MEG 800 Finite Element Analysis 2.0 Weighted Integral Formulation

INSTRUCTOR: OA Fakinlede

oafak@unilag.edu.ng oafak@hotmail.com

Department of Systems Engineering,
University of Lagos

What is Finite Element Method?

* The Finite Element Method is a technique for constructing approximate solutions in an element wise application of the variational method.
* We have seen in the previous work that a problem can, on its own have a variational formulation such that the differential equations we are trying to solve is the Euler-Lagrange equations pertaining to the extremization of a functional. Here, we expand the scope and instead of starting with a variational formulation, we begin with a regular differential equation we want to solve.

Salient Problem

* The question is: Can we still solve this equation even when it is not known to be the Euler-Lagrange equation of an extremization? The answer to this lies in the method of weighted residuals.
* Recall that we obtained a weak form of the variation problem by an ingenious transfer of a gradient (differentiation) away from the primary variable. This in turn yields the secondary variables – the variation of which, on the boundaries constitute the Natural boundary conditions.
* The essential boundary conditions are the kinematic specifications that must be met a-priori by the trial functions to be admissible. These trial functions are used to approximate the primary variables (or their derivatives) that we are seeking.

2nd Order ODE

In this section, beginning with a second order ODE, we do not assume a variational formulation. Rather, we obtain an integral formulation using the method of weighted residuals. Consider the ODE:

$$-\frac{d}{dx}\left(a(x)\frac{d}{dx}u(x)\right)+f(x)=0, \qquad x\in[0,L]$$

$$\text{subject to: } u(0)=u_0, \qquad \left(a\frac{du}{dx}\right)_L=Q_L$$

Approximation Functions

Select a set of approximation functions $\phi_j(x)$ such that,

$$U_N(x) = c_j\phi_j(x) + \phi_0(x) \qquad j = 1,2,\dots,N$$

With the summation convention of Einstein as usual.

Substituting, we find

$$R(x, c_1, \dots, c_N) \equiv \frac{d}{dx}\left(a(x)\frac{dU_N(x)}{dx}\right) - f(x) \neq 0$$

If it were zero, then we have an exact solution.

Everything in the above equation is known apart from the N parameters $c_1, \dots, c_N$. We therefore need N equations to find them.

Methods for Generating Solutions

PROBLEM: How do we get this number of equations? What strategies are available?

Several strategies can be employed to generate the number of equations we need to find the N unknown parameters. These are:

* Collocation
* Least Squares
* Garlekin Weighted Residuals
* Other WRM

Collocation Method

Collocation. One straightforward method is the collocation method. We select N points in the domain $[0, L]$ and force equality such that,

$$R(x_i, c_1, \dots, c_N) = 0, \qquad i = 1,2, \dots, N.$$

This is the method of collocation. It has the unique disadvantage of removing the degree of freedom we have in dividing the same domain into smaller elements later as this will mean the points of collocation begin to get closer and closer to each other. This is the same as selecting a weighting function such that

$$\int_0^L w_i(x) R(x, c_1, \dots, c_N) dx$$

Where the weighting function, $w_i(x) = \delta(x - x_i)$. δ being the Dirac delta function.

Other Methods

1. Least Squares Method. We are using the same weighted functions as before. We now select $w_i(x) = \frac{d}{dx}\left(a(x)\frac{d\phi_i(x)}{dx}\right)$ where $\phi_i(x)$ are the same trial functions used to approximate the primary variable.
2. Garlekin Weighted Residuals. Here we choose $w_i(x) = \phi_i(x)$.
3. Other weighted residual methods. The selection of $w_i(x)$ other than the above functions have also been done.

Weak Form

In general, for any chosen set of $w_i(x)$, the approximated integral, $\int_0^L w_i(x)R(x, c_1, \dots, c_N)dx$

Becomes,

$$\int_0^L w_i(x)\left[\frac{d}{dx}\left(a(x)\frac{dU_N(x)}{dx}\right) - f(x)\right]dx$$

$$= \int_0^L a\frac{dw_i}{dx}\frac{dU_N}{dx}dx - \int_0^L w_i f dx - \left[aw_i\frac{dU_N}{dx}\right]_0^L$$

$$= \int_0^L a\frac{dw_i}{dx}\frac{dU_N}{dx}dx - \int_0^L w_i f dx - w_i Q_L + w_i Q_0$$

where we have written $Q \equiv a\frac{dU_N}{dx}$.

Natural Boundary Conditions

* Comparing the above development to the variational formulation of the previous section, we can see immediately that if we take the weighting function as the variation in the primary variables, then the coefficient of this variation becomes the Natural Boundary conditions as before.

* The above is a weak formulation of the differential equation as we have used integration by parts, as before to reduce the differentiation on the primary variable.

Problems with Variational Methods

The variational formulation has an advantage in its demand for a lower differentiation order than the equations resulting from the balance laws resulting from the application of Newton/Cauchy laws of motion.

* The construction of such weak forms, unfortunately, is not always possible for the specific problems under consideration.
* The trial functions to use in the variational approximation needs to satisfy certain conditions: completeness, linear independence, etc.
* They also are required to satisfy both the essential and boundary conditions of the problem.
* These requirements do not allow the variational methods to be viable especially when dealing with geometrically complex domains.

Logic of the FEM

* Despite their apparent appeal, these methods are not even competitive with the regular, time-tested Finite difference schemes as general methods for the approximate solutions for differential equations.
* The Finite Element Method, FEM, overcomes these shortcomings by providing a systematic way of constructing the approximation functions. To become an effective computational method, it contains the following ingredients:

Salient Points: FEM

1. It is able to yield convergent solutions from a sound mathematical and physical basis and hence applicable to practical problems.
2. It is especially powerful in its adaptability to difficult geometrical boundaries. In fact, its provides immediate superiority to FD methods once the boundaries become difficult. Loading, displacement, etcs along irregular boundaries are no longer a serious issue.
3. It allows the increase of the degree of approximation and sensitivity of the solution by simply adding to the number of elements, nodes or order of approximation functions without the requirement of a complete reformulation of the problem.
4. It is easily implementable on a digital computer.

