1

THE POISSON EQUATION

The Poisson equation

$$-\nabla^2 u = f \tag{1.1}$$

is the simplest and the most famous elliptic partial differential equation. The source (or load) function f is given on some two- or three-dimensional domain denoted by $\Omega \subset \mathbb{R}^2$ or $\mathbb{R}^3$. A solution u satisfying (1.1) will also satisfy boundary conditions on the boundary $\partial\Omega$ of Ω; for example,

$$\alpha u + \beta \frac{\partial u}{\partial n} = g \quad \text{on } \partial\Omega, \tag{1.2}$$

where $\partial u/\partial n$ denotes the directional derivative in the direction normal to the boundary $\partial\Omega$ (conventionally pointing outward) and α and β are constant, although variable coefficients are also possible. In a practical setting, u could represent the temperature field in Ω subject to the external heat source f. Other important physical models include gravitation, electromagnetism, elasticity and inviscid fluid mechanics; see Ockendon et al. [174, Chapter 5] for motivation and discussion.

The combination of (1.1) and (1.2) together is referred to as a *boundary value problem*. If the constant β in (1.2) is zero, then the boundary condition is of Dirichlet type, and the boundary value problem is referred to as the Dirichlet problem for the Poisson equation. Alternatively, if the constant α is zero, then we correspondingly have a Neumann boundary condition, and a Neumann problem. A third possibility is that Dirichlet conditions hold on part of the boundary $\partial\Omega_D$, and Neumann conditions (or indeed *mixed* conditions where α and β are both nonzero) hold on the remainder $\partial\Omega \backslash \partial\Omega_D$.

The case $\alpha = 0$, $\beta = 1$ in (1.2) demands special attention. First, since $u =$ constant satisfies the *homogeneous* problem with $f = 0$, $g = 0$, it is clear that a solution to a Neumann problem can only be unique up to an additive constant. Second, integrating (1.1) over Ω using Gauss's theorem gives

$$-\int_{\partial\Omega} \frac{\partial u}{\partial n} = -\int_{\Omega} \nabla^2 u = \int_{\Omega} f; \tag{1.3}$$

thus a necessary condition for the existence of a solution to a Neumann problem is that the source and boundary data satisfy the *compatibility* condition

$$\int_{\partial\Omega} g + \int_{\Omega} f = 0. \tag{1.4}$$

Finite Element Methods and Fast Iterative Solvers. Second Edition. Howard C. Elman, David J. Silvester, Andrew J. Wathen © Howard C. Elman, David J. Silvester, Andrew J. Wathen 2014. Published in 2014 by Oxford University Press.

1.1 Reference problems

The following examples of two-dimensional Poisson problems will be used to illustrate the power of the finite element approximation techniques that are developed in the remainder of the chapter. Since these problems are all of Dirichlet type (that is, the boundary condition associated with (1.1) is of the form $u = g$ on $\partial\Omega$), the problem specification involves the shape of the domain Ω, the source data f and the boundary data g. The examples are posed on one of two domains: a square $\Omega_\square = (-1, 1) \times (-1, 1)$, or an L-shaped domain $\Omega_\lrcorner$ consisting of the complement in $\Omega_\square$ of the quadrant $(-1, 0] \times (-1, 0]$.

1.1.1 Example: Square domain $\Omega_\square$, constant source function $f(x) \equiv 1$, zero boundary condition.

This problem represents a simple diffusion model for the temperature distribution $u(x, y)$ in a square plate. The specific source term in this example models uniform heating of the plate, and the boundary condition models the edge of the plate being kept at an ice-cold temperature. The simple shape of the domain enables the solution to be explicitly represented. Specifically, using separation of variables it can be shown that

$$u(x,y) = \frac{(1-x^2)}{2} - \frac{16}{\pi^3} \sum_{\substack{k=1 \\ k \text{ odd}}}^{\infty} \left\{ \frac{\sin(k\pi(1+x)/2)}{k^3 \sinh(k\pi)} \times \left(\sinh\left(\frac{k\pi(1+y)}{2}\right) + \sinh\left(\frac{k\pi(1-y)}{2}\right)\right)\right\}. \tag{1.5}$$

Series solutions of this type can only be found in the case of geometrically simple domains. Moreover, although such solutions are aesthetically pleasing to mathematicians, they are rather less useful in terms of computation. These are the raisons d'etre for approximation strategies such as the finite element method considered in this monograph.

A finite element solution (computed using our IFISS software) approximating the exact solution u is illustrated in Figure 1.1. The accuracy of the computed solution is explored in Computational Exercise 1.1.

1.1.2 Example: L-shaped domain $\Omega_\lrcorner$, constant source function $f(x) \equiv 1$, zero boundary condition.

A typical finite element solution is illustrated in Figure 1.2 (and is again easily computed using our IFISS software; see Computational Exercise 1.4). Notice that the contours are very close together around the corner at the origin, suggesting that the temperature is rapidly varying in this vicinity. A more careful

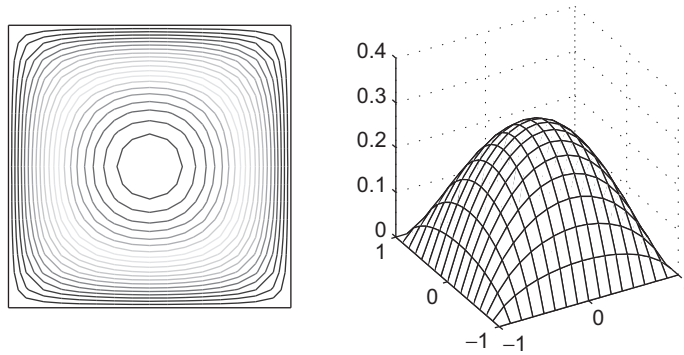


FIG. 1.1. Contour plot (left) and three-dimensional surface plot (right) of a finite element solution of Example 1.1.1.

investigation shows that the underlying Poisson problem has a *singularity*—the solution u is closely approximated at the origin by the function

$$u^*_\Gamma(r,\theta) = r^{2/3}\sin((2\theta+\pi)/3), \tag{1.6}$$

where r represents the radial distance from the origin, and θ the angle with the vertical axis. This singular behavior is identified more precisely in Example 1.1.4. Here we simply note that radial derivatives of u^*_Γ (and by implication those of u) are unbounded at the origin. See Strang & Fix [244, Chapter 8] for further discussion of this type of function.

In order to assess the accuracy of approximations to the solution of boundary value problems in this and subsequent chapters, it will be convenient to refer to

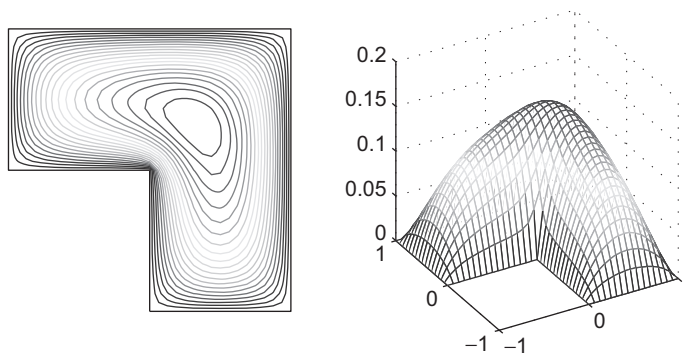


FIG. 1.2. Contour plot (left) and three-dimensional surface plot (right) of a finite element solution of Example 1.1.2.

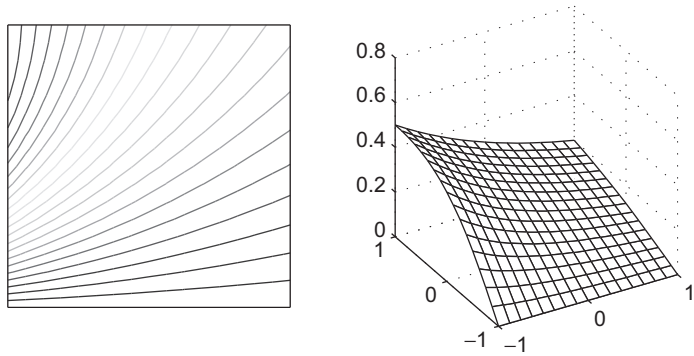


FIG. 1.3. Contour plot (left) and three-dimensional surface plot (right) of a finite element solution of Example 1.1.3.

analytic test problems—these have an exact solution that can be explicitly computed at all points in the domain. Examples 1.1.3 and 1.1.4 are in this category.

1.1.3 Example: Square domain $\Omega_\square$, analytic solution.

This analytic test problem is associated with the following solution of Laplace's equation (that is, (1.1) with $f = 0$):

$$u^*(x, y) = \frac{2(1+y)}{(3+x)^2 + (1+y)^2}. \tag{1.7}$$

Note that this function is perfectly smooth since the domain $\Omega_\square$ excludes the point $(-3, -1)$. A finite element approximation to u^* is given in Figure 1.3. For future reference we note that the boundary data g is given by the finite element interpolant of u^* on $\partial\Omega_\square$. We will return to this example when we consider finite element approximation errors in Section 1.5.1.

1.1.4 Example: L-shaped domain $\Omega_\sqsubset$, analytic solution.

This analytic test problem is associated with the singular solution $u^*_\sqsubset$ introduced in Example 1.1.2; see Problem 1.1. A typical finite element approximation to $u^*_\sqsubset$ is given in Figure 1.4. We will return to this example when discussing a posteriori error estimation in Section 1.5.2.

1.2 Weak formulation

A sufficiently smooth function u satisfying both (1.1) and (1.2) is known as a *classical solution* to the boundary value problem; see Renardy & Rogers [201]. For a Dirichlet problem, u is a classical solution only if it has continuous second derivatives in Ω (that is, u is in $C^2(\Omega)$) and is continuous up to the boundary (u is in $C^0(\overline{\Omega})$); see Braess [26, p. 34] for further details. In cases of nonsmooth domains

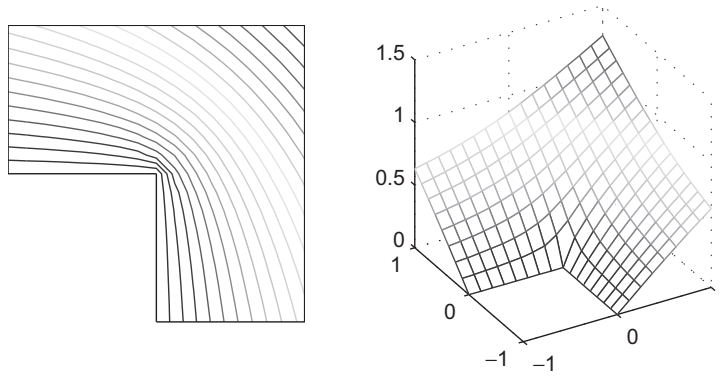


FIG. 1.4. Contour plot (left) and three-dimensional surface plot (right) of a finite element solution of Example 1.1.4.

or discontinuous source functions, the function u satisfying (1.1)–(1.2) may not be smooth (or *regular*) enough to be regarded as a classical solution. As we have observed, on the nonconvex domain Ω_{Γ} of Example 1.1.4, the solution u^*_{Γ} does not have a square-integrable second derivative (see Problem 1.27 and the discussion in Section 1.5.1). Alternatively, suppose that the source function is discontinuous, say $f=1$ on $\{(x,y)\,|\,0<x<1\} \subset \Omega_{\square}$ and $f=0$ on $\{(x,y)\,|\,-1<x<0\}$, which corresponds to a weight placed on part of an elastic membrane. Since f is discontinuous in the x direction, the second partial derivative of the solution u with respect to x is discontinuous, and hence u cannot be in $C^2(\Omega)$, and there is no classical solution. For such problems, which arise from perfectly reasonable mathematical models, an alternative description of the boundary value problem is required. Since this alternative description is less restrictive in terms of the admissible data, it is called a *weak formulation*.

To derive a weak formulation of a Poisson problem, we require that for an appropriate set of *test functions* v,

$$\int_{\Omega} (\nabla^2 u + f)v = 0. \tag{1.8}$$

This formulation exists provided that the integrals are well defined. If u is a classical solution then it must also satisfy (1.8). If v is sufficiently smooth however, then the smoothness required of u can be reduced by using the derivative of a product rule and the divergence theorem,

$$\begin{aligned} -\int_{\Omega} v\nabla^2 u &= \int_{\Omega} \nabla u \cdot \nabla v - \int_{\Omega} \nabla \cdot (v\nabla u) \\ &= \int_{\Omega} \nabla u \cdot \nabla v - \int_{\partial\Omega} v\frac{\partial u}{\partial n}, \end{aligned}$$

so that

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega vf + \int_{\partial\Omega} v\frac{\partial u}{\partial n}. \tag{1.9}$$

The point here is that the problem (1.9) may have a solution u, called a *weak solution*, that is not smooth enough to be a classical solution. If a classical solution does exist then (1.9) is equivalent to (1.1)–(1.2) and the weak solution is classical.

The case of a Neumann problem ($\alpha = 0$, $\beta = 1$ in (1.2)) is particularly straightforward. Substituting (1.2) into (1.9) gives the following formulation: find u defined on Ω such that

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega vf + \int_{\partial\Omega} vg \tag{1.10}$$

for all suitable test functions v.

We need to address an important question at this point, namely, in what sense are the weak solution u and the test functions v in (1.10) meaningful? This is essentially a question of *where* to look to find the solution u, and what is meant by "all suitable v." To provide an answer we use the space of functions that are square integrable in the sense of Lebesgue,

$$L_2(\Omega) := \left\{ u : \Omega \to \mathbb{R} \;\middle|\; \int_\Omega u^2 < \infty \right\}, \tag{1.11}$$

and make use of the L_2 measure,

$$\|u\| := \left(\int_\Omega u^2 \right)^{1/2}. \tag{1.12}$$

The integral on the left-hand side of (1.10) will be well defined if all first derivatives are in $L_2(\Omega)$; for example, if Ω is a two-dimensional domain and $\partial u/\partial x, \partial u/\partial y \in L_2(\Omega)$ with $\partial v/\partial x, \partial v/\partial y \in L_2(\Omega)$, then using the Cauchy–Schwarz inequality,

$$\begin{aligned} \int_\Omega \nabla u \cdot \nabla v &= \int_\Omega \left(\frac{\partial u}{\partial x}\right)\left(\frac{\partial v}{\partial x}\right) + \int_\Omega \left(\frac{\partial u}{\partial y}\right)\left(\frac{\partial v}{\partial y}\right) \\ &\le \left\|\frac{\partial u}{\partial x}\right\| \left\|\frac{\partial v}{\partial x}\right\| + \left\|\frac{\partial u}{\partial y}\right\| \left\|\frac{\partial v}{\partial y}\right\| < \infty. \end{aligned}$$

Similarly, the integrals on the right-hand side of (1.10) will certainly be well defined if $f \in L_2(\Omega)$ and $g \in L_2(\partial\Omega)$.[1]

[1]The boundary term can be shown to be well defined using the *trace inequality* given in Lemma 1.5.

To summarize, if $\Omega \subset \mathbb{R}^2$ then the *Sobolev space* $\mathcal{H}^1(\Omega)$ given by

$$\mathcal{H}^1(\Omega) := \left\{ u : \Omega \to \mathbb{R} \,\middle|\, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \in L_2(\Omega) \right\}$$

is the space where a weak solution of (1.10) naturally exists, and this space is also the natural home for the test functions v. For clarity of exposition, further discussion of such technical issues is postponed until Section 1.5.

We now return to (1.9) and consider other types of boundary conditions. In general, we only need to look for weak solutions among those functions that satisfy the Dirichlet boundary conditions. (Engineers call Dirichlet boundary conditions "essential conditions" whereas Neumann conditions are "natural conditions" for the Laplacian.) To fix ideas, in the remainder of the chapter we restrict our attention to the following generic boundary value problem:

find u such that

$$-\nabla^2 u = f \quad \text{in } \Omega, \tag{1.13}$$

$$u = g_D \text{ on } \partial\Omega_D \quad \text{and} \quad \frac{\partial u}{\partial n} = g_N \text{ on } \partial\Omega_N, \tag{1.14}$$

where $\partial\Omega_D \cup \partial\Omega_N = \partial\Omega$ and $\partial\Omega_D$ and $\partial\Omega_N$ are distinct.

We assume that $\int_{\partial\Omega_D} \mathrm{d}s \neq 0$, so that (1.14) does not represent a Neumann condition. Then we define *solution* and *test* spaces by

$$\mathcal{H}^1_E := \{u \in \mathcal{H}^1(\Omega) \,|\, u = g_D \text{ on } \partial\Omega_D\}, \tag{1.15}$$

$$\mathcal{H}^1_{E_0} := \{v \in \mathcal{H}^1(\Omega) \,|\, v = 0 \text{ on } \partial\Omega_D\}, \tag{1.16}$$

respectively. We should emphasize the difference between the two spaces: the Dirichlet condition from (1.14) is built into the definition of the solution space $\mathcal{H}^1_E$, whereas functions in the test space $\mathcal{H}^1_{E_0}$ are zero on the Dirichlet portion of the boundary. This is in contrast to the Neumann case where the solution and the test functions are not restricted on the boundary. Notice that the solution space is not closed under addition so strictly speaking it is not a vector space.

From (1.9) it is clear that any function u that satisfies (1.13) and (1.14) is also a solution of the following weak formulation:

find $u \in \mathcal{H}^1_E$ such that

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega v f + \int_{\partial\Omega_N} v g_N \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.17}$$

We reiterate a key point here: a classical solution of a Poisson problem has to be twice differentiable in Ω—this is a much more stringent requirement than square integrability of first derivatives. Using (1.17) instead as the starting point

enables us to look for approximate solutions that only need satisfy the smoothness requirement and the essential boundary condition embodied in (1.15). The case of a Poisson problem with a mixed boundary condition (1.2) is explored in Problem 1.2.

1.3 The Galerkin finite element method

We now develop the idea of approximating u by taking a finite-dimensional subspace of the solution space $\mathcal{H}^1_E$. The starting point is the weak formulation (1.15)–(1.17) of the generic problem (1.13)–(1.14). To construct an approximation method, we assume that $S^h_0 \subset \mathcal{H}^1_{E_0}$ is an n-dimensional vector space of test functions for which $\{\phi_1, \phi_2, \ldots, \phi_n\}$ is a convenient basis. Then, in order to ensure that the Dirichlet boundary condition in (1.15) is satisfied, we extend this basis set by defining additional functions $\phi_{n+1}, \ldots, \phi_{n+n_\partial}$ and select fixed coefficients $\mathbf{u}_j, j = n+1, \ldots, n+n_\partial$, so that the function $\sum_{j=n+1}^{n+n_\partial} \mathbf{u}_j \phi_j$ interpolates the boundary data g_D on $\partial\Omega_D$. The finite element approximation $u_h \in S^h_E$ is then uniquely associated with the vector $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n)^T$ of real coefficients in the expansion

$$u_h = \sum_{j=1}^{n} \mathbf{u}_j \phi_j + \sum_{j=n+1}^{n+n_\partial} \mathbf{u}_j \phi_j. \tag{1.18}$$

The functions $\phi_i, i = 1, \ldots, n$ in the first sum in (1.18) define a set of *trial functions*. (In a finite element context, they are often called *shape* functions.)

The construction (1.18) cleverly simplifies the characterization of discrete solutions when faced with difficult-to-satisfy essential boundary data,[2] for example, when solving test problems like that in Example 1.1.3.

The construction of the space S^h_E is achieved by ensuring that the specific choice of trial functions in (1.18) coincides with the choice of test functions that form the basis for S^h_0, and is generally referred to as the *Galerkin* (or more precisely *Bubnov–Galerkin*) approximation method. A more general approach is to construct approximation spaces for (1.15) and (1.16) using different trial and test functions. This alternative is called a *Petrov–Galerkin* approximation method, and a specific example will be discussed in Chapter 6.

The result of the Galerkin approximation is a finite-dimensional version of the weak formulation: find $u_h \in S^h_E$ such that

$$\int_\Omega \nabla u_h \cdot \nabla v_h = \int_\Omega v_h f + \int_{\partial\Omega_N} v_h g_N \quad \text{for all } v_h \in S^h_0. \tag{1.19}$$

[2] But it complicates the error analysis (see Section 1.5); if the data g_D is approximated then $S^h_E \not\subset \mathcal{H}^1_E$.

For computations, it is convenient to enforce (1.19) for each basis function; then it follows from (1.18) that (1.19) is equivalent to finding $\mathbf{u}_j, j = 1, \ldots, n$ such that

$$\sum_{j=1}^{n} \mathbf{u}_j \int_\Omega \nabla\phi_j \cdot \nabla\phi_i = \int_\Omega \phi_i f \; + \int_{\partial\Omega_N} \phi_i g_N \; - \sum_{j=n+1}^{n+n_\partial} \mathbf{u}_j \int_\Omega \nabla\phi_j \cdot \nabla\phi_i \quad (1.20)$$

for $i = 1, \ldots, n$. This can be written in matrix form as the linear system of equations

$$A\mathbf{u} = \mathbf{f} \quad (1.21)$$

with

$$A = [a_{ij}], \quad a_{ij} = \int_\Omega \nabla\phi_j \cdot \nabla\phi_i, \quad (1.22)$$

and

$$\mathbf{f} = [\boldsymbol{f}_i], \qquad \boldsymbol{f}_i = \int_\Omega \phi_i f + \int_{\partial\Omega_N} \phi_i g_N \; - \sum_{j=n+1}^{n+n_\partial} \mathbf{u}_j \int_\Omega \nabla\phi_j \cdot \nabla\phi_i. \quad (1.23)$$

The system of linear equations (1.21) is called the *Galerkin system*, and the function u_h, computed by substituting the solution of (1.21) into (1.18), is called the *Galerkin solution*. The matrix A is also referred to as the *stiffness matrix*.

