FINITE ELEMENT FORMULATIONS

Before you decide to drop the course and curse me for convincing or making you take this class, let me tell you that up till now we have not really discussed finite element methods.

The basic mathematics we have previously discussed provides us with a basis for understanding the inner workings of FEM techniques. The two basic approaches are:

1. The **Ritz** method - utilizes the energy or functional associated with the differential equation as the base for the finite element formulation
2. The **MWR** method - approximates the differential equation directly as the base for the finite element formulation

FINITE ELEMENT FORMULATIONS

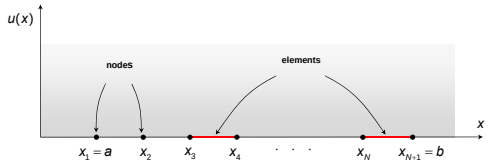
Variational Finite Element Models

The steps involved in generating a FEM model using variational techniques follows the same procedure as we used in our discussion of the spring-mass system:

1. Discretization
2. Interpolation
3. Elemental Description or Formulation
4. Assembly
5. Constraints
6. Solution
7. Computation of Derived Variables

FINITE ELEMENT FORMULATIONS

Discretization - The domain is broken up into a series of subintervals:



The points that separate the subintervals are called **nodes** and the subdomains between the nodes are called **elements**.

The entire discretization is often called a **mesh**. Later we will discuss how the spacing of the mesh may effect the solution.

FINITE ELEMENT FORMULATIONS

Interpolation - At this point we have to decide what type of interpolation or approximation we want over each element.

In past discussions of the axial deformation problem we routinely used a **linear element.**

An element of this type assumes an linear variation of the function over the element.

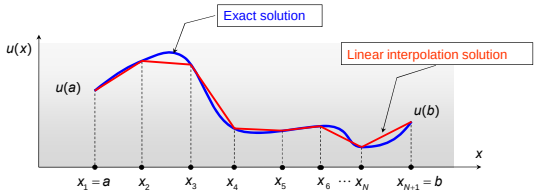
Therefore the function is represented as a series of linear lines that are continuous at the nodes.

One obvious problem with a linear interpolation is that the derivative of the approximation is discontinuous at each node.

FINITE ELEMENT FORMULATIONS

Interpolation - At this point we have to decide what type of interpolation or approximation we want over each element.

If for example, we are interested in information based on the derivative at the nodes, then a linear interpolation will not be appropriate for the problem.



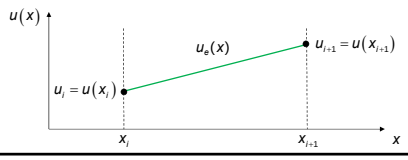
FINITE ELEMENT FORMULATIONS

The variation of the unknown function over an element may be written in global coordinates as:

$$u_e = c_1 + c_2 x$$

Matching the values of the function at the endnodes of each element require u_e to be:

$$u_e = c_1 + c_2 x_i = u_i \qquad u_e = c_1 + c_2 x_{i+1} = u_{i+1}$$



FINITE ELEMENT FORMULATIONS

Solving for c_1 and c_2 results in the following equation:

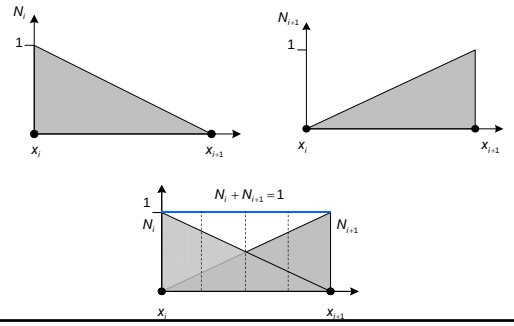
$$u_e = u_i\left(\frac{x_{i+1}-x}{x_{i+1}-x_i}\right)+u_{i+1}\left(\frac{x-x_i}{x_{i+1}-x_i}\right)$$

$$u_e = N_i u_i + N_{i+1}u_{i+1}$$

where: $N_i = \dfrac{x_{i+1}-x}{x_{i+1}-x_i}$ $\qquad$ $N_{i+1} = \dfrac{x-x_i}{x_{i+1}-x_i}$

FINITE ELEMENT FORMULATIONS

$$N_i = \frac{x_{i+1}-x}{x_{i+1}-x_i} \qquad N_{i+1} = \frac{x-x_i}{x_{i+1}-x_i}$$



FINITE ELEMENT FORMULATIONS

The derivative of *u* may be computed as:

$$u_e' = u_i\left(\frac{-1}{x_{i+1}-x_i}\right)+u_{i+1}\left(\frac{1}{x_{i+1}-x_i}\right)$$

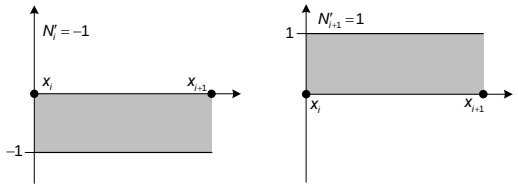
$$u_e' = N_i' u_i + N_{i+1}' u_{i+1}$$

where:

$$N'_i = -\frac{1}{x_{i+1}-x_i} \qquad N'_{i+1} = \frac{1}{x_{i+1}-x_i}$$

FINITE ELEMENT FORMULATIONS

$$N'_i = -\frac{1}{x_{i+1}-x_i} \qquad N'_{i+1} = \frac{1}{x_{i+1}-x_i}$$



FINITE ELEMENT FORMULATIONS

In matrix form we may write the variation of *u* and *u'* over an element as:

$$u_e = \mathbf{N}^{\mathrm{T}}\mathbf{u}_e \qquad u_e' = \mathbf{N'}^{\mathrm{T}}\mathbf{u}_e$$

where the vectors **N**, **N'**, and $\mathbf{u_e}$ are:

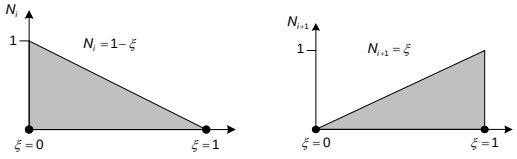
$$\mathbf{N} = \begin{Bmatrix} N_i \\ N_{i+1} \end{Bmatrix} \qquad \mathbf{N'} = \begin{Bmatrix} N_i' \\ N_{i+1}' \end{Bmatrix} \qquad \mathbf{u_e} = \begin{Bmatrix} u_i \\ u_{i+1} \end{Bmatrix}$$

The vector **N** is often called the **elemental interpolation vector** and $\mathbf{u_e}$ is called the **elemental displacement vector**.

FINITE ELEMENT FORMULATIONS

It is often more convenient to express the interpolation functions in terms of an elemental coordinate system ξ.

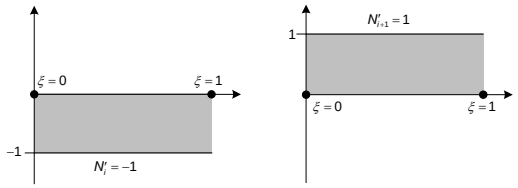
For example, the linear interpolation functions may be rewritten in ξ as:



FINITE ELEMENT FORMULATIONS

These elemental interpolation functions are often called **shape functions** because they determine the shape of the unknown function over each element.