Key Steps in the FEM Process

In FEM, the domain is represented as a collection of simple domains – called finite elements. It is therefore possible to systematically construct approximation functions in a variational or weighted integral approximation to the solution over each simple element. It goes with the 3 steps for construction:

1. Discretization
2. Approximation Functions.
3. Assembly

This procedure offers two degrees of freedom in possible refinements for increased accuracy.

Discretization, Approximation & Assembly

1. **Discretization**. No matter how complex the original domain is, it is discretized into many simple domains.
2. **Approximation Functions**. These are no longer arbitrarily chosen. Instead, they are rooted in approximation and interpolation theory. They are specific and their choice depends on the level of element accuracy desired. The number of element types are usually few.
3. **Assembly.** Assembly returns the problem from the discrete element space to the actual problem space by implementing the actual continuity relationships across connected elements.

Two Degrees

The number of elements to choose; The size reduction of each element space is achieved by increasing this number and consequently increasing the accuracy.
The order of approximation function and increased local accuracy by increasing the number of nodes. Usually there are connecting nodes and there may also be local nodes that are inside the elements themselves depending on the order chosen for the approximation function.

The Interconnection of nodes has the consequence that we are not able to solve the the problem until the assembly stage.

Coefficients & Trial Functions

In specific terms,

* The sum, $c_i\phi_i(x)$ of the undetermined coefficients and trial functions that we arbitrarily choose are replaced by $u_i^e\psi_i^e(x)$ where the superscripts e are referring to the element. The undetermined u_i^e are values of the primary variable at the particular location while $\psi_i^e(x)$ for each i is an interpolation function.
* The choice of u_i^e over an arbitrary undetermined parameter offer the pragmatic advantage that the continuity condition across the boundary of the element can now be imposed automatically.
* After assembly, the coefficients u_i for each element that is, u_i^e are found so that the approximation equation is the residual integral minimized in the weighted residual sense.

Interpolation Functions

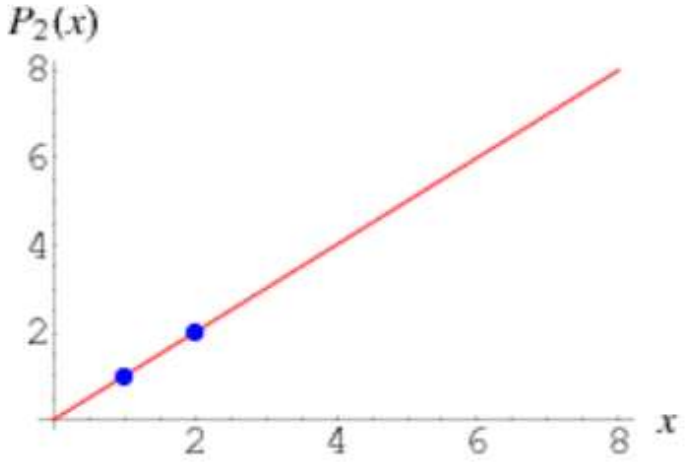
The simplest interpolation fuctions we can choose from are due to Lagrange. The Lagrange Interpolating polynomial $P(x)$ of degree $n-1$ That passes through the points,

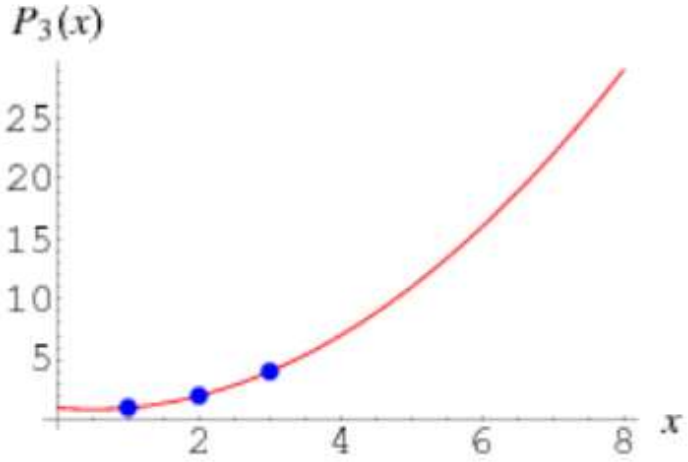
$$\big((x_1, f(x_1)), (x_2, f(x_2), \dots, (x_n, f(x_n)\big)$$

