

*Claudio Borri*  
*Luca Salvatori*

# MECCANICA COMPUTAZIONALE

Capitolo 4

## Libreria di elementi

Rev. 19 maggio 2007



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 1/57



## Argomenti trattati nel capitolo 4

- Biella rivisitata (formulazione isoparametrica)
- Elementi continui isoparametrici 2D
  - ISO 3
  - ISO 4
  - cenni a elementi di ordine superiore
- Elemento trave 2D
- Panoramica di altri elementi

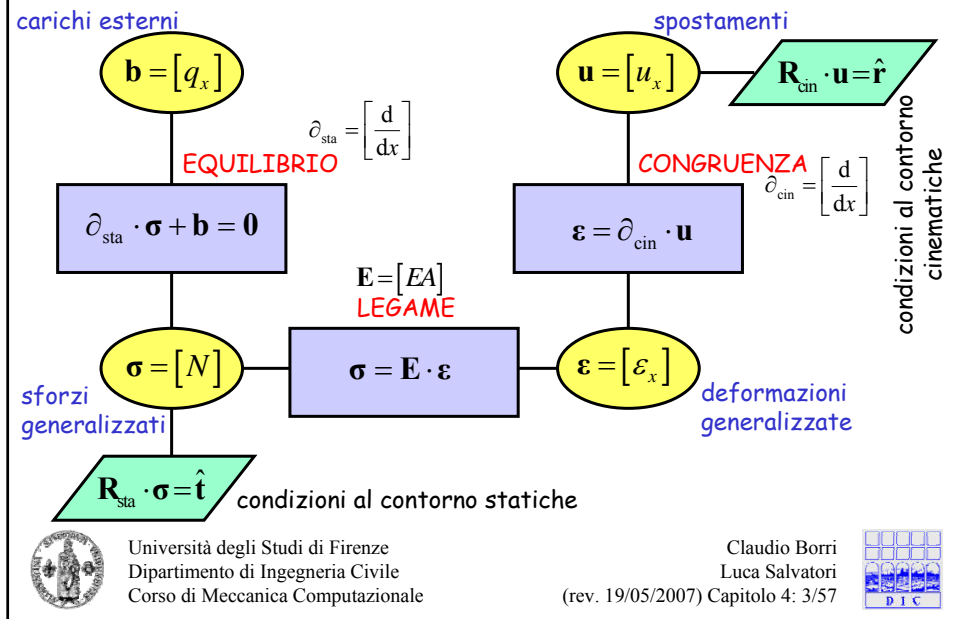


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 2/57



## Biella: diagramma di Tonti



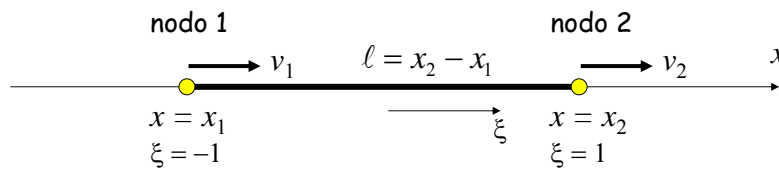
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 3/57



## Biella: coordinate intrinseche

Definiamo un sistema di coordinate locali adimensionali (**coordinate intrinseche**) nel quale la posizione dei nodi sono -1 e +1.



Le trasformazioni di coordinate sono:

$$\xi = 2 \frac{x - x_1}{x_2 - x_1} - 1 \quad \Longrightarrow \quad x = \frac{1 - \xi}{2} x_1 + \frac{1 + \xi}{2} x_2$$

Nota 1: L'utilità delle coordinate intrinseche sarà meglio chiarita analizzando elementi 2D e 3D.

Nota 2: Osservare l'espressione riquadrata, vedremo poi la sua importanza!



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 4/57



## Biella: scelta della funzione interpolante

Ipotizziamo un'interpolazione lineare degli spostamenti:

$$u(\xi) = a_1 + a_2 \xi$$

nella notazione matriciale introdotta nel cap. 3:

$$\mathbf{u}(\xi) = [u(\xi)] = [1 \quad \xi] \cdot \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \Phi(\xi) \cdot \mathbf{a}$$

I parametri  $\mathbf{a}$  sono privi di significato fisico e non sono continui nei nodi!

Si cerca allora una relazione con gli spostamenti nodali, imponendo la congruenza nei nodi stessi:

$$u(\xi) = a_1 + a_2 \xi \implies \begin{cases} u(-1) = a_1 - a_2 = v_1 \\ u(+1) = a_1 + a_2 = v_2 \end{cases} \implies \Psi \cdot \mathbf{a} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \mathbf{v}$$

invertendo si ricavano i coefficienti incogniti:

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \Psi^{-1} \cdot \mathbf{v}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 5/57



## Biella: funzioni di forma

Interpolazione:  $\mathbf{u}(\xi) = [u(\xi)] = [1 \quad \xi] \cdot \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \Phi(\xi) \cdot \mathbf{a}$

Coefficienti in funzione degli spostamenti nodali:  $\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \Psi^{-1} \cdot \mathbf{v}$

Sostituendo:  $\mathbf{u}(\xi) = \Phi(\xi) \Psi^{-1} \mathbf{v} = \Omega(\xi) \mathbf{v}$

$$[u(\xi)] = [1 \quad \xi] \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \frac{1-\xi}{2} & \frac{1+\xi}{2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

si ricava la matrice delle funzioni di forma:

$$\Omega(\xi) = [\Omega_1(\xi) \quad \Omega_2(\xi)]$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 6/57



## Biella: proprietà delle funzioni di forma

L'interpolazione degli spostamenti si scrive dunque come:

$$[u] = [\Omega_1 \quad \Omega_2] \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$\Omega_1 = \frac{1-\xi}{2}$

$\Omega_2 = \frac{1+\xi}{2}$

- Le funzioni di forma sono in numero uguale al numero dei nodi.
- Esse descrivono come lo spostamento nodale unitario venga interpolato all'interno dell'elemento.
- L' $i$ -ma funzione vale 1 al nodo  $i$ -mo e 0 in tutti gli altri nodi.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 7/57



## Biella: congruenza

Dalla congruenza:

$$\boldsymbol{\varepsilon} = \partial_{\text{cin}} \cdot \mathbf{u}$$

sostituendo l'interpolazione degli spostamenti:

$$\mathbf{u} = \boldsymbol{\Omega} \cdot \mathbf{v}$$

si ottiene:

$$\boldsymbol{\varepsilon} = \partial_{\text{cin}} \cdot (\boldsymbol{\Omega} \cdot \mathbf{v}) = (\partial_{\text{cin}} \cdot \boldsymbol{\Omega}) \cdot \mathbf{v} = \mathbf{B} \cdot \mathbf{v}$$

Per la biella:

dalle funzioni di forma:

$$\boldsymbol{\Omega}(\xi) = [\Omega_1(\xi) \quad \Omega_2(\xi)] = \left[ \frac{1-\xi}{2} \quad \frac{1+\xi}{2} \right]$$

si ricava l'interpolazione delle deformazioni:

$$\mathbf{B} = \partial_{\text{cin}} \cdot \boldsymbol{\Omega} = \left[ \frac{d}{dx} \right] [\Omega_1 \quad \Omega_2] = \left[ \frac{d\Omega_1}{dx} \quad \frac{d\Omega_2}{dx} \right]$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 8/57



## Biella: interpolazione delle deformazioni

Poiché le  $\Omega_i$  sono funzione di  $\xi$  e la derivazione presente nella cinematica è rispetto ad  $x$ , si applica la regola di derivazione per funzioni composte:

$$\left\{ \begin{array}{l} \frac{d\Omega_1}{dx} = \frac{d\Omega_1}{d\xi} \frac{d\xi}{dx} = -\frac{1}{2} \frac{2}{\ell} = -\frac{1}{\ell} \\ \frac{d\Omega_2}{dx} = \frac{d\Omega_2}{d\xi} \frac{d\xi}{dx} = +\frac{1}{2} \frac{2}{\ell} = +\frac{1}{\ell} \end{array} \right. \Rightarrow \mathbf{B} = \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix}$$

N.B.: Per la biella è la matrice è costante, in generale è funzione della posizione!

dove si è tenuto conto che:

$$\xi = 2 \frac{x - x_1}{x_2 - x_1} - 1 \Rightarrow \frac{d\xi}{dx} = \frac{2}{x_2 - x_1} = \frac{2}{\ell}$$

Si ottiene infine l'espressione delle deformazioni:

$$\boldsymbol{\varepsilon} = \mathbf{B} \mathbf{v}$$
$$[\varepsilon_x] = \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 9/57



