Chapter 2

The p–version of the Finite Element Method

Chapter Points

- The ends of words and sentences are marked by spaces. It does not matter how many spaces you type; one is as good as 100. The end of a line counts as a space.
- The ends of words and sentences are marked by spaces. It does not matter how many spaces you type; one is as good as 100. The end of a line counts as a space.

The previous chapter is a review on the h–version of the finite element method, where approximation spaces are built by modifying the mesh used for the analysis and fixing the polynomial interpolation degree (to say 1^{st} or 2^{nd} order). Traditionally, this has been the methodology adopted in the industry because having an accurate meshed representation of the geometry necessitates relatively small finite elements. Nevertheless, more accurate finite element spaces can be obtained by increasing the degree of the polynomial interpolants. This is the approach behind the p–version of the finite element method [1, 2], which will be briefly discussed here.

2.1 *P*–FEM IN 1-D

The main idea behind p–FEM is that given a fixed mesh, a more accurate approximate solution can be obtained by increasing the polynomial order p of the elements. The choice of polynomial basis plays an important role in the approximating properties of the method. Consider in Figure 2.1 two polynomial bases in the interval $\xi \in [-1, +1]$. For the basis ξ^p, $p = 1, \ldots, 8$ of Figure 2.1, the functions become increasingly similar with higher polynomial degrees – this also holds for only even or only odd orders when $\xi < 0$.) The inability of the basis to represent new features of the solution as the polynomial order is increased translates to stiffness matrices that become increasingly ill-conditioned. In theory, the stiffness matrix would only become singular when $p \to \infty$, but in finite precision arithmetic this condition is reached for even modest polynomial orders. In the p–FEM, therefore, a suitable basis function space needs to be chosen for the elements' shape functions. In practice, some set of *orthogonal polynomials* are chosen such that increasing the polynomial order does not produce ill-conditioned system matrices. One such basis, which is shown in

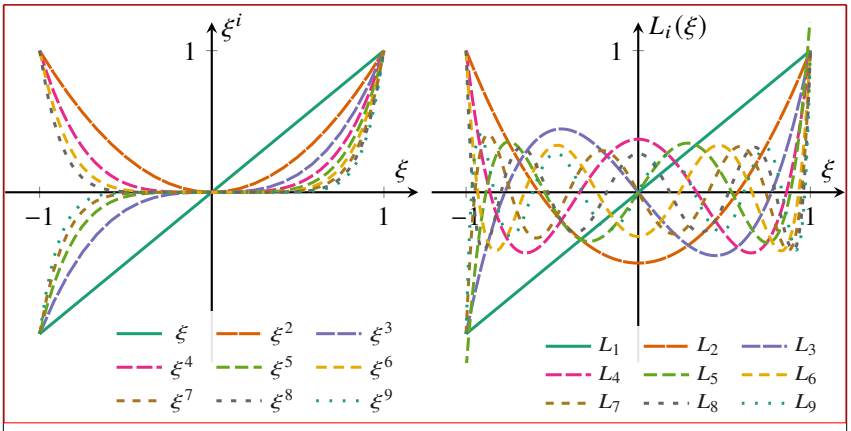

FIGURE 2.1 Polynomial bases on the interval $\xi \in [-1, +1]$ for degrees up to $p = 8$. (a) ξ^p; (b) Legendre polynomials $L_p(\xi)$.

Figure 2.1, is that defined by Legendre polynomials $L_p(\xi)$. The polynomials can be obtained using the following recursive formula:

$$L_p(\xi) = \frac{(2p-1)\,\xi L_{p-1}(\xi) + (1-p)\,L_{p-2}(\xi)}{p}, \tag{2.1}$$

with $L_0(\xi) = 1$ and $L_1(\xi) = \xi$.

The recursive nature of Eq. (2.1) is appropriate for a computer implementation since increasing polynomial orders can make use of already computed quantities.

The Legendre polynomials given in Eq. (2.1) could be used as basis functions for a finite element approximation. However, because building the stiffness matrix involves the derivatives of shape functions, it is more appealing to have Legendre polynomials as the derivatives instead to produce stiffness matrices that are better conditioned. To wit,

$$\varphi_1 = \frac{1-\xi}{2}, \quad \varphi_2 = \frac{1+\xi}{2}, \qquad \varphi_i = \hat{\varphi}_{i-1}, \qquad i = 3, 4, \ldots, p+1. \tag{2.2}$$

where

$$\hat{\varphi}_j(\xi) = \frac{1}{\sqrt{4j-2}}\left(L_j(\xi) - L_{j-2}(\xi)\right). \tag{2.3}$$

Algorithm 1 gives pseudo-code to compute the shape functions given by Eqs. (2.2) and 2.3 and their derivatives. The function HIERARCHICAL receives an input coordinate ξ and a polynomial order p, and uses the LEGENDRE function as a building block. The latter computes the Legendre polynomial for a given master coordinate and polynomial order recursively. Noteworthy, this algorithm works for arbitrary order. Figure 2.2 shows the hierarchical shape functions and their derivatives.

Algorithm 1 Hierarchical shape functions

function LEGENDRE(ξ, p)
  if $p = 0$ **then**
    return 1
  else if $p = 1$ **then**
    return ξ
  else
    return $\frac{(2p-1)\,\xi\,\text{LEGENDRE}(\xi,p-1)+(1-p)\,\text{LEGENDRE}(\xi,p-2)}{p}$
function HIERARCHICAL(ξ, p)
  $\left\{\varphi, \frac{\mathrm{d}\varphi}{\mathrm{d}\xi}\right\} \leftarrow \left\{\left[\frac{1-\xi}{2} \quad \frac{1+\xi}{2}\right], \left[-\frac{1}{2} \quad \frac{1}{2}\right]\right\}$
  if $p \geq 2$ **then**
    for $j \in \{2 \ldots p+1\}$ **do**
      $\varphi \leftarrow \left[\varphi \quad \frac{1}{\sqrt{4j-2}}\left(\text{LEGENDRE}(\xi, j) - \text{LEGENDRE}(\xi, j-2)\right)\right]$
      $\frac{\mathrm{d}\varphi}{\mathrm{d}\xi} \leftarrow \left[\frac{\mathrm{d}\varphi}{\mathrm{d}\xi} \quad \sqrt{\frac{2j-1}{2}}\,\text{LEGENDRE}(\xi, j-1)\right]$
  return $\left\{\varphi, \frac{\mathrm{d}\varphi}{\mathrm{d}\xi}\right\}$

With the shape functions defined by Eqs. (2.2) and (2.3), the finite element approximation in an element of the discretization $\Omega_i = \left[x_i, x_j\right]$ can be written

$$u^h = \sum_{i \in \mathcal{N}} \varphi_i U_i = \boldsymbol{\varphi}\mathbf{U}, \tag{2.4}$$

where $\mathcal{N}$ now denotes the set of all nodes (both standard and generalized). In other words, the set $\mathcal{N}$ comprises nodes associated with linear shape functions, and also generalized nodes associated with the nonlinear shape functions. For an approximation of order p, the sum in (2.4) extends over $p+1$ terms.

