$ is nevertheless uniquely defined as the sum of integrals over the subintervals. This is due to the basic fact of integration that two functions that are equal except at a finite number of points, have the same integral. We illustrate this in Fig. 6.8. By the same token, the value (or lack of

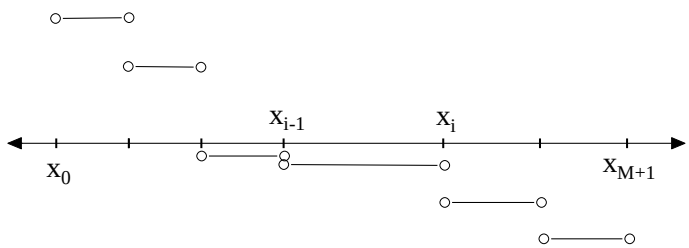

Figure 6.7: The derivative of the continuous piecewise linear function in Fig. 6.6.

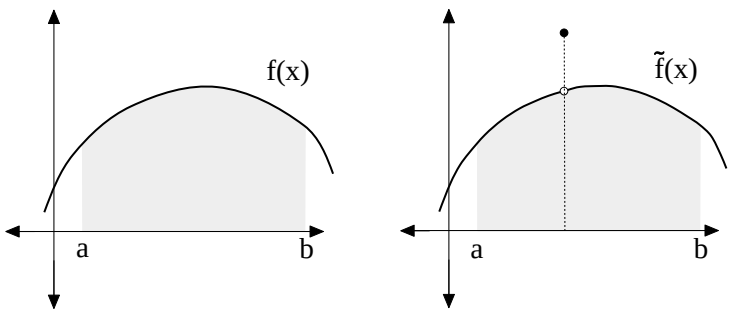

Figure 6.8: Two functions that have a different value at one point have the same area underneath their curves.

value) of U' and v' at the distinct node points x_i does not affect the value of $\int_0^1 U'v'\,dx$.

The equation (6.10) expresses the fact that the residual error $-u''-f$ of the exact solution is orthogonal to *all* test functions v and (6.11) is a reformulation in weak form. Similarly, (6.12) is a way of forcing in weak form the residual error of the finite element solution U to be orthogonal to the finite dimensional set of test functions v in V_h.

6.2.3. The discrete system of equations

Using the basis of hat functions $\{\varphi_j\}_{j=1}^M$, we have

$$U(x) = \sum_{j=1}^{M} \xi_j \varphi_j(x)$$

and determine the nodal values $xi_j = U(x_j)$ using the Galerkin orthogonality (6.12). Substituting, we get

$$\sum_{j=1}^{M} \xi_j \int_0^1 \varphi_j' v' \, dx = \int_0^1 f v \, dx, \tag{6.13}$$

for all $v \in V_h$. It suffices to check (6.13) for the basis functions $\{\varphi_i\}_{i=1}^M$, which gives the $M \times M$ linear system of equations

$$\sum_{j=1}^{M} \xi_j \int_0^1 \varphi_j' \varphi_i' \, dx = \int_0^1 f \varphi_i \, dx, \quad i = 1, ..., M, \tag{6.14}$$

for the unknown coefficients $\{\xi_j\}$. We let $\xi = (\xi_j)$ denote the vector of unknown coefficients and define the $M \times M$ *stiffness matrix* $A = (a_{ij})$ with coefficients

$$a_{ij} = \int_0^1 \varphi_j' \varphi_i' \, dx,$$

and the *load vector* $b = (b_i)$ with

$$b_i = \int_0^1 f \varphi_i \, dx.$$

These names originate from early applications of the finite element method in structural mechanics. Using this notation, (6.14) is equivalent to the linear system

$$A\xi = b. \tag{6.15}$$

In order to solve for the coefficients of U, we first have to compute the stiffness matrix A and load vector b. For the stiffness matrix, we note that a_{ij} is zero unless $i = j - 1$, $i = j$, or $i = j + 1$ because otherwise either $\varphi_i(x)$ or $\varphi_j(x)$ is zero on each sub-interval occurring in the integration. We illustrate this in Fig. 6.9. We compute a_{ii} first. Using the definition of the φ_i,

$$\varphi_i(x) = \begin{cases} (x - x_{i-1})/h_i, & x_{i-1} \le x \le x_i, \\ (x_{i+1} - x)/h_{i+1}, & x_i \le x \le x_{i+1}, \end{cases}$$

and $\varphi_i(x) = 0$ elsewhere, the integration breaks down into two integrals:

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

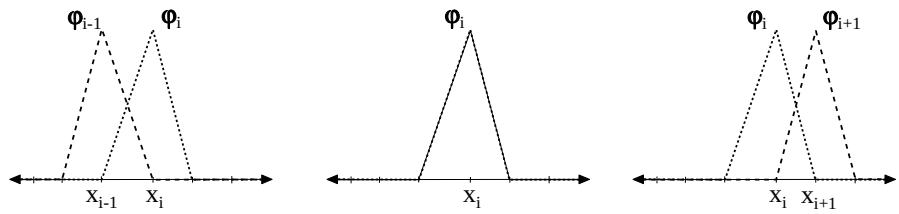

Figure 6.9: Three possibilities to obtain a non-zero coefficient in the stiffness matrix.

since $\varphi_i' = 1/h_i$ on (x_{i-1}, x_i) and $\varphi_i' = -1/h_{i+1}$ on (x_i, x_{i+1}), and φ_i is zero on the rest of the sub-intervals. Similarly,

$$a_{i\,i+1} = \int_{x_i}^{x_{i+1}} \frac{-1}{h_{i+1}} \frac{1}{h_{i+1}}\, dx = -\frac{1}{h_{i+1}}.$$

Problem 6.8. Prove that $a_{i-1\,i} = -1/h_i$ for $i = 2, 3, ..., M$.