## Biella: legame e interpolazione degli sforzi

Dal legame:

$$\boldsymbol{\sigma} = \mathbf{E} \cdot \boldsymbol{\varepsilon}$$

sostituendo l'interpolazione delle deformazioni:

$$\boldsymbol{\varepsilon} = \mathbf{B} \cdot \mathbf{v}$$

si ottiene:

$$\boldsymbol{\sigma} = \mathbf{E} \cdot \mathbf{B} \cdot \mathbf{v}$$

cioè:

$$[N] = [EA] \cdot \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -\frac{EA}{\ell} & +\frac{EA}{\ell} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 10/57



## Biella: PLV

Sostituendo le quantità discretizzate nel PLV:

$$\delta \Pi_{EPT}^u = \int_V \delta \boldsymbol{\varepsilon}^T \cdot \boldsymbol{\sigma} dV - \int_V \delta \mathbf{u}^T \cdot \mathbf{b} dV - \int_{S_t} \delta \mathbf{u}^T \cdot \hat{\mathbf{t}} dA = 0$$

si ottiene:

$$\delta \mathbf{v}^T \cdot [\mathbf{k} \cdot \mathbf{v} - \mathbf{p}] = 0$$

e di conseguenza le equazioni di rigidità:

$$\mathbf{k} \cdot \mathbf{v} = \mathbf{p}$$

con:

$$\mathbf{p} = \int_V \boldsymbol{\Omega}(\mathbf{x})^T \cdot \mathbf{b} dV + \int_{S_t} \boldsymbol{\Omega}(\mathbf{x})^T \cdot \hat{\mathbf{t}} dA$$

e:

$$\mathbf{k} = \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 11/57



## Biella: matrice di rigidità

Sostituendo le matrici:

$$\mathbf{B} = \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix} \quad e \quad \mathbf{E} = [EA]$$

nell'espressione della rigidità:

$$\mathbf{k} = \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV$$

l'integrale di volume diviene un'integrale di linea (la biella è 1D in coordinate locali):

$$\mathbf{k} = \int_{\ell} \left( \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix}^T \cdot [EA] \cdot \begin{bmatrix} -\frac{1}{\ell} & +\frac{1}{\ell} \end{bmatrix} \right) dx = \frac{EA}{\ell} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

- L'espressione della **matrice di rigidità in coordinate locali** è la stessa trovata nel cap. 2 utilizzando direttamente le equazioni fondamentali.
- Ad essa deve venire applicata la trasformazione in coordinate globali già vista nel cap. 2.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 12/57



## Biella: è isoparametrica!

Torniamo adesso sull'interpolazione degli spostamenti:

$$u = \Omega_1 v_1 + \Omega_2 v_2$$

e richiamiamo l'espressione delle coordinate locali in funzione delle coordinate intrinseche:

$$x = \frac{1-\xi}{2} x_1 + \frac{1+\xi}{2} x_2 = \Omega_1 x_1 + \Omega_2 x_2$$

**Le espressioni di  $x$  e di  $u$  sono del tutto analoghe!**

In questo caso, quando geometria e spostamenti utilizzano lo stesso numero di parametri (e dunque la stessa descrizione spaziale), si parla di **elementi isoparametrici**.

È possibile generalizzare interpretando le funzioni di forma isoparametriche come interpolazione di una qualsiasi funzione  $f(\xi)$  i cui valori nodali siano  $f_1$  e  $f_2$  (e.g. posizione  $x$ , spostamento  $u$ , temperatura  $\theta$ , ecc.):

$$f(\xi) = \Omega_1(\xi) \cdot f_1 + \Omega_2(\xi) \cdot f_2$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 13/57



## Elementi isoparametrici

- Usano lo **stesso numero di parametri** (le funzioni di forma) per descrivere **geometria** e per interpolare il campo di **spostamenti**. Vantaggi:
  - ☺ unificazione di formulazione: i passaggi sono identici per tutti gli elementi;
  - ☺ soddisfano le condizioni di convergenza;
  - ☺ la costruzione delle funzioni di forma è semplificata;
  - ☺ elementi di ordine elevato ammettono bordi curvilinei.
- Concetto **ampiamente usato** (cavi, continui 2D e 3D, lastre inflesse, ecc.).
- È possibile formulare elementi iper-parametrici e ipo-parametrici:
  - elementi iper-parametrici (o super-) utilizzano geometrie di grado più elevato (ma con lati rettilinei) rispetto all'interpolazione degli spostamenti, venivano usati prima dell'introduzione del concetto isoparametrico.
  - elementi ipo-parametrici (o sub-) utilizzano geometrie di ordine inferiore (introducendo ipotesi *ad hoc* sul campo di spostamenti per poterne ricavare i parametri).

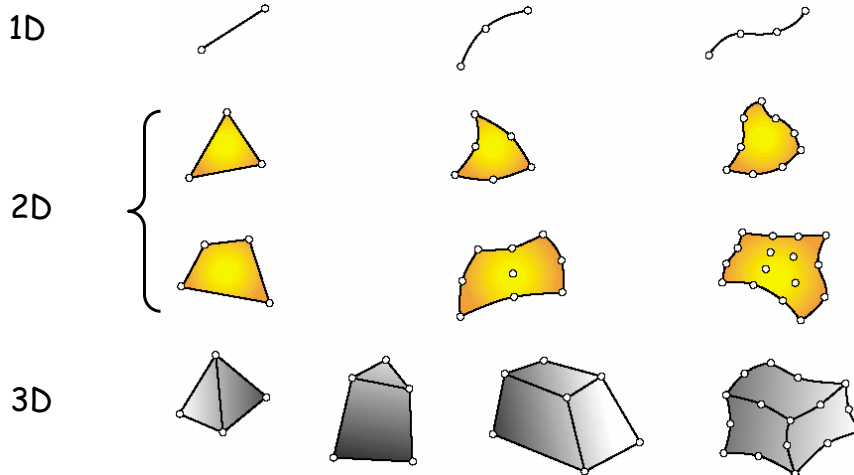


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 14/57



## La geometria degli elementi isoparametrici è definita dalla posizione dei nodi



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 15/57



## Consideriamo adesso elementi isoparametrici per il problema piano di tensione

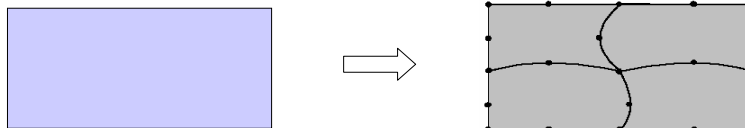
Definiamo i vettori dei carichi di superficie, degli sforzi piani (tensioni moltiplicate per lo spessore  $h$ ), delle deformazioni e degli spostamenti:

$$\mathbf{b} = \begin{bmatrix} q_x \\ q_y \end{bmatrix} \quad \boldsymbol{\sigma} = \begin{bmatrix} n_{xx} \\ n_{yy} \\ n_{xy} \end{bmatrix} \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} u_x \\ u_y \end{bmatrix}$$

L'operatore cinematico e la matrice del legame costitutivo sono:

$$\partial_{\text{cin}} = \begin{bmatrix} \partial_x & 0 \\ 0 & \partial_y \\ \partial_y & \partial_x \end{bmatrix} \quad \mathbf{E} = \frac{Eh}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix}$$

Dividiamo la superficie originale in elementi (nel cap. 5 vedremo come):



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

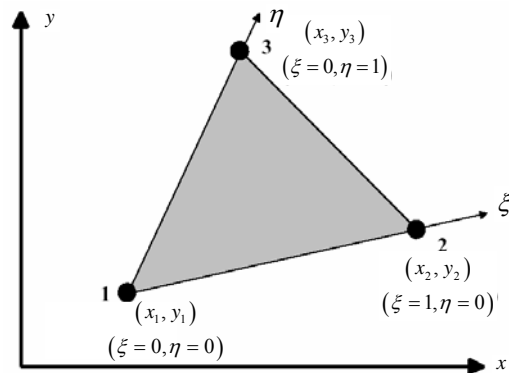
Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 16/57





## ISO 3: elemento isoparametrico piano a 3 nodi

Definizione di un sistema  $(\xi, \eta)$  di coordinate intrinseche:



Attenzione: il sistema  $(x, y)$  è qui il sistema di coordinate globali!