Similarly to the h–FEM case of § 1.2.3, the Galerkin discrete formulation is obtained after interpolating the weight function in the same way as the displacement, *i.e.*, $w^h = \boldsymbol{\Phi}\mathbf{W}$. Then the problem Find $\boldsymbol{u}^h \in \mathcal{V}^\star$ such that

$$B\left(\boldsymbol{u}^h, \boldsymbol{w}^h\right) = L\left(\boldsymbol{w}^h\right) \qquad \forall\, \boldsymbol{w}^h \in \mathcal{V}_0^h\left(\Omega\right), \tag{2.5}$$

leads to the discrete system $\mathbf{KU} = \mathbf{F}$, where $\mathbf{K} = \mathbb{A}_e\, \mathbf{k}_e$, $\mathbf{F} = \mathbb{A}_e\, \mathbf{f}_e$, and

$$\mathbf{k}_e = \int_{x_i}^{x_j} \left(EA\boldsymbol{b}^\intercal\boldsymbol{b} + k\boldsymbol{\varphi}^\intercal\boldsymbol{\varphi}\right)\, \mathrm{d}x = \sum_{i=1}^{p+1} \left(EA\boldsymbol{b}^\intercal\boldsymbol{b} + k\boldsymbol{\varphi}^\intercal\boldsymbol{\varphi}\right) w_i j, \tag{2.6}$$

$$\mathbf{f}_e = \int_{x_i}^{x_j} \boldsymbol{\varphi}^\intercal b\, \mathrm{d}x + \begin{bmatrix} -N_i \\ N_j \end{bmatrix} = \sum_{i=1}^{p+1} \left(\boldsymbol{\varphi}^\intercal b\right) w_i j + \begin{bmatrix} -N_i \\ N_j \end{bmatrix}. \tag{2.7}$$

In Eqs. (2.6) and (2.7) the integrals have been replaced by a discrete quadrature over Gauss points on the master element, whose number has to be chosen

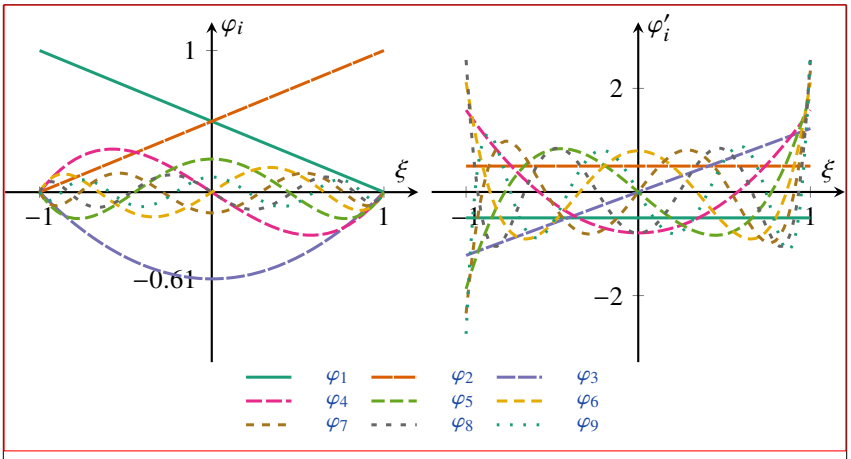

FIGURE 2.2 Hierarchical shape functions (a) and their derivatives (b) for $1 \le p \le 10$.

carefully to ensure exact integration. Noteworthy, the geometry in p–FEM is usually approximated only linearly, and thus for $p \ge 2$ a *sub-parametric* mapping is used (see § 1.1.5).

Example 8 (p**–FEM solution to constrained bar pullout problem**)**.** Extend the finite element program used to solve Example 9 in the previous chapter to accommodate for an arbitrary polynomial interpolation. With a fixed mesh size $h/L = 0.25$ over the domain $x \in [0, 10]$, compute approximate solutions for orders $p = \{1, 2, 4, 8\}$. Create a convergence plot where in abscissas you plot the number of DOFs, and in ordinates the relative error in the energy norm as defined in Eq. (1.51). Convert the results in Example 9 to reflect the number of DOFs (instead of the mesh size) and compare the results.

Solution:

The finite element discretization for this problem was shown earlier in Figure 1.10. The $p + a$ finite element matrix given by Eq. (2.6) for this problem is obtained as

$$\mathbf{k}_e = \sum_{i=1}^{p+1} \left(\boldsymbol{b}^\intercal \boldsymbol{b} + \alpha^2 \boldsymbol{\varphi}^\intercal \boldsymbol{\varphi} \right) w_i j. \tag{2.8}$$

In fact, if not for the Gauss rule, Eqs. (1.52) and (2.8) are identical. As before, the boundary conditions *i.e.*, $U_5 = -0.0000453999$ and $F_1 = -1$ are prescribed. This implies that the DOFs associated with $p \ge 2$ come after the linear ones.

The results for the bar pullout problem are summarized in Figure 2.3. Figure **??** compares the relative error in energy norm ($e_{\mathcal{E}} \equiv \left\| u - u^h \right\|_{\mathcal{E}(\Omega)} / \|u\|_{\mathcal{E}(\Omega)}$) as a function of the total number of degrees of freedom for both h–FEM and p–FEM. Incidentally, the mesh sizes $h = \{4, 8, 16, 32\}$ used in Example 9 result in the same number of total DOFs as the sequence $p = \{1, 2, 4, 8\}$ used here. The figure shows a remarkable gain in accuracy by using p–FEM. In fact, we can show the convergence of p–FEM for this problem is *exponential*, as the rate of convergence increases with the interpolant order p. As it will be explained later in this chapter, exponential convergence is

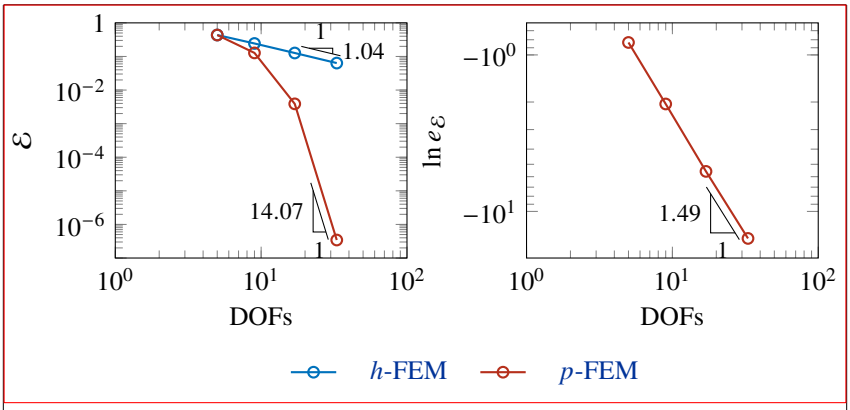

FIGURE 2.3 (a) Convergence comparison between h–FEM and p–FEM; (b) Logarithm of the error as a function of DOFs.

characterized mathematically by

$$\left\| u - u^h \right\|_{\mathcal{E}(\Omega)} \leq \frac{C}{e^{\gamma N^\theta}} \|u\|_{\mathcal{E}(\Omega)} , \tag{2.9}$$

where γ, θ are constants that depend on u and N is the number of DOFs. We can then recover the exponent θ in (2.9) by taking the logarithm of the relative error (see Figure 2.3). For this problem, whose exact solution is smooth, theory predicts a value $\theta \geq 1$ ($\theta = 1.49$ in the figure).