The derivative shape functions are:



FINITE ELEMENT FORMULATIONS

Element Formulation - Let's first look at a variational approach to the Sturm-Liouville problem.

Recall the energy functional is:

$$Z(u) = \int_a^b \left[\frac{p(u')^2 - qu^2}{2} - uf \right] dx + \frac{\alpha u(a)^2}{2} + \frac{\beta u(b)^2}{2} + Au(a) + Bu(b)$$

Using the concept of the **element** as we have discussed previously, we may approximate the functional by replacing the continuous function *u* with a series of connected elements.

FINITE ELEMENT FORMULATIONS

Element Formulation - Therefore the energy functional may be approximated by:

$$Z(u) = \sum_e \int_{x_i}^{x_{i+1}} \left[\frac{p(u')^2 - qu^2}{2} - uf \right] dx + \frac{\alpha u(a)^2}{2} + \frac{\beta u(b)^2}{2} + Au(a) + Bu(b)$$

This type of approximation is possible since one of the fundamental properties of an integral is:

$$\int_a^b dx = \int_a^{x_1} dx + \int_{x_1}^{x_2} dx + \int_{x_2}^{x_3} dx + \ldots + \int_{x_{N-1}}^{x_N} dx + \int_{x_N}^b dx$$

FINITE ELEMENT FORMULATIONS

Element Formulation - The approximation of the energy functional may be written in the following form:

$$Z(u) = \sum_e \left(\frac{Z_{pe} + Z_{qe}}{2} - Z_{fe} \right) + \frac{\alpha u_1^2}{2} + \frac{\beta u_{N+1}^2}{2} - Au_1 - Bu_{N+1}$$

where the integrals Z_{pe}, Z_{qe}, and Z_{fe} are defined as:

$$Z_{pe} = \int_{x_i}^{x_{i+1}} u' p(x) u' \, dx \qquad Z_{qe} = \int_{x_i}^{x_{i+1}} uq(x) u \, dx$$

$$Z_{fe} = \int_{x_i}^{x_{i+1}} uf(x) \, dx$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Each of these integrals is evaluated over each element.

To transform the above integral into element coordinates ξ the differential operator *dx* must be replaced by the appropriate transformation between *x* and ξ.

The global coordinate *x* may be written as:

$$x_e = \mathbf{N}^T \mathbf{x_e} = \langle N_i \quad N_{i+1} \rangle \begin{Bmatrix} x_i \\ x_{i+1} \end{Bmatrix}$$

$$= (1-\xi) x_i + \xi x_{i+1} = x_i + \xi (x_{i+1} - x_i)$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Differentiating *x* with respect to ξ gives:

$$dx = \frac{dx}{d\xi} d\xi = \frac{d}{d\xi} \left[(1-\xi) x_i + \xi x_{i+1} \right] d\xi = (x_{i+1} - x_i) d\xi$$

$dx = l_e d\xi$ $\qquad l_e =$ the length of the element

$$x_e = x_i + \xi (x_{i+1} - x_i) = x_i + \xi l_e \qquad \Rightarrow \xi = \frac{x_e - x_i}{l_e}$$

$$\frac{du}{dx} = \frac{du}{d\xi} \frac{d\xi}{dx} = \frac{du}{d\xi} \frac{d}{dx} \left(\frac{x - x_i}{l_e} \right) = \frac{du}{d\xi} \left(\frac{1}{l_e} \right)$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Substituting the coordinate transformation for x in the integrals Z_{pe}, Z_{qe}, and Z_{fe} result in:

$$Z_{pe} = \int_0^1 u' p(x) u' \, l_e \, d\xi \qquad Z_{qe} = \int_0^1 u q(x) u \, l_e \, d\xi$$

$$Z_{fe} = \int_0^1 u f(x) \, l_e \, d\xi$$

Now we replace the function u and it's derivative u' with the linear elemental approximation using the shape functions in the elemental coordinate ξ. For example, consider the integral Z_{pe}:

$$Z_{pe} \approx \frac{1}{l_e^2} \int_0^1 \mathbf{u}_e^T \mathbf{N}' p(x) \mathbf{N}'^T \mathbf{u}_e \; l_e \, d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Since the vectors $\mathbf{u}_e$ and $\mathbf{u}_e^T$ are not functions of ξ, then these terms may be placed outside the integral. The resulting form of the integral Z_{pe} is:

$$Z_{pe} \approx \mathbf{u}_e^T \left(\frac{1}{l_e^2} \int_0^1 \mathbf{N}' p(x) \mathbf{N}'^T \, l_e \, d\xi \right) \mathbf{u}_e \quad = \quad \mathbf{u}_e^T \mathbf{p}_e \mathbf{u}_e$$

where $\mathbf{p}_e$ is defined as:

$$\mathbf{p}_e = \frac{1}{l_e} \int_0^1 \mathbf{N}' p(x) \mathbf{N}'^T \, d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - The integrals Z_{qe} and Z_{fe} may be written in a similar manner:

$$Z_{qe} \approx \mathbf{u}_e^T \left(\int_0^1 \mathbf{N} q(x) \mathbf{N}^T \, l_e \, d\xi \right) \mathbf{u}_e \quad = \quad \mathbf{u}_e^T \mathbf{q}_e \mathbf{u}_e$$

$$\mathbf{q}_e = \int_0^1 \mathbf{N} q(x) \mathbf{N}^T \, l_e d\xi$$

$$Z_{fe} \approx \mathbf{u}_e^T \left(\int_0^1 \mathbf{N} f(x) \, l_e \, d\xi \right) \quad = \quad \mathbf{u}_e^T \mathbf{f}_e$$

$$\mathbf{f}_e = \int_0^1 \mathbf{N} f(x) \, l_e d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Let's examine in detail the integrals $\mathbf{p}_e$, $\mathbf{q}_e$, and $\mathbf{f}_e$. Consider the integral $\mathbf{p}_e$:

$$\mathbf{p}_e = \frac{1}{l_e} \int_0^1 \mathbf{N}' p(x) \mathbf{N}'^T \, d\xi = \frac{1}{l_e} \int_0^1 \begin{Bmatrix} N_i' \\ N_{i+1}' \end{Bmatrix} p(x) \begin{Bmatrix} N_i' & N_{i+1}' \end{Bmatrix} d\xi$$

$$= \frac{1}{l_e} \int_0^1 \begin{Bmatrix} -1 \\ 1 \end{Bmatrix} p(x) \begin{Bmatrix} -1 & 1 \end{Bmatrix} d\xi \qquad x = x_i + \xi l_e$$

$$= \frac{1}{l_e} \int_0^1 p(x_i + \xi l_e) \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Let's turn our attention to the $\mathbf{q}_e$ integrals.

