Chapter 4

Isoparametric Elements
Today’s Lecture Contents

• The Shape Function: Reminder from the previous lecture

• Looking for a uniform mapping across element types

• Isoparametric elements

• 1D Demonstration: Bar elements
The Galerkin Method

In the previous lecture, we defined an approximation for the displacement field:

\[ u = u_1 + u_2x = \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \]

This also defined the first derivative of the approximation (strain):

\[ \frac{du}{dx} = \frac{d}{dx} \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \]
The Galerkin Method

where $u_1, u_2$ are so far random coefficients. Instead, we can choose to write the same relationship using a different basis $N(x)$:

$$\{u\} = [N(x)]\{d\}$$

In this case we express $u$ in terms of the degrees of freedom, i.e., the displacements at the ends of the bar:

$$\{d\} = \begin{pmatrix} u(x = 0) \\ u(x = L) \end{pmatrix} = \begin{pmatrix} u_0 \\ u_L \end{pmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \Leftrightarrow \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix}^{-1} \begin{pmatrix} u_0 \\ u_L \end{pmatrix}$$

Substituting in our initial displacement approximation we obtain:

$$u = \begin{bmatrix} 1 & x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & L \end{bmatrix}^{-1} \begin{pmatrix} u_0 \\ u_L \end{pmatrix} = \begin{bmatrix} L-x & x \end{bmatrix} \begin{pmatrix} u_0 \\ u_L \end{pmatrix} \Rightarrow N(x) = \begin{bmatrix} L-x & x \\ L & L \end{bmatrix}$$
The Galerkin Method

Then:

\[ u = [N(x)] \{d\} \]

Displacement field

The weak form also involves the first derivative of the approximation

\[ \varepsilon = \frac{d[N(x)]}{dx} \{d\} = [B]\{d\} \]

Vector of degrees of freedom at the element nodes

Strain field

where

\[ [B] = \frac{d[N(x)]}{dx} = \frac{1}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \]

Strain Displacement Matrix
The Galerkin Method

Therefore if we return to the weak form:

\[ \int_0^L \frac{dw}{dx} EA \frac{du}{dx} \, dx = - \int_0^L w(x) (-\alpha x) \, dx \]

and set:

\[ u = [N] \{d\} \quad \varepsilon = [B] \{d\} \quad w = [N] \{\tilde{w}\} \]

The following FUNDAMENTAL FEM expression is derived:

\[ \{\tilde{w}\}^T \int_0^L [B]^T EA [B] \, dx \{d\} = \{\tilde{w}\}^T \int_0^L [N]^T (-\alpha x) \, dx \]

or even better:

\[ \int_0^L [B]^T EA [B] \, dx \{d\} = \int_0^L [N]^T (-\alpha x) \, dx \]
The Galerkin Method

\[ \int_{0}^{L} [B]^T E A [B] \, dx \{d\} = \int_{0}^{L} [N]^T (-\alpha x) \, dx \]

*EA* has to do only with material and cross-sectional properties.

We call \([k] = \int_{0}^{L} [B]^T E A [B] \, dx \{d\}\) The Finite Element stiffness Matrix.

*If* \( E \) is a function of \( \{d\} \)

*Material Nonlinearity*

*If* \([B]\) is a function of \( \{d\} \)

*Geometrical Nonlinearity*
The Galerkin Method

Indeed, if we use the proposed formulation for $[N], [B]$: 

$$
[k] = \int_0^L [B]^T EA[B] \, dx \\
[B] = \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \\
[k] = \int_0^L \frac{1}{L} \begin{bmatrix} -1 \\ 1 \end{bmatrix} EA \frac{1}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \, dx \Rightarrow
$$

$$
= \frac{1}{L^2} \begin{bmatrix} -1 \\ 1 \end{bmatrix} EA \begin{bmatrix} -1 & 1 \end{bmatrix} L \Rightarrow
$$

$$
[k] = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
$$

bar stiffness!
The Shape Function

Therefore: **Shape functions** will be defined as **interpolation functions** which relate the variables in the finite element with their values in the element nodes. The latter are obtained through solving the problem using finite element procedures.