Example 9 (First-order shear deformation theory beam). Consider in Fig. 2.4 a cantilever beam of length L subjected to a uniform distributed load q. The beam has cross sectional area $A = BH$, moment of inertia I, and Young's and shear moduli E and G, respectively. First-order shear deformation (FSDT) theory, contrary to classical beam theory, accounts for shear deformation. While in classical beam theory the beam angle of rotation θ is related to the transverse displacement w as $\theta = \mathrm{d}w/\mathrm{d}x$, in FSDT the rotation angle is approximated independently from the transverse displacement. However, FSDT is not usually used to model thin beams since the formulation suffers from locking: As the thickness of the beam reduces, the formulation introduces inaccurate shear strain energy that results in spurious stiffness and thus in overstiff models. Show strong and weak forms of the problem. In addition, show that while linear approximations indeed exhibit locking, a simple hierarchical polynomial approximation can be used to remove it.

Solution:

The strong form of the problem can be written as: Given the distributed force $q : \Omega = [0, L] \rightarrow$

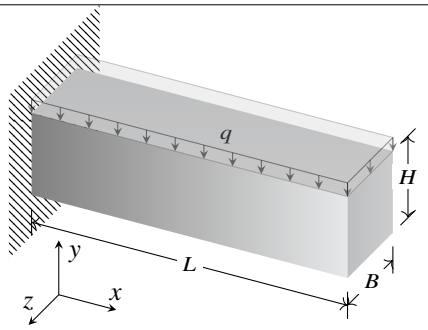

FIGURE 2.4 Cantilever beam of length L and cross sectional area $A = BH$. The beam is loaded with a uniform distributed load q.

$\mathbb{R}$, find the transverse displacement w and the rotation angle θ such that

$$\begin{aligned} \frac{\mathrm{d}^2}{\mathrm{d}x^2}\left(EI\frac{\mathrm{d}\theta}{\mathrm{d}x}\right) &= q, && \text{on } \Omega, \\ \frac{\mathrm{d}}{\mathrm{d}x}\left(EI\frac{\mathrm{d}\theta}{\mathrm{d}x}\right) + GA\left(\frac{\mathrm{d}w}{\mathrm{d}x} - \theta\right) &= 0, && \text{on } \Omega, \end{aligned} \tag{2.10}$$

with boundary conditions at $x = 0$, $w = 0$, $\theta = 0$, and at $x = L$, $Q(L) = 0$, and $M(L) = 0$; $M = -EI\kappa$ and $Q = -K_s GA\gamma$ are the bending moment and shear force, respectively, with $\kappa = \mathrm{d}\theta/\mathrm{d}x$ the curvature, $\gamma = -\theta + \mathrm{d}w/\mathrm{d}x$ the shear strain, and K_s a shear correction factor (5/6 for rectangular cross section). The exact solution to this problem is given by:

$$\begin{aligned} w(x) &= \frac{q\left(6L^2x^2 - 4Lx^3 + x^4\right)}{24EI} + \frac{q\left(2Lx - x^2\right)}{2GA}, \\ \theta(x) &= \frac{q\left(12L^2x - 12Lx^2 + 4x^3\right)}{24EI}. \end{aligned} \tag{2.11}$$

The weak form of equilibrium can be written as: Find $w, \theta \in \mathcal{W} \times \Theta$ such that

$$B\left(w, \theta; \hat{w}, \hat{\theta}\right) = L\left(\hat{w}, \hat{\theta}\right), \qquad \forall \hat{w}, \hat{\theta} \in \hat{\mathcal{W}} \times \hat{\Theta}, \tag{2.12}$$

where w, θ $(\hat{w}, \hat{\theta})$ are the trial (test) functions for the transverse displacement and the angle of ration, respectively, and $\mathcal{W}, \Theta$ $(\hat{\mathcal{W}}, \hat{\Theta})$ their corresponding spaces. By denoting the bending moment as $M = -EI\kappa$ and the shear force as $Q = -K_s GA\gamma$, where $\kappa = \mathrm{d}\theta/\mathrm{d}x$ is the curvature, $\gamma = -\theta + \mathrm{d}w/\mathrm{d}x$ the shear strain, and K_s a shear correction factor (5/6 for rectangular cross section), the bilinear and linear forms are given by

$$B\left(w, \theta; \hat{w}, \hat{\theta}\right) = \int_0^L \left(M\hat{\theta}' + Q\left(\hat{w}' - \hat{\theta}\right)\right) \mathrm{d}x \tag{2.13}$$

$$L\left(\hat{w}, \hat{\theta}\right) = \int_0^L q\hat{w}\, \mathrm{d}x\,. \tag{2.14}$$

Similarly to Eq. (1.17), the energy norm is $\|(w, \theta)\|_{\mathcal{E}(\Omega)} = \sqrt{\frac{1}{2}B(w, \theta; w, \theta)}$. Notice that by setting $\hat{w} = \delta w$, $\hat{\theta} = \delta\theta$, (2.12) can be written as

$$\int_0^L \left(M\delta\kappa + Q\delta\gamma\right) \mathrm{d}x - \int_0^L q\delta w\, \mathrm{d}x = 0, \tag{2.15}$$

which is the commonly seen expression for the principle of virtual work.

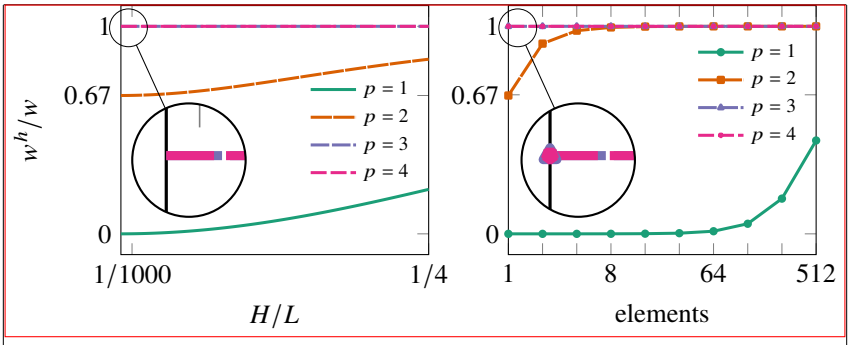

FIGURE 2.5 Ratio between the transverse displacement at the tip w^h, normalized by the exact value w given by Eq. 2.11, as a function of the number of elements (left) and beam height for a beam discretized by four elements.