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 17/57



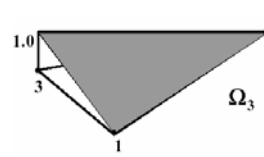
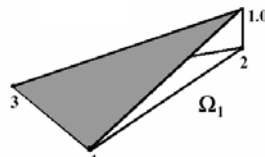
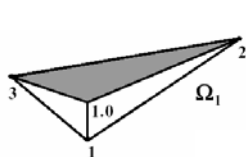
## ISO 3: interpolazione

Ci sono 3 nodi, dunque l'interpolazione richiede 3 costanti.  
Si utilizza un **polinomio di primo grado completo**:

$$f(\xi, \eta) = a_1 + a_2\xi + a_3\eta$$

Nei nodi la grandezza  $f$  deve assumere i propri valori nodali. Procedendo come fatto per la biella si ricavano le **funzioni di forma**:

$$\begin{cases} f(0,0) = a_1 = f_1 \\ f(1,0) = a_1 + a_2 = f_2 \\ f(0,1) = a_1 + a_3 = f_3 \end{cases} \Rightarrow \dots \Rightarrow \begin{cases} \Omega_1 = 1 - \xi - \eta \\ \Omega_2 = \xi \\ \Omega_3 = \eta \end{cases}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 18/57



## ISO 3: funzioni di forma

La funzione generica  $f$  è dunque interpolata come:

$$f(\xi, \eta) = \Omega_1(\xi, \eta) f_1 + \Omega_2(\xi, \eta) f_2 + \Omega_3(\xi, \eta) f_3$$

Applicando l'interpolazione alla geometria e agli spostamenti si ha:

$$\begin{bmatrix} x(\xi, \eta) \\ y(\xi, \eta) \end{bmatrix} = \mathbf{x}(\xi, \eta) = \mathbf{\Omega}(\xi, \eta) \cdot \mathbf{y} \qquad \begin{bmatrix} u_x(\xi, \eta) \\ u_y(\xi, \eta) \end{bmatrix} = \mathbf{u}(\xi, \eta) = \mathbf{\Omega}(\xi, \eta) \cdot \mathbf{v}$$

dove i vettori  $\mathbf{y}$  e  $\mathbf{v}$  raccolgono le coordinate nodali e gli spostamenti nodali rispettivamente e  $\mathbf{\Omega}$  è la matrice delle funzioni di forma:

$$\mathbf{y} = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} v_{x1} \\ v_{y1} \\ v_{x2} \\ v_{y2} \\ v_{x3} \\ v_{y3} \end{bmatrix} \qquad \mathbf{\Omega} = \begin{bmatrix} \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 \\ 0 & \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 19/57



## ISO 3: funzioni di forma per le deformazioni

Applicando l'operatore cinematico si ottiene il vettore delle deformazioni:

$$\boldsymbol{\varepsilon} = \partial_{\text{cin}} \cdot \mathbf{\Omega} \cdot \mathbf{v} = \mathbf{B} \cdot \mathbf{v}$$

dove:

$$\mathbf{B} = \begin{bmatrix} \partial_x & 0 \\ 0 & \partial_y \\ \partial_y & \partial_x \end{bmatrix} \begin{bmatrix} \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 \\ 0 & \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 \end{bmatrix} = \begin{bmatrix} \frac{\partial \Omega_1}{\partial x} & 0 & \frac{\partial \Omega_2}{\partial x} & 0 & \frac{\partial \Omega_3}{\partial x} & 0 \\ 0 & \frac{\partial \Omega_1}{\partial y} & 0 & \frac{\partial \Omega_2}{\partial y} & 0 & \frac{\partial \Omega_3}{\partial y} \\ \frac{\partial \Omega_1}{\partial y} & \frac{\partial \Omega_1}{\partial x} & \frac{\partial \Omega_2}{\partial y} & \frac{\partial \Omega_2}{\partial x} & \frac{\partial \Omega_3}{\partial y} & \frac{\partial \Omega_3}{\partial x} \end{bmatrix}$$

Poiché  $\mathbf{\Omega}$  è espressa in coordinate intrinseche  $(\xi, \eta)$  mentre l'operatore cinematico contiene le derivate rispetto alle coordinate globali  $(x, y)$ , applichiamo la regola di derivazione di funzione composta:

$$\frac{\partial \Omega_i(\xi, \eta)}{\partial x} = \frac{\partial \Omega_i}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \Omega_i}{\partial \eta} \frac{\partial \eta}{\partial x} \qquad \frac{\partial \Omega_i(\xi, \eta)}{\partial y} = \frac{\partial \Omega_i}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \Omega_i}{\partial \eta} \frac{\partial \eta}{\partial y} \qquad i = 1, 2, 3$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 20/57



### ISO 3: cambio di coordinate (quadro)

In forma matriciale: 
$$\begin{bmatrix} \frac{\partial \Omega_i}{\partial x} \\ \frac{\partial \Omega_i}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial \Omega_i}{\partial \xi} \\ \frac{\partial \Omega_i}{\partial \eta} \end{bmatrix} \quad i=1,2,3$$

matrice jacobiana 
$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}$$

inversa della jacobiana 
$$\mathbf{J}^{-1} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix}$$

derivate delle funzioni di forma rispetto alle coordinate intrinseche

$$\frac{\partial \Omega_1}{\partial \xi} = -1 \quad \frac{\partial \Omega_2}{\partial \xi} = 1 \quad \frac{\partial \Omega_3}{\partial \xi} = 0$$

$$\frac{\partial \Omega_1}{\partial \eta} = -1 \quad \frac{\partial \Omega_2}{\partial \eta} = 0 \quad \frac{\partial \Omega_3}{\partial \eta} = 1$$

in questo caso sono costanti!

derivate delle coordinate globali rispetto alle coordinate intrinseche

$$\frac{\partial x}{\partial \xi} = \frac{\partial}{\partial \xi}(\Omega_1 x_1 + \Omega_2 x_2 + \Omega_3 x_3) = -x_1 + x_2 \quad \frac{\partial y}{\partial \xi} = \frac{\partial}{\partial \xi}(\Omega_1 y_1 + \Omega_2 y_2 + \Omega_3 y_3) = -y_1 + y_2$$

$$\frac{\partial x}{\partial \eta} = \frac{\partial}{\partial \eta}(\Omega_1 x_1 + \Omega_2 x_2 + \Omega_3 x_3) = -x_1 + x_3 \quad \frac{\partial y}{\partial \eta} = \frac{\partial}{\partial \eta}(\Omega_1 y_1 + \Omega_2 y_2 + \Omega_3 y_3) = -y_1 + y_3$$


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 21/57



### ISO 3: cambio di coordinate (sostituzione)

matrice jacobiana: 
$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{bmatrix}$$

determinante jacobiano:  $J = \det \mathbf{J} = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1) = 2A_{\text{elemento}}$

jacobiana inversa: 
$$\mathbf{J}^{-1} = \frac{1}{J} \begin{bmatrix} y_3 - y_1 & y_1 - y_2 \\ x_1 - x_3 & x_2 - x_1 \end{bmatrix}$$

È dunque possibile ricavare le componenti di **B**  
(che per questo elemento sono costanti):

$$\begin{bmatrix} \frac{\partial \Omega_1}{\partial x} \\ \frac{\partial \Omega_1}{\partial y} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} -1 \\ -1 \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} y_2 - y_3 \\ x_3 - x_2 \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial \Omega_2}{\partial x} \\ \frac{\partial \Omega_2}{\partial y} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} y_3 - y_1 \\ x_1 - x_3 \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial \Omega_3}{\partial x} \\ \frac{\partial \Omega_3}{\partial y} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} y_1 - y_2 \\ x_2 - x_1 \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 22/57



## ISO 3: matrice di rigidezza

La matrice **B** è costante:

$$\mathbf{B} = \frac{1}{2A} \begin{bmatrix} y_2 - y_3 & 0 & x_3 - x_2 & 0 & y_1 - y_2 & 0 \\ 0 & x_3 - x_2 & y_2 - y_3 & 0 & x_1 - x_3 & 0 & x_2 - x_1 \\ x_3 - x_2 & y_2 - y_3 & x_1 - x_3 & y_3 - y_1 & x_2 - x_1 & y_1 - y_2 \end{bmatrix}$$

