Finite Element Approximation I
MATH46052|66052

David J. Silvester
School of Mathematics, University of Manchester
d.silvester@manchester.ac.uk.

Version 1.1 | 10 May 2018

Contents

1 Smoothness of functions ........................................ 1
  Square integrable function .................................. 1
  Square differentiable function ................................. 3
  Sobolev space .................................................. 3

2 A model PDE problem ............................................ 3
  Classical solution ............................................. 4
  Weak solution .................................................. 5

3 Galerkin approximation ........................................... 6
  Galerkin solution ............................................... 6
  Finite element solution ....................................... 7

4 A worked example ................................................ 8

5 Analysis of the convergence rate ............................... 12
  Higher-order Sobolev space .................................. 13
  Linear convergence ............................................ 13
  Minimum angle condition .................................... 14
  Bramble–Hilbert lemma ....................................... 16
  Quadratic convergence ....................................... 18

6 Estimation of the approximation error ......................... 18
  Local error estimator ......................................... 21

*These notes provide a summary of the essential material from the lectures. Other details can be found in Chapter 1 of the book by Elman, Silvester and Wathen [ESW] (second edition, 2014).
1. Smoothness of functions  This section reviews some important concepts that will be needed to understand the material in subsequent sections of the notes.

(Function) A function maps one or more inputs (real numbers) to a unique output number; for example, \( y = f(x) \) or \( z = f(x, y) \) or \( z = f(x_1, x_2, \ldots, x_n) = f(\vec{x}) \), where \( \vec{x} \) is the \( n \)-dimensional vector \((x_1, x_2, \ldots, x_n)\). The domain of a function is the set \( \Omega \) of all possible input values. For a real-valued function the mapping is conveniently expressed by writing \( f : \Omega \to \mathbb{R} \).

Functions are characterised by their smoothness. The simplest example of a smooth one-dimensional function is one that does not jump when plotted as a graph. Expressed mathematically we have the following characterization.

(Continuous function) A one-dimensional function \( f \) is continuous when
\[
\lim_{\epsilon \to 0} f(x - \epsilon) = f(x) = \lim_{\epsilon \to 0} f(x + \epsilon)
\]
for all possible values of \( x \) in \( \Omega \). In this case we write \( f \in C^0(\Omega) \).

A simple example of a one-dimensional function that is not continuous...

\[
\begin{array}{c|c|c}
 x & f(x) \\
\hline
 -1 & -1 \\
 1 & 1 \\
\end{array}
\]

An alternative way of measuring the smoothness of a function is to consider their integrability. The discontinuous function above is smooth in the sense that it is square integrable.

(Square integrable function) An \( n \)-dimensional function \( f : \Omega \to \mathbb{R} \) is square integrable if and only if
\[
\int_{\Omega} \{f(x_1, x_2, \ldots, x_n)\}^2 \, dx_1 \, dx_2 \ldots \, dx_n < \infty.
\]
In the simple case above we have \( \int_{\Omega} f^2 = \int_{-1}^{0} (-1)^2 \, dx + \int_{0}^{1} (1)^2 \, dx = 2 < \infty \) and we write \( f \in L^2(\Omega) \).
A stronger definition of a smooth one-dimensional function arises when we insist that the two tangents at a given point \( x \) do not jump. Such a function is said to be continuously differentiable.

**Differentiable function**

A continuous function \( f \) is differentiable when

\[
\lim_{\epsilon \to 0} \frac{f(x) - f(x - \epsilon)}{\epsilon} = f'(x) = \lim_{\epsilon \to 0} \frac{f(x + \epsilon) - f(x)}{\epsilon}
\]

for all possible values of \( x \) in \( \Omega \). In this case we write \( f \in C^1(\Omega) \).

A simple example of a continuous function that is not differentiable...

\[
f(x) = \begin{cases} 
-x & \text{if } x < 0 \\
0 & \text{if } x = 0 \\
x & \text{if } x > 0
\end{cases}
\]

Differentiable functions with more than one input variable have more than one tangent at every point \( \vec{x} \) and are associated with partial derivatives.

**Partial derivative**

A two-dimensional function \( f(x, y) \) has a partial derivative whenever

\[
\lim_{\epsilon \to 0} \frac{f(x, y) - f(x - \epsilon, y)}{\epsilon} = f_x = \lim_{\epsilon \to 0} \frac{f(x + \epsilon, y) - f(x, y)}{\epsilon}
\]

(the notation \( f_x \) is an alternative way of writing \( \frac{\partial f}{\partial x} \)). It represents the rate of change with respect to the first variable when the second variable is kept fixed.

**Gradient**

The gradient of a multidimensional function \( f(\vec{x}) \), where \( \vec{x} \) is the \( n \)-dimensional vector \((x_1, x_2, \ldots, x_n)\), is the vector of partial derivatives (usually written as a column vector) given by

\[
\nabla f = \begin{bmatrix} f_{x_1} \\ f_{x_2} \\ \vdots \\ f_{x_n} \end{bmatrix}
\]
The non-differentiable function pictured above is smooth in the sense that it is square integrable over $[-1, 1]$ and it has a square integrable (weak) derivative.

**Square integrable derivative**

An $n$-dimensional function $f : \Omega \to \mathbb{R}$ has a **square integrable derivative** if and only if

$$\int_{\Omega} \nabla f \cdot \nabla f < \infty.$$ 

In this case we write $\nabla f \in (L^2(\Omega))^n$.

Functions that are square integrable with a square integrable derivative are members of a special function space.

**Sobolev space**

The **Sobolev space** $H^1(\Omega)$ consists of functions that are square integrable and have a square integrable derivative:

$$H^1(\Omega) = \{v|v \in L^2(\Omega), \nabla v \in (L^2(\Omega))^n\}.$$ 

One final point that needs to be emphasised is that functions that have a square integrable derivative need not be differentiable in the classical sense.\footnote{The connection between continuously differentiable functions and square integrable functions is given by Sobolev’s **embedding** theorem.}

**2. A model PDE problem**

The finite element method is a classical procedure that can be used to generate accurate numerical solutions of boundary value problems.\footnote{A **boundary value problem** is a PDE (or a system of PDEs) together with appropriate boundary and initial conditions.} The simplest example of such a problem is associated with the Poisson equation posed over a two-dimensional domain $\Omega \subset \mathbb{R}^2$ or a three-dimensional domain $\Omega \subset \mathbb{R}^3$.

The classical problem we will study is:

given a smooth forcing function ($f \in C^0(\Omega)$) and smooth boundary data $g \in C^0(\partial \Omega)$, compute (or approximate) the solution function $u : \Omega \to \mathbb{R}$ satisfying

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

together with mixed boundary conditions:

$$\begin{align*}
(2.1a) & \quad u = g_D \quad \text{on} \; \partial \Omega_D \quad \text{(Dirichlet)}, \\
(2.1b) & \quad \frac{\partial u}{\partial n} = g_N \quad \text{on} \; \partial \Omega_N \quad \text{(Neumann)}.
\end{align*}$$
Note that we tacitly assume that the boundary can be broken into disjoint pieces so that \( \partial \Omega = \partial \Omega_D \cup \partial \Omega_N \) and \( \partial \Omega_D \cap \partial \Omega_N = \emptyset \).

