Numerical Methods in Engineering

Interpolation
(Shape Functions)
Interpolation Problem 1D

Let’s assume that we have measured an unknown magnitude \( u = u(x) \) on a set of data points \( x = [x_1, x_2, \ldots, x_N] \) obtaining a set of values \( u = [u_1, u_2, \ldots, u_N] \)
Let’s assume that we have measured an unknown magnitude $u = u(x)$ on a set of data points $x = [x_1, x_2, ..., x_N]$ obtaining a set of values $u = [u_1, u_2, ..., u_N]$. How can we guess the value of a new point?
Interpolation Problem 1D

Let’s assume that we have **measured** an unknown magnitude \( u = u(x) \) on a set of data points \( x = [x_1, x_2, ..., x_N] \) obtaining a set of values \( u = [u_1, u_2, ..., u_N] \)

Actual function

\[
f(x) = x^3 - 3x^2 - 3\sin(5x)
\]
Interpolation Problem 1D

To **Interpolate** a set of data measured points \((x_i, u_i)\) means to build a function \(P_n(x)\) (usually a **polynomial**) passing through these points. That is

\[
u_i = P_n(x_i), \ i = 1 \ldots N
\]

**Theorem:**
The interpolation problem **has a unique** solution when \(n = N - 1\).

**Idea of the proof:**

\[
P_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n \quad \rightarrow \quad n + 1 \equiv N \text{ unknowns}
\]

\[
u_i = P_n(x_i), \ i = 1 \ldots N \quad \rightarrow \quad N \text{ linear equations}
\]

**Unique** solution if the system is compatible determined. That means the interpolation polynomial has **degree one less** than the number of points □

© Numerical Factory
Shape Functions 1D
Shape Functions 1D

Lagrange Polynomials (Shape functions)

One (N-1) degree Polynomial for each measure point $x_i, \ i = 1, ..., N$

$$\psi_i(x) = \frac{(x - x_1)(x - x_2) \ldots (x - x_i) \ldots (x - x_{N-1})(x - x_N)}{(x_i - x_1)(x_i - x_2) \ldots (x_i - x_i) \ldots (x_i - x_{N-1})(x_i - x_N)}$$

Properties:

1. Lagrange Polynomials are a base of the set $\mathbb{P}_n[x]$ 
2. $\psi_i(x_j) = 0 \quad \text{if} \quad i \neq j$ 
3. $\psi_i(x_i) = 1$

Interpolation Polynomial (a linear combination of the shape functions)

$$P_n(x) = u_1\psi_1(x) + u_2\psi_2(x) + \ldots + u_N\psi_N(x)$$
Shape Functions 1D

N = 2 only two points: \( x = [x_1, x_2] \) and their measures \( u = [u_1, u_2] \)

\[
\psi_1(x) = \frac{(x - x_2)}{(x_1 - x_2)}
\]

\( x = [1, 2] \)

It is a straight line
Shape Functions 1D

N = 2 only two points: \( x = [x_1, x_2] \) and their measures \( u = [u_1, u_2] \)

\[
\psi_1(x) = \frac{(x - x_2)}{(x_1 - x_2)}
\]

\[
\psi_2(x) = \frac{(x - x_1)}{(x_2 - x_1)}
\]

\[ x = [1, 2] \]
Shape Functions 1D

\( N = 2 \) only two points: \( x = [x_1, x_2] \) and their measures \( u = [u_1, u_2] \)

\[
\psi_1(x) = \frac{(x - x_2)}{(x_1 - x_2)}
\]

\[
\psi_2(x) = \frac{(x - x_1)}{(x_2 - x_1)}
\]

Interpolation Polynomial

\[
P_1(x) = u_1 \psi_1(x) + u_2 \psi_2(x)
\]

It is also a straight line (linear interpolation)
Shape Functions 1D

N = 2  only two points: \( x = [x_1, x_2] \) and their measures \( u = [u_1, u_2] \)

\[
\psi_1(x) = \frac{(x - x_2)}{(x_1 - x_2)}
\]

\[
\psi_2(x) = \frac{(x - x_1)}{(x_2 - x_1)}
\]

Interpolation Polynomial

\[
P_1(x) = u_1 \psi_1(x) + u_2 \psi_2(x)
\]