Nel calcolo della **matrice di rigidezza**, l'integrale di volume diviene un integrale di superficie e, poiché **B** è costante, **B<sup>T</sup>EB** può essere portato fuori dall'integrale:

$$\mathbf{k} = \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} \, dV = \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} \cdot \int_S dA =$$

$$= \frac{1}{2A} \begin{bmatrix} y_2 - y_3 & 0 & x_3 - x_2 \\ 0 & x_3 - x_2 & y_2 - y_3 \\ y_3 - y_1 & 0 & x_1 - x_3 \\ 0 & x_1 - x_3 & y_3 - y_1 \\ y_1 - y_2 & 0 & x_2 - x_1 \\ 0 & x_2 - x_1 & y_1 - y_2 \end{bmatrix} \cdot \frac{Eh}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \cdot \frac{1}{2A} \begin{bmatrix} y_2 - y_3 & 0 & x_3 - x_2 & 0 & y_1 - y_2 & 0 \\ 0 & x_3 - x_2 & y_2 - y_3 & 0 & x_1 - x_3 & 0 & x_2 - x_1 \\ x_3 - x_2 & y_2 - y_3 & x_1 - x_3 & y_3 - y_1 & x_2 - x_1 & y_1 - y_2 \end{bmatrix} \cdot A =$$

$$= \frac{1}{4A} \frac{Eh}{1-\nu^2} \begin{bmatrix} (y_2 - y_3)^2 + (x_3 - x_2)^2 (1-\nu)/2 & \dots & k_{13} & k_{14} & k_{15} & k_{16} \\ \dots & \dots & k_{23} & k_{24} & k_{25} & k_{26} \\ \dots & \dots & k_{33} & k_{34} & k_{35} & k_{36} \\ \dots & \dots & \dots & k_{44} & k_{45} & k_{46} \\ \dots & \dots & \dots & \dots & k_{55} & k_{56} \\ \dots & \dots & \dots & \dots & \dots & k_{66} \end{bmatrix}$$

*sym*



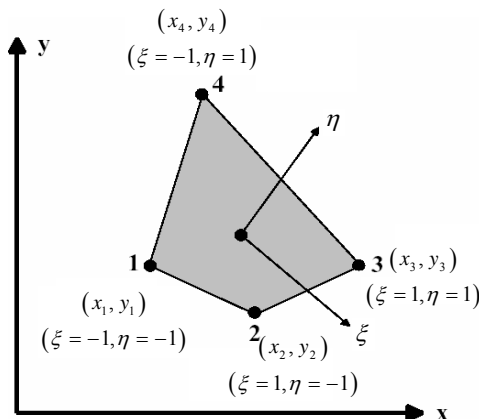
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 23/57



## ISO 4: elemento isoparametrico piano a 4 nodi

Definizione di un sistema  $(\xi, \eta)$  di coordinate intrinseche:



Vettori delle coordinate nodali e degli spostamenti nodali:

$$\mathbf{y} = \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ x_4 \\ y_4 \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} v_{x1} \\ v_{y1} \\ v_{x2} \\ v_{y2} \\ v_{x3} \\ v_{y3} \\ v_{x4} \\ v_{y4} \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 24/57



## ISO 4: interpolazione

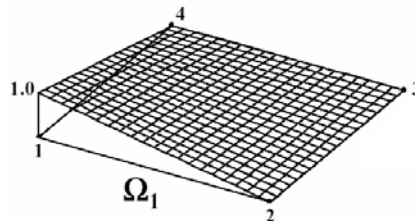
Il polinomio interpolante deve contenere 4 coefficienti (uno per nodo). Per garantire l'isotropia geometrica (cfr. cap. 3), deve essere scelto simmetricamente rispetto alle coordinate  $(\xi, \eta)$ . Si utilizza un polinomio bi-lineare (secondo grado incompleto):

$$f(\xi, \eta) = a_1 + a_2\xi + a_3\eta + a_4\xi\eta$$

Imponendo i valori nodali, si ricavano le funzioni di forma:

$$\begin{cases} f(-1, -1) = a_1 - a_2 - a_3 + a_4 = f_1 \\ f(+1, -1) = a_1 + a_2 - a_3 - a_4 = f_2 \\ f(+1, +1) = a_1 + a_2 + a_3 + a_4 = f_3 \\ f(-1, +1) = a_1 - a_2 + a_3 - a_4 = f_4 \end{cases} \Rightarrow$$

$$\Rightarrow \dots \Rightarrow \begin{cases} \Omega_1 = \frac{1}{4}(1-\xi)(1-\eta) \\ \Omega_2 = \frac{1}{4}(1+\xi)(1-\eta) \\ \Omega_3 = \frac{1}{4}(1+\xi)(1+\eta) \\ \Omega_4 = \frac{1}{4}(1-\xi)(1+\eta) \end{cases}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 25/57



## ISO 4: funzioni di forma

La funzione generica  $f$  è dunque interpolata come:

$$f(\xi, \eta) = \Omega_1(\xi, \eta)f_1 + \Omega_2(\xi, \eta)f_2 + \Omega_3(\xi, \eta)f_3 + \Omega_4(\xi, \eta)f_4$$

Applicando l'interpolazione alla geometria e agli spostamenti si ha:

$$\begin{bmatrix} x(\xi, \eta) \\ y(\xi, \eta) \end{bmatrix} = \mathbf{x}(\xi, \eta) = \mathbf{\Omega}(\xi, \eta) \cdot \mathbf{y} \qquad \begin{bmatrix} u_x(\xi, \eta) \\ u_y(\xi, \eta) \end{bmatrix} = \mathbf{u}(\xi, \eta) = \mathbf{\Omega}(\xi, \eta) \cdot \mathbf{v}$$

dove  $\mathbf{\Omega}$  è la matrice delle funzioni di forma:

$$\mathbf{\Omega} = \begin{bmatrix} \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 & \Omega_4 & 0 \\ 0 & \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 & \Omega_4 \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 26/57



## ISO 4: interpolazione delle deformazioni

Applicando l'operatore cinematico si ottiene:

$$\mathbf{B} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \begin{bmatrix} \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 & \Omega_4 & 0 \\ 0 & \Omega_1 & 0 & \Omega_2 & 0 & \Omega_3 & 0 & \Omega_4 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\partial \Omega_1}{\partial x} & 0 & \frac{\partial \Omega_2}{\partial x} & 0 & \frac{\partial \Omega_3}{\partial x} & 0 & \frac{\partial \Omega_4}{\partial x} & 0 \\ 0 & \frac{\partial \Omega_1}{\partial y} & 0 & \frac{\partial \Omega_2}{\partial y} & 0 & \frac{\partial \Omega_3}{\partial y} & 0 & \frac{\partial \Omega_4}{\partial y} \\ \frac{\partial \Omega_1}{\partial y} & \frac{\partial \Omega_1}{\partial x} & \frac{\partial \Omega_2}{\partial y} & \frac{\partial \Omega_2}{\partial x} & \frac{\partial \Omega_3}{\partial y} & \frac{\partial \Omega_3}{\partial x} & \frac{\partial \Omega_4}{\partial y} & \frac{\partial \Omega_4}{\partial x} \end{bmatrix}$$

e con la regola di derivazione di funzione composta:

$$\frac{\partial \Omega_i(\xi, \eta)}{\partial x} = \frac{\partial \Omega_i}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \Omega_i}{\partial \eta} \frac{\partial \eta}{\partial x} \quad \frac{\partial \Omega_i(\xi, \eta)}{\partial y} = \frac{\partial \Omega_i}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \Omega_i}{\partial \eta} \frac{\partial \eta}{\partial y} \quad i = 1, 2, 3, 4$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 27/57



## ISO 4: cambio di coordinate (quadro)

In forma matriciale:

$$\begin{bmatrix} \frac{\partial \Omega_i}{\partial x} \\ \frac{\partial \Omega_i}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial \Omega_i}{\partial \xi} \\ \frac{\partial \Omega_i}{\partial \eta} \end{bmatrix} \quad i = 1, 2, 3, 4$$

matrice jacobiana

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}$$

$\mathbf{J}^{-1}$

derivate delle funzioni di forma  
rispetto alle coordinate intrinseche

$$\frac{\partial \Omega_1}{\partial \xi} = -\frac{1}{4}(1-\eta) \quad \frac{\partial \Omega_2}{\partial \xi} = +\frac{1}{4}(1-\eta) \quad \dots$$

$$\frac{\partial \Omega_1}{\partial \eta} = -\frac{1}{4}(1-\xi) \quad \frac{\partial \Omega_2}{\partial \eta} = -\frac{1}{4}(1+\xi) \quad \dots$$

Non sono costanti ma  
dipendono da  $(\xi, \eta)$ !

derivate delle coordinate  
globali rispetto alle  
coordinate intrinseche

$$\frac{\partial x}{\partial \xi} = \sum_{i=1}^4 \frac{\partial \Omega_i}{\partial \xi} x_i \quad \frac{\partial y}{\partial \xi} = \sum_{i=1}^4 \frac{\partial \Omega_i}{\partial \xi} y_i$$

$$\frac{\partial x}{\partial \eta} = \sum_{i=1}^4 \frac{\partial \Omega_i}{\partial \eta} x_i \quad \frac{\partial y}{\partial \eta} = \sum_{i=1}^4 \frac{\partial \Omega_i}{\partial \eta} y_i$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 28/57