Problem 6.9. Determine the stiffness matrix A in the case of a uniform mesh with meshsize $h_i = h$ for all i.

We compute the coefficients of b in the same way to get

$$b_i = \int_{x_{i-1}}^{x_i} f(x) \frac{x - x_{i-1}}{h_i} dx + \int_{x_i}^{x_{i+1}} f(x) \frac{x_{i+1} - x}{h_{i+1}} dx, \quad i = 1, ..., M.$$

Problem 6.10. Verify this formula.

Problem 6.11. Using a uniform mesh with $h = .25$, compute the Galerkin finite element approximation for $f(x) = x$ by hand.

The matrix A is a *sparse* matrix in the sense that most of its entries are zero. In this case, A is a *banded* matrix with non-zero entries occurring only in the diagonal, super-diagonal and sub-diagonal positions. This contrasts sharply with the coefficient matrix in the first example which is "full" in the sense that all of its entries are non-zero. The bandedness of A reflects the fact that the basis functions $\{\varphi_i\}$ for V_h are "nearly" orthogonal, unlike the basis used in the first example. Moreover, A is a *symmetric* matrix since $\int_0^1 \varphi_i'\varphi_j'\, dx = \int_0^1 \varphi_j'\varphi_i'\, dx$. Finally, it is possible to show that A is *positive-definite*, which means that

$\sum_{i,j=1}^{M} \eta_i a_{ij} \eta_j > 0$ unless all $\eta_i = 0$, which implies that A is invertible, so that (6.15) has a unique solution.

We expect to increase the accuracy of the approximate solution by increasing the dimension M. Systems of dimension $100 - 1,000$ in one dimension and up to $100,000$ in higher dimensions are common. Thus, it is important to study the solution of (6.15) in detail and we begin to do this in Chapter 7.

Problem 6.12. Prove that A is positive-definite. Hint: use that $\eta^\top A\eta = \int_0^1 (v(x)')^2\, dx$ where $v(x)$ is the function in V_h defined by $v(x) = \sum_i \eta_i \varphi_i(x)$.

6.2.4. A concrete example

As a concrete example, we consider (6.9) with

$$f(x) = 10(1 - 10(x - .5)^2)\, e^{-5(x-.5)^2}.$$

This is an exceptional case in which the solution $u(x)$ can be computed exactly:

$$u(x) = e^{-5(x-.5)^2} - e^{-5/4},$$

and we may easily compare the approximation with the exact solution.

In this example, f varies quite a bit over the interval. We plot it in Fig. 6.10. Recalling the discussions in Chapters 3 and 5, we expect the

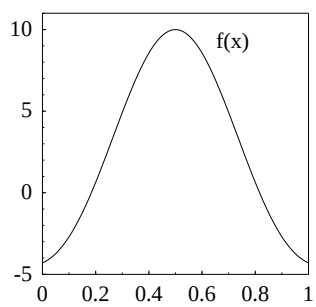

Figure 6.10: Load function f for the first example.

mesh size to vary if we want the computational method to be efficient

with a minimal number of mesh points. The following choice of the mesh points minimizes $\|u' - U'\|_{L_2(0,1)}$ while keeping $M = 16$:

$$\begin{aligned}
h_1 = h_{16} &\approx .2012 \\
h_2 = h_{15} &\approx .0673 \\
h_3 = h_{14} &\approx .0493 \\
h_4 = h_{13} &\approx .0417 \\
h_5 = h_{12} &\approx .0376 \\
h_6 = h_{11} &\approx .0353 \\
h_7 = h_{10} &\approx .0341 \\
h_8 = h_9 &\approx .0335
\end{aligned}$$

This partition is the result of applying an adaptive algorithm based on information obtained from the data f and the computed approximation U, and does not require any knowledge of the exact solution u. We will present the adaptive algorithm in Chapter 8 and explain how it is possible to get around the apparent need to know the exact solution.

In Fig. 6.11, we plot the exact solution u and the Galerkin finite element approximation U together with their derivatives. Notice that

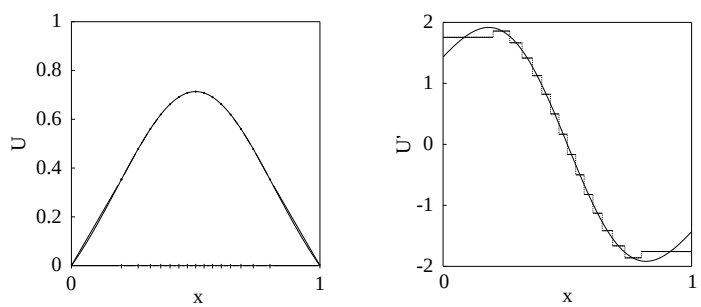

Figure 6.11: The true solution u and the finite element approximation U together with their derivatives.

the adaptive procedure has clustered mesh points towards the middle of the interval. With $M = 16$ we have $\|u' - U'\|_2 \leq .25$, while $\|u - U\|_2 \leq .01$. If we perform the computation using a mesh with uniform spacing we need to take $M = 24$ to obtain the same accuracy for the derivatives. This disparity increases as the number of elements is increased and may be much more substantial for more complex problems. We will meet many examples below.

