



**NPTel ONLINE CERTIFICATION COURSES**

# **Finite Element Method**

**Week 7: 2D Formulation**

**Lecture 32: Introduction to FEM in 2D**

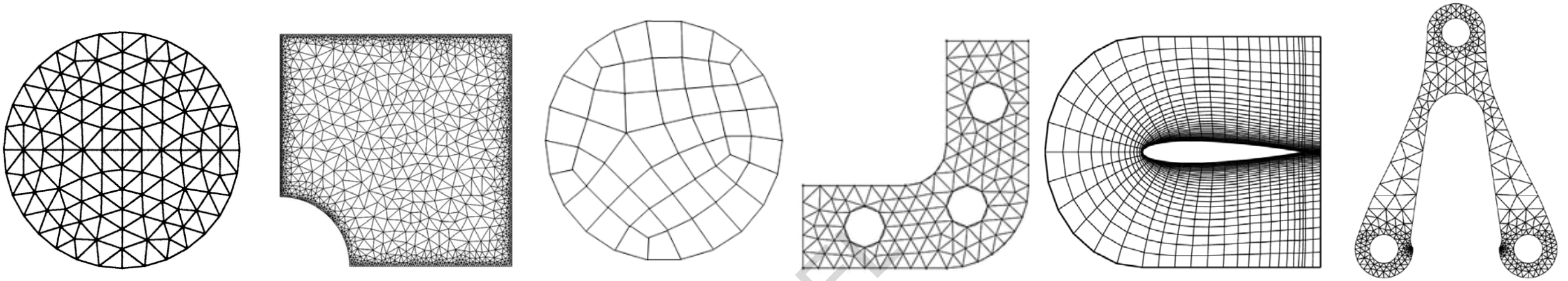
**Biswanath Banerjee**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

**Amit Shaw**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

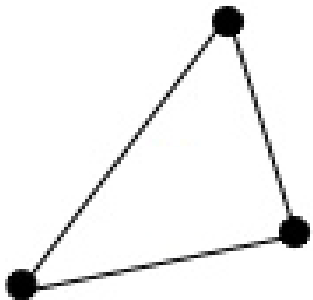
# **Recall: Steps in Finite Element Method**

- ↓ Boundary Value Problem
- ↓ Discretization of the Domain
- ↓ Derivation of Element Equations
- ↓ Assembly of Element Equations based on Element Connectivity
- ↓ Imposition of Boundary Conditions
- ↓ Solutions
- Post-processing

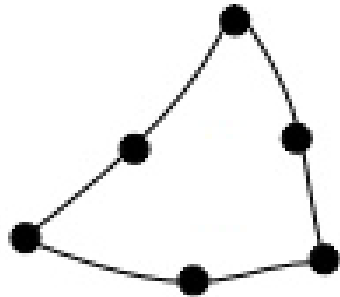
## *Discretization in 2D*



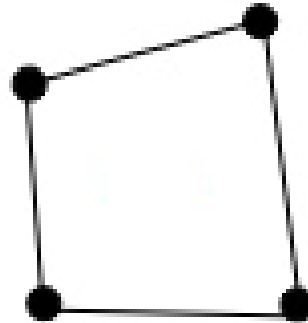
## *Few 2D Finite Elements*



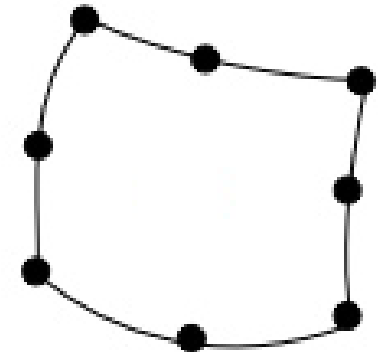
3 Node  
Triangle



6 Node  
Triangle



4 Node  
Quadrilateral



8 Node  
Quadrilateral

# Recall: Steps in Finite Element Method

- ↓ Boundary Value Problem
- ↓ Discretization of the Domain
- ↓ Derivation of Element Equations
- ↓ Assembly of Element Equations based on Element Connectivity
- ↓ Imposition of Boundary Conditions
- ↓ Solutions
- Post-processing

## Recall: Variational (Weak) Problem

$$B(w, u) = l(w)$$

$w$  = weight / test function

$u$  = trial function

Find  $u$  such that the above equation holds for any  $w$  which

- a) is sufficiently continuous as required by the weak form
- b) satisfies the specified homogeneous form of essential boundary conditions on  $u$

## Recall: Variational (Weak) Problem

$$B(w, u) = l(w)$$

Following Rayleigh-Ritz procedure,  $u = \sum_{j=1}^n \phi_j^e u_j^e$  and  $w = \{\phi_j^e\}$

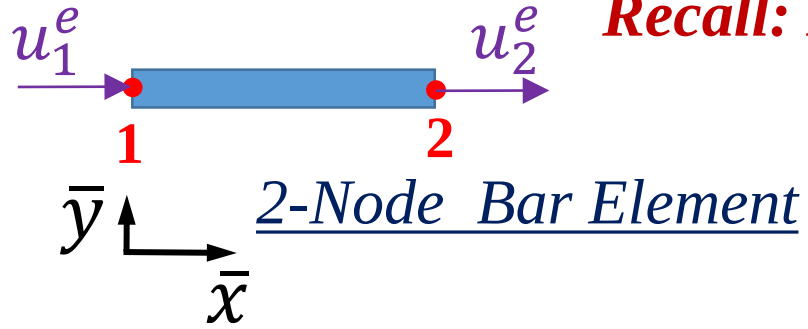
*Shape function for element* (pointing to  $\phi_j^e$ )

*Nodal values* (pointing to  $u_j^e$ )

### *Shape functions*

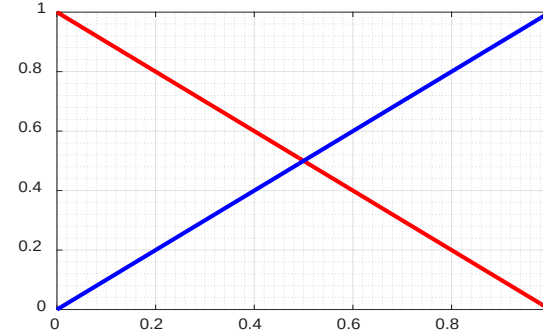
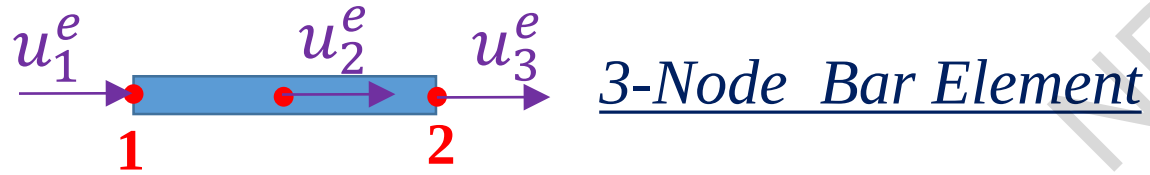
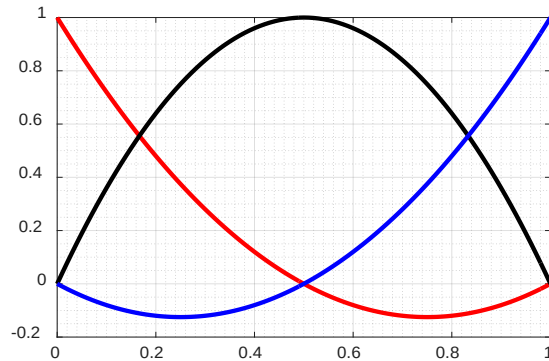
- Continuity – Sufficiently differentiable as required by the weak form
- Completeness – include all lower-order terms up to the highest order used
- Interpolation – interpolate the primary variables at the nodes of the element

## Recall: FEM in 1D



$$\phi_1^e = 1 - \frac{\bar{x}}{h_e}$$

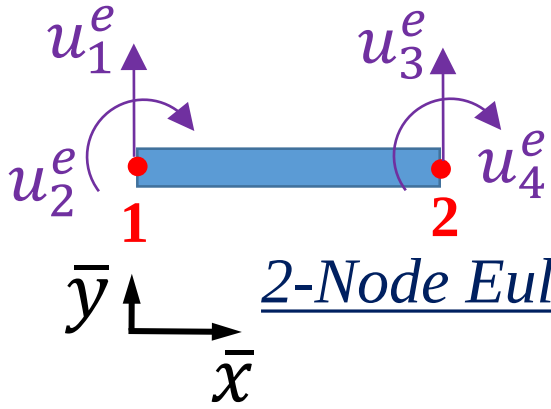
$$\phi_2^e = \frac{\bar{x}}{h_e}$$



$$\phi_1^e = \left(1 - \frac{\bar{x}}{h_e}\right) \left(1 - \frac{2\bar{x}}{h_e}\right)$$

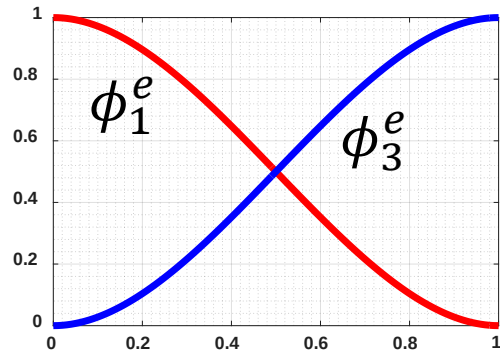
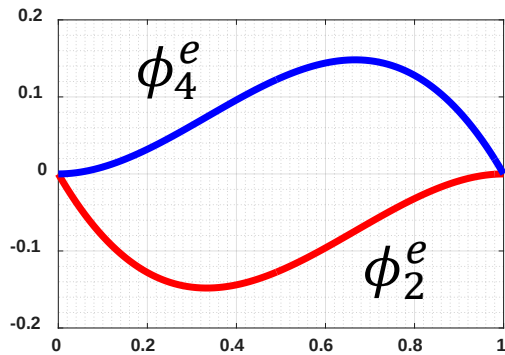
$$\phi_2^e = 4 \frac{\bar{x}}{h_e} \left(1 - \frac{\bar{x}}{h_e}\right)$$

$$\phi_3^e = -\frac{\bar{x}}{h_e} \left(1 - \frac{2\bar{x}}{h_e}\right)$$



## Recall: FEM in 1D

### 2-Node Euler-Bernoulli Beam Element



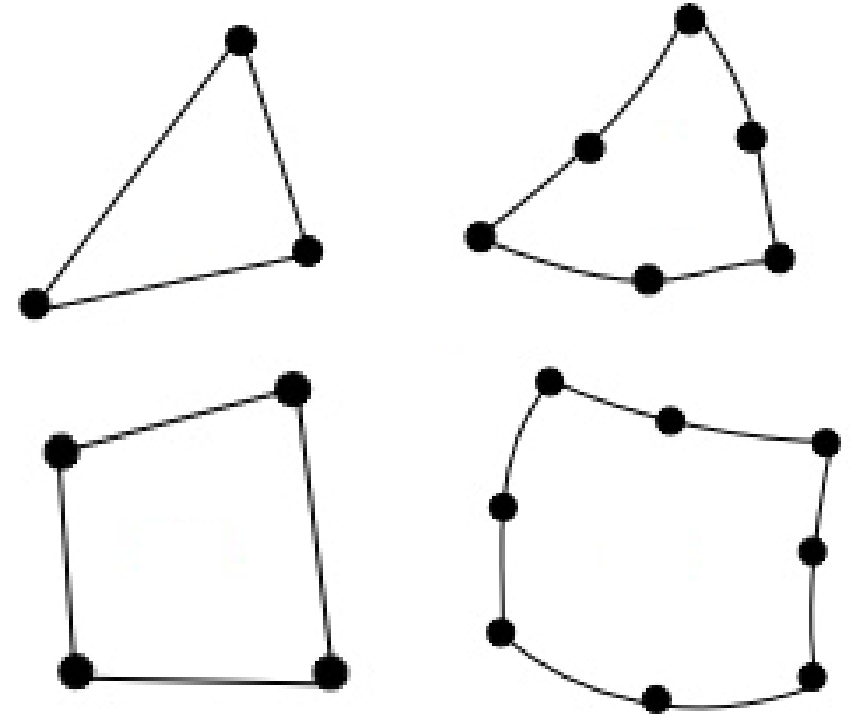
$$\phi_1^e = 1 - 3\left(\frac{\bar{x}}{h_e}\right)^2 + 2\left(\frac{\bar{x}}{h_e}\right)^3$$

$$\phi_2^e = -\bar{x}\left(1 - \frac{\bar{x}}{h_e}\right)^2$$

$$\phi_3^e = 3\left(\frac{\bar{x}}{h_e}\right)^2 - 2\left(\frac{\bar{x}}{h_e}\right)^3$$

$$\phi_4^e = -\bar{x}\left[\left(\frac{\bar{x}}{h_e}\right)^2 - \frac{\bar{x}}{h_e}\right]$$

## FEM in 2D



**Task 1: Construction of shape function over 2D element**

# Recall: Variational (Weak) Problem

$$B(w, u) = l(w)$$

**Boundary value problem**

$$-\frac{d}{dx}\left(a\frac{du}{dx}\right) + cu - q = 0 \quad \text{for } x \in (0, L)$$

$$u = 0 \quad \text{at } x = 0$$

$$\left(a\frac{du}{dx}\right) = Q_0 \quad \text{at } x = L$$

**Weak form**

$$\int_{\Omega^e} \left( a \frac{dw}{dx} \frac{du}{dx} + cwu \right) dx$$

$$= \int_{\Omega^e} wq dx + w(x_A)Q_A + w(x_B)Q_B$$

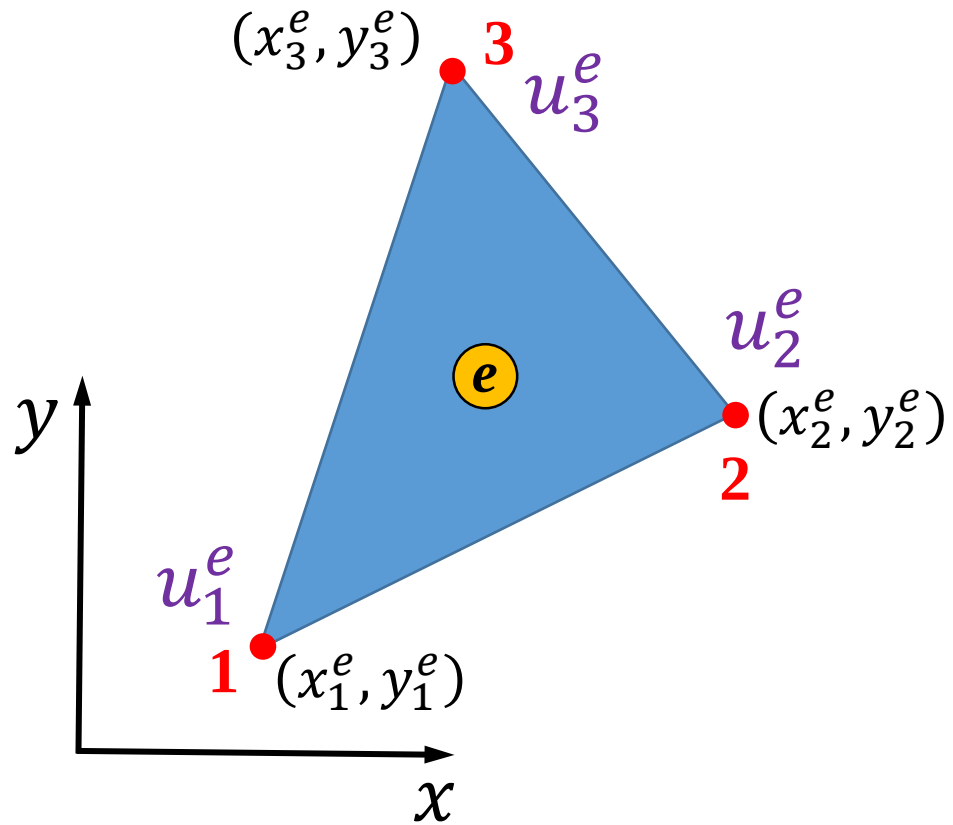
Element domain  $\Omega^e = (x_A, x_B)$

**Task 2: Numerical integration**

## This Week

- Different elements (Triangular, Rectangular, Quadrilateral etc.)
- Completeness, Continuity and Shape functions
- Gauss Quadrature technique for numerical integration

# Shape Function For 3-Node Triangle



*Polynomial approximation*

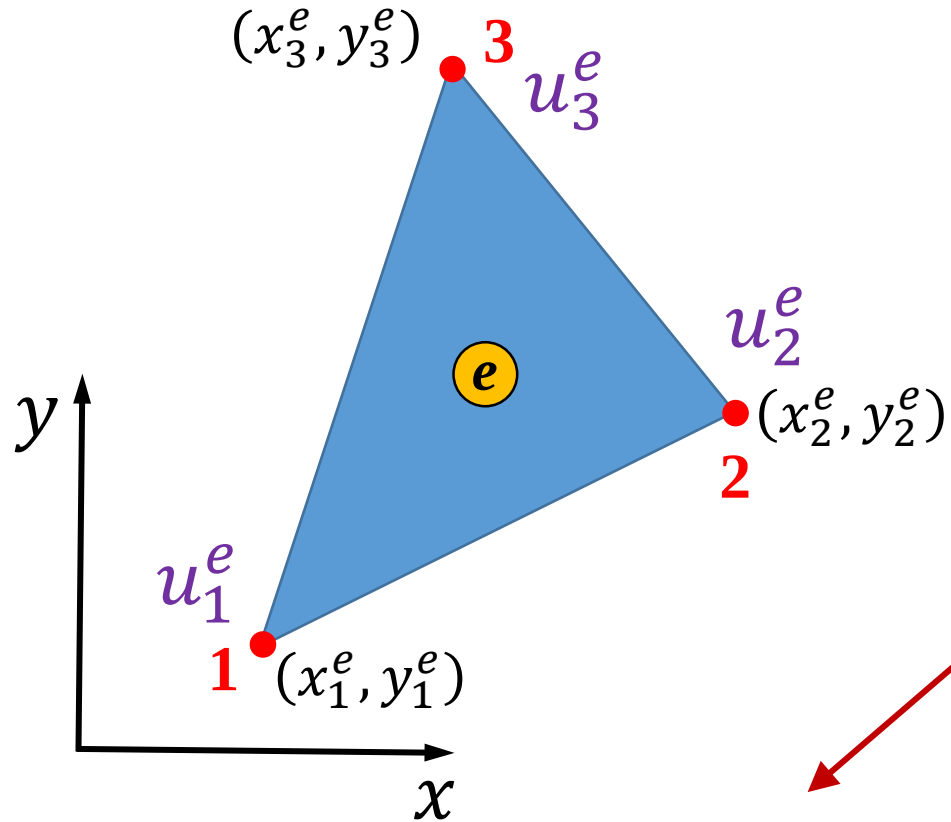
$$u^e(x, y) = c_1 + c_2x + c_3y$$



*Completeness*

NPTEL

# Shape Function For 3-Node Triangle



*Polynomial approximation*

$$u^e(x, y) = c_1 + c_2x + c_3y$$

*Interpolation*

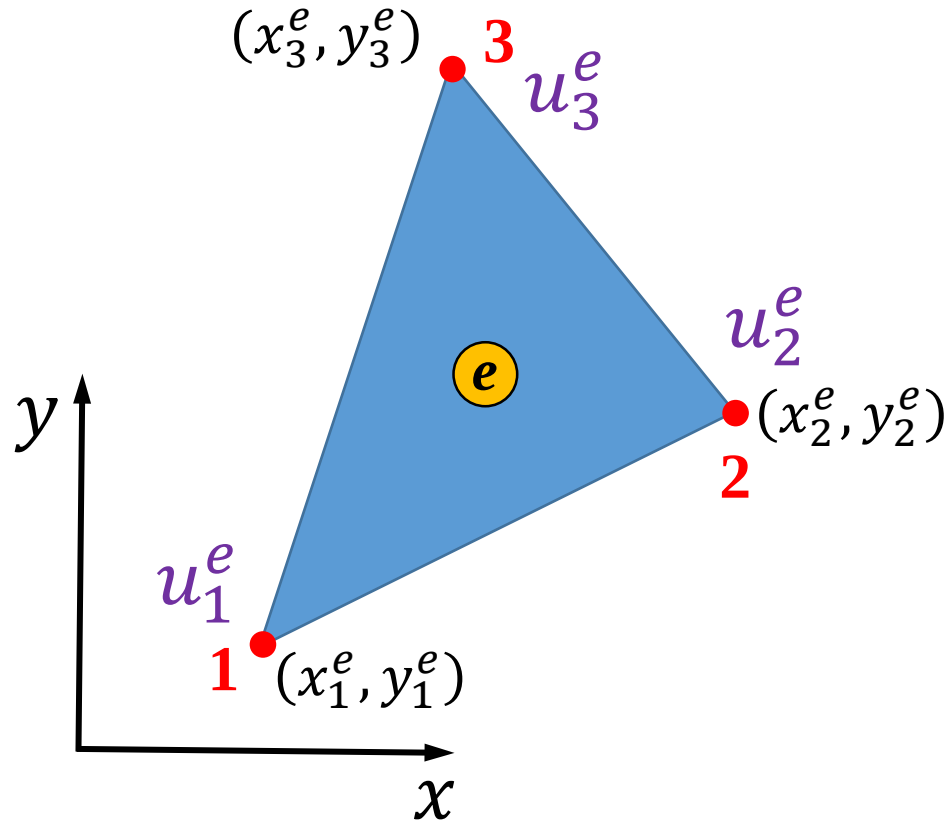
$$u_1^e = u^e(x_1^e, y_1^e) = c_1 + c_2x_1^e + c_3y_1^e$$

$$u_2^e = u^e(x_2^e, y_2^e) = c_1 + c_2x_2^e + c_3y_2^e$$

$$u_3^e = u^e(x_3^e, y_3^e) = c_1 + c_2x_3^e + c_3y_3^e$$

$$\begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix} = \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \Rightarrow \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



*Polynomial approximation*

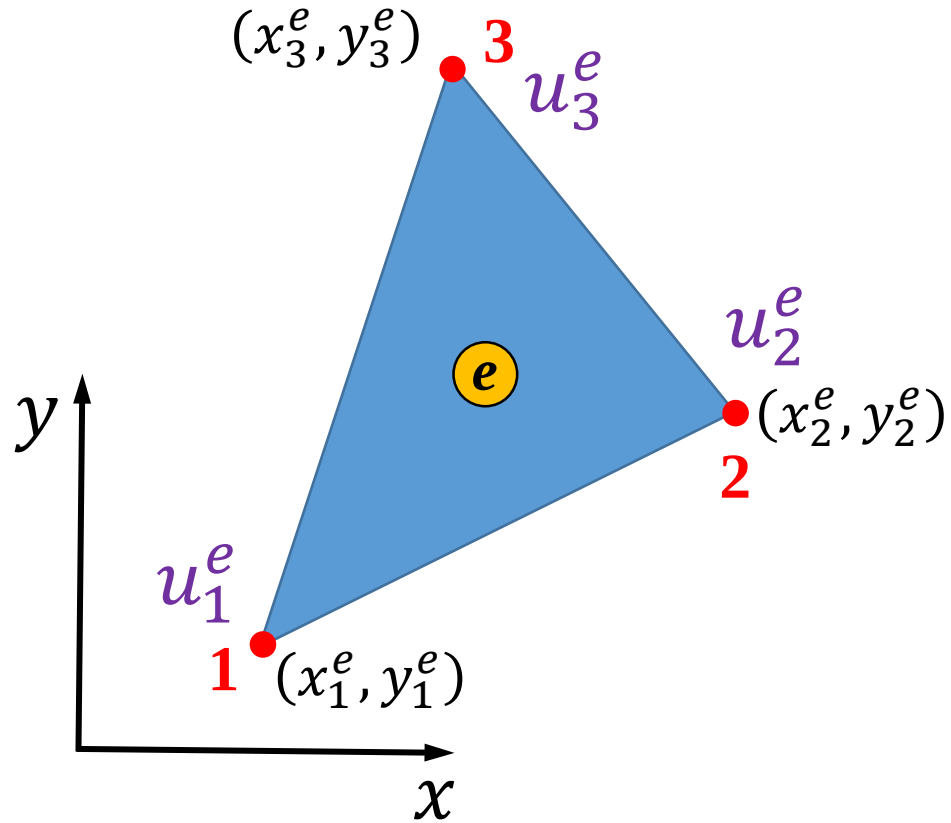
$$u^e(x, y) = c_1 + c_2x + c_3y$$

$$\begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} = \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}$$

$$2A_e = \alpha_1 + \alpha_2 + \alpha_3$$

$A_e$  = Area of the element

# Shape Function For 3-Node Triangle



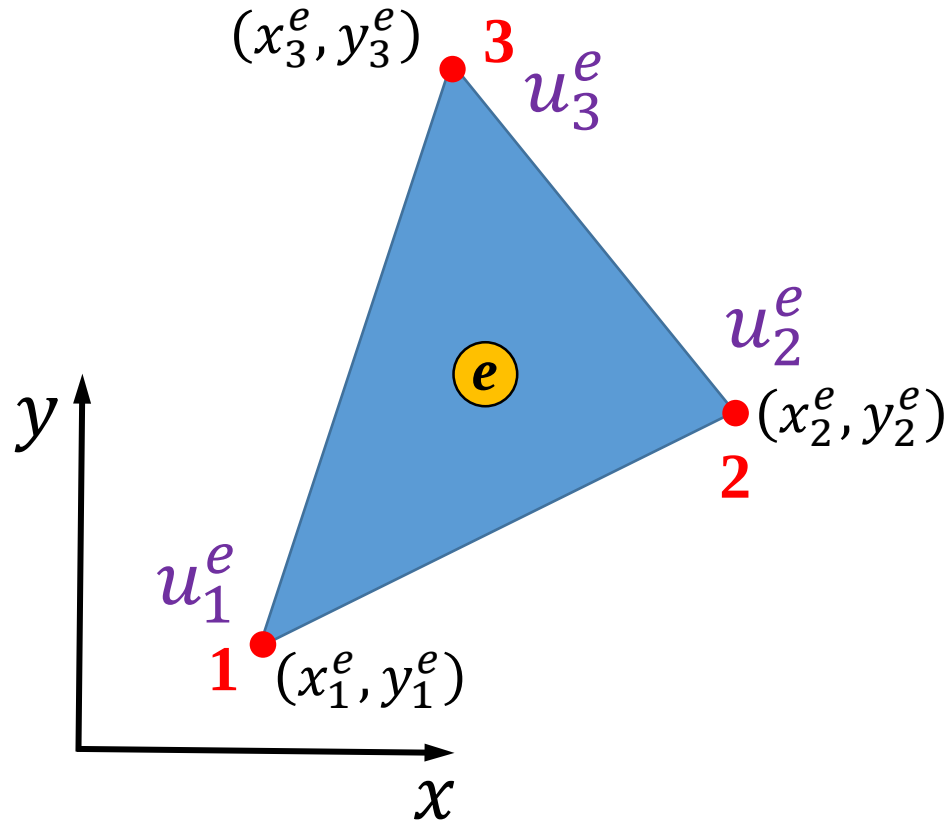
*Polynomial approximation*

$$u^e(x, y) = c_1 + c_2x + c_3y$$

$$\begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} = \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}$$

$$u^e(x, y) = \{1 \quad x \quad y\} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