The Galerkin coefficient matrix (1.22) is clearly symmetric (in contrast, using different test and trial functions necessarily leads to a nonsymmetric system matrix), and it is also positive definite. To see this, consider a general coefficient vector $\mathbf{v}$ corresponding to a specific function $v_h = \sum_{j=1}^{n} \mathbf{v}_j\phi_j \in S_0^h$, so that

$$\begin{aligned}
\mathbf{v}^T A\mathbf{v} &= \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{v}_j a_{ji}\mathbf{v}_i \\
&= \sum_{j=1}^{n}\sum_{i=1}^{n} \mathbf{v}_j \left(\int_\Omega \nabla\phi_j \cdot \nabla\phi_i \right) \mathbf{v}_i \\
&= \int_\Omega \left(\sum_{j=1}^{n} \mathbf{v}_j \nabla\phi_j \right) \cdot \left(\sum_{i=1}^{n} \mathbf{v}_i \nabla\phi_i \right) \\
&= \int_\Omega \nabla v_h \cdot \nabla v_h \\
&\geq 0.
\end{aligned}$$

Thus we see that A is at least semidefinite. Definiteness follows from the fact that $\mathbf{v}^T A\mathbf{v} = 0$ if and only if $\nabla v_h = 0$, that is, if and only if v_h is constant in Ω. Since $v_h \in S_0^h$, it is continuous up to the boundary and is zero on $\partial\Omega_D$, thus $\nabla v_h = 0$ implies $v_h = 0$. Finally, since the test functions are a basis for S_0^h we have that $v_h = 0$ implies $\mathbf{v} = \mathbf{0}$.

Once again the Neumann problem (1.10) requires special consideration. The Galerkin matrix is only semidefinite in this case and has a null space of vectors $\mathbf{v}$ corresponding to functions $\nabla v_h = 0$. In this situation it is essential to constrain the subspace $S_h \subset \mathcal{H}^1$ by choosing a set of trial functions $\{\phi_j\}, j = 1, \ldots, n$ that define a *partition of unity*, that is, every vector in S_h must be associated with a coefficient vector $v_h = \sum_{j=1}^n \mathbf{v}_j \phi_j$ satisfying

$$\sum_{j=1}^{n} \phi_j = 1. \tag{1.24}$$

The construction (1.24) ensures that if v_h is a constant function, say $v_h \equiv \alpha$, then v_h is associated with a discrete vector that satisfies $\mathbf{v}_j = \alpha$ for all the coefficients. This means that the null space of the Galerkin matrix associated with (1.10) is one-dimensional, consisting of constant coefficient vectors. Notice that the solvability of the discrete Neumann system (the analogue of (1.21)) requires that the null space of the Galerkin matrix A be orthogonal to the right-hand-side vector $\mathbf{f}$, that is, we require that $(1, \ldots, 1)^T \mathbf{f} = 0$ with

$$\mathbf{f} = [\boldsymbol{f}_i], \qquad \boldsymbol{f}_i = \int_\Omega \phi_i \boldsymbol{f} + \int_{\partial\Omega} \phi_i g. \tag{1.25}$$

Using the property (1.24) shows that the discrete Neumann problem is solvable if and only if the underlying boundary value problem is well posed in the sense that (1.4) holds.

Returning to the general case (1.19), it is clear that the choices of S_E^h and S_0^h are central in that they determine whether or not u_h has any relation to the weak solution u. The inclusions $S_E^h \subset \mathcal{H}_E^1$ and $S_0^h \subset \mathcal{H}_{E_0}^1$ lead to *conforming* approximations; more general *nonconforming* approximation spaces containing specific discontinuous functions are also possible (see for example Braess [26, pp. 104–106]), but these are not considered here. The general desire is to choose S_E^h and S_0^h so that approximation to any required accuracy can be achieved if the dimension n is large enough. That is, it is required that the error $\|u - u_h\|$ reduces rapidly as n is increased, and moreover that the computational effort associated with solving (1.21) is acceptable—the choice of basis is critical in this respect. These issues are addressed in Section 1.5 and in Chapter 2.

The mathematical motivation for finite element approximation is the observation that a smooth function can often be approximated to arbitrary accuracy using *piecewise polynomials*. Starting from the Galerkin system (1.19), the idea is to choose basis functions $\{\phi_j\}$ in (1.18) that are locally nonzero on a mesh of triangles ($\mathbb{R}^2$) or tetrahedra ($\mathbb{R}^3$) or a grid of rectangles or bricks. We discuss two-dimensional elements first.

1.3.1 *Triangular elements* ($\mathbb{R}^2$)

For simplicity, we assume that $\Omega \subset \mathbb{R}^2$ is polygonal (as is often the case in practice), so that we are able to *tile* (or *tessellate*) the domain with a set of

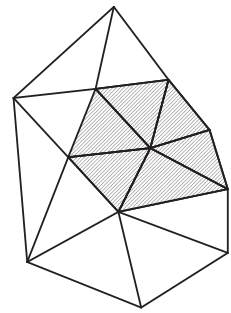


FIG. 1.5. A triangular mesh with a patch shaded.

triangles $\triangle_k, k = 1, \ldots, K$ defining a *triangulation* $\mathcal{T}_h$. This means that vertices of neighboring triangles coincide and that

- $\bigcup_k \overline{\triangle}_k = \overline{\Omega}$;
- $\triangle_\ell \cap \triangle_m = \emptyset$ for $\ell \neq m$.

The points where triangle vertices meet are called *nodes*. Surrounding any node is a *patch* of triangles that each have that node as a vertex (see Figure 1.5). If we label the nodes $j = 1, \ldots, n$, then for each j, we define a basis function ϕ_j that is nonzero only on that patch. The simplest choice here (leading to a conforming approximation) is the $\boldsymbol{P}_1$ or piecewise linear basis function: ϕ_j is a linear function on each triangle, which takes the value 1 at the node point j and 0 at all other node points on the mesh. Notice that ϕ_j is clearly continuous on Ω (see Figure 1.6). Moreover, although ϕ_j has discontinuities in slope across element boundaries, it is smooth enough that $\phi_j \in \mathcal{H}^1(\Omega)$, and so it leads to a conforming approximation space $S_0^h = \text{span}(\phi_1, \phi_2, \ldots, \phi_n)$ for use with (1.19).

In terms of approximation, the precise choice of basis for the space is not important; for practical application however, the availability of a locally defined

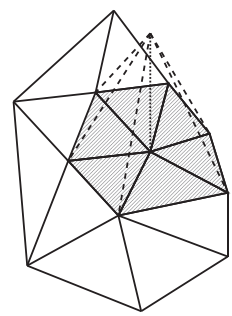


FIG. 1.6. A $\boldsymbol{P}_1$ basis function.

basis such as this one is crucial. Having only three basis functions that are not identically zero on a given triangle means that the construction of the Galerkin matrix A in (1.21) is easily automated. Another important point is that the Galerkin matrix has a well-defined *sparse* structure: $a_{ij} \neq 0$ only if the node points labeled i and j lie on the same edge of a triangular element. This is important for the development of efficient methods for solving the linear system (1.21); see Chapter 2.

Summarizing, $\boldsymbol{P}_1$ approximation can be characterized by saying that the overall approximation is continuous, and that on any element with vertices i, j and k there are only the three basis functions ϕ_i, ϕ_j and ϕ_k that are not identically zero. Within an element, ϕ_i is a linear function that takes the value 1 at node i and 0 at nodes j and k . This local characterization is convenient for implementation of the finite element method (see Section 1.4) and it is also useful for the description of piecewise polynomial approximation spaces of higher degree.

For piecewise quadratic (or $\boldsymbol{P}_2$) approximation it is convenient to introduce additional nodes at the midpoint of each edge. Thus on each triangle there are six nodes, giving six basis functions that are not identically zero (recall that quadratic functions are of the form $ax^2 + bxy + cy^2 + dx + ey + f$ and thus have six coefficients). As in the linear case, we choose basis functions that have the value 1 at a single node and 0 at the other nodes, as illustrated in Figure 1.7; see also Problem 1.6. These define a global approximation space of piecewise quadratic functions on the triangulation $\mathcal{T}_h$. Note that there are now "edge" as well as "vertex" functions, and that continuity across edges is guaranteed since there is a unique univariate quadratic (parabola) that takes given values at three points.

The illustration in Figure 1.8 is a convenient way to represent the $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ triangular elements. In Section 1.5 we will show that there can be advantages in the use of higher-order approximations, in terms of the accuracy of approximation. The construction of higher-order approximations ($\boldsymbol{P}_m$ with $m \geq 3$) is a straightforward generalization; see Braess [26, pp. 65ff.].

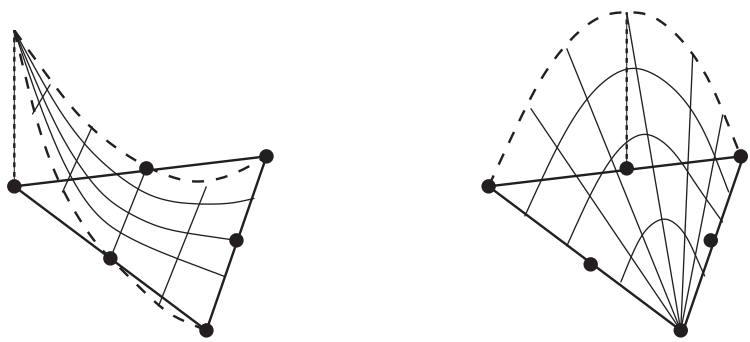


FIG. 1.7. $\boldsymbol{P}_2$ basis functions of vertex type (left) and edge type (right).

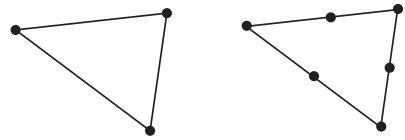


FIG. 1.8. Representation of $\boldsymbol{P}_1$ (left) and $\boldsymbol{P}_2$ (right) elements.

1.3.2 *Quadrilateral elements* ($\mathbb{R}^2$)

Although they are less flexible than triangular elements, it is often convenient to consider grids made up of rectangular (or more general quadrilateral) elements. For the simplest domains such as $\Omega_\square$ or $\Omega_\square$ in Section 1.1, it is clearly trivial to tile using square or rectangular elements. For more general domains, it is possible to use rectangles in the interior and then use triangles to match up to the boundary.

The simplest conforming quadrilateral element for a Poisson problem is the bilinear $\boldsymbol{Q}_1$ element defined as follows. On a rectangle, each function is of the form $(ax+b)(cy+d)$ (hence bilinear). Again, for each of the four basis functions ϕ_j that are not identically zero on an element, the four coefficients are defined by the conditions that ϕ_j has the value 1 at vertex j and 0 at all other vertices. For example, on an element $x \in [0,h]$, $y \in [0,h]$, the *element basis functions* are

$$(1-x/h)(1-y/h), \quad x/h(1-y/h), \quad xy/h^2, \quad (1-x/h)y/h,$$

starting with the function that is 1 at the origin and then moving anticlockwise. The global basis function on a patch of four elements is shown in Figure 1.9. Note that the $\boldsymbol{Q}_1$ element has the additional "twist" term xy, which is not present in the $\boldsymbol{P}_1$ triangle, and this generally gives the approximate solution some nonzero curvature on each element. Notice however, that when restricted to an edge, the $\boldsymbol{Q}_1$ element behaves like the $\boldsymbol{P}_1$ triangle since it varies linearly. In both cases the approximation is continuous but has a discontinuous normal derivative. The upshot is that the $\boldsymbol{Q}_1$ rectangle is in $\mathcal{H}^1(\Omega)$ and is hence conforming for (1.19).

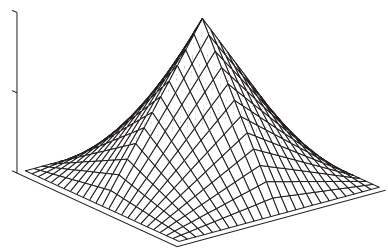


FIG. 1.9. A typical $\boldsymbol{Q}_1$ basis function.

In the case of arbitrary quadrilaterals, straightforward bilinear approximation as described above does not lead to a conforming approximation. A bilinear function is generally quadratic along an edge that is not aligned with a coordinate axis, and so it is not uniquely defined by its value at the two endpoints. This difficulty can be overcome by defining the approximation through an *isoparametric transformation*. The idea is to define the element basis functions

$$\begin{aligned}\chi_1(\xi,\eta) &= (\xi-1)(\eta-1)/4,\\ \chi_2(\xi,\eta) &= -(\xi+1)(\eta-1)/4,\\ \chi_3(\xi,\eta) &= (\xi+1)(\eta+1)/4,\\ \chi_4(\xi,\eta) &= -(\xi-1)(\eta+1)/4\end{aligned} \tag{1.26}$$

on a *reference* element $\xi \in [-1,1]$, $\eta \in [-1,1]$, and then to map to any general quadrilateral with vertex coordinates (x_ν, y_ν), $\nu = 1,2,3,4$ by the change of variables

$$x(\xi,\eta) = \sum_{\nu=1}^{4} x_\nu \chi_\nu(\xi,\eta), \quad y(\xi,\eta) = \sum_{\nu=1}^{4} y_\nu \chi_\nu(\xi,\eta); \tag{1.27}$$

see Figure 1.10. The outcome is that the mapped element basis function defined on the general element through (1.27) is linear along each element edge, and so it will connect continuously to the adjacent quadrilateral element whose basis will be defined isoparametrically based on its own vertex positions; see Problem 1.8. Element mappings will be discussed in detail in Section 1.4. Notice that when using triangles, one could employ a similar isoparametric transformation to a reference triangle based on the $\boldsymbol{P}_1$ basis; see Section 1.4.1 for details.

Higher-order approximations are defined analogously. For example, we can define a biquadratic finite element approximation on rectangles by introducing four additional midside node points, together with a ninth node at the centroid, as illustrated in the pictorial representation of Figure 1.11. In this case there are

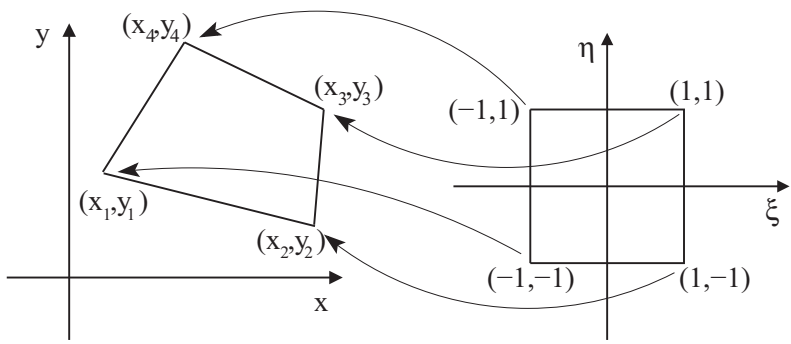


FIG. 1.10. Isoparametric mapping of a $\boldsymbol{Q}_1$ element.

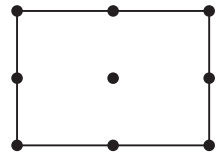


FIG. 1.11. Representation of a $\boldsymbol{Q}_2$ element.

four vertex functions, four edge functions and one internal (or *bubble*) function in the element basis. The resulting approximation—which on each rectangle is of the form $(ax^2 + bx + c)(dy^2 + ey + f)$—is a linear combination of the nine terms $1, x, y, x^2, xy, y^2, x^2y, xy^2, x^2y^2$ and is called $\boldsymbol{Q}_2$. Note that just as $\boldsymbol{Q}_1$ approximation is a complete linear polynomial together with the xy term of a bivariate quadratic, $\boldsymbol{Q}_2$ has all six terms of a complete quadratic plus the two cubic terms x^2y, xy^2 and the single quartic term x^2y^2.

Clearly $\boldsymbol{Q}_2$ approximation on rectangles is continuous (for exactly the same reason as $\boldsymbol{P}_2$) and so is conforming for (1.19); $\boldsymbol{Q}_2$ approximation may also be employed on arbitrary quadrilaterals through use of the bilinear mapping (1.27) (in which case the mapping is *subparametric*). Another point worth noting is that triangles and quadrilaterals may be used together in a conforming approximation space. For example, $\boldsymbol{P}_2$ and $\boldsymbol{Q}_2$ both have quadratic variation along edges and so can be used together.

Higher-degree piecewise polynomials may also be defined on rectangles (and thus on quadrilaterals) but other possibilities also present themselves; for example, by excluding the centroid node, one is left with eight degrees of freedom, which allows the construction of a basis including all $\boldsymbol{Q}_2$ terms except for the x^2y^2 term; see for example Braess [26, pp. 66ff.]. Such an element is a member of the "serendipity" family.

1.3.3 *Tetrahedral elements* ($\mathbb{R}^3$)

The natural counterpart to triangular elements in three dimensions is tetrahedral (or simplex) elements. Any polyhedral region $\Omega \subset \mathbb{R}^3$ can be completely filled with tetrahedra $\triangle_k$, where each triangular face is common to only two tetrahedra, or else is part of the boundary $\partial\Omega$. Thus in a manner analogous to how triangles are treated, we define nodes at the vertices of the faces of the tetrahedra, and we define a $\boldsymbol{P}_1$ basis function ϕ_j that is only nonzero on the set of tetrahedra for which node j is a vertex of one of its faces.

For each node j in a tessellation of Ω, we define $\phi_j(x, y, z)$ to be a linear function (that is, of the form $a + bx + cy + dz$) on each tetrahedral element satisfying the interpolation condition

$$\phi_j(\text{node } i) = \begin{cases} 1 & \text{when } i = j, \\ 0 & \text{when } i \neq j. \end{cases} \tag{1.28}$$

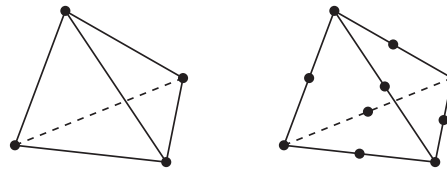


FIG. 1.12. Representation of $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ tetrahedral elements.

Each basis function ϕ_j is continuous, thus ensuring a conforming approximation space, $S_0^h = \text{span}(\phi_1, \phi_2, \ldots, \phi_n) \subset \mathcal{H}^1_{E_0}$, where n is the number of nodes, as before. Note that there are precisely four basis functions that are nonzero on any particular tetrahedral element, corresponding to the four coefficients needed to define the linear approximation in the element. This also leads to a convenient implementation, exactly as in the triangular case.

Higher-order tetrahedral elements are defined by introducing additional nodes. For example, the $\boldsymbol{P}_2$ element has additional midedge nodes as depicted in Figure 1.12. This gives ten nodes in each element, matching the ten coefficients needed to define a trivariate quadratic polynomial (of the form $a + bx + cy + dz + ex^2 + fy^2 + gz^2 + hxy + kxz + lyz$). On any triangular face (which defines a plane, $\hat{a}x + \hat{b}y + \hat{c}z = \hat{d}$), one of the variables, z say, can be eliminated in terms of a linear combination of $1, x$ and y, to give a bivariate quadratic (in x, y) that is uniquely determined by its value at the six nodes of the $\boldsymbol{P}_2$ triangular element. As a result, continuity across interelement faces, and hence a conforming approximation for (1.19), is ensured.

More generally, $\boldsymbol{P}_m$ elements corresponding to continuous piecewise mth-degree trivariate polynomial approximation are defined by an obvious generalization of the bivariate triangular analogue. All such tetrahedral elements have a gradient whose normal component is discontinuous across interelement faces.

1.3.4 *Brick elements* ($\mathbb{R}^3$)

Three-dimensional approximation on cubes (or more generally *bricks*, which have six rectangular faces) is realized by taking the *tensor product* of lower-dimensional elements. Thus the simplest conforming element for (1.19) is the trilinear $\boldsymbol{Q}_1$ element that takes the form $(ax + b)(cy + d)(ez + f)$ on each brick. Written as a linear combination, there are eight terms, $1, x, y, z, xy, xz, yz, xyz$, and the eight coefficients are determined using the eight corner nodes, as illustrated in Figure 1.13. As a result, adopting the standard definition of a trilinear basis satisfying (1.28), there are precisely eight basis functions that are not identically zero within each element.

The recipe for higher-order approximation on bricks is obvious. The $\boldsymbol{Q}_2$ triquadratic represented in Figure 1.13 has twenty-seven nodes; there are eight corner basis functions, twelve midedge basis functions, six midface basis functions and a single bubble function associated with the node at the centroid.

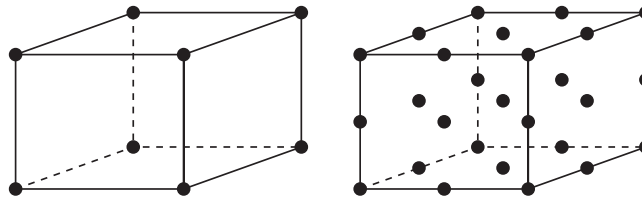


FIG. 1.13. Representation of $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ brick elements.

Finally, we note that elements with six quadrilateral faces can be defined analogously to the two-dimensional case via a trilinear parametric mapping to the unit cube.

1.4 Implementation aspects

The computation of a finite element approximation consists of the following tasks. These are all built into the IFISS software.

(i) Input of the data on $\Omega, \partial\Omega_N, \partial\Omega_D$ defining the problem to be solved
(ii) Generation of a grid or mesh of elements
(iii) Construction of the Galerkin system
(iv) Solution of the discrete system, using a linear solver that exploits the sparsity of the finite element coefficient matrix
(v) A posteriori error estimation

In this section we focus on the core aspect (iii) of setting up the discrete Galerkin system (1.21). Other key aspects, namely, the solution of this system and a posteriori error analysis, are treated in Chapter 2 and Section 1.5, respectively. Postprocessing of the solution is also required in general. This typically involves visualization and the calculation of derived quantities (for example, boundary derivatives). A posteriori error analysis is particularly important. If the estimated errors are larger than desired, then the approximation space may be increased in dimension, either through local mesh subdivision (*h-refinement*) or by increasing the order of the local polynomial basis (*p-refinement*). An acceptable solution may then be calculated by cycling through steps (ii)–(v) in an efficient way that builds on the existing structure, until the required error tolerance is satisfied.

The key idea in the implementation of finite element methodology is to consider everything "elementwise," that is, locally one element at a time. In effect the discrete problem is broken up; for example, (1.20) is rewritten as

$$\sum_{j=1}^{n} \mathbf{u}_j \int_\Omega \nabla\phi_j \cdot \nabla\phi_i = \sum_{j=1}^{n} \mathbf{u}_j \left\{ \sum_{\triangle_k \in \mathcal{T}_h} \int_{\triangle_k} \nabla\phi_j \cdot \nabla\phi_i \right\}. \tag{1.29}$$

Notice that when forming the sum over the elements in (1.29), we need only take account of those elements where the basis functions ϕ_i and ϕ_j are both nonzero. This means that entries a_{ij} and f_i in the Galerkin system (1.21) can be computed by calculating contributions from each of the elements, and then gathering (or *assembling*) them together.