For solving the discrete counterpart of (2.12), we consider steel as the material with $E = 200\,\text{GPa}$, Poisson ratio $\nu = 0.3$, and thus the ratio $E/G = 2.6$ (as in Ref. [25]). We take our trial functions for the deflection and rotation similarly to (2.4) as

$$w^h = \sum_{i \in \mathcal{N}} \varphi_i W_i, \qquad \theta^h = \sum_{i \in \mathcal{N}} \varphi_i \Theta_i, \tag{2.16}$$

and similarly for the test functions. We discretize the beam with a single element and solve the problem for polynomial orders $p = \{1, \ldots, 4\}$. The ratio w^h/w between the numerical deflection at the tip of the beam ($x = L$) and the exact value given by Eq. (2.11) is plotted in Fig. 2.6 as a function of the ratio H/L. As apparent from the figure, the linear formulation exhibits locking. Discretizing the problem with more elements does not improve the situation considerable. The figure also shows the results obtained for $L/H = 1000$ by discretizing the beam in $\{1, 2, 4, \ldots, 512\}$ finite elements, where for 512 elements the deflection is only about 45% of the exact value.

In Fig. 2.6 the tip deflection is indistinguishable between $p = 3$ and $p = 4$, even though the exact solution is a quartic polynomial. In Fig. 2.6 we plot the relative error in energy norm squared and the condition number of the stiffness matrix, as a function of the total number of DOFs. The figure reveals a few interesting points. The solution for $p = 4$ is exact for any number of elements, so the maximum accuracy we can obtain for this problem is about 10^{-7}. Notice we plot the square of the error since taking the square root would chop that accuracy by half due to floating point arithmetic (double precision with 16 digits of accuracy). In this figure we can see the difference between $p = 3$ and $p = 4$. Notice the curve labeled $p = \{1, \ldots, 4\}$, for which we used a single finite element. The figure also shows the condition number of the stiffness matrix as a function of DOFs. The condition number was computed as $\text{cond}(\boldsymbol{K}) := \lambda_{\max}/\lambda_{\min}$, where $\lambda_{\max}$ and $\lambda_{\min}$ are the maximum and minimum eigenvalues, respectively, after prescribing essential boundary conditions (thus suppressing zero eigenvalues due to rigid body bodes). The figure shows the problem becomes ill-conditioned for even a single element since $\text{cond}(\boldsymbol{K}) = O(10^8)$. For any fixed value of polynomial order, as problems get larger and larger, the condition number follows the well known estimate for standard h-FEM of $\text{cond}(\boldsymbol{K}) = O(n^2)$.

2.2 *P*–FEM IN 2-D

2.2.1 Basis functions for quadrilaterals

The orthogonal functions described in the previous section can also be used to construct higher-order shape functions in 2-D and 3-D. Consider a 4-node quad-

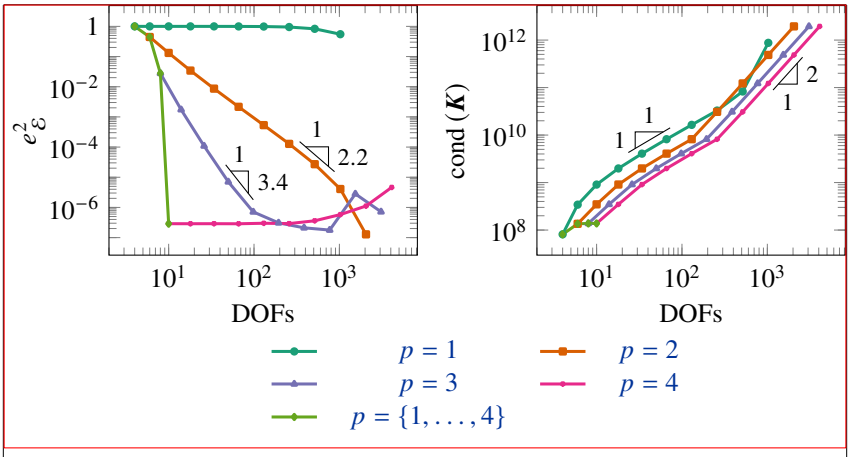

FIGURE 2.6 Relative error in energy norm squared (left) and condition number (right) as a function of the total number of DOFs. In addition to the results obtained for each polynomial interpolation and for $\{1, 2, 4, \ldots, 512\}$ number of elements, the curve $p = \{1, \ldots, 4\}$ shows the results for a single beam element.

rangle, described by a master element $\square = \left\{\boldsymbol{\xi} = (\xi, \eta) \in [-1, 1] \times [-1, 1] \subset \mathbb{R}^2\right\}$. Eqs. (2.2) can also be used as a basis for orthogonal polynomials along the η axis. One approach for constructing 2-D basis functions is obtained by $\phi_i(\xi)\,\phi_j(\eta)$, as illustrated in Figure 2.7. In the figure we can distinguish three types of functions:

1. **Nodal functions** These are obtained by the product of linear functions, *i.e.*, $\varphi_i(\xi)\,\varphi_j(\eta)$, $i, j = \{1, 2\}$, and are equivalent to the shape functions of the 4-node bilinear standard finite element.

Higher-order shape functions are obtained by multiplying functions of higher order.

2. **Edge functions** Taking a linear function in one axis and and a higher-order function in the orthogonal direction, *i.e.*, $\varphi_i\hat{\varphi}_j$, $i = \{1, 2\}$, $j = \{2, \ldots\}$, gives rise to edge functions. Edge functions, which are shown in the figure up to $p = 5$, are constructed as

$$\begin{aligned}
&\varphi_1(\eta)\,\hat{\varphi}_i(\xi), \; i = 2, \ldots && \text{first edge}\\
&\varphi_2(\xi)\,\hat{\varphi}_i(\eta), \; i = 2, \ldots && \text{second edge}\\
&\varphi_2(\eta)\,\hat{\varphi}_i(\xi), \; i = 2, \ldots && \text{third edge}\\
&\varphi_1(\xi)\,\hat{\varphi}_i(\eta), \; i = 2, \ldots && \text{fourth edge}
\end{aligned}$$

3. **Bubble functions** These functions are obtained as $\hat{\varphi}_i\hat{\varphi}_j$, $i, j = \{2, \ldots\}$.

A question one may ask is: How many basis functions do I need to solve a determined problem? This, of course, depends on the problem, as there may be cases that require a higher-order interpolation only along a preferred direction.