*Polynomial approximation*

$$u^e(x, y) = c_1 + c_2x + c_3y$$

*Interpolation*

$$u_1^e = u^e(x_1^e, y_1^e) = c_1 + c_2x_1^e + c_3y_1^e$$

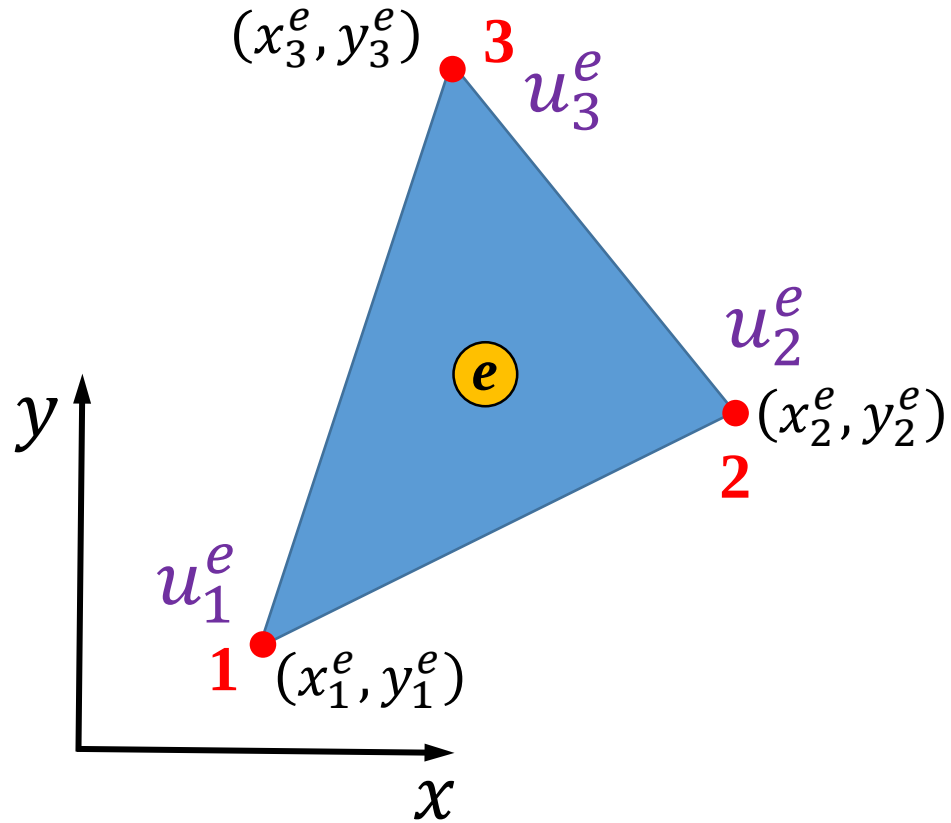
$$u_2^e = u^e(x_2^e, y_2^e) = c_1 + c_2x_2^e + c_3y_2^e$$

$$u_3^e = u^e(x_3^e, y_3^e) = c_1 + c_2x_3^e + c_3y_3^e$$

$$\begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix} = \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix}$$

$$\Rightarrow \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



$$u^e(x, y) = \{1 \quad x \quad y\} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix}$$

*Polynomial approximation*

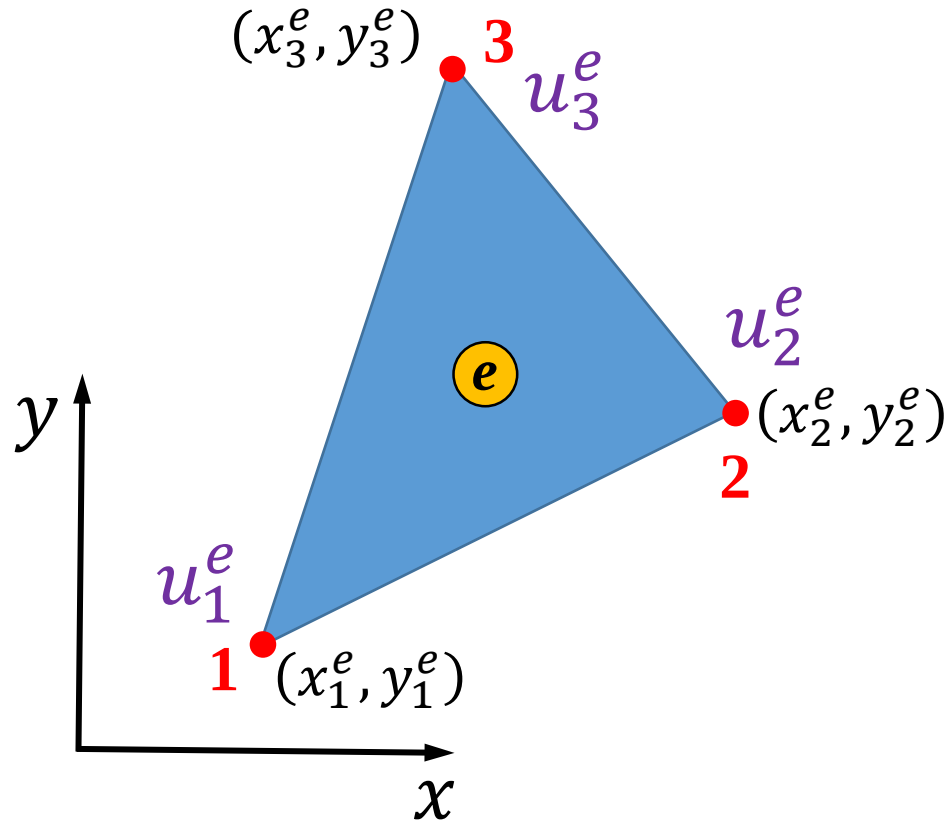
$$u^e(x, y) = c_1 + c_2x + c_3y$$

$$\begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} = \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}$$

$$\Rightarrow u^e(x, y)$$

$$= \{1 \quad x \quad y\} \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



$$u^e(x, y) = \{1 \quad x \quad y\} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix}$$

*Polynomial approximation*

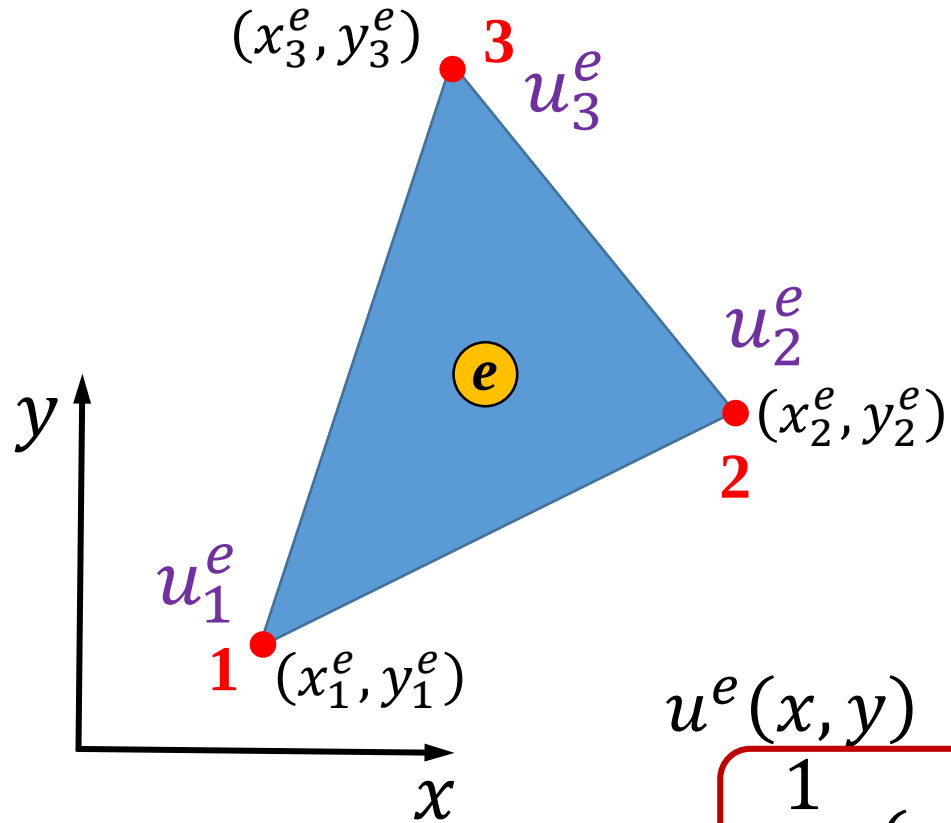
$$u^e(x, y) = c_1 + c_2x + c_3y$$

$$\begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} = \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix}$$

$$\Rightarrow u^e(x, y)$$

$$= \{1 \quad x \quad y\} \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



*Polynomial approximation*

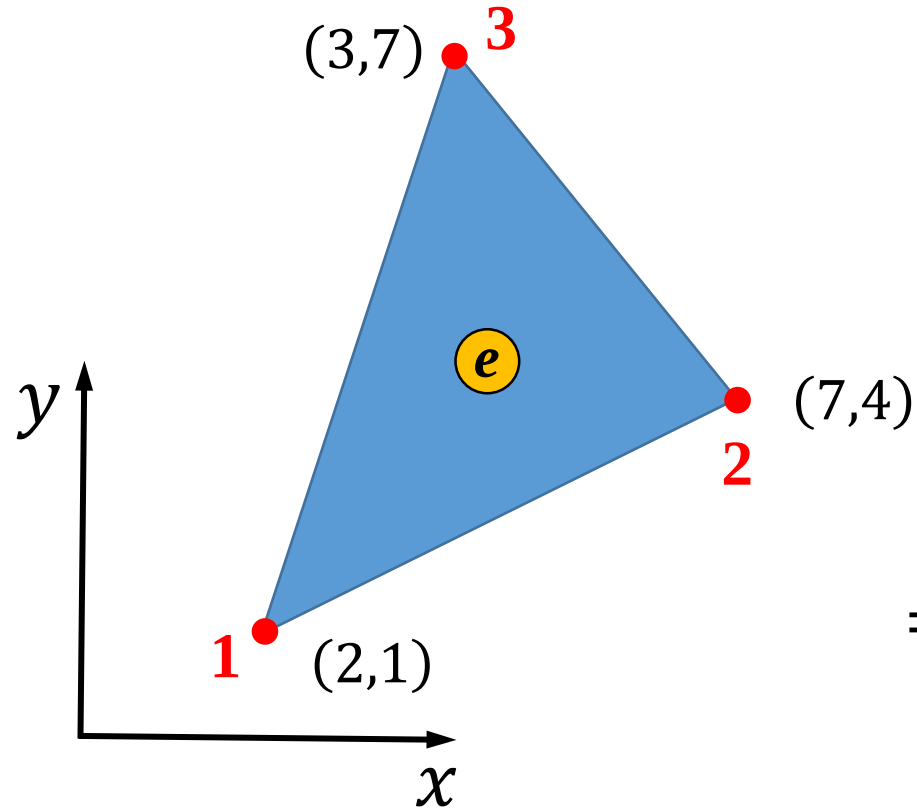
$$u^e(x, y) = c_1 + c_2x + c_3y$$

$$u^e(x, y)$$

$$= \{1 \quad x \quad y\} \frac{1}{2A_e} \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\begin{aligned} u^e(x, y) &= \underbrace{\frac{1}{2A_e} (\alpha_1 + \beta_1 x + \gamma_1 y)}_{\phi_1^e} u_1^e + \underbrace{\frac{1}{2A_e} (\alpha_2 + \beta_2 x + \gamma_2 y)}_{\phi_2^e} u_2^e \\ &\quad + \underbrace{\frac{1}{2A_e} (\alpha_3 + \beta_3 x + \gamma_3 y)}_{\phi_3^e} u_3^e \\ &\Rightarrow u^e(x, y) = \sum_{i=1}^3 \phi_i^e(x, y) u_i^e \end{aligned}$$

# Shape Function For 3-Node Triangle

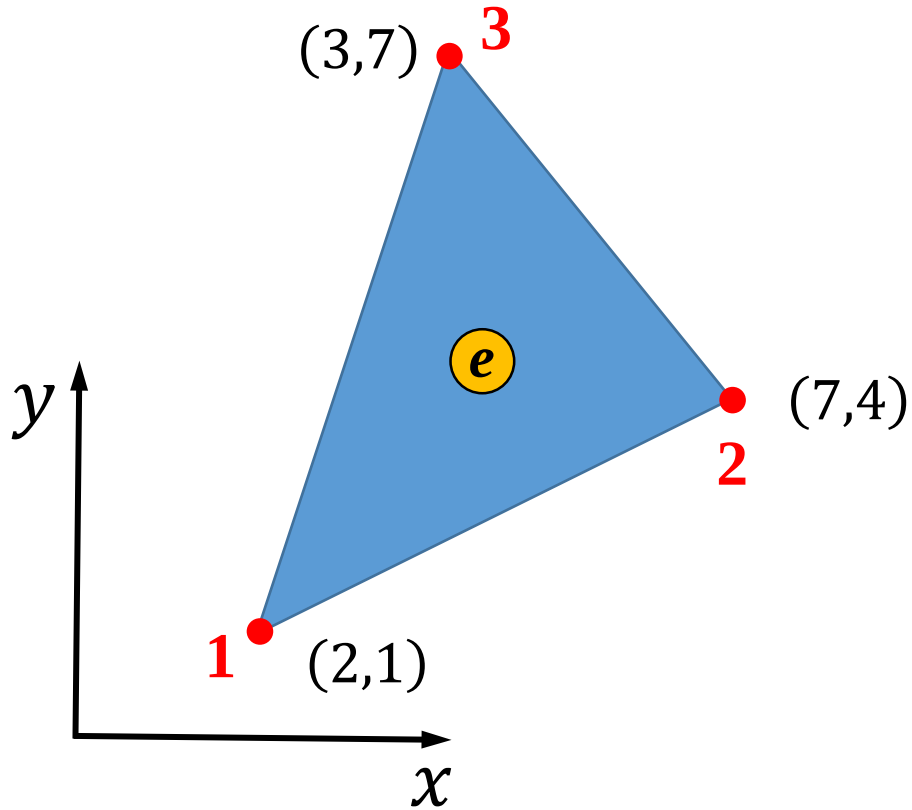


$$u^e(x, y) = \{1 \quad x \quad y\} \begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\Rightarrow u^e(x, y) = \{1 \quad x \quad y\} \begin{bmatrix} 1 & 2 & 1 \\ 1 & 7 & 4 \\ 1 & 3 & 7 \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\Rightarrow u^e(x, y) = \{1 \quad x \quad y\} \frac{1}{27} \begin{bmatrix} 37 & -11 & 1 \\ -3 & 6 & -3 \\ -4 & -1 & 5 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

# Shape Function For 3-Node Triangle



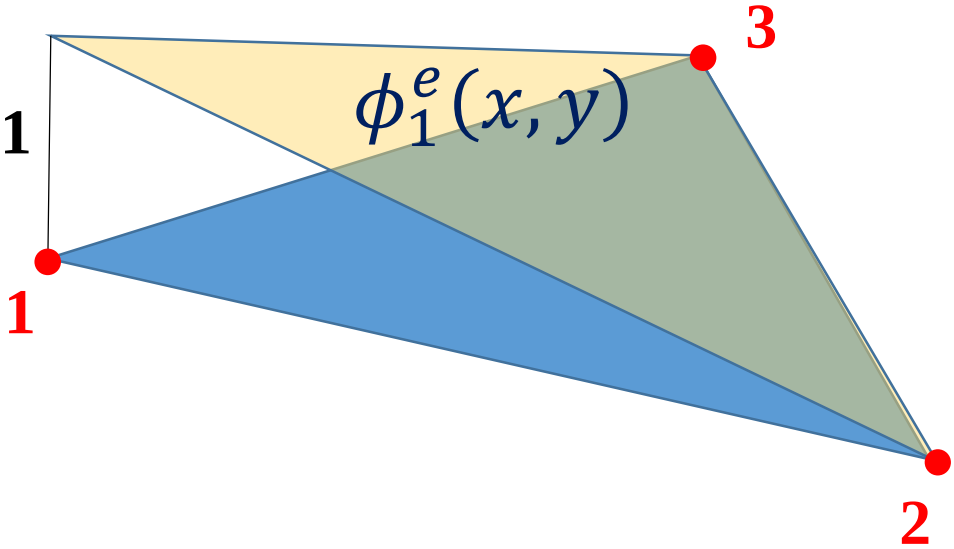
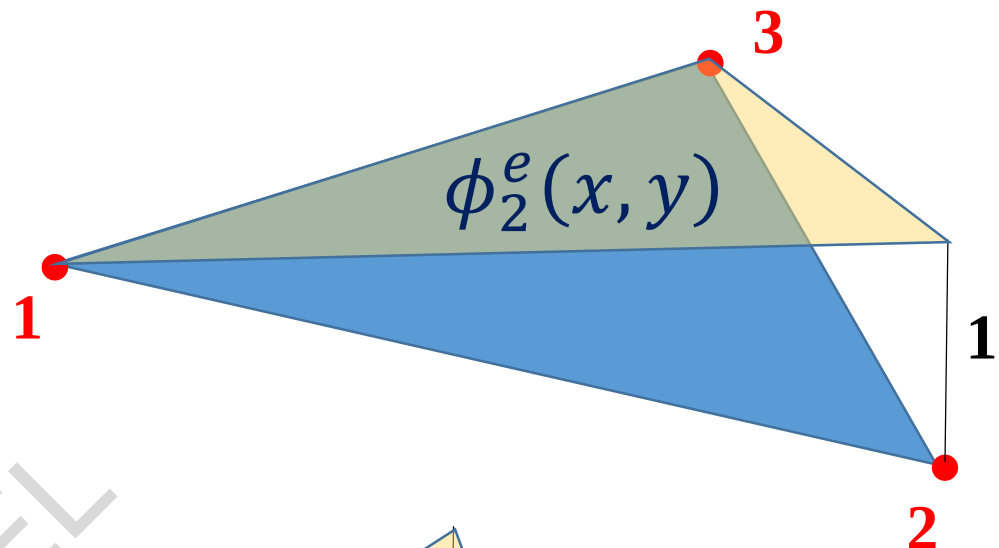
$$u^e(x, y) = \{1 \quad x \quad y\} \frac{1}{27} \begin{bmatrix} 37 & -11 & 1 \\ -3 & 6 & -3 \\ -4 & -1 & 5 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\phi_1^e = \frac{1}{27} (37 - 3x - 4y)$$

$$\phi_2^e = \frac{1}{27} (-11 + 6x - y)$$

$$\phi_3^e = \frac{1}{27} (1 - 3x + 5y)$$

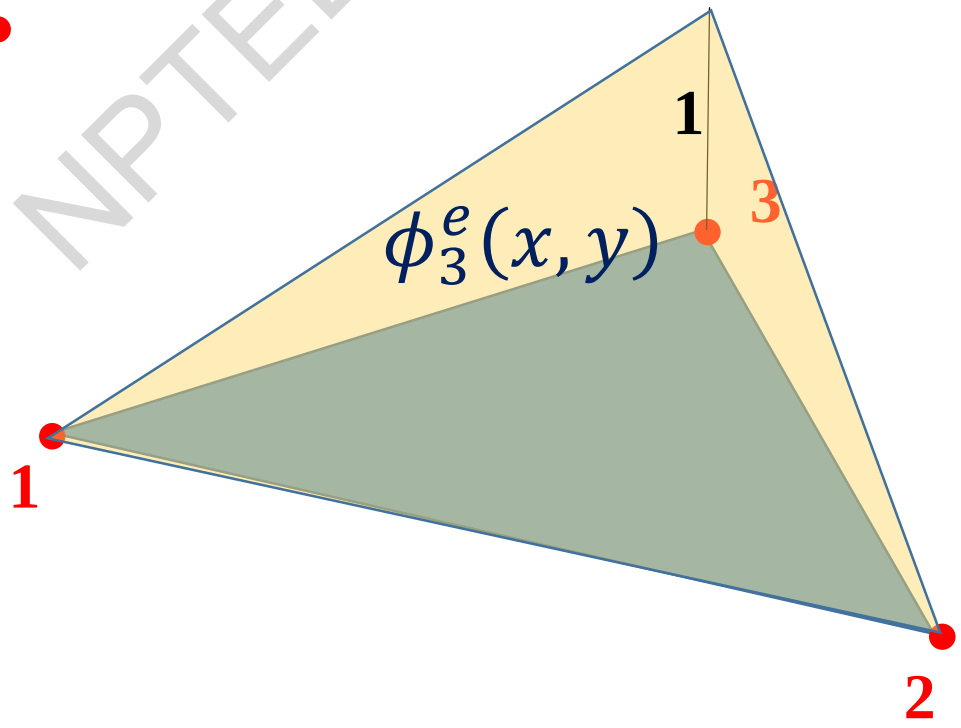
# Shape Function For 3-Node Triangle



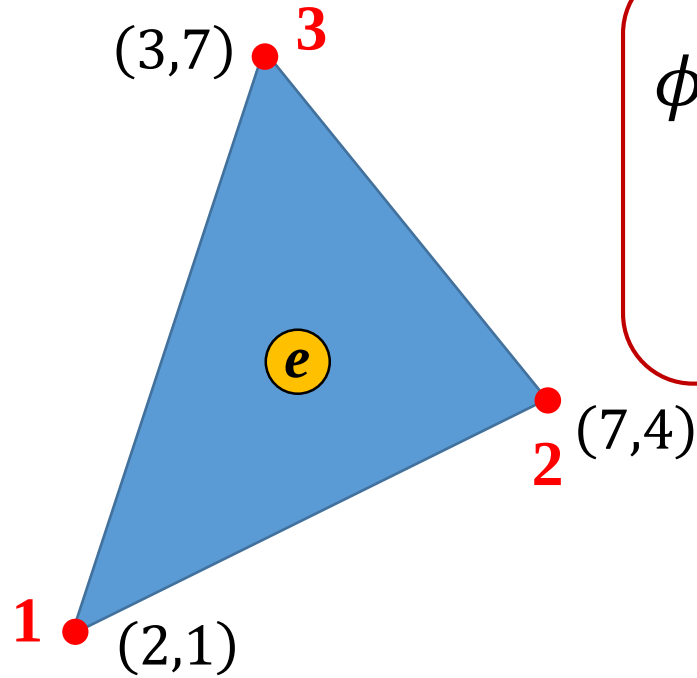
*Kronecker delta property*

$$\phi_i^e(x_j^e, y_j^e) = \delta_{ij}$$

NPTEL



# Shape Function For 3-Node Triangle



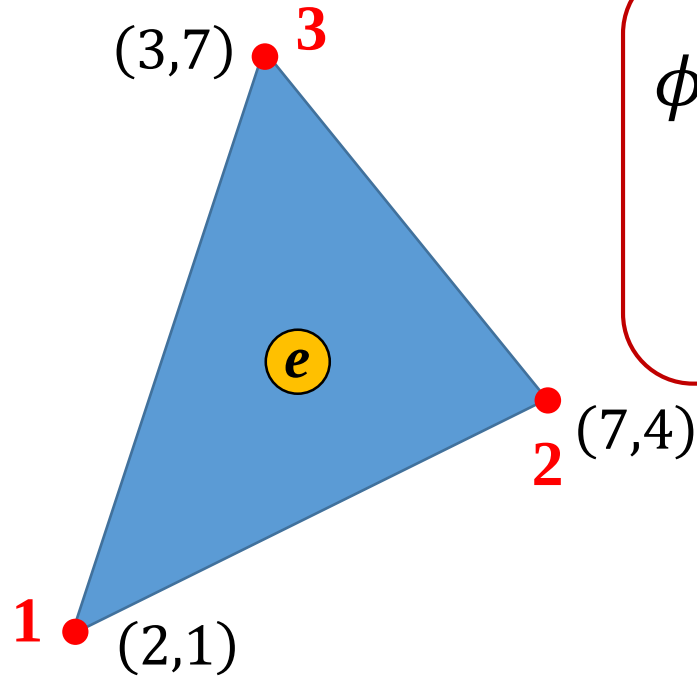
$$\phi_1^e = \frac{1}{27} (37 - 3x - 4y) \quad \phi_2^e = \frac{1}{27} (-11 + 6x - y)$$

$$\phi_3^e = \frac{1}{27} (1 - 3x + 5y)$$

$$\sum_{i=1}^3 \phi_i^e(x, y) = 1$$

*Partition of unity property*

# Shape Function For 3-Node Triangle



$$\phi_1^e = \frac{1}{27} (37 - 3x - 4y) \quad \phi_2^e = \frac{1}{27} (-11 + 6x - y)$$

$$\phi_3^e = \frac{1}{27} (1 - 3x + 5y)$$

$$\sum_{i=1}^3 \phi_i^e(x, y) = 1$$

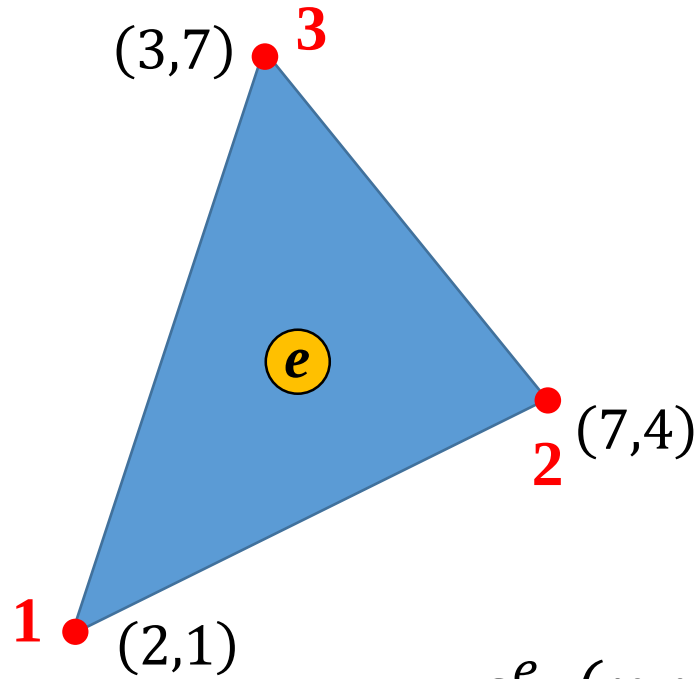
$$\frac{\partial \phi_1^e}{\partial x} = -\frac{3}{27} \quad \frac{\partial \phi_2^e}{\partial x} = \frac{6}{27} \quad \frac{\partial \phi_3^e}{\partial x} = -\frac{3}{27}$$

$$\sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial x} = 0$$

$$\frac{\partial \phi_1^e}{\partial y} = -\frac{4}{27} \quad \frac{\partial \phi_2^e}{\partial y} = -\frac{1}{27} \quad \frac{\partial \phi_3^e}{\partial y} = \frac{5}{27}$$

$$\sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial y} = 0$$

# Shape Function For 3-Node Triangle



*Strains*

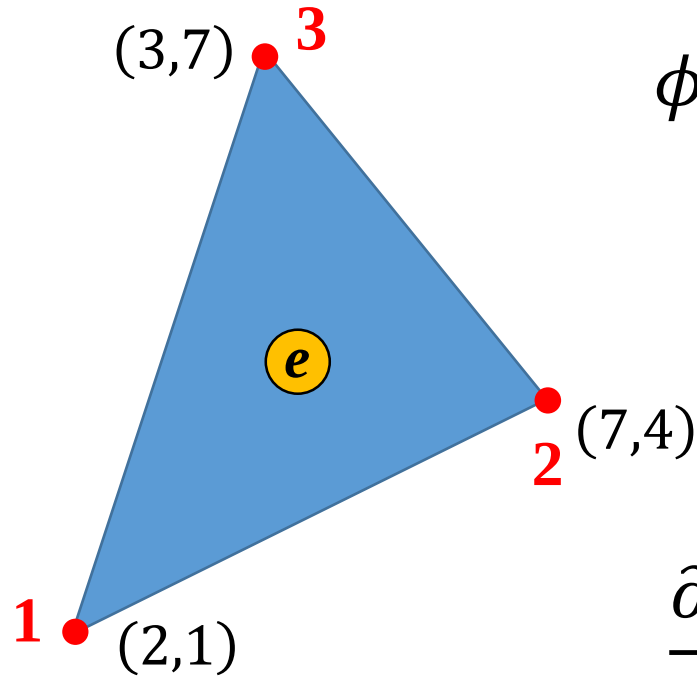
$$u^e(x, y) = \sum_{i=1}^3 \phi_i^e u_i^e$$

$$\epsilon_{xx}^e(x, y) = \frac{\partial u^e(x, y)}{\partial x} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial x} u_i^e$$

$$\epsilon_{yy}^e(x, y) = \frac{\partial u^e(x, y)}{\partial y} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial y} u_i^e$$

$$\epsilon_{xy}^e(x, y) = \frac{1}{2} \left( \frac{\partial u^e(x, y)}{\partial x} + \frac{\partial u^e(x, y)}{\partial y} \right) = \frac{1}{2} \sum_{i=1}^3 \left( \frac{\partial \phi_i^e(x, y)}{\partial x} + \frac{\partial \phi_i^e(x, y)}{\partial y} \right) u_i^e$$