$$\mathbf{q}_e = \int_0^1 \mathbf{N} q(x) \mathbf{N}^T \, l_e d\xi = \int_0^1 \begin{Bmatrix} N_i \\ N_{i+1} \end{Bmatrix} q(x) \begin{Bmatrix} N_i & N_{i+1} \end{Bmatrix} l_e d\xi$$

$$= \int_0^1 \begin{Bmatrix} 1-\xi \\ \xi \end{Bmatrix} q(x) \begin{Bmatrix} 1-\xi & \xi \end{Bmatrix} l_e d\xi \qquad x = x_i + \xi l_e$$

$$= \int_0^1 q(x_i + \xi l_e) \begin{bmatrix} (1-\xi)^2 & \xi(1-\xi) \\ \xi(1-\xi) & \xi^2 \end{bmatrix} l_e d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Now let us consider the $\mathbf{f}_e$ integrals:

$$\mathbf{f}_e = \int_0^1 \mathbf{N} f(x) \, l_e d\xi = \int_0^1 \begin{Bmatrix} N_i \\ N_{i+1} \end{Bmatrix} f(x) \, l_e d\xi$$

$$\mathbf{f}_e = \int_0^1 \begin{Bmatrix} 1-\xi \\ \xi \end{Bmatrix} f(x_i + \xi l_e) \, l_e d\xi$$

FINITE ELEMENT FORMULATIONS

Assembly - The functional $Z(u)$, through the discretization and interpolation procedures has been converted into an approximate function $Z(u_1, u_2, u_3, \ldots, u_{N+1})$, which may be written as:

$$Z(u_1,u_2,u_3,\ldots,u_{N+1}) = \frac{\sum \mathbf{u_e}^T \mathbf{k_e} \mathbf{u_e}}{2} - \sum \mathbf{u_e}^T \mathbf{f_e} + \frac{\beta u_{N+1}^2}{2} + \frac{\alpha u_1^2}{2} - B u_{N+1} - A u_1$$

where $\mathbf{k_e} = \mathbf{p_e} + \mathbf{q_e}$. Consider the $\mathbf{k_e}$ term from the first element. On an element level, the term $\mathbf{u_e}^T \mathbf{k_e} \mathbf{u_e}$ has the form:

$$\mathbf{u_e^T k_e u_e} = \langle u_1 \quad u_2 \rangle \begin{bmatrix} (k_{11})_1 & (k_{12})_1 \\ (k_{21})_1 & (k_{22})_1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Assembly - Substituting this element "stiffness" matrix into the global system would result in something like this:

$$\mathbf{u_e^T k_e u_e} = \langle u_1 \quad u_2 \quad u_3 \cdots u_{N+1} \rangle \begin{bmatrix} (k_{11})_1 & (k_{12})_1 & 0 & \cdots & 0 \\ (k_{21})_1 & (k_{22})_1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N+1} \end{Bmatrix} = \mathbf{u_G^T k_{G1} u_G}$$

The above equation is intended to show how each element contributes to the global system.

FINITE ELEMENT FORMULATIONS

Assembly - In a similar manner, the contribution of the forcing function f from the first element may be written as:

$$\mathbf{u_e^T f_e} = \langle u_1 \quad u_2 \rangle \begin{Bmatrix} (f_1)_1 \\ (f_2)_1 \end{Bmatrix}$$

The global contribution of the first element is:

$$\mathbf{u_e^T f_e} = [u_1 \; u_2 \; u_3 \; \ldots \; u_{N+1}] \begin{Bmatrix} (f_1)_1 \\ (f_2)_1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{Bmatrix} = \mathbf{u_G^T f_{G1}}$$

FINITE ELEMENT FORMULATIONS

Assembly - In a global sense the linear approximation for the energy functional may be written as:

$$Z(u_1 \; u_2 \; u_3 \; \ldots \; u_{N+1}) = \frac{\mathbf{u_G}^T \mathbf{K_G} \mathbf{u_G}}{2} - \mathbf{u_G}^T \mathbf{F_G}$$

$$\mathbf{K_G} = \sum_e \mathbf{k_G} + \mathbf{BT_G} \qquad \mathbf{F_G} = \sum_e \mathbf{f_G} + \mathbf{bt_G}$$

$$\mathbf{BT_G} = \begin{bmatrix} \alpha & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \beta \end{bmatrix} \qquad \mathbf{bt_G} = \begin{Bmatrix} A \\ 0 \\ 0 \\ \vdots \\ B \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Assembly - Recall the energy functional $Z(u)$ has the form:

$$Z(u_1,u_2,u_3,\ldots,u_{N+1})$$

and has a stationary value that is obtained by requiring each partial derivative to vanish:

$$\frac{\partial Z}{\partial u_i} = 0 \qquad i = 1, 2, \ldots, N+1$$

$$\frac{\partial Z}{\partial \mathbf{u_G}} = 0 = \frac{\partial}{\partial \mathbf{u_G}} \left(\frac{\mathbf{u_G}^T \mathbf{K_G} \mathbf{u_G}}{2} - \mathbf{u_G}^T \mathbf{F_G} \right) \rightarrow \boxed{\mathbf{K_G u_G} = \mathbf{F_G}}$$

We have converted the original functional into a function of the nodal unknowns and then required the functional to be stationary with respect to each node using the Ritz method.

FINITE ELEMENT FORMULATIONS

Constraints - Essential or forced boundary conditions may be inserted into the system equations by simply rewriting the appropriate equation corresponding to the variable where the condition is prescribed.

For example, the global system "stiffness" matrix constrained by the boundary condition $u(x_1) = C$ is:

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ K_{21} & K_{22} & K_{23} & \cdots & K_{2,N+1} \\ K_{31} & K_{32} & K_{33} & \cdots & K_{3,N+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ K_{N+1,1} & K_{N+1,2} & K_{N+1,3} & \cdots & K_{N+1,N+1} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N+1} \end{Bmatrix} = \begin{Bmatrix} C \\ F_2 \\ F_3 \\ \vdots \\ F_{N+1} \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Constraints - If *C* is zero then the first column in the "stiffness" matrix may also be replace by zeros.

However, for non-zero values of *C* we cannot replace the stiffness components in the first column.

The unconstrained system equations were originally represented by a symmetric stiffness matrix.

With boundary conditions accounted for, the stiffness matrix has become unsymmetrical.

FINITE ELEMENT FORMULATIONS

Constraints - To symmetrize the equations the terms in corresponding column (in this case the first column), the terms are multiplied by the value of the boundary condition and moved to the right-hand side of the equations.

The resulting symmetric set of equations is:

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & K_{22} & K_{23} & \cdots & K_{2,N+1} \\ 0 & K_{32} & K_{33} & \cdots & K_{3,N+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & K_{N+1,2} & K_{N+1,3} & \cdots & K_{N+1,N+1} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N+1} \end{Bmatrix} = \begin{Bmatrix} C \\ F_2 - K_{21}C \\ F_3 - K_{31}C \\ \vdots \\ F_{N+1} - K_{N+1,1}C \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Solution - The equations are ready to be solved.

Since we have spent some energy symmetrizing the global equations, we should use an equation solver that uses a symmetric storage algorithm.