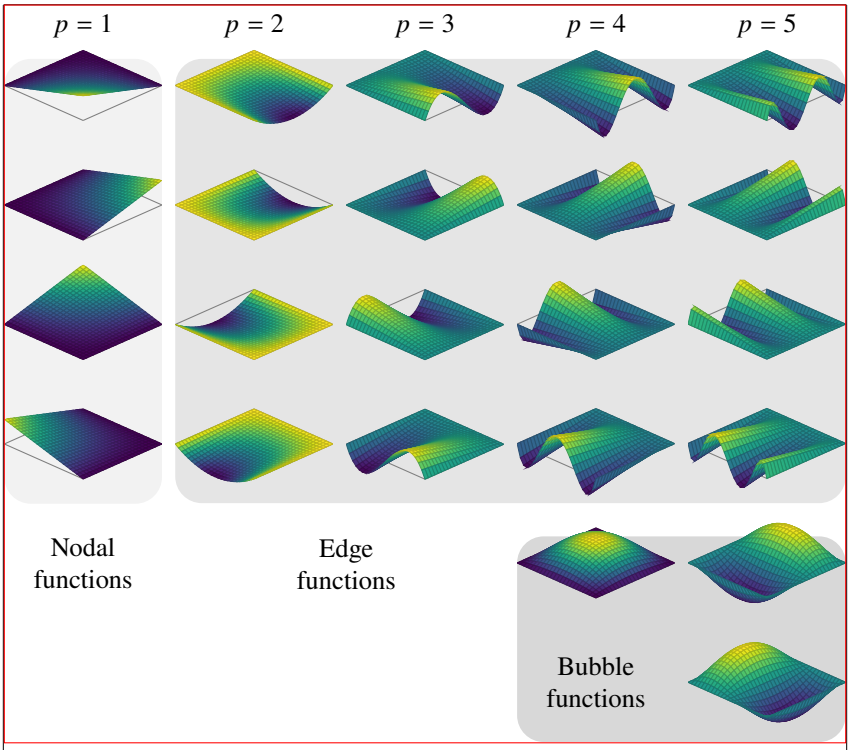

FIGURE 2.7 Shape functions for a 2-D quadrangular elements.

Nevertheless, intuitively we know that the more functions are considered, the richer the finite element space considered. We may also intuit that it may be a bad idea to leave some lower-order functions out of the approximation–particularly the linear ones!

When discussing polynomial function spaces for p–FEM there are many options available. Two spaces that have been thoroughly studied [2, 26] are the tensor product space $\mathcal{S}_{\text{ps}}^{p_\xi, p_\eta}$ and the trunk space $\mathcal{S}_{\text{ts}}^{p_\xi, p_\eta}$, where p_ξ and p_η denote, respectively, the polynomial order along the ξ and η directions. These spaces can be easily visualized in the Pascal triangle shown in Figure 2.8. Monomials are shown for simplicity since these span the same space as those more complex functions given by Eq. (2.2).

2.2.2 Basis functions for triangles

As in the previous section, shape functions for triangles are constructed with the aid of orthogonal polynomials. We use Chebyshev polynomials as an alternative to the Legendre polynomials used previously [27]. These are given by the

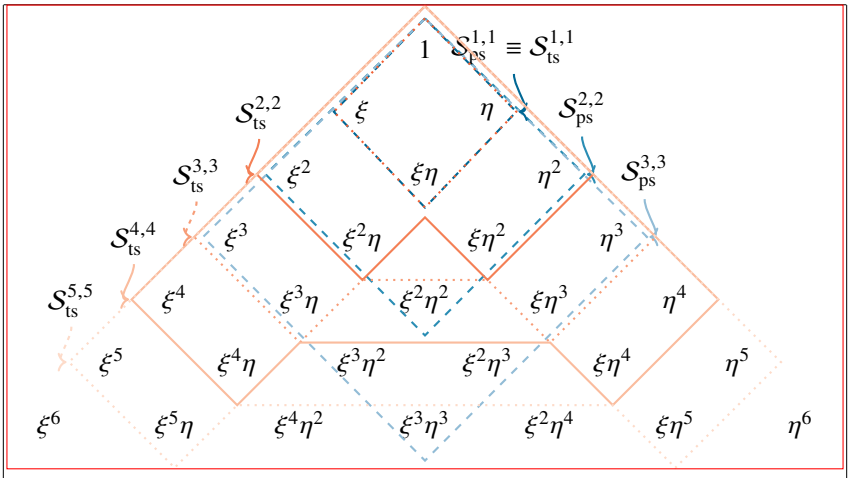

FIGURE 2.8 Pascal triangle that shows the trunk and tensor product polynomial spaces used with 4-node quadrangular elements.

recursive formula:

$$C_p(\xi) = 2\xi C_{p-1}(\xi) - C_{p-2}(\xi), \tag{2.17}$$

with $C_0(\xi) = 1$ and $C_1(\xi) = \xi$. This polynomial basis is shown in Figure 2.9, and contrary to the Legendre polynomials presented earlier in Figure **??**, Chebyshev polynomials attain a maximum value max $(|C_i(\xi)|) = 1$ $\xi \in [-1, 1]$.

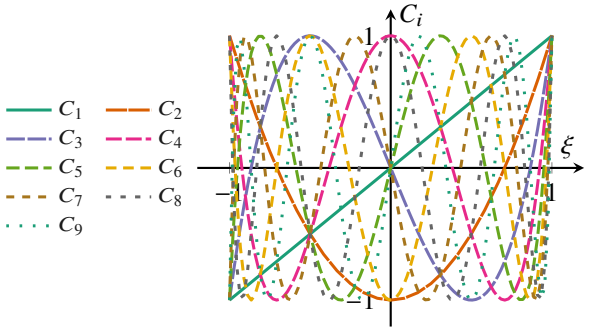

FIGURE 2.9 Chebyshev polynomials $C_p(\xi)$.

Similarly to the quadrangular case, we define three types of shape functions on a master element $\triangle = \{\boldsymbol{\xi} = (\xi, \eta) \in \mathbb{R}^2 : \xi \geq 0 \wedge \eta \geq 0 \wedge \xi + \eta \leq 1\}$:

1. Nodal functions These are the same as those for the constant-strain triangle:

$$\varphi_1 = 1 - \xi - \eta, \quad \varphi_2 = \xi, \quad \varphi_3 = \eta.$$

Higher-order shape functions are obtained by multiplying functions of higher order.

2. **Edge functions** Edge functions are constructed by using the nodal functions to ensure high-order functions vanish at two edges of the triangle:

$$\begin{array}{ll} \varphi_1\varphi_2\, C_i\,(2\xi+\eta-1) & \text{first edge,} \\ \varphi_2\varphi_3\, C_i\,(\eta-\xi) & \text{second edge,} \\ \varphi_1\varphi_3\, C_i\,(1-\xi-2\eta) & \text{third edge,} \end{array}$$

   with $i = 0, 1, \ldots, p-2$.