# Shape Function For 3-Node Triangle



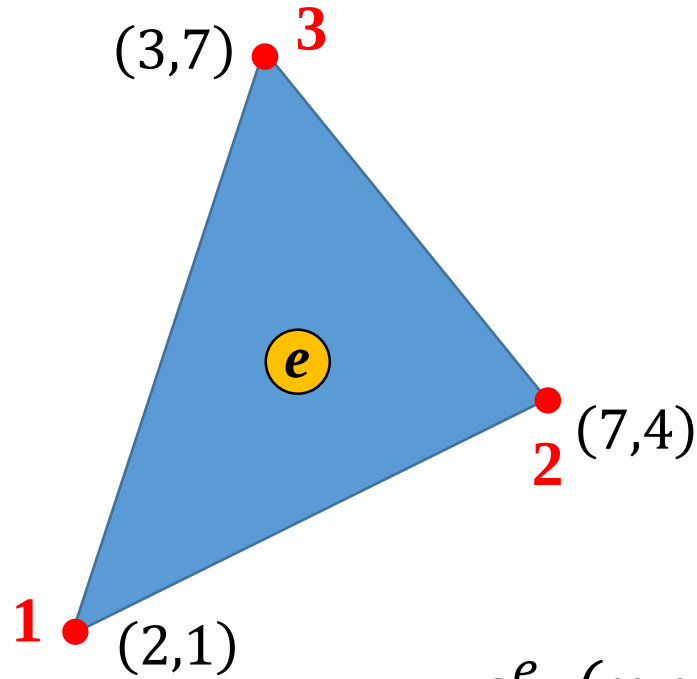
$$\phi_1^e = \frac{1}{27}(37 - 3x - 4y) \quad \phi_2^e = \frac{1}{27}(-11 + 6x - y)$$

$$\phi_3^e = \frac{1}{27}(1 - 3x + 5y)$$

$$\frac{\partial \phi_1^e}{\partial x} = -\frac{3}{27} \quad \frac{\partial \phi_2^e}{\partial x} = \frac{6}{27} \quad \frac{\partial \phi_3^e}{\partial x} = -\frac{3}{27}$$

$$\frac{\partial \phi_1^e}{\partial y} = -\frac{4}{27} \quad \frac{\partial \phi_2^e}{\partial y} = -\frac{1}{27} \quad \frac{\partial \phi_3^e}{\partial y} = \frac{5}{27}$$

# Shape Function For 3-Node Triangle



*Strains*

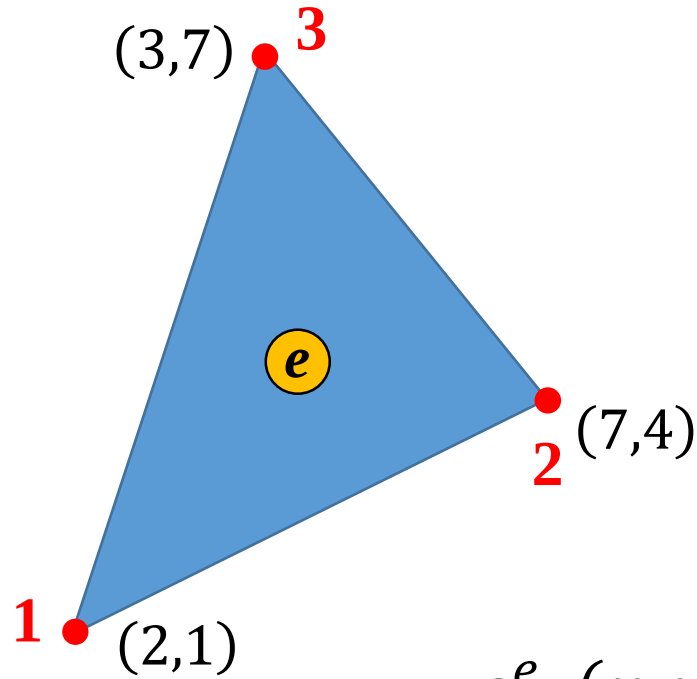
$$u^e(x, y) = \sum_{i=1}^3 \phi_i^e u_i^e$$

$$\epsilon_{xx}^e(x, y) = \frac{\partial u^e(x, y)}{\partial x} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial x} u_i^e$$

$$\epsilon_{yy}^e(x, y) = \frac{\partial u^e(x, y)}{\partial y} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial y} u_i^e$$

$$\epsilon_{xy}^e(x, y) = \frac{1}{2} \left( \frac{\partial u^e(x, y)}{\partial x} + \frac{\partial u^e(x, y)}{\partial y} \right) = \frac{1}{2} \sum_{i=1}^3 \left( \frac{\partial \phi_i^e(x, y)}{\partial x} + \frac{\partial \phi_i^e(x, y)}{\partial y} \right) u_i^e$$

# Shape Function For 3-Node Triangle



*Strains*

$$u^e(x, y) = \sum_{i=1}^3 \phi_i^e u_i^e$$

$$\epsilon_{xx}^e(x, y) = \frac{\partial u^e(x, y)}{\partial x} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial x} u_i^e = \text{Constant}$$

$$\epsilon_{yy}^e(x, y) = \frac{\partial u^e(x, y)}{\partial y} = \sum_{i=1}^3 \frac{\partial \phi_i^e(x, y)}{\partial y} u_i^e = \text{Constant}$$

$$\epsilon_{xy}^e(x, y) = \frac{1}{2} \left( \frac{\partial u^e(x, y)}{\partial x} + \frac{\partial u^e(x, y)}{\partial y} \right) = \frac{1}{2} \sum_{i=1}^3 \left( \frac{\partial \phi_i^e(x, y)}{\partial x} + \frac{\partial \phi_i^e(x, y)}{\partial y} \right) u_i^e$$

= Constant

*Constant Strains Triangle (CST)*



**THANK  
YOU !**



## **NPTel ONLINE CERTIFICATION COURSES**

# **Finite Element Method**

## **Week 7: 2D Formulation**

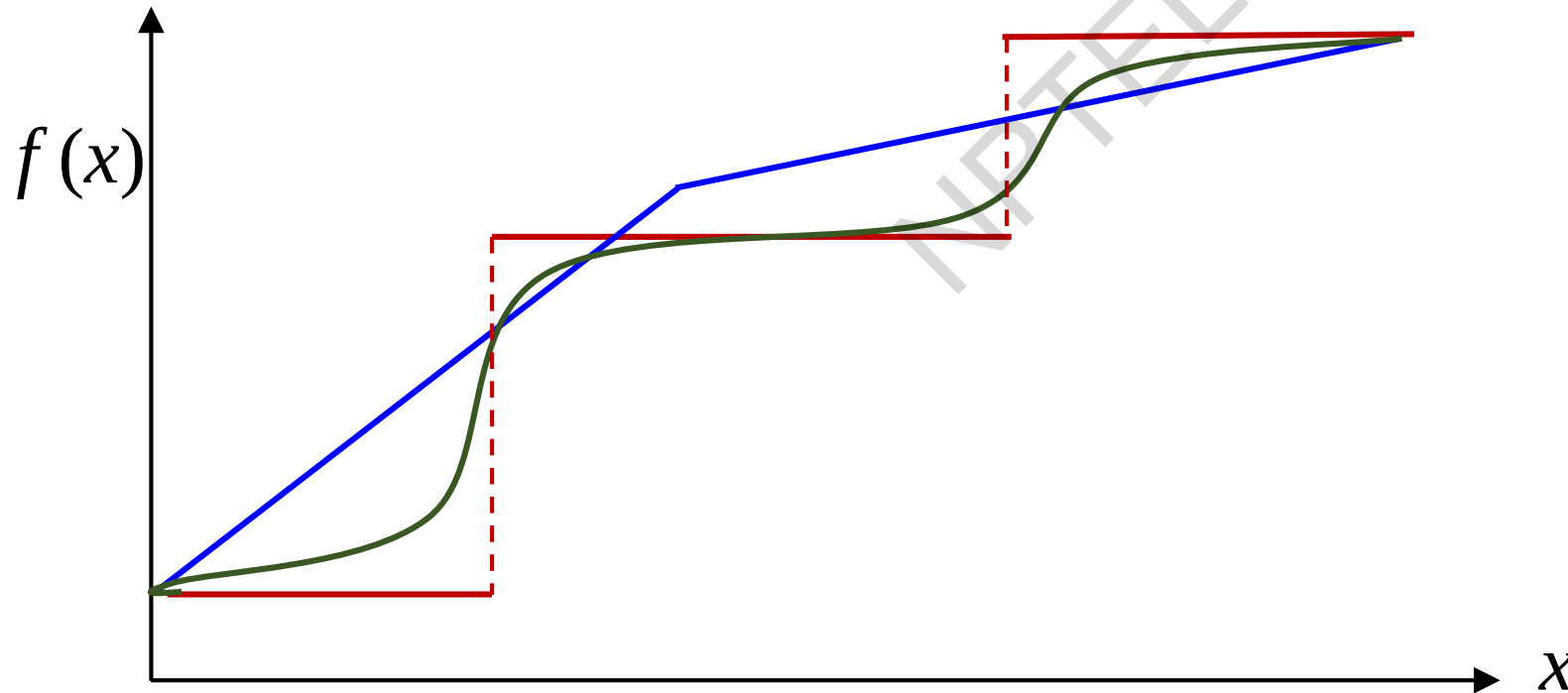
### **Lecture 33: Continuity and completeness**

**Biswanath Banerjee**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

**Amit Shaw**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

# Continuity

A function  $f$  defined over a domain  $\Omega$  is called a  $C^n$  function if its derivatives of order  $j$  for  $0 \leq j \leq n$  exist and are continuous function in the entire domain.



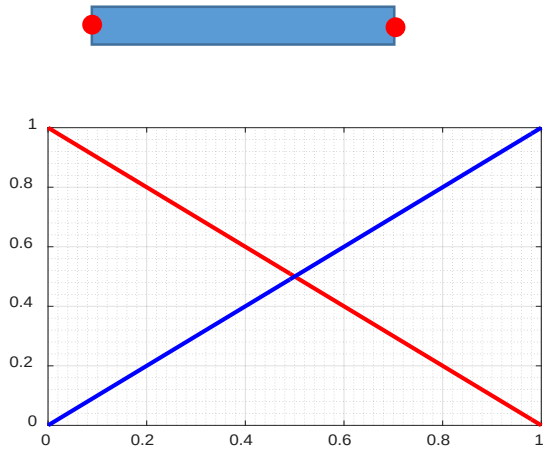
# Global Continuity

$$u^e(\bar{x}) = \sum_{i=1}^n \phi_i^e(\bar{x}) u_i^e$$

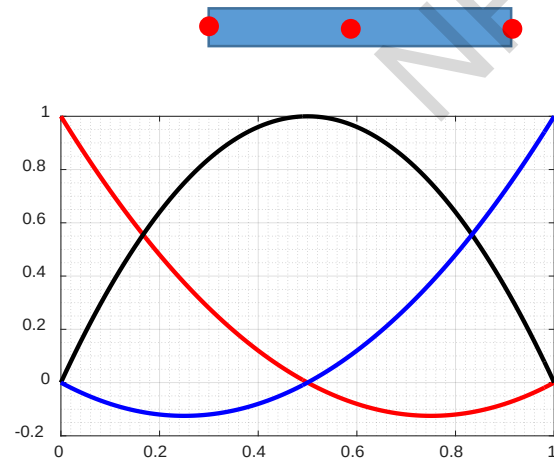
*Element level approximation*

$n$  = number of node in the element

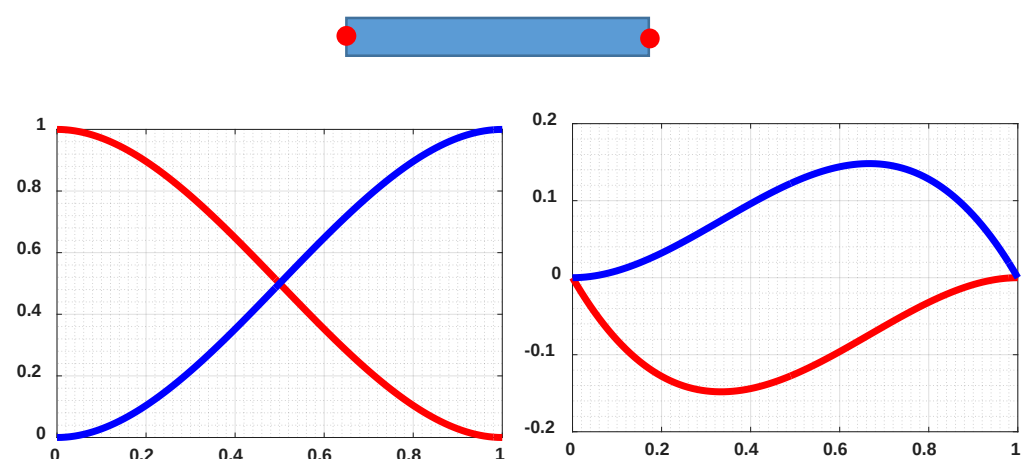
$\phi_i^e$  = shape function within element



2 – node bar



3 – node bar



2 – node beam

# Global Continuity

$$u^e(\bar{x}) = \sum_{i=1}^n \phi_i^e(\bar{x}) u_i^e$$

*Element level approximation*

n = number of node in the element

$\phi_i^e$  = shape function within element

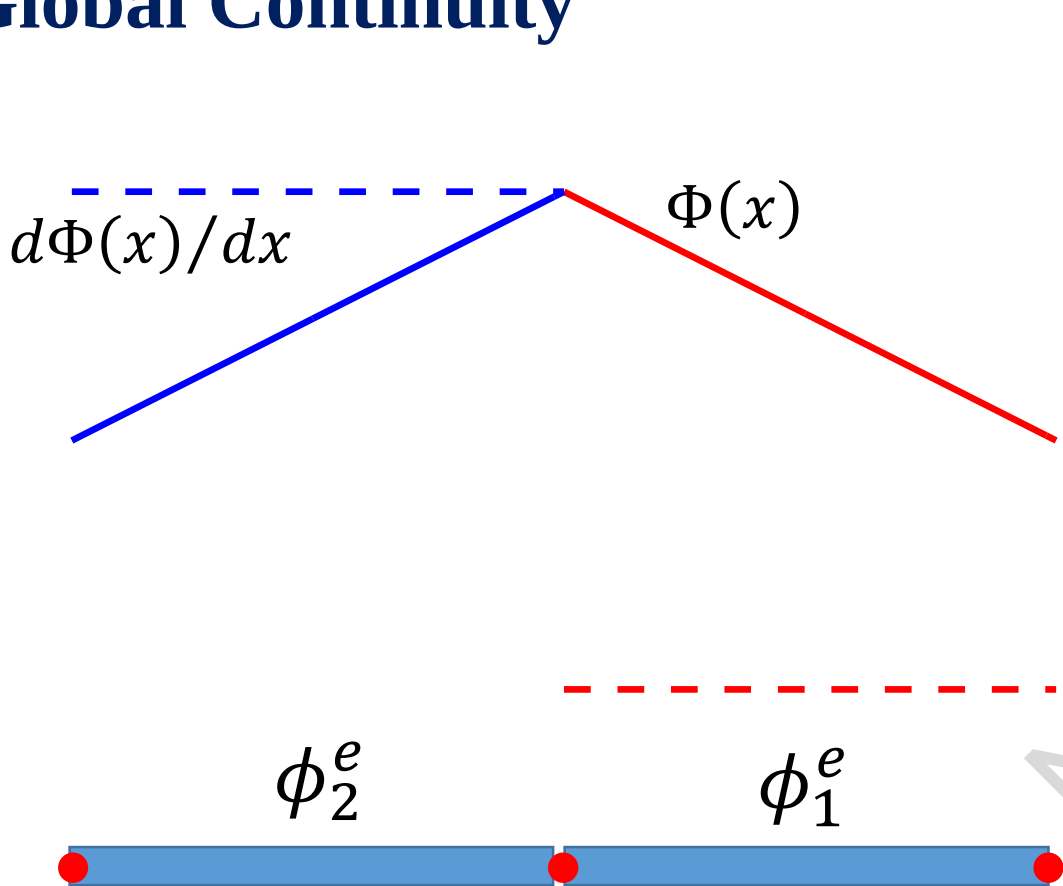
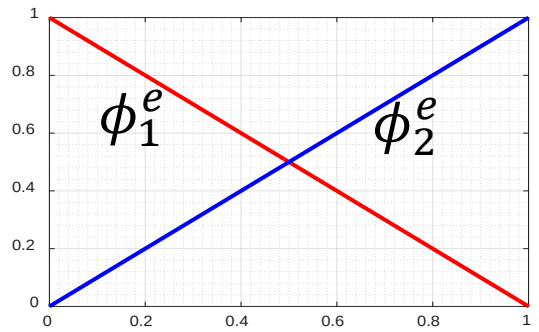
$$u(x) = \sum_{i=1}^N u_i \Phi_i(x)$$

*Global approximation*

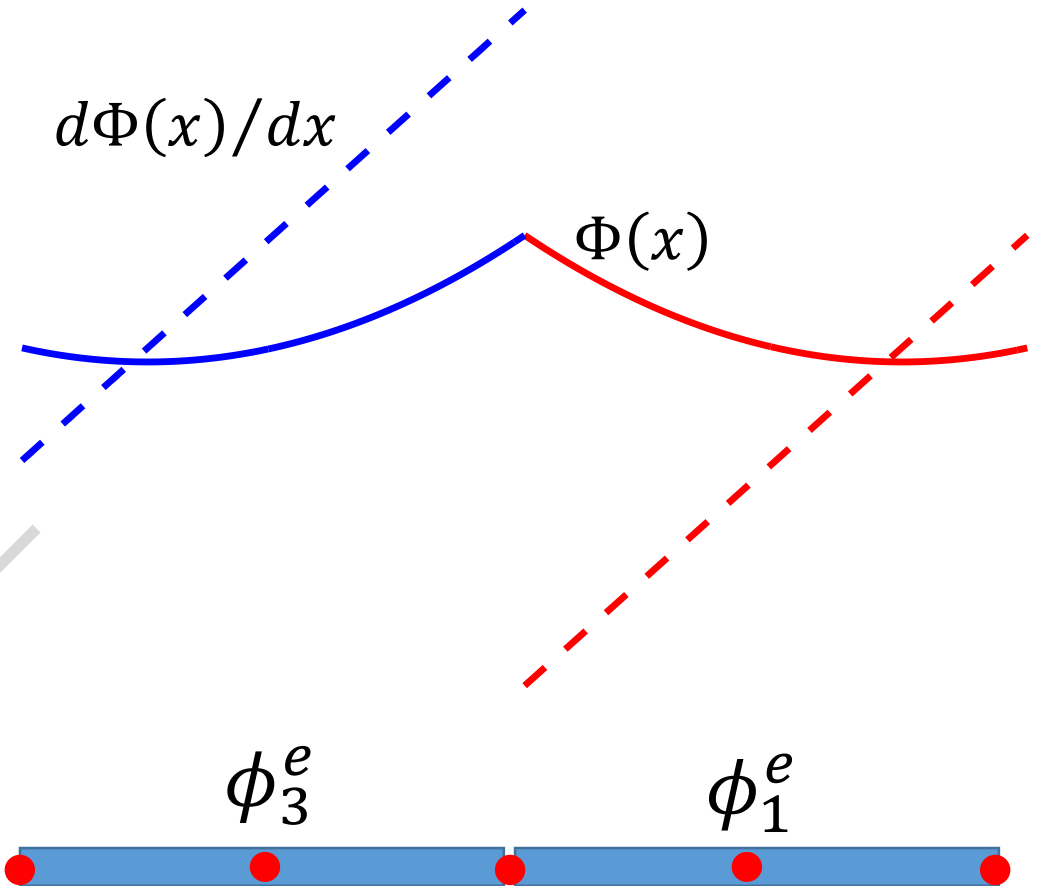
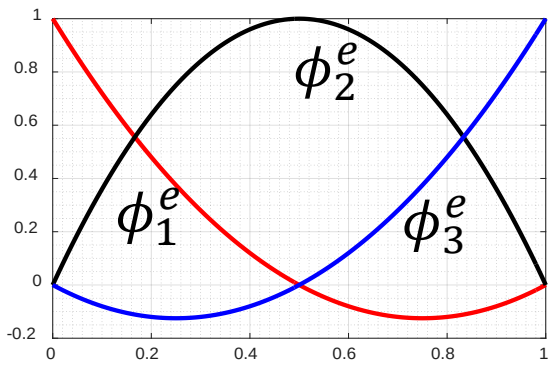
N = total number of node

$\Phi_i$  = global shape function

# Global Continuity

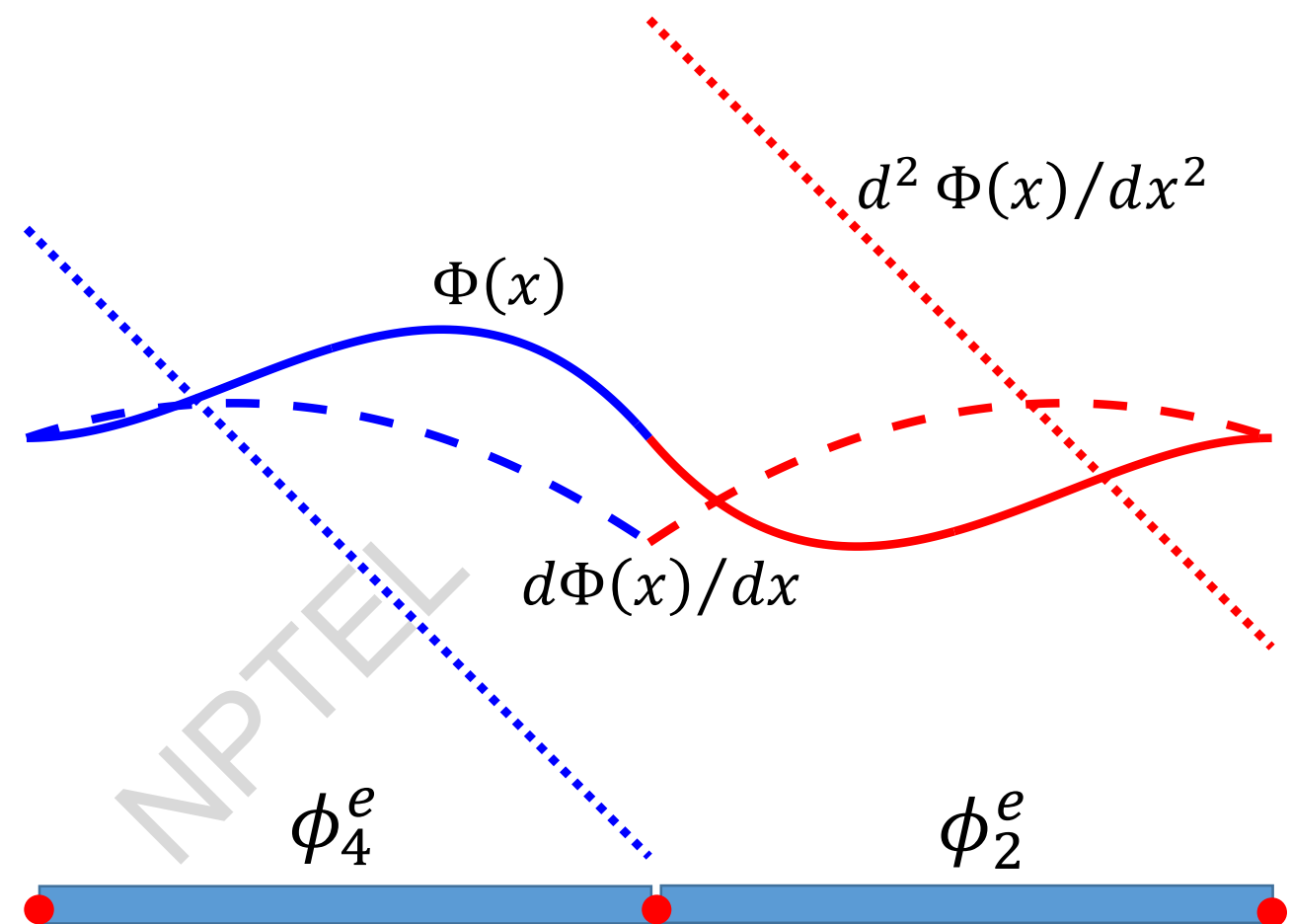
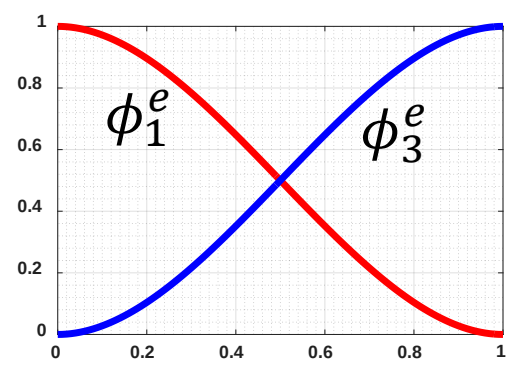
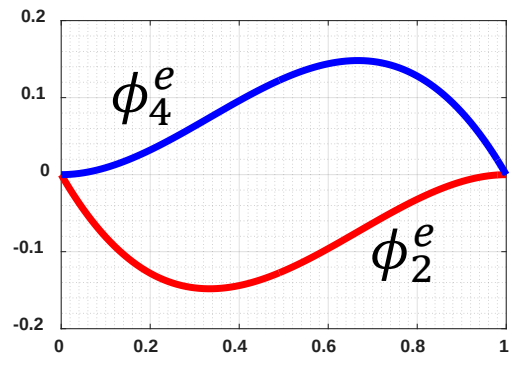


2 – node bar  
 $C^0$  continuity



3 – node bar  
 $C^0$  continuity

# Global Continuity



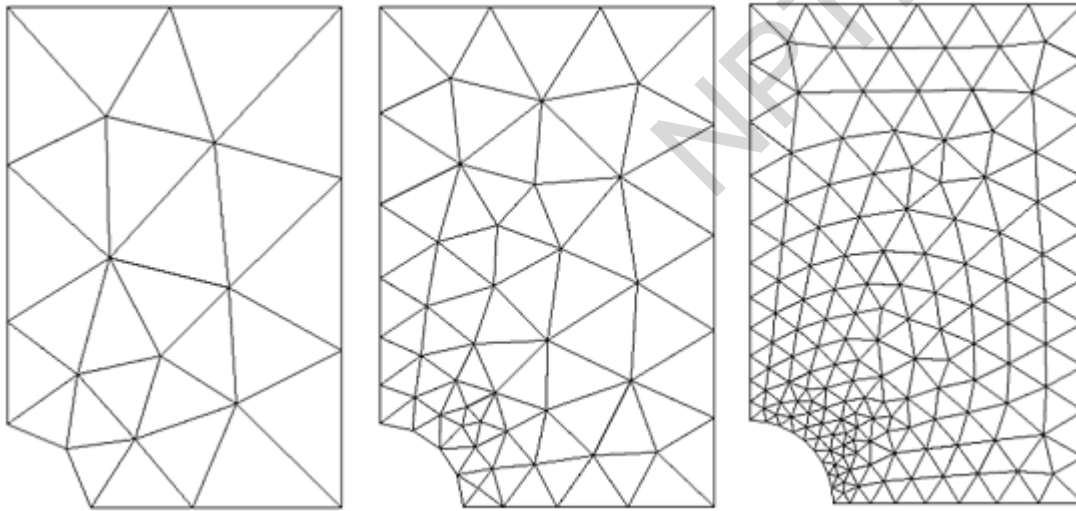
2 – node beam

$C^1$  continuity

# Error in Finite Element Solutions: *Major Sources*

- *Domain approximation error*

Idealization/ approximation of computational domain  
(not of much relevance in one-dimensional problems)