If the kth element has n_k local degrees of freedom, then there are n_k basis functions that are not identically zero on the element. For example, in the case of a mesh made up entirely of $\boldsymbol{P}_1$ triangles, we have $n_k = 3$ for all elements, so that in each $\triangle_k$ there are three *element* basis functions associated with the restriction of three different *global* basis functions ϕ_j. In the case of a mesh containing a mixture of $\boldsymbol{Q}_2$ rectangles and $\boldsymbol{P}_2$ triangles, we have $n_k = 9$ if element k is a rectangle and $n_k = 6$ otherwise. In all cases, the local functions form an (element) basis set

$$\Xi_k := \{\psi_{k,1}, \psi_{k,2}, \ldots, \psi_{k,n_k}\}, \tag{1.30}$$

so that the solution within the element takes the form

$$u_h \mid_{\triangle_k} := u_h^{\textcircled{k}} = \sum_{i=1}^{n_k} \mathbf{u}_i^{\textcircled{k}} \psi_{k,i}. \tag{1.31}$$

Using triangular elements, for example, and localizing (1.22) and (1.23), we need to compute a set of $n_k \times n_k$ element matrices $A^{\textcircled{k}}$ and a set of n_k-dimensional vectors $\mathbf{f}^{\textcircled{k}}$ such that

$$A^{\textcircled{k}} = [a_{ij}^{\textcircled{k}}], \quad a_{ij}^{\textcircled{k}} = \int_{\triangle_k} \nabla\psi_{k,i} \cdot \nabla\psi_{k,j}, \tag{1.32}$$

$$\mathbf{f}^{\textcircled{k}} = [\boldsymbol{f}_i^{\textcircled{k}}], \quad \boldsymbol{f}_i^{\textcircled{k}} = \int_{\triangle_k} f\,\psi_{k,i} + \int_{\partial\Omega_N \cap \partial\triangle_k} g_N\,\psi_{k,i}. \tag{1.33}$$

The matrix $A^{\textcircled{k}}$ is referred to as the *element stiffness matrix* (or *local stiffness matrix*) associated with element $\triangle_k$. Its construction for the cases of triangular and quadrilateral elements is addressed in Sections 1.4.1 and 1.4.2, respectively. (A completely analogous construction is required for $\mathbb{R}^3$; see Hughes [132, Chapter 3].) Notice that for computational convenience, the essential boundary condition has not been enforced in (1.33). This is the standard implementation; essential conditions are usually imposed after the assembly of the element contributions into the Galerkin matrix has been completed. We will return to this point in the discussion of the assembly process in Section 1.4.3.

1.4.1 *Triangular element matrices*

The first stage in the computation of the element stiffness matrix $A^{\textcircled{k}}$ is to map from a reference element $\triangle_*$ onto the given element $\triangle_k$, as illustrated in

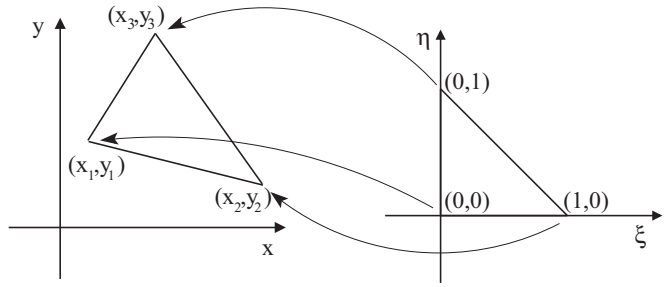


FIG. 1.14. Isoparametric mapping of a $\boldsymbol{P}_1$ element.

Figure 1.14. For straight-sided triangles the local–global mapping is defined for all points $(x, y) \in \triangle_k$ and is given by

$$x(\xi,\eta) = x_1\chi_1(\xi,\eta) + x_2\chi_2(\xi,\eta) + x_3\chi_3(\xi,\eta), \tag{1.34}$$
$$y(\xi,\eta) = y_1\chi_1(\xi,\eta) + y_2\chi_2(\xi,\eta) + y_3\chi_3(\xi,\eta), \tag{1.35}$$

where

$$\begin{aligned} \chi_1(\xi,\eta) &= 1 - \xi - \eta, \\ \chi_2(\xi,\eta) &= \xi, \\ \chi_3(\xi,\eta) &= \eta \end{aligned} \tag{1.36}$$

are the $\boldsymbol{P}_1$ basis functions defined on the reference element. We note in passing that elements with curved sides can be generated using the analogous mapping defined by the $\boldsymbol{P}_2$ reference element basis functions illustrated in Figure 1.7.

Clearly, the map from the reference element onto $\triangle_k$ is (and has to be) differentiable. Thus, given a differentiable function $\varphi(\xi, \eta)$, we can transform derivatives via

$$\begin{bmatrix} \dfrac{\partial\varphi}{\partial\xi} \\ \dfrac{\partial\varphi}{\partial\eta} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial\xi} & \dfrac{\partial y}{\partial\xi} \\ \dfrac{\partial x}{\partial\eta} & \dfrac{\partial y}{\partial\eta} \end{bmatrix} \begin{bmatrix} \dfrac{\partial\varphi}{\partial x} \\ \dfrac{\partial\varphi}{\partial y} \end{bmatrix}. \tag{1.37}$$

The *Jacobian matrix* in (1.37) may be simply calculated by substituting (1.36) into (1.34)–(1.35) and differentiating to give

$$J_k = \frac{\partial(x,y)}{\partial(\xi,\eta)} = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{bmatrix}. \tag{1.38}$$

Thus in this simple case, we see that J_k is a constant matrix over the reference element, and that the determinant

$$|J_k| = \begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix} = \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} = 2|\triangle_k| \tag{1.39}$$

is simply the ratio of the area of the mapped element $\triangle_k$ to that of the reference element $\triangle *$. The fact that $|J_k(\xi, \eta)| \neq 0$ for all points (ξ, η) in $\triangle *$ is very important; it ensures that the inverse mapping from $\triangle_k$ onto the reference element is uniquely defined and is differentiable. This means that the derivative transformation (1.37) can be inverted to give

$$\begin{bmatrix} \dfrac{\partial \varphi}{\partial x} \\ \dfrac{\partial \varphi}{\partial y} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \eta}{\partial x} \\ \dfrac{\partial \xi}{\partial y} & \dfrac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \varphi}{\partial \xi} \\ \dfrac{\partial \varphi}{\partial \eta} \end{bmatrix}. \tag{1.40}$$

Thus we see that derivatives of functions defined on $\triangle_k$ satisfy

$$\begin{aligned} \frac{\partial \xi}{\partial x} &= \frac{1}{|J_k|} \frac{\partial y}{\partial \eta}, & \frac{\partial \eta}{\partial x} &= -\frac{1}{|J_k|} \frac{\partial y}{\partial \xi}, \\ \frac{\partial \xi}{\partial y} &= -\frac{1}{|J_k|} \frac{\partial x}{\partial \eta}, & \frac{\partial \eta}{\partial y} &= \frac{1}{|J_k|} \frac{\partial x}{\partial \xi}. \end{aligned} \tag{1.41}$$

Given the basis functions on the master element $\psi_{*,i},\ i = 1, \ldots, n_k$ (see, for example, Problem 1.6), the $\boldsymbol{P}_m$ element stiffness matrix $A^{\textcircled{k}}$ in (1.32) is easily computed:

$$\begin{aligned} a_{ij}^{\textcircled{k}} &= \int_{\triangle_k} \frac{\partial \psi_{k,i}}{\partial x} \frac{\partial \psi_{k,j}}{\partial x} + \frac{\partial \psi_{k,i}}{\partial y} \frac{\partial \psi_{k,j}}{\partial y} \mathrm{d}x\, \mathrm{d}y, \\ &= \int_{\triangle *} \left\{ \frac{\partial \psi_{*,i}}{\partial x} \frac{\partial \psi_{*,j}}{\partial x} + \frac{\partial \psi_{*,i}}{\partial y} \frac{\partial \psi_{*,j}}{\partial y} \right\} |J_k|\, \mathrm{d}\xi\, \mathrm{d}\eta, \quad i, j = 1, \ldots, n_k. \end{aligned} \tag{1.42}$$

In the specific case of the linear mapping given by (1.36), it is convenient to define the coefficients

$$\begin{aligned} b_1 &= y_2 - y_3, \quad b_2 = y_3 - y_1, \quad b_3 = y_1 - y_2, \\ c_1 &= x_3 - x_2, \quad c_2 = x_1 - x_3, \quad c_3 = x_2 - x_1; \end{aligned} \tag{1.43}$$

in which case (1.38)–(1.41) implies that

$$\begin{bmatrix} \dfrac{\partial \varphi}{\partial x} \\ \dfrac{\partial \varphi}{\partial y} \end{bmatrix} = \frac{1}{2|\triangle_k|} \begin{bmatrix} b_2 & b_3 \\ c_2 & c_3 \end{bmatrix} \begin{bmatrix} \dfrac{\partial \varphi}{\partial \xi} \\ \dfrac{\partial \varphi}{\partial \eta} \end{bmatrix}. \tag{1.44}$$

Combining (1.44) with (1.42) gives the general form of the stiffness matrix expressed in terms of the local derivatives of the element basis functions:

$$a_{ij}^{\textcircled{k}} = \int_{\triangle *} \left(b_2 \frac{\partial \psi_{*,i}}{\partial \xi} + b_3 \frac{\partial \psi_{*,i}}{\partial \eta} \right) \left(b_2 \frac{\partial \psi_{*,j}}{\partial \xi} + b_3 \frac{\partial \psi_{*,j}}{\partial \eta} \right) \frac{1}{|J_k|} \mathrm{d}\xi \, \mathrm{d}\eta + \int_{\triangle *} \left(c_2 \frac{\partial \psi_{*,i}}{\partial \xi} + c_3 \frac{\partial \psi_{*,i}}{\partial \eta} \right) \left(c_2 \frac{\partial \psi_{*,j}}{\partial \xi} + c_3 \frac{\partial \psi_{*,j}}{\partial \eta} \right) \frac{1}{|J_k|} \mathrm{d}\xi \, \mathrm{d}\eta. \tag{1.45}$$

With the simplest linear approximation, that is, $\psi_{*,i} = \chi_i$ (see (1.36)), the local derivatives $\partial \psi_{*,i}/\partial \xi$, $\partial \psi_{*,i}/\partial \eta$ are constant, so the local stiffness matrix is trivial to compute (see Problem 1.9).

From a practical perspective, the simplest way of effecting the local–global transformation given by (1.36) is to define local element functions using *triangular* or *barycentric* coordinates (see Problem 1.3).

1.4.2 *Quadrilateral element matrices*

In the case of quadrilateral elements (and rectangular elements in particular), the stiffness matrix $A^{\textcircled{k}}$ is typically computed by mapping as in Figure 1.10 from a reference element $\square_*$ onto the given element $\square_k$, and then using quadrature. For quadrilaterals, the local–global mapping is defined for all points $(x, y) \in \square_k$ and is given by

$$x(\xi, \eta) = x_1 \chi_1(\xi, \eta) + x_2 \chi_2(\xi, \eta) + x_3 \chi_3(\xi, \eta) + x_4 \chi_4(\xi, \eta), \tag{1.46}$$

$$y(\xi, \eta) = y_1 \chi_1(\xi, \eta) + y_2 \chi_2(\xi, \eta) + y_3 \chi_3(\xi, \eta) + y_4 \chi_4(\xi, \eta), \tag{1.47}$$

where

$$\begin{aligned} \chi_1(\xi, \eta) &= (\xi - 1)(\eta - 1)/4, \\ \chi_2(\xi, \eta) &= -(\xi + 1)(\eta - 1)/4, \\ \chi_3(\xi, \eta) &= (\xi + 1)(\eta + 1)/4, \\ \chi_4(\xi, \eta) &= -(\xi - 1)(\eta + 1)/4 \end{aligned}$$

are the $\boldsymbol{Q}_1$ basis functions defined on the reference element (see Figure 1.10).

The map from the reference element onto $\square_k$ is differentiable, and derivatives are defined via (1.37), as in the triangular case. The big difference here is that the entries in the Jacobian matrix are *linear* functions of the coordinates (ξ, η) (cf. (1.38)):

$$J_k = \frac{\partial (x, y)}{\partial (\xi, \eta)} = \begin{bmatrix} \sum_{j=1}^4 x_j \dfrac{\partial \chi_j}{\partial \xi} & \sum_{j=1}^4 y_j \dfrac{\partial \chi_j}{\partial \xi} \\ \sum_{j=1}^4 x_j \dfrac{\partial \chi_j}{\partial \eta} & \sum_{j=1}^4 y_j \dfrac{\partial \chi_j}{\partial \eta} \end{bmatrix}. \tag{1.48}$$

Note that the determinant $|J_k|$ is always a linear function of the coordinates; see Problem 1.11. In simple terms, the mapped element must have straight edges. If

the mapped element $\square_k$ is a parallelogram then the Jacobian turns out to be a constant matrix.

A sufficient condition for a well-defined inverse mapping ($|J_k(\xi,\eta)| > 0$ for all points $(\xi,\eta) \in \square_*$) is that the mapped element be convex. In this case, derivatives on $\square_k$ can be computed[3] using (1.40), with

$$\begin{aligned} \frac{\partial \xi}{\partial x} &= \frac{1}{|J_k|}\sum_{j=1}^{4} y_j \frac{\partial \chi_j}{\partial \eta}, \qquad \frac{\partial \eta}{\partial x} = -\frac{1}{|J_k|}\sum_{j=1}^{4} y_j \frac{\partial \chi_j}{\partial \xi}, \\ \frac{\partial \xi}{\partial y} &= -\frac{1}{|J_k|}\sum_{j=1}^{4} x_j \frac{\partial \chi_j}{\partial \eta}, \quad \frac{\partial \eta}{\partial y} = \frac{1}{|J_k|}\sum_{j=1}^{4} x_j \frac{\partial \chi_j}{\partial \xi}, \end{aligned} \tag{1.49}$$

and the $\boldsymbol{Q}_m$ element stiffness matrix is computed via

$$a_{ij}^{\textcircled{k}} = \int_{\square_*} \left\{ \frac{\partial \psi_{*,i}}{\partial x}\frac{\partial \psi_{*,j}}{\partial x} + \frac{\partial \psi_{*,i}}{\partial y}\frac{\partial \psi_{*,j}}{\partial y} \right\} |J_k| \, \mathrm{d}\xi \, \mathrm{d}\eta. \tag{1.50}$$

Note that if general quadrilateral elements are used then the integrals in (1.50) involve rational functions of polynomials.

Gauss quadrature is almost always used to evaluate the definite integrals that arise in the calculation of the element matrices $A^{\textcircled{k}}$ and the vectors $\mathbf{f}^{\textcircled{k}}$. Quadrilateral elements are particularly amenable to quadrature because integration rules can be constructed by taking tensor products of the standard one-dimensional Gauss rules. This is the approach adopted in the IFISS software. The definite integral (1.50) is approximated by the summation

$$\bar{a}_{ij}^{\textcircled{k}} = \sum_{s=1}^{m}\sum_{t=1}^{m} w_{st} \big|J_k(\xi_s,\eta_t)\big| \left\{ \frac{\partial \psi_{*,i}}{\partial x}\frac{\partial \psi_{*,j}}{\partial x} + \frac{\partial \psi_{*,i}}{\partial y}\frac{\partial \psi_{*,j}}{\partial y} \right\} \Bigg|_{(\xi_s,\eta_t)},$$

where the quadrature points (ξ_s,η_t) are those associated with one of the Gauss tensor-product hierarchy illustrated in Figure 1.15. The quadrature weights w_{st} are computed by taking the tensor product of the weights associated with the classical one-dimensional rule; see Hughes [132, pp. 141–145] for further details.

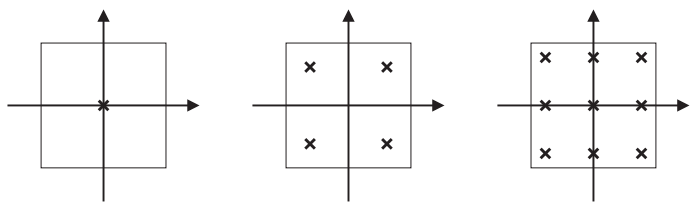


FIG. 1.15. Sampling points for 1×1, 2×2 and 3×3 Gauss quadrature rules.

[3]Using `deriv.m` and `qderiv.m` in IFISS.

In one dimension, all polynomials of degree $2m-1$ can be integrated exactly using the classical m point Gauss rule. This is *optimal* in the sense that any rule with m points has precisely $2m$ free parameters (namely, the weights and positions of the quadrature points). Although the tensor-product rules are not optimal in this sense, the $m \times m$ rule does have the nice property that it exactly integrates all $\boldsymbol{Q}_{2m-1}$ functions. This means that in the case of grids of rectangular (or more generally parallelogram) elements, the bilinear element matrix $A^{\textcircled{k}}$ can be exactly computed using the 2×2 rule; see Problem 1.16. Similarly, the biquadratic element matrix $A^{\textcircled{k}}$ can be exactly integrated (for a rectangular element) if the 3×3 rule is used.

The element source vector (1.33) is also typically computed using quadrature. For example, the interior contribution to the source vector (1.33),

$$\boldsymbol{f}_i^{\textcircled{k}} = \int_{\square *} f\psi_{*,i}\,|J_k|\mathrm{d}\xi\,\mathrm{d}\eta, \tag{1.51}$$

can be approximated via[4]

$$\bar{\boldsymbol{f}}_i^{\textcircled{k}} = \sum_{s=1}^{m}\sum_{t=1}^{m} w_{st} f(\xi_s,\eta_t)\,\psi_{*,i}(\xi_s,\eta_t)\,\big|J_k(\xi_s,\eta_t)\big|. \tag{1.52}$$

The 2×2 rule would generally be used in the case of bilinear approximation, and the 3×3 rule if the approximation is biquadratic. Gauss integration rules designed for triangular elements are tabulated in [132, pp. 173–174].

1.4.3 *Assembly of the Galerkin system*

The assembly of the element contributions $A^{\textcircled{k}}$ and $\mathbf{f}^{\textcircled{k}}$ into the Galerkin system is a reversal of the localization process illustrated in Figure 1.16.

The main computational issue is the need for careful bookkeeping to ensure that the element contributions are added into the correct locations in the coefficient matrix A and the vector $\mathbf{f}$. The simplest way of implementing the process is to represent the mapping between local and global entities using

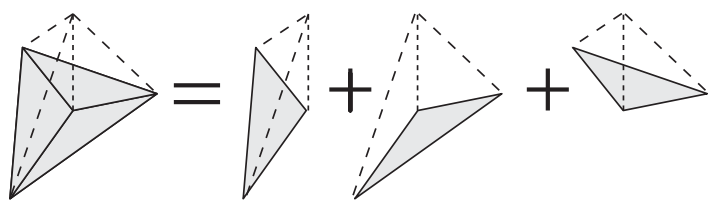


FIG. 1.16. Assembly of a $\boldsymbol{P}_1$ global basis function from component element functions.

[4]Using `gauss_source.m` in IFISS.

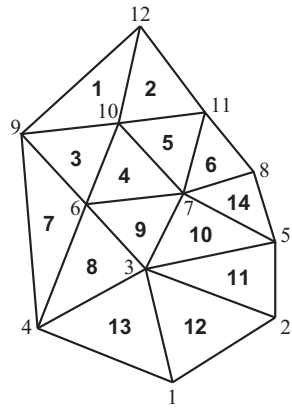


FIG. 1.17. Nodal and element numbering for the mesh in Figure 1.5.

a connectivity matrix. For example, in the case of the mesh of $\boldsymbol{P}_1$ triangles illustrated in Figure 1.17 we introduce the connectivity matrix defined by

$$\mathtt{P}^T = \begin{array}{c} \begin{matrix} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} & \mathbf{7} & \mathbf{8} & \mathbf{9} & \mathbf{10} & \mathbf{11} & \mathbf{12} & \mathbf{13} & \mathbf{14} \end{matrix} \\ \begin{bmatrix} 9 & 12 & 9 & 6 & 10 & 11 & 4 & 4 & 6 & 5 & 5 & 2 & 1 & 8 \\ 10 & 10 & 6 & 7 & 7 & 7 & 6 & 3 & 3 & 7 & 3 & 3 & 3 & 7 \\ 12 & 11 & 10 & 10 & 11 & 8 & 9 & 6 & 7 & 3 & 2 & 1 & 4 & 5 \end{bmatrix} \end{array},$$

so that the index $\mathtt{j} = \mathtt{P}(\mathtt{k}, \mathtt{i})$ specifies the global node number of local node `i` in element `k`, and thus identifies the coefficient $\mathbf{u}_i^{(k)}$ in (1.31) with the global coefficient $\mathbf{u}_j$ in the expansion (1.18) of u_h. Given `P`, the matrices $A^{(k)}$ and vectors $\mathbf{f}^{(k)}$ for the mesh in Figure 1.17 can be assembled into the Galerkin system matrix and vector using a set of nested loops.

```
k = 1:14
  j = 1:3
    i = 1:3
      Agal(P(k,i),P(k,j)) = Agal(P(k,i),P(k,j)) + A(k,i,j)
    endloop i
    fgal(P(k,j)) = fgal(P(k,j)) + f(k,j)
  endloop j
endloop k
```

A few observations are appropriate here. First, in a practical implementation, the Galerkin matrix `Agal` will be stored in an appropriate sparse format—it

must be initialized to zero! Second, it should be apparent that as the elements are assembled in order, then for any node s say, a stage will be reached when subsequent assemblies do not affect node s (that is, the sth row and column of the Galerkin matrix). When this stage is reached the variable is said to be fully summed; for example, variable 6 is fully summed after assembly of element **9**. This observation motivates the development of specialized direct solvers (known as *frontal solvers*) whereby the assembly process is intertwined with Gaussian elimination. In essence, as soon as a variable becomes fully summed, row operations can be performed to make entries below the diagonal zero and the modified row can then be saved for subsequent back-substitution; see for example Johnson [138, pp. 117–120].