6.3. Galerkin's method with trigonometric polynomials

We noted at the very end of Chapter 3 that for $n = 1, 2, ...$, the trigonometric function $\sin(n\pi x)$ solves the boundary value problem $u'' + n^2\pi^2 u = 0$ in $(0, 1)$, $u(0) = u(1) = 0$. In fact, the functions $\sin(n\pi x)$, $n = 1, 2, ...$, are the solutions to the *eigenvalue problem* of finding nonzero functions $\varphi(x)$ and scalars λ such that

$$\begin{cases} -\varphi'' = \lambda\varphi & \text{in } (0,1), \\ \varphi(0) = \varphi(1) = 0. \end{cases} \tag{6.16}$$

Eigenvalue problems play an important role in mathematics, mechanics and physics and we will study such problems in more detail below. This gives a different perspective on the trigonometric functions $\sin(n\pi x)$ as the eigenfunctions of the particular boundary value problem (6.16). In particular the mutual orthogonality of the functions $\sin(n\pi x)$ on $(0, 1)$ reflect a general orthogonality property of eigenfunctions. The general idea of the *spectral* Galerkin method is to seek a solution of a given boundary value problem as a linear combination of certain eigenfunctions. The given boundary value and the eigenvalue problem do not have to be directly related in the sense that the differential operators involved are the same, but the boundary conditions should match. As an example we consider the application of the spectral Galerkin method to the model (6.8) of stationary heat flow with variable heat conductivity

$$\begin{cases} -\big((1+x)u'(x)\big)' = f(x) & \text{for } 0 < x < 1, \\ u(0) = u(1) = 0, \end{cases} \tag{6.17}$$

where $f(x) = 1 + (1 + 3x - x^2)\exp(-x)$, based on the eigenfunctions $\sin(n\pi x)$, $n = 1, 2, ...$, of (6.16). We then seek to express the solution $u(x)$ of (6.17) as a sine series, cf. (5.33),

$$u(x) = \sum_{n=1}^{\infty} b_n \sin(n\pi x).$$

In the corresponding discrete analog we seek for some fixed $q > 0$ an approximate solution $U(x)$ of the form

$$U(x) = \sum_{j=1}^{q} \xi_j \sin(j\pi x), \tag{6.18}$$

using Galerkin's method to determine the coefficients ξ_j. Denoting by $V^{(q)}$ the space of trigonometric polynomials that are linear combinations of the functions $\{\sin(i\pi x)\}_{i=1}^q$, Galerkin's method reads: find $U \in V^{(q)}$ such that

$$-\int_0^1 \big((1+x)U'(x)\big)' v(x)\,dx = \int_0^1 f(x)v(x)\,dx \quad \text{for all } v \in V^{(q)}. \tag{6.19}$$

Because the functions in $V^{(q)}$ are smooth, we have no difficulty defining the residual error $\big((1+x)U'(x)\big)' - f(x)$ of the approximate solution U in $V^{(q)}$. By (6.19), the Galerkin approximation U is the function in $V^{(q)}$ whose residual error is orthogonal to all functions in $V^{(q)}$. Using integration by parts, we can rewrite this as

$$\int_0^1 (1+x)U'(x)v'(x)\,dx = \int_0^1 f(x)v(x)\,dx \quad \text{for all } x \in V^{(q)}, \tag{6.20}$$

which is the usual formulation of a Galerkin method for the boundary value problem (6.17). As before, we substitute the expansion (6.18) for U into (6.20) and choose $v = \sin(i\pi x)$, $i = 1, ..., q$. This gives a linear system of equations

$$A\xi = d$$

where $A = (a_{ij})$ and $d = (d_i)$ have coefficients

$$\begin{aligned} a_{ij} &= \pi^2 ij \int_0^1 (1+x)\cos(j\pi x)\cos(i\pi x)\,dx \\ d_i &= \int_0^1 \big(1 + (1+3x-x^2)\exp(-x)\big)\sin(i\pi x)\,dx. \end{aligned} \tag{6.21}$$

Problem 6.13. Prove that $V^{(q)}$ is a subspace of the continuous functions on $[0,1]$ that satisfy the boundary conditions and show that $\{\sin(i\pi x)\}_{i=1}^q$ is an orthogonal basis for $V^{(q)}$ with respect to the L_2 inner product.

Problem 6.14. Derive (6.20).

Problem 6.15. Verify the formulas for A and d then compute explicit formulas for the coefficients.

In this problem, we are able to compute these integrals exactly using a symbolic manipulation program, though the computations are messy. For a general problem we are usually unable to evaluate the corresponding integrals exactly. For these reasons, it is natural to also consider the use of quadrature to evaluate the integrals giving the coefficients of A and d. We examine three quadrature rules discussed in Chapter 5, the composite rectangle, midpoint, and trapezoidal rules respectively. We partition $[0,1]$ into $M+1$ intervals with equally spaced nodes $x_l = l/(M+1)$ for $l = 0, ..., M+1$. For example, the composite rectangle rule is

$$a_{ij} \approx \frac{\pi^2 ij}{M+1} \sum_{l=1}^{M+1} (1+x_l)\cos(j\pi x_l)\cos(i\pi x_l)$$

$$d_i \approx \frac{1}{M+1} \sum_{l=1}^{M+1} \big(1 + (1+3x_l - x_l^2)\exp(-x_l)\big)\sin(i\pi x_l).$$

Problem 6.16. Write out the corresponding formulas for schemes that use the midpoint and trapezoidal rules to evaluate the integrals.