3. **Bubble functions** The bubble functions for triangles take the form

$$\varphi_1\varphi_2\varphi_3\, C_i\,(2\xi-1)\; C_j\,(2\eta-1)\,,$$

   with $0 \le i+j \le p-3$. As for edge functions, nodal functions are used to ensure high-order bubble functions are exactly zero at all edges.

The shape functions given above are shown in Figure 2.10.

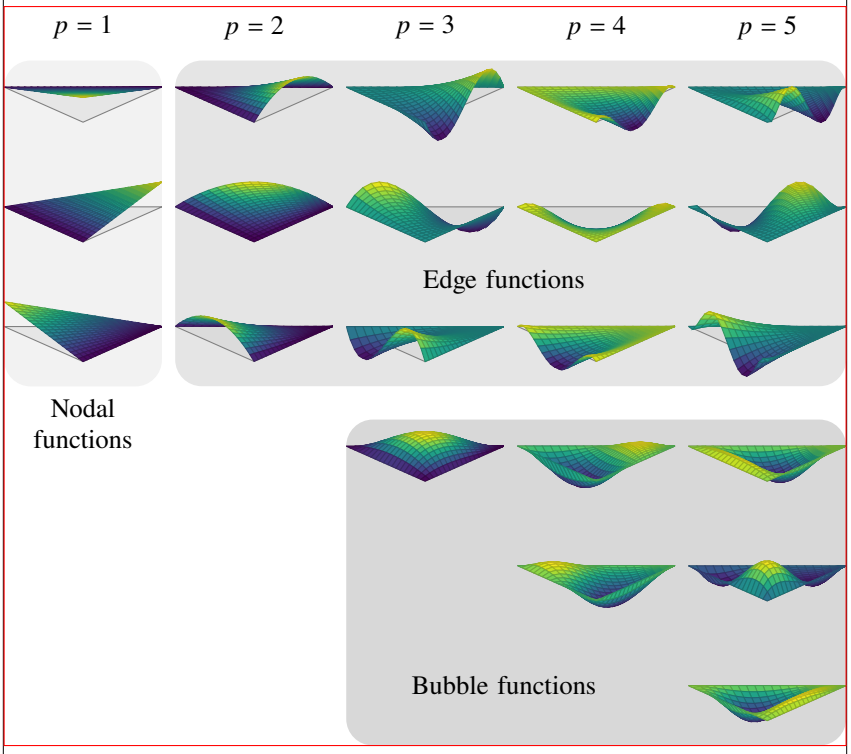

FIGURE 2.10 Shape functions for 2-D triangular elements.

2.2.2.0.1 *A priori* error estimates for p–FEM

In the previous chapter we looked at error estimates for h–FEM. We discussed that, depending on the regularity of the exact solution, problems can be classified in one of three categories. For non-smooth problems (*e.g.*, problems with singularities), p–FEM exhibits algebraic convergence:

$$\left\|\boldsymbol{u}-\boldsymbol{u}^h\right\|_{\mathcal{E}(\Omega)} \leq C_2 \frac{1}{p^{\beta_p}} \|\boldsymbol{u}\|_{\mathcal{E}(\Omega)}\,, \tag{2.18}$$

where p is the minimum interpolation order in the mesh. If the problem is smooth enough, however, p–FEM can result in *exponential convergence*, as demonstrated earlier in Exercises 8 and 2.1. This behavior, that is attractive because the error reduces more rapidly with increasing the interpolant order, shows the error as

$$\left\|\boldsymbol{u}-\boldsymbol{u}^h\right\|_{\mathcal{E}(\Omega)} \leq \frac{C}{e^{\gamma N^\theta}} \|\boldsymbol{u}\|_{\mathcal{E}(\Omega)}\,, \tag{2.19}$$

where N is the number of DOFs, and γ, θ are constants. Defining the relative error as $e_{\mathcal{E}} \equiv \left\|\boldsymbol{u}-\boldsymbol{u}^h\right\|_{\mathcal{E}(\Omega)} / \|\boldsymbol{u}\|_{\mathcal{E}(\Omega)}$, and applying the natural logarithm twice to (2.19):

$$\ln e_{\mathcal{E}} \leq \ln C - \gamma N^\theta,$$
$$\ln\left(\ln e_{\mathcal{E}}\right) \leq -\ln\gamma - \theta \ln N,$$

where the last inequality, which holds for $C \geq 1$, represents a line with slope $-\theta$ in $\ln N \times \ln\left(\ln e_{\mathcal{E}}\right)$ space.

The p–FEM convergence rates for problems in each of the three categories are summarized in Table 2.1.

	Category		
	A	B	C
Convergence type	exponential	algebraic	algebraic
Convergence rate	$\theta \geq 1/d$ †	$\beta_p = 2\,(k-1)$ ‡ for uniform h	$\beta_p > 0$

† $d \equiv$ dimension of the problem
‡ $k \equiv$ regularity of the solution. For $\boldsymbol{u} \in \mathcal{H}^k$, k is very large or $k > 1$ for categories A and B, respectively.

TABLE 2.1 Convergence types and rates for p–FEM based on problem category.

EXERCISE 2.1.— p–FEM SOLUTION FOR THE LAPLACE EQUATION

Consider the square plate Ω of unit side length, as shown in Figure 2.11a. It can be shown that the function

$$u = \operatorname{csch}(\pi)\sin(\pi x)\sinh(\pi - \pi y)\,, \tag{2.20}$$

where u represents a vector-valued scalar field (*e.g.*, temperature) as shown in Figure 2.11b, satisfies the Laplace equation $\Delta u = 0$. Solve the problem numerically by using the p–version of FEM and determine the exponent θ for this problem.

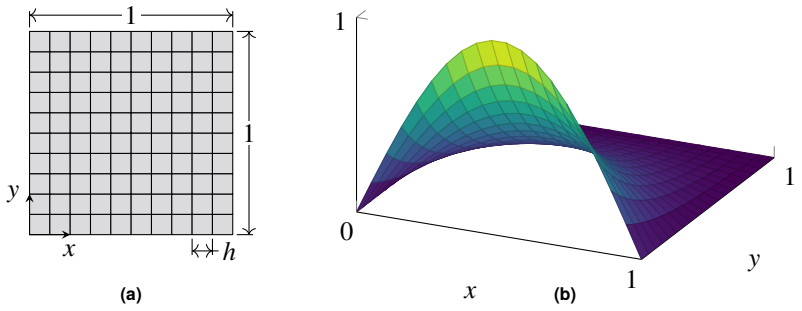

FIGURE 2.11 (a) Domain $\Omega = [0, 1] \times [0, 1]$ discretized with 4-node bilinear elements; (b) Exact solution (2.20).