(Classical solution)

A **classical solution** \( u \) satisfying (2.1) and (2.2) is a solution that is suitably smooth. In the pure Dirichlet case (\( \partial \Omega = \partial \Omega_D \)) we require that \( u \in C^2(\Omega) \) and that \( u \) is continuous up to the boundary (so that (2.2a) can be satisfied).

The starting point for finite element analysis is the concept of a **weak solution**. This is discussed next. The starting point is to introduce a suitable space of “test functions”, say \( X \), and to make the PDE residual orthogonal (in an \( L^2(\Omega) \) sense) to all functions in this space, that is, we enforce

\[
\int_{\Omega} \{\nabla^2 u + f\} \, v = 0 \quad \forall v \in X.
\]

The function \( u \) satisfying (2.3) is called a **strong solution**. A clever strategy is to reduce the differentiability requirements on \( u \) by transferring derivatives onto the test function \( v \) using integration by parts,

\[
\int_{\Omega} \nabla^2 u \, v = \int_{\Omega} (\nabla \cdot \nabla u) \, v = -\int_{\Omega} \nabla v \cdot \nabla u + \int_{\partial \Omega} v \nabla u \cdot \vec{n} \, ds
= -\int_{\Omega} \nabla u \cdot \nabla v + \int_{\partial \Omega} v \frac{\partial u}{\partial n} \, ds.
\]

Substituting this expression into (2.3) and integrating over the boundary sections seperately gives

\[
\int_{\Omega} \nabla u \cdot \nabla v = \int_{\Omega} f v + \int_{\partial \Omega} v \frac{\partial u}{\partial n} \, ds
\]

\[
\int_{\Omega} \nabla u \cdot \nabla v = \int_{\Omega} f v + \int_{\partial \Omega_D} v \frac{\partial u}{\partial n} \, ds + \int_{\partial \Omega_N} v g_N \, ds.
\]

We then simplify the formulation (remove the first boundary integral) by insisting that the test functions \( v \) be zero on the Dirchlet part of the boundary. This gives a **weak formulation** of the original boundary value problem

\[
\int_{\Omega} \nabla u \cdot \nabla v = \int_{\Omega} f v + \int_{\partial \Omega_N} g_N v \, ds,
\]

which is required to hold for all functions \( v \) in the test space \( X \). The specification of the weak problem is completed by identifying the largest space \( X \) for which the integrals over \( \Omega \) in (2.4) remain finite. Using the Cauchy–Schwarz inequality we can show that this is precisely the space of functions that are square integrable and which have a square integrable first derivative. This immediately leads to the identification

\[
X = \{ v | v \in L^2(\Omega), \nabla v \in (L^2(\Omega))^n, v = 0 \text{ on } \partial \Omega_D \}.
\]
A model PDE problem

(W)\n
A weak solution \( u \) satisfies the essential boundary condition \((2.2a)\) and is the particular member of the solution space,

\[
X_E = \{ v | v \in L^2(\Omega), \nabla v \in (L^2(\Omega))^n; v = g_D \text{ on } \partial \Omega_D \},
\]

that satisfies the weak formulation \((2.4)\) for all functions in the associated test space \( X \). Note that the weak solution will satisfy the natural boundary condition \((2.2b)\) automatically.

What is the connection between classical and weak solutions? By construction, any function \( u \) that satisfies \((2.1)\) and \((2.2)\) must also be a weak solution. (Sobolev’s inequality ensures that \( u \in C^2(\Omega) \implies u \in H^1(\Omega) \) in one, two and three dimensions.) It can also be shown using functional analysis that if a weak solution is smooth enough then it is also a classical solution.

The uniqueness of the weak solution can be established by contradiction, making use of the Poincaré–Freidrichs inequality.\(^3\) This inequality bounds the norm of a test function by the norm of its (weak) derivative, that is

\[
\| u \|_{L^2(\Omega)} \leq L \| \nabla u \|_{L^2(\Omega)}, \quad u \in X,
\]

and involves the additional requirement that \( \int_{\partial\Omega_D} ds \neq 0 \). (Otherwise, we have \( \partial\Omega = \partial\Omega_N \), which is the so-called pure Neumann problem.)

One big advantage of working with the weak formulation \((2.4)\) is that it is well defined (all integrals are finite) for discontinuous forcing data \( f \) and rough boundary data \( g \). This can be seen by applying the Cauchy–Schwarz inequality to the integrals on the right-hand side of \((2.4)\)

\[
\int_{\Omega} fv + \int_{\partial\Omega_N} g_N v \, ds \leq \| f \|_{L^2(\Omega)} \| v \|_{L^2(\Omega)} + \| g_N \|_{L^2(\partial\Omega_N)} \| v \|_{L^2(\partial\Omega_N)},
\]

where the boundedness of the second term follows from extending \( v \) by zero to the whole of the boundary and then using a trace inequality\(^4\) to relate it’s norm to that of a smooth extension that is defined inside the domain

\[
\| v \|_{L^2(\partial\Omega_N)} = \| v \|_{L^2(\partial\Omega)} \leq C \| v \|_{H^1(\Omega)}.
\]

The implication of \((2.7)\) is that the weak formulation \((2.4)\) is valid for (discontinuous) forcing data \( f \in L^2(\Omega) \) and for boundary data \( g_N \in L^2(\partial\Omega_N) \). This is much larger class of functions than the data specification that is required when solving the classical problem.

\(^3\)\) The Poincaré–Freidrichs inequality is formally stated in Lemma 1.2 of [ESW].

\(^4\)\) The trace inequality is formally stated in Lemma 1.5 of [ESW].
3. Galerkin approximation  The weak formulation is not tractable because the test space \( X \) is infinite dimensional. A good way of generating a computational approximation to the weak solution is to find the best approximation to \( u \) from a finite-dimensional subspace of the solution space \( X_E \). The resulting function \( u_h \) is called the Galerkin solution.

(Galerkin solution)

A Galerkin solution (denoted \( u_h \)) satisfies the essential boundary condition (2.2a) and is the particular member of a finite-dimensional approximation space, \( X_h \subset X_E \) that satisfies the weak formulation (2.4) for all functions in the associated approximation test space \( X_h \subset X \). Note that the Galerkin solution will not satisfy the natural boundary condition (2.2b) in general.