## ISO 4: cambio di coordinate (sostituzione)

Derivate delle funzioni di forma rispetto alle coordinate intrinseche:

$$\frac{\partial \Omega_1}{\partial \xi} = -\frac{1}{4}(1-\eta) \quad \frac{\partial \Omega_2}{\partial \xi} = +\frac{1}{4}(1-\eta) \quad \dots$$

$$\frac{\partial \Omega_1}{\partial \eta} = -\frac{1}{4}(1-\xi) \quad \frac{\partial \Omega_2}{\partial \eta} = -\frac{1}{4}(1+\xi) \quad \dots$$

polinomi (lineari)

matrice jacobiana della trasformazione:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \sum_{i=1}^4 \begin{bmatrix} \frac{\partial \Omega_i}{\partial \xi} x_i & \frac{\partial \Omega_i}{\partial \xi} y_i \\ \frac{\partial \Omega_i}{\partial \eta} x_i & \frac{\partial \Omega_i}{\partial \eta} y_i \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}$$

polinomi (lineari)

determinante jacobiano:

$$J = \det \mathbf{J} = J_{11}J_{22} - J_{12}J_{21}$$

polinomio (quadratico)

matrice jacobiana inversa:

$$\mathbf{J}^{-1} = \frac{1}{J} \begin{bmatrix} J_{22} & -J_{21} \\ -J_{12} & J_{11} \end{bmatrix}$$

funzioni razionali



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 29/57



## ISO 4: funzioni di forma per le deformazioni

Le derivate delle funzioni di forma rispetto alle coordinate globali sono dunque **funzioni razionali** (rapporto di polinomi):

$$\begin{bmatrix} \frac{\partial \Omega_i}{\partial x} \\ \frac{\partial \Omega_i}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial \Omega_i}{\partial \xi} \\ \frac{\partial \Omega_i}{\partial \eta} \end{bmatrix} \quad i = 1, 2, 3, 4 \quad (\text{funzioni razionali}) \times (\text{polinomi})$$

||  
 $\mathbf{J}^{-1}$

la **matrice delle funzioni di forma per le deformazioni** è perciò una funzione non polinomiale delle coordinate intrinseche:

$$\mathbf{B}(\xi, \eta) = \begin{bmatrix} \frac{\partial \Omega_1}{\partial x} & 0 & \frac{\partial \Omega_2}{\partial x} & 0 & \frac{\partial \Omega_3}{\partial x} & 0 & \frac{\partial \Omega_4}{\partial x} & 0 \\ 0 & \frac{\partial \Omega_1}{\partial y} & 0 & \frac{\partial \Omega_2}{\partial y} & 0 & \frac{\partial \Omega_3}{\partial y} & 0 & \frac{\partial \Omega_4}{\partial y} \\ \frac{\partial \Omega_1}{\partial y} & \frac{\partial \Omega_1}{\partial x} & \frac{\partial \Omega_2}{\partial y} & \frac{\partial \Omega_2}{\partial x} & \frac{\partial \Omega_3}{\partial y} & \frac{\partial \Omega_3}{\partial x} & \frac{\partial \Omega_4}{\partial y} & \frac{\partial \Omega_4}{\partial x} \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 30/57



## ISO 4: matrice di rigidità (cambio di coordinate)

Consideriamo adesso l'espressione della matrice di rigidità così come ottenuta tramite il PLV:

$$\mathbf{k} = \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV =$$

in un elemento 2D l'integrale di volume diviene un integrale di superficie

$$= \int_{S_{elem}} (\mathbf{B}(\xi, \eta))^T \cdot \mathbf{E} \cdot \mathbf{B}(\xi, \eta) dx dy =$$

si effettua il cambio delle variabili di integrazione (ricordando che:  $dA = dx dy = d\xi d\eta J$ )

$$= \int_{-1}^{+1} \int_{-1}^{+1} (\mathbf{B}(\xi, \eta))^T \cdot \mathbf{E} \cdot \mathbf{B}(\xi, \eta) J(\xi, \eta) d\xi d\eta$$

le funzioni integrande sono razionali: non è possibile integrare analiticamente!

Integrazione numerica!



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 31/57



## Integrazione numerica: idea

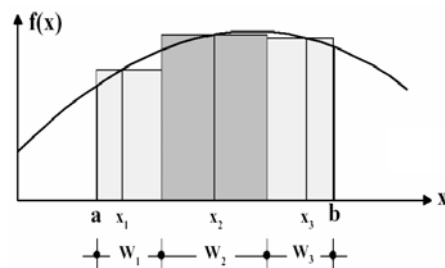
L'integrazione numerica è necessaria in tutti gli elementi continui 2D e 3D (esclusi il triangolo a 3 nodi ed il tetraedro a 4 nodi).

L'algoritmo più usato è la **quadratura di Gauss**.

- La funzione viene campionata in specifici punti  $\tilde{x}_i$  (punti di integrazione o di Gauss).
- A ciascun punto di integrazione corrisponde un dominio la cui ampiezza è detta **peso** del punto di integrazione.
- L'integrale si trasforma in sommatoria, ad esempio in 1D:

$$\int_a^b f(x) dx \approx \sum_{i=1}^N f(\tilde{x}_i) w_i$$

Esempio con tre punti di integrazione



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 32/57





## Integrazione numerica: teoria

La sommatoria è estesa ad  $N$  punti ( $N$  è detto ordine di integrazione).

Un polinomio di grado  $p$  è integrato esattamente per  $N \geq \frac{p+1}{2}$

Polinomi di grado superiore o funzioni non polinomiali (come le funzioni razionali) sono pertanto integrate solo in maniera approssimata (e l'approssimazione è tanto migliore quanto più alto è  $N$ ).

Le posizioni dei punti di Gauss e i relativi pesi sono noti dall'analisi matematica nell'intervallo  $[-1,1]$  (ogni altro intervallo  $[a,b]$  è ottenibile con un semplice cambio di coordinate):

$$\int_{-1}^1 f(\xi) d\xi \approx \sum_{i=1}^N f(\tilde{\xi}_i) w_i$$

$N$	$\tilde{\xi}_i$	$w_i$
1	0	2
2	$-\sqrt{\frac{1}{3}}, \sqrt{\frac{1}{3}}$	$\frac{1}{2}, \frac{1}{2}$
3	$-\sqrt{\frac{3}{5}}, 0, \sqrt{\frac{3}{5}}$	$\frac{8}{9}, \frac{8}{9}, \frac{8}{9}$
4	...	...



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 33/57



## Integrazione numerica: estensione 2D e 3D

L'integrazione numerica può essere immediatamente estesa a integrali bidimensionali:

$$\int_{-1}^1 \int_{-1}^1 f(\xi, \eta) d\xi d\eta = \sum_{j=1}^N \sum_{i=1}^N f(\tilde{\xi}_i, \tilde{\eta}_j) w_i w_j$$

... e tridimensionali:

$$\int_{-1}^1 \int_{-1}^1 \int_{-1}^1 f(\xi, \eta, \zeta) d\xi d\eta d\zeta = \sum_{k=1}^N \sum_{j=1}^N \sum_{i=1}^N f(\tilde{\xi}_i, \tilde{\eta}_j, \tilde{\zeta}_k) w_i w_j w_k$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 34/57

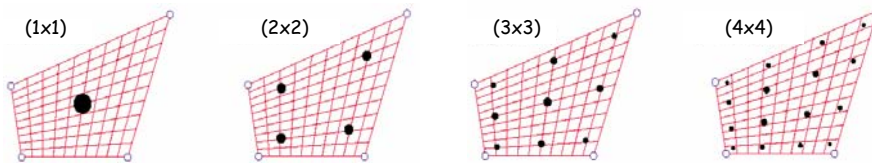


## ISO 4: matrice di rigidità

Applichiamo l'integrazione numerica all'elemento ISO 4:

$$\begin{aligned} \mathbf{k} &= \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV = \\ &= \int_{-1}^{+1} \int_{-1}^{+1} (\mathbf{B}(\xi, \eta))^T \cdot \mathbf{E} \cdot \mathbf{B}(\xi, \eta) J(\xi, \eta) d\xi d\eta = \\ &= \sum_{i=1}^N \sum_{j=1}^N w_i w_j (\mathbf{B}(\tilde{\xi}_i, \tilde{\eta}_j))^T \cdot \mathbf{E} \cdot \mathbf{B}(\tilde{\xi}_i, \tilde{\eta}_j) J(\tilde{\xi}_i, \tilde{\eta}_j) \end{aligned}$$