# Error in Finite Element Solutions: *Major Sources*

- *Quadrature and finite arithmetic error*

Round-off errors in the computation of numbers

Numerical evaluation of integrals

NPTTEL

# Error in Finite Element Solutions: *Major Sources*

- *Approximation error*

Approximation of variables

$$u_h = \sum_{i=1}^N u_i \Phi_i$$

$N$  = Total number of nodes

$u_i = u$  at global node  $i$

$\Phi_i$  = Global interpolation function associated with global node  $i$

# Error in Finite Element Solutions: *Measures*

## *Supmetric measure*

$$\| u - u_h \|_{\infty} = \max_{a \leq x \leq b} |u - u_h|$$

$u_h$  is the finite element solution over domain  $\Omega = (a, b)$

$u$  is true solution

## *Energy norm*

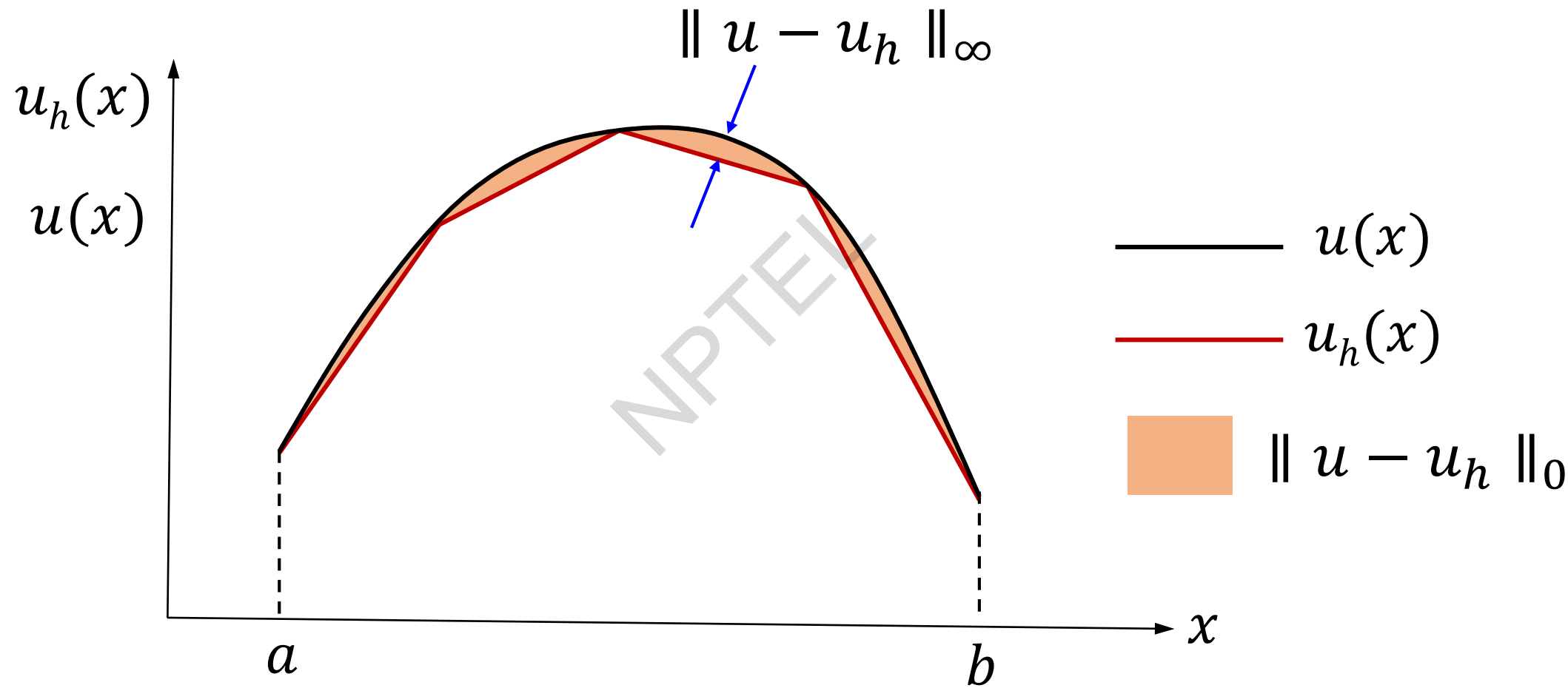
$$\| u - u_h \|_m = \left( \int_a^b \sum_{i=0}^m \left[ \frac{d^i u}{dx^i} - \frac{d^i u_h}{dx^i} \right]^2 dx \right)^{1/2}$$

$2m$  is the order of differential equation being solved

## *$L_2$ norm*

$$\| u - u_h \|_0 = \left( \int_a^b |u - u_h|^2 dx \right)^{1/2}$$

# Error in Finite Element Solutions: *Measures*



# Convergence in Finite Element Solutions

If the finite element interpolation functions  $\Phi_i$  are **complete polynomials** of degree  $k$ , the error in the energy norm can be shown as

$$\|u - u_h\|_m \leq ch^p, \quad p = k + 1 - m > 0$$

$h$  is the element size

$2m$  is the order of differential equation  
being solved

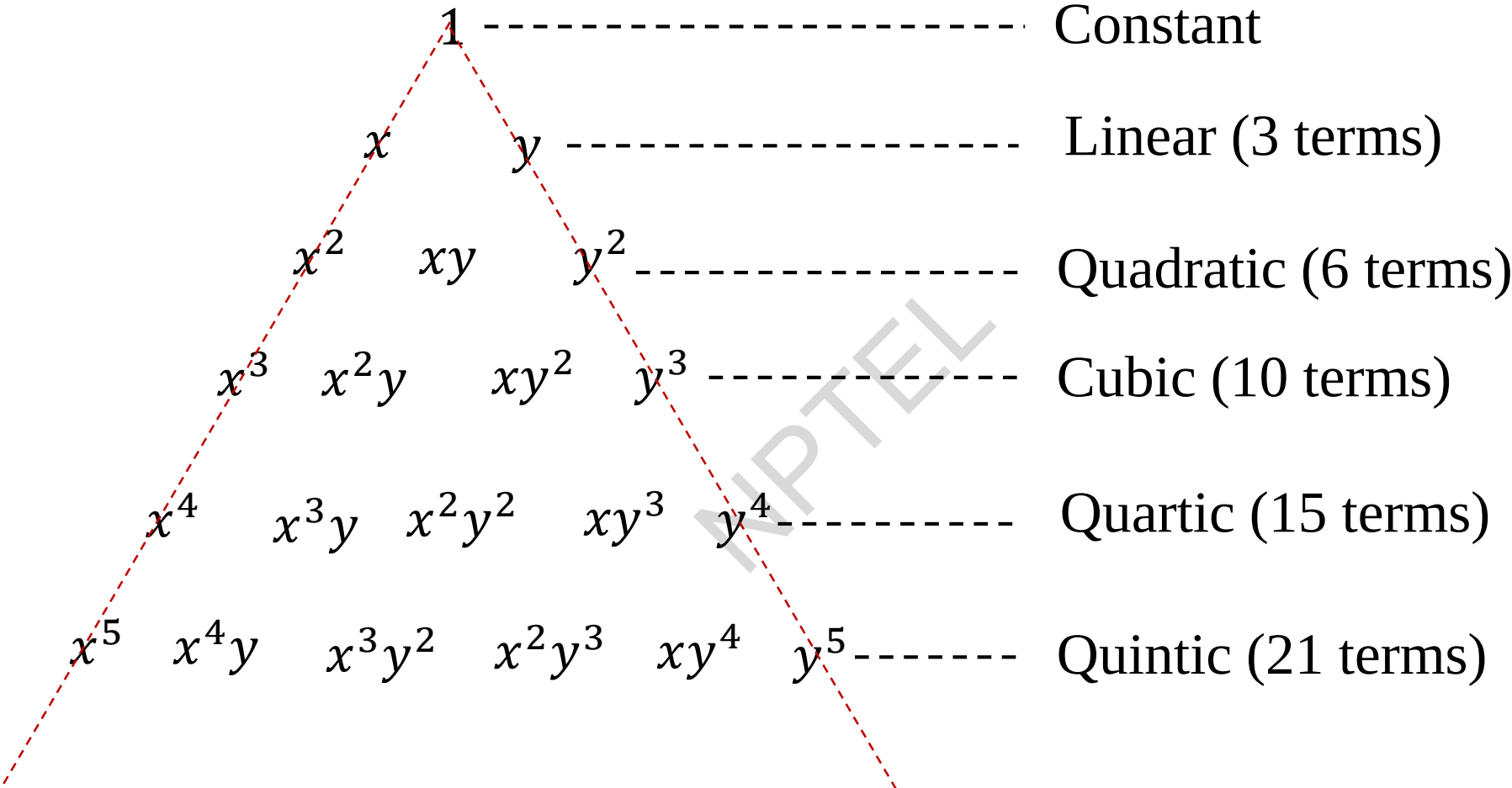
$c$  is a constant

Completeness – include all  
lower-order terms up to the  
highest order used

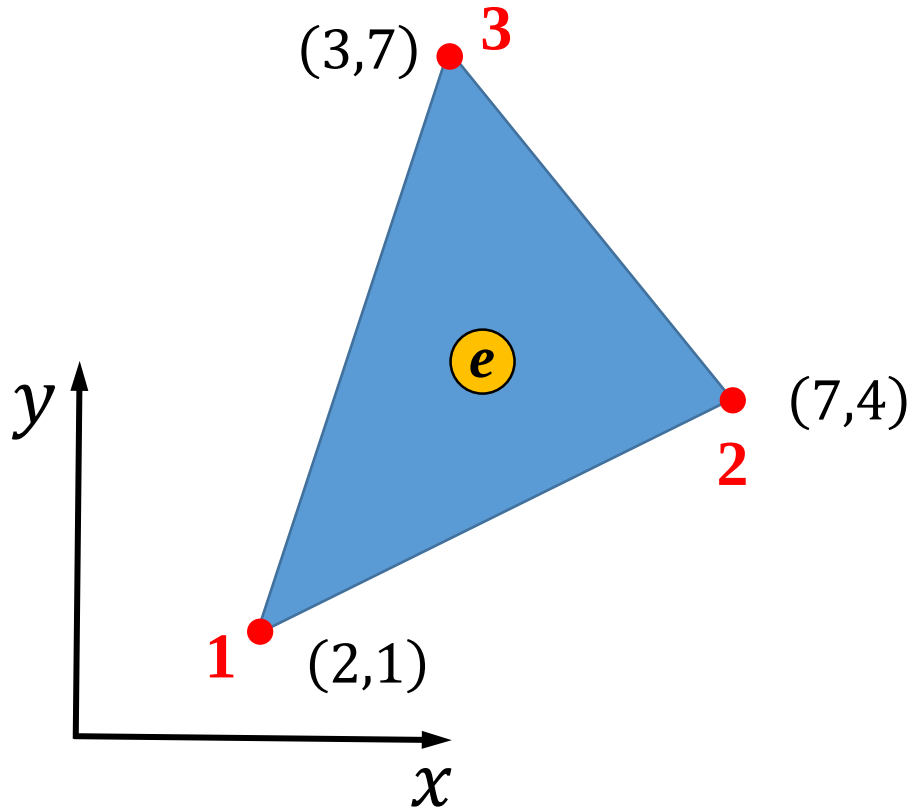
$$f(x) = c_0 + c_1x + c_2x^2 + c_4x^4$$

$$k = 2$$

# Pascal Triangle in 2D



# Let us take incomplete polynomial



*Polynomial approximation*

$$u^e(x, y) = c_2x + c_3y + c_4xy$$

*Interpolation*

$$u_1^e = u^e(x_1^e, y_1^e) = c_2x_1^e + c_3y_1^e + c_4x_1^ey_1^e$$

$$u_2^e = u^e(x_2^e, y_2^e) = c_2x_2^e + c_3y_2^e + c_4x_2^ey_2^e$$

$$u_3^e = u^e(x_3^e, y_3^e) = c_2x_3^e + c_3y_3^e + c_4x_3^ey_3^e$$

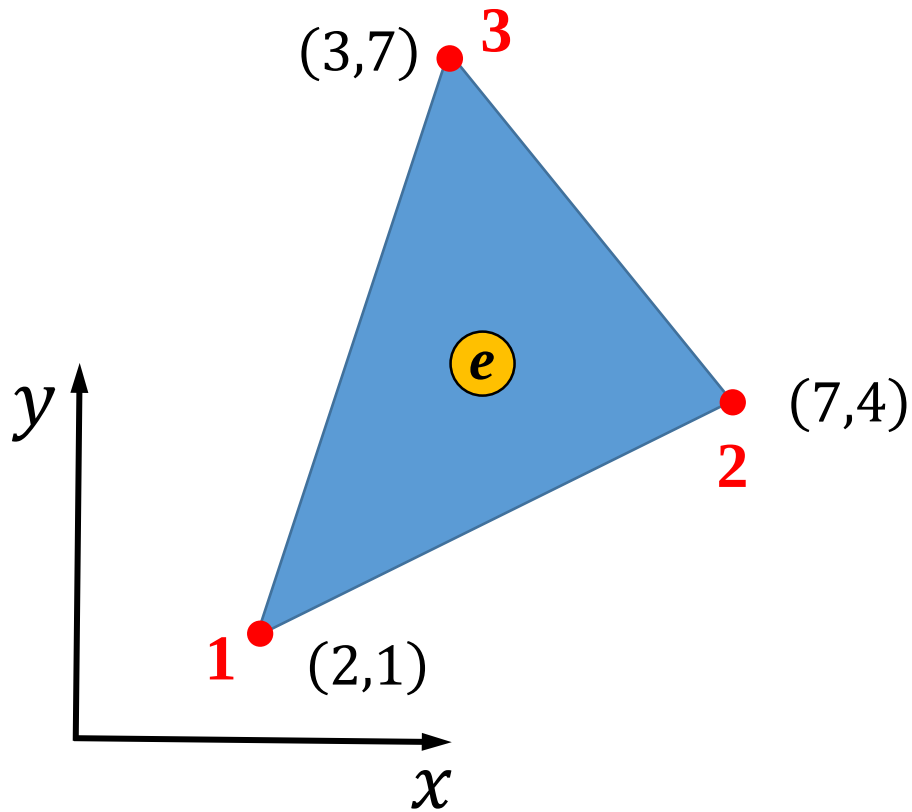
$$u_1^e = 2c_2 + c_3 + 2c_4$$

$$u_2^e = 7c_2 + 4c_3 + 28c_4$$

$$u_3^e = 3c_2 + 7c_3 + 21c_4$$

$$\Rightarrow u^e(x, y) = \{x \quad y \quad xy\} \begin{bmatrix} 2 & 1 & 2 \\ 7 & 4 & 28 \\ 3 & 7 & 21 \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

Let us take incomplete polynomial

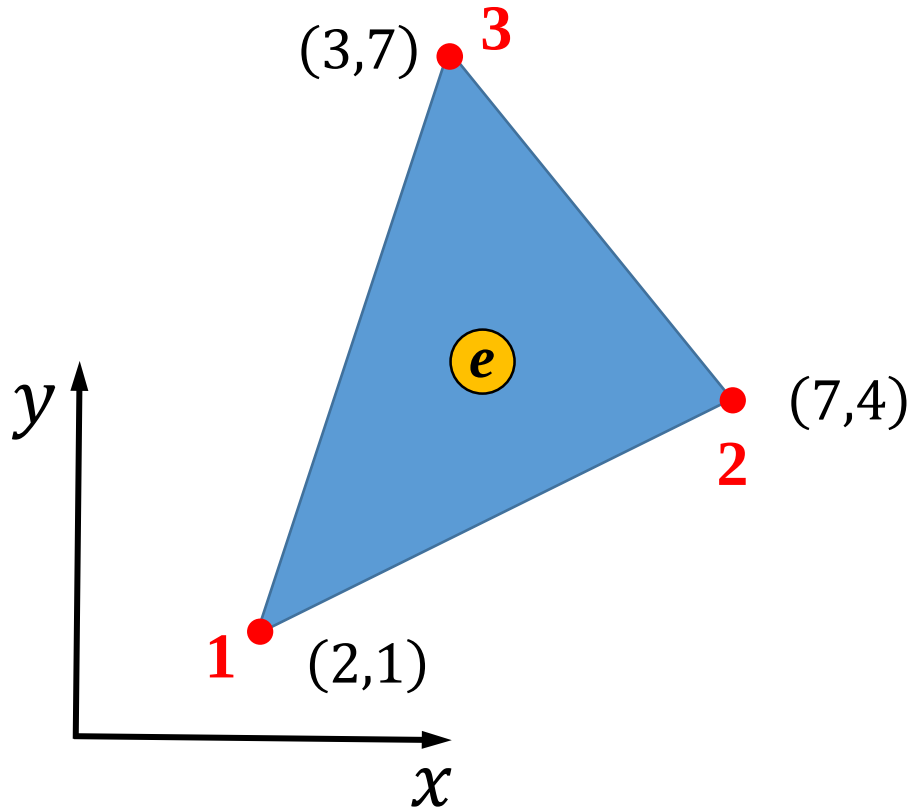


$$u^e(x, y) = \{x \quad y \quad xy\} \begin{bmatrix} 2 & 1 & 2 \\ 7 & 4 & 28 \\ 3 & 7 & 21 \end{bmatrix}^{-1} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\Rightarrow u^e(x, y)$$

$$= \{x \quad y \quad xy\} \frac{1}{213} \begin{bmatrix} 112 & 7 & -20 \\ 63 & -36 & 42 \\ -37 & 11 & -1 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

## Let us take incomplete polynomial



$$u^e(x, y)$$

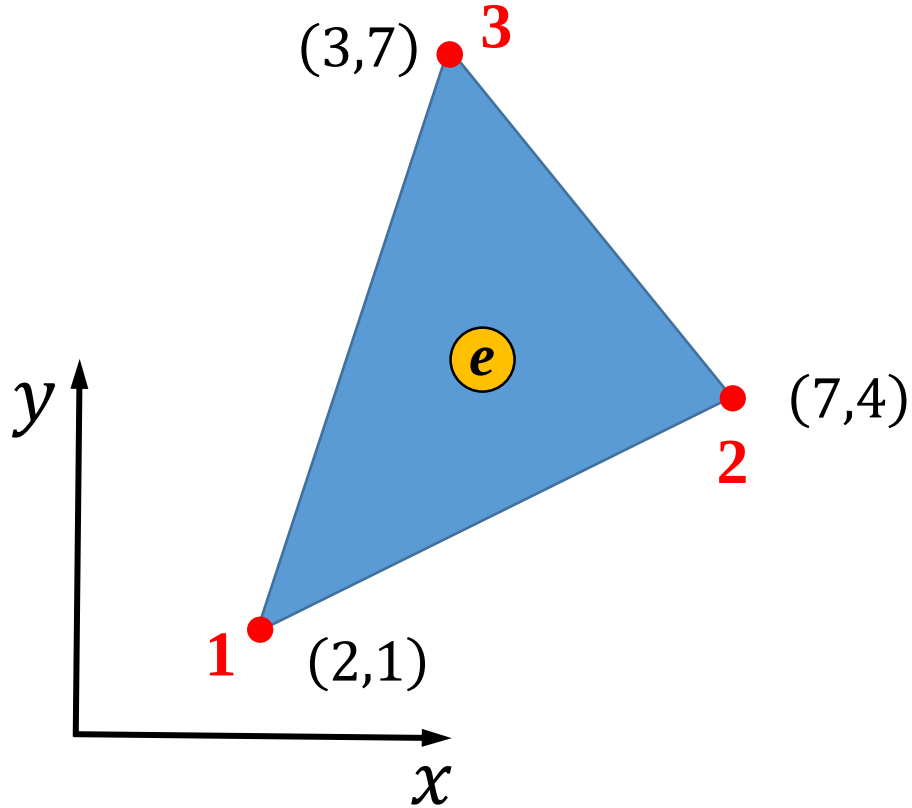
$$= \{x \quad y \quad xy\} \frac{1}{213} \begin{bmatrix} 112 & 7 & -20 \\ 63 & -36 & 42 \\ -37 & 11 & -1 \end{bmatrix} \begin{Bmatrix} u_1^e \\ u_2^e \\ u_3^e \end{Bmatrix}$$

$$\phi_1^e = \frac{1}{213} (112x + 63y - 37xy)$$

$$\phi_2^e = \frac{1}{213} (7x - 36y + 11xy)$$

$$\phi_3^e = \frac{1}{213} (-20x + 42y - xy)$$

## Let us take incomplete polynomial



$$\phi_1^e = \frac{1}{213} (112x + 63y - 37xy)$$

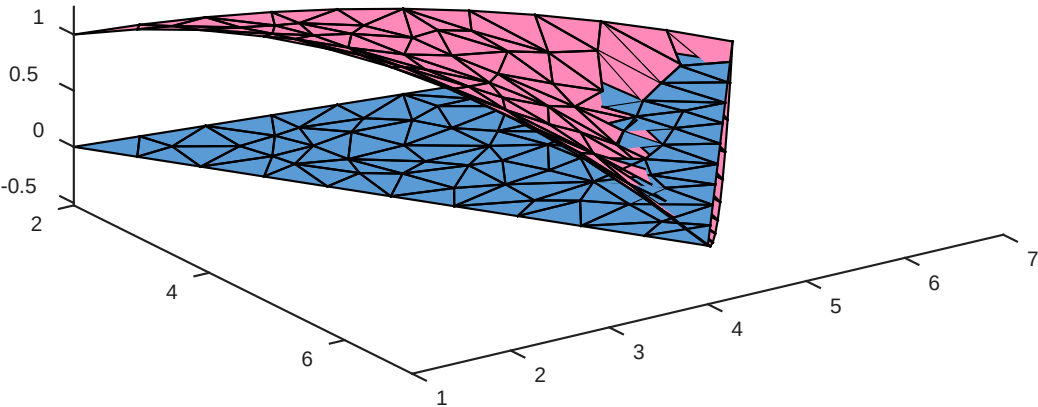
$$\phi_2^e = \frac{1}{213} (7x - 36y + 11xy)$$

$$\phi_3^e = \frac{1}{213} (-20x + 42y - xy)$$

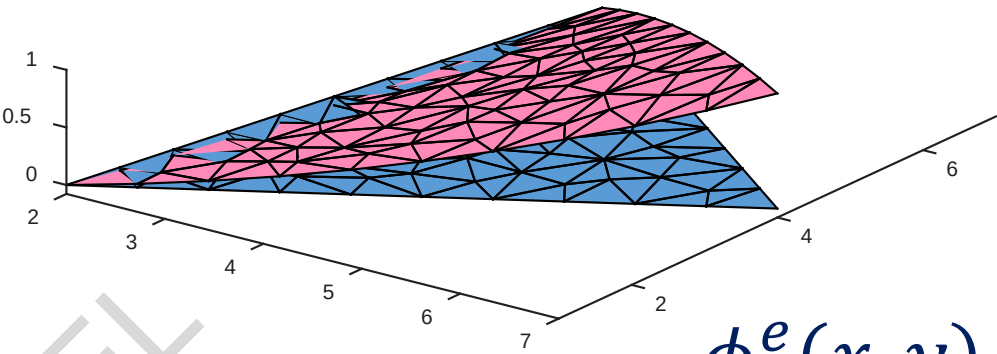
$$\phi_i^e(x_j^e, y_j^e) = \delta_{ij}$$

Unnecessary to check – interpolation is used as constraint to determine constants  $c_2$ ,  $c_3$  and  $c_4$

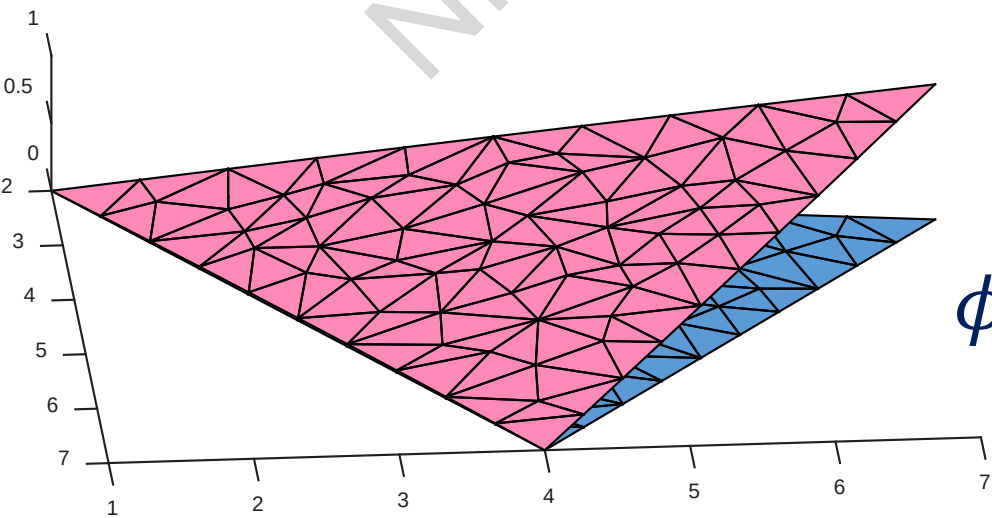
# Let us take incomplete polynomial



$$\phi_1^e(x, y)$$

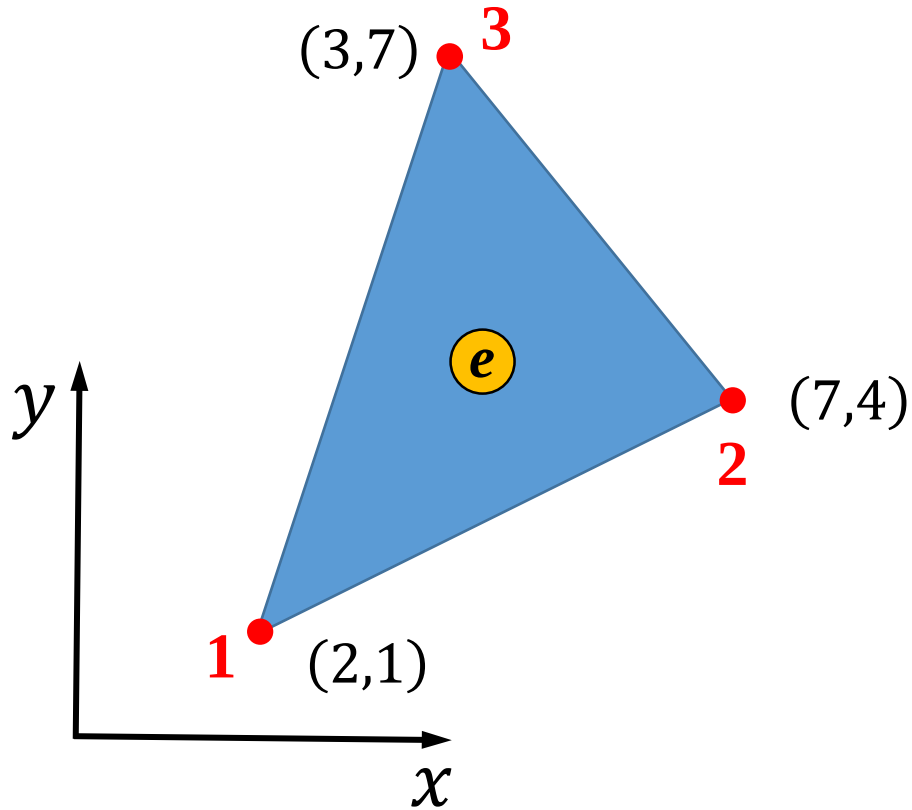


$$\phi_2^e(x, y)$$



$$\phi_3^e(x, y)$$

## Let us take incomplete polynomial



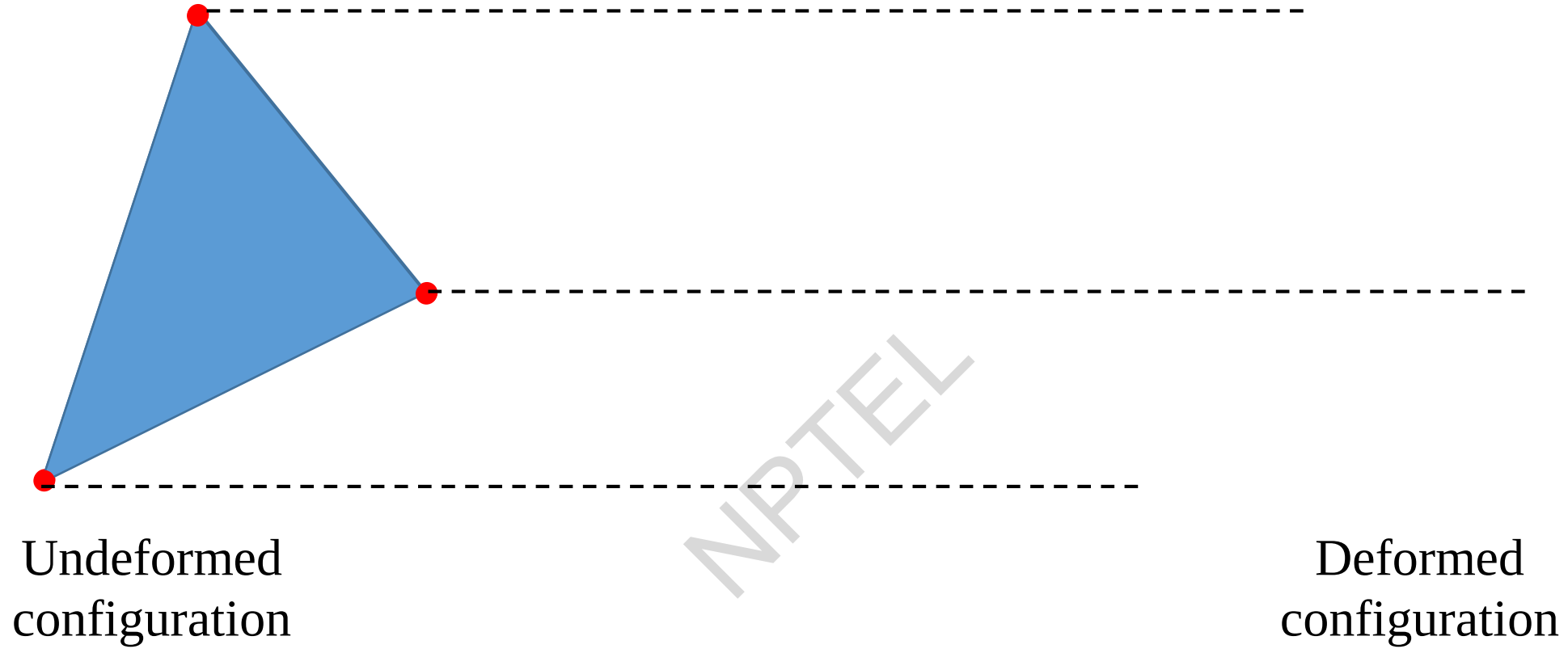
$$\phi_1^e = \frac{1}{213} (112x + 63y - 37xy)$$