It should also be emphasized that the intuitive element-by-element assembly embodied in the loop structure above is likely to be very inefficient; the inner loop involves indirect addressing and is too short to allow effective vectorization. The best way of generating efficient finite element code[5] is to work with blocks of elements and to reorder the loops so that the element loop `k` is the innermost. For real efficiency, the number of elements in a block should be set so that all required data can fit into cache memory.

We now turn our attention to the imposition of essential boundary conditions on the assembled Galerkin system (1.21). We assume here that the basis functions are of Lagrangian type, that is, each basis function ϕ_j has a node x_j in $\overline{\Omega}$ associated with it such that

$$\phi_j(x_j) = 1, \quad \phi_j(x_i) = 0 \qquad \text{for all nodes } x_i \neq x_j.$$

This property is depicted for the $\boldsymbol{P}_2$ basis functions in Figure 1.7. It follows from this assumption that for x_j on $\partial\Omega_D$, $u_h(x_j) = \mathbf{u}_j$, where the required value of $\mathbf{u}_j$ is interpolated from the Dirichlet boundary data. See Ciarlet [53, Section 2.2] for treatment of more general basis functions.

Now consider how to impose this condition at node 5 of the mesh in Figure 1.17. Suppose that a preliminary version of the Galerkin matrix A is constructed via (1.22) for $1 \leq i, j \leq n + n_\partial$, and that in addition, all the contributions $\int_\Omega \phi_i f$ have been assembled into the right-hand-side vector $\mathbf{f}$. There are then two things needed to specify the system (1.21) via (1.20) and (1.23): the given value of $\mathbf{u}_5$ must be included in the definition of the vector $\mathbf{f}$ of (1.23), and the fifth row and column of the preliminary Galerkin matrix must be deleted (since ϕ_5 is being removed from the space of test functions). The first step can be achieved by multiplying the fifth column of A by the specified boundary value $\mathbf{u}_5$ and then subtracting the result from $\mathbf{f}$. An alternative technique[6] is to retain the imposed degree of freedom in the Galerkin system by modifying the row and column (5, here) of the Galerkin matrix corresponding to the boundary node so that the diagonal value is unity and the off-diagonal entries are set to zero,

[5]Embodied in the IFISS routines `femq1_diff.m` and `femq2_diff.m`.

[6]Embodied in the IFISS routine `nonzerobc.m`.

and then setting the corresponding value of $\mathbf{f}$ to the boundary value $\mathbf{u}_5$. Notice that the modified Galerkin matrix thus has a multiple eigenvalue of unity, with multiplicity equal to the number of nodes on the Dirichlet part of the boundary.

Finally we remark that it is easiest to treat any nonzero Neumann boundary conditions in the system (1.21) after the assembly process and the imposition of essential boundary conditions has been completed. At this stage, the boundary contribution in (1.23) can be assembled by running through the boundary edges on $\partial\Omega_N$ and evaluating the component edge contributions using standard (one-dimensional) Gauss quadrature.

1.5 Theory of errors

Our starting point is the generic problem (1.13)–(1.14). The associated weak formulation is the following: find $u \in \mathcal{H}^1_E$ such that

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega v f + \int_{\partial\Omega} v\, g_N \quad \text{for all } v \in \mathcal{H}^1_{E_0}, \tag{1.53}$$

with spaces $\mathcal{H}^1_E$ and $\mathcal{H}^1_{E_0}$ given by (1.15) and (1.16), respectively.

To simplify the notation, we follow the established convention of not distinguishing between scalar-valued functions (for example, $u : \Omega \to \mathbb{R}$) and vector-valued functions (for example, $\vec{u} : \Omega \to \mathbb{R}^d$) as long as there is no ambiguity. In general, a bold typeface is used to represent a space of vector-valued functions, and norms and inner products are to be interpreted componentwise.

Definition 1.1 ($\boldsymbol{L}_2$ inner product and norm). Let $L_2(\Omega)$ denote the space of square-integrable scalar-valued functions defined on Ω (see (1.11)), with associated inner product $(\cdot, \cdot)$. The space $\boldsymbol{L}_2(\Omega)$ of square-integrable vector-valued functions defined on Ω consists of functions with each component in $L_2(\Omega)$, and has inner product

$$(\vec{u}, \vec{v}\,) := \int_\Omega \vec{u} \cdot \vec{v}$$

and norm

$$\|\vec{u}\,\| := (\vec{u}, \vec{u}\,)^{1/2}.$$

For example, for two-dimensional vectors $\vec{u} = (u_x, u_y)$ and $\vec{v} = (v_x, v_y)$, $(\vec{u}, \vec{v}) = (u_x, v_x) + (u_y, v_y)$ and $\|\vec{u}\,\|^2 = \|u_x\|^2 + \|u_y\|^2$.

Our first task is to establish that a weak solution is uniquely defined. To this end, we assume two weak solutions satisfying (1.53), $u_1 \in \mathcal{H}^1_E$ and $u_2 \in \mathcal{H}^1_E$ say, and then try to establish that $u_1 = u_2$ everywhere. Subtracting the two variational equations shows that $u_1 - u_2 \in \mathcal{H}^1_{E_0}$ satisfies the equation

$$\int_\Omega \nabla(u_1 - u_2) \cdot \nabla v = 0 \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.54}$$

Substituting $v = u_1 - u_2$ then shows that $\|\nabla(u_1 - u_2)\| = 0$, and this implies that $u_1 - u_2$ is a constant function. To make progress, the case of a pure Neumann problem needs to be excluded. We can then use the additional fact that $u_1 = u_2$ on the Dirichlet part of the boundary. The following lemma holds the key to this.

Lemma 1.2 (Poincaré–Friedrichs inequality). *Assume that $\Omega \subset \mathbb{R}^2$ is contained in a square with side length L (and, in the case $\int_{\partial\Omega_N} \mathrm{d}s \neq 0$, that it has a sufficiently smooth boundary). Given that $\int_{\partial\Omega_D} \mathrm{d}s \neq 0$, it follows that*

$$\|v\| \leq L\,\|\nabla v\| \quad \text{for all } v \in \mathcal{H}^1_{E_0}.$$

The constant L is called the Poincaré constant.

This inequality is discussed in many texts on finite element error analysis, for example, Braess [26, pp. 30–31] and Brenner & Scott [36, pp. 128–130]. Establishing the inequality in the simplest case of a square domain Ω is a worthy exercise; see Problem 1.19. Making the choice $v = u_1 - u_2$ in Lemma 1.2 implies that the solution is unique.

Returning to (1.53), we identify the left-hand side with the bilinear form $a : \mathcal{H}^1(\Omega) \times \mathcal{H}^1(\Omega) \to \mathbb{R}$, and the right-hand side with the linear functional $\ell : \mathcal{H}^1(\Omega) \to \mathbb{R}$, so that

$$a(u, v) := (\nabla u, \nabla v), \qquad \ell(v) := (f, v) + (g_N, v)_{\partial\Omega_N}, \tag{1.55}$$

and restate the problem as

find $u \in \mathcal{H}^1_E$ such that
$$a(u, v) = \ell(v) \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.56}$$

The corresponding discrete problem (1.19) is then given by

find $u_h \in S^h_E$ such that
$$a(u_h, v_h) = \ell(v_h) \quad \text{for all } v_h \in S^h_0. \tag{1.57}$$

Assuming that the approximation is conforming, $S^h_E \subset \mathcal{H}^1_E$ and $S^h_0 \subset \mathcal{H}^1_{E_0}$, our task here is to estimate the quality of the approximation $u_h \approx u$.

We will outline the conventional (a priori) analysis of the approximation error arising using finite element approximation spaces in (1.57) in Section 1.5.1. Such error bounds are asymptotic in nature, and since they involve the true solution u they are not readily computable. We go on to discuss computable error bounds (usually referred to as a posteriori estimates) in Section 1.5.2.

1.5.1 *A priori error bounds*

To get a handle on the error, we can simply pick a generic $v \in \mathcal{H}^1_{E_0}$ and subtract $a(u_h, v)$ from (1.56) to give

$$a(u, v) - a(u_h, v) = \ell(v) - a(u_h, v).$$

This is the basic equation for the error: our assumption that $S^h_E \subset \mathcal{H}^1_E$ implies[7] that $e = u - u_h \in \mathcal{H}^1_{E_0}$ and satisfies

$$a(e, v) = \ell(v) - a(u_h, v) \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.58}$$

Note that $e \in \mathcal{H}^1_{E_0}$ since $S^h_E \subset \mathcal{H}^1_E$.

We now make explicit use of the fact that the underlying bilinear form $a(\cdot, \cdot)$ defines an inner product over the space $\mathcal{H}^1_{E_0} \times \mathcal{H}^1_{E_0}$, with an associated (*energy*) norm $\|\nabla u\| = a(u, u)^{1/2}$; see Problem 1.17. The starting point is the *Galerkin orthogonality property*: taking $v_h \in S^h_0$ in (1.58) and using (1.57) we have

$$a(u - u_h, v_h) = 0 \quad \text{for all } v_h \in S^h_0. \tag{1.59}$$

In simple terms, the error $e \in \mathcal{H}^1_{E_0}$ is orthogonal to the subspace S^h_0, with respect to the energy inner product. An immediate consequence of (1.59) is the *best approximation property* established below.

Theorem 1.3. $\|\nabla u - \nabla u_h\| = \min\{\|\nabla u - \nabla v_h\|\colon v_h \in S^h_E\}$.

Proof Let $v_h \in S^h_E$, and note that $u - u_h \in \mathcal{H}^1_{E_0}$; so by definition,

$$\begin{aligned}
\|\nabla(u - u_h)\|^2 &= a(u - u_h, u - u_h) \\
&= a(u - u_h, u - v_h + v_h - u_h) \\
&= a(u - u_h, u - v_h) + a(u - u_h, v_h - u_h) \\
&= a(u - u_h, u - v_h) \quad \text{(using Galerkin orthogonality)} \\
&\le \|\nabla(u - u_h)\|\, \|\nabla(u - v_h)\| \quad \text{(using Cauchy–Schwarz).}
\end{aligned}$$

Hence, for either $\|\nabla(u - u_h)\| = 0$ or $\|\nabla(u - u_h)\| \neq 0$, we have

$$\|\nabla(u - u_h)\| \le \|\nabla(u - v_h)\| \quad \text{for all } v_h \in S^h_E. \tag{1.60}$$

Notice that the minimum is achieved since $u_h \in S^h_E$. □

The energy error bound (1.60) is appealingly simple, and moreover in the case $g_D = 0$ it leads to the useful characterization that

$$\|\nabla(u - u_h)\|^2 = \|\nabla u\|^2 - \|\nabla u_h\|^2\,; \tag{1.61}$$

[7] Recall that in practice, the essential boundary condition is interpolated (see (1.18)) so that $u_h \neq g_D$ on $\partial\Omega_D$ whenever the boundary data g_D is not a polynomial. In such cases the error $u - u_h$ must be estimated using a more sophisticated *nonconforming* analysis; see Brenner & Scott [36, pp. 195ff.] for details.

see Problem 1.18. (If u is known analytically, then (1.61) can be used to calculate the error in the energy norm without using elementwise integration.) The synergy between Galerkin orthogonality (1.59) and the best approximation (1.60) is a reflection of the fact that u_h is the *projection* of u into the space S_E^h. This property will be exploited in Chapter 2, where fast solution algorithms are developed for the discrete problem (1.57).

The remaining challenge is to derive bounds on the error $u - u_h$ with respect to other norms, in particular, that associated with the *Hilbert space*[8] $\mathcal{H}^1(\Omega)$ introduced in Section 1.2. A formal definition is the following.

Definition 1.4 ($\mathcal{H}^1(\Omega)$ norm). Let $\mathcal{H}^1(\Omega)$ denote the set of functions u in $L_2(\Omega)$ possessing generalized[9] first derivatives. An inner product on $\mathcal{H}^1(\Omega)$ is given by

$$(u, v)_{1,\Omega} := (u, v) + (\nabla u, \nabla v), \tag{1.62}$$

and this induces the associated norm

$$\|u\|_{1,\Omega} := (\|u\|^2 + \left\|D^1 u\right\|^2)^{1/2}, \tag{1.63}$$

where $D^1 u$ denotes the sum of squares of the first derivatives; for a two-dimensional domain Ω,

$$\left\|D^1 u\right\|^2 := \int_\Omega \left(\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 \right).$$

An important property of functions v in $\mathcal{H}^1(\Omega)$ is that they have a well-defined restriction to the boundary $\partial\Omega$. (This is an issue because functions in $\mathcal{H}^1(\Omega)$ need not be continuous.) The theoretical basis for this assertion is the following lemma.

Lemma 1.5 (Trace inequality). *Given a bounded domain Ω with a sufficiently smooth (for example, polygonal) boundary $\partial\Omega$, a constant $C_{\partial\Omega}$ exists such that*

$$\|v\|_{\partial\Omega} \leq C_{\partial\Omega} \|v\|_{1,\Omega} \quad \textit{for all } v \in \mathcal{H}^1(\Omega).$$

Notice that, in contrast, there is no constant C such that $\|v\|_{\partial\Omega} \leq C \|v\|$ for every v in $L_2(\Omega)$, hence associating boundary values with $L_2(\Omega)$ functions is not meaningful. The proof of Lemma 1.5 is omitted; for details see Braess [26, pp. 48ff.]. Applications of the trace inequality will be found in later sections.

Extending the energy error estimate of Theorem 1.3 to a general error bound in $\mathcal{H}^1(\Omega)$ is a simple consequence of the Poincaré–Friedrichs inequality.

[8] This means a vector space with an inner product, which contains the limits of every Cauchy sequence that is defined with respect to the norm $\|\cdot\|_{1,\Omega}$.

[9] This includes functions like $|x|$ that are differentiable except at a finite number of points. To keep the exposition simple, we omit a formal definition; for details see Braess [26, p. 28] or Brenner & Scott [36, pp. 24–27].

Proposition 1.6. *Let Ω satisfy the assumptions in Lemma 1.2. Then there is a constant C_Ω, independent of v, such that*

$$\|\nabla v\| \leq \|v\|_{1,\Omega} \leq C_\Omega \|\nabla v\| \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.64}$$

Proof See Problem 1.20. □

We are now ready to state a *quasi-optimal* error bound that reflects the fact that $\|u - u_h\|_{1,\Omega}$ is proportional to the best possible approximation from the space S^h_E.

Theorem 1.7. *Let Ω satisfy the assumptions in Lemma 1.2. Then*

$$\|u - u_h\|_{1,\Omega} \leq C_\Omega \min_{v_h \in S^h_E} \|u - v_h\|_{1,\Omega}. \tag{1.65}$$

Proof Note that $u - v_h \in \mathcal{H}^1_{E_0}$ if $v_h \in S^h_E$. Combining (1.60) with (1.64) then gives (1.65). □

The best approximation error bound (1.60) is quite general in the sense that it is valid for any problem where the bilinear form $a(\cdot,\cdot)$ in (1.56) defines an inner product over the test space $\mathcal{H}^1_{E_0}$. However, the line of analysis above is not valid if the bilinear form $a(\cdot,\cdot)$ in the variational formulation is not symmetric, as, for example, for the convection–diffusion equation; see Chapter 6. In such cases, a priori error bounds in the underlying function space must be established using a different theoretical argument—typically using the coercivity and continuity of the underlying bilinear form over the space $\mathcal{H}^1_{E_0}$; see Problem 1.21. Further details are given in Chapter 6.

We now develop the general error bound (1.65) in the case of the finite element approximation spaces that were introduced in Sections 1.3.1 and 1.3.2. We first consider the simplest case of triangular elements using $\boldsymbol{P}_1$ (piecewise linear) approximation. That is, given a partitioning of the domain $\mathcal{T}_h$ consisting of triangular elements $\triangle_k$ we make the specific choice $S^h_0 = X^1_h$, where

$$X^1_h := \{v \in C^0(\Omega), v = 0 \text{ on } \partial\Omega_D;\ v|_\triangle \in \boldsymbol{P}_1 \text{ for all } \triangle \in \mathcal{T}_h\}. \tag{1.66}$$

We will state the error bound in the form of a theorem. We also need a couple of preliminary definitions.

Definition 1.8 ($\mathcal{H}^2(\Omega)$ norm). The set of functions $u \in \mathcal{H}^1(\Omega)$ that also possess generalized second derivatives can be identified with the Sobolev space $\mathcal{H}^2(\Omega)$. More precisely, $\mathcal{H}^2(\Omega) \subset \mathcal{H}^1(\Omega)$ is a Hilbert space that is complete with respect to the norm

$$\|u\|_{2,\Omega} := \left(\|u\|^2_{1,\Omega} + \|D^2 u\|^2\right)^{1/2},$$

where D^2u denotes the sum of squares of second derivatives. More specifically, in the case of a two-dimensional domain Ω,

$$\left\|D^2u\right\|^2 := \int_\Omega \left(\left(\frac{\partial^2 u}{\partial x^2}\right)^2 + \left(\frac{\partial^2 u}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 u}{\partial y^2}\right)^2 \right).$$

Definition 1.9 ($\mathcal{H}^2$ regularity). The variational problem (1.56) is said to be $\mathcal{H}^2$ *regular* if there exists a constant C_Ω such that for every $f \in L_2(\Omega)$, there is a solution $u \in \mathcal{H}^1_E$ that is in the space $\mathcal{H}^2(\Omega)$, and satisfies the bound

$$\|u\|_{2,\Omega} \leq C_\Omega \, \|f\|.$$

Theorem 1.10. *If the variational problem* (1.56) *is solved using a mesh of linear triangular elements, so that* $S^h_0 = X^1_h$ *in* (1.57)*, and if a minimal angle condition is satisfied (see Definition* 1.15*), then there exists a constant* C_1 *such that*

$$\|\nabla(u - u_h)\| \leq C_1 \, h \, \left\|D^2u\right\|, \tag{1.67}$$

where $\left\|D^2u\right\|$ *measures the* $\mathcal{H}^2$ *regularity of the target solution, and* h *is the length of the longest triangle edge in the mesh.*

Notice that if (1.56) is $\mathcal{H}^2$ regular, then (1.67) implies that the finite element solution u_h converges to the exact solution u in the limit $h \to 0$. The fact that the right-hand side of (1.67) is proportional to h is referred to as *first-order* (or *linear*) convergence. Furthermore, Proposition 1.6 implies that if Lemma 1.2 is valid, then the order of convergence in $\mathcal{H}^1$ is the same as the order of convergence in the energy norm.

The issue of $\mathcal{H}^2$ regularity is central to the proof of Theorem 1.10, the first step of which is to break the bound (1.60) into pieces by introducing an appropriate interpolant $\pi_h u$ from the approximation space S^h_E. Making the specific choice $v_h = \pi_h u$ in (1.60) then gives

$$\|\nabla(u - u_h)\| \leq \|\nabla(u - \pi_h u)\|. \tag{1.68}$$

What is important here[10] is that $u \in \mathcal{H}^2(\Omega) \subset C^0(\Omega)$, so the simple piecewise linear interpolant $\pi_h u$, satisfying $\pi_h u(\mathbf{x}_i) = u(\mathbf{x}_i)$ at every vertex $\mathbf{x}_i$ of the triangulation, is a well-defined function in S^h_E (since $\pi_h u \in X^1_h$ in the case of zero boundary data). The localization of the error is now immediate since (1.68) can be broken up into elementwise error bounds

$$\|\nabla(u - \pi_h u)\|^2 = \sum_{\triangle_k \in \mathcal{T}_h} \|\nabla(u - \pi_h u)\|^2_{\triangle_k}. \tag{1.69}$$

[10]The relationship between continuous functions and Sobolev spaces is dependent on the domain Ω; if Ω is one-dimensional then $\mathcal{H}^1(\Omega) \subset C^0(\Omega)$, but for two-dimensional domains functions exist that are not bounded (hence not continuous) yet still have square-integrable first derivatives; see Braess [26, pp. 31–32].

The problem of estimating the overall error is now reduced to one of approximation theory—we need good estimates for the interpolation error on a typical element.

It is at this point that the local–global mapping in Section 1.4 plays an important role. Rather than estimating the error for every individual element, the idea is to map the element interpolation error from (1.69) onto the reference element, since the error can easily be bounded there in terms of derivatives of the interpolated function. This type of construction is referred to as a *scaling argument*. Each of the three stages in the process is summarized below in the form of a lemma. Note that h_k denotes the length of the longest edge of $\triangle_k$, and $\bar{u}$ denotes the mapped function defined on the reference element $\triangle *$.

Lemma 1.11. $\|\nabla(u - \pi_h u)\|^2_{\triangle_k} \leq 2\dfrac{h_k^2}{|\triangle_k|} \|\nabla(\bar{u} - \pi_h \bar{u})\|^2_{\triangle *}$.

Proof Define $e_k = (u - \pi_h u)|_{\triangle_k}$ and let $\bar{e}_k$ denote the mapped function defined on $\triangle *$. By definition,

$$\begin{aligned} \|\nabla e_k\|^2_{\triangle_k} &= \int_{\triangle_k} \left(\frac{\partial e_k}{\partial x}\right)^2 + \left(\frac{\partial e_k}{\partial y}\right)^2 \mathrm{d}x\,\mathrm{d}y \\ &= \int_{\triangle *} \left(\left(\frac{\partial \bar{e}_k}{\partial x}\right)^2 + \left(\frac{\partial \bar{e}_k}{\partial y}\right)^2\right) 2|\triangle_k|\,\mathrm{d}\xi\,\mathrm{d}\eta, \end{aligned} \tag{1.70}$$

where the derivatives satisfy (1.44); in particular the first term is of the form

$$\left(\frac{\partial \bar{e}_k}{\partial x}\right)^2 = \frac{1}{4|\triangle_k|^2}\left(b_2 \frac{\partial \bar{e}_k}{\partial \xi} + b_3 \frac{\partial \bar{e}_k}{\partial \eta}\right)^2,$$

with b_2 and b_3 defined by (1.43). Using the facts that $(a+b)^2 \leq 2(a^2+b^2)$ and $|b_i| \leq h_k$, we get the bound

$$2|\triangle_k| \left(\frac{\partial \bar{e}_k}{\partial x}\right)^2 \leq \frac{h_k^2}{|\triangle_k|}\left(\left(\frac{\partial \bar{e}_k}{\partial \xi}\right)^2 + \left(\frac{\partial \bar{e}_k}{\partial \eta}\right)^2\right).$$

The second term in (1.70) can be bounded in exactly the same way ($|c_i| \leq h_k$). Summing the terms gives the stated result. □

The following bound is a special case of a general estimate for the interpolation error in Sobolev spaces, known as the *Bramble–Hilbert lemma*. In simple terms, the error due to linear interpolation in a unit triangle measured in the energy norm is bounded by the L_2 norm of the second derivative of the interpolation error.