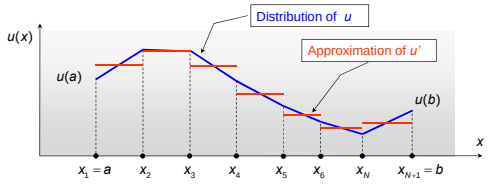
Computation of Derived Variables - For the Sturm-Liouville problem we have been discussing, the quantity *u'* is not solved for directly.

FINITE ELEMENT FORMULATIONS

Computation of Derived Variables - Since we used a linear interpolation in the variational formulation we can calculate an approximate value of *u*' the linear elemental interpolation functions:

$$u_e' = \frac{u_{i+1} - u_i}{l_e}$$

Notice that since linear interpolation function were employed, the derivative is constant over each element:



FINITE ELEMENT FORMULATIONS

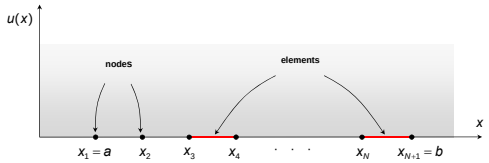
Galerkin Finite Element Models

The steps involved in generating a FEM model using variational techniques follows the same procedure as we used in our discussion of the spring-mass system:

1. Discretization
2. Interpolation
3. Elemental Description or Formulation
4. Assembly
5. Constraints
6. Solution
7. Computation of Derived Variables

FINITE ELEMENT FORMULATIONS

Discretization - The domain is broken up into a series of subintervals:



The points that separate the subintervals are called **nodes** and the subdomains between the nodes are called **elements**.

The entire discretization is often called a **mesh**. Later we will discuss how the spacing of the mesh may effect the solution.

FINITE ELEMENT FORMULATIONS

Interpolation - At this point we have to decide what type of interpolation or approximation we want over each element.

In past discussions of variational model we used a **linear element.**

An element of this type assumes a linear variation of the function over the element.

For a Galerkin FEM model we will reexamine the concept of interpolation over an element.

FINITE ELEMENT FORMULATIONS

Interpolation - In developing a Galerkin FEM model, the solution is represented in the form of a set of admissible functions:

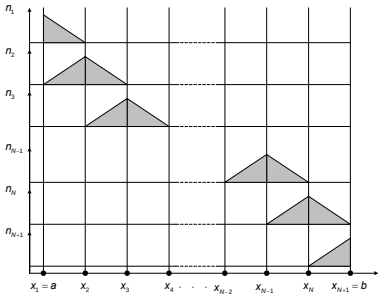
$$u = \sum_{i=1}^{N+1} u_i n_i(x)$$

where each of the function $n_i(x)$ is composed of parts of the adjacent interpolation functions $N_i(x)$.

The functions $n_i(x)$ may be called the **basis functions** for the piecewise linear approximation over the interval [a, b].

FINITE ELEMENT FORMULATIONS

Interpolation - In developing a Galerkin FEM model, the solution was represented in the form of a set of admissible functions:



FINITE ELEMENT FORMULATIONS

Element Formulation - Let's look at a Galerkin approach to the Sturm-Liouville problem.

Recall the general form of the MWR model of the Sturm-Liouville equation:

$$\int_a^b \left(v'pu' - vqu\right)dx + \alpha v(a)u(a) + \beta v(b)u(b) = \int_a^b (vf)\,dx + Av(a) + Bv(b)$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Substituting our approximation of the function u into the above expression from the weighted residual or the error statement:

$$E_{N+1}(U) = \int_a^b \left(v'p\sum_1^{N+1} u_i n_i' - vq\sum_1^{N+1} u_i n_i\right)dx + \alpha v(a)u(a) + \beta v(b)u(b) - \int_a^b (vf)\,dx - Av(a) - Bv(b) = 0$$

The Galerkin MWR model requires that the sum of the residual E_{N+1} be zero.

FINITE ELEMENT FORMULATIONS

Element Formulation - If we take the weighting functions $v(x)$ as the basis function then the MWR statement becomes:

$$\sum_i^{N+1}\left[\int_a^b \left(n_k'pn_i' - n_k q n_i\right)u_i dx\right] + \alpha u_i\delta_{k1} + \beta u_{N+1}\delta_{kN+1} = \int_a^b (n_k f)\,dx + A\delta_{k1} + B\delta_{kN+1}$$

where δ_{ij} is the **Knonecker delta function** given as:

$$\delta_{ij} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

which implies that δ_{kN+1} term occurs only during the N+1 equation and δ_{k1} during the first equation.

FINITE ELEMENT FORMULATIONS

Element Formulation - The approximation equations may be written in matrix form as:

$$\sum A_{ki} u_i = b_k \qquad k = 1, 2, \ldots, N+1 \qquad \rightarrow \quad \mathbf{Au} = \mathbf{b}$$

where the terms A_{ki} and b_k are defined as:

$$A_{ki} = \int_a^b \left(n'_k p n'_i - n_k q n_i \right) u_i dx + \alpha u_i \delta_{k1} + \beta u_{N+1} \delta_{kN+1}$$

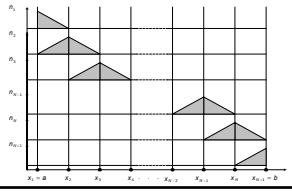
$$b_k = \int_a^b \left(n_k f \right) dx + A \delta_{k1} + B \delta_{kN+1}$$

FINITE ELEMENT FORMULATIONS

Element Formulation - This formulation produces a set of N+1 linear equations in N+1 unknowns.

For the k^{th} equation, the basis functions n_k and n'_k supply what we will call **local support**.

This means the basis function n_k and n'_k are nonzero in a local neighborhood of the node at x_k



FINITE ELEMENT FORMULATIONS

Element Formulation - Therefore, the k^{th} equation corresponding to the node at x_k reduces to:

$$\sum \left(\int_{x_{k-1}}^{x_{k+1}} \left(n'_k p n'_i - n_k q n_i \right) dx \right) u_i = \int_{x_{k-1}}^{x_{k+1}} \left(n_k f \right) dx$$

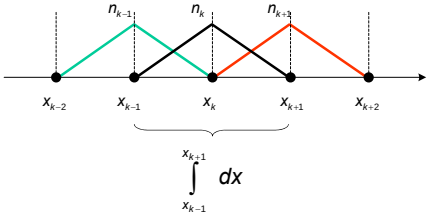
Let's look a little closer at the equation associated with the i^{th} node:

$$u_{k-1} \int_{x_{k-1}}^{x_{k+1}} \left(n'_k p n'_{k-1} - n_k q n_{k-1} \right) dx + u_k \int_{x_{k-1}}^{x_{k+1}} \left(n'_k p n'_k - n_k q n_k \right) dx$$

$$+ u_{k+1} \int_{x_{k-1}}^{x_{k+1}} \left(n'_k p n'_{k+1} - n_k q n_{k+1} \right) dx = \int_{x_{k-1}}^{x_{k+1}} \left(n_k f \right) dx$$

FINITE ELEMENT FORMULATIONS

Element Formulation



Notice that n_{k-1} is zero over the interval $[x_k, x_{k+1}]$ and n_{k+1} is zero over the interval $[x_{k-1}, x_k]$.

