2.9 Generalization of finite element procedures for linear elasticity

1. Basic finite element method

   As discussed in Section 2.7, after eliminating prescribed displacements in finite element equations (Eq. 2.34), the solution of global displacement vector $\boldsymbol{u}$ can be solved by the following code:

```
.
.
[ndime,nnode,nelem,nelnd,npres,ntrac,mate,coor,conn,pres,trac] = ReadInput(infile);
kglob = GlobStif(ndime,nnode,nelem,nelnd,mate,coor,conn);
rglob = GlobTrac(ndime,nnode,nelem,nelnd,ntrac,mate,coor,conn,trac);
kpres= kglob;
rpres= rglob;
for i = 1:npres
  idof = ndime*(pres(1,i)-1)+pres(2,i);
  for ir = 1:ndime*nnode
    kpres(ir,idof) = 0;
    rpres(ir) = rpres(ir) - kglob(ir,idof)*pres(3,i);
  end
end
for i = 1:npres
  idof = ndime*(pres(1,i)-1)+pres(2,i);
  kpres(idof,:) = 0;
  kpres(idof,idof) = 1;
  rpres(idof) = pres(3,i);
end
uglob = kpres\rpres;
.
.
```

   Then, the postprocessing discussed in Section 2.1.6.13 needs to be done for obtaining the strain and the stress distributios in each element. In this Section, the FEM which can solve problems involving complex materials and large shape changes will be introduced.

2. Review of linear elasticity and Hooke's law

   First, the usual governing equations of linear elasticity, which must be solved by the FEA code, are summarized as:

   (1) The strain-displacement equation

$$\text{(2.35)}\quad \varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

   (2) The elastic stress-strain law

$$\text{(2.36)}\quad \sigma_{ij} = C_{ijk\ell}\,\varepsilon_{k\ell}$$

   (3) The equation of static equilibrium for stresses

$$\text{(2.37)}\quad \frac{\partial \sigma_{ij}}{\partial x_i} + b_j = 0$$

   (4) The boundary conditions on displacement and stress

$$\text{(2.38)}\quad u_i = u_i^* \text{ on } \partial_1 R$$
$$\text{(2.39)}\quad \sigma_{ij} n_i = t_j^* \text{ on } \partial_2 R$$

   All notations are Identical to the ones commonly used in the *theory of elasticity*. Here, brief review of the generalized Hooke's law defining the most general linear relation among all the components of the stress and strain tensor is discussed below:

$$\text{(2.40)}\quad \sigma_{ij} = C_{ijk\ell}\,\varepsilon_{k\ell}$$

In this expression, $C_{ijk\ell}$ are the components of the fourth-order stiffness tensor of material properties or elastic moduli. The fourth-order stiffness tensor has 81 and 16 components for three-dimensional and two-dimensional problems, respectively. The strain energy density is a quadratic function of the strain:

$$U = \frac{1}{2} C_{ijk\ell}\, \varepsilon_{ij}\, \varepsilon_{k\ell} \tag{2.41}$$

The stiffness tensor has the following *minor symmetries* which result from the symmetry of the stress and strain tensors, so the number of material constants to 36 and 9 for three-dimensional and two-dimensional problems, respectively as:

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma 33 \\ \sigma 23 \\ \sigma 13 \\ \sigma 12 \end{bmatrix} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1123} & C_{1113} & C_{1112} \\ & C_{2222} & C_{2233} & C_{2223} & C_{2213} & C_{2212} \\ & & C_{3333} & C_{3323} & C_{3313} & C_{3312} \\ & & & C_{2323} & C_{2313} & C_{2312} \\ & \text{symm.} & & & C_{1313} & C_{1312} \\ & & & & & C_{1212} \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{23} \\ 2\varepsilon_{13} \\ 2\varepsilon_{12} \end{bmatrix} \tag{2.42}$$

With Voigt notation as $()_{11} \to ()_1$, $()_{22} \to ()_2$, $()_{33} \to ()_3$, $()_{23} \to ()_4$, $()_{13} \to ()_5$, $()_{12} \to ()_6$ and defining engineering shear strains as the sum of symmetric components, e.g., $\varepsilon_4 = 2\varepsilon_{23} = \varepsilon_{23} + \varepsilon_{32}$, etc., the generalized Hooke's law can be made more sense for matrix operation:

$$\begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma 3 \\ \sigma 4 \\ \sigma 5 \\ \sigma 6 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ & & C_{33} & C_{34} & C_{35} & C_{36} \\ & & & C_{44} & C_{45} & C_{46} \\ & \text{symm.} & & & C_{55} & C_{56} \\ & & & & & C_{66} \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{bmatrix} \tag{2.43}$$

Let' s take a brief look at various classes of material symmetry.