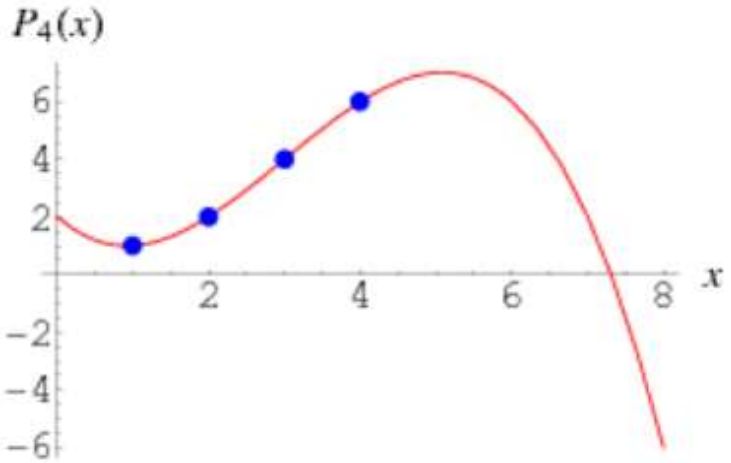
it is given by

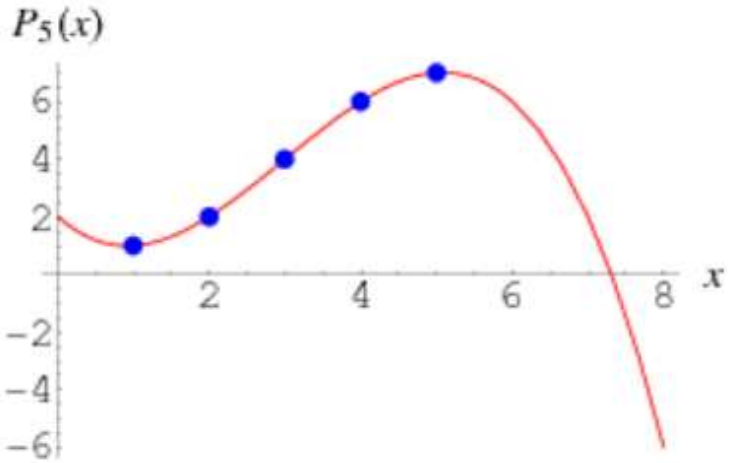
$$P(x) = \sum_{j=1}^{n} P_j(x) \quad \text{where} \quad P_j(x) = f(x_j) \prod_{k=1, k\neq j}^{n} \frac{x - x_k}{x_j - x_k}$$

Lagrangian Functions









Linear Element Interpolation

In a linear element in the domain $[x_1, x_2]$we take the initial point as our local origin and assume the length of the element is $h = x_2 - x_1$so that the point $x_1 = 0$, and $x_2 = h$. We can write the interpolating function over the element as,

$$P(x) = \sum_{j=1}^{n} P_j(x) = \sum_{i=1}^{2} u_i^e \psi_i^e(x)$$

Linear Interpolation

Clearly,

$$P_1(x) = u_1^e \frac{x - x_2}{x_1 - x_2} = u_1^e \left(1 - \frac{x}{h}\right)$$

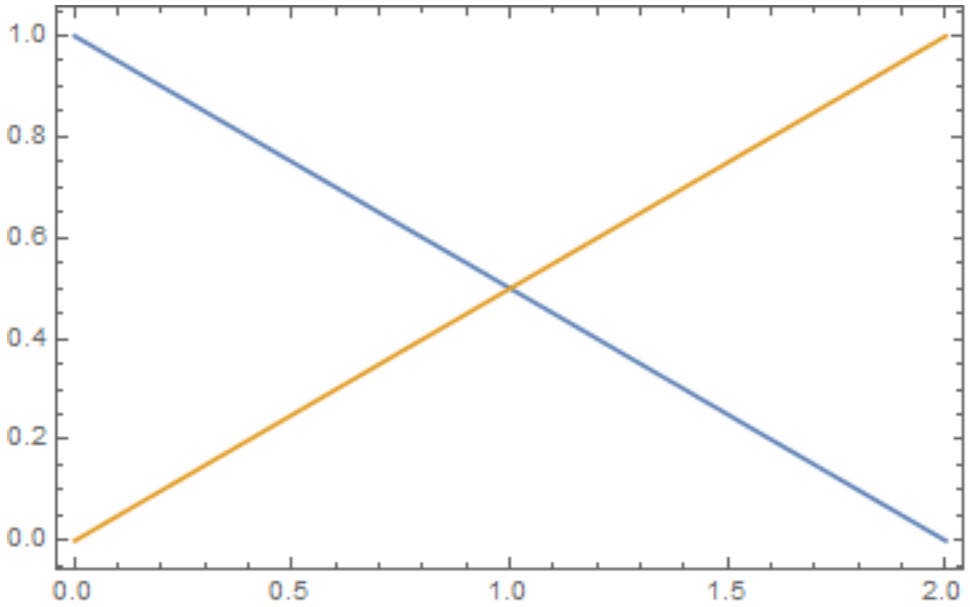
$$\text{and } P_2(x) = u_2^e \frac{x - x_1}{x_2 - x_1} = u_2^e \frac{x}{h}$$

$$\text{so that, } \quad \psi_1^e(x) = 1 - \frac{x}{h} \text{ and } \psi_2^e(x) = \frac{x}{h}$$

And $P(x) = P_1(x) + P_2(x) \rightarrow P(0) = u_1^e$ and $P(h) = u_2^e$ producing the primary variable estimate at each node.

Linear Interpolation

In the Mathematica Graph below, we have allowed $h = 2$



Quadratic Interpolation

For a second-order Lagrangian interpolation, we have three points in order as x_1, x_2, x_3. Let $h = x_3 - x_1$ so that the point $x_1 = 0$, and $x_2 = \alpha h$ and $x_3 = h$.

Clearly, $x_2 - x_1 = \alpha h$; $x_3 - x_2 = (1 - \alpha)h$ and $x_3 - x_1 = h$. Consequently,

$$P_1(x) = u_1^e \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} = u_1^e \left(1 - \frac{x}{h}\right)\left(1 - \frac{x}{\alpha h}\right)$$

Quadratic Interpolation

$$P_2(x) = u_2^e \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}$$

$$= u_2^e \frac{x(x-h)}{(\alpha h)h(\alpha-1)} = \frac{u_2^e}{\alpha(1-\alpha)} \frac{x}{h}\left(1-\frac{x}{h}\right)$$

$$\text{and } P_3(x) = u_3^e \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

$$= u_3^e \frac{x(x-\alpha h)}{h(h-\alpha h)} = \frac{u_3^e}{1-\alpha} \frac{x}{h}\left(\frac{x}{h}-\alpha\right)$$

Quadratic Interpolatioin

It follows that

$$P(0) = \sum_{i=1}^{2} u_i^e \psi_i^e(0) = u_1^e(1-0)(1-0) = u_1^e,$$

$$P(\alpha h) = \sum_{i=1}^{2} u_i^e \psi_i^e(\alpha h) = \frac{u_2^e}{\alpha(1-\alpha)} \frac{\alpha h}{h}\left(1 - \frac{\alpha h}{h}\right) = u_2^e \text{ and}$$

$$P(h) = \sum_{i=1}^{2} u_i^e \psi_i^e(\alpha h) = \frac{u_3^e}{1-\alpha} \frac{h}{h}\left(\frac{h}{h} - \alpha\right) = u_3^e.$$

So that, again, at each node, our interpolating function produces the value of the primary variable at the respective nodal point.