Lemma 1.12.

$$\|\nabla(\bar{u} - \pi_h \bar{u})\|_{\triangle *} \leq C \left\|D^2(\bar{u} - \pi_h \bar{u})\right\|_{\triangle *} \equiv C \left\|D^2 \bar{u}\right\|_{\triangle *}. \tag{1.71}$$

While proving the analogous result in $\mathbb{R}^1$ is a straightforward exercise (see Problem 1.22), the proof of (1.71) is technical and so is omitted. An accessible discussion can be found in Braess [26, pp. 75–76], and a complete and rigorous treatment is given in Brenner & Scott [36, Chapter 4].

Lemma 1.13. $\left\|D^2\bar{u}\right\|^2_{\triangle *} \leq 18h_k^2 \dfrac{h_k^2}{|\triangle_k|} \left\|D^2u\right\|^2_{\triangle_k}$.

Proof By definition,

$$\begin{aligned}\left\|D^2\bar{u}\right\|^2_{\triangle *} &= \int_{\triangle *} \left(\frac{\partial^2\bar{u}}{\partial\xi^2}\right)^2 + \left(\frac{\partial^2\bar{u}}{\partial\xi\partial\eta}\right)^2 + \left(\frac{\partial^2\bar{u}}{\partial\eta^2}\right)^2 \mathrm{d}\xi\,\mathrm{d}\eta \\ &= \int_{\triangle_k} \left(\left(\frac{\partial^2 u}{\partial\xi^2}\right)^2 + \left(\frac{\partial^2 u}{\partial\xi\partial\eta}\right)^2 + \left(\frac{\partial^2 u}{\partial\eta^2}\right)^2\right) \frac{1}{2|\triangle_k|}\,\mathrm{d}x\,\mathrm{d}y, \qquad (1.72)\end{aligned}$$

where the derivatives are mapped using (1.37); in particular, the first term is of the form

$$\begin{aligned}\left(\frac{\partial}{\partial\xi}\left(\frac{\partial u}{\partial\xi}\right)\right)^2 &= \left(c_3\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial\xi}\right) - b_3\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial\xi}\right)\right)^2 \\ &= \left(c_3^2\frac{\partial^2 u}{\partial x^2} - 2c_3b_3\frac{\partial^2 u}{\partial x\partial y} + b_3^2\frac{\partial^2 u}{\partial y^2}\right)^2 \\ &\leq 3\left(c_3^4\left(\frac{\partial^2 u}{\partial x^2}\right)^2 + 4c_3^2b_3^2\left(\frac{\partial^2 u}{\partial x\partial y}\right)^2 + b_3^4\left(\frac{\partial^2 u}{\partial y^2}\right)^2\right) \\ &\leq 12h_k^4\left(\left(\frac{\partial^2 u}{\partial x^2}\right)^2 + \left(\frac{\partial^2 u}{\partial x\partial y}\right)^2 + \left(\frac{\partial^2 u}{\partial y^2}\right)^2\right). \qquad (1.73)\end{aligned}$$

The second and third terms in (1.72) can be bounded in exactly the same way. Summing the three terms gives the stated result. □

The bound in Lemma 1.11 (and that in Lemma 1.13) involves the triangle aspect ratio $h_k^2/|\triangle_k|$. Keeping the aspect ratio small is equivalent to a minimum angle condition, as is shown in the following; see Figure 1.18.

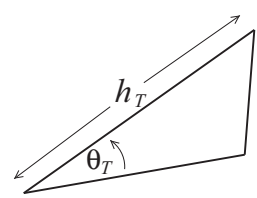


FIG. 1.18. Minimum angle condition.

Proposition 1.14. *Given any triangle, we have the equivalence relation*

$$\frac{h_T^2}{4}\sin\theta_T \le |\triangle_T| \le \frac{h_T^2}{2}\sin\theta_T, \tag{1.74}$$

where $0 < \theta_T \le \pi/3$ *is the smallest of the interior angles.*

Proof See Problem 1.23. □

The result (1.74) shows that bounding the aspect ratio is equivalent to ensuring that the minimum interior angle is bounded away from zero. Combining (1.74) with the bounds in Lemmas 1.11–1.13, we see that the interpolation error bound (1.69) satisfies

$$\|\nabla(u - \pi_h u)\|^2 \le C \sum_{\triangle_k \in \mathcal{T}_h} \frac{1}{\sin^2\theta_k} h_k^2 \left\|D^2 u\right\|_{\triangle_k}^2. \tag{1.75}$$

The bound (1.75) can be further simplified by making the assumption that the mesh refinement is shape regular as follows.

Definition 1.15 (Minimum angle condition). A sequence of triangular grids $\{\mathcal{T}_h\}$ is said to be *shape regular* if there exists a minimum angle $\theta_* \neq 0$ such that every element in $\mathcal{T}_h$ satisfies $\theta_T \ge \theta_*$.

In particular, shape regularity ensures that $1/\sin\theta_k \le 1/\sin\theta_*$ for all triangles in $\mathcal{T}_h$, so that (1.75) simplifies to

$$\|\nabla(u - \pi_h u)\|^2 \le C(\theta_*) \sum_{\triangle_k \in \mathcal{T}_h} h_k^2 \left\|D^2 u\right\|_{\triangle_k}^2. \tag{1.76}$$

Noting that $h_k \le h$ for all triangles $\triangle_k$ gives the desired uniform bound (that is, independent of the triangulation)

$$\|\nabla(u - \pi_h u)\|^2 \le Ch^2 \sum_{\triangle_k \in \mathcal{T}_h} \left\|D^2 u\right\|_{\triangle_k}^2 = Ch^2 \left\|D^2 u\right\|^2.$$

Combining with (1.68) then gives the error bound (1.67) in Theorem 1.10. We also note that the less stringent *maximum angle condition*, which requires all angles to be uniformly bounded away from π, can also be used to obtain these results; see Křížek [154].

A similar argument can be used to establish a bound for the L_2 interpolation error associated with the function u itself. Shape regularity is not needed since derivatives are not mapped from $\triangle_k$ to the reference element. Thus, for a mesh of linear elements the following result can be readily established.

Proposition 1.16. $\|u - \pi_h u\|^2 \le C \sum_{\triangle_k \in \mathcal{T}_h} h_k^4 \left\|D^2 u\right\|_{\triangle_k}^2$.

Proof See Problem 1.24. □

We now consider the analogue of Theorem 1.10 in the case of grids of rectangular elements using $\boldsymbol{Q}_1$ approximation. (Recall from Problem 1.13 that the

Jacobian reduces to a constant diagonal matrix in this case.) The analogues of Lemmas 1.11 and 1.13 are given below.

Proposition 1.17. *Given a rectangular element $\square_k$, with horizontal and vertical edges of lengths h_x and h_y, respectively, let π_h^1 be the standard bilinear interpolant which agrees with the underlying function at the four vertices. Then*

$$\left\|\nabla(u-\pi_h^1 u)\right\|_{\square_k}^2 \leq \max\left\{\frac{h_x}{h_y},\frac{h_y}{h_x}\right\}\left\|\nabla(\bar{u}-\pi_h^1\bar{u})\right\|_{\square *}^2, \tag{1.77}$$

$$\left\|D^2\bar{u}\right\|_{\square *}^2 \leq \frac{h_k^2}{4}\max\left\{\frac{h_x}{h_y},\frac{h_y}{h_x}\right\}\left\|D^2 u\right\|_{\square_k}^2, \tag{1.78}$$

where $h_k = \max\{h_x, h_y\}$.

Proof See Problem 1.25. □

Notice that the rectangle aspect ratio $\beta_T = \max\{h_x/h_y, h_y/h_x\}$ plays the same role as the triangle aspect ratio in Lemmas 1.11 and 1.13.

Definition 1.18 (Aspect ratio condition). A sequence of rectangular grids $\{\mathcal{T}_h\}$ is said to be *shape regular* if there exists a maximum rectangle edge ratio β_* such that every element in $\mathcal{T}_h$ satisfies $1 \leq \beta_T \leq \beta_*$.

A second key point is that the analogue of Lemma 1.12 also holds in this case:

$$\left\|\nabla(\bar{u}-\pi_h^1\bar{u})\right\|_{\square *} \leq C\left\|D^2(\bar{u}-\pi_h^1\bar{u})\right\|_{\square *} \equiv C\left\|D^2\bar{u}\right\|_{\square *}. \tag{1.79}$$

Combining the bounds (1.77), (1.79) and (1.78) gives the anticipated error estimate.

Theorem 1.19. *If the variational problem* (1.57) *is solved using a mesh of bilinear rectangular elements, and if the aspect ratio condition is satisfied (see Definition* 1.18*), then there exists a constant C_1 such that*

$$\|\nabla(u-u_h)\| \leq C_1 h \left\|D^2 u\right\|, \tag{1.80}$$

where h is the length of the longest edge in $\mathcal{T}_h$.

Remark 1.20. If the degree of element distortion is small, a similar bound to (1.80) also holds in the case of $\boldsymbol{Q}_1$ approximation on grids of isoparametrically mapped quadrilateral elements. For grids of parallelograms, given an appropriate definition of shape regularity (involving a minimum angle and an aspect ratio condition) the convergence bound is identical to (1.80); see Braess [26, Theorem 7.5].

The construction of the error estimate (1.80) via the intermediate results (1.77), (1.79) and (1.78) provides the basis for establishing error bounds when higher-order ($\boldsymbol{P}_m$, $\boldsymbol{Q}_m$, with $m \geq 2$) approximation spaces are used.

Theorem 1.21. *Using a higher-order finite element approximation space* $\boldsymbol{P}_m$ *or* $\boldsymbol{Q}_m$ *with* $m \geq 2$ *leads to the higher-order convergence bound*

$$\|\nabla(u - u_h)\| \leq C_{\mathbf{m}}\, h^m \, \left\|D^{m+1}u\right\|. \tag{1.81}$$

In other words, we get m*th order convergence as long as the regularity of the target solution is good enough. Note that* $\left\|D^{m+1}u\right\| < \infty$ *if and only if the* $(m+1)$*st generalized derivatives of* u *are in* $L_2(\Omega)$.

For example, using biquadratic approximation on a square element grid, we have the following analogue of Proposition 1.17.

Proposition 1.22. *For a grid of square elements* $\square_k$ *with edges of length* h*, let* π_h^2 *be the standard biquadratic interpolant, which agrees with the underlying function at nine points; see Figure* 1.11*. Then*

$$\left\|\nabla(u - \pi_h^2 u)\right\|_{\square_k}^2 = \left\|\nabla(\bar{u} - \pi_h^2 \bar{u})\right\|_{\square *}^2, \tag{1.82}$$

$$\left\|D^3 \bar{u}\right\|_{\square *}^2 \leq \frac{h^4}{16} \left\|D^3 u\right\|_{\square_k}^2. \tag{1.83}$$

Proof See Problem 1.26. □

Combining (1.82) and (1.83) with the reference element bound given by the Bramble–Hilbert lemma (in this case bounding in terms of the third derivatives; cf. Lemma 1.12),

$$\left\|\nabla(\bar{u} - \pi_h^2 \bar{u})\right\|_{\square *} \leq C \left\|D^3(\bar{u} - \pi_h^2 \bar{u})\right\|_{\square *} \equiv C \left\|D^3 \bar{u}\right\|_{\square *} \tag{1.84}$$

leads to (1.81) with $m = 2$.

To conclude this section, we will use the problems in Examples 1.1.3 and 1.1.4 to illustrate that the orders of convergence suggested by the error bounds (1.80) and (1.81) are typical of the behavior of the error as the grid is successively refined. An assessment of the orders of convergence that are obtained for the problems in Examples 1.1.1 and 1.1.2, can be found in Computational Exercises 1.1 and 1.4, respectively (with a definitive statement given in Computational Exercise 1.2.) Results for the problem in Example 1.1.3 are given in Table 1.1. The error measure E_h used here is the difference between the exact and the discrete energy, that is,

$$E_h = \left| \, \|\nabla u\|^2 - \|\nabla u_h\|^2 \, \right|^{1/2}. \tag{1.85}$$

If zero essential boundary conditions are imposed, then $\|\nabla u\| \geq \|\nabla u_h\|$ and E_h is identical to the energy error $\|\nabla(u - u_h)\|$; see Problem 1.18. Notice how the $\boldsymbol{Q}_1$ errors in Table 1.1 decrease by a factor of two for every successive refinement,[11] whereas the $\boldsymbol{Q}_2$ errors ultimately decrease by a factor of four. The outcome is that biquadratic elements are more accurate than bilinear elements—in fact they

[11] ℓ is the *grid parameter* specification in the IFISS software that is associated with the tabulated entry.

Table 1.1 *Energy error E_h for Example* 1.1.3*; ℓ is the grid refinement level, h is $2^{1-\ell}$ for $\boldsymbol{Q}_1$ approximation and $2^{2-\ell}$ for $\boldsymbol{Q}_2$ approximation.*

ℓ	$\boldsymbol{Q}_1$	$\boldsymbol{Q}_2$	n
2	5.102×10^{-2}	6.537×10^{-3}	9
3	2.569×10^{-2}	2.368×10^{-3}	49
4	1.287×10^{-2}	5.859×10^{-4}	225
5	6.437×10^{-3}	1.460×10^{-4}	961
6	3.219×10^{-3}	3.646×10^{-5}	3969

are generally more cost effective wherever the underlying solution is sufficiently smooth if reasonable accuracy is required. For example, the $\boldsymbol{Q}_2$ solution on the coarsest grid has approximately $1/4^3$ of the degrees of freedom of the $\boldsymbol{Q}_1$ solution on the second-finest grid, yet both are of comparable accuracy.

If the weak solution is not smooth, then the superiority of the $\boldsymbol{Q}_2$ approximation method over the simpler $\boldsymbol{Q}_1$ method is not so clear. To illustrate this, the energy differences E_h computed in the case of the singular problem in Example 1.1.4 are presented in Table 1.2. Notice that—in contrast to the behavior in Table 1.1—the $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ errors both decrease by a factor of approximately $2^{2/3} \approx 1.5874$ with every successive refinement of the grid. The explanation for this is that the solution regularity is between $\mathcal{H}^1$ and $\mathcal{H}^2$ in this case;[12] see Problem 1.27. The upshot is that in place of (1.80) and (1.81), the following convergence bound is the best that can be achieved (for all $\varepsilon > 0$):

$$\|\nabla(u - u_h)\| \leq C_{\mathbf{m}}(\epsilon)\, h^{2/3-\varepsilon} \tag{1.86}$$

using an approximation of arbitrary order $m \geq 1$!

Table 1.2 *Energy error E_h for Example* 1.1.4*.*

ℓ	$\boldsymbol{Q}_1$	$\boldsymbol{Q}_2$	n
2	1.478×10^{-1}	9.860×10^{-2}	33
3	9.162×10^{-2}	6.207×10^{-2}	161
4	5.714×10^{-2}	3.909×10^{-2}	705
5	3.577×10^{-2}	2.462×10^{-2}	2945

[12]Introducing Sobolev spaces with fractional indices as in Johnson [138, pp. 92–94], it may be shown that $u \in \mathcal{H}^{5/3-\varepsilon}$.

Table 1.3 *Energy error E_h for stretched grid solutions of Example 1.1.4; $\ell = 4$.*

α	$\boldsymbol{Q}_1$	$\boldsymbol{Q}_2$
1	9.162×10^{-2}	6.207×10^{-2}
5/4	6.614×10^{-2}	3.723×10^{-2}
3/2	7.046×10^{-2}	2.460×10^{-2}
2	1.032×10^{-1}	2.819×10^{-2}

When solving problems like those in Examples 1.1.3 and 1.1.4, it is natural to try to design rectangular or triangular meshes that concentrate the degrees of freedom in the neighborhood of the singularity. The motivation for doing this is the intermediate bound (1.76), which suggests that it is important to try to balance the size of h_k with that of $\|D^2 u\|_{\triangle_k}$. Roughly speaking, h_k should be small in those elements where the derivatives of u are large. To illustrate the idea, Table 1.3 lists the errors E_h obtained when solving the problem in Example 1.1.4 using tensor-product grids that are geometrically stretched toward the singularity, with successive element edges a factor α times longer than the adjacent edge; see Figure 1.19. Notice that comparing the results in Table 1.3 with those in Table 1.2, we see that an appropriately stretched grid of $\boldsymbol{Q}_2$ elements with 161 degrees of freedom gives better accuracy than that obtained using a uniform grid with 2945 degrees of freedom—the challenge here is to determine the optimal stretching a priori!

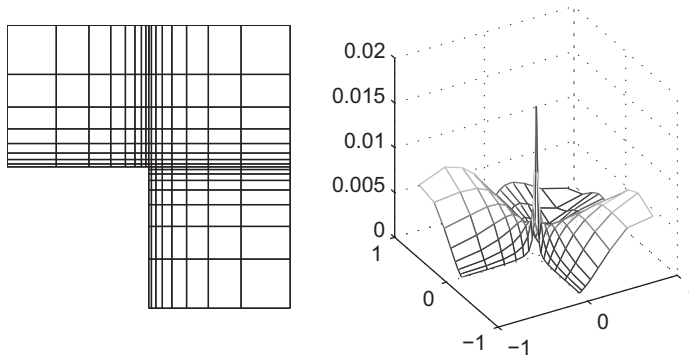


FIG. 1.19. Stretched level 4 grid with $\alpha = \frac{3}{2}$ (left) for Example 1.1.4 and surface plot (right) of the estimated error using $\boldsymbol{Q}_1$ approximation (see Section 1.5.2).

1.5.2 *A posteriori error bounds*

The fact that physically interesting problems typically have singularities is what motivates the concept of a posteriori error estimation. Specifically, given a finite element subdivision $\mathcal{T}_h$ and a solution u_h, we want to compute a local (element) error estimator η_T such that $\|\nabla\eta_T\|$ approximates the local energy error $\|\nabla(u-u_h)\|_T$ for every element T in $\mathcal{T}_h$. An important factor is the requirement that η_T should be cheap to compute—as a rule of thumb, the computational work should scale linearly as the number of elements is increased—yet there should be guaranteed accuracy in the sense that the estimated global error should give an upper bound on the exact error, so that

$$\|\nabla(u-u_h)\|^2 \equiv \sum_{T\in\mathcal{T}_h} \|\nabla(u-u_h)\|_T^2 \le C(\theta_*) \sum_{T\in\mathcal{T}_h} \eta_T^2 \tag{1.87}$$

with a constant C that depends only on shape regularity. If, in addition to satisfying (1.87), η_T provides a lower bound for the exact local error,

$$\eta_T \le C(\theta_{\omega_T}) \|\nabla(u-u_h)\|_{\omega_T}, \tag{1.88}$$

where ω_T typically represents a local patch of elements adjoining T, then the estimator η_T is likely to be effective if it is used to drive an adaptive refinement process. For the problem in Example 1.1.4, such a process will give rise to successive meshes that are selectively refined in the vicinity of the singularity so as to equidistribute the error among all elements and enhance overall cost effectiveness.

The two key aspects of error estimation are *localization* and *approximation*. The particular strategy that is built into the IFISS software is now described. The starting point is the characterization (1.58) of the error $e = u - u_h \in \mathcal{H}^1_{E_0}$:

$$a(e,v) = \ell(v) - a(u_h,v) \quad \text{for all } v \in \mathcal{H}^1_{E_0}. \tag{1.89}$$

For simplicity, it is assumed here that Neumann data is homogeneous, so that $\ell(v) = (f,v)$. Using the shorthand notation $(u,v)_T := \int_T uv$ and $a(u,v)_T := \int_T \nabla u\cdot\nabla v$ to represent the localized L_2 and energy inner products, respectively, the error equation (1.89) may be broken up into element contributions:

$$\sum_{T\in\mathcal{T}_h} a(e,v)_T = \sum_{T\in\mathcal{T}_h} (f,v)_T - \sum_{T\in\mathcal{T}_h} a(u_h,v)_T. \tag{1.90}$$

Integrating by parts elementwise then gives

$$-a(u_h,v)_T = (\nabla^2 u_h, v)_T - \sum_{E\in\mathcal{E}(T)} \langle \nabla u_h\cdot\vec{n}_{E,T}, v\rangle_E, \tag{1.91}$$

where $\mathcal{E}(T)$ denotes the set of edges (faces in $\mathbb{R}^3$) of element T, $\vec{n}_{E,T}$ is the outward normal with respect to E, $\langle\cdot,\cdot\rangle_E$ is the L_2 inner product on E and $\nabla u_h\cdot\vec{n}_{E,T}$ is the discrete (outward-pointing) normal flux. The finite element

approximation typically has a discontinuous normal derivative across interelement boundaries. Consequently it is convenient to define the *flux jump* across the edge or face E adjoining elements T and S as

$$\left[\!\left[\frac{\partial v}{\partial n}\right]\!\right] := (\nabla v|_T - \nabla v|_S)\cdot \vec{n}_{E,T} = (\nabla v|_S - \nabla v|_T)\cdot \vec{n}_{E,S}, \tag{1.92}$$

and then to *equidistribute* the flux jump contribution in (1.90) to the adjoining elements in equal proportion (with an appropriate modification for elements that have one or more edges/faces adjoining $\partial\Omega$):

$$\sum_{T\in\mathcal{T}_h} a(e,v)_T = \sum_{T\in\mathcal{T}_h}\left[(f+\nabla^2 u_h, v)_T - \frac{1}{2}\sum_{E\in\mathcal{E}(T)}\left\langle\left[\!\left[\frac{\partial u_h}{\partial n}\right]\!\right], v\right\rangle_E\right]. \tag{1.93}$$

It is evident from the structure of the right-hand side of equation (1.93) that e has two distinct components: these are the (element) *interior residual* $R_T := \{f+\nabla^2 u_h\}|_T$ and the (interelement) flux jump $R_E := [\![\partial u_h/\partial n]\!]$. Notice also that if u_h agrees with the classical solution everywhere then both R_T and R_E are identically zero. The residual terms R_T and R_E enter either implicitly or explicitly into the definition of many finite element error estimators.