(1) Orthotropic linear elastic materials (Number of independent coefficients: 9)

$$\boldsymbol{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ & C_{22} & C_{23} & 0 & 0 & 0 \\ & & C_{33} & 0 & 0 & 0 \\ & & & C_{44} & 0 & 0 \\ & \text{symm.} & & & C_{55} & 0 \\ & & & & & C_{66} \end{bmatrix} \tag{2.44}$$

(2) Transversely isotropic linear elastic materials (Number of independent coefficients: 5)

$$\boldsymbol{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ & C_{11} & C_{13} & 0 & 0 & 0 \\ & & C_{33} & 0 & 0 & 0 \\ & & & C_{44} & 0 & 0 \\ & \text{symm.} & & & C_{44} & 0 \\ & & & & & \frac{1}{2}(C_{11} - C_{12}) \end{bmatrix} \tag{2.45}$$

(3) Isotropic linear elastic materials (Number of independent coefficients: 2)

$$\boldsymbol{C} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ & & \lambda + 2\mu & 0 & 0 & 0 \\ & & & \mu & 0 & 0 \\ & \text{symm.} & & & \mu & 0 \\ & & & & & \mu \end{bmatrix} \tag{2.46}$$

where $\lambda = \dfrac{E\,\nu}{(1+\nu)(1-2\nu)}$ and $\mu = G = \dfrac{E}{2(1+\nu)}$

3. The principle of virtual work

The principle of virtual work can be used to replace the stress equilibrium equations. To express the principle, we define a kinematically admissible virtual velocity field $\delta \boldsymbol{v}(\boldsymbol{x})$, satisfying $\delta \boldsymbol{v} = 0$ on $\partial_1 R$. We can visualize this virtual field as a small

change in the displacement of the solid, but it is really just an arbitrary differentiable vector field. Thus, the associated virtual strain field:

$$\delta\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial \delta v_i}{\partial x_j} + \frac{\partial \delta v_j}{\partial x_i}\right) \tag{2.47}$$

The principle of virtual work states that, if the stress field σ_{ij} satisfies:

$$\int_R \sigma_{ij}\delta\varepsilon_{ij}\, dV - \int_R b_i \delta v_i\, dV - \int_{\partial_2 R} t_i^* \delta v_i\, dA = 0 \tag{2.48}$$

for all possible virtual velocity fields and corresponding virtual strains, it will automatically satisfy the equation of stress equilibrium $\partial\sigma_{ij}/\partial x_i + b_j = 0$ and also the traction boundary condition σ n $\sigma_{ij} n_i = t_j^*$ on $\partial_2 R$.

4. Weak form of the governing equations of linear elasticity

The *principle of virtual work* can be used to write the governing equation for the displacement field in a linear elastic solid in an integral form (called the "weak form"). Instead of solving the governing equations directly, the displacements, strains, and stresses are calculated as follows:

(1) Find a displacement field $u_i(x_j)$ satisfying

$$\int_R C_{ijk\ell}\frac{\partial u_k}{\partial x_\ell}\frac{\partial \delta v_i}{\partial x_j} dV - \int_R b_i \delta v_i\, dV - \int_{\partial_2 R} t_i^* \delta v_i\, dA = 0, \quad u_i = u_i^* \text{ on } \partial_1 R \tag{2.49}$$

for all virtual velocity fields δv_i satisfying $\delta v_i = 0$ on $\partial_1 R$.

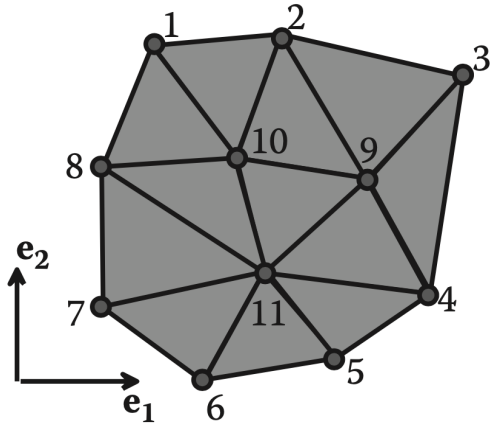
(2) Compute the strains from the definition $\varepsilon_{ij} = \frac{1}{2}(\partial u_i/\partial x_j + \partial u_j/\partial x_i)$

(3) Compute the stresses from the stress-strain law $\sigma_{ij} = C_{ijk\ell}\varepsilon_{k\ell}$. The stress will automatically satisfy the equilibrium equation and boundary conditions, so all the field equations and boundary conditions will be satisfied.