$$\phi_2^e = \frac{1}{213} (7x - 36y + 11xy)$$

$$\phi_3^e = \frac{1}{213} (-20x + 42y - xy)$$

$$\phi_1^e + \phi_2^e + \phi_3^e = \frac{1}{213} (99x + 69y - 27xy) \neq 1$$

# Rigid body motion





**THANK  
YOU !**



**NPTel ONLINE CERTIFICATION COURSES**

# **Finite Element Method**

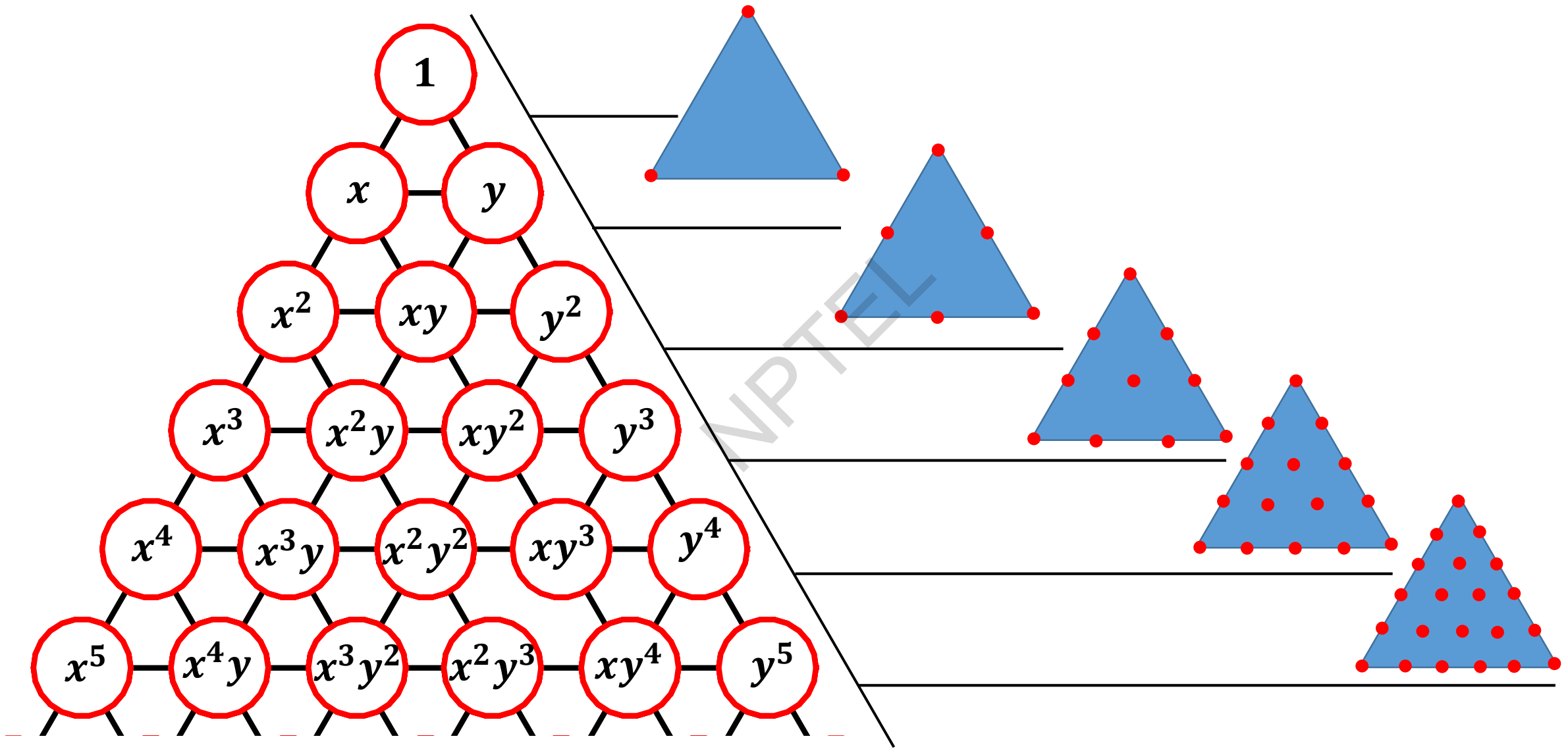
**Week 7: 2D Formulation**

**Lecture 34: Shape Functions**

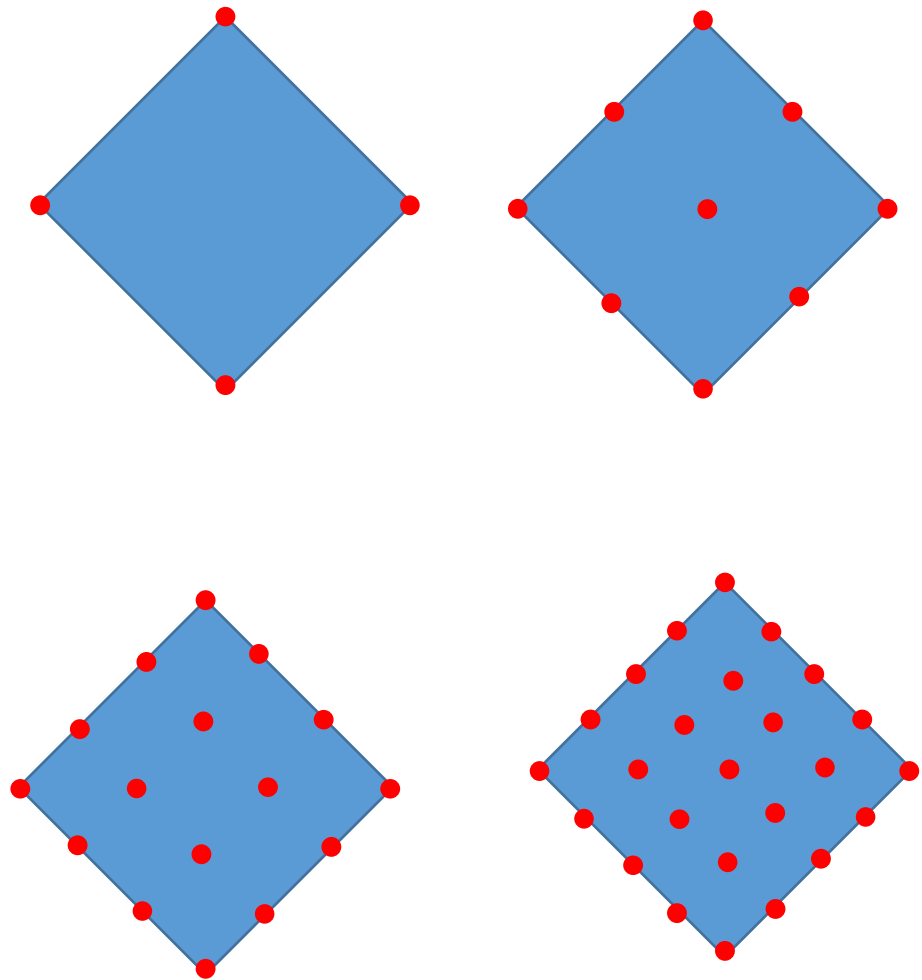
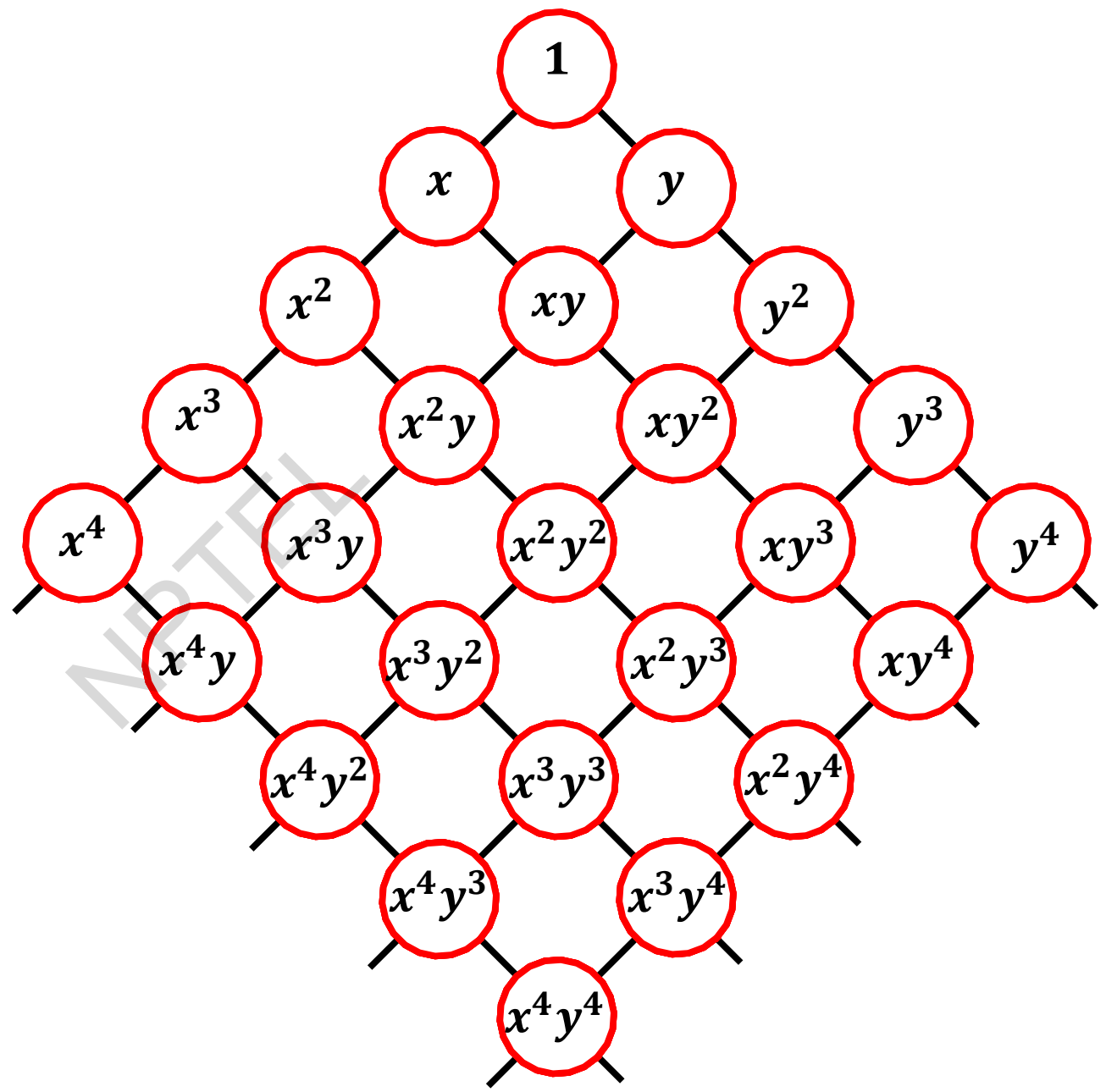
**Biswanath Banerjee**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

**Amit Shaw**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

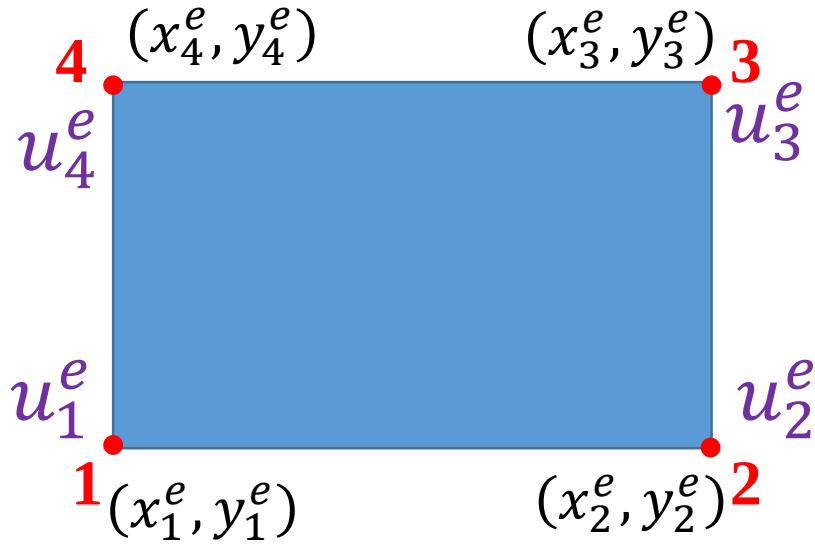
# Lagrange Simplexes



# Lagrange Hypercubes



# Four-node Rectangular Element



## *Polynomial approximation*

$$u^e(x, y) = c_1 + c_2x + c_3y + c_4xy$$

Constants  $c_1, c_2, c_3$  and  $c_4$  are determined in terms of  $u_1^e, u_2^e, u_3^e$  and  $u_4^e$  by imposing interpolation condition  $u^e(x_i, y_i) = u_i^e$  for all  $i = 1, 2, 3, 4$

Very tedious job especially for higher order shape functions – *need a different approach*

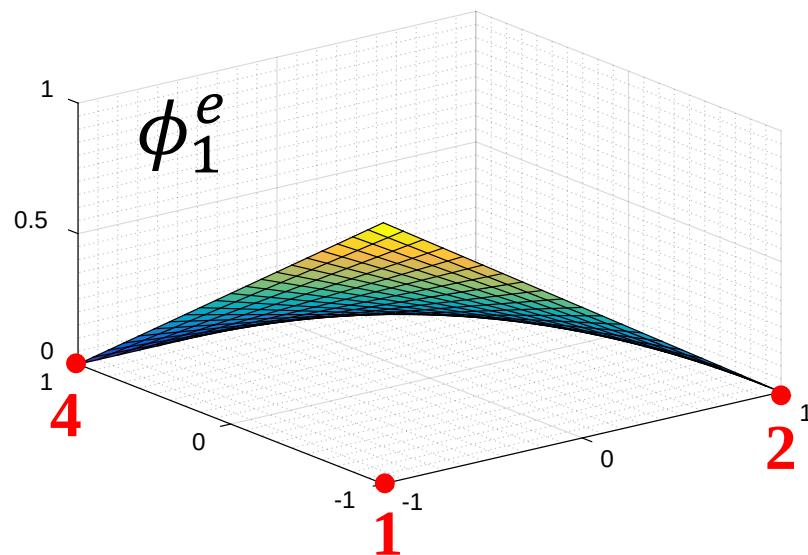
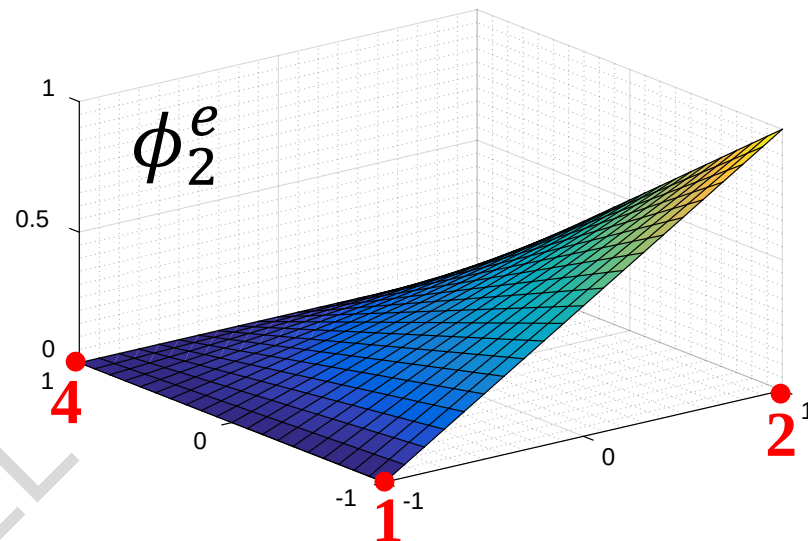
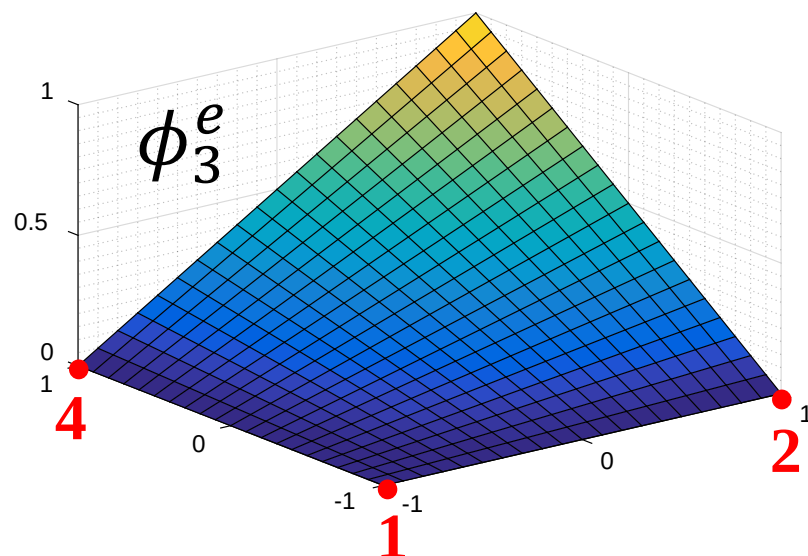
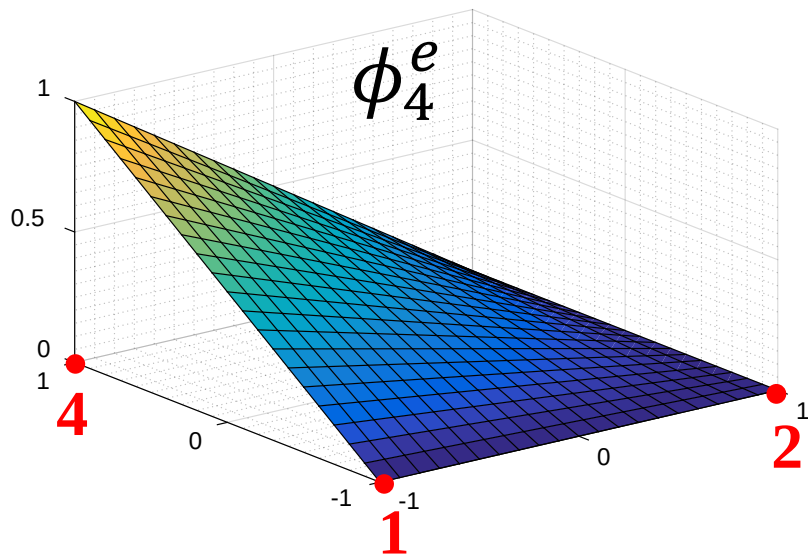
$$\phi_1^e = \frac{1}{A^e} (x - x_2^e)(y - y_4^e)$$

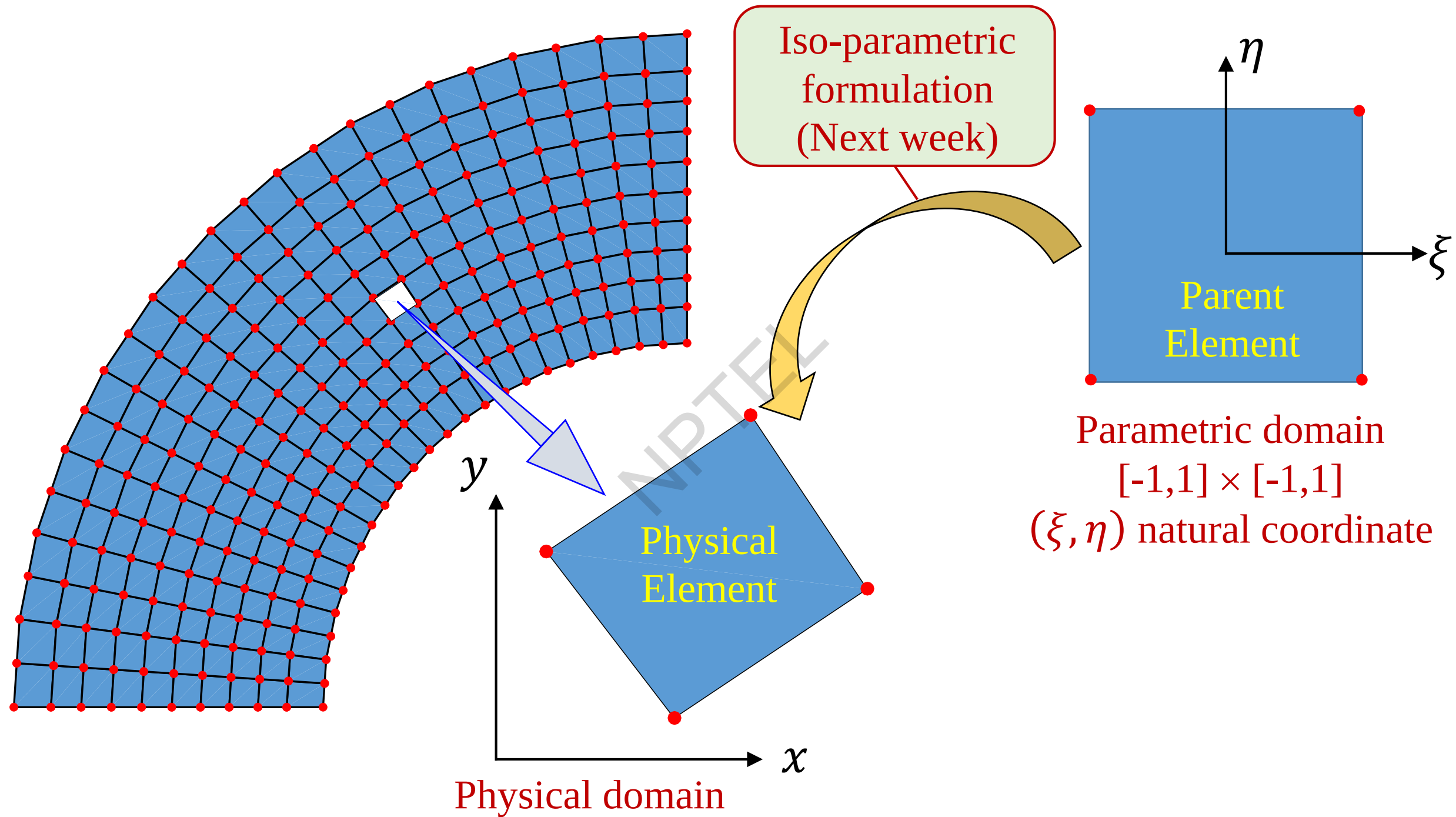
$$\phi_3^e = \frac{1}{A^e} (x - x_1^e)(y - y_1^e)$$