Quadratic Interpolatioin

Where $\psi_1^e(x) = \left(1 - \frac{x}{h}\right)\left(1 - \frac{x}{\alpha h}\right)$,

$\psi_2^e(x) = \frac{1}{\alpha(1-\alpha)}\frac{x}{h}\left(1 - \frac{x}{h}\right)$ and $\psi_3^e = \frac{1}{1-\alpha}\frac{x}{h}\left(\frac{x}{h} - \alpha\right)$.

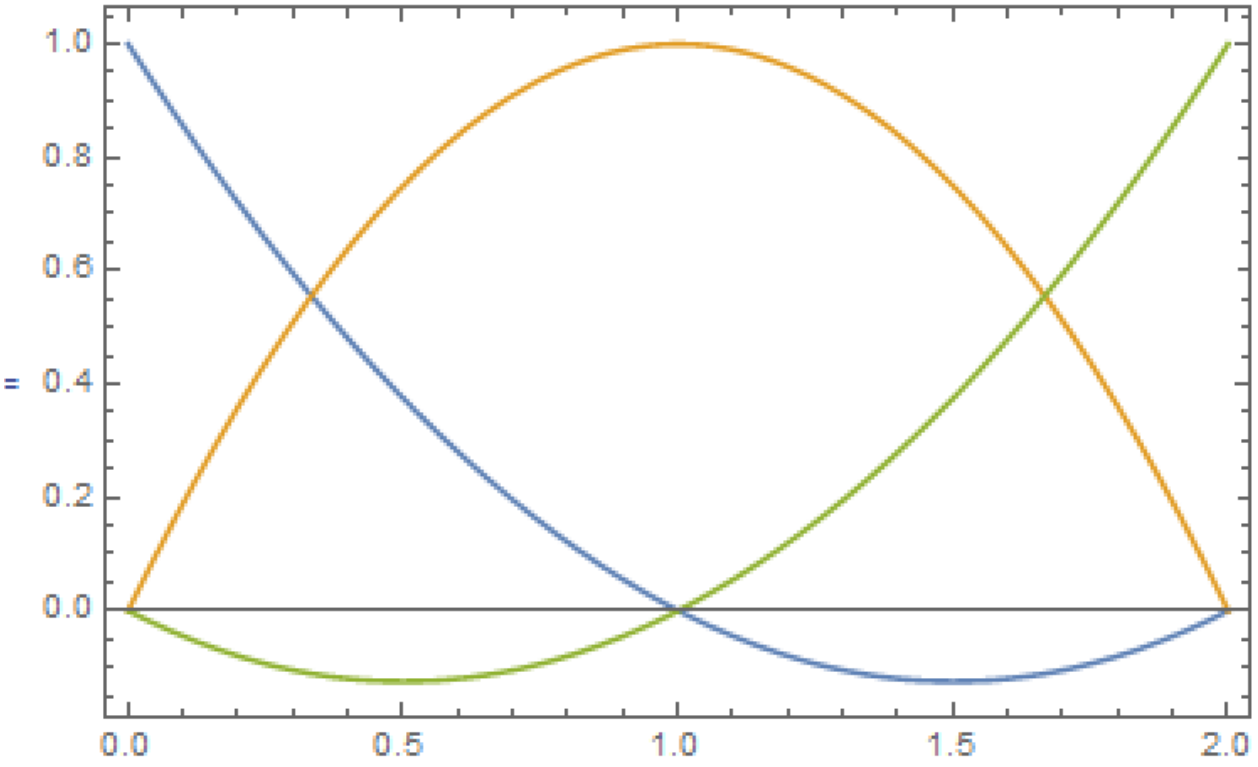
If we set $\alpha = \frac{1}{2}$, we have,

$\psi_1^e(x) = \left(1 - \frac{x}{h}\right)\left(1 - \frac{2x}{h}\right)$,

$\psi_2^e(x) = \frac{4x}{h}\left(1 - \frac{x}{h}\right)$ and $\psi_3^e = \frac{x}{h}\left(\frac{2x}{h} - 1\right)$

Quadratic Interpolation

```
Plot[{(1 - x/2) (1 - 2 x/2), (4 x/2) (1 - x/2),
  -(1 - 2 x/2) (x/2)}, {x, 0, 2}, Frame → True]
```



General Quadratics

The three quadratic Interpolation functions assuming a size $h = 2$, for arbitrary location of the internal node are:

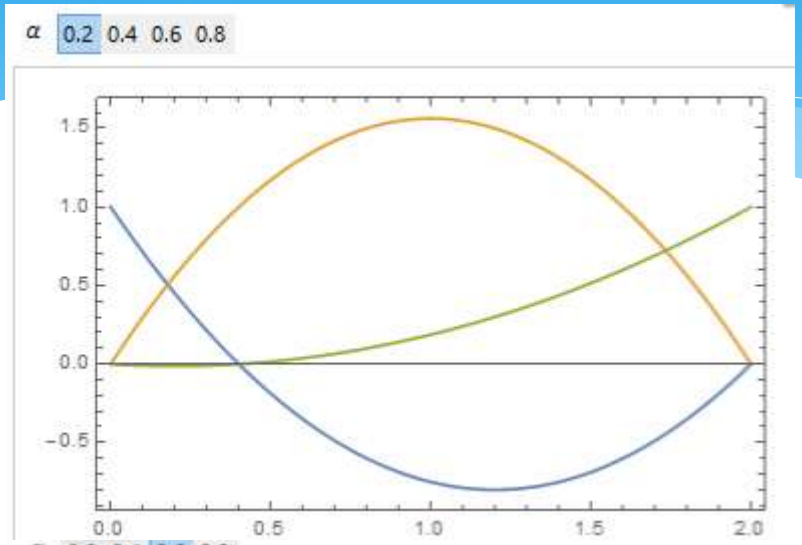
$$\psi_1^e(x) = \left(1 - \frac{x}{2}\right)\left(1 - \frac{x}{2\alpha}\right),$$