The significance of this result is that it replaces the derivatives in the partial differential equations of equilibrium with an equivalent integral, which is easier to handle numerically. It is essentially equivalent to replacing the equilibrium equation with the principle of minimum potential energy, but the procedure based on the principle of virtual work is very easily extended to dynamic problems, other stress-strain laws, and even to problems involving large shape changes.

5. Interpolating the displacement field and the virtual velocity field

To solve the integral form of the elasticity equations given above, we discretize the displacement field. A representative finite element mesh is sketched in the figure. We choose to calculate the displacement field at a set of n discrete points in the solid (called "nodes" in finite element terminology). We will denote the coordinates of these special points by x_i^a, where the superscript a ranges from 1 to n. The unknown displacement vector at each nodal point will be denoted by u_i^a.



The displacement field at an arbitrary point within the solid will be specified by interpolating between nodal values in some convenient way. An efficient and robust implementation of the FEM requires the interpolation in a general way as:

$$u_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x})\, u_i^a \tag{2.50}$$

Here, $\boldsymbol{x}$ denotes the coordinates of an arbitrary point in the solid. The interpolation functions $N^a(\boldsymbol{x})$ are functions of position only, which must have the property that:

$$u_i^b = \sum_{a=1}^{n} N^a(\boldsymbol{x}^b)\, u_i^a \tag{2.51}$$

for all $b = 1 \ldots n$. This is to ensure that the displacement field has the correct value at each node as:

$$N^a(x^b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{2.52}$$

The simple constant strain triangle elements introduced in Section 2.1 are one example of interpolation schemes. We will define more complicated interpolation functions shortly. Furthermore, the virtual velocity field can be obviously interpolated in exactly the same way:

$$\delta v_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x})\, \delta v_i^a \tag{2.53}$$

where δv_i^a are arbitrary nodal values of a virtual velocity field.

2.10 Finite element equations

1. By substituting the interpolated unknown displacement and virtual velocity fields into the virtual work equation, we find that:

$$\int_R C_{ijk\ell} \frac{\partial N^b(\boldsymbol{x})}{\partial x_\ell} u_k^b \frac{\partial N^a(\boldsymbol{x})}{\partial x_j} \delta v_i^a \, dV - \int_R b_i N^a(\boldsymbol{x}) \delta v_i^a \, dV - \int_{\partial_2 R} t_i^* N^a(\boldsymbol{x}) \delta v_i^a \, dA = 0 \tag{2.54}$$

where summation on a and b is implied, in addition to the usual summation on i, j, k, and ℓ. Noting that the interpolation functions are known functions of position, the virtual work equation in matrix form can be formed as:

$$\left(K_{aibk}\, u_k^b - F_i^a\right)\, \delta v_i^a = 0 \tag{2.55}$$

where

$$K_{aibk} = \int_R C_{ijk\ell} \frac{\partial N^a(\boldsymbol{x})}{\partial x_j} \frac{\partial N^b(\boldsymbol{x})}{\partial x_\ell} dV \tag{2.56}$$

$$F_i^a = \int_R b_i N^a(\boldsymbol{x})\, dV + \int_{\partial_2 R} t_i^* N^a(\boldsymbol{x})\, dA \tag{2.57}$$

Here, $\boldsymbol{K}$ is known as the stiffness matrix and $\boldsymbol{F}$ is known as the force vector. $\boldsymbol{K}$ is a function only of the elastic properties of the solid, its geometry, and the interpolation functions and nodal positions. It is therefore a known matrix. Similarly, $\boldsymbol{F}$ is a function only of the known boundary loading and body force field, and the interpolation scheme and nodal positions. Observe that the symmetry of the elasticity tensor implies that K also has some symmetry, i.e., $K_{aibk} = K_{bkai}$.