In the remainder of this section we concentrate on the specific case of S_0^h being defined by the $\boldsymbol{P}_1$ or $\boldsymbol{Q}_1$ approximation over a triangular or rectangular element subdivision. The appeal of these lowest-order methods is their simplicity: the flux jump is piecewise constant in the $\boldsymbol{P}_1$ case, and in both cases the interior residual $R_T = f|_T$ is independent of u_h and thus can be computed a priori. As a further simplification, R_T can be approximated by a constant R_T^0 by projecting f onto the space of piecewise constant functions.

To define a consistent flux jump operator with respect to elements adjoining $\partial\Omega$, some additional notation is needed. We let $\mathcal{E}_h = \cup_{T\in\mathcal{T}_h}\mathcal{E}(T)$ denote the set of all edges split into interior and boundary edges via

$$\mathcal{E}_h := \mathcal{E}_{h,\Omega}\cup\mathcal{E}_{h,D}\cup\mathcal{E}_{h,N},$$

where $\mathcal{E}_{h,\Omega} := \{E\in\mathcal{E}_h\colon E\subset\Omega\}$, $\mathcal{E}_{h,D} := \{E\in\mathcal{E}_h\colon E\subset\partial\Omega_D\}$ and $\mathcal{E}_{h,N} := \{E\in\mathcal{E}_h\colon E\subset\partial\Omega_N\}$. We then define the operator

$$R_E^* = \begin{cases} \frac{1}{2}[\![\partial u_h/\partial n]\!], & E\in\mathcal{E}_{h,\Omega},\\ -\nabla u_h\cdot\vec{n}_{E,T}, & E\in\mathcal{E}_{h,N},\\ 0, & E\in\mathcal{E}_{h,D}.\end{cases}$$

The fact that the exact error e is characterized by the enforcement of (1.93) over the space $\mathcal{H}^1_{E_0}$ provides us with a handle for estimating the local error in each element T. Specifically, if a suitable (finite-dimensional) approximation space, $\mathcal{Q}_T$ say, is constructed, then an approximation to $e\,|_T$ can be obtained

by enforcing (1.93) elementwise. Specifically, a function $e_T \in \mathcal{Q}_T$ is computed such that

$$(\nabla e_T, \nabla v)_T = (R_T^0, v)_T - \sum_{E \in \mathcal{E}(T)} \langle R_E^*, v \rangle_E \tag{1.94}$$

for all $v \in \mathcal{Q}_T$, and the local error estimator is the energy norm of e_T,

$$\eta_T = \|\nabla e_T\|_T . \tag{1.95}$$

Making an appropriate choice of approximation space $\mathcal{Q}_T$ in (1.94) is clearly crucial. A clever choice (due to Bank & Weiser [11]) is the "correction" space

$$\mathcal{Q}_T = Q_T \oplus B_T \tag{1.96}$$

consisting of edge and interior bubble functions,

$$Q_T = \text{ span } \{\psi_E : E \in \mathcal{E}(T) \cap (\mathcal{E}_{h,\Omega} \cup \mathcal{E}_{h,N})\}, \tag{1.97}$$

where $\psi_E : T \to \mathbb{R}$ is the quadratic (or biquadratic) edge bubble that is 0 on the other two (or three) edges of T and B_T is the space spanned by interior cubic (or biquadratic) bubbles ϕ_T such that $0 \le \phi_T \le 1$, $\phi_T = 0$ on ∂T and $\phi_T = 1$ only at the centroid. The upshot is that for each triangular (or rectangular) element, a 4×4 (or 5×5) system of equations must be solved to compute e_T.[13]

A feature of the choice of space (1.97) is that $(\nabla v, \nabla v)_T > 0$ for all functions v in $\mathcal{Q}_T$ (intuitively, a constant function in T cannot be represented as a linear combination of bubble functions), so the local problem (1.94) is well posed. This means that the element matrix systems are all nonsingular; see Problem 1.28. This is important for a typical element T that has no boundary edges, since the local problem (1.94) represents a weak formulation of the Neumann problem,

$$-\nabla^2 e_T = f \quad \text{in } T, \tag{1.98}$$

$$\frac{\partial e_T}{\partial n} = -\frac{1}{2} \left[\!\!\left[\frac{\partial u_h}{\partial n} \right]\!\!\right] \quad \text{on } E \in \mathcal{E}(T), \tag{1.99}$$

suggesting that a compatibility condition (cf. (1.4))

$$\int_T f - \frac{1}{2} \sum_{E \in \mathcal{E}(T)} \int_E \left[\!\!\left[\frac{\partial u_h}{\partial n} \right]\!\!\right] = 0 \tag{1.100}$$

needs to be satisfied in order to ensure the existence of e_T. The difficulty associated with the need to enforce (1.100) is conveniently circumvented by the choice (1.97).

Remark 1.23. The idea of estimating the energy error locally by solving a low-dimensional problem of the form (1.94) immediately extends to $\boldsymbol{Q}_2$ approximation. The only issue is the need to construct an appropriate extended basis

[13] This strategy is embodied in the IFISS routine `diffpost_q1.m`.

Table 1.4 *Comparison of estimated and exact errors for Example* 1.1.3.

ℓ	$\|\nabla(u-u_h)\|$	η	X_η
2	5.032×10^{-2}	4.954×10^{-2}	0.9845
3	2.516×10^{-2}	2.511×10^{-2}	0.9980
4	1.258×10^{-2}	1.257×10^{-2}	0.9992
5	6.291×10^{-3}	6.288×10^{-3}	0.9995

set $\mathcal{Q}_T$. The default strategy that is built into IFISS is to use a reduced $\boldsymbol{Q}_4$ basis set so that a 12×12 system of equations must be solved to compute e_T.[14]

To illustrate the remarkable effectiveness of this error estimation procedure, the analytic test problem in Example 1.1.3 is discretized using uniform grids of $\boldsymbol{Q}_1$ elements, and a comparison between the exact energy error $\|\nabla e\|$ and the estimated global error $\eta = \left(\sum_{T\in\mathcal{T}_h} \eta_T^2\right)^{1/2}$ is given in Table 1.4. The close agreement between the estimated and exact errors is quite amazing.[15] Another virtue of the estimator illustrated by Table 1.4 is the fact that the *global effectivity index* $X_\eta := \eta / \|\nabla e\|$ converges to unity as $h \to 0$. This property is usually referred to as *asymptotic exactness*.

The results in Table 1.4 suggest that the estimator η_T satisfies the required error bound (1.87) (with a proportionality constant $C(\theta_*)$ that is close to unity if the elements are not too distorted). A precise result is stated below. This should also be compared with the a priori error bound given in Theorem 1.19.

Theorem 1.24. *If the variational problem* (1.57) *is solved using a mesh of bilinear rectangular elements, and if the rectangle aspect ratio condition is satisfied with* β_* *given in Definition* 1.18*, then the estimator* $\eta_T \equiv \|\nabla e_T\|_T$ *computed via* (1.94) *using the approximation space* (1.97) *gives the bound*

$$\|\nabla(u-u_h)\| \le C(\beta_*) \left(\sum_{T\in\mathcal{T}_h} \eta_T^2 + h^2 \sum_{T\in\mathcal{T}_h} \left\| R_T - R_T^0 \right\|_T^2 \right)^{1/2}, \qquad (1.101)$$

where h *is the length of the longest edge in* $\mathcal{T}_h$*.*

Remark 1.25. If f is a piecewise constant function then the consistency error term $\left\| R_T - R_T^0 \right\|_T$ is identically zero. Otherwise, if f is smooth, this term represents a high-order perturbation. In any case the estimator η_T is reliable; for further details see Verfürth [264].

[14] This strategy is embodied in the IFISS routine `diffpost_q2_with_q4.m`.

[15] Although performance deteriorates using stretched meshes, the agreement between exact and estimated errors is quite acceptable; see Computational Exercise 1.5.

A proof of Theorem 1.24 is outlined below. An important difference between the a priori bound (1.80) and the a posteriori bound (1.101) is that $\mathcal{H}^2$ regularity is not assumed in the latter case. This adds generality (since the bound (1.101) applies even if the problem is singular) but raises the technical issue within the proof of Theorem 1.24 of having to approximate a possibly discontinuous $\mathcal{H}^1$ function. Since point values of $\mathcal{H}^1(\Omega)$ functions are not defined for $\Omega \subset \mathbb{R}^2$, an alternative to interpolation using local averaging over neighborhoods of the vertices of the subdivision is required. This leads to the quasi-interpolation estimates (due to Clément [54]) given in the following lemma. For a detailed discussion, see Brenner & Scott [36, pp. 118–120].

Lemma 1.26. *Given $e \in \mathcal{H}^1_{E_0}$ there exists a quasi-interpolant $e_h^* \in S_0^h$ such that*

$$\|e - e_h^*\|_T \leq C_1(\beta_{\tilde{\omega}_T})\, h_T\, \|\nabla e\|_{\tilde{\omega}_T} \quad \textit{for all } T \in \mathcal{T}_h, \tag{1.102}$$

$$\|e - e_h^*\|_E \leq C_2(\beta_{\tilde{\omega}_T})\, h_E^{1/2}\, \|\nabla e\|_{\tilde{\omega}_T} \quad \textit{for all } E \in \mathcal{E}_h, \tag{1.103}$$

where $\tilde{\omega}_T$ is the patch of all the neighboring elements that have at least one vertex connected to a vertex of element T.

Notice that the constants in (1.102) and (1.103) depend only on the maximum aspect ratio over all elements in the patch. The proof of Theorem 1.24 will also require so-called *local inverse estimates*. A typical example is given in the lemma below. A proof for low-order basis functions is provided; see Brenner & Scott [36, Section 4.5] and Ciarlet [53, Section 3.2] for more general analysis.

Lemma 1.27. *Given a polynomial function u_k defined in a triangular or rectangular element T, a constant C exists, depending only on the element aspect ratio, such that*

$$\|\nabla u_k\|_T \leq C h_T^{-1}\, \|u_k\|_T\,, \tag{1.104}$$

where h_T is the length of the longest edge of T.

Proof This is a standard scaling argument of the type used in the proof of Lemma 1.11. In the case of triangular elements with $\boldsymbol{P}_1$ (linear) basis functions, the argument of that proof gives

$$\|\nabla u_k\|^2_{\triangle_k} \leq 2\frac{h_k^2}{|\triangle_k|}\, \|\nabla \bar{u}_k\|^2_{\triangle_*}\,. \tag{1.105}$$

Note that

$$\bar{u}_k \mapsto \|\nabla \bar{u}_k\|_{\triangle_*}\,, \quad \bar{u}_k \mapsto \|\bar{u}_k\|_{\triangle_*}$$

constitute a seminorm and norm, respectively, on finite-dimensional spaces. It follows that, as in the equivalence of norms on finite-dimensional spaces,

$$\|\nabla \bar{u}_k\|_{\triangle_*} \leq C\, \|\bar{u}_k\|_{\triangle_*}\,. \tag{1.106}$$

Mapping back to the original element, we have

$$\|\bar{u}_k\|^2_{\triangle *} = \frac{1}{2|\triangle_k|}\,\|u_k\|^2_{\triangle_k}\,, \tag{1.107}$$

and combining (1.105)–(1.107) with (1.74) gives the stated result. The proof for a rectangular element is left as an exercise; see Problem 1.29. □

Returning to the proof of Theorem 1.24, the first step is to use Galerkin orthogonality (1.59), the error equation (1.89) and the definition of R^*_E:

$$\begin{aligned}
\|\nabla e\|^2 &= a(e,e)\\
&= a(e, e-e^*_h) \quad \text{(setting } v_h = e^*_h \text{ in (1.59))}\\
&= \ell(e-e^*_h) - a(u_h, e-e^*_h) \quad \text{(setting } v = e-e^*_h \text{ in (1.89))}\\
&= \sum_{T\in\mathcal{T}_h}\Big\{(R_T, e-e^*_h)_T - \sum_{E\in\mathcal{E}(T)}\langle R^*_E, e-e^*_h\rangle_E\Big\} \quad \text{(using (1.91))}\\
&\le C(\beta_*)\sum_{T\in\mathcal{T}_h}\Big\{h_T\,\|R_T\|_T\,\|\nabla e\|_{\tilde{\omega}_T} + \sum_{E\in\mathcal{E}(T)} h_E^{1/2}\,\|R^*_E\|_E\,\|\nabla e\|_{\tilde{\omega}_T}\Big\}\\
&\le C(\beta_*)\Big(\sum_{T\in\mathcal{T}_h}\|\nabla e\|^2_{\tilde{\omega}_T}\Big)^{1/2}\Big(\sum_{T\in\mathcal{T}_h}\Big\{h_T\,\|R_T\|_T + \sum_E h_E^{1/2}\,\|R^*_E\|_E\Big\}^2\Big)^{1/2}.
\end{aligned}$$

For a rectangular subdivision, the union of the patches $\tilde{\omega}_T$ covers Ω at most nine times, thus $\sum_{T\in\mathcal{T}_h}\|\nabla e\|^2_{\tilde{\omega}_T} \le 9\,\|\nabla e\|^2$. Noting that $(a+b)^2 \le 2a^2+2b^2$ then leads to the following *residual estimator* error bound:

$$\|\nabla(u-u_h)\| \le C(\beta_*)\Big(\sum_{T\in\mathcal{T}_h}\Big\{h_T^2\,\|R_T\|^2_T + \sum_{E\in\mathcal{E}(T)} h_E\,\|R^*_E\|^2_E\Big\}\Big)^{1/2}. \tag{1.108}$$

Remark 1.28. The combination of the interior residual and flux jump terms on the right-hand side of (1.108) can be used to define a simple *explicit* estimator $\bar{\eta}_T$; see Verfürth [264]. In practice, the far superior accuracy of the local problem estimator (1.95) outweighs the computational cost incurred in solving the local problems (1.94), so the use of the cheaper estimator $\bar{\eta}_T$ in place of η_T is not recommended.

To show that the residual bound (1.108) implies the bound (1.101), we take the trivial bound $\|R_T\|_T \le \|R^0_T\|_T + \|R_T - R^0_T\|_T$, and exploit the fact that R^0_T and R^*_E are piecewise constant to show that the terms $h_T^2\,\|R^0_T\|^2_T$ and $h_E\,\|R^*_E\|^2_E$ on the right-hand side of (1.108) are individually bounded by η^2_T.

The interior residual term is dealt with first. For $T \in \mathcal{T}_h$, we note that $R_T^0|_T \in \boldsymbol{P}_0$ and define $w_T = R_T^0 \phi_T \in B_T \subset \mathcal{Q}_T$. It follows that

$$\begin{aligned}\left\|R_T^0\right\|_T^2 = C(R_T^0, w_T)_T &= C(\nabla e_T, \nabla w_T)_T \quad \text{(setting } v = w_T \text{ in (1.94))} \\ &\le C \left\|\nabla e_T\right\|_T \left\|\nabla w_T\right\|_T \\ &\le C \left\|\nabla e_T\right\|_T h_T^{-1} \left\|w_T\right\|_T \quad \text{(applying (1.104))} \\ &\le C h_T^{-1} \left\|\nabla e_T\right\|_T \left\|R_T^0\right\|_T ,\end{aligned}$$

where in the last step we use the fact that $0 \le \phi_T \le 1$. This gives

$$h_T \left\|R_T^0\right\|_T \le C \left\|\nabla e_T\right\|_T . \tag{1.109}$$

The jump term is handled in the same way. For an interior edge, we define ω_E to be the union of the two elements adjoining edge $E \in \mathcal{E}(T)$, and define $w_E = R_E \psi_E \in Q_T \subset \mathcal{Q}_T$. From (1.94) we then have

$$\left\|R_E^*\right\|_E^2 \le C \left\langle R_E^*, w_E \right\rangle_E = C \sum_{T' \subset \omega_E} \left\{ -(\nabla e_{T'}, \nabla w_E)_{T'} + (R_{T'}^0, w_E)_{T'} \right\},$$

which, when combined with the scaling results $\left\|w_E\right\|_{T'} \le h_E^{1/2} \left\|w_E\right\|_E$ and $\left\|\nabla w_E\right\|_{T'} \le h_E^{-1/2} \left\|w_E\right\|_E$, leads to the desired bound,

$$h_E^{1/2} \left\|R_E^*\right\|_E \le C \sum_{T' \subset \omega_E} \left\|\nabla e_{T'}\right\|_{T'} . \tag{1.110}$$

Combining (1.109) and (1.110) with (1.108) then gives the upper bound (1.101) in Theorem 1.24.

The remaining issue is whether or not the estimated error η_T gives a lower bound on the local error. A precise statement is given below.

Proposition 1.29. *If the variational problem* (1.57) *is solved using a grid of bilinear rectangular elements, and if the rectangle aspect ratio condition is satisfied, then the estimator* $\eta_T \equiv \left\|\nabla e_T\right\|_T$ *computed via* (1.94) *using the approximation space* (1.97) *gives the bound*

$$\eta_T \le C(\beta_{\omega_T}) \left\|\nabla (u - u_h)\right\|_{\omega_T} , \tag{1.111}$$

where ω_T *represents the patch of five elements that have at least one boundary edge* E *from the set* $\mathcal{E}(T)$.

Proof See Problem 1.30. □

To illustrate the usefulness of a posteriori error estimation, plots of the estimated error e_T associated with computed solutions u_h to the problems in Examples 1.1.3 and 1.1.4, are presented in Figures 1.20 and 1.21, respectively. The structure of the error can be seen to be very different in these two cases. Whereas the error distribution is a smooth function when solving Example 1.1.3, the effect of the singularity on the error distribution is very obvious in Figure 1.21.

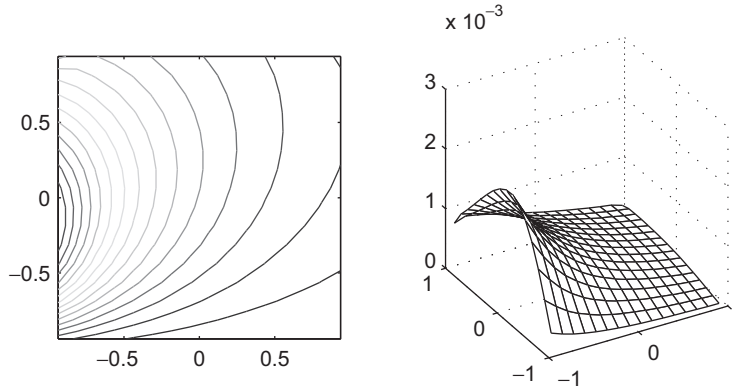


FIG. 1.20. Contour plot (left) and three-dimensional surface plot (right) of the estimated error associated with the finite element solution to Example 1.1.3 given in Figure 1.3.

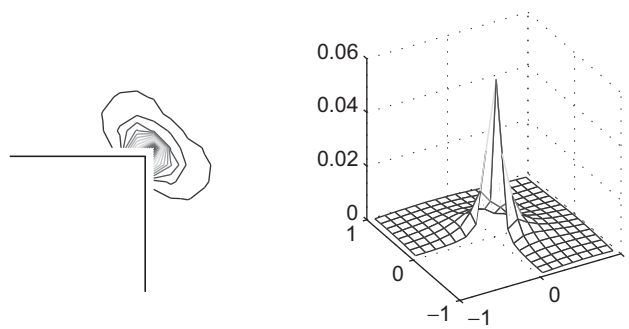


FIG. 1.21. Contour plot (left) and three-dimensional surface plot (right) of the estimated error associated with the finite element solution to Example 1.1.4 given in Figure 1.4.

Moreover, comparing this error distribution, which comes from a uniform grid, with that in Figure 1.19 (derived from a stretched grid with the same number of degrees of freedom) clearly suggests that the most effective way of increasing accuracy at minimal cost is to perform local refinement in the neighborhood of the corner. These issues are explored further in Computational Exercise 1.4.

1.6 Matrix properties

In this section, we describe some properties of the matrices arising from finite element discretization of the Poisson equation. These results will be used in the

next chapter to analyze the behavior of iterative solution algorithms applied to the discrete systems of equations.

Let $\langle \mathbf{v}, \mathbf{w} \rangle = \mathbf{v}^T \mathbf{w}$ denote the Euclidean inner product on $\mathbb{R}^n$, with associated norm $\|\mathbf{v}\| = \langle \mathbf{v}, \mathbf{v} \rangle^{1/2}$. We begin with the observation that for *any* symmetric positive-definite matrix A of order n, the bilinear form given by

$$\langle \mathbf{v}, \mathbf{w} \rangle_A := \langle A\mathbf{v}, \mathbf{w} \rangle \tag{1.112}$$

defines an inner product on $\mathbb{R}^n$ with associated norm $\|\mathbf{v}\|_A = \langle \mathbf{v}, \mathbf{v} \rangle_A^{1/2}$. Given that A, the discrete Laplacian operator introduced in Section 1.3, is indeed symmetric and positive definite, the inner product (1.112) and norm are well defined in this case. Any vector $\mathbf{v} \in \mathbb{R}^n$ uniquely corresponds to a finite element function $v_h \in S_0^h$, and in particular there is a unique correspondence between the finite element solution u_h and the solution $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n)^T$ to the matrix equation (1.21). If v_h and w_h are two functions in S_0^h, with coefficient vectors $\mathbf{v}$ and $\mathbf{w}$, respectively, in $\mathbb{R}^n$, then the bilinear form derived from the Poisson equation, that is, $a(\cdot, \cdot)$ of (1.55), satisfies

$$a(v_h, w_h) = \int_\Omega \nabla v_h \cdot \nabla w_h = \langle \mathbf{v}, \mathbf{w} \rangle_A. \tag{1.113}$$

In simple terms, there is a one-to-one correspondence between the bilinear form $a(\cdot, \cdot)$ defined on the function space S_0^h, and the discrete inner product (1.112).