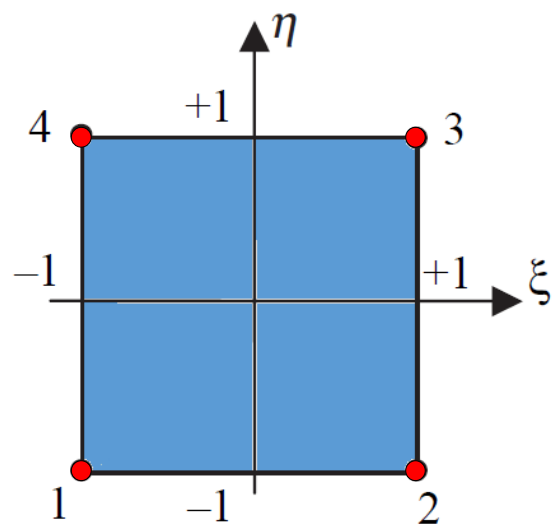
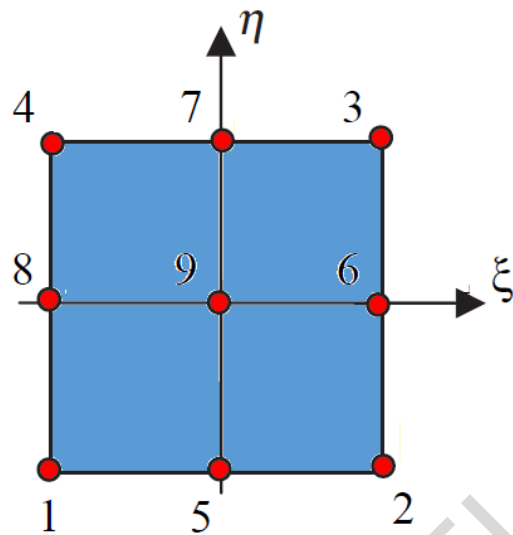
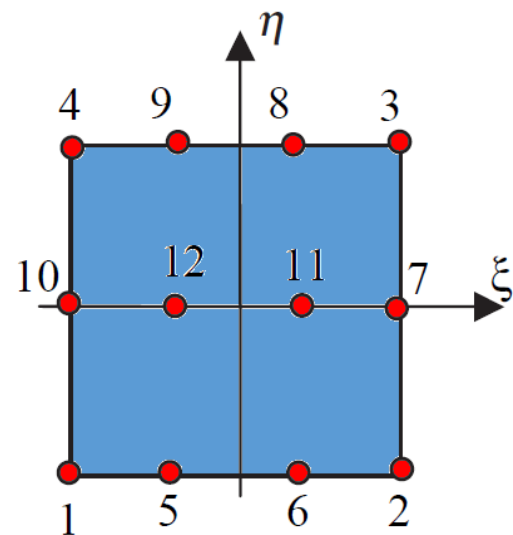
$$\phi_2^e = -\frac{1}{A^e} (x - x_1^e)(y - y_4^e)$$

$$\phi_4^e = -\frac{1}{A^e} (x - x_2^e)(y - y_1^e)$$

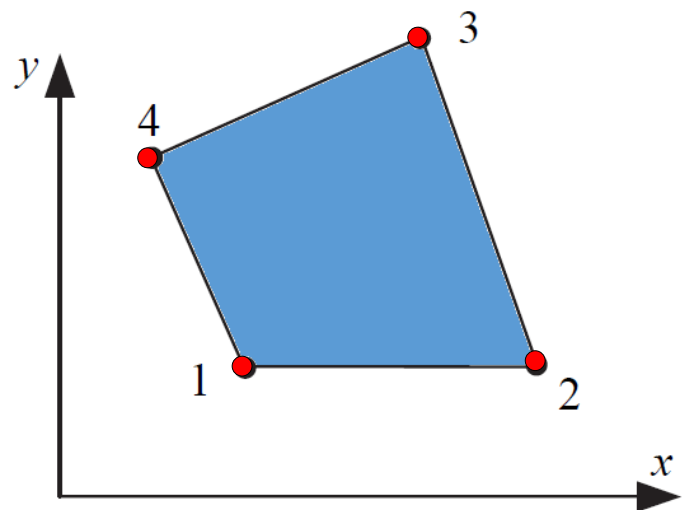
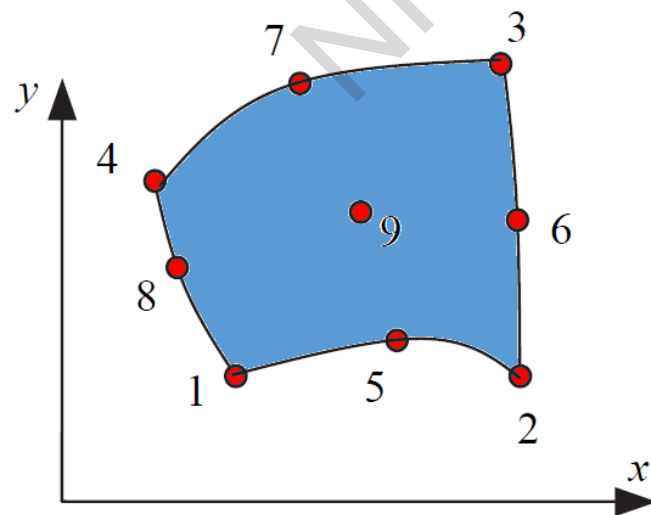
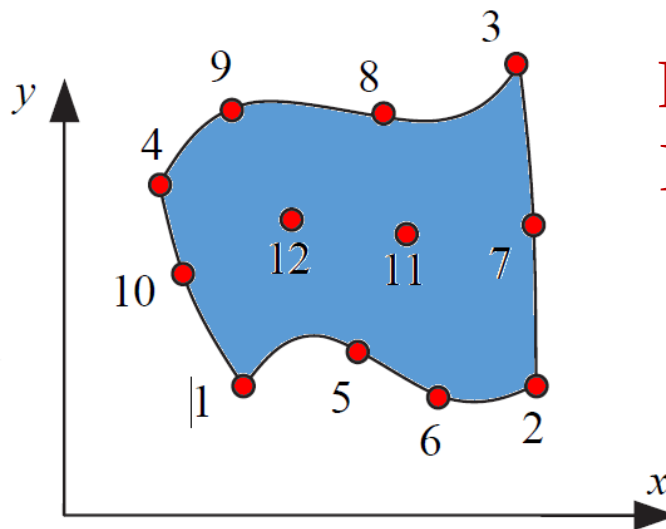
# Four-node Rectangular Element



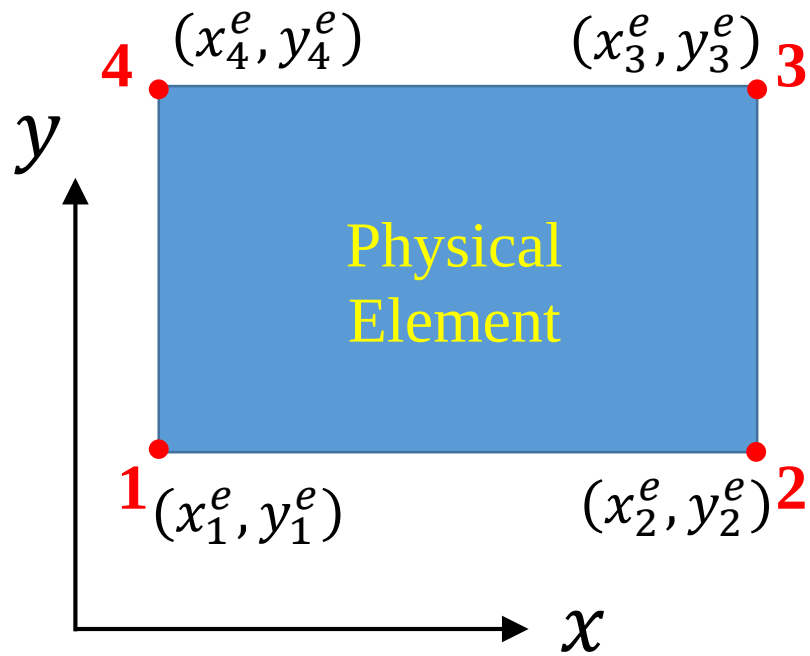




Parent  
Element



Physical  
Element

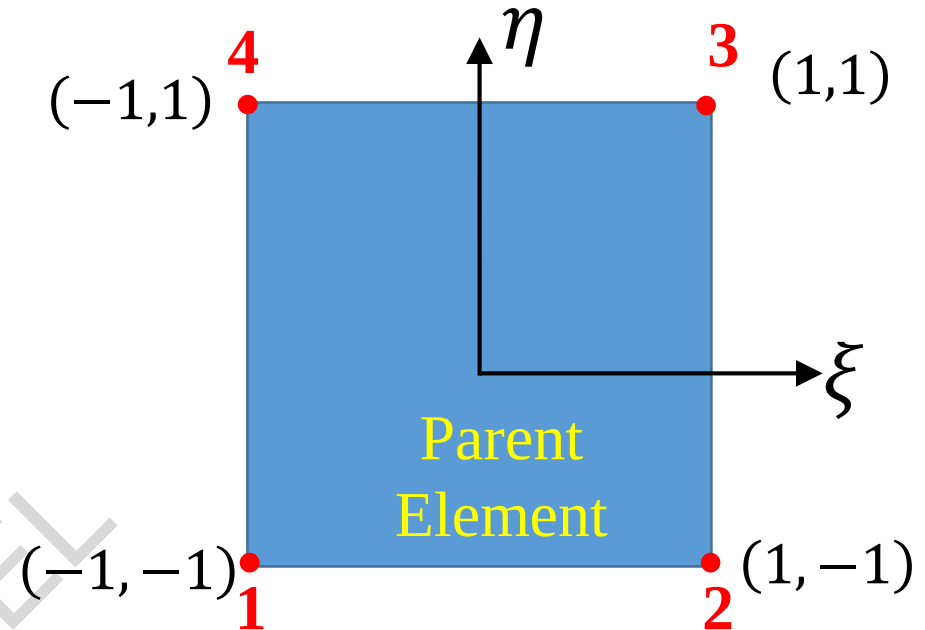


$$\phi_1^e(x, y) = (x - x_2^e)(y - y_4^e)/A^e$$

$$\phi_2^e(x, y) = -(x - x_1^e)(y - y_4^e)/A^e$$

$$\phi_3^e(x, y) = (x - x_1^e)(y - y_1^e)/A^e$$

$$\phi_4^e(x, y) = -(x - x_2^e)(y - y_1^e)/A^e$$



$$\phi_1^e(\xi, \eta) = (1 - \xi)(1 - \eta)/4$$

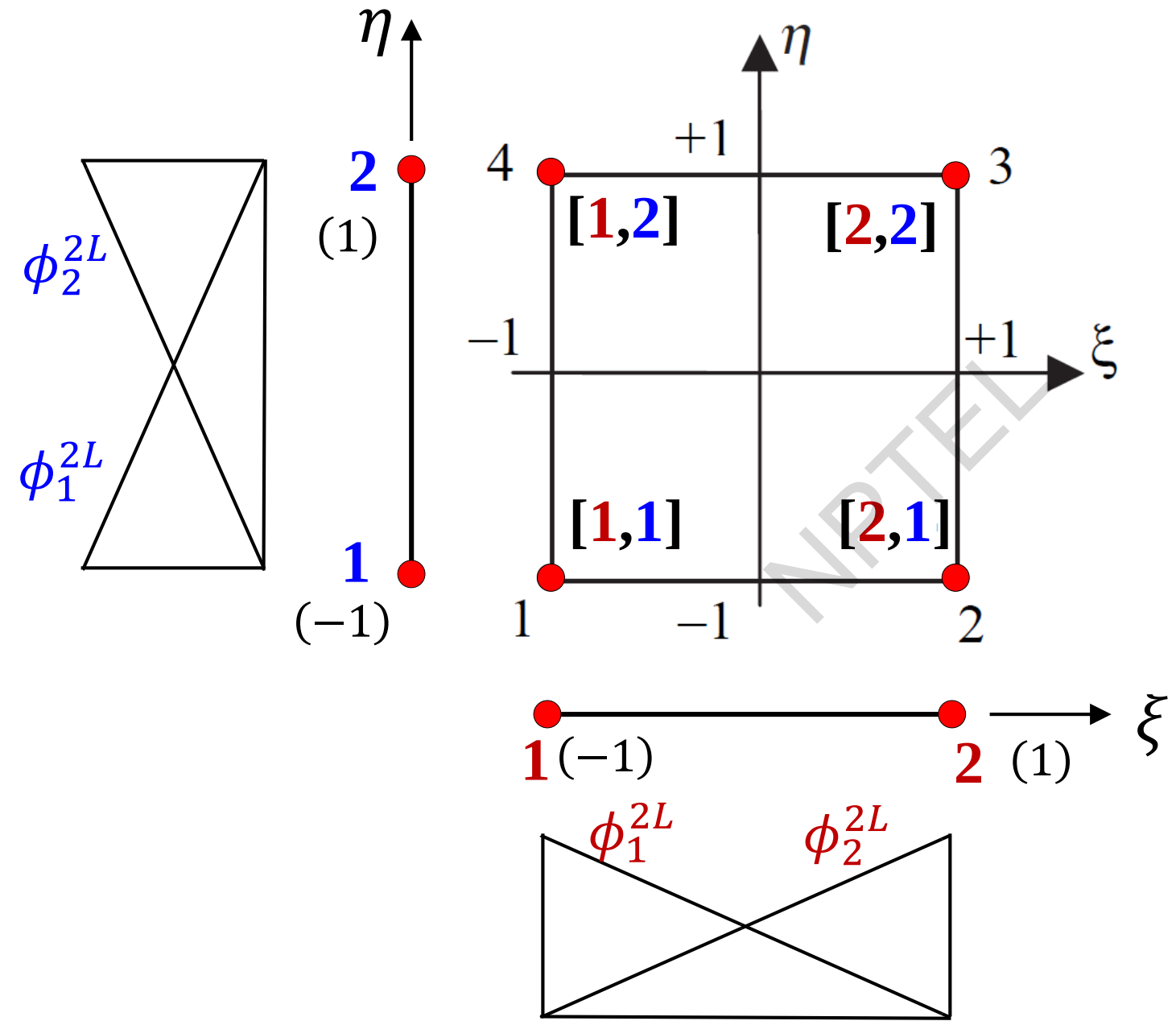
$$\phi_2^e(\xi, \eta) = (1 + \xi)(1 - \eta)/4$$

$$\phi_3^e(\xi, \eta) = (1 + \xi)(1 + \eta)/4$$

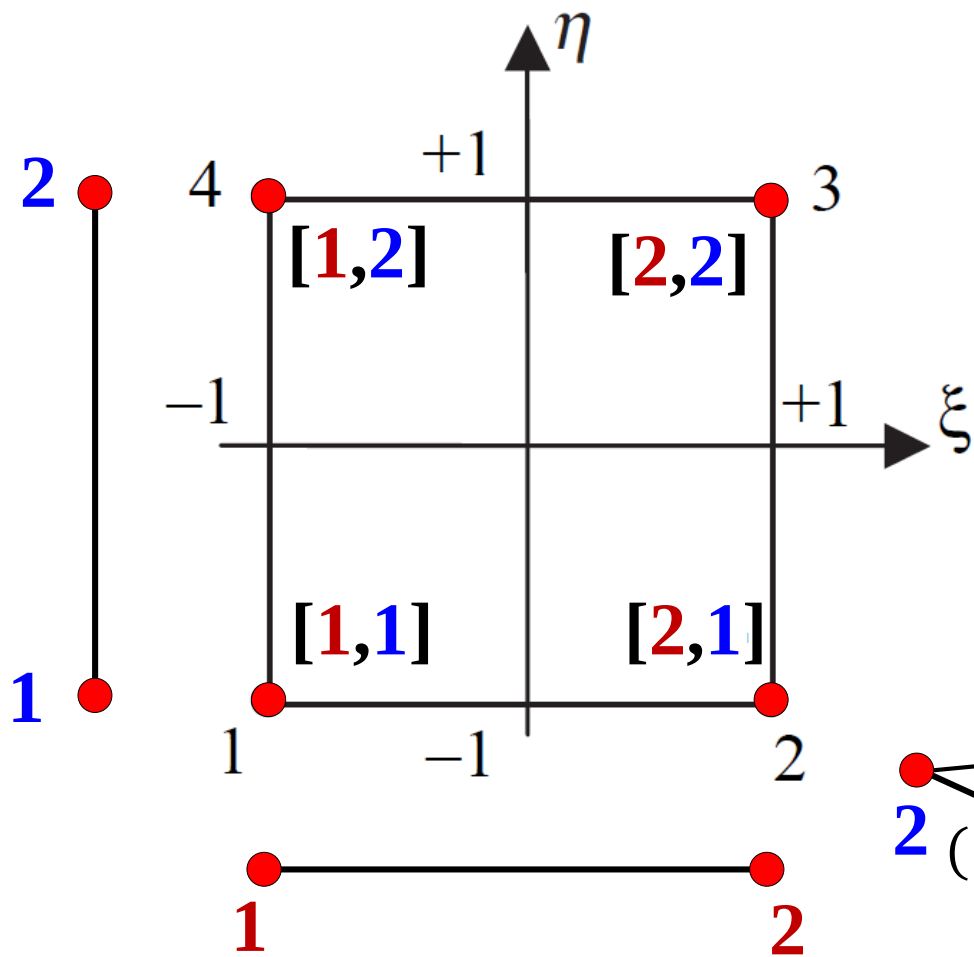
$$\phi_4^e(\xi, \eta) = (1 - \xi)(1 + \eta)/4$$

$$\phi_i^e(\xi, \eta) = \frac{1}{4} (1 + \xi_i \xi)(1 + \eta_i \eta)$$

# Shape function through tensor product

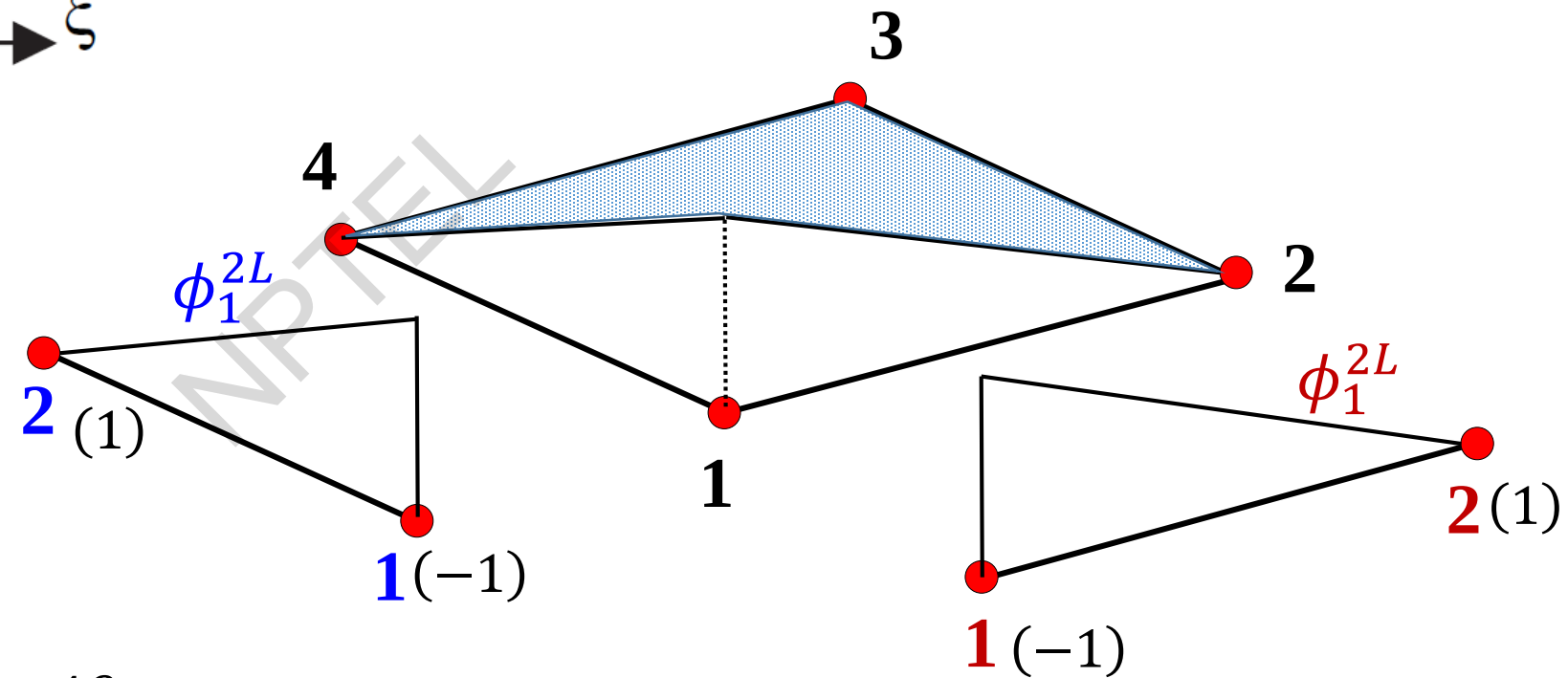


$$\begin{aligned}\phi_k^{4Q}(\xi, \eta) &= \phi_{[I,J]}^{4Q}(\xi, \eta) \\ &= \phi_I^{2L}(\xi) \phi_J^{2L}(\eta)\end{aligned}$$



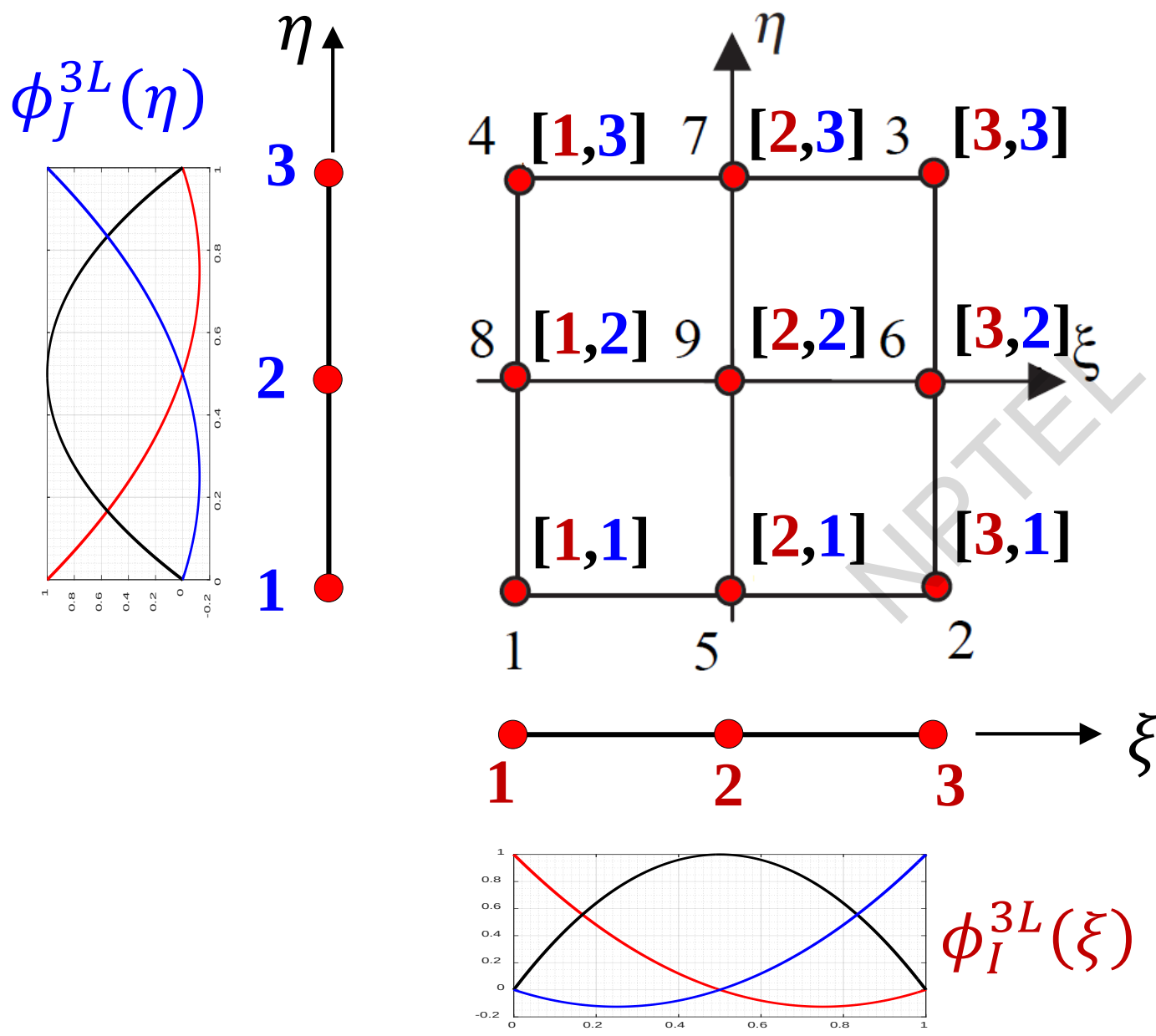
$$\left. \begin{aligned} \phi_1^{2L}(\xi) &= (1 - \xi)/2 \\ \phi_1^{2L}(\eta) &= (1 - \eta)/2 \end{aligned} \right\}$$

$$\phi_1^{4Q}(\xi, \eta) = \phi_1^{2L}(\xi)\phi_1^{2L}(\eta)$$

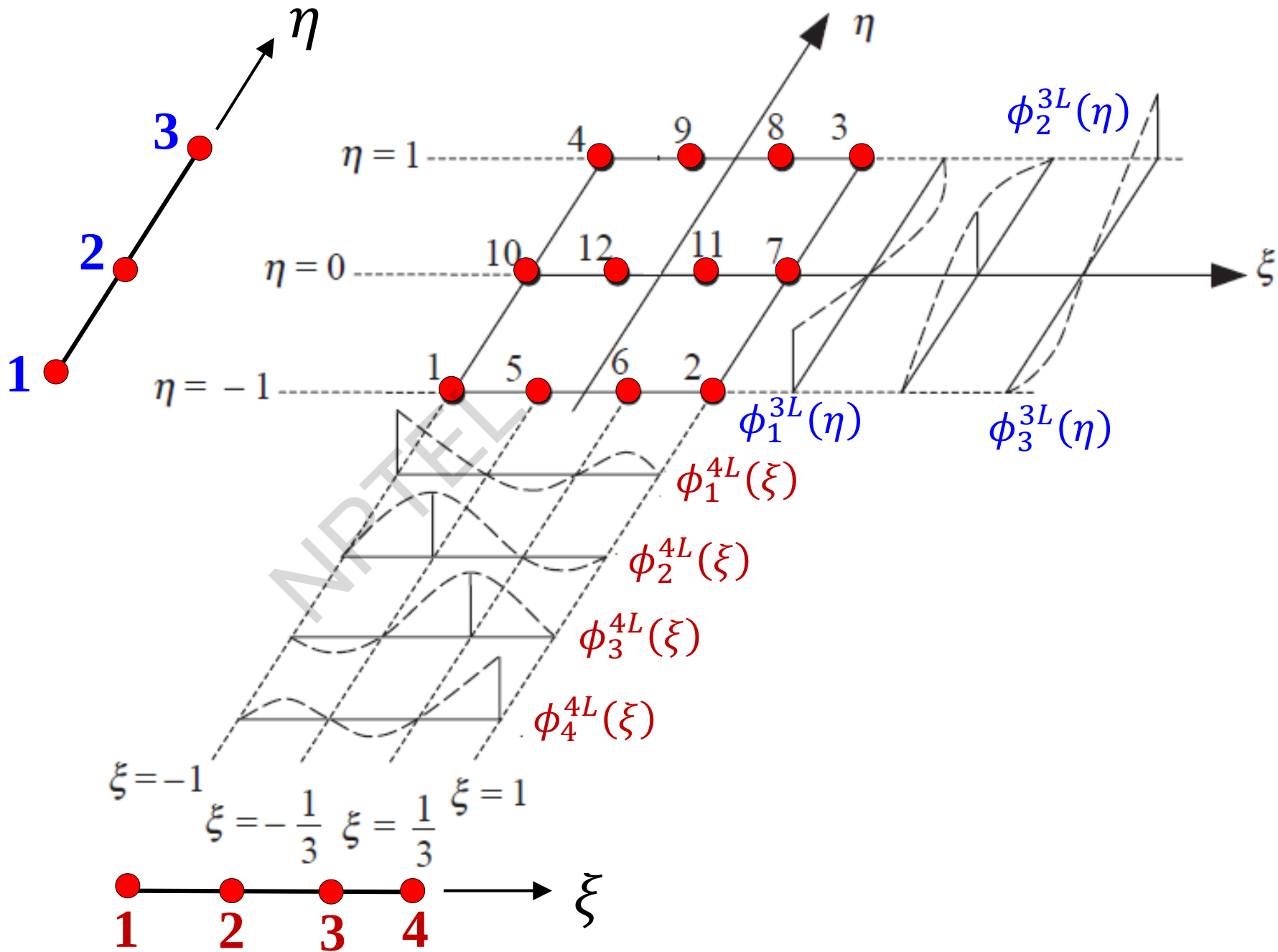


$$\phi_1^{4Q}(\xi, \eta) = (1 - \xi)(1 - \eta)/4$$

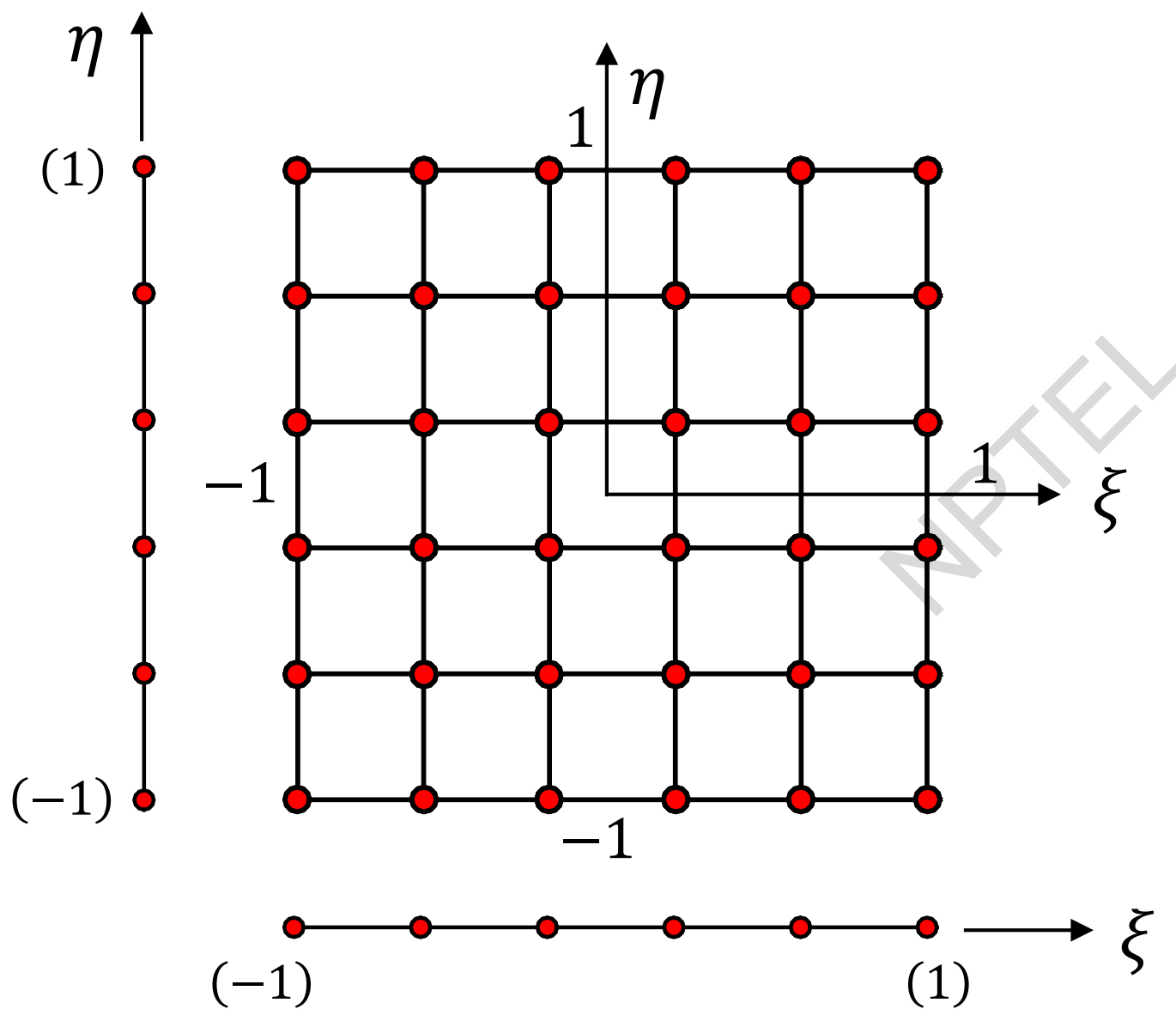
# Shape function through tensor product



$$\begin{aligned} \phi_k^{9Q}(\xi, \eta) &= \phi_{[I,J]}^{9Q}(\xi, \eta) \\ &= \phi_I^{3L}(\xi) \phi_J^{3L}(\eta) \end{aligned}$$