Schemi di integrazione e localizzazione dei punti di Gauss



[figura per gentile concessione del Prof. C. Felippa]



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori

(rev. 19/05/2007) Capitolo 4: 35/57



## ISO 4: procedura operativa

- si inizializza la rigidità a zero
- si itera sui punti di Gauss
  - in ciascun punto si valutano le derivate delle funzioni di forma lungo le coordinate intrinseche (essendo valutate in un punto adesso sono numeril)
  - con le coordinate nodali si calcola la matrice jacobiana nel punto
  - si calcola il determinante jacobiano e si inverte numericamente la matrice  $\mathbf{J}$  (che essendo valutata in un punto ben preciso è ora fatta di numeril);
  - si valutano le derivate delle funzioni di forma lungo le coordinate globali (componenti di  $\mathbf{B}$ );
  - si calcola il prodotto  $\mathbf{B}^T \mathbf{E} \mathbf{B}$ , lo si moltiplica per il peso di integrazione del punto di Gauss e lo si somma a  $\mathbf{k}$ .

$$\mathbf{k} := \mathbf{0}$$

$$i, j = 1, \dots, N$$

$$\frac{\partial \Omega_i}{\partial \xi} = -\frac{1}{4}(1 - \tilde{\eta}_j) \dots$$

$$\frac{\partial \Omega_i}{\partial \eta} = -\frac{1}{4}(1 - \tilde{\xi}_i) \dots$$

$$\mathbf{J}(\tilde{\xi}_i, \tilde{\eta}_j) = \sum_{n=1}^4 \begin{bmatrix} \frac{\partial \Omega_n}{\partial \xi} x_i & \frac{\partial \Omega_n}{\partial \xi} y_n \\ \frac{\partial \Omega_n}{\partial \eta} x_n & \frac{\partial \Omega_n}{\partial \eta} y_n \end{bmatrix}$$

$$J(\tilde{\xi}_i, \tilde{\eta}_j) = \det \mathbf{J}, \rightarrow \mathbf{J}^{-1}(\tilde{\xi}_i, \tilde{\eta}_j)$$

$$\begin{bmatrix} \frac{\partial \Omega_i}{\partial x} \\ \frac{\partial \Omega_i}{\partial y} \end{bmatrix} = \mathbf{J}^{-1}(\tilde{\xi}_i, \tilde{\eta}_j) \begin{bmatrix} \frac{\partial \Omega_i}{\partial \xi} \\ \frac{\partial \Omega_i}{\partial \eta} \end{bmatrix}$$

$$\mathbf{k} := \mathbf{k} + w_i w_j \mathbf{B}^T(\tilde{\xi}_i, \tilde{\eta}_j) \cdot \mathbf{E} \cdot \mathbf{B}(\tilde{\xi}_i, \tilde{\eta}_j) J(\tilde{\xi}_i, \tilde{\eta}_j)$$



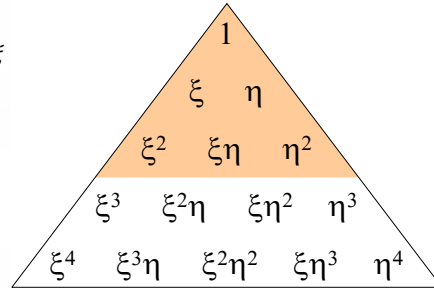
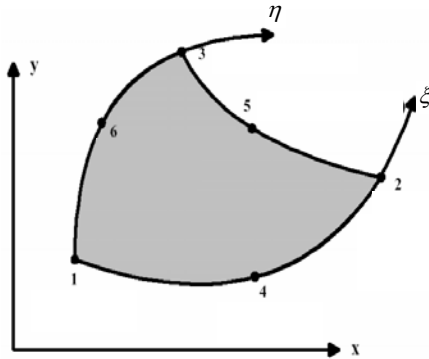
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori

(rev. 19/05/2007) Capitolo 4: 36/57



## ISO 6: scelta del polinomio



polinomio completo di secondo grado

$$f(\xi, \eta) = a_1 + a_2\xi + a_3\eta + a_4\xi^2 + a_5\xi\eta + a_6\eta^2$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 37/57



## ISO 6: funzioni di forma

$$\Omega_1(\xi, \eta) = 1 - 3\xi - 3\eta + 2\xi^2 + 4\xi\eta + 2\eta^2$$

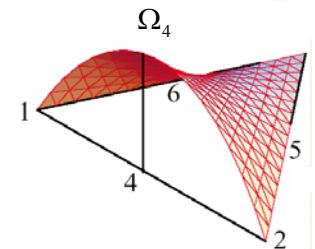
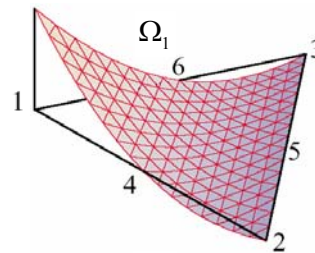
$$\Omega_2(\xi, \eta) = -\xi + 2\xi^2$$

$$\Omega_3(\xi, \eta) = -\eta + 2\eta^2$$

$$\Omega_4(\xi, \eta) = 4\xi - 4\xi^2 - 4\xi\eta$$

$$\Omega_5(\xi, \eta) = 4\xi\eta$$

$$\Omega_6(\xi, \eta) = 4\eta - 4\xi\eta - 4\eta^2$$

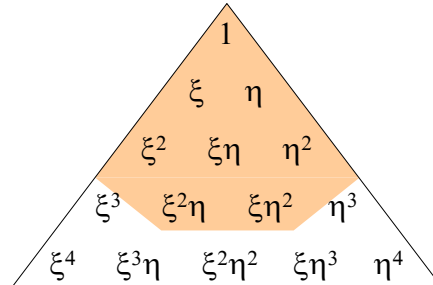
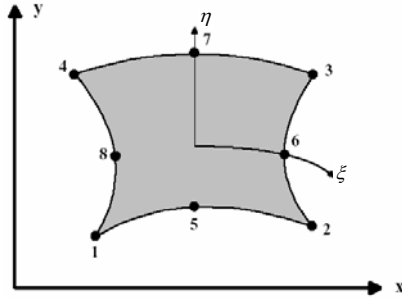


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 38/57



## ISO 8: scelta del polinomio



polinomio incompleto di terzo grado ("serendipity")

$$f(\xi, \eta) = a_1 + a_2\xi + a_3\eta + a_4\xi^2 + a_5\xi\eta + a_6\eta^2 + a_7\xi^2\eta + a_8\xi\eta^2$$

NB: la simmetria fra  $\xi$  e  $\eta$  è rispettata (isotropia geometrica)



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 39/57



## ISO 8: funzioni di forma

$$\Omega_1(\xi, \eta) = \frac{1}{4}(-1 + \xi^2 + \xi\eta + \eta^2 - \xi^2\eta - \xi\eta^2)$$

$$\Omega_2(\xi, \eta) = \frac{1}{4}(-1 + \xi^2 - \xi\eta + \eta^2 - \xi^2\eta + \xi\eta^2)$$

$$\Omega_3(\xi, \eta) = \frac{1}{4}(-1 + \xi^2 + \xi\eta + \eta^2 + \xi^2\eta + \xi\eta^2)$$