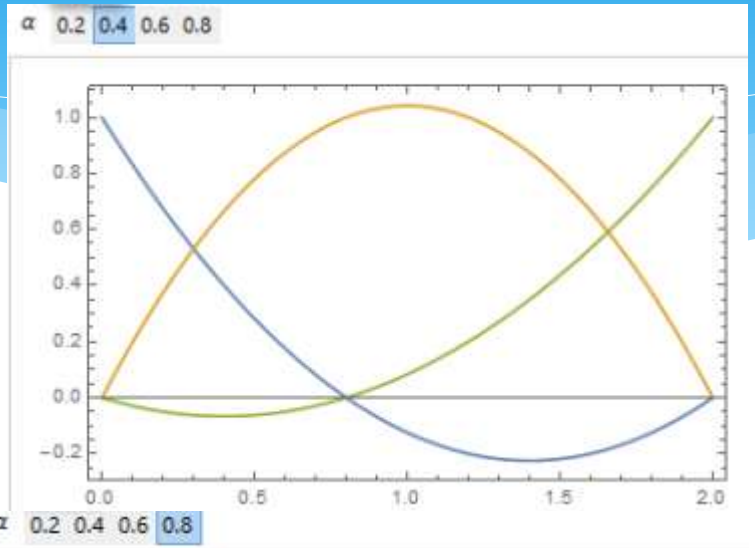
$$\psi_2^e(x) = \frac{1}{\alpha(\alpha - 1)}\frac{x}{2}\left(1 - \frac{x}{2}\right)$$

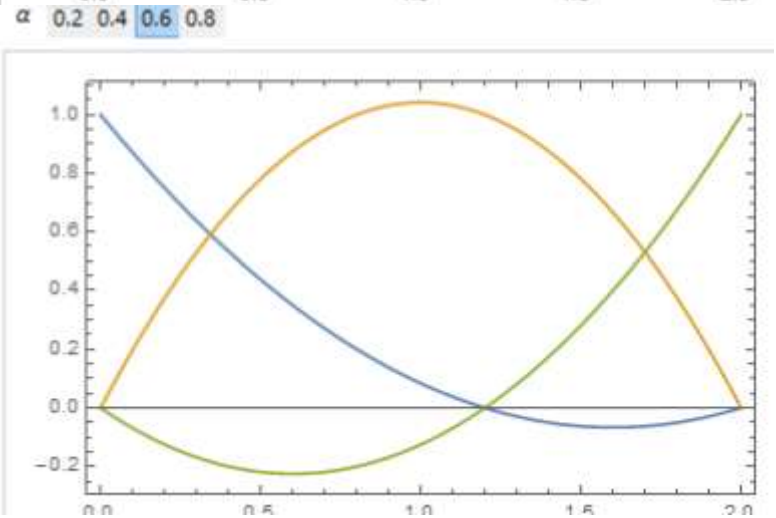
$$\text{and } \psi_3^e(x) = \frac{1}{1 - \alpha}\frac{x}{2}\left(\frac{x}{2} - \alpha\right)$$

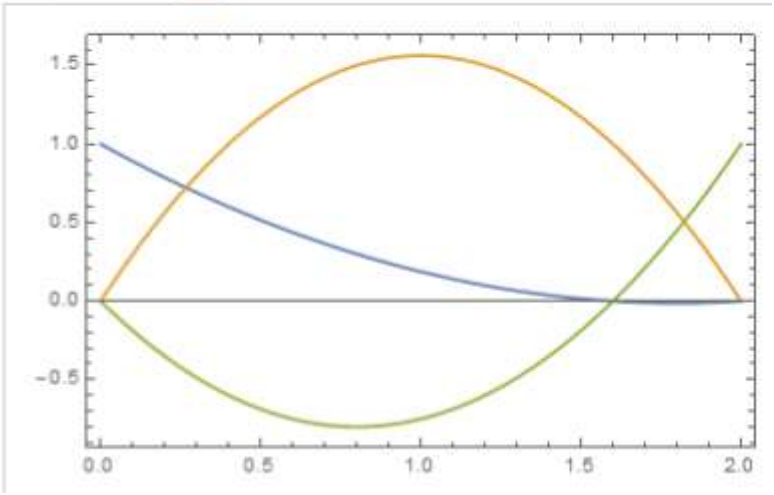
The figure below shows the relative weights at each point for values of $\alpha = 0.2, 0.4, 0.6$ and 0.8.

Discrete Values









The Element Model.

We now formulate the weighted residual equations on the elements.

* The weighted residual integral of interest, from Slide 2.09, is,

$$J = \int_0^L \left(a \frac{dw}{dx} \frac{du}{dx} + cwu - wf \right) dx - w(x_L) Q_L + w(x_0) Q_0$$

where we have written $Q \equiv a \frac{du}{dx}$ is the secondary varuable.

The interval $[0, L]$ is now made of elements $e = 1,2, \dots, n$ occupying the intervals $[x_1^e, x_2^e]$ for each value of the element.

