Numerical Implementation – 1-D FEA Problems
Evaluation of Element Equations

Bar Element Equation Matrices

$$K_{ij} = \int_{x_a}^{x_b} AE \frac{d\psi_i}{dx} \frac{d\psi_i}{dx} dx \ , \quad f_i = \int_{x_a}^{x_b} f\psi_i dx$$

Beam Element Equation Matrices

$$K_{ij} = \int_{x_a}^{x_b} EI \frac{d^2\phi_i}{dx^2} \frac{d^2\phi_i}{dx^2} dx \ , \ F_i = \int_{x_a}^{x_b} q\phi_i dx$$

Computer codes want to be able to handle many different cases, and so they commonly use numerical methods to evaluate these integral expressions. This requires a special mapping of the element geometry onto the domain [-1,1] and numerical integration over this special domain.

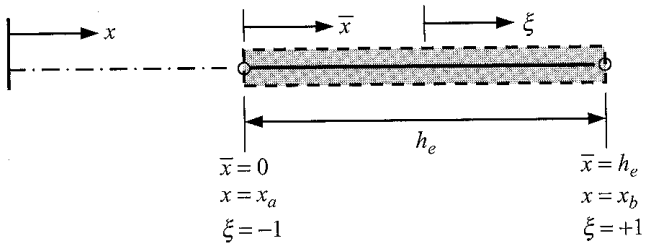

Figure 7.1.2 Global coordinate x, local coordinate $\bar{x}$, and normalized coordinate ξ.

7.1.2 Natural Coordinates

Of all the quadrature formulae, as will be discussed in the subsequent sections, the Gauss–Legendre one is the most commonly used. The details of the method itself will be discussed shortly. The formula requires the integral to be cast as one to be evaluated over the interval $[-1, 1]$. This requires the transformation of the problem coordinate x to a local coordinate ξ such that (see Fig. 7.1.2):

$$\text{when} \quad x = x_a, \quad \xi = -1; \quad \text{when} \quad x = x_b, \quad \xi = 1$$

The transformation between x and ξ can be represented by the linear "stretch" transformation

$$x = a + b\xi$$

where a and b are constants to be determined such that the above conditions hold:

$$x_a = a + b(-1), \quad x_b = a + b(1)$$

Solving for a and b, we obtain

$$b = \frac{1}{2}(x_b - x_a) = \frac{1}{2}h_e, \quad a = \frac{1}{2}(x_b + x_a) = x_a + \frac{1}{2}h_e$$

Hence, the transformation takes the form

$$x(\xi) = x_a + \frac{1}{2}h_e(1 + \xi) \tag{7.1.6}$$

where x_a denotes the global coordinate of the left-end node of the element Ω_e and h_e is the element length (see Fig. 7.1.2).

The local coordinate ξ is called the *normal coordinate* or *natural coordinate*, and its values always lie between -1 and 1, with its origin at the center of the element. The local coordinate ξ is useful in two ways: It is (*a*) convenient in constructing the interpolation functions and (*b*) required in numerical integration using Gauss–Legendre quadrature.

The derivation of the Lagrange family of interpolation functions in terms of the natural coordinate ξ is made easy by the following interpolation property of the approximation functions:

$$\psi_i(\xi_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{7.1.7}$$

where ξ_j is the ξ coordinate of the jth node in the element. For an element with n nodes, ψ_i $(i = 1, 2, \ldots, n)$ are polynomials of degree $n - 1$. To construct ψ_i satisfying (7.1.7), we