Stated formally, the Galerkin solution is the function \( u_h \in X_h \subset X_E \) that satisfies a finite-dimensional version of (2.4),

\[
\int_{\Omega} \nabla u_h \cdot \nabla v_h = \int_{\Omega} f v_h + \int_{\partial \Omega_N} g_N v_h \, ds, \quad \forall v_h \in X_h.
\]

To show that \( u_h \) is the best approximation of \( u \), we pick an arbitrary test function \( v_h = z_h \) in (2.4) and subtract (3.8) with \( v_h = z_h \) to give

\[
\int_{\Omega} \nabla (u - u_h) \cdot \nabla z_h = 0, \quad \forall z_h \in X_h.
\]

This result is usually referred to as Galerkin orthogonality: the error function \( e = u - u_h \) is \( H^1 \) orthogonal to the test subspace \( X_h \). We can exploit this orthogonality by considering a specific measure (norm) of the error, namely

\[
\| u - u_h \|_E = \| \nabla u - \nabla u_h \|_{L^2(\Omega)}.
\]

In particular,

\[
\| u - u_h \|_E^2 = \| \nabla (u - u_h) \|_{L^2(\Omega)}^2 = \int_{\Omega} \nabla (u - u_h) \cdot \nabla (u - u_h)
\]

\[
= \int_{\Omega} \nabla (u - u_h) \cdot \nabla (u - v_h + v_h - u_h), \quad \text{for any } v_h \in X_E
\]

\[
= \int_{\Omega} \nabla (u - u_h) \cdot \nabla (u - v_h) + \int_{\Omega} \nabla (u - u_h) \cdot \nabla (v_h - u_h)
\]

\[
= \int_{\Omega} \nabla (u - u_h) \cdot \nabla (u - v_h) \quad \text{(from (3.9), } v_h - u_h \in X_h)
\]

Applying Cauchy–Schwarz to the right-hand side then gives

\[
\| u - u_h \|_E^2 \leq \| \nabla (u - u_h) \|_{L^2(\Omega)} \| \nabla (u - v_h) \|_{L^2(\Omega)} \leq \| u - u_h \|_E \| u - v_h \|_E.
\]
Thus, if \( \| u - u_h \|_E \neq 0 \) the Galerkin solution \( u_h \in X_h^E \) satisfying (3.8) is the best approximation\(^5\) to the weak solution \( u \) when the error is measured the energy norm (3.10),

\[
\| u - u_h \|_E \leq \| u - v_h \|_E, \quad \forall v_h \in X_h^E.
\]

(3.11)

A key question needs to be addressed at this point: how easy is it to construct the solution space \( X_h^E \) and test space \( X_h^E \)? What make this difficult is the need to construct basis functions that satisfy the essential boundary condition. Piecewise polynomial approximations are the solution to this: they provide a mechanism for constructing a sequence of increasingly refined approximation spaces \( X_h^E \subset X_{h+1}^E \subset \ldots \subset X^E \) in a computational setting.

(Finite element solution)

A finite element approximation (also denoted \( u_h \)) to the weak solution \( u \) satisfying (2.4) is a Galerkin solution computed using approximation spaces \( X_h^E \subset H^1(\Omega), X^h \subset H^1(\Omega) \) that are spanned by \( C^0 \) piecewise polynomials of low degree (typically linear or quadratic). The finite element approximation is thus a best approximation.

Specific examples of two-dimensional finite element approximations will be presented after discussion of some linear algebra aspects. To enable this, let us consider the generic case of a domain \( \Omega \) partitioned into \( n_k \) elements (for example, triangles in \( \mathbb{R}^2 \), bricks in \( \mathbb{R}^3 \)) with a total of \( n \) interpolation points in the interior of \( \Omega \cup \partial \Omega_D \) together with an additional \( n_\partial \) interpolation points lying on the Dirichlet boundary \( \partial \Omega_D \). As an example, for the mesh illustrated, if the interpolation points are taken to be vertices then we have \( n = 3 \) (the marked vertices 1, 2 and 3 are interior) and \( n_\partial = 9 \) (the vertices 4, 5, \ldots, 12 are on \( \partial \Omega_D \)).

In this example the solution approximation space will be spanned by 12 linear \( \text{hat functions} \),

\[
X_h^E = \text{span} \{ \phi_j(x,y) \}_{j=1}^{n+n_\partial},
\]

satisfying the interpolation conditions

\[
\phi_j = \begin{cases} 
1 & \text{at vertex } j, \\
0 & \text{at vertex } i \neq j. 
\end{cases}
\]

Note that we also have a best approximation (3.11) in the case \( \| u - u_h \|_E = 0 \).
and the Galerkin solution is constructed so that

\begin{equation}
\begin{split}
    u_h(x, y) &= \sum_{j=1}^{n} u_j \phi_j(x, y) + \sum_{j=n+1}^{n+n_D} u_j \bar{\phi}_j(x, y) \\
    &= 0 \quad \text{on} \ \partial \Omega_D \\
    &= g_D \quad \text{on} \ \partial \Omega_D
\end{split}
\end{equation}

In this representation, \( u_1, u_2, \ldots, u_n \) are unknown coefficients and \( \bar{\phi}_j \) are fixed values (they represent the value of \( g_D(x, y) \) at the \( j \)th boundary vertex). A key point is that hat functions have a square integrable derivative (they are not differentiable in a classical sense) so they are admissible functions in \( H^1(\Omega) \).

Note that the function \( g_D \) will not be represented exactly if it is not itself a piecewise linear function. The essential boundary condition will however be approximated increasingly accurately if a refined triangular mesh is used.

An important observation is that the interior functions \( \phi_1, \phi_2, \ldots, \phi_n \) form a basis for the approximation test space \( X^h \) (all these basis functions are 0 on \( \partial \Omega_D \)). This means that the Galerkin formulation (3.8) can be written as a system of \( n \) equations

\begin{equation}
\int_{\Omega} \nabla u_h \cdot \nabla \phi_i = \int_{\Omega} f \phi_i + \int_{\partial \Omega_N} g_N \phi_i \, ds, \quad i = 1, 2, \ldots, n.
\end{equation}

Substituting (3.12) into (3.13) gives an \( n \times n \) linear algebra system

\begin{equation}
Au = f,
\end{equation}

where \( u \) is the vector of all the unknown coefficients. We will refer to (3.14) as the Galerkin system and \( A \) as the Galerkin matrix, \( A_{i,j} = \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i \).

The right-hand side \( f \) is a vector of known values

\[ f_i = \int_{\Omega} f \phi_i + \int_{\partial \Omega_N} g_N \phi_i \, ds - \sum_{j=n+1}^{n+n_D} u_j \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i, \]

that characterise the problem input data; that is, \( f, g_D \) and \( g_N \).

The Galerkin matrix is clearly symmetric: \( A_{i,j} = A_{j,i} \). It can also be shown to be positive definite \( u^T Au > 0 \) for all \( u \neq 0 \), which means that it is nonsingular. This means that the Galerkin coefficient vector \( u \) is unique, which in turn implies that the Galerkin solution \( u_h \) given by (3.12) is always uniquely defined.

4. A worked example The best way to get a handle on the finite element solution process is to work through a specific problem on paper. The figure shows a finite element solution of Poisson’s equation computed on an L-shaped
domain with a constant forcing function \((f = 1)\) with a zero Dirichlet condition \(u = 0\) imposed everywhere on \(\partial\Omega\). What makes the problem challenging is the reentrant corner. The contours of solution height are increasingly closely packed near the origin. Converting to polar coordinates we find that the radial derivative of the solution \(u\) becomes unbounded in the limit \(r \to 0\).

To link with the discussion in the previous section we will exploit the inherent symmetry of the solution and solve the PDE over the bottom half of the domain with a zero Neumann condition \(\frac{\partial u}{\partial n} = 0\) on the line of symmetry \((y = x)\). If we want to compute the solution on the triangular mesh introduced in the previous section then we simply need to construct the Galerkin system (3.14) corresponding to the basis functions at the three interior vertices. Note that since we have \(g_N = 0\) and \(g_D = 0\) the components of the vector \(f\) have a relatively simple form: \(f_i = \int_{\Omega} \phi_i\) for \(i = 1, 2, 3\).

