

# Chapter 4

## ONE-DIMENSIONAL INTEGRATION

### 4.1 Introduction

Since the finite element method is based on integral relations it is logical to expect that one should strive to carry out the integrations as efficiently as possible. In some cases we will employ exact integration. In others we may find that the integrals can become too complicated to integrate exactly. In such cases the use of numerical integration will prove useful or essential. The important topics of local coordinate integration and Gaussian quadratures will be introduced here. They will prove useful when dealing with higher order interpolation functions in complicated element integrals.

### 4.2 Local Coordinate Jacobian

We have previously seen that the utilization of local element coordinates can greatly reduce the algebra required to establish a set of interpolation functions. Later we will see that some 2-D elements must be formulated in local coordinates in order to meet the interelement continuity requirements. Thus, we should expect to often encounter local coordinate interpolation. However, the governing integral expressions must be evaluated with respect to a unique global or physical coordinate system. Clearly, these two coordinate systems must be related. The relationship for integration with a change of variable (change of coordinate) was defined in elementary concepts from calculus. At this point it would be useful to review these concepts from calculus. Consider a definite integral

$$I = \int_a^b f(x) dx, \quad a < x < b \quad (4.1)$$

where a new variable of integration,  $r$ , is to be introduced such that  $x = x(r)$ . Here it is required that the function  $x(r)$  be continuous and have a continuous derivative in the interval  $\alpha \leq r \leq \beta$ . The region of  $r$  directly corresponds to the region of  $x$  such that when  $r$  varies between  $\alpha$  and  $\beta$ , then  $x$  varies between  $a = x(\alpha)$  and  $b = x(\beta)$ . In that case

$$I = \int_a^b f(x) dx = \int_\alpha^\beta f(x(r)) \frac{dx}{dr} dr \quad (4.2)$$

or

$$I = \int_{\alpha}^{\beta} f(r) J dr \quad (4.3)$$

where  $J = dx/dr$  is called the *Jacobian* of the coordinate transformation.

### 4.3 Exact Polynomial Integration \*

If we utilize the unit coordinates, then  $\alpha = 0$  and  $\beta = 1$ . Then from Sec. 3.2, the Jacobian is  $J = L^e$  in an element domain defined by linear interpolation. By way of comparison, if one employs natural coordinates, then  $\alpha = -1$ ,  $\beta = +1$ , and from Eq. 3.16  $J = L^e/2$ . Generally, we will use interpolation functions that are polynomials. Thus, the element integrals of them and/or their derivatives will also contain polynomial terms. Therefore, it will be useful to consider expressions related to typical polynomial terms. A typical polynomial term is  $r^m$  where  $m$  is an integer. Thus, from the above

$$I = \int_{x_1^e}^{x_2^e} r^m dx = \int_0^1 r^m L^e dr = L^e \left. \frac{r^{(1+m)}}{1+m} \right|_0^1 = \frac{L^e}{(1+m)}. \quad (4.4)$$

A similar expression can be developed for the natural coordinates. It gives

$$I = \int_{L^e} n^m dx = \frac{L^e}{m+1} \begin{cases} 0 & \text{if } m \text{ is odd} \\ 1 & \text{if } m \text{ is even} . \end{cases} \quad (4.5)$$

Later we will tabulate the extension of these concepts to two- and three-dimensional integrals. As an example of the use of Eq. 4.5, consider the integration of the outer product of  $\mathbf{H}^e$  with itself:

$$I = \int_{L^e} \mathbf{H}^{eT} \mathbf{H}^e dx .$$

Recall that the integral of a matrix is the matrix resulting from the integration of each of the elements of the original matrix. If linear interpolation is selected for  $\mathbf{H}^e$  on a line element then typical terms will include  $H_1^2$ ,  $H_1 H_2$ , etc. Thus, one obtains :

$$I_{11} = \int_{L^e} H_1^2(r) dx = \int_{L^e} (1-r)^2 dx = L^e(1 - 2/2 + 1/3) = L^e/3$$

$$I_{12} = \int_{L^e} H_1 H_2 dx = \int_{L^e} (1-r) r dx = L^e (1/2 - 1/3) = L^e/6$$

$$I_{22} = \int_{L^e} H_2^2 dx = \int_{L^e} r^2 dx = L^e/3$$

so that

$$I = \frac{L^e}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}. \quad (4.6)$$