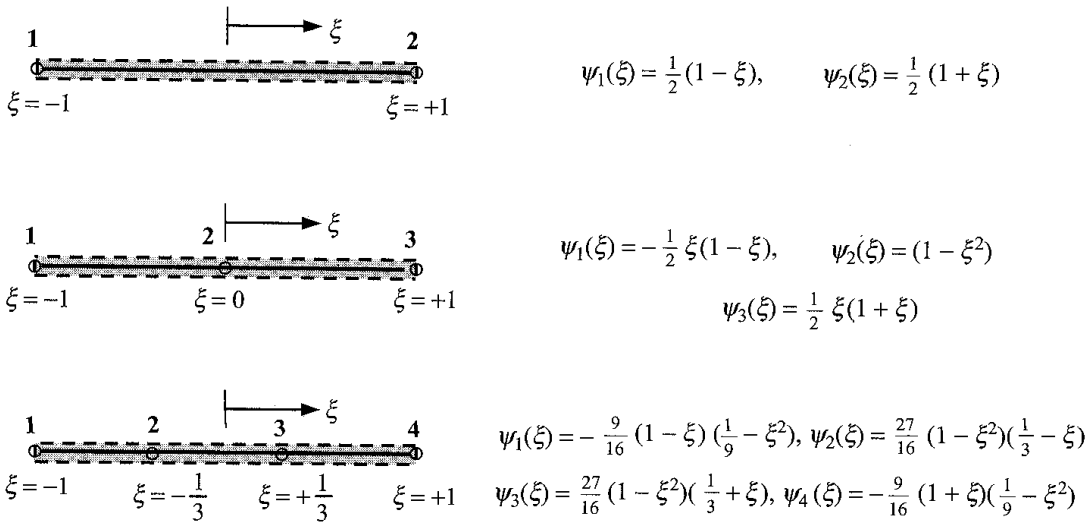

Figure 7.1.3 Lagrange family of one-dimensional interpolation functions in terms of the normalized coordinate.

proceed as follows: For each ψ_i, we form the product of $n-1$ linear functions $\xi - \xi_j$ ($j = 1, 2, \ldots, i-1, i+1, \ldots, n;\ j \neq i$):

$$\psi_i = c_i(\xi - \xi_1)(\xi - \xi_2)\cdots(\xi - \xi_{i-1})(\xi - \xi_{i+1})\cdots(\xi - \xi_n)$$

Note that ψ_i is zero at all nodes except the ith. Next, we determine the constant c_i such that $\psi_i = 1$ at $\xi = \xi_i$:

$$c_i = [(\xi_i - \xi_1)(\xi_i - \xi_2)\cdots(\xi_i - \xi_{i-1})(\xi_i - \xi_{i+1})\cdots(\xi_i - \xi_n)]^{-1}$$

Thus, the interpolation function associated with node i is

$$\psi_i(\xi) = \frac{(\xi - \xi_1)(\xi - \xi_2)\cdots(\xi - \xi_{i-1})(\xi - \xi_{i+1})\cdots(\xi - \xi_n)}{(\xi_i - \xi_1)(\xi_i - \xi_2)\cdots(\xi_i - \xi_{i-1})(\xi_i - \xi_{i+1})\cdots(\xi_i - \xi_n)} \tag{7.1.8}$$

The linear, quadratic, and cubic Lagrange interpolation functions in terms of the natural coordinate (for equally spaced nodes) are shown in Fig. 7.1.3.

7.1.3 Approximation of Geometry

We wish to use the Gauss–Legendre quadrature to numerically evaluate all integrals in the finite element method. The integrals are generally expressed in terms of the coordinate appearing in the problem description (like x or r). We shall call x and r as the *problem coordinates* or global coordinates. The Gauss–Legendre quadrature requires us to express the integral in terms of ξ over the interval -1 to $+1$. We assume a relation (or transformation) between the problem coordinate x and natural coordinate ξ in the form

$$x = f(\xi) \tag{7.1.9}$$

where f is assumed to be a one-to-one transformation. An example of $f(\xi)$ is provided by (7.1.6):

$$f(\xi) = x_a + \frac{1}{2}h_e(1 + \xi)$$

In this case, $f(\xi)$ is a linear function of ξ. Hence, a straight line is transformed into a straight line.