In general, the finite element solution process has three component parts; element integrations, element assembly and Galerkin system solution. First, element integrations are needed to compute the local contributions to the global Galerkin system. The idea is to construct the \(3 \times 3\) element contribution to the Galerkin matrix and a \(3 \times 1\) element vector to the right-hand side vector. If we focus on the specific triangle \(\vartriangle\) that is highlighted in the figure, then the three localised linear basis functions are given by the triangle area coordinates

\[
L_1 := \phi_7(x, y)_{\vartriangle}, \quad L_2 := \phi_1(x, y)_{\vartriangle}, \quad L_3 := \phi_5(x, y)_{\vartriangle}.
\]
The element contributions are thus given by

\[ A_{i,j}^{\mathcal{D}} = \int_{\mathcal{D}} \nabla L_j \cdot \nabla L_i = \frac{1}{4|\Delta|} \{ b_i b_j + c_i c_j \}, \quad i = 1, 2, 3, \quad j = 1, 2, 3; \]

\[ f_i^{\mathcal{D}} = \int_{\mathcal{D}} L_i = \frac{|\Delta|}{3}, \quad i = 1, 2, 3, \]

where \(|\Delta| = 0.125\) is the area of triangle \(\mathcal{D}\) and the \(b_i\)'s and \(c_i\)'s are the vertex distances of the local coordinates (shown in the shaded region of the figure) given by

\[ b_1 = y_2 - y_3 = 0, \quad b_2 = y_3 - y_1 = 0.5, \quad b_3 = y_1 - y_2 = -0.5, \]

\[ c_1 = x_3 - x_2 = -0.5, \quad c_2 = x_1 - x_3 = 0.5, \quad c_3 = x_2 - x_1 = 0. \]

Working out the explicit matrix entries gives

\[ A^{\mathcal{D}} = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} & 0 \\ -\frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad f^{\mathcal{D}} = \begin{pmatrix} \frac{1}{24} \\ \frac{1}{24} \\ \frac{1}{24} \end{pmatrix}. \]

A feature of our triangular subdivision is that all \(n_k\) elements are congruent. As a consequence, if a consistent numbering convention is adopted then all the element matrices and vectors have exactly the same form: \(A^{\mathcal{D}} = A^{\mathcal{D}} = A^{\mathcal{D}} = \ldots\) and \(f^{\mathcal{D}} = f^{\mathcal{D}} = f^{\mathcal{D}} = \ldots\).

To illustrate the element assembly process we consider the Galerkin matrix entry

\[ A_{1,2} = \int_{\Omega} \nabla \phi_2 \cdot \nabla \phi_1 = \sum_{k=1}^{12} \int_{\mathcal{D}_k} \nabla \phi_2 \cdot \nabla \phi_1. \]

We can clearly exclude all triangles from this sum which do not contain both of the two vertices (otherwise either \(\phi_1 = 0\) or \(\phi_2 = 0\)). (The same argument implies that the entry \(A_{1,3}\) is zero. If two vertices are not connected then the associated Galerkin matrix entry is zero.) This leaves the two triangles \(\mathcal{D}_6\) (lighter shading) and \(\mathcal{D}_7\) (darker shading) shown in the figure. With the local numbering illustrated we find that

\[ A_{1,2} = \int_{\mathcal{D}_6} \nabla L_2 \cdot \nabla L_1 + \int_{\mathcal{D}_7} \nabla L_1 \cdot \nabla L_2 = A^{\mathcal{D}_6} + A^{\mathcal{D}_7} = \begin{pmatrix} -1/2 \\ 1/2 \\ 1/2 \end{pmatrix} + \begin{pmatrix} -1/2 \\ 1/2 \\ 1/2 \end{pmatrix} = -1. \]
Generating the other entries in the same way gives rise to the *fully assembled* system for the three interior (red) nodes,

\[
\begin{pmatrix}
4 & -1 & 0 \\
-1 & 4 & -1 \\
0 & -1 & 2
\end{pmatrix}
\begin{pmatrix}
u_1 \\
u_2 \\
u_3
\end{pmatrix} = \begin{pmatrix}6/24 \\
5/24 \\
4/24\end{pmatrix}.
\]

This can be readily solved to give the Galerkin solution

\[u_h(x, y) = 0.08974\phi_1(x, y) + 0.10897\phi_2(x, y) + 0.13782\phi_3(x, y).\]

Recall that, by construction, the computed values \(u_1 = 0.08974, u_2 = 0.10897\) and \(u_3 = 0.13782\) represent the values of the solution at the three interior vertices.

There are two ways of increasing the accuracy of this solution. The simplest way would be to generate a finer triangulation (\(h\)-refinement), for example by subdividing every element into four smaller ones by connecting the mid-edge points. An alternative approach would be to increase the degree of the polynomial approximation within each triangle (\(p\)-refinement). To give an illustration, the picture shows what happens if the interpolation points (or nodes) are taken to be vertices and mid-edge points. Note that while we still have \(n_k = 12\) elements, we now have \(n = 18\) (interior nodes marked with a circle) and \(n_\partial = 17\) (nodes on \(\partial \Omega_D\) marked with a cross). Thus, in this case, the finite element solution space will be spanned by 35 piecewise quadratic hat functions,

\[X_h^E = \text{span} \{\psi_j(x, y)\}_{j=1}^{n+n_\partial},\]

satisfying the interpolation conditions

\[\psi_j = \begin{cases} 1 & \text{at node } j, \\ 0 & \text{at node } i \neq j. \end{cases}\]

If we focus on the specific triangle \(\Box\) that is highlighted in the figure then the six localised quadratic basis functions are given by

\[N_1 = 2L_1^2 - L_1, \quad N_2 = 2L_2^2 - L_2, \quad N_3 = 2L_3^2 - L_3, \]
\[N_4 = 4L_2L_3, \quad N_5 = 4L_1L_3, \quad N_6 = 4L_1L_2, \]
and the element contributions are given by
\[ A_{ij}^O = \int_{\Omega} \nabla N_j \cdot \nabla N_i, \quad i, j = 1, 2, \ldots, 6; \quad f_i^O = \int_{\Omega} N_i, \quad i = 1, 2, \ldots, 6. \]

Working out the matrix entries explicitly gives
\[ A^O = \begin{pmatrix} \frac{1}{2} & \frac{1}{6} & 0 & 0 & 0 & -\frac{2}{3} \\ \frac{1}{6} & 1 & \frac{1}{6} & -\frac{2}{3} & 0 & -\frac{2}{3} \\ 0 & \frac{1}{6} & 1 & -\frac{2}{3} & 0 & 0 \\ 0 & -\frac{2}{3} & -\frac{2}{3} & \frac{8}{3} & -\frac{4}{3} & 0 \\ 0 & 0 & -\frac{4}{3} & \frac{8}{3} & -\frac{4}{3} & \frac{1}{24} \\ -\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -\frac{4}{3} & \frac{8}{3} \end{pmatrix}, \quad f^O = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{24} \end{pmatrix}. \]

Assembling the 12 \( (n_k) \) element contributions gives an 18 × 18 Galerkin system
\[ \begin{pmatrix} 4 & \frac{1}{3} & 0 & \cdots & 0 \\ \frac{1}{3} & 4 & \frac{1}{3} & \cdots & 0 \\ 0 & \frac{1}{3} & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{8}{3} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{18} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \frac{1}{24} \end{pmatrix}, \]

which can be solved to give the refined Galerkin solution
\[ u_h(x, y) = 0.10207 \psi_1(x, y) + 0.13536 \psi_2(x, y) + \cdots + 0.05901 \psi_{18}(x, y). \]

To quantify the improvement in solution accuracy, we note that the value \( u_1 = 0.10207 \) is much closer to the exact value \( u(0.5, -0.5) = 0.10236 \ldots \) (computed on a very fine grid) than the original estimate \( u_1 = 0.08974 \).