From the discussion in Chapter 5, we expect the composite midpoint and trapezoidal rules to be more accurate than the composite rectangle rule on a given partition. In Fig. 6.12, we plot the results obtained by using the four indicated methods. In this case, using the lower order accurate rectangle rule affects the results significantly. We list the coefficients obtained by the four methods below.

q	U	U_r	U_m	U_t
1	0.20404	0.21489	0.20419	0.20375
2	0.017550	0.031850	0.017540	0.017571
3	0.0087476	0.013510	0.0087962	0.0086503
4	0.0022540	0.0095108	0.0022483	0.0022646
5	0.0019092	0.0048023	0.0019384	0.0018508
6	0.00066719	0.0054898	0.00066285	0.00067495
7	0.00069279	0.0026975	0.00071385	0.00065088
8	0.00027389	0.0038028	0.00026998	0.00028064
9	0.00030387	0.0014338	0.00032094	0.00027024

Several questions about the error arise in this computation. We would like to know the accuracy of the Galerkin approximation U and

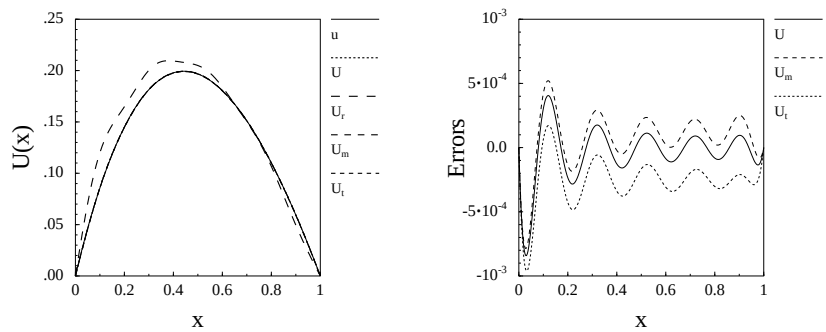

Figure 6.12: On the left, we plot four approximate solutions of (6.17) computed with $q = 9$. U was computed by evaluating the integrals in (6.21) exactly. U_r is computed using the composite rectangle rule, U_m is computed using the composite midpoint rule, and U_t is computed using the composite trapezoidal rule with $M = 20$ to evaluate the integrals respectively. On the right, we plot the errors versus x for U, U_m, and U_t.

whether the approximation converges as we increase q. Moreover, we would also like to know the effects of choosing different quadrature rules on the accuracy. Moreover, since $a = (1 + x)$ and $f(x) = 1 + (1 + 3x - x^2)e^{-x}$ vary through $[0, 1]$, it would be better to adapt the mesh used for the quadrature rules in order to achieve the desired accuracy in the coefficients (6.21). This also requires knowledge of the effects of quadrature error on the accuracy of the approximation.

Problem 6.17. Formulate a Galerkin approximation method using trigonometric polynomials for (6.9). Note in this case that the linear system giving the coefficients of the approximation is trivial to solve because it is diagonal. Why didn't this happen when we discretized (6.17)? Interpret the formula for the coefficients of the data vector in terms of Fourier series.

Problem 6.18. Use the spectral method to discretize $-u'' + u' = f$ in $(0, 1)$ with $u(0) = u(1) = 0$.

6.4. Comments on Galerkin's method

We have considered Galerkin's method for three kinds of approximations: the q-method (6.2) with global polynomials of order q (this method is also often referred to as the p-method), the h-method (6.12) with continuous piecewise linear polynomials on a partition $\mathcal{T}_h$ of mesh size h, and the spectral method (6.20) with global trigonometric polynomials. In rough terms, we can distinguish different Galerkin approximations by the following properties of the basis functions:

- local or global support
- (near) orthogonality or non-orthogonality.

The *support* of a function is the set of points where the function is different from zero. A function is said to have *local support* if it is zero outside a set of small size in comparison to the domain, while a function with *global support* is non-zero throughout the domain except at a few points. The h-method uses basis functions with local support, while the basis functions of the q-method in Section 6.1 or the spectral method are polynomials or trigonometric polynomials with global support. The basis functions used for the spectral method are mutually orthogonal, while the basis functions used in Section 6.1 for the global polynomials was not. Using orthogonal basis functions tends to reduce the effects of rounding errors that occur when computing the approximation. For example, in the spectral method in Section 6.3, the matrix determining the approximation is sparse with roughly half the coefficients equal to zero. The basis functions used in the h-method in Section 6.2 are not orthogonal, but because they have local support, they are "nearly" orthogonal and again the matrix determining the approximation is sparse. Recent years have seen the development of *wavelet methods*, which are Galerkin methods with basis functions that combine L_2 orthogonality with small support. Wavelets are finding increasing applications in image processing, for example.

In Galerkin's method, we expect to increase accuracy by increasing the dimension of the trial and test spaces used to define the approximation. In the h-method this is realized by decreasing the mesh size and in the $q - method$ by increasing the degree of the polynomials. More generally, we can use a combination of decreasing the mesh size and

increasing the polynomial degree, which is called the (h, q)-method. Finally, in the spectral Galerkin method, we try to improve accuracy by increasing the degree of the trigonometric polynomial approximation.