It is natural to think of approximating the geometry in the same way as we approximated a dependent variable. In other words, the transformation $x = f(\xi)$ can be written as

$$x = \sum_{i=1}^{m} x_i^e \hat{\psi}_i^e(\xi) \tag{7.1.10}$$

where x_i^e is the global coordinate of the ith node of the element Ω_e and $\hat{\psi}_i^e$ are the Lagrange interpolation functions of degree $m-1$. When $m=2$, we have a linear transformation, and Eq. (7.1.10) is exactly the same as (7.1.6). When $m=3$, Eq.(7.1.10) expresses a quadratic relation between x and ξ. The functions $\hat{\psi}_i^e$ are called *shape functions* because they are used to express the geometry or shape of the element. When the element is a straight line, the mapping is linear (because two points, x_1^e and x_2^e, are sufficient to define a line).

The transformation (7.1.10) allows us to rewrite integrals involving x as those in terms of ξ:

$$\int_{x_a}^{x_b} F(x)\,dx = \int_{-1}^{1} \hat{F}(\xi)\,d\xi, \quad \hat{F}(\xi)\,d\xi = F(x(\xi))\,dx \tag{7.1.11}$$

so that the Gauss–Legendre quadrature can be used to evaluate the integral over $[-1, 1]$. The differential element dx in the global coordinate system x is related to the differential element $d\xi$ in the natural coordinate system ξ by

$$dx = \frac{dx}{d\xi} d\xi = J_e\, d\xi$$

where J_e is called the *Jacobian* of the transformation. We have

$$J_e = \frac{dx}{d\xi} = \frac{d}{d\xi}\left(\sum_{i=1}^{m} x_i^e \hat{\psi}_i^e\right) = \sum_{i=1}^{m} x_i^e \frac{d\hat{\psi}_i^e}{d\xi} \tag{7.1.12}$$

For a linear transformation [i.e., when $m=2$ in (7.1.10)], we have

$$\hat{\psi}_1^e = \frac{1}{2}(1-\xi), \quad \hat{\psi}_2^e = \frac{1}{2}(1+\xi)$$

$$J_e = x_1^e\left(-\frac{1}{2}\right) + x_2^e\left(\frac{1}{2}\right) = \frac{1}{2}(x_2^e - x_1^e) = \frac{1}{2}h_e \tag{7.1.13}$$

It can be shown that $J_e = \frac{1}{2}h_e$ whenever the element is a straight line, irrespective of the degree of interpolation used in the transformation (7.1.10).

7.1.4 Isoparametric Formulations

Recall that a dependent variable u is approximated in an element Ω_e by expressions of the form

$$u(x) = \sum_{j=1}^{n} u_j^e \psi_j^e(x) \tag{7.1.14}$$

In general, the degree of approximation used to describe the coordinate transformation (7.1.10) is not equal to the degree of approximation (7.1.14) used to represent a dependable

variable, i.e., $\hat{\psi}_i^e \neq \psi_i^e$. In other words, two independent meshes of elements may be used in the finite element formulation of a problem: one for the approximation of the geometry x and the other for the interpolation of the dependent variable u. Depending on the relationship between the degree of approximation used for the coordinate transformation and that used for the dependent variable, the finite element formulations are classified into three categories:

1. Subparametric formulations: $m < n$
2. Isoparametric formulations: $m = n$
3. Superparametric formulations: $m > n$

(7.1.15)

In subparametric formulations, the geometry is represented by lower-order elements than those used to approximate the dependent variable. An example of this category is provided by the Euler–Bernoulli beam element, where the Hermite cubic functions are used to approximate the deflection $w(x)$ and linear interpolation can be used, when straight beams are analyzed, to represent the geometry. In isoparametric formulations (which are the most common in practice), the same element is used to approximate the geometry as well as the dependent unknowns: $\psi_i(x) = \hat{\psi}_i(\xi)$. In the superparametric formulations, the geometry is represented with higher-order elements than those used to approximate the dependent variables. The superparametric formulation is seldom used in practice.