# Shape function through tensor product



*Lagrange Polynomial*

$$L_k(x) = \prod_{\substack{m=1 \\ m \neq k}}^n \frac{x - x_m}{x_k - x_m}$$

# Shape function through tensor product

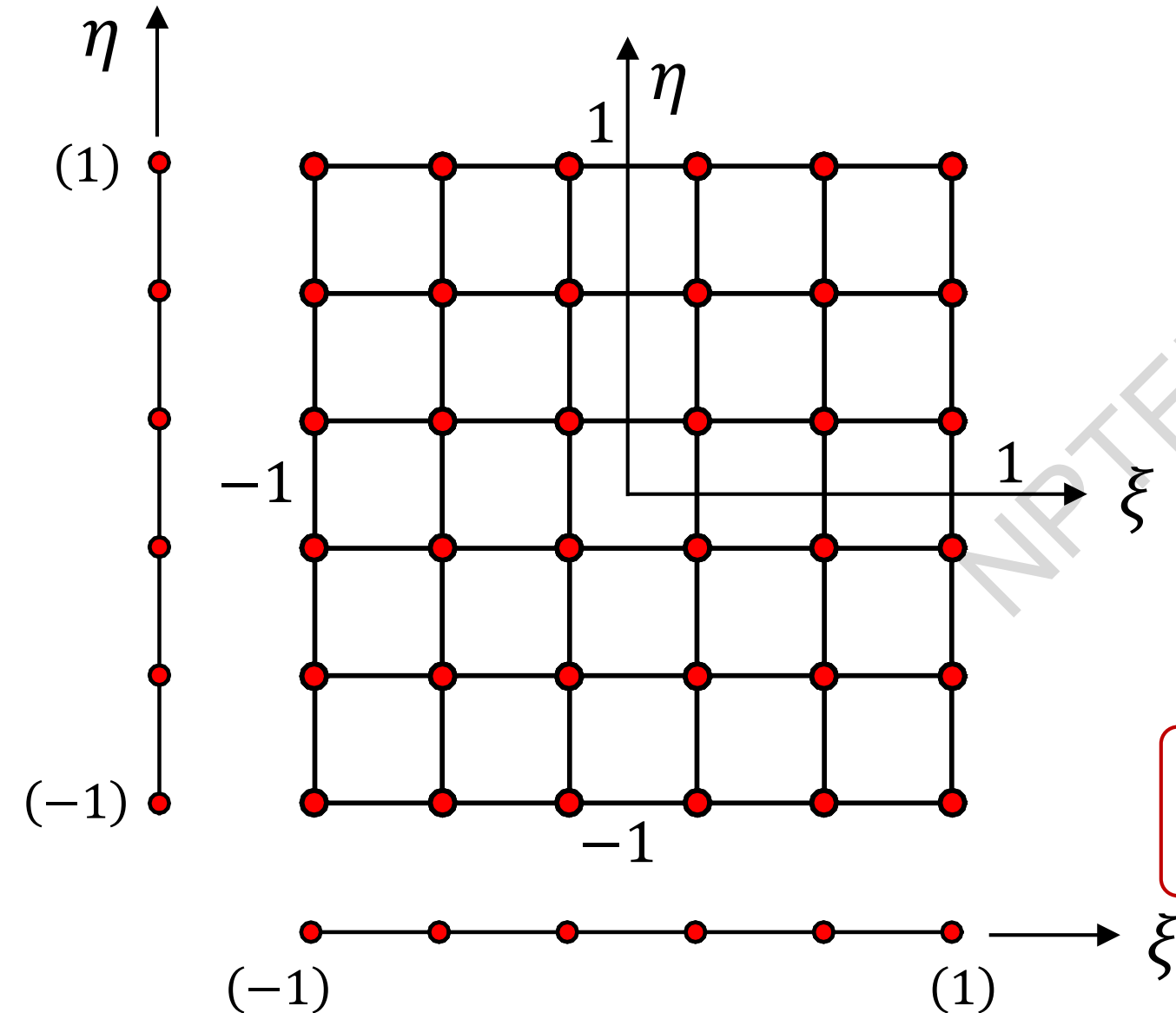
## *Lagrange Polynomial*

$$L_k(x) = \prod_{\substack{m=1 \\ m \neq k}}^n \frac{x - x_m}{x_k - x_m}$$

$$L_k = \frac{(x - x_1)(x - x_2) \dots (x - x_{k-1})(x - x_{k+1}) \dots (x - x_n)}{(x_k - x_1)(x_k - x_2) \dots (x_k - x_{k-1})(x_k - x_{k+1}) \dots (x_k - x_n)}$$

# Shape function through tensor product

## Lagrange Polynomial



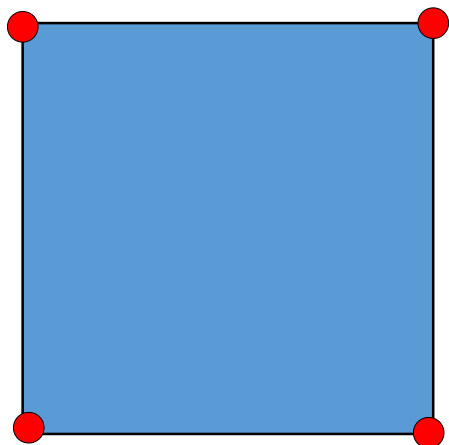
$$L_I(\xi) = \prod_{\substack{m=1 \\ m \neq I}}^{n_\xi} \frac{\xi - \xi_m}{\xi_I - \xi_m} = \phi_I^{n_\xi L}(\xi)$$

$$L_J(\eta) = \prod_{\substack{m=1 \\ m \neq J}}^{n_\eta} \frac{\eta - \eta_m}{\eta_J - \eta_m} = \phi_J^{n_\eta L}(\eta)$$

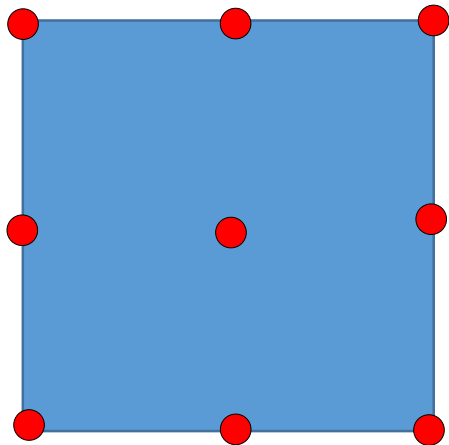
$$\phi_{[I,J]}^{(n_\xi \times n_\eta)Q}(\xi, \eta) = \phi_I^{n_\xi L}(\xi) \phi_J^{n_\eta L}(\eta)$$

# Serendipity Element

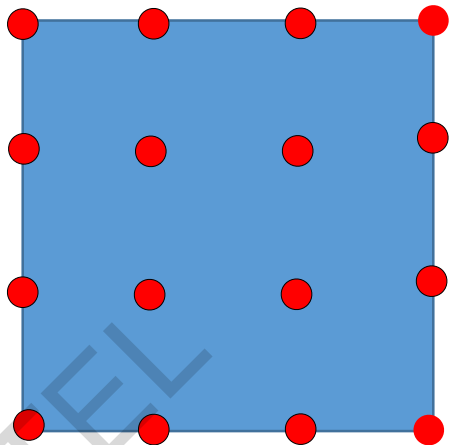
*Recall: Lagrange Family*



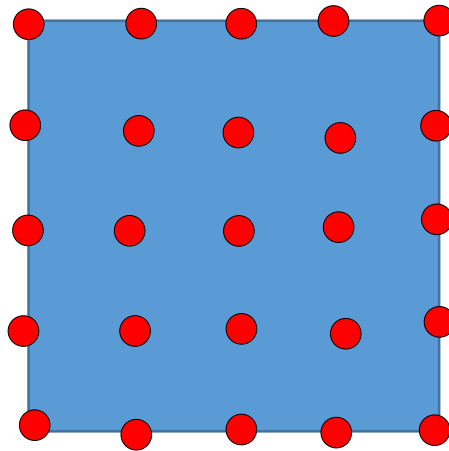
Linear



Quadratic



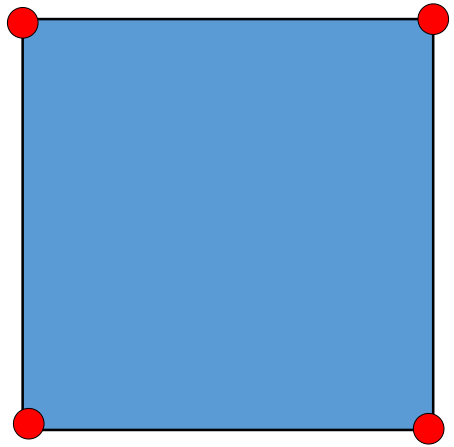
Cubic



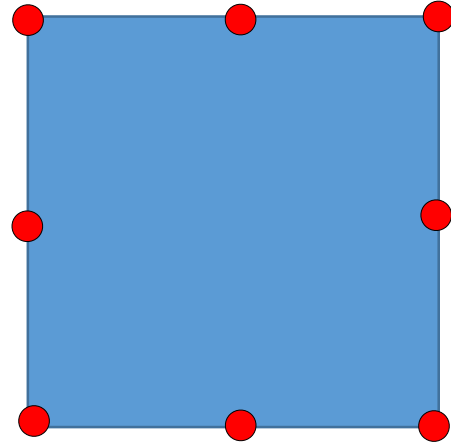
Quartic

NPTEL

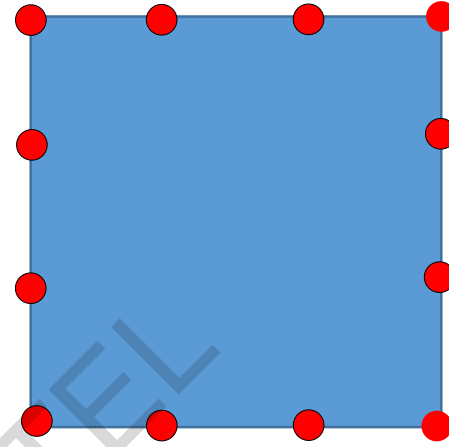
# Serendipity Element



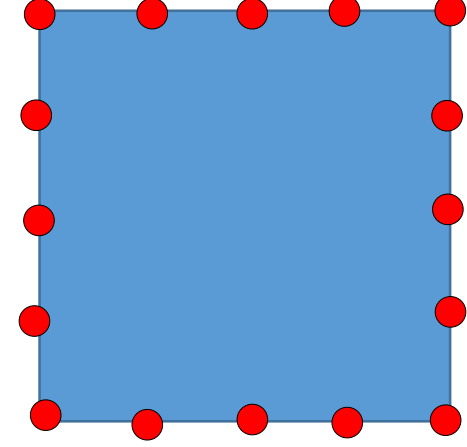
Linear



Quadratic



Cubic



Quartic

Ergatoudis, I.R.O.N.S., Irons, B.M. and Zienkiewicz, O.C., 1968. Curved, isoparametric, “quadrilateral” elements for finite element analysis. *International journal of solids and structures*, 4(1), pp.31-42.

# Conforming Element

When the choice of nodal points, element geometry, and local interpolating functions lead to a global interpolant that is continuous across inter-element boundaries in the connected finite element model, then the individual finite elements are said to be conforming.

The Lagrange families of finite elements contain  $C^0$ -conforming elements

To develop smoother conforming elements, e.g.,  $C^k$  elements,  $k > 0$ , it is also necessary to also prescribe values of derivatives of the interpolant at element nodes.

Hermite Families



**THANK  
YOU !**



## **NPTel ONLINE CERTIFICATION COURSES**

# **Finite Element Method**

## **Week 7: 2D Formulation**

### **Lecture 35: Numerical Integration (Gaussian Quadrature)**

**Biswanath Banerjee**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

**Amit Shaw**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

# Recall: Weak Form and Integration

$$B(w, u) = l(w)$$

## Boundary value problem

$$-\frac{d}{dx} \left( a \frac{du}{dx} \right) + cu - q = 0 \quad \text{for } x \in (0, L)$$

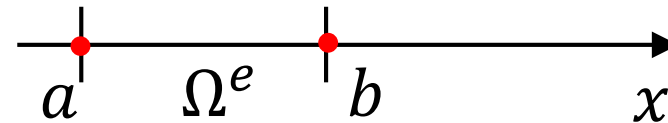
$$u = 0 \quad \text{at } x = 0$$

$$\left( a \frac{du}{dx} \right) = Q_0 \quad \text{at } x = L$$

## Weak form

$$\int_{\Omega^e} \left( a \frac{dw}{dx} \frac{du}{dx} + cwu \right) dx$$

$$= \int_{\Omega^e} wq dx + w(x_A)Q_A + w(x_B)Q_B$$



$$\Omega^e = [a, b]$$

$$h = b - a$$

# Problem

$$I = \int_a^b f(x) dx = ?$$

NPTEL

# Physical domain and Parametric domain

$$I = \int_a^b f(x) dx$$

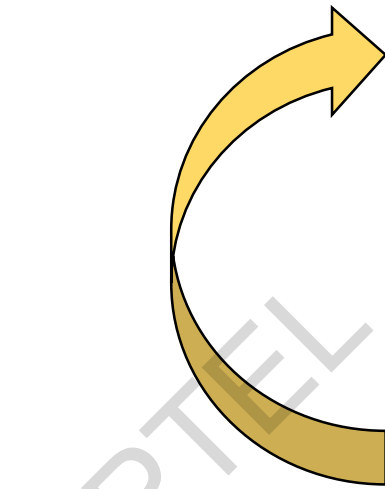
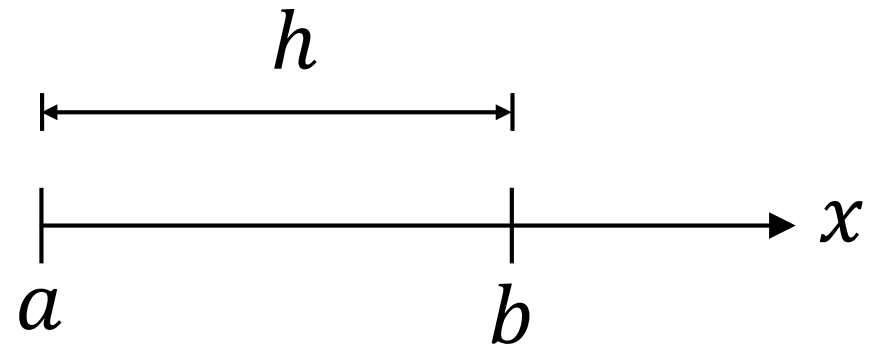
$$x = \frac{1}{2}(a + b) + \frac{1}{2}\xi(b - a)$$

$$\Rightarrow dx = \frac{1}{2}(b - a)d\xi \Rightarrow dx = Jd\xi$$

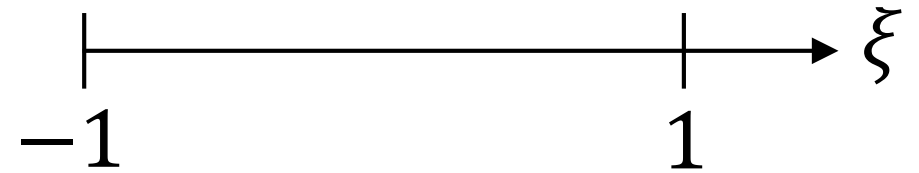
$$J = (b - a)/2$$

Jacobian

$$I = J \int_{-1}^1 f(\xi) d\xi = J \hat{I} \quad \text{where} \quad \hat{I} = \int_{-1}^1 f(\xi) d\xi$$



Physical domain



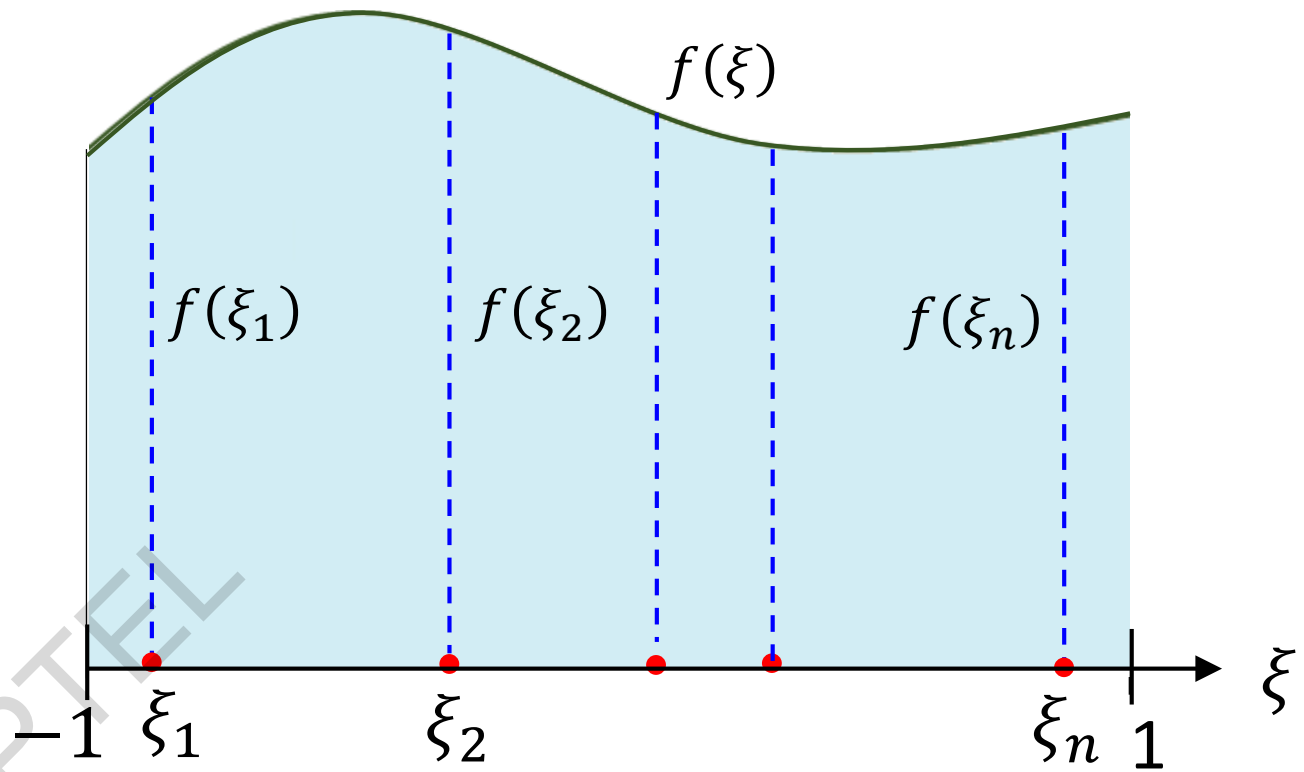
Parametric domain

# Gaussian Quadrature: Concept

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi$$

$$\hat{I} = W_1 f(\xi_1) + W_2 f(\xi_2) + \dots$$

$$\Rightarrow \hat{I} = \underbrace{\{W_1 \quad W_2 \quad \dots \quad W_n\}}_{\mathbf{W}^T} \underbrace{\begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix}}_{\mathbf{f}} \Rightarrow \hat{I} = \mathbf{W}^T \mathbf{f}$$



Integration points

$$\{\xi_1 \quad \xi_1 \quad \dots \quad \xi_n\}$$

Weights

$$\{W_1 \quad W_1 \quad \dots \quad W_n\}$$

# Gaussian Quadrature: *Concept*

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \cdots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

## *Philosophy*

Determine weights  $\{W_1 \quad W_1 \quad \cdots \quad W_n\}$  and integration points  $\{\xi_1 \quad \xi_1 \quad \cdots \quad \xi_n\}$  so that the polynomial of a given order (say  $p$ ) is integrated **exactly**

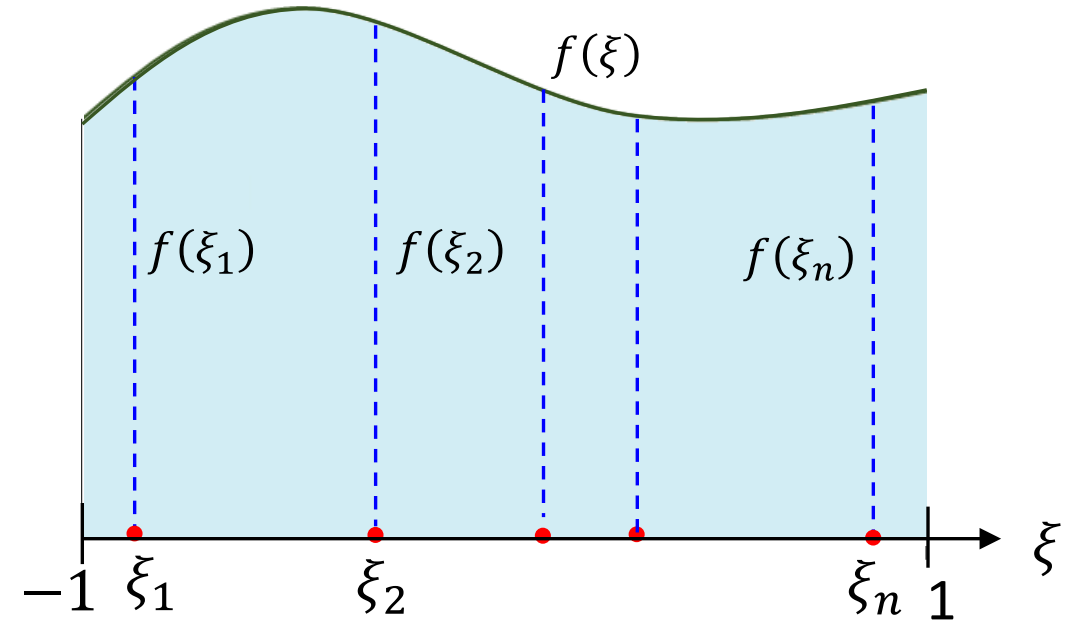
If  $f(\xi) = \alpha_1 + \alpha_2 \xi + \alpha_3 \xi^2 + \cdots + \alpha_{p+1} \xi^p$ ,  $\int_{-1}^1 f(\xi) d\xi$  is exact

# Gaussian Quadrature: Integration points and weights

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \dots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

$$f(\xi) = \alpha_1 + \alpha_2 \xi + \alpha_3 \xi^2 + \dots + \alpha_{p+1} \xi^p$$

$$\Rightarrow f(\xi) = \underbrace{\{1 \quad \xi \quad \xi^2 \quad \dots \quad \xi^p\}}_{\mathbf{p}} \underbrace{\begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{p+1} \end{Bmatrix}}_{\boldsymbol{\alpha}}$$
$$\Rightarrow f(\xi) = \mathbf{p}(\xi) \boldsymbol{\alpha}$$



# Gaussian Quadrature: Integration points and weights

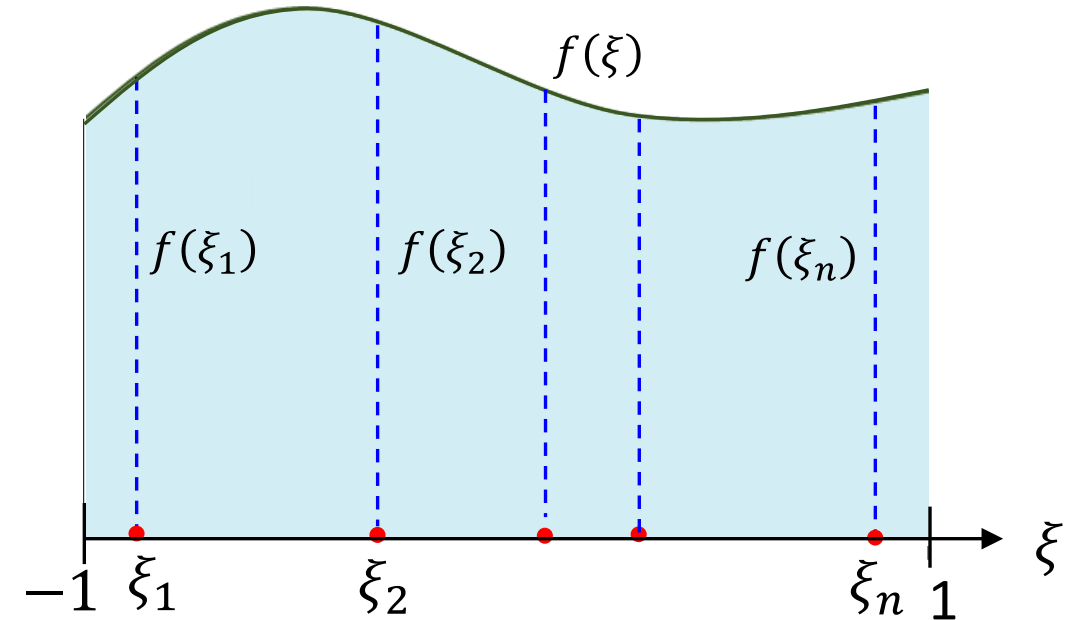
$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \cdots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

$$f(\xi_1) = \alpha_1 + \alpha_2 \xi_1 + \alpha_3 \xi_1^2 + \cdots + \alpha_{p+1} \xi_1^p$$

$$f(\xi_2) = \alpha_1 + \alpha_2 \xi_2 + \alpha_3 \xi_2^2 + \cdots + \alpha_{p+1} \xi_2^p$$

$$\vdots$$

$$f(\xi_n) = \alpha_1 + \alpha_2 \xi_n + \alpha_3 \xi_n^2 + \cdots + \alpha_{p+1} \xi_n^p$$