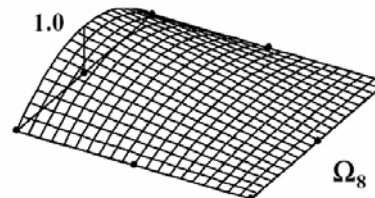
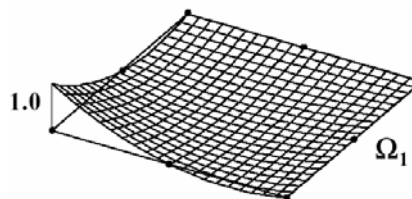
$$\Omega_4(\xi, \eta) = \frac{1}{4}(-1 + \xi^2 - \xi\eta + \eta^2 + \xi^2\eta - \xi\eta^2)$$

$$\Omega_5(\xi, \eta) = \frac{1}{2}(1 - \eta)(1 - \xi^2)$$

$$\Omega_6(\xi, \eta) = \frac{1}{2}(1 + \xi)(1 - \eta^2)$$

$$\Omega_7(\xi, \eta) = \frac{1}{2}(1 + \eta)(1 - \xi^2)$$

$$\Omega_8(\xi, \eta) = \frac{1}{2}(1 - \xi)(1 - \eta^2)$$

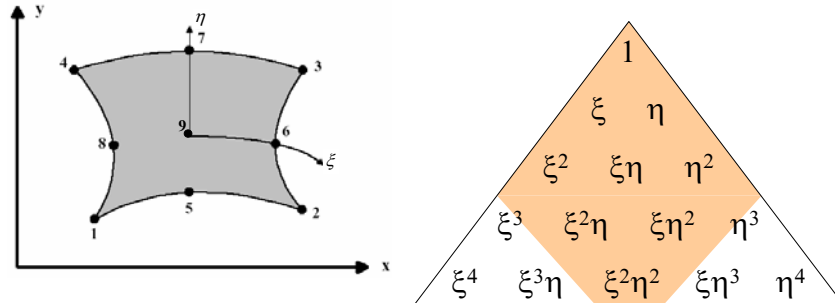


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 40/57



## ISO 9: scelta del polinomio



polinomio bi-quadratico (incompleto di quarto grado)

$$f(\xi, \eta) = a_1 + a_2\xi + a_3\eta + a_4\xi^2 + a_5\xi\eta + a_6\eta^2 + a_7\xi^2\eta + a_8\xi\eta^2 + a_9\xi^2\eta^2$$

N.B.: la simmetria fra  $\xi$  e  $\eta$  è rispettata.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 41/57



## ISO9: funzioni di forma

$$\Omega_1(\xi, \eta) = \frac{1}{2}(\xi^2 - \xi)\frac{1}{2}(\eta^2 - \eta)$$

$$\Omega_2(\xi, \eta) = \frac{1}{2}(\xi^2 + \xi)\frac{1}{2}(\eta^2 - \eta)$$

$$\Omega_3(\xi, \eta) = \frac{1}{2}(\xi^2 + \xi)\frac{1}{2}(\eta^2 + \eta)$$

$$\Omega_4(\xi, \eta) = \frac{1}{2}(\xi^2 - \xi)\frac{1}{2}(\eta^2 + \eta)$$

$$\Omega_5(\xi, \eta) = (1 - \xi^2)\frac{1}{2}(\eta^2 - \eta)$$

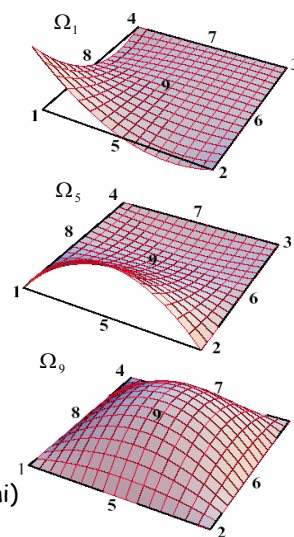
$$\Omega_6(\xi, \eta) = \frac{1}{2}(\xi^2 + \xi)(1 - \eta^2)$$

$$\Omega_7(\xi, \eta) = (1 - \xi^2)\frac{1}{2}(\eta^2 + \eta)$$

$$\Omega_8(\xi, \eta) = \frac{1}{2}(\xi^2 - \xi)(1 - \eta^2)$$

$$\Omega_9(\xi, \eta) = (1 - \xi^2)(1 - \eta^2)$$

(sono i prodotto delle funzioni di forma monodimensionali per tre nodi nelle due dimensioni)



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 42/57



## Tabella riassuntiva dei gradi di approssimazione per gli elementi isoparametrici piani

	grado del polinomio (x,y)				
	ISO 3	ISO 4	ISO 6	ISO 8	ISO 9
$u_x$	1	1	2	2	2
$u_y$	1	1	2	2	2
$n_{xx}$	0	0,1	1	1,2	1,2
$n_{xy}$	0	1	1	1	2
$n_{yy}$	0	1,0	1	2,1	2,1

È possibile costruire elementi di grado elevato quanto si vuole, tuttavia è in genere preferibile infittire la discretizzazione piuttosto che utilizzare elementi di grado superiore al terzo.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 43/57



## Condizioni di convergenza per gli elementi isoparametrici

- 1) **Continuità e differenziabilità:** questi requisiti sono automaticamente soddisfatti da qualsiasi approssimazione polinomiale.
- 2) **Completezza:** è possibile dimostrare che in un'interpolazione polinomiale:
  - 2a) i moti rigidi vengono rappresentati se il polinomio è almeno completo al primo grado;
  - 2b) i moti a deformazione costante vengono rappresentati se il polinomio è almeno completo al grado pari all'ordine di derivazione presente nella congruenza (nel caso dei continui in piccole deformazioni, ancora il primo grado).
 Come si è visto, già l'elemento ISO 3 è completo al primo grado.
- 3) **Conformità:** ci si pone su un bordo dell'elemento (e.g. nell'elemento ISO 4  $\xi=-1$ ) e si verifica che si annullano tutte le funzioni di forma dei nodi che non appartengono al bordo considerato (nell'esempio  $\Omega_2=\Omega_3=0$ ).
- 4) **Isotropia geometrica:** scegliendo il polinomio interpolante si è avuto cura di far comparire simmetricamente le coordinate  $(\xi, \eta)$ .



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

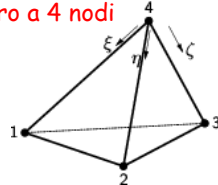
Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 44/57



## Elementi isoparametrici continui 3D

L'estensione alla terza dimensione di quanto visto per i continui 2D è immediata, solo a titolo di esempio:

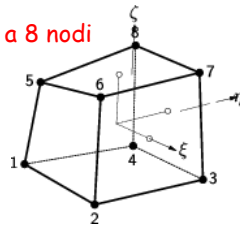
tetraedro a 4 nodi



polinomio completo di primo grado:

$$f(\xi, \eta, \zeta) = a_1 + a_2\xi + a_3\eta + a_4\zeta$$

esaedro a 8 nodi



polinomio tri-lineare:

$$f(\xi, \eta, \zeta) = a_1 + a_2\xi + a_3\eta + a_4\zeta + a_5\xi\eta + a_6\xi\zeta + a_7\eta\zeta + a_8\xi\eta\zeta$$



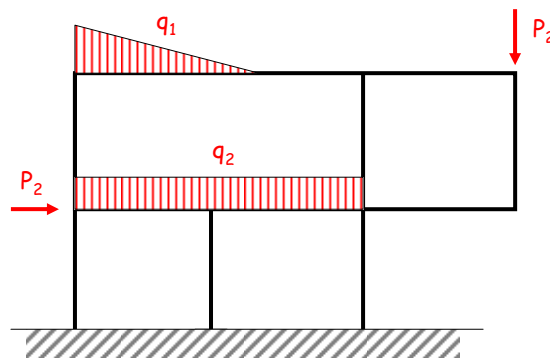
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 45/57



## Trave 2D

La trave è un elemento inflesso, usato frequentemente nella pratica dell'ingegneria civile (modellazione di strutture intelaiate).

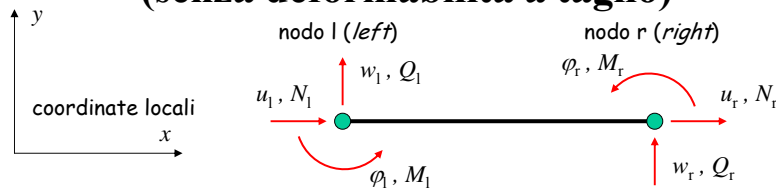


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 46/57



## Trave 2D: rigidezza in coordinate locali (senza deformabilità a taglio)



- I gradi di libertà nodali sono le due traslazioni e la rotazione.
- Le corrispondenti azioni sono le caratteristiche di sollecitazione (sforzo normale, taglio e momento flettente).
- La matrice di rigidità in coordinate locali può facilmente essere ricavata con il metodo dello spostamento unitario (cfr. cap. 3).

$$\begin{bmatrix} N_l \\ Q_l \\ M_l \\ N_r \\ Q_r \\ M_r \end{bmatrix} = \begin{bmatrix} \frac{EA}{l} & 0 & 0 & -\frac{EA}{l} & 0 & 0 \\ 0 & \frac{12EJ}{l^3} & -\frac{6EJ}{l^2} & 0 & -\frac{12EJ}{l^3} & -\frac{6EJ}{l^2} \\ 0 & -\frac{6EJ}{l^2} & \frac{4EJ}{l} & 0 & \frac{6EJ}{l^2} & \frac{2EJ}{l} \\ -\frac{EA}{l} & 0 & 0 & \frac{EA}{l} & 0 & 0 \\ 0 & -\frac{12EJ}{l^3} & \frac{6EJ}{l^2} & 0 & \frac{12EJ}{l^3} & \frac{6EJ}{l^2} \\ 0 & -\frac{6EJ}{l^2} & \frac{2EJ}{l} & 0 & \frac{6EJ}{l^2} & \frac{4EJ}{l} \end{bmatrix} \begin{bmatrix} u_l \\ w_l \\ \varphi_l \\ u_r \\ w_r \\ \varphi_r \end{bmatrix}$$