7.1.5 Numerical Integration

As discussed in the introduction, the evaluation of integrals of the form

$$\int_a^b F(x)\, dx \tag{7.1.16}$$

by exact means is either difficult or impossible owing to the complicated form of the integrand F. Numerical integration is also required when the integrand is to be evaluated inexactly (as in the Timoshenko beam element) or when the integrand is known only at discrete points (e.g., experimentally obtained data).

The basic idea behind all numerical integration techniques is to find a function $P(x)$, often a polynomial, that is both a suitable approximation of $F(x)$ and simple to integrate. The interpolating polynomials of degree n, denoted by P_n, which interpolate the integrand at $n + 1$ points of the interval $[a, b]$, often produce a suitable approximation and possess the desired property of simple integrability. An illustration of the approximation of the function $F(x)$ by the polynomial $P_4(x)$ that exactly matches the function $F(x)$ at the indicated base points is given in Fig. 7.1.4(a). The exact value of (7.1.16) is given by the area under the solid curve, while the approximate value

$$\int_a^b P_4(x)\, dx$$

is given by the area under the dashed curve. It should be noted that the difference (i.e., the error in the approximation) $E = F(x) - P_4(x)$ is not always of the same sign, and therefore the overall integration error may be small (because positive errors in one part cancel negative errors in other parts), even when P_4 is not a good approximation of F.

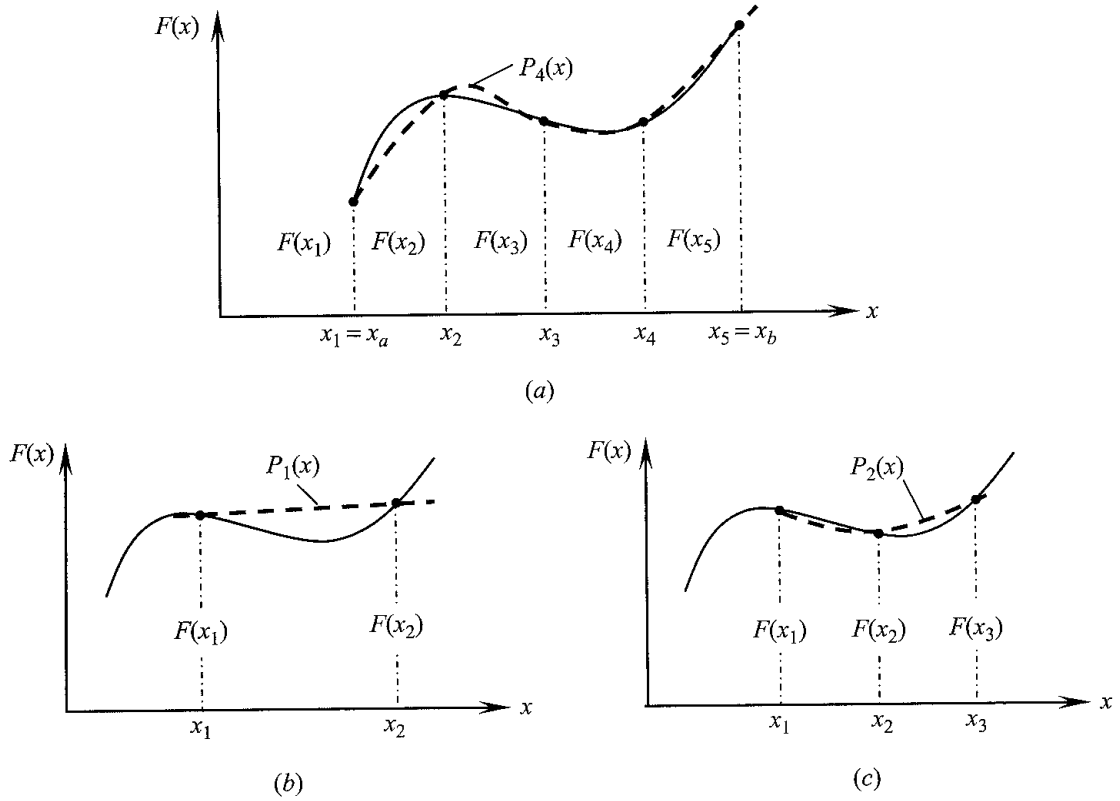

Figure 7.1.4 Numerical integration by the Newton–Cotes quadrature: (*a*) approximation of a function by $P_4(x)$; (*b*) the trapezoidal rule; and (*c*) Simpson's rule.