# Gaussian Quadrature: Integration points and weights

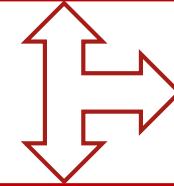
$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \cdots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

$$f(\xi_1) = \alpha_1 + \alpha_2 \xi_1 + \alpha_3 \xi_1^2 + \cdots + \alpha_{p+1} \xi_1^p$$

$$f(\xi_2) = \alpha_1 + \alpha_2 \xi_2 + \alpha_3 \xi_2^2 + \cdots + \alpha_{p+1} \xi_2^p$$

$$\vdots$$

$$f(\xi_n) = \alpha_1 + \alpha_2 \xi_n + \alpha_3 \xi_n^2 + \cdots + \alpha_{p+1} \xi_n^p$$



$$\hat{I} = \mathbf{W}^T \mathbf{M} \boldsymbol{\alpha}$$

$$\underbrace{\begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix}}_{\mathbf{f}} = \underbrace{\begin{bmatrix} 1 & \xi_1 & \xi_1^2 & \cdots & \xi_1^p \\ 1 & \xi_2 & \xi_2^2 & \cdots & \xi_2^p \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_n & \xi_n^2 & \cdots & \xi_n^p \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{p+1} \end{Bmatrix}}_{\boldsymbol{\alpha}}$$

# Gaussian Quadrature: Integration points and weights

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \dots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

$$\hat{I} = \mathbf{W}^T \mathbf{M} \boldsymbol{\alpha}$$

*Integration computed  
numerically via  
Gaussian quadrature*

$$\underbrace{\begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix}}_{\mathbf{f}} = \underbrace{\begin{bmatrix} 1 & \xi_1 & \xi_1^2 & \dots & \xi_1^p \\ 1 & \xi_2 & \xi_2^2 & \dots & \xi_2^p \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_n & \xi_n^2 & \dots & \xi_n^p \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{p+1} \end{Bmatrix}}_{\boldsymbol{\alpha}}$$

# Gaussian Quadrature: Integration points and weights

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi$$

$$f(\xi) = \alpha_1 + \alpha_2 \xi + \alpha_3 \xi^2 + \cdots + \alpha_{p+1} \xi^p$$

$$\hat{I}_{Exact} = \int_{-1}^1 f(\xi) d\xi = \int_{-1}^1 (\alpha_1 + \alpha_2 \xi + \alpha_3 \xi^2 + \cdots + \alpha_{p+1} \xi^p) d\xi$$

$$\Rightarrow \hat{I}_{Exact} = \left[ \alpha_1 \xi + \alpha_2 \frac{\xi^2}{2} + \alpha_3 \frac{\xi^3}{3} \cdots + \alpha_{p+1} \frac{\xi^{p+1}}{p+1} \right]_{-1}^1$$

$$\Rightarrow \hat{I}_{Exact} = \underbrace{\left\{ 2 \quad 0 \quad \frac{2}{3} \cdots \frac{1 - (-1)^{p+1}}{p+1} \right\}}_{\hat{P}} \underbrace{\left\{ \begin{matrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{p+1} \end{matrix} \right\}}_{\alpha} \Rightarrow \boxed{\hat{I}_{Exact} = \hat{P} \alpha}$$

*Exact Integration*

# Gaussian Quadrature: Integration points and weights

$$\hat{I} = \int_{-1}^1 f(\xi) d\xi = \{W_1 \quad W_2 \quad \dots \quad W_n\} \begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix} = \mathbf{W}^T \mathbf{f}$$

$$\hat{I} = \mathbf{W}^T \mathbf{M} \boldsymbol{\alpha}$$

*Integration computed  
numerically via  
Gaussian quadrature*

$$\underbrace{\begin{Bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{Bmatrix}}_{\mathbf{f}} = \underbrace{\begin{bmatrix} 1 & \xi_1 & \xi_1^2 & \dots & \xi_1^p \\ 1 & \xi_2 & \xi_2^2 & \dots & \xi_2^p \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_n & \xi_n^2 & \dots & \xi_n^p \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{p+1} \end{Bmatrix}}_{\boldsymbol{\alpha}}$$

# Gaussian Quadrature: Integration points and weights

$$\hat{I} = \hat{I}_{Exact}$$

$$\hat{I} = \mathbf{W}^T \mathbf{M} \boldsymbol{\alpha}$$

$$\hat{I}_{Exact} = \hat{\mathbf{P}} \boldsymbol{\alpha}$$

$$\Rightarrow \mathbf{W}^T \mathbf{M} \boldsymbol{\alpha} = \hat{\mathbf{P}} \boldsymbol{\alpha} \Rightarrow \mathbf{M}^T \mathbf{W} = \hat{\mathbf{P}}^T$$

$$\Rightarrow \begin{bmatrix} 1 & 1 & \dots & 1 \\ \xi_1 & \xi_2 & \dots & \xi_n \\ \xi_1^2 & \xi_2^2 & \dots & \xi_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \xi_1^p & \xi_2^p & \dots & \xi_n^p \end{bmatrix} \begin{Bmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{Bmatrix} = \begin{Bmatrix} 2 \\ 0 \\ 2/3 \\ \vdots \\ \frac{1 - (-1)^{p+1}}{p+1} \end{Bmatrix}$$

$p + 1$  nonlinear  
algebraic equations

Number of equations =  $p + 1$

Number of unknown =  $2n$

$$\{\xi_i\}_{i=1}^n \quad \{W_i\}_{i=1}^n$$

*$n$ -point Gaussian quadrature can integrate a  $(2n-1)$ -order polynomial exactly*

# Legendre Polynomials

*Rodrigues' formula*

$$P_n(\xi) = \frac{1}{2^n n!} \frac{d^n}{d\xi^n} (\xi^2 - 1)^n$$

$$n = 0 \quad P_0(\xi) = 1$$

$$n = 1 \quad P_1(\xi) = \xi$$

$$n = 2 \quad P_2(\xi) = \frac{1}{2}(3\xi^2 - 1)$$

$$n = 3 \quad P_3(\xi) = \frac{1}{2}(5\xi^3 - 3\xi)$$

$$n = 4 \quad P_4(\xi) = \frac{1}{8}(35\xi^4 - 30\xi^2 + 3)$$

$$n = 5 \quad P_5(\xi) = \frac{1}{8}(63\xi^5 - 70\xi^3 + 15\xi)$$

$$n = 6 \quad P_6(\xi) = \frac{1}{16}(231\xi^6 - 315\xi^4 + 105\xi^2 - 5)$$

$$n = 7 \quad P_7(\xi) = \frac{1}{16}(429\xi^7 - 693\xi^5 + 315\xi^3 - 35\xi)$$

# Gauss-Legendre Quadrature

| $n$ | $\xi_i$                                                        | $W_i$                                              |
|-----|----------------------------------------------------------------|----------------------------------------------------|
| 1   | 0.0                                                            | 2.0                                                |
| 2   | $\pm 1/\sqrt{3} = \pm 0.5773502692$                            | 1.0                                                |
| 3   | $\pm 0.7745966692$<br>0.0                                      | 0.555 555 5556<br>0.888 888 8889                   |
| 4   | $\pm 0.8611363116$<br>$\pm 0.3399810436$                       | 0.347 854 8451<br>0.652 145 1549                   |
| 5   | $\pm 0.9061798459$<br>$\pm 0.5384693101$<br>0.0                | 0.236 926 8851<br>0.478 628 6705<br>0.568 888 8889 |
| 6   | $\pm 0.9324695142$<br>$\pm 0.6612093865$<br>$\pm 0.2386191861$ | 0.171 324 4924<br>0.360 761 5730<br>0.467 913 9346 |



**THANK  
YOU !**



**NPTEL ONLINE CERTIFICATION COURSES**

# **Finite Element Method**

**Week 7: 2D Formulation**

## **Lecture 36: Gaussian Quadrature in two dimension**

**Biswanath Banerjee**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

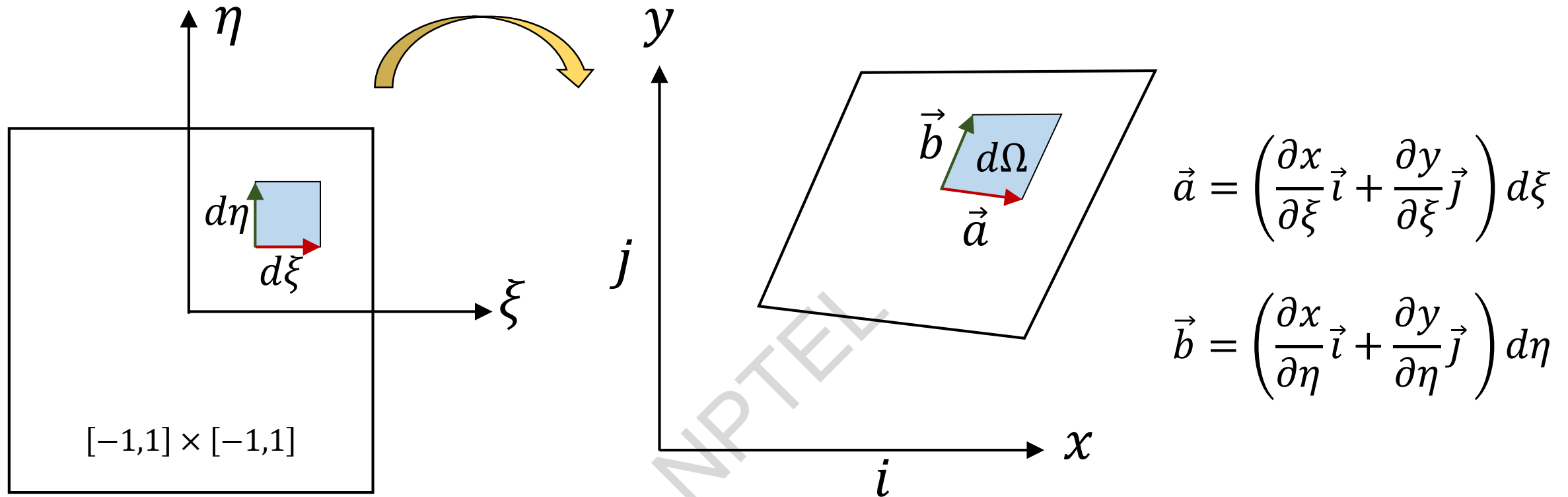
**Amit Shaw**  
Associate Professor  
Department of Civil Engineering  
IIT Kharagpur

## Problem

$$I = \int_{\Omega^e} f(x, y) d\Omega = ?$$

NPTEL

# Integration Over Quadrilateral Domain



$$d\Omega = \vec{k} \cdot (\vec{a} \times \vec{b}) = \vec{k} \cdot \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ \frac{\partial x}{\partial \xi} d\xi & \frac{\partial y}{\partial \xi} d\xi & 0 \\ \frac{\partial x}{\partial \eta} d\eta & \frac{\partial y}{\partial \eta} d\eta & 0 \end{bmatrix} = \det \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} d\xi d\eta = |\mathbf{J}| d\xi d\eta$$

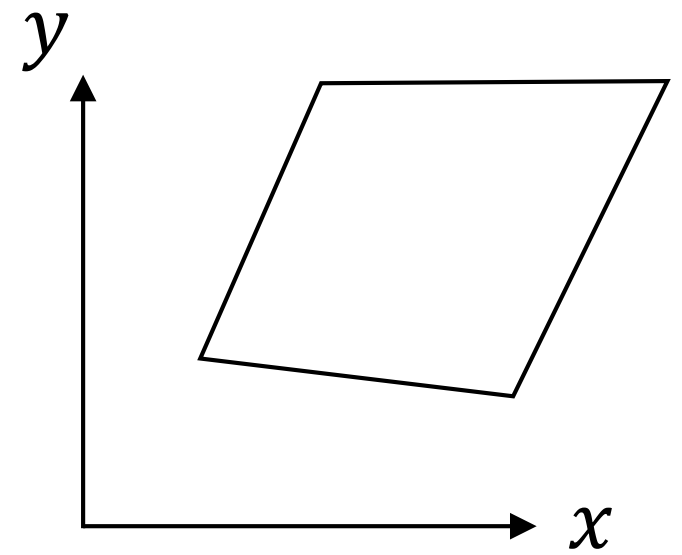
**Jacobian**

# Integration Over Quadrilateral Domain

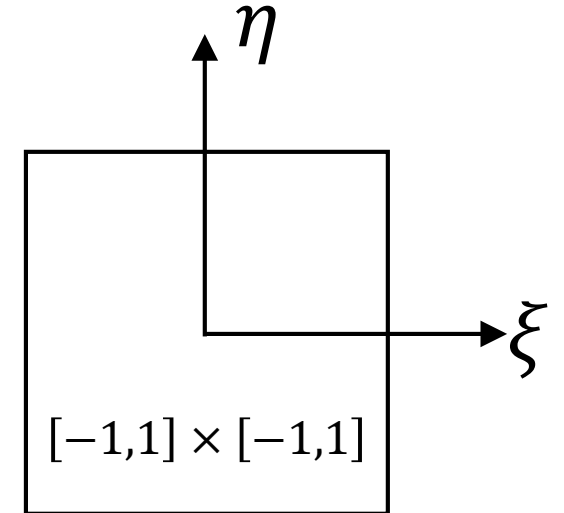
$$I = \int_{\Omega^e} f(x, y) d\Omega$$

$$\Rightarrow I = \int_{\eta=-1}^1 \int_{\xi=-1}^1 |\mathbf{J}(\xi, \eta)| f(\xi, \eta) d\xi d\eta$$

$$\Rightarrow I = \sum_{i=1}^n \sum_{j=1}^n w_i w_j |\mathbf{J}(\xi_i, \eta_j)| f(\xi_i, \eta_j)$$



Physical domain



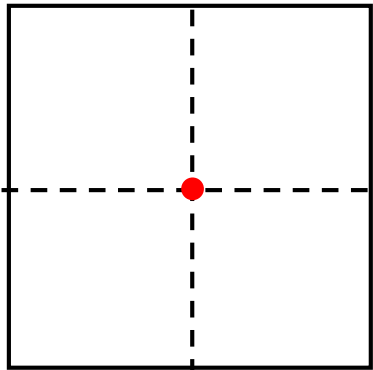
Parametric domain  
(natural coordinate)

# Integration Over Quadrilateral Domain

$$I = \sum_{i=1}^n \sum_{j=1}^n W_i W_j |\mathbf{J}(\xi_i, \eta_j)| f(\xi_i, \eta_j)$$

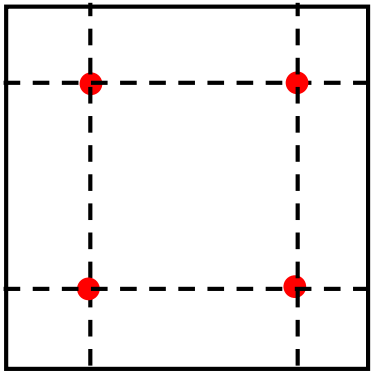
| $n$ | $\xi_i$                                  | $W_i$                            |
|-----|------------------------------------------|----------------------------------|
| 1   | 0.0                                      | 2.0                              |
| 2   | $\pm 1/\sqrt{3} = \pm 0.5773502692$      | 1.0                              |
| 3   | $\pm 0.7745966692$<br>0.0                | 0.555 555 5556<br>0.888 888 8889 |
| 4   | $\pm 0.8611363116$<br>$\pm 0.3399810436$ | 0.347 854 8451<br>0.652 145 1549 |

$n = 1$



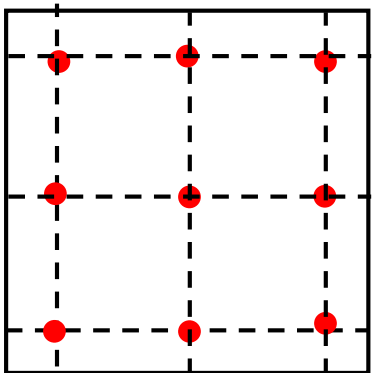
$[-1,1] \times [-1,1]$

$n = 2$



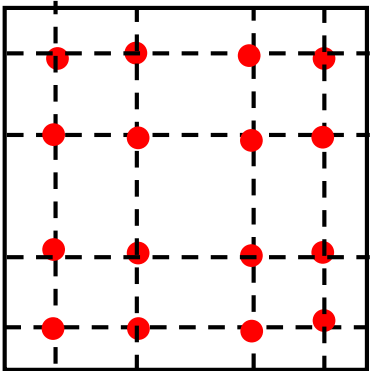
$[-1,1] \times [-1,1]$

$n = 3$

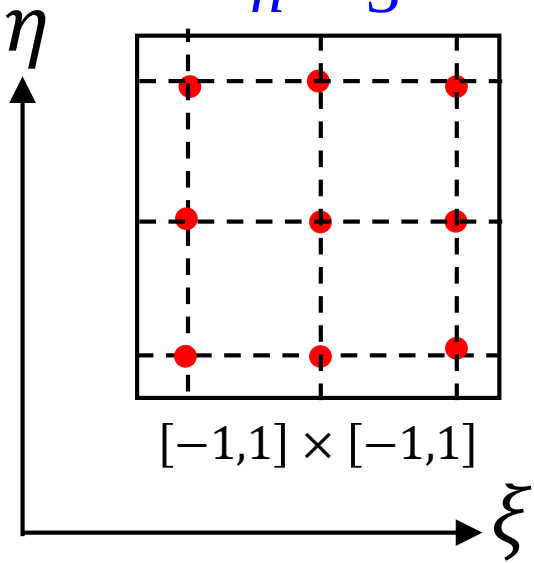


$[-1,1] \times [-1,1]$

$n = 4$



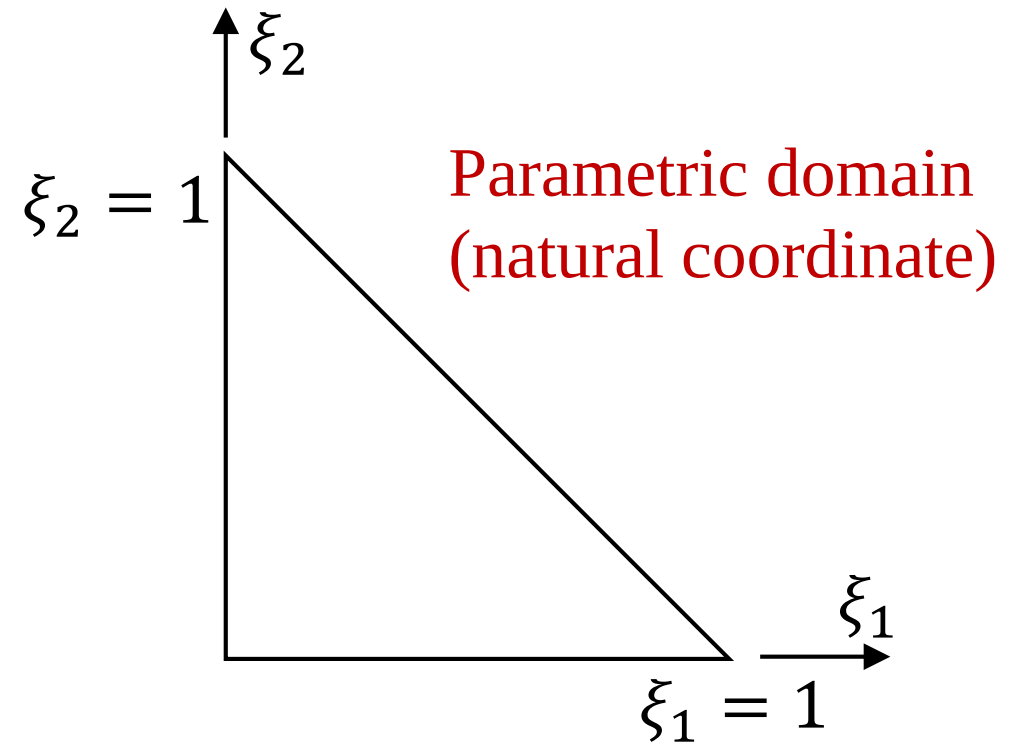
$[-1,1] \times [-1,1]$



# Integration Over Triangular Domain

$$I = \sum_{i=1}^n W_i |\mathbf{J}(\xi_{1i}, \xi_{2i})| f(\xi_{1i}, \xi_{2i})$$

$$\mathbf{J}(\xi_{1i}, \xi_{2i}) = \begin{bmatrix} \frac{\partial x}{\partial \xi_1} & \frac{\partial y}{\partial \xi_1} \\ \frac{\partial x}{\partial \xi_2} & \frac{\partial y}{\partial \xi_2} \end{bmatrix}$$



For straight – sided 3 – node triangle

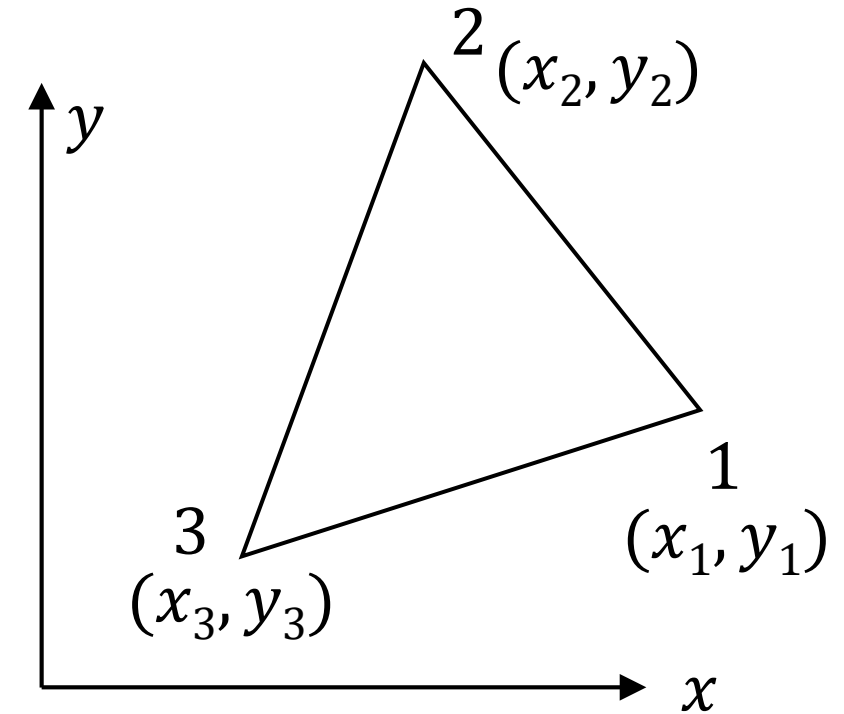
# Integration Over Triangular Domain

$$I = \sum_{i=1}^n W_i |\mathbf{J}(\xi_{1i}, \xi_{2i})| f(\xi_{1i}, \xi_{2i})$$

$$\mathbf{J}(\xi_{1i}, \xi_{2i}) = \begin{bmatrix} \frac{\partial x}{\partial \xi_1} & \frac{\partial y}{\partial \xi_1} \\ \frac{\partial x}{\partial \xi_2} & \frac{\partial y}{\partial \xi_2} \end{bmatrix}$$

For straight – sided 3 – node triangle

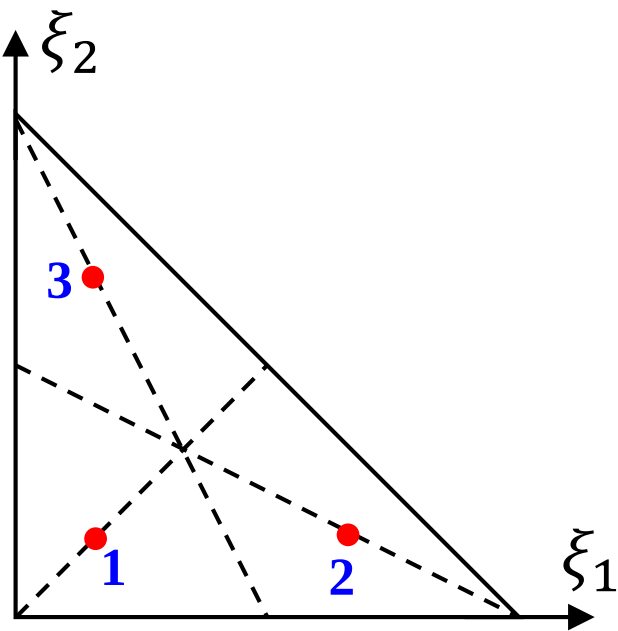
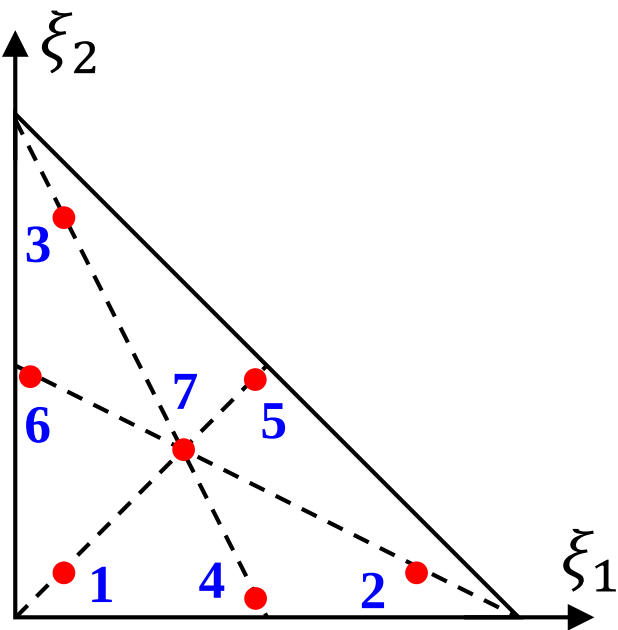
$$\mathbf{J} = \begin{bmatrix} x_1 - x_3 & y_1 - y_3 \\ x_2 - x_3 & y_2 - y_3 \end{bmatrix}$$



# Integration Over Triangular Domain

$$I = \sum_{i=1}^n W_i |\mathbf{J}(\xi_{1_i}, \xi_{2_i})| f(\xi_{1_i}, \xi_{2_i})$$

| Integration order | $\xi_{1_i}$     | $\xi_{2_i}$     | $W_i$           |
|-------------------|-----------------|-----------------|-----------------|
| Three-point       | 0.1 666 666 666 | 0.1 666 666 666 | 0.1 666 666 666 |
|                   | 0.6 666 666 666 | 0.1 666 666 666 | 0.1 666 666 666 |
|                   | 0.1 666 666 666 | 0.6 666 666 666 | 0.1 666 666 666 |
| Seven-point       | 0.1 012 865 073 | 0.1 012 865 073 | 0.0 629 695 903 |
|                   | 0.7 974 269 853 | 0.1 012 865 073 | 0.0 629 695 903 |
|                   | 0.1 012 865 073 | 0.7 974 269 853 | 0.0 629 695 903 |
|                   | 0.4 701 420 641 | 0.0 597 158 717 | 0.0 661 970 764 |
|                   | 0.4 701 420 641 | 0.4 701 420 641 | 0.0 661 970 764 |
|                   | 0.0 597 158 717 | 0.4 701 420 641 | 0.0 661 970 764 |
|                   | 0.3 333 333 333 | 0.3 333 333 333 | 0.1125          |



# Integration Over Triangular Domain

## *Cowper's formula*

Monomial of any order can be integrated on a straight sided triangle in the closed form as

$$\int_{\Omega_e} \xi_1^i \xi_2^j \xi_3^k d\Omega_e = \frac{i! j! k!}{(i + j + k + 2)!} 2A^e$$



**THANK  
YOU !**