Similarly, if one employs the Lagrangian quadratic  $\mathbf{H}$  in Eq. 4.17 one obtains :

$$I = \frac{L^e}{30} \begin{bmatrix} 4 & -1 & 2 \\ -1 & 4 & 2 \\ 2 & 2 & 16 \end{bmatrix}. \tag{4.7}$$

By way of comparison, if one selects the hierarchical quadratic polynomial in Eq. 4.22 the above integral becomes

$$I = \frac{L^e}{6} \begin{bmatrix} 2 & 1 & | & -1/4 \\ 1 & 2 & | & -1/4 \\ - & - & | & - \\ -1/4 & -1/4 & | & 1/10 \end{bmatrix}.$$

Note that the top left portion of this equation is the same as Eq. 4.7 which was obtained from the linear polynomial. This desirable feature of hierarchical elements was mentioned in Sec. 3.6.

Of course, all of the above integrals are so simple one could note that the integral is simply the area under the curve and there are simple algebraic relations for defining the area under power law curves. Figure 4.3.1 reviews such relations and illustrates the terms needed in Eq. 4.6.

Before leaving the subject of simplex integrations one should give consideration to the common special case of axisymmetric geometries, with coordinates  $(\rho, z)$ . Recall from calculus that the Theorem of Pappus relates a differential volume and surface area to a differential area and length in the  $(\rho, z)$  plane of symmetry, respectively. That is,  $dV = 2\pi\rho dA$  and  $dS = 2\pi\rho dl$ , where  $\rho$  denotes the radial distance to the differential element. Thus, typical axisymmetric surface integrals reduce to

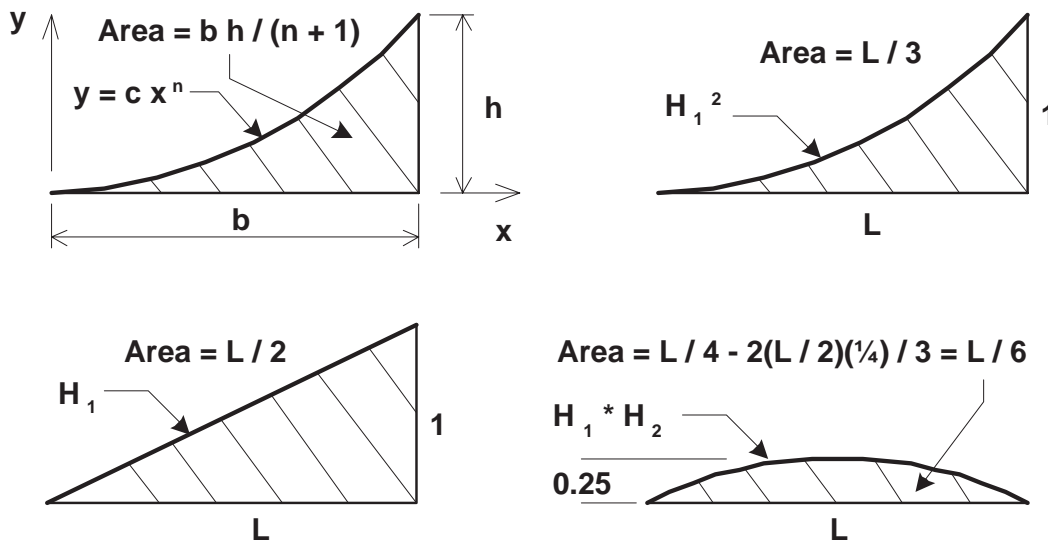


Figure 4.3.1 Exact integrals by inspection

$$\mathbf{I}_\Gamma = \int_\Gamma \mathbf{H}^T d\Gamma = 2\pi \int_{L^e} \mathbf{H}^T \rho dl = 2\pi \left( \int_{L^e} \mathbf{H}^T \mathbf{H} dl \right) \boldsymbol{\rho}^e = \frac{2\pi L^e}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \boldsymbol{\rho}^e,$$

since  $\rho = \mathbf{H}\boldsymbol{\rho}^e$ . Many workers like to omit the  $2\pi$  term and work on a per-unit-radian basis so that they can more easily do both two-dimensional and axisymmetric calculations with a single program.