In general we may write:

\[ u(x) = \sum_i N_i(x) u_i = Nu \]

where:

- \( u(x) \) is the function under investigation (for example: displacement field)
- \( N(x) \) is the shape functions matrix
- \( u = \begin{bmatrix} u_1 & u_2 & \ldots & u_N \end{bmatrix}^T \) is the vector of unknowns in the nodes (for example: displacements)
The Shape Function

Regardless of the dimension of the element used, we have to bear in mind that Shape Functions need to satisfy the following constraints:

• $N_i(x)$ in node $i$ has a value of 1 and in all other nodes assumes a value of 0.
• Furthermore we have to satisfy the continuity between the adjoining elements.

Example: rectangular element with 6 nodes
The Shape Function

Usually, polynomial functions are used as interpolation functions, for example:

$$P_n(x) = \sum_{i=0}^{N} a_i x^i$$

where $n$ is the order of the polynomial; is equal to the number of unknowns in the nodes (degrees of freedom).

In the MFE we use three different polynomials:
- Lagrange
- Serendipity and
- Hermitian polynomials.
The Shape Function

Lagrange Polynomials

A function \( \varphi(x) \) can be approximated by a polynomial of the order \( m \) and the values of \( \varphi(x) \) in those \( m+1 \) points:

\[
\overline{\varphi}(x) \approx \varphi(x) = \sum_{i=1}^{m+1} L_i(x) \phi^i
\]

where:

\[
L_i(x) = \prod_{\substack{j=1 \atop j \neq i}}^{m+1} \frac{x - x_j}{x_i - x_j} = \frac{(x - x_1)(x - x_2)\ldots(x - x_{m+1})}{(x_i - x_1)(x_i - x_2)\ldots(x_i - x_{m+1})}
\]
Lagrange Polynomials

The Shape Function

A function $\phi(x)$ can be approximated by a polynomial of the order $m$ and the values of $\phi(x)$ in those $m + 1$ points:

$L_i(x)$ - Lagrange polynomials

\[
\begin{align*}
L_i(x) &= 1 \quad x = x_i \\
L_i(x) &= 0 \quad x = x_j, \quad i \neq j
\end{align*}
\]
Lagrange Polynomials – 2D Example

Let us break the process into steps:

1. Interpolate along borders 1-2, 4-3:
   \[ \phi_{1-2} = L_{1-2}^1(x) \phi_1 + L_{1-2}^2(x) \phi_2 \]
   \[ \phi_{3-4} = L_{1-3}^4(x) \phi_3 + L_{2-3}^4(x) \phi_4 \]

Therefore:

\[ L_{1-2}^1(x) = \frac{x_2 - x}{x_2 - x_1}, \quad L_{1-2}^2(x) = \frac{x - x_1}{x_2 - x_1} \]

\[ L_{1-3}^4(x) = \frac{x_4 - x}{x_4 - x_3}, \quad L_{2-3}^4(x) = \frac{x - x_3}{x_4 - x_3} \]
Lagrange Polynomials – 2D Example

\[ \phi_{1-2} = \frac{x_2 - x}{x_2 - x_1} \phi_1 + \frac{x - x_1}{x_2 - x_1} \phi_2 \]

\[ \phi_{3-4} = \frac{x_4 - x}{x_4 - x_3} \phi_3 + \frac{x - x_3}{x_4 - x_3} \phi_4 \]

2. Interpolate along borders 1-2, 4-3: (one possible way is below)

\[ \phi = L_1(y) \phi_{1-2} + L_2(y) \phi_{3-4} = \]

\[ = L_1^{1-4}(y) L_1^{1-2}(x) \phi_1 + L_1^{2-3}(y) L_2^{1-2}(x) \phi_2 + L_2^{2-3}(y) L_1^{4-3}(x) \phi_3 + L_2^{1-4}(y) L_2^{4-3}(x) \phi_4 \]
To interpolate the displacement field for this 2D element, using the horizontal \( u_i \) and vertical displacements \( v_i \) at each one of the 4 nodes \( (i) \) of this element, we can use the following approximation:

\[
\begin{align*}
    u(x, y) &= \frac{(x_2 - x)(y_4 - y)}{(x_2 - x_1)(y_4 - y_1)} u_1 + \frac{(x - x_1)(y_3 - y)}{(x_2 - x_1)(y_3 - y_2)} u_2 + \frac{(x_4 - x)(y - y_2)}{(x_4 - x_3)(y_3 - y_2)} u_3 + \frac{(x - x_3)(y - y_1)}{(x_4 - x_3)(y_4 - y_1)} u_4 \\
    v(x, y) &= \frac{(x_2 - x)(y_4 - y)}{(x_2 - x_1)(y_4 - y_1)} v_1 + \frac{(x - x_1)(y_3 - y)}{(x_2 - x_1)(y_3 - y_2)} v_2 + \frac{(x_4 - x)(y - y_2)}{(x_4 - x_3)(y_3 - y_2)} v_3 + \frac{(x - x_3)(y - y_1)}{(x_4 - x_3)(y_4 - y_1)} v_4
\end{align*}
\]
Lagrange Polynomials – 2D Example

The polynomials we used in this simple example are of order 1

If polynomials of higher order are to be used, further nodes should be typically introduced within the elements.

In fact, in the 2D domain a simple geometrical rule, based on Pascal’s triangle defines how many nodes are required for the representation of displacement fields of any order, according to the type of element used.
Pascal’s Triangle – 2D Example

For triangular elements:

\[ n = \frac{1}{2} (p + 1)(p + 2) \]
Pascal’s Triangle – 2D Example

For square elements & Lagrange Polynomials (which are complete:

Polynomial terms can be chosen

Polynomial Degree  Element

1

2

3

4

No. of Nodes = \( n = (p + 1)^2 \)

©PM Moite, IITK
Serendipity Polynomials

These functions are similar to Lagrangian polynomials, but for their incompleteness (missing terms). Due to this fact we do not need to introduce additional inner nodes, as for Lagrangian polynomials of higher order.

Lagrange Polynomials

Serendipity Polynomials
Pascal’s Triangle – 2D Example

For square elements & serendipity Polynomials (which are incomplete:)

Polynomial terms can be chosen

Polynomial Degree | Element
---|---
1 | 
2 | 
3 | 
4 | 

©PM Moite, IITK
Hermitian Polynomials

Lagrangian polynomials and serendipity functions provide a $C^0$ continuity. If we additionally need continuity of the first derivatives between the finite elements we use **Hermitian** polynomials.

A Hermitian polynomial of the order $n$, $H^n(x)$, is a $2n+1$ order polynomial. For example a Hermitian polynomial of the first order is actually a third order polynomial.

Let us consider a bar element with nodes on its ends. Unknowns are values of the function $\varphi$ in the nodes 1 and 2, $\varphi_1$ and $\varphi_2$, and first derivatives of $\varphi$ in respect to $x$, $\varphi_{1,x}$ and $\varphi_{2,x}$.

$\varphi_{1,x} = \frac{d\varphi(x)}{dx} \bigg|_{x=x_1}$, $\varphi_{2,x} = \frac{d\varphi(x)}{dx} \bigg|_{x=x_2}$

**Remember:** The 1st derivative of displacement, corresponds to rotation.
Hermitian Polynomials

Hermitian shape functions relate not only the displacements at nodes to displacements within the elements but also to the first order derivatives (e.g. rotational DOFs for a beam element).

Shape function of \( u(x) \):

\[
\begin{align*}
& u(x) = \sum_{i=1}^{2} N_{0i}(x) u_i + N_{1i}(x) \frac{\partial u_i(x)}{\partial x} \\
& N_{0i}(x) = 1 \text{ at node } i \text{ and } 0 \text{ at other nodes} \\
& N_{0i}^\prime(x) = 0 \text{ at all nodes} \\
& N_{1i}(x) = 0 \text{ at all nodes} \\
& N_{1i}^\prime(x) = 1 \text{ at node } i \text{ and } 0 \text{ at other nodes}
\end{align*}
\]

Shape function of the derivative \( u'(x) \).
Isoparametric Elements

In general, we would like to be able to represent any element in a standardized manner – introducing a transformation between a set of standardized (natural) coordinates and the real (global) coordinates.