Galerkin was born in 1871 in Russia and was educated at the St. Petersburg Technological Institute and in the Kharkhov locomotive works. He began doing research in engineering while he was in prison in 1906-7 for his participation in the anti-Tsarist revolutionary movement. He later made a distinguished academic career and had a strong influence on engineering in the Soviet Union. This was due in no small part to the success of Galerkin's method, first introduced in a paper on elasticity published in 1915. From its inception, which was long before the computer age, Galerkin's method was used to obtain high accuracy with minimal computational effort using a few cleverly chosen basis functions. Galerkin's method belongs to the long tradition of variational methods in mathematics, mechanics and physics going back to the work by Euler and Lagrange, and including important work by e.g. Hamilton, Rayleigh, Ritz and Hilbert. The finite element method, first developed in aerospace engineering in the 1950s, may be viewed as the computer-age realization of this variational methodology. In the advanced companion volume we give a more detailed account of the development of the finite element method into the first general technique for solving differential equations numerically, starting with applications to elasticity problems in the 1960s, to the diffusion problems in the 1970s and to fluid flow and wave propagation in the 1980s.

6.5. Important issues

The examples we have presented raise the following questions concerning a finite element approximation U of the solution u of a given differential equation:

- How big is the *discretization error* $u - U$ or $u' - U'$ in some norm?
- How should we choose the mesh to control the discretization error $u - U$ or $u' - U'$ to a given tolerance in some norm using the least amount of work (computer time)?
- How can we compute the discrete solution U efficiently and what is the effect of errors made in computing U?

This book seeks to answer these questions for a variety of problems.

> Between my finger and my thumb
> The squat pen rests
> I'll dig with it. (Heaney)

> The *calculus ratiocinator* of Leibniz merely needs to have an engine put into it to become a *machina ratiocinatrix*. The first step in this direction is to proceed from the calculus to a system of *ideal* reasoning machines, and this was taken several years ago by Turing. (Wiener)

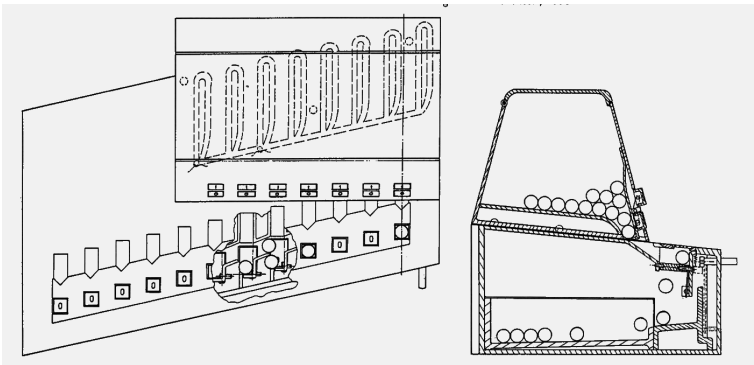

Figure 6.13: Proposal by Leibniz for a "ball-computer" for the multiplication of binary numbers.

Part II

The archetypes

In this part, we explore some basic aspects of Galerkin's method by applying it in three kinds of problems: scalar linear two-point boundary value problems, scalar linear initial value problems, and initial value problems for linear systems of ordinary differential equations. Using these three types of problems, we introduce the three basic types of differential equations: elliptic, parabolic, and hyperbolic problems typically modelling stationary heat conduction, nonstationary heat conduction, and wave propagation respectively. In doing so, we set up a framework that we apply to linear partial differential equations in the third part of this book and to nonlinear systems of differential equations in the companion volume.

Part III

Problems in several dimensions

In the last part of this book, we extend the scope to the real world of two and three dimensions. We start by recalling some basic results from calculus of several variables including some facts about piecewise polynomial approximation. We then consider the basic types of linear partial differential equations including the elliptic Poisson equation, the parabolic heat equation, the hyperbolic wave equation, and the mixed parabolic/elliptic-hyperbolic convection-diffusion equation. We also consider eigenvalue problems and conclude with an abstract development of the finite element method for elliptic problems.

Bibliography

[1] V. I. Arnold, *Huygens and Barrow, Newton and Hooke*, Birkhäuser, New York, 1990.

[2] K. Atkinson, *An Introduction to Numerical Analysis*, John Wiley & Sons, Inc., New York, 1989.

[3] L. Bers, F. John, and M. Schecter, *Partial Differential Equations*, John Wiley & Sons, Inc., New York, 1964.

[4] S. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[5] P. Ciarlet, *The Finite Element Method for Elliptic Problems*, North–Holland, New York, 1978.

[6] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Introduction to adaptive methods for differential equations*, Acta Numerica, (1995), pp. 105–158.

[7] G. Golub and C. V. Loan, *Matrix Computations*, Johns Hopkins University Press, Maryland, 1983.

[8] P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley & Sons, Inc., New York, 1962.

[9] E. Isaacson and H. Keller, *Analysis of Numerical Methods*, John Wiley & Sons, Inc., New York, 1966.

[10] C. Johnson, *Numerical solution of partial differential equations by the finite element method*, Studentlitteratur and Cambridge University Press, Lund, Sweden, and Cambridge, England, 1987.

[11] N. Jolley, *The Cambridge Companion to Leibniz*, Cambridge University Press, Cambridge, England, 1995.

[12] M. Kline, *Mathematical Thought from Ancient to Modern Times*, Oxford University Press, Oxford, England, 1972.

[13] G. W. Leibniz, *Nova methodus pro maximis et minimis, itemque tangentibus, que nec fractas, nec irrationales quantitates moratur, et singulare pro illis calculi genus*, Acta Eruditorum, (1684), pp. 466–473.