### 4.4 Numerical Integration

*Numerical integration* is simply a procedure that approximates (usually) an integral by a summation. To review this subject we refer to Fig. 4.4.1. Recall that the integral

$$I = \int_a^b f(x) dx \tag{4.8}$$

can be viewed graphically as the area between the  $x$ -axis and the curve  $y = f(x)$  in the region of the limits of integration. Thus, we can interpret numerical integration as an approximation of that area. The *trapezoidal rule* of numerical integration simply approximates the area by the sum of several equally spaced trapezoids under the curve between the limits of  $a$  and  $b$ . The height of a trapezoid is found from the integrand,  $y_j = y(x_j)$ , evaluated at equally spaced points,  $x_j$  and  $x_{j+1}$ . Thus, a typical contribution is  $A = h(y_j + y_{j+1})/2$ , where  $h = x_{j+1} - x_j$  is the spacing. Thus, for  $n_q$  points (and  $n_q - 1$  spaces), the well-known approximation is

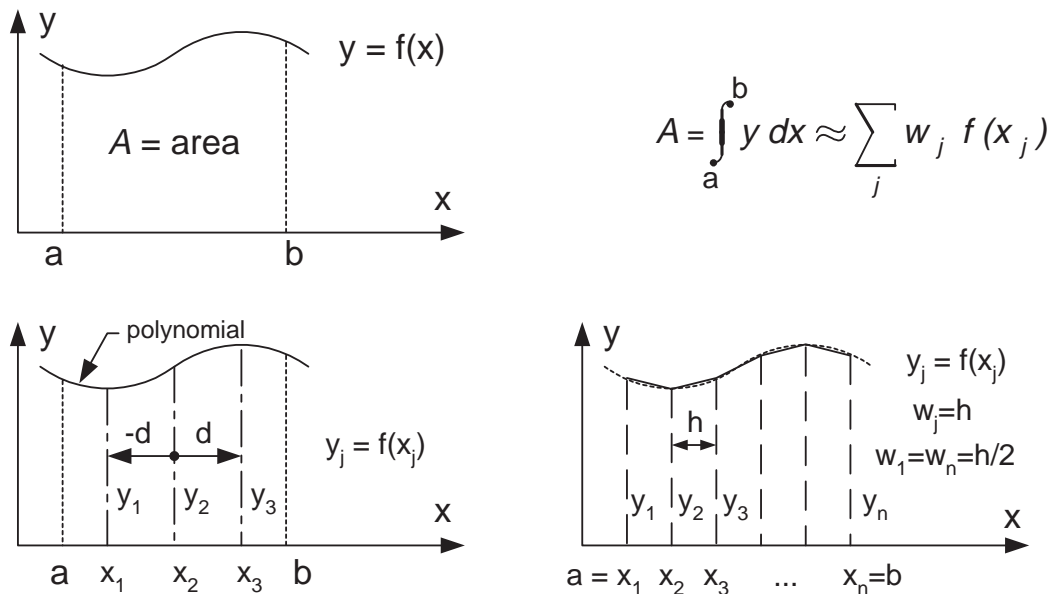


Figure 4.4.1 One-dimensional numerical integration

$$I \approx h \left( \frac{1}{2} y_1 + y_2 + y_3 + \dots + y_{n-1} + \frac{1}{2} y_n \right), \quad I \approx \sum_{j=1}^n w_j f(x_j) \quad (4.9)$$