The Element Model…

Hence, we can write,

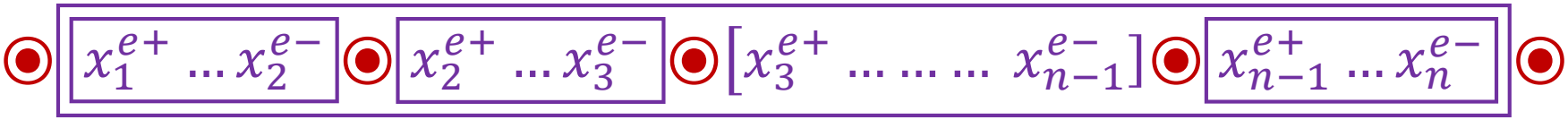
$$J = \int_{x_1^e}^{x_2^e} \left(a \frac{dw}{dx} \frac{du}{dx} + cwu - wf \right) dx - w(x_2^e) Q_2 + w(x_1^e) Q_1$$

For the two-node linear element lying between $[x_1^e, x_2^e]$. In general, an element can have n −nodes. The above element formulation will therefore be,

$$\sum_{i=1}^{n-1} \int_{x_i^e}^{x_{i+1}^e} \left(a \frac{dw}{dx} \frac{du}{dx} + cwu - wf \right) dx - w(x_{i+1}^e) Q_{i+1}^e + w(x_i^e) Q_i^e$$

Interior & Boundary Nodes

Evaluations of Conditions at nodes



The Element Model…

$$= \int_{x_1^e}^{x_n^e} \left(a \frac{dw}{dx}\frac{du}{dx} + cwu - wf \right) dx - w(x_2^{e-})Q_2^{e-} + w(x_1^{e+})Q_1^{e+} - w(x_3^{e-})Q_3^{e-}$$
$$+ w(x_2^{e+})Q_2^{e+} - \cdots - w(x_n^{e-})Q_n^{e-} + w(x_{n-1}^{e+})Q_{n-1}^{e+}$$

$$= \int_{x_1^e}^{x_n^e} \left(a \frac{dw}{dx}\frac{du}{dx} + cwu - wf \right) dx - [-w(x_1^{e+})Q_1^{e+}]$$
$$- [w(x_2^{e-})Q_2^{e-} - w(x_2^{e+})Q_2^{e+}] - [w(x_3^{e-})Q_3^{e-} - w(x_3^{e+})Q_3^{e+}] - \cdots$$
$$\cdot - [w(x_{n-1}^{e-})Q_{n-1}^{e-} - w(x_{n-1}^{e+})Q_{n-1}^{e+}] - [w(x_n^{e-})Q_n^{e-}]$$

$$= \int_{x_1^e}^{x_n^e} \left(a \frac{dw}{dx}\frac{du}{dx} + cwu - wf \right) dx - w(x_1^e)Q_1^e - w(x_2^e)Q_2^e - w(x_3^e)Q_3^e - \cdots$$
$$\cdot - w(x_{n-1}^e)Q_{n-1}^e - w(x_n^e)Q_n^e = 0$$

The 1-D Element

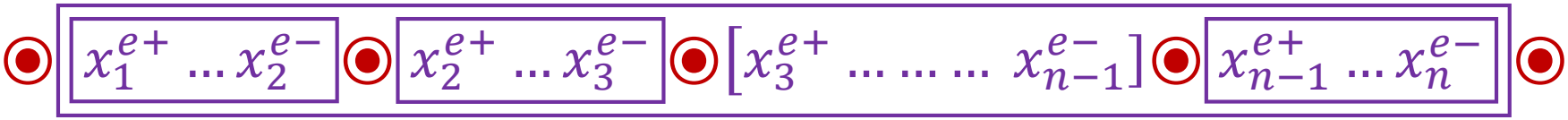
$$\int_{x_1^e}^{x_n^e}\left(a\frac{dw}{dx}\frac{du}{dx}+cwu-wf\right)dx-w(x_1^e)Q_1^e-w(x_2^e)Q_2^e$$
$$-w(x_3^e)Q_3^e-\cdots-w(x_{n-1}^e)Q_{n-1}^e-w(x_n^e)Q_n^e=0$$

where for each nodal condition, we have written,

$$\begin{aligned}
w(x_1^e)Q_1^e &= [-w(x_1^{e+})Q_1^{e+}]\\
w(x_2^e)Q_2^e &= [w(x_2^{e-})Q_2^{e-}-w(x_2^{e+})Q_2^{e+}]\\
w(x_3^e)Q_3^e &= [w(x_3^{e-})Q_3^{e-}-w(x_3^{e+})Q_3^{e+}]\\
w(x_i^e)Q_i^e &= \left[w(x_i^{e-})Q_i^{e-}-w\left(x_i^{e+}\right)Q_i^{e+}\right]\\
w(x_{n-1}^e)Q_{n-1}^e &= [w(x_{n-1}^{e-})Q_{n-1}^{e-}-w(x_{n-1}^{e+})Q_{n-1}^{e+}]\\
w(x_n^e)Q_n^e &= [w(x_n^{e-})Q_n^{e-}]
\end{aligned}$$

Interior & Boundary Nodes

Evaluations of Conditions at nodes



For n —nodes, the appropriate interpolation function is the $n-1^{th}$ polynomial. There will be n of these equations. One for each weight. We write the first and typical ones:

$$\int_{x_a}^{x_b}\left[a\frac{d\psi_1^e}{dx}\left(\sum_{j=1}^{n}u_j^e\frac{d\psi_j^e}{dx}\right)+c\psi_1^e\left(\sum_{j=1}^{n}u_j^e\,\psi_j^e(x)\right)\right.$$