[14] M. Renardy and R. Rogers, *An Introduction to Partial Differential Equations*, Springer-Verlag, New York, 1993.

[15] W. Rudin, *Principles of Mathematical Analysis, 3rd ed.*, McGraw–Hill, New York, 1976.

[16] E. Stein and A. Heinekamp, *Leibniz, Mathematiker, Physiker, Techniker*, Leibniz-Gesellschaft, Hannover, Germany, 1990.

[17] G. Strang, *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, Cambridge, Mass., 1986.

[18] W. A. Strauss, *Partial Differential Equations: An Introduction*, John Wiley & Sons, Inc., New York, 1992.

Index

$(\ ,\ )$, 63, 72
$(?\ \times I)_+$, 453
$(?\ \times I)_-)$, 453
D, 34
F_n, 483
G_z, 352
H^1, 517
H^{-1}, 521
K, 321
$L(\ )$, 511
L_2 projection, 100
L_p, 72, 74
P_h, 338
R_2, 411
S_n, 406
V_h, 325
V_n, 407
$W_k^{(r)}$, 407
$[\,]$, 213
Δ_h, 358
$\mathbb{R}$, 26
β_n^h, 482
δ_z, 348
$\hat{\epsilon}$, 462
$\int$, 40
λ_i, 327
$\underline{\lim}$, 26
$\overline{W}_{kn}$, 484
φ_i, 327
π_k, 414
τ_K, 322
τ_n, 493
$\top$, 61
$a(\ ,\ )$, 511
h refinement, 323
h_K, 322
h_n, 406
l_p, 64
$n(x)$, 453
$\mathcal{N}_h$, 321
$\mathcal{S}_h$, 321
$\mathcal{T}_h$, 321
(h,p)-method, 126

a posteriori error estimate, 131, 166, 190, 193, 221, 223, 236, 264, 288, 373, 390, 394, 411, 444, 467, 469, 472
    calibration, 194
a priori error estimate, 156, 166, 189, 191, 230, 231, 264, 288, 370, 392, 393, 412, 448, 468, 474, 494, 495, 501
absorption, 452
abstract Galerkin's method, 515
accumulation of errors, 206, 208, 218, 231, 235, 269, 449
accuracy, 215
action integral, 276
adapted
    mesh, 54
adaptive
    algorithm, 55, 195, 225, 375, 413, 446
    error control, 194, 374
advancing front, 323
Ampere's law, 396
approximate characteristics, 482
arclength, 304

artificial viscosity, 461, 462
atmosphere, 248
autonomous system, 241
average, 100
average rate of change, 33

Babbage, 15
backward Euler difference scheme, 234
backward heat equation, 402
bandwidth, 147
beam problem, 202, 520
bending moment, 203
bilinear, 64
bilinear finite element method, 334
bilinear form, 512
block structure, 364
bound states, 506
boundary, 453
boundary condition, 114
  Dirichlet, 184
  essential, 187, 384
  natural, 187, 384
  Neumann, 184
  Robin, 184
boundary layer, 391, 459
boundary value problem, 114
bounded support, 478
brachistonchrone problem, 279
Brownian motion, 344

calculus of variations, 13, 275
calibration, 194
Cantor, 18
Cauchy, 18
Cauchy sequence, 27, 32
Cauchy's inequality, 72
  weighted, 190
Cauchy-Navier's elasticity equations, 522
Cauchy-Schwarz inequality, 64
Cavalieri, 13
cG(1), 182, 262, 440
cG(1)dG(r), 407
cG(2), 199
cG(q), 211, 262
characteristic, 424
  approximate, 482
characteristic Galerkin method, 476
characteristic layer, 458
characteristic streamline method, 476
characteristics, 478
chG, 476
children, 323
Cholesky's method, 146
circle of curvature, 306
clamped beam, 202
classification of differential equations, 256, 424
climate, 248
closed interval, 29
closed streamline, 457
coercive, 512
compact factorization, 146
complete, 511
completeness, 27
composite
  midpoint rule, 96
  one point Gauss rule, 96
  rectangle rule, 96
  trapezoidal rule, 98
  two point Gauss rule, 99
computational error, 285
computational mathematical modelling, 284
condition number, 155
cone of dependence, 436
conservation of energy, 114, 260, 438, 443
conservation of heat, 453
conservation of mass, 423, 428
conservation of rabbits, 207
constitutive equation, 114, 314
constitutive law, 454
continuous bilinear form, 512
continuous dual problem, 218
continuous function, 31

continuous linear form, 512
continuous stability factor, 289
continuous system, 130
continuum limit, 130
contour plot, 298
convection, 423, 452
convection dominated problem, 456
convection velocity, 453
convection-diffusion, 452
convection-diffusion problem, 522
convection-diffusion-absorption, 452
convergence, 26, 32
    uniform, 32
convex domain, 313
coordinate map, 483
coordinates, 62
Copernicus, 12
Coulomb's law, 344, 396
creeping fluid flow, 524
curvature, 305
curve, 294

d'Alembert, 14
d'Alembert's formula, 429
dashpot, 244
data error, 198, 285
decomposition, 68
Dedekind, 18
definite integration, 40
delta function, 348
density, 428
derivative, 33
dG(0), 214, 262
dG(q), 213, 261
diagonal matrix, 136
diagonalizable matrix, 69
diagonally dominant, 165
diameter of a triangle, 322
diffusion, 452
diffusion dominated, 522
diffusion dominated problem, 456
direction of time, 205
Dirichlet condition, 114, 184
discrete Laplacian, 359
discrete stability, 418
discrete stability factor, 288
discrete system, 130
discretization, 130
displacement, 522
dissipative, 206
divergence, 295
divergence theorem, 309
domain of definition, 29
dot-product, 63
dual problem, 218, 232, 264, 270, 388, 389, 415, 420, 444, 449, 470, 471
duality, 388
Duhamel's formula, 250
dynamical system, 242