5. Analysis of the convergence rate

From a numerical analysis perspective one would like to show that the approximation \( u_h \) is guaranteed to converge to the solution \( u \) when the grid is increasingly refined (that is, when the mesh parameter \( h \to 0 \)). Since point values of \( H^1(\Omega) \) functions are not defined when \( \Omega \subset \mathbb{R}^2 \) (or \( \Omega \subset \mathbb{R}^3 \)) we will need to introduce a refined Sobolev space if we are to make progress.

(higer-order Sobolev space)

The refined Sobolev space \( H^2(\Omega) \) contains all \( H^1(\Omega) \) functions that have square integrable second derivatives. If \( \Omega \subset \mathbb{R}^2 \) we have
\[ H^2(\Omega) = \{ v | v \in H^1(\Omega), \frac{\partial^2 v}{\partial x^2} \in L^2(\Omega), \frac{\partial^2 v}{\partial x \partial y} \in L^2(\Omega), \frac{\partial^2 v}{\partial y^2} \in L^2(\Omega) \}. \]

Ideally, one would also like to quantify the rate of convergence.
Analysis of the convergence rate

It will simplify the analysis if, at this point, we make an assumption that the weak solution \( u \) is smooth enough to ensure that \( u \in H^2(\Omega) \) (usually written as \( \| D^2 u \|_{L^2(\Omega)} < \infty \), where \( D^2 u \) is the square root of the sum of the squares of second weak derivatives of \( u \)). The desired smoothness is also referred to as \( H^2 \) regularity.

\( (H^2 \text{ regularity}) \)
The weak solution \( u \in X_E \) of (2.1) and (2.2) is said to be \( H^2 \) regular if there exists a constant \( C < \infty \) such that \( \| D^2 u \|_{L^2(\Omega)} \leq C \| f \|_{L^2(\Omega)} \) for every \( f \in L^2(\Omega) \).

We note in passing that the worked example discussed in the previous section will not be covered by the following analysis; the unbounded radial derivative at the corner means that \( u \not\in H^2(\Omega) \). In general, if we have \( H^2 \) regularity then the finite element approximation is guaranteed to converge linearly.

\( (\text{linear convergence}) \)
If the weak solution \( u \in X_E \) of (2.1)–(2.2) is \( H^2 \) regular then the linear finite element function \( u_h \) solving (3.8) using a triangular mesh \( T_h \) satisfies the error bound

\[
\| u - u_h \|_E \leq C_1 h \| f \|_{L^2(\Omega)},
\]

(5.15)

where \( C_1 \) is a constant\(^8\) and where the mesh parameter \( h \) is the longest triangle edge in \( T_h \). Note that (5.15) implies that \( u_h \to u \) in the limit \( h \to 0 \).

The first step in establishing (5.15) is to introduce the linear interpolation function \( \pi_h u \) defined on \( T_h \). This function agrees with the solution \( u \) at all vertices of the triangulation and is a member of the solution approximation space \( X^h_E \). Then, using the best approximation property (3.11) and breaking the integration into element contributions we see that

\[
\| u - u_h \|_E^2 \leq \| u - \pi_h u \|_E^2
= \| \nabla (u - \pi_h u) \|_{L^2(\Omega)}^2
= \sum_{\triangle k \in T_h} \| \nabla (u - \pi_h u) \|_{L^2(\triangle k)}^2.
\]

(5.16)

The element contribution to the interpolation error can be bounded using a scaling argument, which is discussed next. This argument will give the local interpolation error bound

\[
\| \nabla (u - \pi_h u) \|_{L^2(\triangle k)} \leq C \frac{h_k^3}{|\triangle k|} \| D^2 u \|_{L^2(\triangle k)},
\]

(5.17)

\(^8\)This constant only depends on the shape regularity of the triangulation.
where $C$ is a constant, $|\triangle_k|$ is the area of triangle $\triangle_k$ and $h_k$ is the length of the longest edge. Plugging (5.17) into (5.16) and then using the trigonometric bound $\frac{h_k^2}{\sin \theta_k} \leq |\triangle_k|$ where $\theta_k$ is the smallest angle, gives

\begin{equation}
\|u - u_h\|^2 \leq C \sum_{\Theta \in T_h} \frac{h_k^2}{|\triangle_k|} \left\| \nabla^2 u \right\|^2_{L^2(\triangle_k)} \leq C \sum_{\Theta \in T_h} \frac{h_k^2}{\sin^2 \theta_k} \left\| \nabla^2 u \right\|^2_{L^2(\triangle_k)}.
\end{equation}

This is the point where we need to precise the notion of \textit{shape regularity}.

\textbf{(minimum angle condition)}

A sequence of meshes $\{T_h\}$ is said to be \textit{shape regular} if there exists a minimum angle $\theta_\ast$ such that $\theta_k \geq \theta_\ast$ for all triangles $\triangle_k \in T_h$.

Note that since $\theta_\ast \leq \theta_k \leq \frac{\pi}{3}$ we have $\sin \theta_k \geq \sin \theta_\ast$ so that $1/\sin^2 \theta_k \leq C_\ast$.

Finally since $h_k \leq h = \max h_k$ we can further simplify the right-hand side to give

\begin{equation}
\|u - u_h\|^2 \leq C C_\ast h^2 \sum_{\Theta \in T_h} \left\| \nabla^2 u \right\|^2_{L^2(\triangle_k)} \leq \overline{C} C_\ast h^2 \left\| \nabla^2 u \right\|^2_{L^2(\Omega)},
\end{equation}

which, when combined with $H^2$ regularity, immediately leads to the linear convergence bound (5.15).

The interpolation error bound (5.17) is the key result. It is established by mapping the error from the physical element $\triangle_k$ to a reference element (defined in $\xi, \eta$ coordinate space), bounding the error in terms of norms of higher derivatives, and then mapping the higher derivatives back to the physical element.