Typical Weight on a Node

The red subscript showing the only thing that now varies from one variable to another. Consequently, the typical equation for the ith weight is,

$$\int_{x_a}^{x_b}\left[a\frac{d\psi_{\color{red}i}^e}{dx}\left(\sum_{j=1}^{n}u_j^e\frac{d\psi_j^e}{dx}\right)+c\psi_{\color{red}i}^e\left(\sum_{j=1}^{n}u_j^e\,\psi_j^e(x)\right)+\psi_{\color{red}i}^e f(x)\right]dx-\sum_{j=1}^{n}\psi_{\color{red}i}^e\left(x_j^e\right)Q_j^e=0$$

and the nth equation is,

$$\int_{x_a}^{x_b}\left[a\frac{d\psi_{\color{red}n}^e}{dx}\left(\sum_{j=1}^{n}u_j^e\frac{d\psi_j^e}{dx}\right)+c\psi_{\color{red}n}^e\left(\sum_{j=1}^{n}u_j^e\,\psi_j^e(x)\right)+\psi_{\color{red}n}^e f(x)\right]dx-\sum_{j=1}^{n}\psi_{\color{red}n}^e\left(x_j^e\right)Q_j^e=0$$

Which we can write as,

$$\sum_{j=1}^{n} K_{ij}^{e} u_j^e - f_i^e - Q_i^e = 0, \qquad i = 1,2,\dots,n$$

where $K_{ij}^{e} = B^e(\psi_i^e, \psi_j^e) = \int_{x_a}^{x_b} \left[a \frac{d\psi_i^e}{dx} \frac{d\psi_j^e}{dx} + c\psi_i^e(x)\psi_j^e(x) \right] dx$

$$f_i^e = \int_{x_a}^{x_b} \psi_i^e f(x) dx$$

and the interpolation property allows us to write

$$\sum_{j=1}^{n} \psi_i^e(x_j^e) Q_j^e = Q_i^e$$

We can now proceed to derive the linear and quadratic elements in 1-D as special cases of this general formula.

Linear 1-D Elements

For the linear element, let $h_e = x_b - x_a$, we have from Slide 2.21 that,

$$\psi_1^e(x) = 1 - \frac{\bar{x}}{h_e} \text{ and } \psi_2^e(x) = \frac{\bar{x}}{h_e}$$

Where we have chosen new coordinates, treating the beginning of the node as the local origin, so that,

$$x = x_1^e + \bar{x} = x_a + \bar{x}$$

Linearity of this relationship means that,

$dx = d\bar{x}$ and $\frac{d\psi_i^e}{dx} = \frac{d\psi_i^e}{d\bar{x}}$

Linear Interpolation

A direct integration of the integral,

$$K_{ij}^e = B^e\left(\psi_i^e, \psi_j^e\right) = \int_{x_a}^{x_b} \left[a \frac{d\psi_i^e}{dx} \frac{d\psi_j^e}{dx} + c\psi_i^e(x)\psi_j^e(x) \right] dx$$

Can be found in the Mathematica notebook, http://1drv.ms/1HMT3c8

Clearly,

$$[K^e] = \frac{a_e}{h_e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \frac{c_e h_e}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

and

$$[f^e] = \frac{f_e h_e}{2} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix}$$

With the Mathematica-based formulation, it is trivial to change the kind of function $a(x)$ can be. The results follow immediately.

Element Computation Code

```
ψ1[x_] := 1 - x/h;
ψ2[x_] := x/h;
K11 = Integrate[a D[ψ1[x], x] D[ψ1[x], x] + c ψ1[x] ψ1[x], {x, 0, h}]
```

$$\frac{a}{h} + \frac{c\,h}{3}$$

```
K12 = Integrate[a D[ψ1[x], x] D[ψ2[x], x] + c ψ1[x] ψ2[x], {x, 0, h}]
```

$$-\frac{a}{h} + \frac{c\,h}{6}$$

```
K22 = Integrate[a D[ψ2[x], x] D[ψ2[x], x] + c ψ2[x] ψ2[x], {x, 0, h}]
```

$$\frac{a}{h} + \frac{c\,h}{3}$$

```
K21 = Integrate[a D[ψ2[x], x] D[ψ1[x], x] + c ψ2[x] ψ1[x], {x, 0, h}]
```

$$-\frac{a}{h} + \frac{c\,h}{6}$$

Homework: Use a linear and a trigonometric function for the input data $a(x)$

1-D Quadratic Element

We proceed as before. The quadratic interpolation functions were derived in Slide 2.26. If we set $\alpha = \frac{1}{2}$, we have,

$$\psi_1^e(x) = \left(1 - \frac{x}{h}\right)\left(1 - \frac{2x}{h}\right),$$

$$\psi_2^e(x) = \frac{4x}{h}\left(1 - \frac{x}{h}\right) \text{ and } \psi_3^e = \frac{x}{h}\left(\frac{2x}{h} - 1\right)$$

Derivation Program

The element coefficient matrix is therefore obtained by a direct integration of the integral,

$$K_{ij}^e = B^e\left(\psi_i^e, \psi_j^e\right)$$
$$= \int_{x_a}^{x_b} \left[a \frac{d\psi_i^e}{dx} \frac{d\psi_j^e}{dx} + c\psi_i^e(x)\psi_j^e(x) \right] dx$$

Can be found in the Mathematica notebook,

Clearly,

$$[K^e] = \frac{a_e}{3h_e} \begin{bmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{bmatrix} + \frac{c_e h_e}{30} \begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix}$$