2. The virtual work equation must be satisfied for all possible sets of δv_i^a with $\delta v_i^a = 0$ for nodes a which lie on $\partial_1 R$, i.e.,

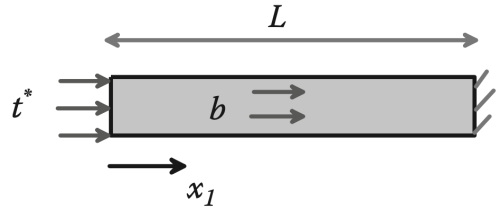
$$K_{aibk} u_k^b = F_i^a \qquad \forall \{a, i\} : \; x_k^a \text{ not on } \partial_1 R \tag{2.58}$$

$$u_i^a = u_i^*(x_i^a) \qquad \forall \{a, i\} : \; x_k^a \text{ on } \partial_1 R \tag{2.59}$$

Again, this is a system of linear equations of the nodal displacements u_k^b, where $b = 1, \ldots, n$, $k = 1, 2$ for 2D and $k = 1, 2, 3$ for 3D problems.

2.11 Generalization of finite element procedures for linear elasticity

1. Before describing a fully general 2D and 3D implementation of the FEM, we will illustrate all the key ideas using a simple 1D example. Consider a long linear elastic bar in the stress state of plane strain, as shown in the figure. Assume the following:
   (1) The bar has shear modulus μ and Poisson's ratio ν.
   (2) The bar has cross section $h \times h$ and length L.
   (3) It is constrained on all its sides so that $u_2 = u_3 = 0$.
   (4) The bar is subjected to body force $\boldsymbol{b} = b(x_1)\, \boldsymbol{e}_1$.
   (5) The bar is either loaded or constrained at its ends, so that the boundary conditions are either $t_1(0) = t^*(0)$, $t_1(L) = t^*(L)$ or displacement $u_1(0) = u^*(0)$, $u_1(L) = u^*(L)$ at $x = 0$ and $x = L$.



2. For the 1D example, the finite element equations reduce to:

$$K_{a1b1}u_1^b = F_1^a \tag{2.60}$$

where

$$K_{a1b1} = \int_R C_{1111}\frac{\partial N^a(x_1)}{\partial x_1}\frac{\partial N^b(x_1)}{\partial x_1}dV = h^2\int_0^L C_{1111}\frac{\partial N^a(x_1)}{\partial x_1}\frac{\partial N^b(x_1)}{\partial x_1}dx_1 \tag{2.61}$$

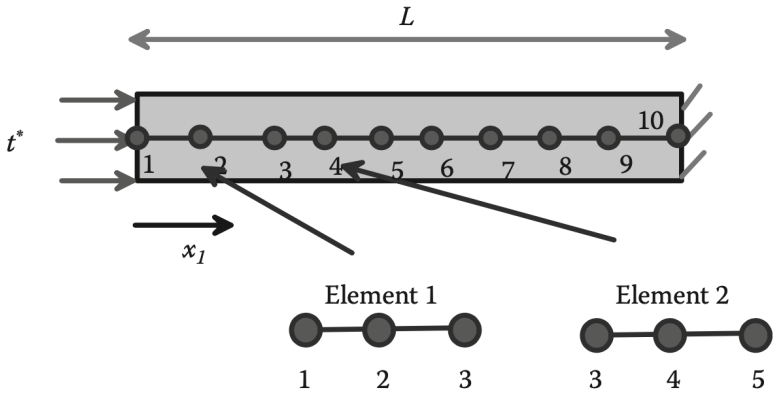
$$F^a = \int_0^L h^2 b(x_1)N^a(x_1)dx_1 + h^2 t^*(0)N^a(0) + h^2 t^*(L)N^a(L)) \tag{2.62}$$

$$C_{1111} = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} = \frac{2\mu(1-\nu)}{(1-2\nu)} \tag{2.63}$$

We could obviously choose any interpolation scheme, evaluate the necessary integrals, and solve the resulting system of equations to compute the solution. It turns out to be particularly convenient to use a *piecewise Lagrangian interpolation scheme* and to evaluate the integrals numerically using a *Gaussian quadrature scheme*.

3. Lagrangian interpolation scheme

(1) To implement the Lagrangian interpolation scheme, we subdivide the region $0 \le x_1 \le L$ into a series of elements, as illustrated in the figure. Each element is bounded by two nodal points. The displacement field within the element is interpolated between the nodes attached to the element. So, we would use a linear interpolation between the nodes on a two-noded element. Such two-noded element may also contain one or more interior nodes, and a quadratic interpolation has to be applied between the nodes on a three-noded element, and so on.