where  $w_j = h$ , except  $w_1 = w_n = h/2$ . A geometrical interpretation of this is that the area under curve,  $I$ , is the sum of the products of certain heights,  $f(x_j)$  times some corresponding widths,  $w_j$ . In the terminology of numerical integration, the locations of the points,  $x_j$ , where the heights are computed are called *abscissae* and the widths,  $w_j$ , are called *weights*. Another well-known approximation is the *Simpson rule*, which uses parabolic segments in the area approximation. For most functions the above rules may require 20 to 40 terms in the summation to yield acceptable accuracy. We want to carry out the summation with the minimum number of terms,  $n_q$ , in order to reduce the computational cost. What is the minimum number of terms? The answer depends on the form of the integrand  $f(x)$ . Since the parametric geometry usually involves polynomials we will consider that common special case for  $f(x)$ .

The famous mathematician Gauss posed this question: What is the minimum number of points,  $n_q$ , required to exactly integrate a polynomial, and what are the corresponding abscissae and weights? If we require the summation to be exact when  $f(x)$  is any one of the power functions  $1, x, x^2, \dots, x^{2n-1}$ , we obtain a set of  $2n$  conditions that allow us to determine the  $n_q$  abscissae,  $x_i$ , and their corresponding  $n_q$  weights,  $w_j$ . The  $n_q$  *Gaussian quadrature* points are symmetrically placed with respect to the center of the interval, and will exactly integrate a polynomial of order  $(2n_q - 1)$ . The center point location is included in the abscissae list when  $n_q$  is odd, but the end points are never utilized in a Gauss rule. The Gauss rule data are usually tabulated for a non-dimensional *unit coordinate* range of  $0 \leq t \leq 1$ , or for a *natural coordinate* range of  $-1 \leq t \leq +1$ . Table 4.1 presents the low-order Gauss rule data in natural coordinates, and the alternate unit coordinate data are in Table 4.2. A two-point Gauss rule can often exceed the accuracy of a 20-point trapezoidal rule. When computing norms of an exact solution, to be compared to the finite element solution, we often use the trapezoidal rule. That is because the exact solution is usually not a polynomial and the Gauss rule may not be accurate.

Sometimes it is desirable to have a numerical integration rule that specifically includes the two end points in the abscissae list when  $(n \geq 2)$ . The *Lobatto rule* is such an alternate choice. Its  $n_q$  points will exactly integrate a polynomial of order  $(2n - 3)$  for  $n_q > 2$ . Its data are included in Table 4.3. It is usually less accurate than the Gauss rule but it can be useful. Mathematical handbooks give tables of Gauss or Lobatto data for much higher values of  $n_q$ . Some results of Gauss's work are outlined below. Let  $y$  denote  $f(x)$  in the integral to be computed. Define a change of variable

$$x(n) = 1/2(b - a)n + 1/2(b + a) \quad (4.10)$$

so that the non-dimensional limits of integration of  $n$  become  $-1$  and  $+1$ . The new value of  $y(n)$  is

$$y = f(x) = f[1/2(b - a)n + 1/2(b + a)] = \Phi(n). \quad (4.11)$$

Noting from Eq. 4.10 that  $dx = 1/2(b - a)dn$ , the original integral becomes

<b>Table 4.1 Abscissas and weights for Gaussian quadrature</b>										
$\int_{-1}^{+1} f(x) dx = \sum_{i=1}^{n_q} w_i f(x_i)$										
$\pm x_i$						$w_i$				
0.00000	00000	00000	00000	0000	$n_q = 1$	2.00000	00000	00000	00000	000
0.57735	02691	89625	76450	9149	$n_q = 2$	1.00000	00000	00000	00000	000
0.77459	66692	41483	37703	5835	$n_q = 3$	0.55555	55555	55555	55555	556
0.00000	00000	00000	00000	0000		0.88888	88888	88888	88888	889
0.86113	63115	94052	57522	3946	$n_q = 4$	0.34785	48451	37453	85737	306
0.33998	10435	84856	26480	2666		0.65214	51548	62546	14262	694
0.90617	98459	38663	99279	7627	$n_q = 5$	0.23692	68850	56189	08751	426
0.53846	93101	05683	09103	6314		0.47862	86704	99366	46804	129
0.00000	00000	00000	00000	0000		0.56888	88888	88888	88888	889
0.93246	95142	03152	02781	2302	$n_q = 6$	0.17132	44923	79170	34504	030
0.66120	93864	66264	51366	1400		0.36076	15730	48138	60756	983
0.23861	91860	83196	90863	0502		0.46791	39345	72691	04738	987
0.94910	79123	42758	52452	6190	$n_q = 7$	0.12948	49661	68869	69327	061
0.74153	11855	99394	43986	3865		0.27970	53914	89276	66790	147
0.40584	51513	77397	16690	6607		0.38183	00505	05118	94495	037
0.00000	00000	00000	00000	0000		0.41795	91836	73469	38775	510