Example:
In the case \( x = [1, 2] \) and \( u = [0.7, 1.8] \)

\[
\psi_1(x) = \frac{(x - 2)}{(1 - 2)} = -(x - 2)
\]

\[
\psi_2(x) = \frac{(x - 1)}{(2 - 1)} = (x - 1)
\]

How to approximate the value for \( x = 1.5 \)?

\[
P_1(1.5) = 0.7\psi_1(1.5) + 1.8\psi_2(1.5)
\]

\[
= 0.7 \times 0.5 + 1.8 \times 0.5 = 1.25
\]
Shape Functions 1D

N = 3  three points: \( x = [x_1, x_2, x_3] \) and measures \( u = [u_1, u_2, u_3] \)

\[
\psi_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}
\]

\[
\psi_2(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)}
\]

\[
\psi_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}
\]
Shape Functions 1D

\( N = 3 \) three points: \( x = [x_1, x_2, x_3] \) and measures \( u = [u_1, u_2, u_3] \)

\[
\psi_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} \\
\psi_2(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} \\
\psi_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}
\]

Interpolation Polynomial

\[
P_2(x) = u_1 \psi_1(x) + u_2 \psi_2(x) + u_3 \psi_3(x)
\]

Now it is a parabola (quadratic interpolation)
Shape Functions 1D

Example:

Consider $x = [-1, 0, 1]$ and $u = [1.2, 0.7, 1.8]$. Compute $u(0.2)$?

\[
\psi_1(x) = \frac{(x - 0)(x - 1)}{(-1 - 0)(-1 - 1)} = \frac{1}{2} x(x - 1)
\]

\[
\psi_2(x) = \frac{(x + 1)(x - 1)}{(0 + 1)(0 - 1)} = -(x + 1)(x - 1)
\]

\[
\psi_3(x) = \frac{(x + 1)(x - 0)}{(1 + 1)(1 - 0)} = \frac{1}{2} x(x + 1)
\]

\[
P_2(0.2) = 1.2\psi_1(0.2) + 0.7\psi_2(0.2) + 1.8\psi_3(0.2) = 0.7920
\]
Shape Functions 1D

- Matlab code:

```matlab
x = 0:0.25:1; % measure points
f = @(x) x.^3-3*x.^2-3*sin(5*x); %inline exemple function
y = f(x); %measured values
figure()
plot(x,y,'o','Marker','o','MarkerFaceColor','red');
title('Measured values'); % plot only measured values
figure()
plot(x,y,'o','Marker','o','MarkerFaceColor','red');
hold on;
xx = 0:0.01:1; % take many points for a better plot
yy = f(xx);
plot(xx, yy,'-.'

```

```matlab
title('True function'); % plot the function
hold off
```
Shape Functions 1D

- Matlab code (continuation):

```matlab
figure() % Polyfit degree 4 function (interpolator)
plot(x,y,'o','Marker','o','MarkerFaceColor','red');
hold on;
plot(xx, yy, '-.')
p = polyfit(x,y,4);
yyy = polyval(p, xx);
plot(xx, yyy, '-b')
title('Degree 4 approximation');
hold off
```
Shape Functions 1D

- Interpolation Results:
Approximation: In case $n < (N - 1)$ there is NO solution (incompatible system) because the polynomial cannot pass through all the points. In that case: mean square approx

$$P_n(x) = \min(\| \tilde{P}_n(x) - f(x) \|)$$
Shape Functions 1D

• **Higher interpolation:** When $N$ is big (>8) there is a better solution than using a polynomial of high degree.

• **Runge’s Phenomenon:**

$$f(x) = \frac{1}{1 + 25x^2}$$

- The red curve is the Runge function $f(x)$.
- The blue curve is a 5th-order interpolating polynomial (using six equally spaced interpolating points).
- The green curve is a 9th-order interpolating polynomial (using ten equally spaced interpolating points).
Shape Functions 1D

Two possible approaches:

**Polygonal curve:** Take the points in pairs and build a line segment (1st order interp) in each case. **Continuous but not derivable.**

Matlab code:

```matlab
%% Polygonal
figure()
x=0:0.1:1;
y=f(x);
plot(x,y,'o','Marker','o','MarkerFaceColor','red');
hold on;
plot(xx,yy,'-.' )
xxx=0:0.01:1;
yyy=interp1(x,y,xx);  %only substitution is possible
plot(xxx,yyy,'b')
hold off
```
**Shape Functions 1D**

Two possible approaches:

**Spline curve** *(cubic)*: Take the points in groups of 2 and build a third order Polynomial (cubic interp) in each case imposing $C^2$ continuity.