Different schemes exist for establishing such transformations:

1. **sub parametric representations** (less nodes for geometric than for displacement representation)

2. **isoparametric representations** (same nodes for both geometry and displacement representation)

3. **super parametric representations** (more nodes for geometric than for displacement representation)
Isoparametric Elements

Displacement fields as well as the geometrical representation of the finite elements are approximated using the same approximating functions – shape functions.

This transformation allows us to refer to similar elements (e.g., truss, beams, 2D elements, in a standard manner, using the natural coordinates \((r,s)\), without every time referring to the specific global coordinate system \((x,y)\).
Isoparametric Elements

Isoparametric elements can be one-, two- or three-dimensional:

Geometric interpolation

\[ x = \sum_{i=1}^{n} h_i x_i ; \quad y = \sum_{i=1}^{n} h_i y_i ; \quad z = \sum_{i=1}^{n} h_i z_i \]

Displacement Interpolation

\[ u = \sum_{i=1}^{n} h_i \hat{u}_i ; \quad v = \sum_{i=1}^{n} h_i \hat{v}_i ; \quad w = \sum_{i=1}^{n} h_i \hat{w}_i \]

The principle is to assure that the value of the shape function \( h_i \) is equal to one in node \( i \) and equals zero in other nodes.
Isoparametric Elements

In order to establish the stiffness matrixes we must differentiate the displacements with respect to the coordinates $(x, y, z)$.

Since the shape functions are usually defined in natural coordinates we must introduce the necessary coordinate transformation between natural and global or local coordinate system. For example in the 3D domain we can express the derivative of a function $\phi$, which is expressed in terms of coordinates $(x,y,z)$, with respect to another set of coordinates $(r,s,t)$ as:

\[
\frac{\partial \phi}{\partial r} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial r} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial r}
\]

\[
\frac{\partial \phi}{\partial s} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial s} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial s} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial s}
\]

\[
\frac{\partial \phi}{\partial t} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial t}
\]

Chain rule of differentiation!
Isoparametric Elements

Written in matrix notation:

\[
\begin{align*}
\frac{\partial \phi}{\partial r} &= \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial r} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial r} \\
\frac{\partial \phi}{\partial s} &= \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial s} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial s} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial s} \\
\frac{\partial \phi}{\partial t} &= \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial t}
\end{align*}
\]

\[
\begin{bmatrix}
\frac{\partial \phi}{\partial r} \\
\frac{\partial \phi}{\partial s} \\
\frac{\partial \phi}{\partial t}
\end{bmatrix} = 
\begin{bmatrix}
\frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} & \frac{\partial z}{\partial r} \\
\frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} & \frac{\partial z}{\partial s} \\
\frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} & \frac{\partial z}{\partial t}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial \phi}{\partial x} \\
\frac{\partial \phi}{\partial y} \\
\frac{\partial \phi}{\partial z}
\end{bmatrix}
\]
**Isoparametric Elements**

We recall the Jacobian operator $J$:

\[
\begin{bmatrix}
\frac{\partial \phi}{\partial r} \\
\frac{\partial \phi}{\partial s} \\
\frac{\partial \phi}{\partial t}
\end{bmatrix} =
\begin{bmatrix}
\frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} & \frac{\partial z}{\partial r} \\
\frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} & \frac{\partial z}{\partial s} \\
\frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} & \frac{\partial z}{\partial t}
\end{bmatrix}
\begin{bmatrix}
\frac{\partial \phi}{\partial x} \\
\frac{\partial \phi}{\partial y} \\
\frac{\partial \phi}{\partial z}
\end{bmatrix}
\]

\[
\frac{\partial}{\partial r} = J \frac{\partial}{\partial x} \quad \Rightarrow \quad \frac{\partial}{\partial x} = J^{-1} \frac{\partial}{\partial r}
\]
Consider the bar element with two end nodes at points $x_1, x_2$ defined on the Cartesian axis $x$.