Solution:

To solve this problem, an essential boundary condition $u = 0$ is prescribed in three edges of the plate. For the remaining edge, a natural boundary condition $\bar{t} = \pi \coth(\pi) \sin(\pi x)$. The problem is then solved for $p = \{1, 2, \ldots, 10\}$. The results are shown in Figures 2.12. A value $\theta = 0.65$ is determined for the simulation, even though this value decreases with the number of DOFs, with a minimum value $\theta = 0.56$ for the last two data points. It is shown that for $p = 10$ the results approach numerical machine precision.

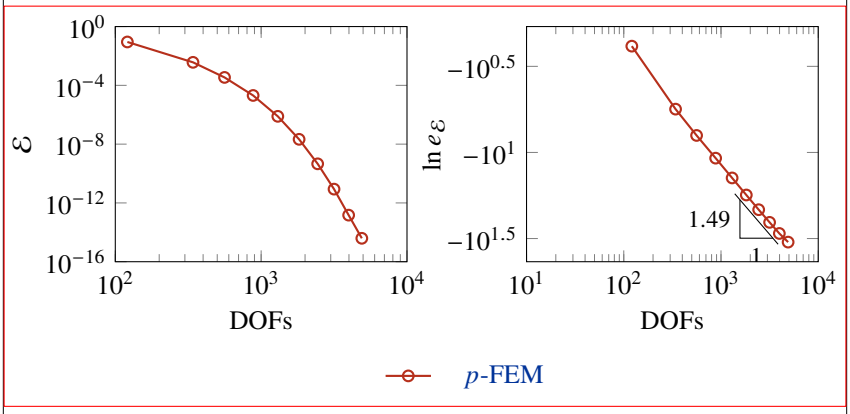

FIGURE 2.12 (a) Convergence plot; (b) Logarithm of the error in the energy norm as a function of DOFs.

2.3 NON-HOMOGENEOUS ESSENTIAL BOUNDARY CONDITIONS

An important topic that needs some discussion is how to apply non-zero essential boundary conditions. In 1-D this is straightforward because all shape functions

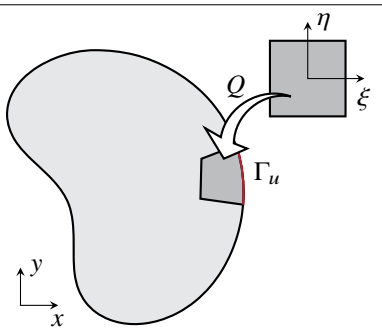

FIGURE 2.13 Quadrangular finite element has an entire edge on the Dirichlet boundary Γ_u.

higher than linear vanish at the end nodes of the element. In higher-dimensions this is no longer the case. The problem is schematized in Figure 2.13, where a quadrangular element has an edge over Γ_u, the part of the boundary where non-zero essential BCs are to be prescribed. Through the map $Q(\xi,\eta) : \hat{\Omega} \rightarrow \Omega$, the right edge corresponds to $\xi = 1$.

Let us consider a scalar problem with a quartic field interpolation. If the quadrangle were a Lagrange element, prescribing non-zero Dirichlet BCs on the right edge is straightforward because of the Kronecker-δ property of Lagrange shape functions, *i.e.*, $\varphi_i(\boldsymbol{x}_j) = \delta_{ij}$. The nodes on the right edge would be located equidistantly at coordinates $(1,\eta_i), \eta_i = \{-1,-0.5,0,0.5,1\}$. The field at the right edge is therefore written as

$$u(1,\eta) = \varphi_2(1,\eta)U_2 + \varphi_3(1,\eta)U_3 + \varphi_6(1,\eta)U_6 + \varphi_{10}(1,\eta)U_{10} + \varphi_{14}(1,\eta)U_{14}, \tag{2.21}$$

where only non-zero shape functions along the right edge were considered. Because of the Kronecker-δ property above, evaluating (2.21) is trivial:

$$\begin{aligned} u(1,-1) &= \varphi_2(1,-1)U_2 = U_2 = \bar{u}(1,-1), \\ u(1,-0.5) &= \varphi_3(1,-0.5)U_3 = U_3 = \bar{u}(1,-0.5), \\ u(1,0) &= \varphi_6(1,0)U_6 = U_6 = \bar{u}(1,0), \\ u(1,0.5) &= \varphi_{10}(1,0.5)U_{10} = U_{10} = \bar{u}(1,0.5), \\ u(1,1) &= \varphi_{14}(1,1)U_{14} = U_{14} = \bar{u}(1,1). \end{aligned} \tag{2.22}$$

Note, however, that the function to be prescribed over the boundary $\bar{u}$ is usually a function of global coordinates $\boldsymbol{x}$, thus above it is understood that $\bar{u}(\xi,\eta) \equiv \bar{u}(\boldsymbol{x}(\xi,\eta))$. Once the DOFs are known by solving Eqs. (2.22), they can be prescribed strongly.

In p–FEM prescribing non-zero BCs gets more involved because there is no Kronecker-δ property among p–hierarchical shape functions. Still, generalized DOFs of high-order shape functions can be obtained by solving a local problem,

and once known, they can still be enforced strongly as before. The field over the right edge, if using hierarchical shape functions, can be written as

$$u(1,\eta) = \underbrace{\varphi_2(1,\eta)U_2 + \varphi_3(1,\eta)U_3}_{\text{nodal functions}} + \underbrace{\hat{\varphi}_2(\eta)U_6 + \hat{\varphi}_3(\eta)U_{10} + \hat{\varphi}_4(\eta)U_{14}}_{\text{edge functions}}, \tag{2.23}$$

where the edge functions are written in terms of (2.3)—for instance $\hat{\varphi}_3$ is the cubic function. An important attribute of (2.23) is that all edge functions vanish at the end nodes of the edge, and it is this very property what allows us to solve a local problem to obtain DOFs of the nonlinear terms. However, there are many ways to accomplish this, so in the following we will explore two of them.