The commonly used numerical integration methods can be classified into two groups:

1. The Newton–Cotes formulae that employ values of the function at equally spaced points
2. The Gauss–Legendre quadrature formula that employs unequally spaced points

These two methods are described here.

The Gauss–Legendre Quadrature. In the Newton–Cotes quadrature, the base point locations have been specified. If the x_I are not specified, then there will be $2r + 2$ undetermined parameters, $r + 1$ weights w_I and $r + 1$ base points x_I, which define a polynomial of degree $2r + 1$. The Gauss–Legendre quadrature is based on the idea that the base points x_I and the weights w_I can be chosen so that the sum of the $r + 1$ appropriately weighted values of the function yields the integral exactly when $F(x)$ is a polynomial of degree $2r + 1$ or less. The Gauss–Legendre quadrature formula is given by

$$\int_a^b F(x)\,dx = \int_{-1}^{1} \hat{F}(\xi)\,d\xi \approx \sum_{I=1}^{r} \hat{F}(\xi_I) w_I \tag{7.1.20}$$

where w_I are the weight factors, ξ_I are the base points [roots of the Legendre polynomial $P_{r+1}(\xi)$], and $\hat{F}$ is the transformed integrand

$$\hat{F}(\xi) = F(x(\xi))J(\xi), \quad dx = J\,d\xi \tag{7.1.21}$$

Table 7.1.2 Weights and Gauss points for the Gauss–Legendre quadrature.

$$\int_{-1}^{1} F(\xi)d\xi = \sum_{i=1}^{r} F(\xi_i)w_i$$

Points, $\xi_i^{\dagger}$	r	Weights, w_i
0.0000000000	1	2.0000000000
±0.5773502692	2	1.0000000000
0.0000000000	3	0.8888888889
±0.7745966692		0.5555555555
±0.3399810435	4	0.6521451548
±0.8611363116		0.3478548451
0.0000000000	5	0.5688888889
±0.5384693101		0.4786286705
±0.9061798459		0.2369268850
±0.2386191861	6	0.4679139346
±0.6612093865		0.3607615730
±0.9324695142		0.1713244924

†Note that $0.57735\ldots = 1/\sqrt{3}, 0.77459\ldots = \sqrt{3/5}$, and $0.888\ldots = 8/9$, and $0.555\ldots = 5/9$.

where J is the Jacobian of the transformation between x anbd ξ. The weight factors and Gauss points for the Gauss–Legendre quadrature (7.1.20) are given for $r = 1, \ldots, 6$ in Table 7.1.2.

The Gauss–Legendre quadrature is more frequently used than the Newton–Cotes quadrature because it requires fewer base points (hence, a saving in computation) to achieve the same accuracy. The error in the approximation is zero if the $(2r + 2)$th derivative of the integrand vanishes. In other words, a polynomial of degree p is integrated exactly by employing $r = \frac{1}{2}(p + 1)$ Gauss points. When $p + 1$ is odd, one should pick the nearest larger integer:

$$r = \left[\frac{1}{2}(p+1)\right] \tag{7.1.22}$$

In finite element formulations, we encounter integrals whose integrands F are functions of x, $\psi_i(x)$, and derivatives of $\psi_i(x)$ with respect to x. For the Gauss–Legendre quadrature, we must transform $F(x)dx$ to $\hat{F}(\xi)d\xi$ in order to use the formula (7.1.20). For example, consider the integral

$$K_{ij}^e = \int_{x_a}^{x_b} a(x)\frac{d\psi_i^e}{dx}\frac{d\psi_j^e}{dx}\,dx \tag{7.1.23}$$