and now consider the following standard truss element defined on the natural axis $r$

Is there a function $M$ that can map each point on the $x$ axis to a point on the $r$ axis, so that:

1. $M(x_1) = -1$
2. $M(x_2) = 1$
Bar Element

The relation between the $x$-coordinate and the $r$-coordinate is given as:

$$x = \frac{1}{2} (1 - r)x_1 + \frac{1}{2} (1 + r)x_2$$

$$= \sum_{i=1}^{2} N_i (r) x_i$$

The relation between the displacement $u$ and the nodal displacements are defined in the same way:

$$u(r) = \frac{1}{2} (1 - r)\hat{u}_1 + \frac{1}{2} (1 + r)\hat{u}_2$$

$$= \sum_{i=1}^{2} N_i (r) \hat{u}_i$$
Isoparametric Elements

Let us now consider the derivation of the stiffness matrix $K$. Firstly, we write for the strain matrix (using matrix $B$):

$$
\varepsilon = B\hat{u}
$$

and then we can write up the integrals for calculating the stiffness matrix:

$$
K = \int_V B^T E A B \, dV
$$

$$
= \int_V B^T E A B \det J \, dr \, ds \, dt
$$
Bar Element

We need to be able to establish the strains – meaning we need to be able to take the derivatives of the displacement field in regard to the x-coordinate

$$\varepsilon = \frac{du}{dx} = \frac{du}{dr} \frac{dr}{dx}$$

$$\frac{du}{dr} = \frac{d}{dr} \left( \frac{1}{2} (1-r)\hat{u}_1 + \frac{1}{2} (1+r)\hat{u}_2 \right) = \frac{1}{2} (\hat{u}_2 - \hat{u}_1)$$

$$\frac{dx}{dr} = \frac{d}{dr} \left( \frac{1}{2} (1-r)x_1 + \frac{1}{2} (1+r)x_2 \right) = \frac{1}{2} (x_2 - x_1) = \frac{L}{2} = J$$

$$\varepsilon = \frac{du}{dx} = \frac{du}{dr} \frac{dr}{dx} = \frac{\hat{u}_2 - \hat{u}_1}{x_2 - x_1} = \frac{\hat{u}_2 - \hat{u}_1}{L} = \frac{1}{L} \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} \hat{u}_2 \\ \hat{u}_1 \end{bmatrix}$$

$$\frac{dx}{dr} = J^{-1} = \frac{2}{L}$$
Bar Element

The strain-displacement matrix then becomes:

\[ \mathbf{B} = \frac{1}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \] since \( \varepsilon = \mathbf{B} \begin{bmatrix} \hat{u}_2 \\ \hat{u}_1 \end{bmatrix} \)

or this also occurs as

\[
\mathbf{B} = \begin{bmatrix} \frac{\partial N(\hat{x})}{\partial \hat{x}} & \frac{\partial N(\hat{r})}{\partial \hat{r}} \end{bmatrix} = \frac{d}{dr} \left( \frac{1}{2} \begin{bmatrix} 1-r & 1+r \end{bmatrix} \right) J^{-1} = \frac{J_{L/2}^L}{L} \begin{bmatrix} -1 & 1 \end{bmatrix}
\]

and the stiffness matrix is calculated as:

\[
\mathbf{K} = \frac{AE}{L^2} \int_{-1}^{1} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \mathbf{J} \mathbf{d}r, \quad \mathbf{J} = \frac{dx}{dr} = \frac{L}{2}
\]

\[
\mathbf{K} = \frac{AE}{L^2} \left[ \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} r \right]_{-1}^{1} \Rightarrow \mathbf{K} = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
\]
Quadrature

\[ K = \frac{AE}{L^2} \int_{-1}^{1} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix} Jdr \]

How can we calculate such integrals in an automated manner for higher order elements and higher dimensions?

Demo following next