Recall from (1.22) that the discrete Laplacian can be viewed as the *Grammian matrix* of the basis $\{\phi_j\}$ associated with the inner product $a(\cdot, \cdot)$. It will also turn out to be useful to identify the Grammian with respect to the L_2 inner product,

$$Q = [q_{ij}], \qquad q_{ij} = \int_\Omega \phi_j \phi_i \, . \tag{1.114}$$

With this definition, it follows that for v_h, $w_h \in S_0^h$,

$$(v_h, w_h) = \langle Q\mathbf{v}, \mathbf{w} \rangle.$$

An immediate consequence is that Q is symmetric and positive definite, and the inner product $\langle \cdot, \cdot \rangle_Q$ defined by it constitutes a representation in $\mathbb{R}^n$ of the L_2 inner product in S_0^h. The matrix Q in (1.114) is referred to as the *mass matrix*.

The key property of the mass matrix is the following.

Proposition 1.30. *For $\boldsymbol{P}_1$ or $\boldsymbol{Q}_1$ approximation on a subdivision in $\mathbb{R}^2$ for which a shape-regularity condition holds (as given in Definitions* 1.15 *and* 1.18*), the mass matrix Q approximates the scaled identity matrix in the sense that*

$$c\underline{h}^2 \leq \frac{\langle Q\mathbf{v}, \mathbf{v} \rangle}{\langle \mathbf{v}, \mathbf{v} \rangle} \leq Ch^2 \tag{1.115}$$

for all $\mathbf{v} \in \mathbb{R}^n$. Here $\underline{h} = \min_{\triangle_k \in \mathcal{T}_h} h_k$ and $h = \max_{\triangle_k \in \mathcal{T}_h} h_k$. The constants c and C are independent of both $\underline{h}$ and h.

Proof See Problem 1.35. □

The bound (1.115) can be further refined by making the assumption that the subdivision is *quasi-uniform*.

Definition 1.31 (Quasi-uniform subdivision). A sequence of triangular grids $\{\mathcal{T}_h\}$ is said to be *quasi-uniform* if there exists a constant $\rho > 0$ such that $\underline{h} \geq \rho h$ for every grid in the sequence.

For a quasi-uniform subdivision in $\mathbb{R}^2$ of shape-regular elements, the bound (1.115) simplifies:

$$ch^2 \leq \frac{\langle Q\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle} \leq Ch^2 \quad \text{for all } \mathbf{v} \in \mathbb{R}^n. \tag{1.116}$$

Remark 1.32. If the subdivision is quasi-uniform, then the bound (1.116) holds for any degree of approximation, $\boldsymbol{P}_m$, $\boldsymbol{Q}_m$ with $m \geq 2$ (see Problem 1.36). However, the constants c and C depend on m.

The bound (1.116) depends on the spatial dimension. For tetrahedral or brick elements on a quasi-uniform discretization of a domain in $\mathbb{R}^3$, the corresponding bound is

$$ch^3 \leq \frac{\langle Q\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle} \leq Ch^3 \quad \text{for all } \mathbf{v} \in \mathbb{R}^n. \tag{1.117}$$

The mass matrix is a fundamental component of finite element analysis, arising naturally, for example, in the study of unsteady problems in Chapter 10. Here, however, it is only making a "cameo appearance" for the purposes of developing bounds on the eigenvalues of the discrete Laplacian A; see Problem 1.33. The mass matrix will also play a key role in Chapters 4 and 9.

One other property of the mass matrix will be useful in the next chapter. Given a Poisson problem (1.13), (1.14) and an approximation space S_E^h, the finite element solution u_h in S_E^h satisfying (1.19) is identical to that with the source function f replaced by its projection $f_h \in S_0^h$ with respect to the L_2 norm. This is simply because f_h so defined satisfies $(f - f_h, w_h) = 0$ for every $w_h \in S_0^h$, so that

$$\int_\Omega \phi_i \, f_h = \int_\Omega \phi_i \, f \tag{1.118}$$

for each i, and thus there is no change in (1.23) when f_h is used instead of f. Note that if f_h in (1.118) is expressed in terms of the basis set $\{\phi_i\}_{i=1}^n$, then the coefficients are determined by solving the linear system $Q\mathbf{x} = \mathbf{f}$, where Q is the mass matrix; see Problem 1.32.

Another fundamental concept used for the analysis of matrix computations is the *condition number* of a matrix,

$$\kappa = \kappa(A) := \|A\| \, \|A^{-1}\|,$$

where the matrix norm is

$$\|A\| := \max_{\mathbf{v}\neq 0} \frac{\|A\mathbf{v}\|}{\|\mathbf{v}\|}.$$

When A is a symmetric positive-definite matrix, $\|A\| = \lambda_{\max}(A)$, the largest eigenvalue of A, and $\|A^{-1}\| = 1/\lambda_{\min}(A)$. Consequently, the condition number is

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

For direct solution methods, the size of the condition number is usually related to the number of accurate decimal places in a computed solution (see Higham [130]). In the next chapter, the convergence behavior of iterative solution methods will be precisely characterized in terms of $\kappa(A)$. In anticipation of this development, bounds on the condition number of the discrete Laplacian A are derived here. Two alternative approaches that can be used to establish such bounds will be described.

The first approach uses tools developed in Section 1.5 and is applicable in the case of arbitrarily shaped domains and nonuniform grids.

Theorem 1.33. *For $\boldsymbol{P}_1$ or $\boldsymbol{Q}_1$ approximation on a shape-regular, quasi-uniform subdivision of $\mathbb{R}^2$, the Galerkin matrix A in* (1.21) *satisfies*

$$ch^2 \leq \frac{\langle A\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle} \leq C \quad \textit{for all } \mathbf{v} \in \mathbb{R}^n. \tag{1.119}$$

Here h is the length of the longest edge in the mesh or grid, and c and C are positive constants that are independent of h. In terms of the condition number, $\kappa(A) \leq C_* h^{-2}$ where $C_* = C/c$.

Proof Suppose that λ is an eigenvalue of A, that is, $A\mathbf{v} = \lambda\mathbf{v}$ for some eigenvector $\mathbf{v}$. Then $\lambda = \langle A\mathbf{v}, \mathbf{v}\rangle/\langle \mathbf{v}, \mathbf{v}\rangle$, and it follows that

$$\min_{\mathbf{v}\in\mathbb{R}^n} \frac{\langle A\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle} \leq \lambda \leq \max_{\mathbf{v}\in\mathbb{R}^n} \frac{\langle A\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle}. \tag{1.120}$$

For any $\mathbf{v} \in \mathbb{R}^n$, let v_h denote the corresponding function in S_0^h. The Poincaré–Friedrichs inequality (Lemma 1.2) implies that there is a constant c_Ω that is independent of the mesh parameter h such that

$$c_\Omega \|v_h\|^2 \leq \|\nabla v_h\|^2 = a(v_h, v_h)$$

for all $v_h \in S_0^h$. Rewriting in terms of matrices gives

$$c_\Omega \langle Q\mathbf{v}, \mathbf{v}\rangle \leq \langle A\mathbf{v}, \mathbf{v}\rangle \quad \text{for all } \mathbf{v} \in \mathbb{R}^n. \tag{1.121}$$

Combining the left-hand inequality of (1.116) and the characterization (1.120) shows that the smallest eigenvalue of A is bounded below by a quantity of order h^2.

For a bound on the largest eigenvalue of A, we turn to the local inverse estimate derived in Lemma 1.27, which states that for the restriction of v_h to an element T,

$$\|\nabla v_h\|_T^2 \leq C h_T^{-2} \|v_h\|_T^2 .$$

Summing over all the elements and using the quasi-uniformity bound $h_T^{-1} \leq Ch^{-1}$ together with the right-hand inequality of (1.116) gives

$$\langle A\mathbf{v}, \mathbf{v}\rangle = a(v_h, v_h) \leq Ch^{-2}\|v_h\|^2 \leq C\,\langle \mathbf{v}, \mathbf{v}\rangle.$$

Thus, the bound on the largest eigenvalue is independent of h. □

Remark 1.34. The Galerkin matrix bound (1.119) holds for any degree of approximation $\boldsymbol{P}_m$, $\boldsymbol{Q}_m$ with $m \geq 2$. The constants c and C depend on m.

Remark 1.35. With tetrahedral or brick elements on a quasi-uniform discretization of a domain in $\mathbb{R}^3$, the corresponding bound is

$$ch^3 \leq \frac{\langle A\mathbf{v}, \mathbf{v}\rangle}{\langle \mathbf{v}, \mathbf{v}\rangle} \leq Ch \quad \text{for all } \mathbf{v} \in \mathbb{R}^n. \tag{1.122}$$

This leads to an identical bound, $\kappa(A) \leq C_* h^{-2}$, on the condition number of the discrete Laplacian in arbitrary dimensions.

The second way of obtaining eigenvalue bounds is with Fourier analysis. This avoids the use of functional analytic tools, but the assumptions on the mesh are more restrictive than those expressed in Theorem 1.33.

A typical result is worked out in Problem 1.37. We use a double index notation to refer to the nodes ordered in a so-called lexicographic order as illustrated in Figure 1.22. Consider the concrete case of Example 1.1.1 discretized using $\boldsymbol{Q}_1$ approximation; Ω is a square of size $2(= L)$, and a uniform $k \times k$ grid is used, so that the matrix dimension is $n = (k-1)^2$ with $k = L/h$. The analysis leads to the explicit identification of *all* of the eigenvalues:

$$\lambda^{(r,s)} = \frac{8}{3} - \frac{2}{3}\left(\cos\frac{r\pi}{k} + \cos\frac{s\pi}{k}\right) - \frac{4}{3}\cos\frac{r\pi}{k}\cos\frac{s\pi}{k}, \quad r, s = 1, \ldots, k-1, \tag{1.123}$$

together with the associated eigenvectors $\mathbf{U}^{(r,s)}$:

$$\mathbf{U}_{i,j}^{(r,s)} = \sin\frac{ri\pi}{k}\sin\frac{sj\pi}{k}, \tag{1.124}$$

where the index $i, j = 1, \ldots, k-1$ refers to the grid location.

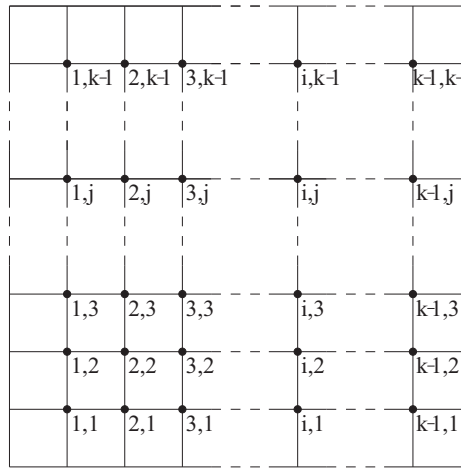


FIG. 1.22. Lexicographic ordering of node points with an double index.

From (1.123), we see that the extreme eigenvalues of the $\boldsymbol{Q}_1$ discrete Laplacian are thus

$$\lambda_{\min} = \lambda^{(1,1)} = \frac{8}{3} - \frac{4}{3}\cos\frac{\pi}{k} - \frac{4}{3}\cos^2\frac{\pi}{k} = \frac{2\pi^2}{L^2}h^2 + O(h^4),$$
$$\lambda_{\max} = \lambda^{(1,k-1)} = \lambda^{(k-1,1)} = \frac{8}{3} + \frac{4}{3}\cos^2\frac{\pi}{k} = 4 - \frac{4\pi^2}{3L^2}h^2 + O(h^4),$$

and the condition number is

$$\kappa(A) = \frac{2L^2}{\pi^2}h^{-2} - \frac{1}{6} + O(h^2). \tag{1.125}$$

Notice that the bound of Theorem 1.33 is tight in this case. (See also Computational Exercise 1.7.) Analogous estimates can also be established in the three-dimensional case; see Problem 1.38. Fourier analysis will be used in later chapters to give insight in other contexts, for example, to explore the convergence properties of multigrid methods (Section 2.5), and to investigate discrete approximations that exhibit high-frequency oscillations in cases where the continuous solution is not oscillatory (Section 6.5).

Problems

1.1. By changing to polar coordinates in $\mathbb{R}^2$ show that Laplace's equation takes the form

$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} = 0.$$

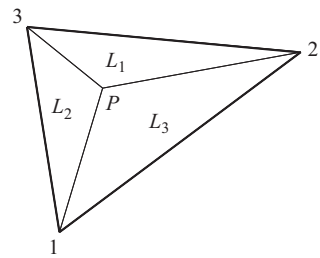
Hence show that the function $u(r,\theta) = r^{2/3}\sin((2\theta+\pi)/3)$ satisfies Laplace's equation in $\mathbb{R}^2$.

1.2. Show that any function u satisfying the Poisson equation together with a Robin boundary condition, $\alpha u + \frac{\partial u}{\partial n} = 0$ on $\partial\Omega$, where $\alpha > 0$ is a constant, also satisfies the following weak formulation: find $u \in \mathcal{H}^1(\Omega)$ such that

$$\int_\Omega \nabla u \cdot \nabla v + \alpha \int_{\partial\Omega} u\, v \,\mathrm{d}s = \int_\Omega v f \quad \text{for all } v \in \mathcal{H}^1(\Omega). \qquad (V)$$

Show that $c(u,v) := \int_\Omega \nabla u \cdot \nabla v + \alpha \int_{\partial\Omega} uv \,\mathrm{d}s$ defines an *inner product* over $\mathcal{H}^1(\Omega)$. Deduce that the solution of the weak formulation (V) is unique.

1.3. A generic point P in a triangle is parameterized by three triangular (or *barycentric*) coordinates L_1, L_2 and L_3, which are simple ratios of the triangle areas illustrated $(L_1, L_2, L_3) \equiv (|\triangle_{3P2}|/|\triangle|, |\triangle_{1P3}|/|\triangle|, |\triangle_{2P1}|/|\triangle|)$.



For a general triangle with vertices (x_1,y_1), (x_2,y_2) and (x_3,y_3), the barycentric coordinates (L_1,L_2,L_3) can also be defined by the linear mapping

$$\begin{aligned} x &= x_1L_1 + x_2L_2 + x_3L_3,\\ y &= y_1L_1 + y_2L_2 + y_3L_3,\\ 1 &= L_1 + L_2 + L_3. \end{aligned}$$

By inverting this mapping, show that $L_i = \frac{1}{2|\triangle|}(a_i + b_i x + c_i y)$, $i = 1,2,3$, where a_i satisfies $\sum_{i=1}^3 a_i = 2|\triangle|$ with $|\triangle|$ the area of the triangle, and with coefficients b_i and c_i given by (1.43).

1.4. A superlinear approximation $u^{\textcircled{k}} = aL_1 + bL_2 + cL_3 + dL_1L_2L_3$, where (L_1,L_2,L_3) are barycentric coordinates, is characterized by the local basis set $\{N_i\}_{i=1}^4$ satisfying the standard interpolation property,

$$N_i(x,y) = \begin{cases} 1 & \text{at node } i,\\ 0 & \text{at the other three nodes.} \end{cases}$$

Derive the form of the four basis functions if the nodes correspond to the three vertices and the centroid of the triangle.
By evaluating the interpolant of two triangles sharing a common edge, PQ say, show that the concatenated global interpolant is continuous. Is the normal derivative of the global interpolant also continuous?

1.5. A linear approximation $u^{\textcircled{k}} = aL_1 + bL_2 + cL_3$, where (L_1, L_2, L_3) are barycentric coordinates, is characterized by the $\boldsymbol{P}_1$ basis set $\{N_i\}_{i=1}^3$ satisfying the standard interpolation property,

$$N_i(x, y) = \begin{cases} 1 & \text{at node } i, \\ 0 & \text{at the other two nodes.} \end{cases}$$

Derive the form of the three basis functions if the nodes correspond to the three midedge points of the triangle.
By evaluating the interpolant of two triangles sharing a common edge, PQ say, show that the concatenated global interpolant is not continuous.

1.6. A quadratic approximation $u^{\textcircled{k}} = aL_1 + bL_2 + cL_3 + dL_1L_2 + eL_1L_3 + fL_2L_3$, where (L_1, L_2, L_3) are barycentric coordinates, is characterized by the $\boldsymbol{P}_2$ basis set $\{N_i\}_{i=1}^6$ satisfying the standard interpolation property,

$$N_i(x, y) = \begin{cases} 1 & \text{at node } i, \\ 0 & \text{at the other five nodes.} \end{cases}$$

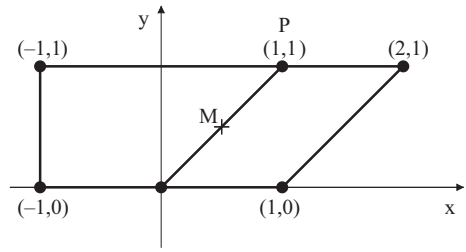
Derive the form of the six basis functions if the nodes correspond to the three vertices and the three midedges of the triangle.
By evaluating the interpolant of two triangles sharing a common edge, PQ say, show that the concatenated global interpolant is continuous. Is the normal derivative of the global interpolant also continuous?

1.7. Verify that the bilinear mapping to the general quadrilateral shown on the left of Figure 1.10 is given by

$$\begin{aligned} x &= x_1\chi_1 + x_2\chi_2 + x_3\chi_3 + x_4\chi_4, \\ y &= y_1\chi_1 + y_2\chi_2 + y_3\chi_3 + y_4\chi_4. \end{aligned}$$

Show that horizontal straight lines $\eta = C$ are mapped onto straight lines in (x, y) coordinates.

1.8. For the pair of elements illustrated below, show that the bilinear function that takes the value 1 at vertex P and 0 at the other vertices gives different values at the midpoint M on the common edge (and is hence discontinuous).



Show that the corresponding isoparametrically mapped bilinear function (defined via (1.26) and (1.27)) is continuous along the common edge.

1.9. This problem builds on Problem 1.3. By substituting (1.36) into (1.45), show that the $\boldsymbol{P}_1$ stiffness matrix is given by

$$A^{\textcircled{k}}(i,j) = \frac{1}{4|\triangle_k|}(b_i b_j + c_i c_j),$$

where b_i and c_i, $i = 1, 2, 3$ are the coefficients given in (1.43).

1.10. Compare the element trial functions along the common face, PQR say, of two *tetrahedral* elements with $\boldsymbol{P}_1$ approximation and hence show that the global trial function is continuous across interelement boundaries. Are the normal and tangential derivatives of the global trial function also continuous across the common face?

1.11. Show that the determinant of the $\boldsymbol{Q}_1$ Jacobian matrix (1.48) is a *linear* function of the coordinates (ξ, η).

1.12. Show that if a mapped $\boldsymbol{Q}_1$ element is a parallelogram with vertices (x_0, y_0), $(x_0 + h_x, y_1)$, $(x_1 + h_x, y_1 + h_y)$ and $(x_1, y_0 + h_y)$, then the Jacobian (1.48) is a constant matrix.

1.13. Consider a rectangular element $\square_k$, with horizontal and vertical edges of lengths h_x, h_y, respectively. Show that the Jacobian matrix (1.48) associated with the mapping to $\square_k$ from the reference element $\square_*$ is the diagonal matrix

$$J_k = \frac{1}{2}\begin{pmatrix} h_x & 0 \\ 0 & h_y \end{pmatrix}.$$

By mapping the derivatives of the reference element basis functions, show that the $\boldsymbol{Q}_1$ stiffness matrix associated with $\square_k$ is given by

$$A^{\textcircled{k}} = \begin{bmatrix} \frac{1}{3}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & -\frac{1}{3}\frac{h_y}{h_x} + \frac{1}{6}\frac{h_x}{h_y} & -\frac{1}{6}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & \frac{1}{6}\frac{h_y}{h_x} - \frac{1}{3}\frac{h_x}{h_y} \\ -\frac{1}{3}\frac{h_y}{h_x} + \frac{1}{6}\frac{h_x}{h_y} & \frac{1}{3}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & -\frac{1}{3}\frac{h_x}{h_y} + \frac{1}{6}\frac{h_y}{h_x} & -\frac{1}{6}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) \\ -\frac{1}{6}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & -\frac{1}{3}\frac{h_x}{h_y} + \frac{1}{6}\frac{h_y}{h_x} & \frac{1}{3}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & -\frac{1}{3}\frac{h_y}{h_x} + \frac{1}{6}\frac{h_x}{h_y} \\ \frac{1}{6}\frac{h_y}{h_x} - \frac{1}{3}\frac{h_x}{h_y} & -\frac{1}{6}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) & -\frac{1}{3}\frac{h_y}{h_x} + \frac{1}{6}\frac{h_x}{h_y} & \frac{1}{3}\left(\frac{h_y}{h_x} + \frac{h_x}{h_y}\right) \end{bmatrix}.$$

1.14. Solve the problem in Example 1.1.1, but exploit the symmetry by taking a square domain $\Omega = (0,1)\times(0,1)$ and supposing that $\partial\Omega_D$ is given by edges $\{1\}\times[0,1]\cup[0,1]\times\{1\}$. Subdivide into four square elements with $\boldsymbol{Q}_1$ approximation so that the element matrices take the form above with $h_x = \frac{1}{2} = h_y$. Then, assemble the element matrices and compute the finite element solution. (You can use IFISS to check your results by simply solving the problem in Computational Exercise 1.3 using a 4×4 grid.)

1.15. Imagine solving the problem in Example 1.1.1 using a *triangular* mesh obtained by taking a uniform square subdivision of size h and then dividing each square into two triangles in a fixed direction. Compute the 4×4 matrix that results from assembly of the linear element matrices from the two subtriangles. You should find that the diagonal entries of this matrix are different from those in Problem 1.13. (You cannot create a bilinear approximation on a square by simply combining two linear approximations!)

1.16. Given the Gauss points $\xi_s = \pm 1/\sqrt{3}$ and $\eta_t = \pm 1/\sqrt{3}$ as illustrated in Figure 1.15, show that if f is bilinear, that is, $f(\xi,\eta) = (a+b\xi)(c+d\eta)$, where a, b, c and d are constants, then

$$\int_{-1}^{1}\int_{-1}^{1} f\,\mathrm{d}\xi\,\mathrm{d}\eta = \sum_s \sum_t f(\xi_s,\eta_t).$$

Give an example of a polynomial function which is *not* integrated exactly by this Gauss rule but is integrated exactly by the 3×3 Gauss rule shown in Figure 1.15.