Using the chain rule of differentiation, we have

$$\frac{d\psi_i^e(x)}{dx} = \frac{d\psi_i^e(\xi)}{d\xi}\frac{d\xi}{dx} = J^{-1}\frac{d\psi_i^e(\xi)}{d\xi} \tag{7.1.24}$$

Therefore, the integral in (7.1.23) can be written, with the help of (7.1.10), as

$$K_{ij}^e = \int_{-1}^{1} a(x(\xi))\frac{1}{J}\frac{d\psi_i^e}{d\xi}\frac{1}{J}\frac{d\psi_j^e}{d\xi}\,J\,d\xi \tag{7.1.25}$$

$$\approx \sum_{I=1}^{r} \hat{F}_{ij}^e(\xi_I)w_I \tag{7.1.26}$$

where

$$\hat{F}_{ij}^e = a\frac{1}{J}\frac{d\psi_i^e}{d\xi}\frac{d\psi_j^e}{d\xi}, \qquad J = \sum_{i=1}^{m} x_i^e \frac{d\hat{\psi}_i^e}{d\xi} \tag{7.1.27}$$

For the isoparametric formulation, we take $\psi_i^e = \hat{\psi}_i^e$. As noted earlier, the Jacobian matrix will be the same ($J_e = \frac{1}{2}h_e$) when the element is a straight line, even if the coordinate transformation is quadratic or cubic. However, when the element is curved, the Jacobian is a function of ξ for transformations other than linear. The transformation from x to ξ is not required in the Newton–Cotes quadrature.

It is possible to determine the exact number of Gauss points required to evaluate the following element coefficients:

$$\begin{aligned}
K_{ij}^e &= \int_{x_a}^{x_b} \frac{d\psi_i^e}{dx}\frac{d\psi_j^e}{dx}\,dx = \int_{-1}^{+1} \frac{d\psi_i^e}{d\xi}\frac{d\psi_j^e}{d\xi}(J)^{-2}\,J\,d\xi \equiv \int_{-1}^{+1} G_{ij}^K(\xi)\,d\xi \\
&\approx \sum_{I=1}^{N^K} G_{ij}^K(\xi_I)\,W_I \\
M_{ij}^e &= \int_{x_a}^{x_b} \psi_i^e \psi_j^e\,dx = \int_{-1}^{+1} \psi_i^e(\xi)\,\psi_j^e(\xi)\,J\,d\xi \equiv \int_{-1}^{+1} G_{ij}^M(\xi)\,d\xi \\
&\approx \sum_{I=1}^{N^M} G_{ij}^M(\xi_I)\,W_I \\
f_i^e &= \int_{x_a}^{x_b} \psi_i^e\,dx = \int_{-1}^{+1} \psi_i^e(\xi)\,J\,d\xi \equiv \int_{-1}^{+1} G_{ij}^F(\xi)\,d\xi \\
&\approx \sum_{I=1}^{N^F} G_{ij}^F(\xi_I)\,W_I
\end{aligned} \tag{7.1.28}$$

when linear, quadratic, and cubic interpolation functions are used. For linear interpolation functions, the integrand of K_{ij}^e is constant, requiring only one-point Gauss–Legendre quadrature ($N^K = 1$). The integrand of the mass matrix M_{ij}^e is quadratic ($p = 2$), requiring [$r = \frac{1}{2}(p+1) = \frac{3}{2}$], the two-point quadrature ($N^M = 2$). The coefficients f_i^e are evaluated exactly by one-point quadrature ($N^F = 1$). Similarly, for quadratic and cubic elements, we can estimate the number of Gauss points needed to evaluate K_{ij}^e, M_{ij}^e, and f_i^e exactly. The results are summarized in Table 7.1.3. Note that, in estimating the quadrature points, it is assumed that the Jacobian J is a constant, which holds true when the element is a straight line.

If the matrices in (7.1.28) have variable coefficients or the elements are curved [and hence $J_e = J_e(\xi)$], the degree of the variation of the integrands changes and the number of Gauss points needed to exactly evaluate the integral changes.

Functional Units in Standard Finite Element Computer Codes