(2) Generic linear and quadrilateral 1D elements are illustrated in the following table. The local nodes on the element are numbered 1 and 2 for the *linear element* and 1, 2, and 3 for the *quadratic element* as shown. We suppose that the element lies in the region $-1 \le \xi_1 \le 1$. The displacements within the element are then interpolated as:

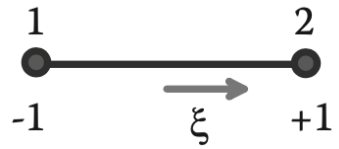
$$u_1(\xi_1) = \sum_{a=1}^{N_e} N^a(\xi_1)\, u_1^a \tag{2.64}$$

where u_1^a denotes the value of the displacement at each node, N_e is the number of nodes on the element, and the shape functions $N^a(\xi_1)$ are given in the table.

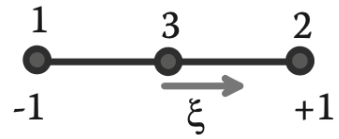
(3) Of course, the actual nodal coordinates do not lie at -1, 0, and $+1$ for all the elements. For a general element, we *map* this special one to the region of interest. A particularly convenient way to do this is to set:

$$x_1(\xi_1) = \sum_{a=1}^{N_e} N^a(\xi_1)x_1^a \tag{2.65}$$

where x_1^a denotes the coordinates of each node on the element, and N_e is the number of nodes on the element (2 or 3). Elements that interpolate displacements and position using the same shape functions are called *isoparametric elements*.



Linear 1D Element	
$N^1(\xi) = 0.5\,(1-\xi)$	$N^2(\xi) = 0.5\,(1+\xi)$



Quadratic 1D Element		
$N^1(\xi) = -0.5\,\xi\,(1-\xi)$	$N^2(\xi) = 0.5\,\xi\,(1+\xi)$	$N^3(\xi) = (1-\xi)(1+\xi)$

4. Gaussian quadrature scheme

(1) Next, we need to devise a way to do the integrals in the expressions for the stiffness matrix and force vector (Eqs. (2.61) and (2.62)). We can evidently divide up the integral so as to integrate over each element in turn:

$$\text{(2.66)}\quad K_{ab} = \sum_{\ell=1}^{N_m} h^2 \int_{x_0}^{x_1} \frac{2\mu(1-\nu)}{1-2\nu} \frac{\partial N^a(x_1)}{\partial x_1} \frac{\partial N^b(x_1)}{\partial x_1}\, dx_1$$

$$\text{(2.67)}\quad F^a = \sum_{\ell=1}^{N_m} h^2 \int_{x_0}^{x_1} b\, N^a(x_1) dx_1 + h^2 t^*(0) N^a(0) + h^2 t^*(L) N^a(L))$$

where N_m is the total number of elements, and x_0 and x_1 denote the coordinates of the ends of the ℓ-th element. We now notice an attractive feature of our interpolation scheme. The integral over the ℓ-th element depends only on the shape functions associated with the nodes on the ℓ-th element, because the displacement in this region is completely determined by its values at these nodes. We can therefore define element stiffness matrix and element force matrix:

$$\text{(2.68)}\quad k_{ab} = h^2 \int_{x_0}^{x_1} \frac{2\mu(1-\nu)}{1-2\nu} \frac{\partial N^a(x_1)}{\partial x_1} \frac{\partial N^b(x_1)}{\partial x_1}\, dx_1$$

$$\text{(2.69)}\quad f^a = h^2 \int_{x_0}^{x_1} b\, N^a(x_1)\, dx_1$$

for each element, which depend on the the geometry, interpolation functions, and material properties of the element. The first and last elements have additional contributions to the element force vector from the boundary terms $h^2 t^*(0) N^a(0)$ and $h^2 t^*(L) N^a(L))$. The global stiffness matrix is thus computed by summing all the element stiffness matrices:

$$\text{(2.70)}\quad K_{ab} = \sum_{\ell=1}^{N_m} k_{ab}$$

$$\text{(2.71)}\quad F^a = \sum_{\ell=1}^{N_m} f_a + h^2 t^*(0) N^a(0) + h^2 t^*(L) N^a(L)$$

(2) Moreover, we need to devise a way to compute the integrals for each element stiffness matrix. It is convenient to *map* the domain of integration to $[-1, +1]$ and integrate with respect to the normalized coordinate ξ; thus,

$$\text{(2.72)}\quad k_{ab} = h^2 \int_{-1}^{+1} \frac{2\mu(1-\nu)}{1-2\nu} \frac{\partial N^a(x_1)}{\partial x_1} \frac{\partial N^b(x_1)}{\partial x_1}\, J\, d\xi$$

$$\text{(2.73)}\quad f^a = h^2 \int_{-1}^{+1} b\, N^a(x_1)\, J\, d\xi$$

where $J = |\partial x / \partial \xi|$ is the Jacobian associated with tehe mapping when expressing x_1 as x for simplicity. Furthermore, the contents of the Jacobian can be computed as:

$$\frac{\partial x}{\partial \xi} = \frac{\partial}{\partial \xi}\sum_{a=1}^{N_e} N^a(\xi) x^a = \sum_{a=1}^{N_e} \frac{\partial N^a}{\partial \xi} x^a \tag{2.74}$$