eigenfunction, 398, 497
eigenfunction expansion, 497
eigenmode, 251, 398
eigenspace, 69, 497
eigenvalue, 69, 398, 497
eigenvalue problem, 122, 398, 497
    finite element method, 501
    weak form, 500
eigenvector, 68
Einstein, 14
elastic bar, 175
elastic membrane, 343, 498
elastic string, 174
electric circuit, 248
electric field, 344
electrostatics, 344
element, 321
    basis function, 327
    nodes, 330
    stiffness matrix, 367
elementary functions, 59
elementary row operations, 136
elliptic problem, 173
elliptic projection, 419, 441
elliptic smoothing, 176
endpoint rule, 95, 234
energy, 181, 356, 434

energy method, 259
energy norm, 190, 370, 512
equidistribution of error, 56, 93, 195, 381
equilibrium, 132
equilibrium equation, 523
equilibrium relation, 134
equivalent norms, 512
error, 285
error representation formula, 81, 193, 220, 223, 232, 235, 265, 270, 336, 372, 388, 416, 420, 473
essential boundary condition, 187, 384
Euclidean norm, 63
Euclidean space, 61
Euler, 13
Euler coordinates, 478
Euler equations, 317
Euler-Lagrange equation, 276
existence, 39, 238, 249

Faraday's law, 396
femions, 368
fill-in, 150
finite speed of propagation, 433
fixed point, 37
fixed point iteration, 37
flow, 423
force field, 306
forward problem, 284
Fourier
    coefficient, 177
    expansion, 497
    series, 176, 345
Fourier's law, 114, 314, 454
Fredholm integral equation, 352
freely supported, 203
front, 323
full matrix, 142
function, 29, 293
    average, 100
    generalized average, 100
    inner product, 72
    norm, 72
    support, 126
    vector spaces, 72
fundamental solution, 254, 349, 402
Fundamental Theorem of Calculus, 49

Gödel, 18
Galerkin orthogonality, 107, 213, 357, 473, 515
Galileo, 13
gas, 428
Gauss quadrature formula, 96, 99
Gauss transformation, 138
Gauss' theorem, 309
Gauss-Seidel method, 164
generalized average, 100
generalized Fourier coefficients, 497
global basis function, 327
global error, 215
global Galerkin method, 210
global stiffness matrix, 367
global support, 126
gradient, 295
graph, 30, 294
gravitational field, 347
Green's formula, 310, 466
Green's function, 184, 352
ground state, 506
Gulf Stream, 453

h-method, 126
Hamilton's principle, 281, 426
harmonic function, 316
hat function, 89
heat
    capacity coefficient, 313
    conduction, 313
    conductivity, 314
    conductivity coefficient, 114, 313
    flux, 313
heat equation, 316, 395
    fundamental solution, 402
    stationary, 114

Hilbert space, 511
Hooke's law, 131, 134, 175, 243, 523
Huygen's principle, 438
hydrogen atom, 506
hyperbolic problem, 269, 422, 424, 456

ill-conditioned matrix, 155
ill-conditioned problem, 111
implicit function theorem, 298
incompressible field, 301
infinite sequence, 26
infinite speed of propagation, 404
inflow boundary, 424, 453
initial transient, 268
initial value problem, 38, 105, 205
inner product, 63
integral equation, 352
integrating factor, 207, 258
intermediate value theorem, 38
internal energy, 181
interpolant, 78
interpolation constants, 83
interpolation nodes, 78
inverse function theorem, 300
inverse problem, 285
isolines, 298
isotropic, 302
isotropy, 322
iteration matrix, 160
iterative method, 151

Jacobi method, 164
Jacobian, 300
Jukowski, 14

Kepler, 12
Kirchhoff formula, 438
Korn's inequality, 523
Kutta, 14

Lagrange, 13
Lagrange basis, 76
Lagrange coordinates, 478
    local, 483
Lagrange's formula, 78
Lagrangian, 275, 426
Lamé coefficients, 523
Laplace's equation, 316
Laplacian, 296
    polar coordinates, 302
    spherical coordinates, 303
law of cooling, 248
Lax-Milgram theorem, 511
layers, 458
least squares method, 462
level curve, 152, 298
limit, 26, 30
line integral, 307
linear combination of vectors, 62
linear convergence, 156
linear elasticity, 522
linear form, 512
linear independence, 62
linear transformation, 67
load potential, 181
load vector, 118, 183, 358, 366
local Lagrange coordinates, 483
local support, 126
lower triangular matrix, 136
lumped mass quadrature, 341

mass matrix, 358, 408
mass-dashpot system, 245, 252
mass-spring system, 243, 252, 426
mass-spring-dashpot system, 245, 253
matrix, 68
    factorization, 138
    ill-conditioned, 155
Maxwell, 14
Maxwell's equations, 344, 396
mean value theorem, 35
mesh, 87
    isotropy, 322
    oriented, 476
mesh function, 57, 87, 322
mesh modification criterion, 195
meshes