Matlab code (continuation):

```matlab
%% Spline
figure()
x=0:0.1:1;
y=f(x);
plot(x,y,'o','Marker','o','MarkerFaceColor','red');
hold on;
plot(xx,yy,'-')
xxx = 0:0.01:1;
yyy = spline(x,y,xx);  %only substitution is possible
plot(xxx,yyy,'b')
hold off
```

![Graph of Spline curve](image)
Shape Functions 2D
Shape Functions 2D

How can we extend the interpolation problem to 2D? That is, how can we approximate the value of $f(x, y)$ using a 2D polynomial $P_n(x, y)$?

We restrict to fixed shapes:

**Triangular element**
Numbered counterclockwise

**Quadrilateral element**
Triangular Shape Functions

We need the equivalent to the Lagrange polynomials (2-dim)\n$$\psi^e_i(x, y) = c_1 + c_2 x + c_3 y$$ (three unknowns)\nsuch that, it’s value is 1 for vertex \( i \), and 0 for the other two:

For the first vertex:\n$$1 = \psi^e_1(x_1, y_1) = c_1 + c_2 x_1 + c_3 y_1$$
$$0 = \psi^e_1(x_2, y_2) = c_1 + c_2 x_2 + c_3 y_2$$
$$0 = \psi^e_1(x_3, y_3) = c_1 + c_2 x_3 + c_3 y_3$$

\[
\begin{pmatrix}
1 & x_1 & y_1 \\
1 & x_2 & y_2 \\
1 & x_3 & y_3
\end{pmatrix}
\begin{pmatrix}
c_1 \\
c_2 \\
c_3
\end{pmatrix}
=
\begin{pmatrix}
1 \\
0 \\
0
\end{pmatrix}
\quad \Rightarrow \quad
\begin{pmatrix}
c_1 \\
c_2 \\
c_3
\end{pmatrix} = \begin{pmatrix}
1 \\
0 \\
0
\end{pmatrix}
\]

\( c = A \backslash b \) (matlab solution)
Triangular Shape Functions

Explicitly

\[ \psi_i^e (x, y) = c_{1,i} + c_{2,i}x + c_{3,i}y, \quad i = 1,2,3 \]

\[ c_{1,i} = \frac{x_j y_k - x_k y_j}{2 A_e} \]
\[ c_{2,i} = \frac{y_j - y_k}{2 A_e} \]
\[ c_{3,i} = \frac{x_k - x_j}{2 A_e} \]

*where* \((i, j, k) = (1,2,3)\) *cyclic* and *\(A_e\)* is the *Area* of \(\Omega_e\)

**Remember:**

\[
\text{Area} = \frac{1}{2} \det \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix}
\]
Once we have the shape functions $\psi_i^e(x, y), \ i = 1,2,3$ (geometrically they are planes passing through one of the edges)

finally, the interpolation value of a point $(\tilde{x}, \tilde{y})$ is

\[
f(\tilde{x}, \tilde{y}) = f(x_1, y_1)\psi_1^e(\tilde{x}, \tilde{y}) + f(x_2, y_2)\psi_2^e(\tilde{x}, \tilde{y}) + f(x_3, y_3)\psi_3^e(\tilde{x}, \tilde{y})
\]

The terms $\alpha_i^e = \psi_i^e(\tilde{x}, \tilde{y})$ are known as **Barycentric Coordinates** of $(\tilde{x}, \tilde{y})$ (We’ll see more in the practical sessions)
**Example:** The temperature of a thin triangular plate has been measured at the vertices \( T(v_1) = 40^\circ, \ T(v_2) = 90^\circ, \ T(v_3) = 10^\circ. \)

If the coordinates of the vertices are: \( v_1 = (0,0), \ v_2 = (5,0), \ v_3 = (2,3), \) estimate the temperature at the point \( p = (3,1) \)?