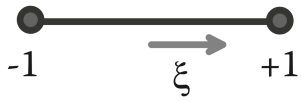
(3) Note that the mapping also enables us to calculate the shape function derivatives in the element stiffness matrix as:

$$\frac{\partial N^a}{\partial x} = \frac{\partial N^a}{\partial \xi}\frac{\partial \xi}{\partial x} = \frac{\partial N^a}{\partial \xi}\left[\frac{\partial x}{\partial \xi}\right]^{-1} \tag{2.75}$$

Finally, the integrals may be computed numerically using a *quadrature formula*, as follows:

$$\int_{-1}^{+1} g(\xi)\, d\xi = \sum_{I=1}^{M} w_I\, g(\xi^{(I)}) \tag{2.76}$$

where $\xi^{(I)}$, $I = 1, 2, ...M$ denotes a set of *integration points* in the region $[-1, +1]$, and w_I is a set of integration weights, which are chosen so as to make the approximation as accurate as possible. Values are given in the following tablefor $M = 1, 2,$ and 3. Higher-order integration schemes exist but are required only for higher-order elements. For the linear 1D element described previously, a single integration point is sufficient to evaluate the stiffness exactly. Similarly, for the quadratic element, two integration points will suffice.



$M = 1$	
$\xi^{(1)} = 0$	$w_1 = 2$

$M = 2$	
$\xi^{(1)} = -0.5773502691$	$w_1 = 1.0$
$\xi^{(2)} = 0.5773502691$	$w_2 = 1.0$

$M = 3$	
$\xi^{(1)} = -0.7745966692$	$w_1 = 0.55555555555$
$\xi^{(2)} = 0$	$w_2 = 0.88888888888$
$\xi^{(3)} = 0.7745966692$	$w_3 = 0.55555555555$

5. Summary of the 1D Finite Element Procedure

(1) For each element, compute the element stiffness matrix as follows:

$$k_{ab} = h^2 \sum_{I=1}^{M} w_I \frac{2\mu(1-\nu)}{1-2\nu}\frac{\partial N^a}{\partial x}\frac{\partial N^b}{\partial x} J(\xi_I)$$

where

$$J = \left|\frac{\partial x}{\partial \xi}\right| = \left|\sum_{a=1}^{N_e} \frac{\partial N^a(\xi_I)}{\partial \xi} x^a\right|$$

$$\frac{\partial N^a}{\partial x} = \frac{\partial N^a(\xi_I)}{\partial \xi}\frac{\partial \xi}{\partial x} = \frac{\partial N^a(\xi_I)}{\partial \xi}\left[\frac{\partial x}{\partial \xi}\right]^{-1}$$

The integration points ξ_I, the corresponding weights w_I ($I = 1, 2, ...M$) and the shape functions $N^a(\xi)$ ($a = 1, 2, ...N_e$) are tabulated above.

(2) Assemble the contribution from each element to the global stiffness:

$$K_{ab} = \sum_{\ell=1}^{N_m} k_{ab}$$

(3) Similarly, if there is a body force, then compute for each element:

$$f^a = h^2 \sum_{I=1}^{M} w_I \, b \, N^a(\xi_I) \, J(\xi_I)$$

and assemble the global force vector:

$$F^a = \sum_{\ell=1}^{N_m} f_a$$

(4) Add contributions to the force vector from prescribed traction boundary conditions at $x = 0$ and $x = L$.

$$F^{(1)} = F^{(1)} + h^2 \, t^*(0)$$
$$F^{(L)} = F^{(L)} + h^2 \, t^*(L)$$

where the $^{(1)}$ and $^{(L)}$ denote the nodes which lies at $x = 0$ and $x = L$ respectively.

(5) Modify the stiffness matrix to enforce the displacement constraints:

$$u^{(1)} = u^*(0)$$
$$u^{(L)} = u^*(L)$$

(6) Solve the system of linear equations:

$$K_{ab} \, u_1^b = F^a$$

for the unknown displacement u_1^b.