nested, 43
midpoint rule, 95, 96, 234
minimization method, 152
minimization problem, 151, 181, 356
minimizing sequence, 514
modeling error, 198, 285
modified Newton's method, 37
modulus of elasticity, 176
momentum, 428
multi-grid method, 369
multi-physics, 452
multiplicity of an eigenvalue, 347

natural boundary condition, 187, 384
Navier-Stokes equations, 317
nested meshes, 43
Neumann condition, 184
Newfoundland, 453
Newton, 12
Newton's law of cooling, 248
Newton's law of gravitation, 350
nodal basis, 76, 90, 201
nodal basis function, 89
non-autonomous system, 241
norm, 63
L_p, 72
l_p, 64
energy, 190, 370
matrix, 70
normal matrix, 70
North American Drift, 453

ocean, 248
Ohm's law, 396
one-sided derivative, 34
open interval, 29
optimal mesh, 375
optimal order, 387
order of convergence, 215
ordinary differential equation, 38
oriented mesh, 476
orthogonal, 65
orthogonal matrix, 69
orthogonal projection, 100
outflow boundary, 453

p-method, 126
parabolic problem, 206, 266
parabolic smoothing, 400, 406
parents, 323
partial derivative, 295
partial pivoting, 144
particle path, 301
partition, 87
Pascal, 15
Petrov-Galerkin method, 462
piecewise continuous, 31
pipe, 423
pivoting, 143
Poincaré inequality, 507
Poincaré-Friedrichs' inequality, 518
point masses, 348
Poisson's equation, 316, 354, 377
minimization problem, 356
variational formulation, 355
Poisson's equation on a square, 359
Poisson's formula, 353
polar coordinates, 302
pollutant, 423
population growth, 105
positive-definite, 119
positive-definite matrix, 70
potential flow, 344
precision, 97, 340
pressure, 428
principal quantum number, 506
Principia Mathematica, 12
Principle of Archimedes, 312
principle of equidistribution, 375
principle of superposition, 242
principle of virtual work, 278
propagation of information, 458, 474
pure hyperbolic problem, 456

quadratic finite element, 329
quadrature, 41, 409
formula, 54, 96
lumped mass, 341

points, 96
weights, 96

radius of curvature, 306
range, 68
rate of convergence, 26, 156
Rayleigh quotient, 500
rectangle rule, 54, 95
reduced problem, 456
reference triangle, 361
refinement strategy, 375
regularity, 179
residual error, 106, 123, 166, 193, 210, 219, 270, 372, 388
Riemann sum, 54
rigid transformations, 302
Robin boundary condition, 184
rotation, 296, 303

scalar function, 294
scalar product, 64
scalar product norm, 64
Schrödinger equation, 506
Sd, 452
search direction, 152
separation of variables, 346, 397
sequence, 26
sequence of real numbers, 26
sharp propagation, 430
similar matrices, 69
skew-symmetric, 260
Sobolev spaces, 517
sound waves, 428
space-time domain, 453
space-time particle paths, 478
space-time prisms, 484
space-time slab, 406
sparse matrix, 119, 150
sparsity pattern of a matrix, 365
special functions, 508
spectral method, 126
spectral radius, 161
speed of propagation, 404
spherical coordinates, 303
spherically symmetric waves, 435
splitting a matrix, 159
spring constant, 243
square domain, 360
stability, 206, 215, 255, 257, 418
strong, 466, 471
weak, 465
stability factor, 220, 231, 256, 264, 388
approximation, 266
standard basis, 62
stationary convection-diffusion problem, 455
steepest descent method, 152
step function, 220
step length, 152
stiffness matrix, 118, 183, 358, 363, 408
Stoke's theorem, 311
Stokes equations, 524
stopping criterion, 54, 195
strain, 175
strain tensor, 523
streamline diffusion, 461
streamline diffusion method, 452, 466
streamlines, 457, 479
closed, 457
stress, 175
stress tensor, 522
strong boundary condition, 187
strong form, 277
strong stability estimate, 258, 406, 466, 471
subtractive cancellation, 169
support, 126, 328
surface, 294
symmetric, 64
symmetric bilinear form, 513
symmetric matrix, 119

Taylor polynomial, 36
Taylor's theorem, 36, 299
temperature, 113, 248, 313
tent functions, 327

test functions, 178
test space, 107, 378
tilting velocity, 483
total energy, 132, 133, 181, 356, 434
trace inequality, 519
transport, 423
trapezoidal rule, 98
trial functions, 178
trial space, 107, 378
triangle inequality, 63
triangle of dependence, 431
triangle of influence, 431
triangular domain, 364
triangulation, 87, 321
    boundary nodes, 378
    edges, 321
    internal edges, 357
    internal nodes, 321, 357, 378
Turing, 15
two point Gauss rule, 99
two-point boundary value problem, 39, 114

uniform convergence, 32
union jack triangulation, 364
uniqueness, 39
upper triangular matrix, 136

V-elliptic, 512
variation of constants, 207
vector
    inner product, 63
    norm, 63
    space, 61
vector field, 294
vector function, 294
vector space, 62
vector spaces of functions, 72
Vinograd's example, 255
viscosity, 461
    artificial, 461
von Neumann, 16

wave equation, 422, 498, 505
wave function, 506
waves in a resistant medium, 402
weak boundary condition, 187
weak form, 116, 277
weak stability estimate, 465
weak stability factor, 235
Weierstrass, 18
weight, 190
weighted L_2 norm, 190
weighted Cauchy's inequality, 190
weighted least squares, 461
work, 132, 307

zero pivoting, 143