**Solution:**

\[
\begin{align*}
\psi_1^e(x, y) &= 1 - 0.2x - 0.2y \\
\psi_2^e(x, y) &= 0 - 0.2x - 0.1333y \\
\psi_3^e(x, y) &= 0 - 0x - 0.3333y
\end{align*}
\]

\[
\alpha_1 = \psi_1^e(3,1) \quad \alpha_2 = \psi_2^e(3,1) \quad \alpha_3 = \psi_3^e(3,1)
\]

\[
T(p) = 40\alpha_1 + 90\alpha_2 + 10\alpha_3 = 53.3333^\circ
\]

**Exercise:** Compute the shape functions associated to the triangle \( v_1 = (1,1), \ v_2 = (3,2), \ v_3 = (2,4). \)

**Solution:**

\[
\begin{align*}
\psi_1^e(x, y) &= \frac{1}{5}(8 - 2x - y) \\
\psi_2^e(x, y) &= \frac{1}{5}(-2 + 3x - y) \\
\psi_3^e(x, y) &= \frac{1}{5}(-1 - x + 2y)
\end{align*}
\]
Quadrilateral Shape Functions

A similar situation is found when a **rectangular** element is used.

\[ \psi^e_i (x, y) = c_1 + c_2 x + c_3 y + c_4 xy \] (four unknowns)

such that, it’s value is 1 for vertex \( i \), and 0 for the other three:

For the first vertex:

\[ 1 = \psi^e_1 (x_1, y_1) = c_1 + c_2 x_1 + c_3 y_1 + c_4 x_1 y_1 \]
\[ 0 = \psi^e_1 (x_2, y_2) = c_1 + c_2 x_2 + c_3 y_2 + c_4 x_2 y_2 \]
\[ 0 = \psi^e_1 (x_3, y_3) = c_1 + c_2 x_3 + c_3 y_3 + c_4 x_3 y_3 \]
\[ 0 = \psi^e_1 (x_4, y_4) = c_1 + c_2 x_4 + c_3 y_4 + c_4 x_4 y_4 \]

Analogously for the other vertices

(This is not correct for general quadrilateral elements)
Quadrilateral Shape Functions

General expressions for Rectangles:

If we denote by

\[ a = x_2 - x_1 \]
\[ b = y_2 - y_1 \]

Then

\[ \psi_1^e(x, y) = \frac{1}{ab} (x - x_2)(y - y_2) \]
\[ \psi_2^e(x, y) = -\frac{1}{ab} (x - x_1)(y - y_2) \]
\[ \psi_3^e(x, y) = \frac{1}{ab} (x - x_1)(y - y_1) \]
\[ \psi_4^e(x, y) = -\frac{1}{ab} (x - x_2)(y - y_1) \]
Quadrilateral Shape Functions

The **reference** bilinear quadrilateral element $\Omega^R$: $[-1,1] \times [-1,1]

We can use Matlab to compute the coeff.

$$\psi_j^e(x_i, y_i) = c_1 j + c_2 j x_i + c_3 j y_i + c_4 j x_i y_i = \delta_{ij}$$

punts =

```
-1,-1;
1,-1;
1,1;
-1,1;
```

A = [ones(4,1), punts(:,1), punts(:,2), punts(:,1).*punts(:,2)]

b = [1;0;0;0];

C1 = A\b % coeff of the first shape function

% **exercise**: do it for the other shape functions
Quadrilateral Shape Functions

The shape functions for the reference element $\Omega^R$, are:

\[
\psi_1^R(\xi, \eta) = \frac{(1 - \xi)(1 - \eta)}{2}, \quad \psi_2^R(\xi, \eta) = \frac{(1 + \xi)(1 - \eta)}{2},
\]

\[
\psi_3^R(\xi, \eta) = \frac{(1 + \xi)(1 + \eta)}{2}, \quad \psi_4^R(\xi, \eta) = \frac{(1 - \xi)(1 + \eta)}{2}.
\]

Or equivalently:

\[
\psi_1^R(\xi, \eta) = \frac{1 - \xi - \eta + \xi\eta}{4}, \quad \psi_2^R(\xi, \eta) = \frac{1 + \xi - \eta - \xi\eta}{4},
\]

\[
\psi_3^R(\xi, \eta) = \frac{1 + \xi + \eta + \xi\eta}{4}, \quad \psi_4^R(\xi, \eta) = \frac{1 - \xi + \eta - \xi\eta}{4}.
\]
Quadrilateral Shape Functions

Example: The temperature of at the vertices of the reference quadrilateral has been measured as $T(v_1) = 10^\circ$, $T(v_2) = 20^\circ$, $T(v_3) = 30^\circ$, $T(v_4) = 40^\circ$. Estimate the temperature at the point $p = (-0.3, 0.5)$?