And

$$[f^e] = \frac{f_e h_e}{6} \begin{Bmatrix} 1 \\ 4 \\ 1 \end{Bmatrix}$$

```
ψ1[x_] := (1 - x/h) (1 - 2 x/h);
ψ2[x_] := 4 (x/h) (1 - x/h);
ψ3[x_] := -(x/h) (1 - 2 x/h);
K11 = Integrate[a D[ψ1[x], x] D[ψ1[x], x]
    + c ψ1[x] ψ1[x], {x, 0, h}]
```

$$\frac{7\,a}{3\,h} + \frac{2\,c\,h}{15}$$

```
K33 = Integrate[a D[ψ3[x], x] D[ψ3[x], x]
    + c ψ3[x] ψ3[x], {x, 0, h}]
```

$$\frac{7\,a}{3\,h} + \frac{2\,c\,h}{15}$$

```
K12 = Integrate[a D[ψ1[x], x] D[ψ2[x], x]
    + c ψ1[x] ψ2[x], {x, 0, h}]
```

$$-\frac{8\,a}{3\,h} + \frac{c\,h}{15}$$

```
K13 = Integrate[a D[ψ1[x], x] D[ψ3[x], x]
    + c ψ1[x] ψ3[x], {x, 0, h}]

K22 = Integrate[a D[ψ2[x], x] D[ψ2[x], x]
    + c ψ2[x] ψ2[x], {x, 0, h}]

K23 = Integrate[a D[ψ2[x], x] D[ψ3[x], x]
    + c ψ2[x] ψ3[x], {x, 0, h}]
```

* Note that these elements are derived on very restrictive conditions. This includes the symmetric element used in the quadratic case, that functions $a(x)$, $f(x)$ are constants. We are not bound by these constraints as we have the power of the Symbolic processor to compute these integrals and sums as we have shown here.

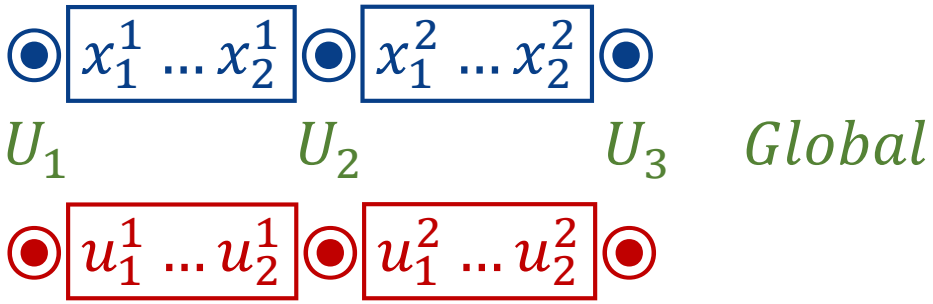
Connecting the Elements

* The next stage is to connect the elements. This is necessary to reduce the number of unknowns and apply the continuity conditions that are quite easy to do and is in fact one of the important advantages of using the FEA approach.
* It is essentially a reversal of the discretization process. It is also a necessary step before the solution of the derived equations can be attempted.

Local and Global Identification

* The most basic issue in assembly is to recognize a global naming system that now translates the identities of the nodes from the local numbering we have used into a consistent global identification system.
* This is relatively straightforward in the case of one-dimensional elements because, for each element, we are dealing with two neighbours at most.
* It is also true that once this step is fully understood in the 1-D case, it is also a straightforward matter to extend it to higher dimensions when we will deal with more neighbours and have more complications.

Simplest Case



In the simplest case, we consider two linear elements so that we have three nodes. Each element has two nodes and they share one node as shown above.

Globalization

* Each element equation is first order and produces two equations: One for each node. For element 1, we have local variables u_1^1 and u_2^1. These are the same as the global variables U_1 and U_2.
* For any element e, at node 1 or 2 we can write:

$$\sum_{j=1}^{2} K_{ij}^e u_j^e = K_{i1}^e u_1^e + K_{i2}^e u_2^e = f_i^e + Q_i^e$$

Element Equations

Consequently, for element 1 we have

$$K_{11}^1 u_1^1 + K_{12}^1 u_2^1 = f_1^1 + Q_1^1$$
$$K_{21}^1 u_1^1 + K_{22}^1 u_2^1 = f_2^1 + Q_2^1$$

And for element 2, the terms remain unchanged except for the element superscripts:

$$K_{11}^2 u_1^2 + K_{12}^2 u_2^2 = f_1^2 + Q_1^2$$
$$K_{21}^2 u_1^2 + K_{22}^2 u_2^2 = f_2^2 + Q_2^2$$

Continuity

* Apply the continuity which is simply regognizing the global numbering for each nodal variable, we have,

$$K_{11}^1 U_1 + K_{12}^1 U_2 = f_1^1 + Q_1^1$$
$$K_{21}^1 U_1 + K_{22}^1 U_2 = f_2^1 + Q_2^1$$

And for element 2, the terms remain unchanged except for the element superscripts:

$$K_{11}^2 U_2 + K_{12}^2 U_3 = f_1^2 + Q_1^2$$
$$K_{21}^2 U_2 + K_{22}^2 U_3 = f_2^2 + Q_2^2$$

Condensing

* We can add the second equation of element 1 to the first of element 2 and obtain,

$$K_{11}^1 U_1 + K_{12}^1 U_2 = f_1^1 + Q_1^1$$
$$K_{21}^1 U_1 + K_{22}^1 U_2 + K_{11}^2 U_2 + K_{12}^2 U_3 = f_2^1 + Q_2^1 + f_1^2 + Q_1^2$$
$$K_{21}^2 U_2 + K_{22}^2 U_3 = f_2^2 + Q_2^2$$

Or,

$$\begin{bmatrix} K_{11}^1 & K_{12}^1 & 0 \\ K_{21}^1 & K_{22}^1 + K_{11}^2 & K_{12}^2 \\ 0 & K_{21}^2 & K_{22}^2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} f_1^1 \\ f_2^1 + f_1^2 \\ f_2^2 \end{bmatrix} + \begin{bmatrix} Q_1^1 \\ Q_2^1 + Q_1^2 \\ Q_2^2 \end{bmatrix}$$

N-Element Case

* A Simple inspection extends this to the case of N linear elements as can be seen on page 129 in the text.