For straight sided triangles the mapping is defined for all points $(x, y) \in \triangle_k$ and is given by

\begin{align}
(5.19a) \quad x(\xi, \eta) &= x_1 \chi_1(\xi, \eta) + x_2 \chi_2(\xi, \eta) + x_3 \chi_3(\xi, \eta) \\
(5.19b) \quad y(\xi, \eta) &= y_1 \chi_1(\xi, \eta) + y_2 \chi_2(\xi, \eta) + y_3 \chi_3(\xi, \eta),
\end{align}

where

\begin{align*}
\chi_1(\xi, \eta) &= 1 - \xi - \eta, \quad \chi_2(\xi, \eta) = \xi, \quad \chi_3(\xi, \eta) = \eta,
\end{align*}

are simply the area basis functions defined on the reference element. We note in passing that elements with curved sides can be generated using the...
analogous mapping defined using the six quadratic basis functions introduced in the previous section. The map from the reference element \( \Delta_* \) onto \( \Delta_k \) is differentiable. Thus, given a differentiable function \( \varphi(\xi, \eta) \), we can transform derivatives via

\[
\begin{bmatrix}
\frac{\partial \varphi}{\partial \xi} \\
\frac{\partial \varphi}{\partial \eta}
\end{bmatrix}
= \begin{bmatrix}
\frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\
\frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial \varphi}{\partial x} \\
\frac{\partial \varphi}{\partial y}
\end{bmatrix}.
\]

The Jacobian matrix in (5.20) can be calculated by differentiating (5.19)

\[
J_k = \frac{\partial (x, y)}{\partial (\xi, \eta)} = \begin{bmatrix}
x_2 - x_1 & y_2 - y_1 \\
x_3 - x_1 & y_3 - y_1
\end{bmatrix} = \begin{bmatrix}
c_3 - b_3 \\
-c_2 & b_2
\end{bmatrix}.
\]

Thus we see that \( J_k \) is a constant matrix over the reference element, and that the determinant

\[
|J_k| = \begin{vmatrix}
x_2 - x_1 & y_2 - y_1 \\
x_3 - x_1 & y_3 - y_1
\end{vmatrix} = \begin{vmatrix}
x_1 \\
x_2 \\
x_3
\end{vmatrix}
= 2|\Delta_k|
\]

is simply the ratio of the area of the mapped element \( \Delta_k \) to that of the reference element \( \Delta_* \). The fact that \( |J_k(\xi, \eta)| \neq 0 \) for all points \( (\xi, \eta) \in \Delta_* \) ensures that the inverse mapping from \( \Delta_k \) onto the reference element is uniquely defined and is differentiable. This means that the derivative transformation (5.20) can be inverted to give

\[
\begin{bmatrix}
\frac{\partial \varphi}{\partial x} \\
\frac{\partial \varphi}{\partial y}
\end{bmatrix}
= \frac{1}{2|\Delta_k|} \begin{bmatrix}
b_2 & b_3 \\
c_2 & c_3
\end{bmatrix}
\begin{bmatrix}
\frac{\partial \varphi}{\partial \xi} \\
\frac{\partial \varphi}{\partial \eta}
\end{bmatrix}.
\]

Returning to (5.17), defining \( e_k = (u - \pi_h u)|_{\Delta_k} \) and letting \( \tau_k \) represent the mapped error on the reference triangle, we have

\[
\|\nabla e_k\|^2_{L^2(\Delta_k)} = \int_{\Delta_k} \left( \frac{\partial e_k}{\partial x} \right)^2 + \left( \frac{\partial e_k}{\partial y} \right)^2 \, dx \, dy
\]

\[
= \int_{\Delta_*} \left( \frac{\partial e_k}{\partial \xi} \right)^2 + \left( \frac{\partial e_k}{\partial \eta} \right)^2 \, 2|\Delta_k| \, d\xi \, d\eta,
\]

where the derivatives satisfy (5.22); in particular the first term is of the form

\[
\left( \frac{\partial e_k}{\partial x} \right)^2 = \frac{1}{4|\Delta_k|^2} \left( b_2 \frac{\partial e_k}{\partial \xi} + b_3 \frac{\partial e_k}{\partial \eta} \right)^2
\]
with $b_2$ and $b_3$ defined by (5.21). Using the facts that $(a + b)^2 \leq 2(a^2 + b^2)$ and $|h_i| \leq h_k$, then gives the bound

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

The second term in (5.23) can be bounded in exactly the same way ($|c_i| \leq h_k$).

Summing the two terms gives

$$\tag{5.24} \|\nabla(u - \pi_h u)\|_{L^2(\Delta_k)}^2 \leq 2 \frac{h_k^2}{|\Delta_k|} \|\nabla(\bar{u} - \pi_h \bar{u})\|_{L^2(\Delta^*)}^2.$$

We will need to use a technical argument known as the Bramble–Hilbert lemma at this point. 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.\(^9\)

**(Bramble–Hilbert lemma)**

If $\pi_h \bar{u}$ is the linear interpolant of a sufficiently smooth function $\bar{u}$ defined on a reference triangle, we have

$$\tag{5.25} \|\nabla(\bar{u} - \pi_h \bar{u})\|_{L^2(\Delta^*)} \leq C \|D^2 \bar{u} - \pi_h \bar{u}\|_{L^2(\Delta^*)} = C \|D^2 \bar{u}\|_{L^2(\Delta^*)}.$$

If $\pi_{\bar{u}} \bar{u}$ represents the quadratic interpolant of a sufficiently smooth function $\bar{u}$ defined on a reference triangle, then we have a refined estimate

$$\tag{5.26} \|\nabla(\bar{u} - \pi_{\bar{u}} \bar{u})\|_{L^2(\Delta^*)} \leq C \|D^3 \bar{u} - \pi_{\bar{u}} \bar{u}\|_{L^2(\Delta^*)} = C \|D^3 \bar{u}\|_{L^2(\Delta^*)},$$

where $D^3 u$ is the sum of squares of the third derivatives of $u$.

Combining (5.25) with (5.24) gives

$$\tag{5.27} \|\nabla(u - \pi_h u)\|_{L^2(\Delta_k)}^2 \leq C \frac{h_k^2}{|\Delta_k|} \|D^2 \bar{u}\|_{L^2(\Delta^*)}^2.$$

The final step of the scaling argument is to map the second derivative terms appearing on the right-hand side of (5.27) back to the physical element. By definition,

$$\|D^2 \bar{u}\|_{L^2(\Delta^*)}^2 = \int_{\Delta^*} \left( \frac{\partial^2 \bar{u}}{\partial \xi^2} \right)^2 + \left( \frac{\partial^2 \bar{u}}{\partial \xi \partial \eta} \right)^2 + \left( \frac{\partial^2 \bar{u}}{\partial \eta^2} \right)^2 \, d\xi \, d\eta,$$

$$\tag{5.28} = \int_{\Delta_k} \left( \frac{\partial^2 u}{\partial \xi^2} \right)^2 + \left( \frac{\partial^2 u}{\partial \xi \partial \eta} \right)^2 + \left( \frac{\partial^2 u}{\partial \eta^2} \right)^2 \, \frac{1}{2|\Delta_k|} \, d\xi \, d\eta,$$