FINITE ELEMENT FORMULATIONS

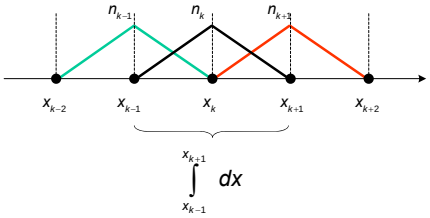
Element Formulation - Since n_{k+1} and n'_{k+1} are zero over the interval $[x_{k-1}, x_k]$ and n_{k-1} and n'_{k-1} are zero over the interval $[x_k, x_{k+1}]$ then the above equation becomes:

$$u_{k-1} \int_{x_{k-1}}^{x_k} \left(n'_k p n'_{k-1} - n_k q n_{k-1} \right) dx + u_k \int_{x_{k-1}}^{x_{k+1}} \left(n'_k p n'_k - n_k q n_k \right) dx$$

$$+ u_{k+1} \int_{x_k}^{x_{k+1}} \left(n'_k p n'_{k+1} - n_k q n_{k+1} \right) dx = \int_{x_{k-1}}^{x_{k+1}} \left(n_k f \right) dx$$

FINITE ELEMENT FORMULATIONS

Element Formulation



Do these functions look familiar?

FINITE ELEMENT FORMULATIONS

Element Formulation - In order to compare this formulation with the Ritz approach we discussed previously, let's write out a set of equations for N+1 = 3 as:

$$\begin{bmatrix} \int_a^{x_2} F_{11}\,dx\ u_1 + \int_a^{x_2} F_{12}\,dx\ u_2 & & +\alpha u_1 = \int_a^{x_2} (n_1 f)dx + A \\ \int_a^{x_2} F_{21}\,dx\ u_1 + \int_a^{x_2} F_{22}\,dx\ u_2 & + \int_{x_2}^{b} F_{22}\,dx\ u_2 + \int_{x_2}^{b} F_{23}\,dx\ u_3 & = \int_a^{x_2} (n_2 f)dx + \int_{x_2}^{b} (n_2 f)dx \\ & \int_{x_2}^{b} F_{32}\,dx\ u_2 + \int_{x_2}^{b} F_{33}\,dx\ u_3 & +\beta u_1 = B + \int_{x_2}^{b} (n_3 f)dx \end{bmatrix}$$

Element #1 Element #2 Element #1 Element #2

$$F_{ij} = n'_i p n'_j + n_i q n_j$$

FINITE ELEMENT FORMULATIONS

PROBLEM #11 - Complete the verification that the equations of the Galerkin approach coincide with the equations derived from the Ritz approach.

***Hint*:** First write the Galerkin method in elemental coordinates ξ and then determine if the elemental stiffness components are comparable and if the terms associated with the function $f(x)$ are identical.

Numerical Integration

Before we attempt to solve some problem using the Ritz or Galerkin FEM formulations, we need to discuss how to evaluate integrals of the form**:**

$$I = \int_0^1 f(\xi)\,d\xi$$

The basic idea behind any approximate integrations or quadrature method is to replace the actual function with a polynomial that accurately estimates the behavior of the integrand.

The form of the approximate functions is such that their exact integration may be easily computed.

Numerical Integration

Trapezoidal Rule - In this method the function $f(\xi)$ is replace by a series of connected linear functions $F(\xi)$. The integral may be estimated as:

$$I = \sum_1^N \int_{\xi_i}^{\xi_{i+1}} F(\xi)\,d\xi$$

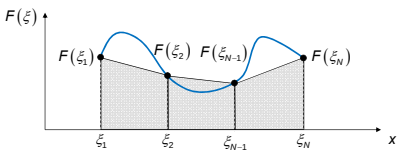
where F is a linear function over the interval $[\xi_i, \xi_{i+1}]$. Computing the area of the trapezoidal region under the line F over each subinterval results in:

$$I = \sum_1^N \left[F(\xi_i) + F(\xi_{i+1})\right]\frac{h_i}{2} \qquad h_i = \xi_{i+1} - \xi_i$$

Numerical Integration

Trapezoidal Rule - If all the subintervals are assumed to be the same size, then the integral may be written as:

$$I = \frac{h}{2}\left(F(\xi_1) + 2\sum_{i=2}^{N-1} F(\xi_i) + F(\xi_N)\right)$$



Numerical Integration

Simpson's Rule - In this method the function $f(\xi)$ is replace by a series of connected quadratic functions $F(\xi)$.

The quadratic function is defined over two subintervals of length $2h_i$.

The approximated quadratic curve may be integrated directly over both subintervals.

The resulting value of the integral is:

$$I = \sum_1^N \left(F(\xi_{i-1}) + 4F(\xi_i) + F(\xi_{i+1})\right)\frac{h_i}{3}$$

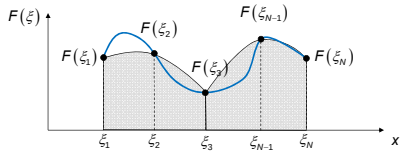
$$h_i = \frac{\xi_{i+1} - \xi_{i-1}}{2}$$

Numerical Integration

Simpson's Rule - If all the subintervals are assumed to be the same size, then the integral may be written as:

$$I = \frac{h}{3}\left(F(\xi_1) + 4\sum_{i=2,4,6...}^{N-1} F(\xi_i) + 2\sum_{i=3,5,7...}^{N-2} F(\xi_i) + F(\xi_N) \right)$$

This formula is often called **Simpson's 1/3 Rule**.



Numerical Integration

Gauss-Legendre Quadrature - In this method, a series of specialized points and weights are developed to evaluate the integral.

We will briefly discuss how these points and weights are determined.

The general form of the quadrature is:

$$I = \int_{-1}^{1} F(u)\,du = \sum_{1}^{N} w_i F(u_i)$$

Numerical Integration

Gauss-Legendre Quadrature - A general integral:

$$I = \int_a^b F(\xi)\,d\xi$$

may be converted into the Gaussian integration space [-1, 1] by the following transformation:

$$\xi = \frac{b+a}{2} + \frac{(b-a)u}{2} \qquad d\xi = \frac{b-a}{2}du$$

Therefore the integral has the form:

$$I = \int_{-1}^{1} F\left(\frac{b+a}{2} + \frac{(b-a)u}{2} \right)\frac{(b-a)}{2}du$$

Numerical Integration

Gauss-Legendre Quadrature - In both the Ritz and Galerkin formulations, integrals of the form were developed:

$$I = \int_0^1 F(\xi)\,d\xi$$

therefore the Gaussian quadrature form is:

$$I = \int_{-1}^{1} F\left(\frac{b+a}{2} + \frac{(b-a)u}{2} \right)\frac{(b-a)}{2}du$$

$$I = \int_{-1}^{1} F\left(\frac{1+u}{2} \right)\frac{du}{2}$$

Numerical Integration

Gauss-Legendre Quadrature - Let's examine how the points and weights for Gaussian quadrature are developed. Assume a one-point method, therefore $N = 1$:

$$I = \int_{-1}^{1} F(u)\,du \quad = \quad w_1 F(u_1)$$

The quadrature point u_1 and weight w_1 are determined such that any linear function of u may be integrated exactly.