<b>Table 4.2 Unit abscissas and weights for Gaussian quadrature</b>										
$\int_0^1 f(x) dx = \sum_{i=1}^{n_q} w_i f(x_i)$										
$x_i$						$w_i$				
0.50000	00000	00000	00000	000	$n_q = 1$	1.00000	00000	00000	00000	000
0.21132	48654	05187	11774	543	$n_q = 2$	0.50000	00000	00000	00000	000
0.78867	51345	94812	88225	457		0.50000	00000	00000	00000	000
0.11270	16653	79258	31148	208	$n_q = 3$	0.27777	77777	77777	77777	778
0.50000	00000	00000	00000	000		0.44444	44444	44444	44444	444
0.88729	83346	20741	68851	792		0.27777	77777	77777	77777	778
0.06943	18442	02973	71238	803	$n_q = 4$	0.17392	74225	68726	92868	653
0.33000	94782	07571	86759	867		0.32607	25774	31273	07131	347
0.66999	05217	92428	13240	133		0.32607	25774	31273	07131	347
0.93056	81557	97026	28761	197		0.17392	74225	68726	92868	653
0.04691	00770	30668	00360	119	$n_q = 5$	0.11846	34425	28094	54375	713
0.02307	65344	94715	84544	818		0.23931	43352	49683	23402	065
0.50000	00000	00000	00000	000		0.28444	44444	44444	44444	444
0.76923	46550	52841	54551	816		0.23931	43352	49683	23402	065
0.95308	99229	69331	99639	881		0.11846	34425	28094	54375	713

**Table 4.3. Abscissas and weight factors for Lobatto integration**

$$\int_{-1}^{+1} f(x) dx \approx \sum_{i=1}^{n_q} w_i f(x_i)$$

$\pm x_i$		$w_i$
0.00000 00000 00000	$n_q = 1$	2.00000 00000 00000
1.00000 00000 00000	$n_q = 2$	1.00000 00000 00000
1.00000 00000 00000 0.00000 00000 00000	$n_q = 3$	0.33333 33333 33333 1.33333 33333 33333
1.00000 00000 00000 0.44721 35954 99958	$n_q = 4$	0.16666 66666 66667 0.83333 33333 33333
1.00000 00000 00000 0.65465 36707 07977 0.00000 00000 00000	$n_q = 5$	0.10000 00000 00000 0.54444 44444 44444 0.71111 11111 11111
1.00000 00000 00000 0.76505 53239 29465 0.28523 15164 80645	$n_q = 6$	0.06666 66666 66667 0.37847 49562 97847 0.55485 83770 35486

$$I = \frac{1}{2} (b - a) \int_{-1}^1 \Phi(n) dn . \tag{4.12}$$

Gauss showed that the integral in Eq. 4.12 is given by

$$\int_{-1}^1 \Phi(n) dn = \sum_{i=1}^{n_q} W_i \Phi(n_i) ,$$

where  $W_i$  and  $n_i$  represent tabulated values of the *weight functions* and *abscissas* associated with the  $n_q$  points in the non-dimensional interval  $(-1, 1)$ . The final result is

$$I = \frac{1}{2} (b - a) \sum_{i=1}^{n_q} W_i \Phi(n_i) = \sum_{i=1}^{n_q} f(x(n_i)) W_i . \tag{4.13}$$