1.17. Show that if $\int_{\partial\Omega_D} \mathrm{d}s \neq 0$ then the bilinear form $a(\cdot,\cdot)$ in (1.55) defines an *inner product* over the space $\mathcal{H}^1_{E_0}\times\mathcal{H}^1_{E_0}$.

1.18. Show that, if u and u_h satisfy (1.56) and (1.57), respectively, in the case of zero Dirichlet data (so that $\mathcal{H}^1_E = \mathcal{H}^1_{E_0}$), then the error in energy satisfies

$$\|\nabla(u-u_h)\|^2 = \|\nabla u\|^2 - \|\nabla u_h\|^2 .$$

(Hint: this is easy. Just use Galerkin orthogonality.)

1.19. Given a square domain $\Omega = [0,L]\times[0,L]$, show that

$$\int_\Omega u^2 \leq \frac{L^2}{2}\int_\Omega \left(\left|\frac{\partial u}{\partial x}\right|^2 + \left|\frac{\partial u}{\partial y}\right|^2 \right)$$

for any function $u\in\mathcal{H}^1(\Omega)$ that is zero everywhere on the boundary. (Hint: $u(x,y) = u(0,y) + \int_0^x \frac{\partial u}{\partial x}(\xi,y)\,\mathrm{d}\xi$.)

1.20. Prove Proposition 1.6. (Hint: use the definition (1.63).)

1.21. Let V_h be a finite element subspace of $V := \mathcal{H}^1(\Omega)$. Define a bilinear form $a(\cdot,\cdot)$ on $V \times V$, and let $u \in V$ and $u_h \in V_h$ satisfy

$$\begin{aligned} a(u,v) &= (f,v) \quad \text{for all } v \in V, \\ a(u_h,v_h) &= (f,v_h) \quad \text{for all } v_h \in V_h, \end{aligned}$$

respectively. If there exist positive constants γ and Γ such that

$$\begin{aligned} a(v,v) &\geq \gamma \|v\|_{1,\Omega}^2 \quad \text{for all } v \in V, \\ |a(u,v)| &\leq \Gamma \|u\|_{1,\Omega} \|v\|_{1,\Omega} \quad \text{for all } u,v \in V, \end{aligned}$$

show that there exists a positive constant $C(\gamma,\Gamma)$ such that

$$\|u-u_h\|_{1,\Omega} \leq C \inf_{v_h \in V_h} \|u-v_h\|_{1,\Omega}.$$

1.22. For any function $w(x)$ defined on $[0,1]$, let Πw be the linear interpolant satisfying $\Pi w(0) = w(0)$ and $\Pi w(1) = w(1)$. Use Rolle's theorem to show that $e = w - \Pi w$ satisfies

$$\int_0^1 (e')^2 \mathrm{d}x \leq \int_0^1 (e'')^2 \mathrm{d}x.$$

(This establishes the Bramble–Hilbert lemma in one space dimension.)

1.23. Prove Proposition 1.14. (Hint: use simple trigonometric identities.)

1.24. Prove Proposition 1.16.

1.25. Prove Proposition 1.17. (Hint: follow the proofs of Lemma 1.11 and Lemma 1.13.)

1.26. Prove Proposition 1.22. (Hint: do Problem 1.25 first.)

1.27. Given that $u \in \mathcal{H}^s(\Omega)$ if and only if $\|D^s u\| < \infty$, show that the function $u(r,\theta) = r^{2/3}\sin((2\theta+\pi)/3)$ defined on the pie-shaped domain Ω, where $0 \leq r \leq 1$ and $-\pi/2 \leq \theta \leq \pi$ is in $\mathcal{H}^1(\Omega)$, but is not in $\mathcal{H}^2(\Omega)$. (Hint: work in polar coordinates, and note that $\mathrm{d}x\,\mathrm{d}y = r\mathrm{d}r\,\mathrm{d}\theta$.)

1.28. Show that if $\mathcal{Q}_T$ is the space (1.97) of edge and bubble functions then there exists a unique solution to the local problem (1.94).

1.29. Show that the inverse estimate (1.104) holds in the case of a rectangular element T.

1.30. Prove Proposition 1.29. (Hint: take $v = e_T$ in the local problem (1.94), and then choose w_T and w_E as in the proof of Theorem 1.24 with a view to separately bounding the interior and jump residual terms by $\|\nabla(u-u_h)\|_T$.)

1.31. Let V_h be a $\boldsymbol{P}_1$ finite element space on a triangulation T_h of a domain Ω. Given $u \in H^2(\Omega)$ let $u_h \in V_h$ be the solution of the boundary value problem

$$\int_\Omega u_h v = \int_\Omega uv \qquad \text{for all } v \in V_h.$$

Thus, u_h is the L_2 projection of u onto V_h. Establish the error estimate

$$\|u - u_h\| \leq \inf_{v_h \in V_h} \|u - v_h\|$$

and show that $\|u_h\| \leq \|u\|$.

1.32. This builds on Problem 1.31. For any $f \in L_2(\Omega)$, let f_h be the L_2 projection onto V_h . Writing $f_h = \sum_{j=1}^n \bar{\boldsymbol{f}}_j \phi_j$, show that the coefficient vector $\bar{\mathbf{f}} = (\bar{\boldsymbol{f}}_1, \bar{\boldsymbol{f}}_2, \ldots, \bar{\boldsymbol{f}}_n)^T$ is the solution of

$$Q\bar{\mathbf{f}} = \mathbf{f},$$

where Q is the mass matrix (1.114) and $\mathbf{f} = [\boldsymbol{f}_i]$ with $\boldsymbol{f}_i = \int_\Omega \phi_i f$.

1.33. Consider the differential problem (E) defined by the eigenvalue equation

$$-\nabla^2 u = \lambda u, \qquad (x, y) \text{ in } \Omega,$$

together with the Dirichlet boundary condition $u = 0$ on $\partial\Omega$. Given the function space

$$\mathcal{H}^1_{E_0} = \big\{v | v \in \mathcal{H}^1(\Omega);\ v = 0 \text{ on } \partial\Omega\big\},$$

show that u satisfying (E) also satisfies a variational formulation (M): find the pair $(\lambda \in \mathbb{R}, u \in \mathcal{H}^1_{E_0})$ such that

$$\int_\Omega \nabla u \cdot \nabla v = \lambda \int_\Omega uv \qquad \text{for all } v \in \mathcal{H}^1_{E_0}.$$

Show that any eigenvalue $\lambda \in \mathbb{R}$ satisfying (M) must satisfy $\lambda > 0$.

Next, given some finite-dimensional subspace $S^h_0 \subset \mathcal{H}^1_{E_0}$ with a basis set $\{\phi_j\}_{j=1}^k$, show that the Galerkin solution is given by solving the generalized eigenvalue problem (M_h):

$$A\mathbf{x} = \lambda_h Q\mathbf{x},$$

where A is the $k \times k$ matrix in (1.22) and Q is the $k \times k$ matrix in (1.114).

1.34. This builds on Problem 1.33. The matrix Q is clearly symmetric. By writing $u_h = \sum_{j=1}^k \mathbf{u}_j \phi_j$ deduce that

$$\|u_h\|^2 = \langle Q\mathbf{u}, \mathbf{u}\rangle$$

where $\langle \mathbf{u}, \mathbf{v}\rangle$ is the usual scalar product for vectors $\mathbf{u} \in \mathbb{R}^k$, $\mathbf{v} \in \mathbb{R}^k$. Hence show that Q is strictly positive definite. (Since A is also symmetric and positive definite, this ensures that λ_h solving (M_h) is real and positive.)

1.35. Prove Proposition 1.30. (Hint: start from Problem 1.34 and write $\langle Q\mathbf{u},\mathbf{u}\rangle = \sum_{k\in\mathcal{T}_h}\langle Q^{\textcircled{k}}\mathbf{v}_k,\mathbf{v}_k\rangle$, where $Q^{\textcircled{k}}$ is the element mass matrix for element k, that is, $Q^{\textcircled{k}} = [q_{ij}]$ with $q_{ij} = \int_{\Delta_k}\phi_j\phi_i$. Then prove that for a shape-regular element, $\underline{c}h_k^2\langle\mathbf{v}_k,\mathbf{v}_k\rangle \le \langle Q^{\textcircled{k}}\mathbf{v}_k,\mathbf{v}_k\rangle \le \bar{c}h_k^2\langle\mathbf{v}_k,\mathbf{v}_k\rangle$ for all functions $\mathbf{v}_k$.)

1.36. Prove that the mass matrix bound (1.116) holds for $\boldsymbol{Q}_2$ approximation on a uniform grid of square elements.

1.37. In the double index notation indicated by Figure 1.22 and with $\mathbf{U}_{i,j}$ denoting the value of u_h at the lattice point i,j, the Galerkin system of equations derived from $\boldsymbol{Q}_1$ approximation on a uniform square grid can be written as

$$\begin{aligned}&\frac{8}{3}\mathbf{U}_{i,j} - \frac{1}{3}\mathbf{U}_{i+1,j+1} - \frac{1}{3}\mathbf{U}_{i+1,j} - \frac{1}{3}\mathbf{U}_{i+1,j-1} - \frac{1}{3}\mathbf{U}_{i,j+1} - \frac{1}{3}\mathbf{U}_{i,j-1}\\ &\quad - \frac{1}{3}\mathbf{U}_{i-1,j+1} - \frac{1}{3}\mathbf{U}_{i-1,j} - \frac{1}{3}\mathbf{U}_{i-1,j-1} = h^2\mathbf{f}_{i,j}\end{aligned}$$

with $\mathbf{U}_{k,j}, \mathbf{U}_{0,j}, \mathbf{U}_{i,0}, \mathbf{U}_{i,k}$ given by the Dirichlet boundary condition. The eigenvalues $\lambda^{r,s}$ of the Galerkin matrix therefore satisfy

$$\begin{aligned}&\frac{8}{3}\mathbf{U}^{r,s}_{i,j} - \frac{1}{3}\mathbf{U}^{r,s}_{i+1,j+1} - \frac{1}{3}\mathbf{U}^{r,s}_{i+1,j} - \frac{1}{3}\mathbf{U}^{r,s}_{i+1,j-1} - \frac{1}{3}\mathbf{U}^{r,s}_{i,j+1} - \frac{1}{3}\mathbf{U}^{r,s}_{i,j-1}\\ &\quad - \frac{1}{3}\mathbf{U}^{r,s}_{i-1,j+1} - \frac{1}{3}\mathbf{U}^{r,s}_{i-1,j} - \frac{1}{3}\mathbf{U}^{r,s}_{i-1,j-1} = \lambda^{r,s}\,\mathbf{U}^{r,s}_{i,j}\end{aligned}$$

for $r,s = 1,\dots,k-1$. Verify that the vector $\mathbf{U}^{r,s}$ with entries

$$\mathbf{U}^{r,s}_{i,j} = \sin\frac{ri\pi}{k}\sin\frac{sj\pi}{k}, \qquad i,j = 1,\dots,k-1$$

is an eigenvector for arbitrary $r,s = 1,\dots,k-1$, and hence that the corresponding eigenvalue is

$$\lambda^{r,s} = \frac{8}{3} - \frac{2}{3}\left(\cos\frac{r\pi}{k} + \cos\frac{s\pi}{k}\right) - \frac{4}{3}\cos\frac{r\pi}{k}\cos\frac{s\pi}{k}, \qquad r,s = 1,\dots,k-1.$$

1.38. In triple index notation with $\mathbf{U}_{i,j,k}$ denoting the value of u_h at the lattice point $i,j,k = 1,\dots,l-1$, show that the Galerkin system derived from trilinear approximation of the Poisson equation with Dirichlet boundary conditions on a uniform grid of cube elements with side length h can be written as

$$\frac{8h}{3}\mathbf{U}_{i,j,k} - \frac{h}{6}\left(\mathbf{U}_{i,j+1,k-1} + \mathbf{U}_{i,j-1,k-1} + \mathbf{U}_{i+1,j,k-1} + \mathbf{U}_{i-1,j,k-1}\right)$$
$$- \frac{h}{12}\left(\mathbf{U}_{i+1,j+1,k-1} + \mathbf{U}_{i+1,j-1,k-1} + \mathbf{U}_{i-1,j+1,k-1} + \mathbf{U}_{i-1,j-1,k-1}\right)$$
$$- \frac{h}{6}\left(\mathbf{U}_{i+1,j+1,k} + \mathbf{U}_{i+1,j-1,k} + \mathbf{U}_{i-1,j+1,k} + \mathbf{U}_{i-1,j-1,k}\right)$$
$$- \frac{h}{12}\left(\mathbf{U}_{i+1,j+1,k+1} + \mathbf{U}_{i+1,j-1,k+1} + \mathbf{U}_{i-1,j+1,k+1} + \mathbf{U}_{i-1,j-1,k+1}\right)$$
$$- \frac{h}{6}\left(\mathbf{U}_{i,j+1,k+1} + \mathbf{U}_{i,j-1,k+1} + \mathbf{U}_{i+1,j,k+1} + \mathbf{U}_{i-1,j,k+1}\right) = h^2\mathbf{f}_{i,j,k}$$

for $i, j, k = 1, \ldots, l-1$, with $\mathbf{U}_{i,j,k}$ given by the Dirichlet boundary conditions when any of i, j or k is 0 or l.

1.39. This builds on Problems 1.37 and 1.38. Show that

$$\mathbf{U}^{r,s,t}_{i,j,k} = \sin\frac{ri\pi}{l}\sin\frac{sj\pi}{l}\sin\frac{tk\pi}{l}, \qquad i,j,k = 1,\ldots,l-1$$

is an eigenvector of the Galerkin matrix in Problem 1.38 for $r, s, t = 1, \ldots, l-1$, and that the eigenvalues $\lambda^{r,s,t}$ satisfying

$$\frac{8h}{3}\mathbf{U}^{r,s,t}_{i,j,k} - \frac{h}{6}\left(\mathbf{U}^{r,s,t}_{i,j+1,k-1} + \mathbf{U}^{r,s,t}_{i,j-1,k-1} + \mathbf{U}^{r,s,t}_{i+1,j,k-1} + \mathbf{U}^{r,s,t}_{i-1,j,k-1}\right)$$
$$- \frac{h}{12}\left(\mathbf{U}^{r,s,t}_{i+1,j+1,k-1} + \mathbf{U}^{r,s,t}_{i+1,j-1,k-1} + \mathbf{U}^{r,s,t}_{i-1,j+1,k-1} + \mathbf{U}^{r,s,t}_{i-1,j-1,k-1}\right)$$
$$- \frac{h}{6}\left(\mathbf{U}^{r,s,t}_{i+1,j+1,k} + \mathbf{U}^{r,s,t}_{i+1,j-1,k} + \mathbf{U}^{r,s,t}_{i-1,j+1,k} + \mathbf{U}^{r,s,t}_{i-1,j-1,k}\right)$$
$$- \frac{h}{12}\left(\mathbf{U}^{r,s,t}_{i+1,j+1,k+1} + \mathbf{U}^{r,s,t}_{i+1,j-1,k+1} + \mathbf{U}^{r,s,t}_{i-1,j+1,k+1} + \mathbf{U}^{r,s,t}_{i-1,j-1,k+1}\right)$$
$$- \frac{h}{6}\left(\mathbf{U}^{r,s,t}_{i,j+1,k+1} + \mathbf{U}^{r,s,t}_{i,j-1,k+1} + \mathbf{U}^{r,s,t}_{i+1,j,k+1} + \mathbf{U}^{r,s,t}_{i-1,j,k+1}\right) = \lambda^{r,s,t}\,\mathbf{U}^{r,s,t}_{i,j,k}$$

are therefore

$$\lambda^{r,s,t} = \frac{8h}{3} - \frac{2h}{3}\left(\cos\frac{r\pi}{l}\cos\frac{s\pi}{l} + \cos\frac{r\pi}{l}\cos\frac{t\pi}{l} + \cos\frac{s\pi}{l}\cos\frac{t\pi}{l}\right)$$
$$- \frac{2h}{3}\cos\frac{r\pi}{l}\cos\frac{s\pi}{l}\cos\frac{t\pi}{l}, \qquad r,s,t = 1,\ldots,l-1.$$

Computational exercises

Two specific domains are built into IFISS by default, $\Omega_\square \equiv (-1,1)\times(-1,1)$ and $\Omega_\lrcorner \equiv \Omega_\square\backslash\{(-1,0)\times(-1,0)\}$. Numerical solutions to a Dirichlet problem defined on $\Omega_\square$ or $\Omega_\lrcorner$ can be computed by running `square_diff` or `ell_diff` as appropriate, with source data f and boundary data g specified in function m-files `./diffusion/specific_rhs` and `./diffusion/specific_bc`, respectively. Running the driver `diff_testproblem` sets up the data files `specific_rhs` and `specific_bc` associated with the reference problems in Examples 1.1.1–1.1.4.

1.1. Consider Example 1.1.1 with a typical solution illustrated in Figure 1.1. Tabulate the a posteriori error estimate η (the variable `errorest`) when using $\boldsymbol{Q}_1$ approximation on a sequence of uniform 16×16, 32×32 and 64×64 grids, and thus estimate the order of convergence of the approximation in the energy norm. Next, repeat the experiment using $\boldsymbol{Q}_2$ approximation. You should find that the order of convergence is increased.

1.2. This builds on Computational Exercise 1.1 but is significantly more demanding since it requires programming expertise. Optionally, fine grid reference solutions can be saved when running `square_diff`. Your task is to write a `matlab` function that postprocesses a reference fine grid solution $u_\star$ and compares it with a computed coarse grid solution u_h. (Hint: use the `matlab` function `interp2` to construct the cubic interpolant $\Pi_h^\star u_h$ of the coarse grid solution at the reference grid nodes and then compute the following measure of the coarse grid solution error:

$$E_h = \|\nabla(\Pi_h^\star u_h - u_h)\| \approx \|\nabla(u - u_h)\|.$$

Second hint: this involves a matrix–vector multiply with the fine grid Galerkin matrix `Aref`; see (1.113).) Use your function to assess the quality of the error estimator η by repeating the computations made in Computational Exercise 1.1 and comparing with a reference $\boldsymbol{Q}_2$ solution computed on a 128×128 grid.

1.3. By truncating and evaluating the series solution (1.5) it can be shown that the maximum value of u solving Example 1.1.1 is $u(0,0) = 0.294685413126$. Tabulate a set of computed approximations $u_h(0,0)$ to $u(0,0)$ using uniform 16×16, 32×32 and 64×64 grids for both $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ approximation. Then, by computing $|u(0,0) - u_h(0,0)|$, find the order of convergence for estimating this specific point value. You might find the result surprising.

1.4. Consider Example 1.1.2 with a typical solution illustrated in Figure 1.2. Tabulate the energy error estimate η for $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ approximation on a sequence of uniform grids, and hence estimate the order of convergence. Then change the source function from $f = 1$ to $f = xy$ and repeat the experiment. Can you explain the difference in the experimental orders of convergence?

1.5. This is a very demanding exercise. Write a `matlab` function that postprocesses a bilinear finite element solution u_h and computes the global error $\|\nabla(u - u_h)\|$ for the analytic test problem in Example 1.1.3. Hence, verify the results given in Table 1.4. Then use your function to generate a table of estimated and exact errors for the set of stretched element grids that is automatically generated by IFISS.

1.6. This exercise illustrates how easy it is to incorporate a homogeneous Neumann boundary condition into a finite element approximation. Consider solving Example 1.1.2 with the "hard" outflow boundary condition $u = 0$ at $x = 1$ replaced by the "symmetry" condition $\frac{\partial u}{\partial x} = 0$. (This can be done by editing

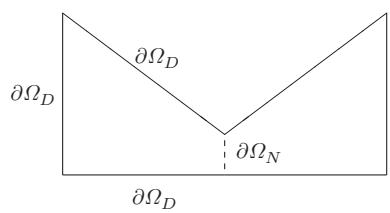
the function `ell_domain` and simply removing the specific entries in the vectors `bound` and `mbound` corresponding to the right-hand boundary edge.) Use the function `ysectionplot` to study the convergence of the finite element solution on the symmetry line for a sequence of increasingly refined uniform grids using $\boldsymbol{Q}_1$ approximation.

1.7. Using the `matlab` function `eig`, compute the eigenvalues of the coefficient matrix for Example 1.1.1 with $k = 8$ using $\boldsymbol{Q}_1$ approximation on a uniform grid, and verify that there are $(k-1)^2$ eigenvalues given by the analytic expression (1.123) together with $4k$ eigenvalues of unity (corresponding to the Dirichlet boundary nodes).

1.8. Use the `matlab` function `eigs` to compute the maximum and minimum eigenvalues of the $\boldsymbol{Q}_1$ stiffness matrix (`Agal`, not `A`) on a sequence of uniformly refined grids, and verify that the condition number grows like $8/(\pi^2 h^2)$ in the limit $h \to 0$. Then, repeat the experiment for the $\boldsymbol{Q}_1$ mass matrix `M`. What do you observe?

1.9. This is a cool exercise that builds on Computational Exercise 1.8. Your task is to write a `matlab` function that calls `eigs` and generates an estimate of the smallest three eigenvalues and associated eigenfunctions of the negative Laplacian operator on $\Omega_{\overline{\nabla}}$. (Hint: look at Problem 1.33 first.) Use your function to generate a table of estimated eigenvalues for a set of uniform square grids using $\boldsymbol{Q}_1$ approximation. Then, repeat the experiment using $\boldsymbol{Q}_2$ approximation and estimate the respective rates of convergence to the fundamental (smallest) eigenvalue.

1.10. Quadrilateral elements are also built into IFISS. Specifically, the function `quad_diff` can be used to solve problems defined on general quadrilateral domains with a Neumann condition on the right-hand boundary. By setting the source function to unity and the Dirichlet boundary data to zero, the effect of geometry on models of the deflection of an elastic membrane stretched over a bow-tie-shaped domain can be explored. (The Neumann condition acts as a symmetry condition, so only half of the bow tie needs to be considered.)



Using the eigenfunction function developed in the previous exercise, determine the smallest eigenvalue of the negative Laplacian defined on the "default" bow-tie domain to an accuracy of three significant digits.