Numerical Integration

Gauss-Legendre Quadrature - We can accomplish this task by solving the following two equations for u_1 and w_1:

$$I = \int_{-1}^{1} 1\,du = 2 = w_1 F(u_1) = w_1$$

$$I = \int_{-1}^{1} u\,du = 0 = w_1 F(u_1) = w_1 u_1$$

$$\begin{cases} w_1 = 2 \\ u_1 = 0 \end{cases}$$

Thus the one-point Gaussian quadrature takes the form:

$$I = \int_{-1}^{1} F(u)\,du = 2F(0)$$

Numerical Integration

Gauss-Legendre Quadrature - To determine the points and weights for a two-point Gaussian quadrature, $N = 2$, the following equations should be solved:

$F = 1 \qquad \int_{-1}^{1} 1\, du = 2 = w_1 F(u_1) + w_2 F(u_2) = w_1 + w_2$

$F = u \qquad \int_{-1}^{1} u\, du = 0 = w_1 F(u_1) + w_2 F(u_2) = w_1 u_1 + w_2 u_2$

$F = u^2 \qquad \int_{-1}^{1} u^2\, du = \frac{2}{3} = w_1 F(u_1) + w_2 F(u_2) = w_1 u_1^2 + w_2 u_2^2$

$F = u^3 \qquad \int_{-1}^{1} u^3\, du = 0 = w_1 F(u_1) + w_2 F(u_2) = w_1 u_1^3 + w_2 u_2^3$

Numerical Integration

Gauss-Legendre Quadrature - Solving this set of equations gives:

$$w_1 = w_2 = 1 \qquad u_1 = -\frac{1}{\sqrt{3}} \qquad u_2 = \frac{1}{\sqrt{3}}$$

Thus the two-point Gaussian quadrature takes the form:

$$I = \int_{-1}^{1} F(u)\, du \quad = \quad F\left(-\frac{1}{\sqrt{3}}\right) + F\left(\frac{1}{\sqrt{3}}\right)$$

Therefore, a two-point Gaussian quadrature formula will exactly integrate up to a third-order polynomial.

Numerical Integration

Gauss-Legendre Quadrature - This type of analysis may be continued to find points and weights for higher-order integrations.

Gauss-Legendre quadrature has the property that an N^{th} order approximation integrates exactly a polynomial of degree $2N$-1 or less.

Order N	Points u_i	Weights w_i
1	0.000000000	2.00000000
2	±0.57735269	1.00000000
3	0.000000000	0.88888889
	±0.774596669	0.55555556
4	±0.339981044	0.65214515
	±0.861136312	0.34785485
5	0.000000000	0.568888889
	±0.538469310	0.478628670
	±0.906179845	0.236926885
6	±0.238610186	0.467913934
	±0.661209386	0.360761573
	±0.932469514	0.171324492

Numerical Integration

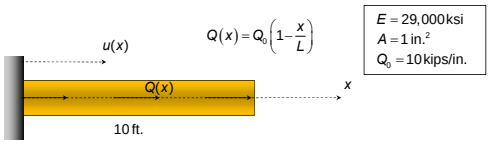
PROBLEM #12 - To compare and contrast the different integration methods we have discussed, evaluate the following integral:

$$I = \int_{1}^{3} \frac{e^x \sin(x)}{1 + x^2}\, dx$$

by: 1) **trapezoidal rule** with one, two, and four intervals;
2) **Simpson's rule** using two and four intervals;
3) **Gaussian quadrature** for $N = 1$ to 6.

FINITE ELEMENT FORMULATIONS

Example - Consider the problem of the axial deformation of a prismatic bar we worked previously.



The boundary value problem for this case is:

$$(AEu')' + Q(x) = 0 \qquad 0 \le x \le L$$

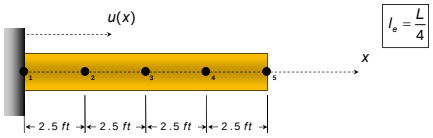
the boundary conditions are: $\begin{cases} u(0) = 0 \\ AEu'(L) = 0 \end{cases}$

FINITE ELEMENT FORMULATIONS

Example - The Sturm-Liouville form of this equation requires that $p = AE$, $q = 0$, and A = B = α = β = 0. The corresponding functional is:

$$Z(u) = \int_0^L \left(\frac{AE(u')^2}{2} - Q(x)u \right) dx$$

Discretization - The domain will be divided into four elements.



FINITE ELEMENT FORMULATIONS

Interpolation - We will use **linear elements.** In developing a Ritz FEM model, the solution was represented in the form of a set of admissible functions:

$$u = N_i u_i + N_{i+1} u_{i+1} \qquad x = N_i x_i + N_{i+1} x_{i+1}$$

Element Formulation - The approximation of the energy functional may be written in the following form:

$$Z(u) = \sum_e \left(\frac{Z_{pe}}{2} - Z_{fe} \right) dx$$

$$Z_{pe} = \frac{1}{l_e} \int_0^1 u' AE u' \, d\xi \qquad Z_{fe} = \int_0^1 Q(x_i + \xi l_e) u \, l_e d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Now we replace *u*' with the linear elemental approximation using the shape functions in the elemental coordinate ξ.

For example, consider the integral Z_{pe}:

$$Z_{pe} \approx \mathbf{u_e^T} \left(\frac{1}{l_e} \int_0^1 \mathbf{N}' AE \mathbf{N}'^{\mathbf{T}} \, d\xi \right) \mathbf{u_e} \quad = \quad \mathbf{u_e^T p_e u_e}$$

$$\mathbf{p_e} = \frac{1}{l_e} \int_0^1 \mathbf{N}' \, AE \, \mathbf{N}'^{\mathbf{T}} \, d\xi \quad = \frac{1}{l_e} \int_0^1 \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} AE \, d\xi$$

FINITE ELEMENT FORMULATIONS

Element Formulation - Now we replace *u*' with the linear elemental approximation using the shape functions in the elemental coordinate ξ.