Solution:

$$
\psi_1^R(p) = 0.1625; \quad \psi_2^R(p) = 0.0875; \\
\psi_3^R(p) = 0.2625; \quad \psi_4^R(p) = 0.4875; \\

T(p) = T_1 \psi_1^R(p) + T_2 \psi_2^R(p) + T_3 \psi_3^R(p) + T_4 \psi_4^R(p) = 30.75^\circ
$$
Quadrilateral Shape Functions

• For a General Quadrilateral element usually the shape function is NOT a 2 degree polynomial. Because of that it is not easy to compute these functions, but we still can compute the **Barycentric Coordinates**.

How to compute $\alpha_i^k = \psi_i^k (P)$?
Quadrilateral Shape Functions

• Change from the Reference quadrilateral to [0,1]x[0,1]

\[ \lambda = \frac{\xi + 1}{2}, \quad \mu = \frac{\eta + 1}{2} \]

Shape functions on [0,1]x[0,1]:

\[ \psi_1(\lambda, \mu) = (1 - \lambda)(1 - \mu) \quad \psi_3(\lambda, \mu) = \lambda \mu \]
\[ \psi_2(\lambda, \mu) = \lambda(1 - \mu) \quad \psi_4(\lambda, \mu) = (1 - \lambda)\mu \]
Quadrilateral Shape Functions

• General Quadrilateral: **Isoparametric** transformation

\[(x, y) = ψ_1(ξ, η)v_1 + ψ_2(ξ, η)v_2 + ψ_3(ξ, η)v_3 + ψ_4(ξ, η)v_4\]

Change to another quadrilateral using the shape functions: for simplicity we will use here the rectangle \([0,1] \times [0,1]:\)

\[P(λ, μ) = (1 − λ)(1 − μ)v_1 + λ(1 − μ)v_2 + λμv_3 + (1 − λ)μv_4\]
Quadrilateral Shape Functions

We can compute them using $\lambda, \mu \in [0,1]$, as a parametrization of the quadrilateral edges.

$$P = (1 - \lambda)(1 - \mu) v_1 + \lambda(1 - \mu) v_2 + \lambda \mu v_3 + (1 - \lambda)\mu v_4$$

Rearranging the terms, the previous equation can be written as:

$$a + \lambda b + \mu c + \lambda \mu d = 0$$

where

$$a = v_1 - P, \quad b = v_2 - v_1, \quad c = v_4 - v_1,$$

$$d = v_1 - v_2 + v_3 - v_4$$
Quadrilateral Shape Functions

We impose that the vector
\[ \mathbf{w} = Q(\lambda, 1) - Q(\lambda, 0) \]
and the vector
\[ \mathbf{u} = P - Q(\lambda, 0) \]
are parallel, i.e. \( \mathbf{w} \times \mathbf{u} = 0 \)

From the isometric transformation we obtain:
\[ \mathbf{w} = v_4 - v_1 + \lambda(v_3 - v_4 + v_1 - v_2) = \mathbf{c} + \lambda \mathbf{d} \]
\[ \mathbf{u} = P - v_1 + \lambda(v_1 - v_2) = -\mathbf{a} - \lambda \mathbf{b} \]
Quadrilateral Shape Functions

If this two vectors are parallel

\[(a + \lambda b) \times (c + \lambda d) = 0\]

or

\[(b \times d)\lambda^2 + (a \times d + b \times c)\lambda + a \times c = 0\]

which is a quadratic equation that can be solved for \(\lambda\).

Hint: All the 2D vectors are extended with a zero third component in order to define the cross product.

Analogously a similar equation is found for \(\mu\).
Quadrilateral Shape Functions

• Example: Given a quadrilateral defined by vertices
  \( v_1 = (0,0), v_2 = (5,-1), v_3 = (4,5), v_4 = (1,4) \)
  compute the barycentric coordinates of point \( p=(3,2) \).
  Sol. \( \lambda = 0.6250, \mu = 0.5 \)
  \[ \alpha = [0.1875 \ 0.3125 \ 0.3125 \ 0.1875] \].

(examples using Matlab will be shown in practices)