\(^9\)It is a multidimensional extension of the error estimates that were established in the approximation theory lecture notes using Rolle’s theorem. A precise statement is beyond the scope of these notes.
where the derivatives are mapped using (5.20); in particular the first term is of the form
\[ \left( \frac{\partial}{\partial \xi} \left( \frac{\partial u}{\partial \xi} \right) \right)^2 = \left( c_3^2 \frac{\partial^2 u}{\partial x^2} - 2c_3 b_3 \frac{\partial^2 u}{\partial x \partial y} + b_3^2 \frac{\partial^2 u}{\partial y^2} \right)^2 \]
\[ \leq 3 \left( c_3^2 \left( \frac{\partial^2 u}{\partial x^2} \right)^2 + 4c_3^2 b_3^2 \left( \frac{\partial^2 u}{\partial x \partial y} \right)^2 + b_3^4 \left( \frac{\partial^2 u}{\partial y^2} \right)^2 \right) \]
\[ \leq 12h_k^4 \left( \frac{\partial^2 u}{\partial x^2} \right)^2 + \left( \frac{\partial^2 u}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 u}{\partial y^2} \right)^2 . \]

The second and third terms in (5.28) can be bounded in exactly the same manner. Summing the three terms gives the result
\[ \|
\n\frac{D^2 \bar{u}}{L^2(\Delta)} \|^2 \leq 18h_k^2 \frac{h_k^2}{|\Delta_k|} \|
\n\frac{D^2 u}{L^2(\Delta_k)} \|^2 , \]
which when combined with (5.27) gives the interpolation error bound (5.17).

It is instructive to consider how the convergence rate would be changed if we were to use piecewise quadratic finite element approximation in place of piecewise linear approximation as done in the worked example. Mapping the interpolation error to the reference triangle gives the estimate (5.24). Next, assuming additional smoothness of the exact solution (third derivatives are square integrable), we can combine the refined estimate (5.26) with (5.24) to give the refined bound
\[ \|
\n\frac{\nabla (u - \pi^2 h u)}{L^2(\Delta_k)} \|^2 \leq C \frac{h_k^4}{|\Delta_k|} \|
\n\frac{D^3 \bar{u}}{L^2(\Delta)} \|^2 . \]

Thus, following the construction above and mapping the (four) third derivative terms on the right-hand side of (5.30) back to the physical element gives the estimate
\[ \|
\n\frac{D^2 \bar{u}}{L^2(\Delta)} \|^2 \leq 48h_k^1 \frac{h_k^2}{|\Delta_k|} \|
\n\frac{D^3 u}{L^2(\Delta_k)} \|^2 , \]
which when combined with (5.30) gives the improved interpolation error bound
\[ \|
\n\frac{\nabla (u - \pi^2 h u)}{L^2(\Delta_k)} \|^2 \leq C \frac{h_k^4}{|\Delta_k|} \|
\n\frac{D^3 u}{L^2(\Delta_k)} \|^2 . \]

The bottom line: if we have a $H^3$ regular problem then the quadratic finite element approximation is guaranteed to converge at a higher-order rate.
Estimation of the approximation error

If the weak solution \( u \in X_E \) of (2.1) and (2.2) is smooth enough then the quadratic finite element function \( u_h \) solving (3.8) using a triangular mesh \( T_h \) satisfies the error bound

\[
\| u - u_h \|_{L^2(\Omega)} \leq C_2 h^2,
\]

where \( h \) is the longest triangle edge in \( T_h \) and \( C_2 \) is a constant that depends on the shape regularity and on \( \| D^3 u \|_{L^2(\Omega)} \).

While the results in this section have been established in the context of two-dimensional approximation, the convergence estimates (5.15) and (5.33) also hold when a three-dimensional Poisson problem is solved on a tetrahedral tessellation of \( \Omega \) using piecewise linear and quadratic approximation respectively. In this case the interpolation points are the vertices and mid-edges as shown in the figure.

6. Estimation of the approximation error

A major reason for using the finite element method to solve the Poisson equation is that one can estimate the approximation error \textit{a posteriori} by postprocessing \( u_h \in X_E^h \) to compute an accurate estimate of the error in any given element, \( \eta_k \approx \| \nabla (u - u_h) \|_{L^2(\Delta_k)} \). This local error estimate can then be used to drive an adaptive refinement process.

An important requirement is that \( \eta_T \) must be cheap to compute—as a rule of thumb, the computational work should scale linearly as the number of elements is increased. For the singular example discussed earlier, adaptive refinement will lead to a succession of meshes like the one shown, that are selectively refined in the vicinity of the re-entrant corner so as to equidistribute the error and enhance overall cost effectiveness.

A key requirement is that the accuracy should be guaranteed in the sense that the estimated global error is an upper bound on the exact error,

\[
\| \nabla (u - u_h) \|_{L^2(\Omega)}^2 = \sum_{\Theta \in T_h} \| \nabla (u - u_h) \|_{L^2(\Delta_\Theta)}^2 \leq C(\theta_*) \sum_{\Theta \in T_h} \eta_\Theta^2,
\]

\[
(6.34)
\]
Estimation of the approximation error

with a constant $C$ that depends only on shape regularity. If, in addition to satisfying (6.34), the estimate $\eta_k$ gives a lower bound for the exact local error

$$\eta_k \leq C(\theta_{\omega_k}) \|\nabla (u - u_h)\|_{L^2(\omega_k)},$$

where $\omega_k$ represents the patch of elements adjoining $\Delta_k$, then the estimator $\eta_k$ is likely to be effective if it is used to drive an adaptive refinement process.

How does one compute $\eta_k$? Borrowing from linear algebra, we can compute the “backward error” by substituting the computed solution $u_h$ into the weak formulation to give a residual function $r$ that satisfies

$$\int_{\Omega} rv = \int_{\Omega} fv + \int_{\partial\Omega_N} g_N v \, ds - \int_{\Omega} \nabla u_h \cdot \nabla v, \quad \forall v \in X^h_*.$$                  (6.36)

where $X^h_*$ is a suitably chosen test space $X^h_* \subset X$ to be discussed later. We will refer to $X^h_*$ as the detail space. Note that, by the definition of the weak solution

$$0 = \int_{\Omega} fv + \int_{\partial\Omega_N} g_N v \, ds - \int_{\Omega} \nabla u \cdot \nabla v, \quad \forall v \in X^h_*,$$                  (6.37)

so if we subtract equations we get a simple relationship between the error function $e = u - u_h \in X$ and the residual function $r$,

$$\int_{\Omega} rv = \int_{\Omega} \nabla e \cdot \nabla v, \quad \forall v \in X^h_*.$$                  (6.38)

The error characterisation (6.38) underlies our a posteriori error estimation procedure. The selection of the detail space is key. First, if the error estimate is to be computable then $X^h_*$ needs to be finite dimensional. Second, if $X^h_* \subseteq X^h$ then $r = 0$ (from Galerkin orthogonality) so the detail space has to include functions that are not in the original approximation space $X^h$. Third, we would like to compute estimates of the error element-by-element. For this reason we consider a broken version of (6.36) and integrate the rightmost term by parts to give

$$\int_{\Omega} rv = \int_{\partial\Omega_N} g_N v \, ds + \int_{\Omega} fv - \sum_{\Theta \in T_h} \int_{\Delta_k} \nabla u_h \cdot \nabla v$$

$$= \int_{\partial\Omega_N} g_N v \, ds + \int_{\Omega} fv + \sum_{\Theta \in T_h} \int_{\Delta_k} \nabla^2 u_h v - \sum_{\Theta \in T_h} \int_{\partial\Delta_k} (\nabla u_h \cdot \vec{n}) v \, ds$$

$$= \int_{\partial\Omega_N} g_N v \, ds + \sum_{\Theta \in T_h} \int_{\Delta_k} \{f + \nabla^2 u_h\} v - \sum_{\Theta \in T_h} \int_{\partial\Delta_k} (\nabla u_h \cdot \vec{n}) v \, ds.$$                  (6.39)
There are actually three residuals embedded in (6.39). The first of these is the interior residual (or the PDE residual) $R_k = \{ f + \nabla^2 u_h \} |_{\Omega}$. Note that using linear finite element approximation $R_k$ simplifies to $f |_{\Omega}$.