For example, consider the integral Z_{fe}:

$$Z_{fe} \approx \mathbf{u_e^T} \left(\int_0^1 \mathbf{N} Q(x) l_e \, d\xi \right) \mathbf{u_e} \quad = \quad \mathbf{u_e^T f_e}$$

$$\mathbf{f_e} = \int_0^1 \mathbf{N} Q(x) l_e d\xi \quad = \int_0^1 \begin{Bmatrix} 1-\xi \\ \xi \end{Bmatrix} Q(x_i + \xi l_e) l_e d\xi$$

FINITE ELEMENT FORMULATIONS

Assembly - The functional $Z(u)$, through the discretization and interpolation procedures has been converted into an approximate function $Z(u_1, u_2, u_3, \ldots, u_{N+1})$, which may be written as:

$$Z(u_1, u_2, u_3, \ldots, u_{N+1}) = \left(\frac{4AE}{L} \right) \frac{\sum \mathbf{u_e}^{\mathbf{T}} \mathbf{k_e} \mathbf{u_e}}{2} - \sum \mathbf{u_e}^{\mathbf{T}} \mathbf{f_e}$$

$$\frac{\partial Z}{\partial \mathbf{u_G}} = 0 \quad \rightarrow \quad \boxed{\mathbf{K_G u_G} = \mathbf{F_G}}$$

where:

$$\mathbf{K_G} = \sum_e \mathbf{k_G} \qquad \mathbf{F_G} = \sum_e \mathbf{f_G}$$

where $\mathbf{k_e} = \mathbf{p_e}$

FINITE ELEMENT FORMULATIONS

Assembly - Consider the $\mathbf{k_e}$ term for each element in the system.

For element #1: $x_i = 0$

$$\mathbf{k_1 u_1} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}$$

For element #2: $x_i = 2.5$ ft.

$$\mathbf{k_2 u_2} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_2 \\ u_3 \end{Bmatrix}$$

For element #3 : $x_i = 5$ ft.

$$\mathbf{k_3 u_3} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_3 \\ u_4 \end{Bmatrix}$$

For element #4 : $x_i = 7.5$ ft.

$$\mathbf{k_4 u_4} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_4 \\ u_5 \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Assembly - The right-hand side terms involving the loading function for each element are:

$$\mathbf{f_e} = Q_0 \int_0^1 \begin{bmatrix} 1-\xi \\ \xi \end{bmatrix} \left(1 - \frac{x_i + \xi l_e}{L} \right) l_e d\xi = \begin{Bmatrix} \frac{Q_0 l_e}{6L} \left(-3x_i - l_e + 3L \right) \\ \frac{Q_0 l_e}{6L} \left(-3x_i - 2l_e + 3L \right) \end{Bmatrix}$$

for $l_e = L/4$ the element integrals become:

$$\mathbf{f_e} = \begin{Bmatrix} \frac{Q_0}{96} \left(11L - 12x_i \right) \\ \frac{Q_0}{96} \left(10L - 12x_i \right) \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Assembly - Consider the f_e term for each element in the system.

For element #1: $x_i = 0$

$$\mathbf{f}_1 = \begin{Bmatrix} 11\dfrac{Q_0 L}{96} \\ 10\dfrac{Q_0 L}{96} \end{Bmatrix}$$

For element #2: $x_i = 2.5$ ft.

$$\mathbf{f}_2 = \begin{Bmatrix} 8\dfrac{Q_0 L}{96} \\ 7\dfrac{Q_0 L}{96} \end{Bmatrix}$$

For element #3 : $x_i = 5$ ft.

$$\mathbf{f}_3 = \begin{Bmatrix} 5\dfrac{Q_0 L}{96} \\ 4\dfrac{Q_0 L}{96} \end{Bmatrix}$$

For element #4 : $x_i = 7.5$ ft.

$$\mathbf{f}_4 = \begin{Bmatrix} 2\dfrac{Q_0 L}{96} \\ 1\dfrac{Q_0 L}{96} \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Assembly - Compiling these terms into the global system equations gives:

$$\mathbf{K_G u_G} = \mathbf{F_G}$$

$$= \frac{4AE}{L}\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{Bmatrix} = \frac{Q_0 L}{96}\begin{Bmatrix} 11 \\ 18 \\ 12 \\ 6 \\ 1 \end{Bmatrix}$$

Element 1 Element 2 Element 3 Element 4

FINITE ELEMENT FORMULATIONS

Constraints - Essential or forced boundary conditions may be inserted into the system equations by simply rewriting the appropriate equation corresponding to the variable where the condition is prescribed.

The global system "stiffness" matrix constrained by the boundary condition $u(0) = 0$ is:

$$\frac{4AE}{L}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{Bmatrix} = \frac{Q_0 L}{96}\begin{Bmatrix} 0 \\ 18 \\ 12 \\ 6 \\ 1 \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Solution - The equations are ready to be solved.

$$u_1 = 0 \qquad u_2 = 37\frac{Q_0 L^2}{384AE} \qquad u_3 = 56\frac{Q_0 L^2}{384AE}$$

$$u_4 = 63\frac{Q_0 L^2}{384AE} \qquad u_5 = 64\frac{Q_0 L^2}{384AE}$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u_1 = 0 \qquad u_2 = 0.4784 \text{ in.} \qquad u_3 = 0.7241 \text{ in.}$$

$$u_4 = 0.8147 \text{ in.} \qquad u_5 = 0.8276 \text{ in.}$$

FINITE ELEMENT FORMULATIONS

Solution - The exact solution may be determined from the following expression:

$$AEu(x) = -\int^{x}\int^{x'} Q(x)\,dx'\,dx = -\int^{x}\int^{x'} Q_0\left(1-\frac{x}{L}\right)dx'\,dx$$

$$= Q_0 L^2\left[\frac{1}{6}\left(\frac{x}{L}\right)^3 - \frac{1}{2}\left(\frac{x}{L}\right)^2 + \frac{1}{2}\left(\frac{x}{L}\right)\right]$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u_1 = 0 \qquad u_2 = 0.4784 \text{ in.} \qquad u_3 = 0.7241 \text{ in.}$$

$$u_4 = 0.8147 \text{ in.} \qquad u_5 = 0.8276 \text{ in.}$$

FINITE ELEMENT FORMULATIONS

Exact Solution - The exact solution may be determined from the following expression:

$$u_1 = 0 \qquad u_2 = 0.4784 \text{ in.} \qquad u_3 = 0.7241 \text{ in.}$$

$$u_4 = 0.8147 \text{ in.} \qquad u_5 = 0.8276 \text{ in.}$$

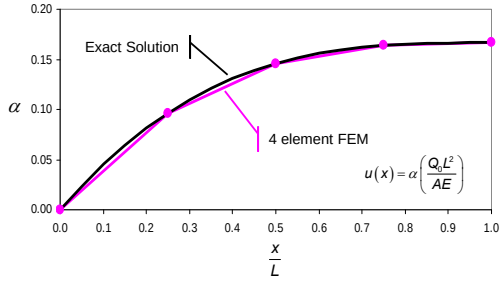
FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u_1 = 0 \qquad u_2 = 0.4784 \text{ in.} \qquad u_3 = 0.7241 \text{ in.}$$

$$u_4 = 0.8147 \text{ in.} \qquad u_5 = 0.8276 \text{ in.}$$

FINITE ELEMENT FORMULATIONS

FEM Solution - Comparison of 4 element FEM formulation with exact solution



FINITE ELEMENT FORMULATIONS

Computation of Derived Variables - For the Sturm-Liouville problem we have been discussing, the quantity u' is not solved for directly.