Gauss also showed that this equation will exactly integrate a polynomial of degree  $(2n_q - 1)$ . For a higher number of space dimensions (which range from  $-1$  to  $+1$ ), one obtains a multiple summation. Since Gaussian quadrature data are often tabulated in references for the range  $-1 \leq n \leq +1$ , it is popular to use the natural coordinates in defining element integrals. However, one can convert the tabulated data to any convenient system such as the unit coordinate system where  $0 \leq r \leq 1$ . The latter may be more useful on triangular regions. As an example of Gaussian quadratures, consider the following one-dimensional integral :

$$I = \int_1^2 \left[ \begin{array}{cc} 2 & 2x \\ 2x & (1 + 2x^2) \end{array} \right] dx = \int_1^2 \mathbf{F}(x) dx .$$

If two Gauss points are selected ( $n_q = 2$ ), then the tabulated values from Table 4.1 give  $W_1 = W_2 = 1$  and  $r_1 = 0.57735 = -r_2$ . The change of variable gives  $x(r) = (r + 3)/2$ , so that  $x(r_1) = 1.788675$  and  $x(r_2) = 1.211325$ . Therefore, from Eq. 4.13

$$\begin{aligned}
I &= \frac{1}{2} (2 - 1) \left[ W_1 \mathbf{F}(x(r_1)) + W_2 \mathbf{F}(x(r_2)) \right] \\
&= \frac{1}{2} (1) \left( (1) \begin{bmatrix} 2 & 2(1.788675) \\ \text{sym.} & 1 + 2(1.788675)^2 \end{bmatrix} + (1) \begin{bmatrix} 2 & 2(1.211325) \\ \text{sym.} & 1 + 2(1.211325)^2 \end{bmatrix} \right) \\
I &= \begin{bmatrix} 2.00000 & 3.00000 \\ 3.00000 & 5.66667 \end{bmatrix},
\end{aligned}$$

which is easily shown to be in good agreement with the exact solution. As another example consider a typical term in Eq. 4.8. Specifically, from Eqs. 4.16 and 4.18

$$I_{33} = \int_{L^e} H_3^2 dx = \frac{L^e}{2} \int_{-1}^{+1} (1 - n^2)^2 dn .$$

Since the polynomial terms to be integrated are fourth order, we should select  $(2n_q - 1) = 4$ , or  $n_q = 3$  for an integer number of Gaussian points. Then,

$$\begin{aligned}
I_{33} &= \int_{L^e} H_3^2 dx = \frac{L^e}{2} \int_{-1}^{+1} H_3^2(n) dn \\
I_{33} &= \frac{L^e}{2} \left( 0.55556 \left[ 1.00000 - (-0.77459)^2 \right]^2 \right. \\
&\quad \left. + 0.88889 \left[ 1.00000 - (0.0)^2 \right]^2 + 0.55556 \left[ 1.00000 - (+0.77459)^2 \right]^2 \right) \\
I_{33} &= \frac{L^e}{2} (0.08889 + 0.88889 + 0.08889) = 0.5333 L^e ,
\end{aligned}$$

which agrees well with the exact value of  $16 L^e / 30$ , when using 6 digits.

## 4.5 Variable Jacobians

When the parametric space and physical space have the same number of dimensions then the Jacobian is a square matrix. Otherwise, we need to use more calculus to evaluate the integrals. For example, we often find the need to execute integrations along a two-dimensional curve defined by our one-dimensional parametric representation. Consider a planar boundary curve in the  $xy$ -plane such as that shown in Fig. 4.4.2. We may need to know its length and first moments (centroid), which are defined as

$$L = \int_L ds , \quad \bar{x} L = \int_L x(r) ds , \quad \bar{y} L = \int_L y(r) ds ,$$

respectively, where  $ds$  denotes the physical length of a segment,  $dr$ , of the parametric length. To evaluate these quantities we need to convert to an integral in the parametric space. For example,

$$L = \int_L ds = \int_0^1 \frac{ds}{dr} dr .$$

To relate the physical and parametric length scales, we use the planar relation that