The other two residuals are associated with the boundary terms in the error representation (6.39). If we introduce the set of triangle edges $E_k = \{ E_1, E_2, E_3 \}$ then the element flux contributions can be written as

$$
\sum_{\Omega \in T_h} \sum_{E \in E_k} \int_E (\nabla u_h \cdot \vec{n}_{E,k}) v \, ds.
$$

The triangle edges can be classified into three types. The first type are Dirichlet boundary edges, $E \in E_D$. On these edges we have $v = 0$ so the contribution to $\int_\Omega r v$ is zero. The second type are Neumann boundary edges $E \in E_N$.

For such edges, the outward flux can be combined with the $\partial \Omega_N$ term in (6.39) to give the boundary flux residual $R_N = \{ g_N - \nabla u_h \cdot \vec{n} \} |_{E \in E_N}$.

The third and final type of edge are interior edges $E \in E_h$. Every interior edge will incorporate two contributions to the residual error (one from the triangle $\Omega$ and the other from the adjoining triangle, $\Omega$ say). Combining the two contributions gives the flux-jump residual\(^{10}\) associated with edge $E$,

$$
R_E = \left[ \frac{\partial u_h}{\partial n} \right]_E = \nabla u_h \cdot \vec{n}_{E,k} + \nabla u_h \cdot \vec{n}_{E,n} = \{ \nabla u_h |_{\Omega} - \nabla u_h |_{\Omega} \} \cdot \vec{n}_{E,k}.
$$

Note that $R_E$ is a constant function on the edge $E$ in the case that the finite element solution $u_h$ is piecewise linear. Next, substituting the definitions $R_k$, $R_N$ and $R_E$ into the error representation gives

$$
\int_\Omega r v = \int_{\partial \Omega_N} R_N v \, ds + \sum_{\Omega \in T_h} \int_{\Delta_k} R_k v - \sum_{E \in E_h} \int_E R_E v \, ds.
$$

Finally, this can be written as an assembly of element contributions by equidistributing the flux jump to the two adjoining elements and then setting $R_E = 2 \{ g_N - \nabla u_h \cdot \vec{n} \}$ on Neumann boundary edges,

$$
\int_\Omega r v = \sum_{\Omega \in T_h} \left\{ \int_{\Delta_k} R_k v - \frac{1}{2} \sum_{E \in E_h} \int_E R_E v \, ds \right\}.
$$

\(^{10}\)The reason that this is a residual is that the classical solution $u$ is a differentiable function so there are no interior flux jumps.
Equating (6.38) with (6.41) suggests a mechanism for estimating the error element-by-element when one is given a suitable detail space $X^*_h$ defined on a given element $\bar{\Omega} \in T_h$.

**Local error estimator**

A local error estimator is the function $e_k \in X^*_h$ that satisfies the local Neumann problem

$$(6.42) \quad \int_{\Delta_k} \nabla e_k \cdot \nabla v_k = \int_{\Delta_k} R_k v_k - \frac{1}{2} \sum_{E \in E_k} \int_{E} R_E v_k \, ds,$$

for all test functions $v_k$ in the detail space $X^*_h$. Here, $R_k$ is the interior residual and $R_E$ is the flux jump residual associated with the finite element solution $u_h \in X_h^E$. The associated (energy) error estimate is

$$\eta_k = \|\nabla e_k\|_{L^2(\Delta_k)} \approx \|\nabla (u - u_h)\|_{L^2(\Delta_k)}.$$  

In a practical setting the detail space can be generated using either $h$-refinement (subdividing the triangle into four smaller ones) or $p$-refinement (adding quadratic interpolation functions at the mid-edge points). To give an illustration, we will reconsider the worked example discussed earlier. The error in the linear finite element solution can be estimated in the highlighted element by first computing the constant interior residual

$$R_k = \{f + \nabla^2 u_h\} |\bar{\Omega} = f |\bar{\Omega}$$

and then computing the flux jumps $R_E$ across edges $E_1$, $E_2$ and $E_3$ using (6.40). Next we set up a local problem (6.42) with a detail space

$$X^*_h = \text{span} \{\phi_{E_1}, \phi_{E_2}, \phi_{E_3}\},$$

consisting of the piecewise linear interpolation functions that are defined on the highlighted patch of four triangles (corresponding to a $h$-refinement). Note that the vertex interpolation functions are not included in the basis, so functions in the detail space are zero at the vertices. Computing the matrix and right-hand side entries explicitly gives the $3 \times 3$ system

$$\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 0.03365 \\ 0.04087 \\ -0.01843 \end{pmatrix},$$
Estimation of the approximation error

which can readily be solved to give the local error estimator

\[ e_\sigma(x, y) = 0.04107 \phi_{E_1}(x, y) + 0.04848 \phi_{E_2}(x, y) + 0.01502 \phi_{E_3}(x, y). \]

The associated element error estimate is also readily computed: \( \eta_k = 0.00309. \)

The error estimate is useful for two reasons. First, the global estimate

\[ \eta = \left( \sum_{k \in T_h} \eta_k^2 \right)^{1/2} = 0.16265... \]

gives a reliably accurate estimate of the overall energy error. Computational testing shows that the effectivity of the local error estimator strategy is always close to unity, typically

\[ 0.9 \leq \frac{\eta}{\| u - u_h \|_E} \leq 1.1. \]

Second, if we compare the element error estimates \( \eta_k \) with the total error \( \eta \), we can selectively refine the specific elements that contribute the most to the estimated error. Thus we don’t refine areas where the solution is flat so that little error reduction would be achieved. This is the process that was used to generate the highly nonisotropic subdivision illustrated at the beginning of this section. When an adaptive refinement procedure is applied to the problem solved in the worked example one is able to reduce the (estimated) error from 0.16265... to less than 0.005 in 25 refinement steps. Moreover, plotting the estimated error against the number of degrees of freedom we see that the rate of convergence is \( O(n^{-1/2}) \). This rate is optimal. It is that convergence rate that one would anticipate when using piecewise linear approximation to solve a \( H^2 \) regular problem in two dimensions.

All the computational results that are presented in these notes were computed using the freely downloadable TIFISS software package [http://www.manchester.ac.uk/ifiss/tifiss.html](http://www.manchester.ac.uk/ifiss/tifiss.html) © David Silvester, Alex Bespalov, Qifeng Liao and Leonardo Rocchi.

---

11 On a uniform grid, \( n \) varies like \( h^{-2} \) so \( O(n^{-1/2}) \) corresponds to linear convergence.