Since we used a linear interpolation in the variational formulation we can calculate an approximate value of u' using the linear elemental interpolation functions:

$$u_e' = \frac{u_{i+1} - u_i}{l_e}$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u'_{element\,1} = 37\frac{Q_0L}{96AE} \qquad u'_{element\,2} = 19\frac{Q_0L}{96AE} \qquad u'_{element\,3} = 6\frac{Q_0L}{96AE} \qquad u'_{element\,4} = \frac{Q_0L}{96AE}$$

FINITE ELEMENT FORMULATIONS

Computation of Derived Variables - The exact solution for the first derivative is:

$$AEu'(x) = -\int^{x'} Q(x)dx'dx = -\int^{x'} Q_0\left(1-\frac{x}{L}\right)dx'dx$$

$$= Q_0L^2\left[\frac{1}{2}\left(\frac{x}{L}\right)^2 - \left(\frac{x}{L}\right) + \frac{1}{2}\right]$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u'_{element\,1} = 37\frac{Q_0L}{96AE} \qquad u'_{element\,2} = 19\frac{Q_0L}{96AE} \qquad u'_{element\,3} = 6\frac{Q_0L}{96AE} \qquad u'_{element\,4} = \frac{Q_0L}{96AE}$$

FINITE ELEMENT FORMULATIONS

Computation of Derived Variables - The exact solution for the first derivative is:

$$u'_1 = 48\frac{Q_0L}{96AE} \qquad u'_2 = 27\frac{Q_0L}{96AE} \qquad u'_3 = 12\frac{Q_0L}{96AE}$$

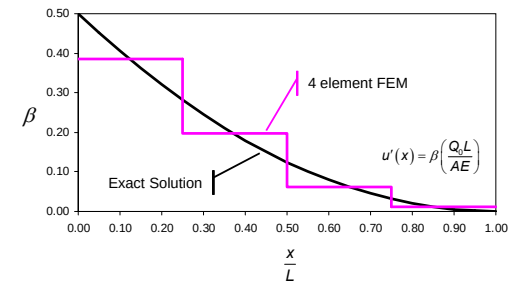
$$u'_4 = 3\frac{Q_0L}{96AE} \qquad u'_5 = 0$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u'_{element\,1} = 37\frac{Q_0L}{96AE} \qquad u'_{element\,2} = 19\frac{Q_0L}{96AE} \qquad u'_{element\,3} = 6\frac{Q_0L}{96AE} \qquad u'_{element\,4} = \frac{Q_0L}{96AE}$$

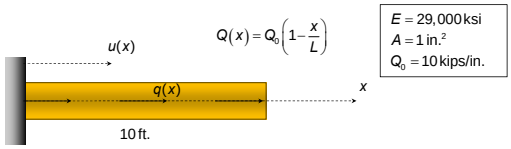
FINITE ELEMENT FORMULATIONS

FEM Solution - Comparison of 4 element FEM formulation with exact solution



FINITE ELEMENT FORMULATIONS

Example - Work the preceding problem again using eight equally-spaced elements.



The boundary value problem for this case is:

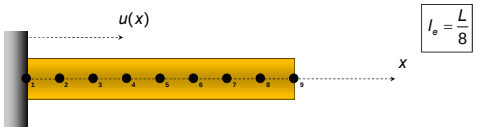
$$(AEu')' + Q(x) = 0 \qquad 0 \le x \le L$$

the boundary conditions are: $\begin{cases} u(0) = 0 \\ AEu'(L) = 0 \end{cases}$

FINITE ELEMENT FORMULATIONS

Example - Work the preceding problem again using eight equally-spaced elements.

Discretization - The domain will be divided into eight elements.



$$l_e = \frac{L}{8}$$

FINITE ELEMENT FORMULATIONS

Example - Applying the values for the geometry, material properties, and loading given in this problem and the boundary condition results in:

$$\frac{8AE}{L}\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{Bmatrix} = \frac{Q_0 L}{384} \begin{Bmatrix} 0 \\ 42 \\ 36 \\ 30 \\ 24 \\ 18 \\ 12 \\ 6 \\ 1 \end{Bmatrix}$$

FINITE ELEMENT FORMULATIONS

Example - The solution of these equations is:

$$u_1 = 0 \quad u_3 = 296\frac{Q_0 L^2}{3{,}072AE} \quad u_5 = 448\frac{Q_0 L^2}{3{,}072AE}$$

$$u_7 = 504\frac{Q_0 L^2}{3{,}072AE} \quad u_9 = 512\frac{Q_0 L^2}{3{,}072AE}$$

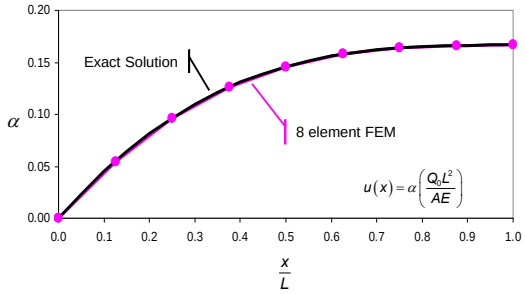
Example - Substituting the numerical values for Q_0, L, A and E results in:

$$u_1 = 0 \quad u_3 = 0.4784 \text{ in.} \quad u_5 = 0.7241 \text{ in.}$$

$$u_7 = 0.8147 \text{ in.} \quad u_9 = 0.8276 \text{ in.}$$

FINITE ELEMENT FORMULATIONS

FEM Solution - Comparison of 8 element FEM formulation with exact solution



FINITE ELEMENT FORMULATIONS

Computation of Derived Variables - The exact solution for the first derivative is:

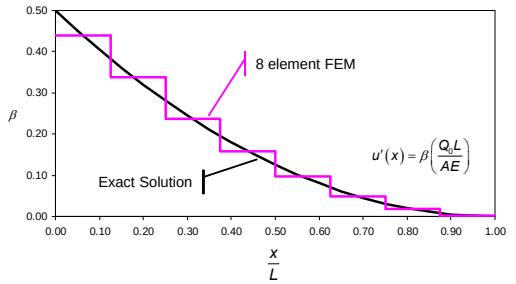
$$u'_1 = 169\frac{Q_0 L}{384AE} \quad u'_3 = 91\frac{Q_0 L}{384AE} \quad u'_5 = 37\frac{Q_0 L}{384AE} \quad u'_7 = 7\frac{Q_0 L}{384AE}$$

FEM Solution - Substituting the numerical values for Q_0, L, A, and E into the displacement expressions gives:

$$u'_1 = 0.01821 \quad u'_3 = 0.00981 \quad u'_5 = 0.00399 \quad u'_7 = 0.00075$$

FINITE ELEMENT FORMULATIONS

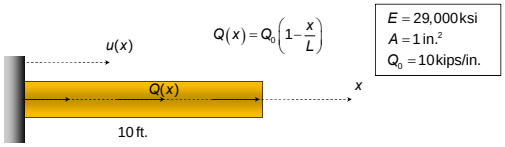
FEM Solution - Comparison of 8 element FEM formulation with exact solution



FINITE ELEMENT FORMULATIONS

PROBLEM #13 - Correct the ***"bleeped"*** version of **ODE2** - linear second-order differential equation solver.

Use the **ODE2** program to solve following axial deformation of the prismatic bar:



End of Chapter 2b