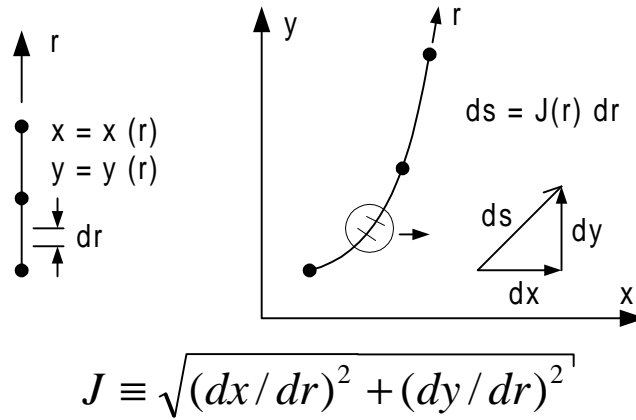


Figure 4.4.2 A variable curve metric or Jacobian

$ds^2 = dx^2 + dy^2$ . Since both  $x$  and  $y$  are defined in terms of  $r$ , we can extend this identity to the needed quantity

$$\left(\frac{ds}{dr}\right)^2 = \left(\frac{dx}{dr}\right)^2 + \left(\frac{dy}{dr}\right)^2$$

where  $dx/dr$  can be found from the spatial interpolation functions, etc. for  $dy/dr$ . Thus, our physical length is defined in terms of the parametric coordinate,  $r$ , and the spatial data for the nodes defining the curve location in the  $xy$ -plane (i.e.,  $x^e$  and  $y^e$ ):

$$L = \int_0^1 \sqrt{\left(\frac{dx}{dr}\right)^2 + \left(\frac{dy}{dr}\right)^2} dr$$

where

$$\frac{dx(r)}{dr} = \sum_{i=1}^{n_b} \frac{dH_i(r)}{dr} x_i^e, \quad \frac{dy(r)}{dr} = \sum_{i=1}^{n_b} \frac{dH_i(r)}{dr} y_i^e.$$

Note that this does preserve the proper units for  $L$ , and it has what we could refer to as a variable Jacobian. The preceding integral is trivial only when the planar curve is a straight line ( $n_b = 2$ ). Then, from linear geometric interpolation  $dx/dr = x_2^e - x_1^e$  and  $dy/dr = y_2^e - y_1^e$  are both constant, and the result simplifies to

$$L^e = \sqrt{\left(x_2^e - x_1^e\right)^2 + \left(y_2^e - y_1^e\right)^2} \int_0^1 dr = L$$

which, by inspection, is exact. For any other curve shape, these integrals become unpleasant to evaluate, and we would consider their automation by means of numerical integration.

To complete this section we outline an algorithm to automate the calculation of the  $y$ -centroid:

1. Recover the  $n_b$  points describing the curve,  $\mathbf{x}^e$ ,  $\mathbf{y}^e$ .
2. Recover the  $n_q$ -point quadrature data,  $r_j$  and  $w_j$ .
3. Zero the integrals:  $L = 0$ ,  $\bar{y} L = 0$ .
4. Loop over all the quadrature points:  $1 \leq j \leq n_q$ . At each local quadrature point,  $t_j$ :
  - A. Find the length scales (i.e., Jacobian):

- (i) Compute local derivatives of the  $n_b$  interpolation functions

$$\mathbf{DL\_H}_j \equiv \left. \frac{\partial \mathbf{H}}{\partial \mathbf{r}} \right|_{r=r_j}$$

- (ii) Get  $x$ - and  $y$ -derivatives from curve data

$$\frac{dx}{dr_j} \equiv \mathbf{DL\_H}_j \mathbf{x}^e = \sum_{i=1}^{n_b} \frac{\partial H_i(r_j)}{\partial r} x_i^e, \quad \frac{dy}{dr_j} \equiv \mathbf{DL\_H}_j \mathbf{y}^e$$