2.3.1 Interpolation at Gauss-Lobatto quadrature points

Once choice for evaluating (2.23) is to take the Gauss-Lobatto quadrature points, which are the roots of the polynomial L'_{p-1}, where L_p is the Legendre polynomial given by (2.1). Following the example given earlier for the Lagrange quartic element, we would evaluate (2.23) at $1\times\{\eta_i\}_{i=1}^5 = 1\times\left\{-1, -\sqrt{21}/7, 0, \sqrt{21}/7, 1\right\}$ and obtain a 5×5 system that can be solved for U_2, U_3, U_6, U_{10} and U_{14}. In fact, we know that U_2 and U_3 are readily available because nodal functions satisfy the Kronecker-δ property and are therefore uncoupled (all edge functions vanish at nodal points); therefore it is only necessary to solve a 3×3 system. To wit,

$$U_2 = \bar{u}(1,\eta_1), \qquad U_3 = \bar{u}(1,\eta_5), \tag{2.24}$$

and

$$\begin{bmatrix} \hat{\varphi}_2(\eta_2) & \hat{\varphi}_3(\eta_2) & \hat{\varphi}_4(\eta_2) \\ \hat{\varphi}_2(\eta_3) & \hat{\varphi}_3(\eta_3) & \hat{\varphi}_4(\eta_3) \\ \hat{\varphi}_2(\eta_4) & \hat{\varphi}_3(\eta_4) & \hat{\varphi}_4(\eta_4) \end{bmatrix} \begin{bmatrix} U_6 \\ U_{10} \\ U_{14} \end{bmatrix} = \begin{bmatrix} \bar{u}(1,\eta_2) - U_2\varphi_2(1,\eta_2) - U_3\varphi_3(1,\eta_2) \\ \bar{u}(1,\eta_3) - U_2\varphi_2(1,\eta_3) - U_3\varphi_3(1,\eta_3) \\ \bar{u}(1,\eta_4) - U_2\varphi_2(1,\eta_4) - U_3\varphi_3(1,\eta_4) \end{bmatrix}. \tag{2.25}$$

Once generalized DOFs are known they can be prescribed as in standard FEM.

2.3.2 L^2–projection on the space of edge functions

Consider in Figure 2.14 the function $\bar{u}$, which has been decomposed into linear and nonlinear components. The linear part is obtained by interpolating the extreme values on the interval $[-1,1]$ with the linear shape functions. Mathematically, $\bar{u}_{\text{lin}} = \bar{u}(-1)\varphi_2(\eta) + \bar{u}(1)\varphi_3(\eta)$, where φ_2 and φ_3 are, as earlier, the linear shape functions on the edge considered.

We now consider only the nonlinear part, *i.e.*, $\tilde{u}(\eta) = \bar{u}(\eta) - \bar{u}_{\text{lin}}$—which is exactly zero at the end nodes—and define the function space (dropping the dependence on η for brevity)

$$e = \tilde{u} - U_6\hat{\varphi}_2 - U_{10}\hat{\varphi}_3 - U_{14}\hat{\varphi}_4, \qquad U_6, U_{10}, U_{14} \in \mathbb{R}. \tag{2.26}$$

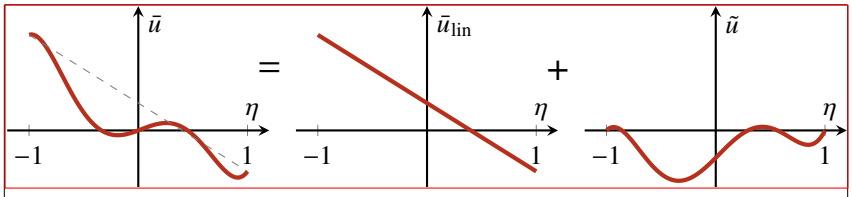

FIGURE 2.14 The function $\bar{u}$ to be prescribed on the boundary can be decomposed into linear $\bar{u}_{\text{lin}}$ and nonlinear $\tilde{u}$ components.

For a particular choice of DOFs U_6, U_{10}, U_{14} in (2.26), the resulting function measures the error between the nonlinear component of $\bar{u}$ and what can be interpolated by the nonlinear terms in our quartic approximation.

The objective is to then to find U_6, U_{10}, and U_{14} so that the L^2-norm $\|e\|_{L^2}$, induced by the inner product

$$\langle e, e \rangle = \int_{-1}^{+1} e^2 \, \mathrm{d}\eta = \|e\|_{L^2}^2 , \tag{2.27}$$

is minimized.

The coefficients that minimize (2.27), dropping henceforth the dependency on the independent variable η, are obtained by

$$\frac{\partial \|e\|_{L^2}^2}{\partial U_i} = 0, \qquad i = 6, 10, 14, \tag{2.28}$$

which results again in a 3×3 system of equations that can be solved for the generalized DOFs. The first equation is

$$\begin{aligned} \frac{\partial \|e\|_{L^2}^2}{\partial U_6} = -2 \int_{-1}^{+1} e\hat{\varphi}_2 \, \mathrm{d}\eta = -2 \int_{-1}^{+1} & \tilde{u}\hat{\varphi}_2 - U_6\hat{\varphi}_2^2 \\ & - U_{10}\hat{\varphi}_2\hat{\varphi}_3 - U_{14}\hat{\varphi}_2\hat{\varphi}_4 \, \mathrm{d}\eta = 0, \end{aligned} \tag{2.29}$$

and the other two equations are obtained analogously. The system of equations to solve is then given by

$$\begin{bmatrix} \langle \hat{\varphi}_2, \hat{\varphi}_2 \rangle & \langle \hat{\varphi}_2, \hat{\varphi}_3 \rangle & \langle \hat{\varphi}_2, \hat{\varphi}_4 \rangle \\ \langle \hat{\varphi}_2, \hat{\varphi}_3 \rangle & \langle \hat{\varphi}_3, \hat{\varphi}_3 \rangle & \langle \hat{\varphi}_3, \hat{\varphi}_4 \rangle \\ \langle \hat{\varphi}_2, \hat{\varphi}_4 \rangle & \langle \hat{\varphi}_3, \hat{\varphi}_4 \rangle & \langle \hat{\varphi}_4, \hat{\varphi}_4 \rangle \end{bmatrix} \begin{bmatrix} U_6 \\ U_{10} \\ U_{14} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \langle \tilde{u}, \hat{\varphi}_2 \rangle \\ \langle \tilde{u}, \hat{\varphi}_3 \rangle \\ \langle \tilde{u}, \hat{\varphi}_4 \rangle \end{bmatrix} . \tag{2.30}$$

Noteworthy, this approach is computationally more involved because it requires computing integrals of the symmetric coefficient matrix.

2.4 PROBLEMS

PROBLEM 2.1.— BOUNDARY VALUE PROBLEM WITH $u \in C^\infty$

Consider Problem 1.1 given in Chapter 1.

1. Solve the BVP with $x_b = 0.2$, $a = 0.5$ and 2 elements of degree $p = 1, \ldots, 5$. Make a log-log plot of the relative error in the energy norm versus the number of DOFs, and compare with the curves obtained for the h–version.
2. Solve the BVP with $x_b = 0.2$, $a = 50$ and 5 elements of degree $p = 1, \ldots, 5$. Make a convergence plot in the energy norm and compare with the h–convergence results.
3. Discuss (compare and draw conclusions) about your results. What conclusions can you draw from problems with strong gradients and the use of both the h– and the p–version of the FEM?

PROBLEM 2.1.— MATERIAL DISCONTINUITY

Solve Problem 1.2 with a bimaterial interface this time using the p–version of the FEM and uniform meshes with elements of degree $p = 1, \ldots, 5$. Use *i)* two elements, and *ii)* three elements. Discuss (compare and draw conclusions) your results.