- (iii) Find length scale at point  $r_j$

$$\frac{ds}{dr_j} = \sqrt{\left( \frac{dx}{dr} \right)_j^2 + \left( \frac{dy}{dr} \right)_j^2}$$

- B. Evaluate the integrand at point  $r_j$ :
  - (i) Evaluate the  $n_b$  interpolation functions:

$$\mathbf{H}_j \equiv \mathbf{H}(r_j)$$

- (ii) Evaluate  $y$  from curve data

$$y_j = \mathbf{H}_j \mathbf{y}^e = \sum_{i=1}^N H_i(r_j) y_i^e$$

- C. Form products and add to previous values

$$L = L + \frac{ds}{dr_j} w_j, \quad \bar{y} L = \bar{y} L + y_j \frac{ds}{dr_j} w_j.$$

4. Evaluate items computed from the completed integrals:  $\bar{y} = \bar{y} L / L$ , etc. for  $\bar{x}$ .

Note that to automate the integration we simply need: 1) storage of the quadrature data,  $r_j$  and  $w_j$ , 2) access to the curve data,  $\mathbf{x}^e$ ,  $\mathbf{y}^e$ , 3) a subroutine to find the parametric derivative of the interpolation functions at any point,  $r$ , and 4) a function program to evaluate the integrand(s) at any point,  $r$ . This usually requires (see Step 4B) a subroutine to evaluate the interpolation functions at any point,  $r$ . The evaluations of the interpolation products, at a point  $r_j$ , can be thought of as a dot product of the data array  $\mathbf{x}^e$  and the evaluated interpolation quantities  $\mathbf{H}_j$  and  $\mathbf{DL\_H}_j$ .

## 4.6 Exercises

1. In Tables 4.1 and 4.3 the sum of the weights is exactly 2, but in Table 4.2 the sum is exactly 1. Explain why.
2. Note that the Gauss abscissas are always interior to the domain and symmetrically placed. The Lobatto rules for  $n_q \geq 2$  always includes the two end points. Discuss an advantage or disadvantage of including the end points.
3. For a one-dimensional quadratic element use Gaussian quadratures to numerically evaluate the matrices:

$$a) \mathbf{C}^e = \int_L^e \mathbf{H}^T dx, \quad b) \mathbf{M}^e = \int_L^e \mathbf{H}^T \mathbf{H} dx,$$

$$c) \mathbf{S}^e = \int_L^e \frac{d\mathbf{H}^T}{dx} \frac{d\mathbf{H}}{dx} dx, \quad d) \mathbf{U}^e = \int_L^e \mathbf{H}^T \frac{d\mathbf{H}}{dx} dx.$$

4. Use the exact integrals for  $\mathbf{S}^e$  and  $\mathbf{M}^e$  from Problem 3 to hard code the equivalent of Fig. 2.10.2 for a three node quadratic line element. Note that such an element will also allow for  $Q(x) = x^2$  as well as the original linear source term. In that case, of course, the exact solution changes.
5. Use Lobatto quadrature to evaluate the matrices in Problem 3. Note that  $\mathbf{M}^e$  becomes a diagonal matrix.

## 4.7 Bibliography

- [1] Abramowitz, M. and Stegun, I.A., *Handbook of Mathematical Functions*, National Bureau of Standards (1964).
- [2] Carey, G.F. and Oden, J.T., *Finite Elements – Computational Aspects*, Englewood Cliffs: Prentice-Hall (1984).
- [3] Hinton, E. and Owen, D.R.J., *Finite Element Programming*, London: Academic Press (1977).
- [4] Hughes, T.J.R., *The Finite Element Method*, Englewood Cliffs: Prentice-Hall (1987).
- [5] Smith, I.M. and Griffiths, D.V., *Programming the Finite Element Method*, 3rd Edition, Chichester: John Wiley (1998).
- [6] Stroud, A.H. and Secrest, D., *Gaussian Quadrature Formulas*, Englewood Cliffs: Prentice-Hall (1966).
- [7] Zienkiewicz, O.C. and Taylor, R.L., *The Finite Element Method*, 5th Edition, London: Butterworth-Heinemann (2000).