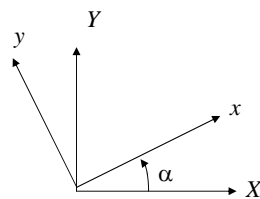


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 47/57



## Trave 2D: trasformazione di coordinate



$$\begin{bmatrix} u_x \\ u_y \\ \varphi \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ w \\ \varphi \end{bmatrix}$$

$$\begin{bmatrix} u_{xl} \\ u_{yl} \\ \varphi_l \\ u_{xr} \\ u_{yr} \\ \varphi_r \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & 0 & 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_l \\ w_l \\ \varphi_l \\ u_r \\ w_r \\ \varphi_r \end{bmatrix}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 48/57





## Panoramica dei tipi di elementi più comuni

- Elementi 1D continui:
  - bielle
  - cavi
- Elementi 1D inflessi:
  - travi
- Elementi 2D continui:
  - triangoli
  - quadrilateri con nodi interni
  - quadrilateri "serendipity"
- Elementi 2D inflessi:
  - lastre inflesse
  - gusci piani
  - gusci curvi
- Elementi 3D continui:
  - tetraedri
  - "cunei"
  - esaedri
- Elementi speciali:
  - continui semi-infiniti
  - pre-fessurati
  - assial-simmetrici
  - macro-elementi
  - interfaccia
  - gusci multi-strato
  - ...



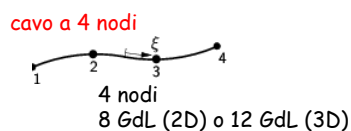
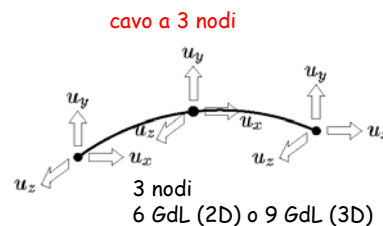
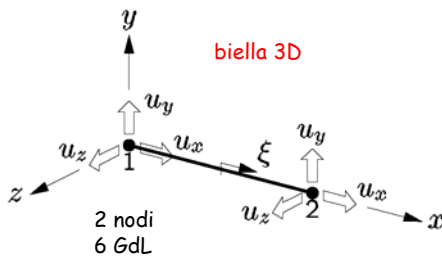
Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

Claudio Borri  
 Luca Salvatori  
 (rev. 19/05/2007) Capitolo 4: 49/57



## Continui 1D: bielle e cavi

Localmente hanno una sola coordinata, ma tramite le opportune trasformazioni divengono elementi bi- e tri- dimensionali con 2 o 3 GdL per nodo (spostamenti).



Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

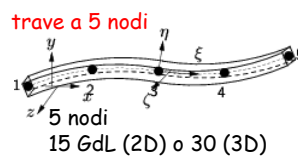
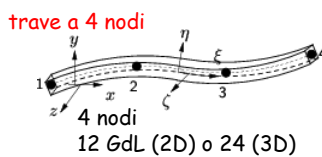
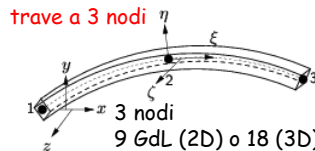
Claudio Borri  
 Luca Salvatori  
 (rev. 19/05/2007) Capitolo 4: 50/57



## Elementi 1D inflessi: travi



Hanno come gradi di libertà nodali anche le rotazioni.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 51/57

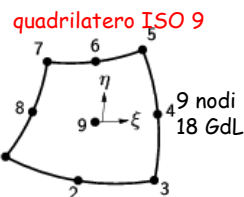
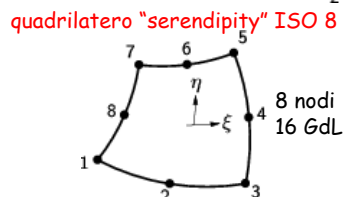
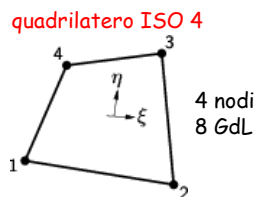
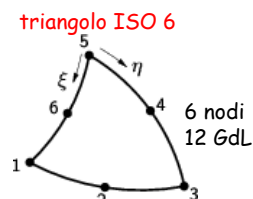
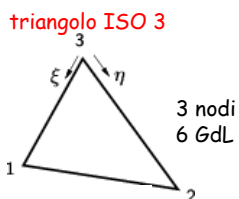


## Continui 2D: 2 GdL per nodo (spostamenti)

Risolvono problemi continui piani nelle versioni: - stato piano di tensione

- stato piano di deformazione

Ruotati in 3D hanno funzionamento a membrana (e 3GdL traslazionali per nodo)



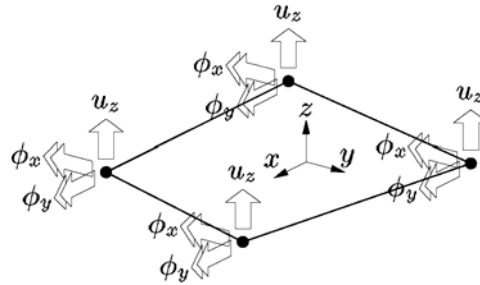
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 52/57

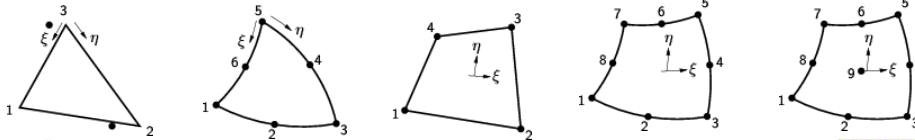


## Elementi 2D inflessi: lastre

I gradi di libertà nodali sono le rotazioni nel piano dell'elemento e lo spostamento ortogonale (3 GdL per nodo)



stessi schemi nodali degli elementi membrana



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

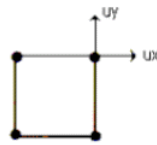
Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 53/57



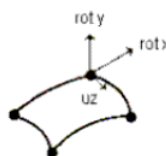
## Elementi 2D inflessi: gusci, gusci curvi

Combinazione di elementi membrana ed elementi lastra inflessa (solitamente viene aggiunta una "piccola" rigidezza fittizia alla rotazione attorno all'asse normale al piano dell'elemento):

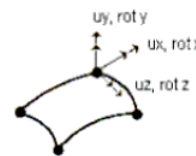
membrana (2 GdL)



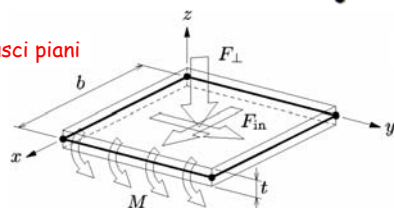
lastra inflessa (3 GdL)



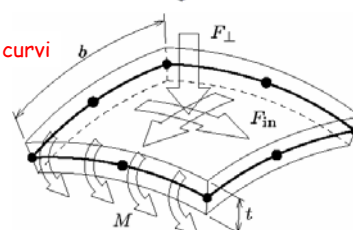
guscio (5+1 GdL)



gusci piani



gusci curvi



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 54/57



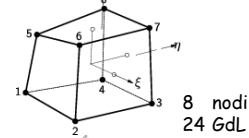
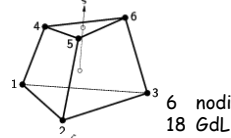
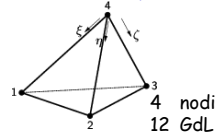
## Continui 3D: 3 GdL per nodo (spostamenti)

**tetraedri** (estensione dei triangoli alla terza dimensione)

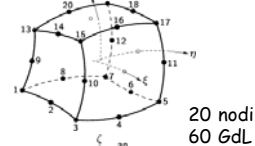
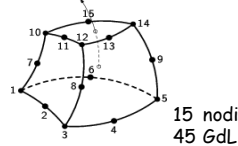
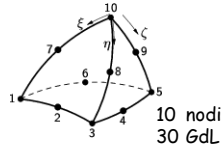
**"cunei"** (triangoli con funzioni di forma della biella nella terza dimensione)

**esaedri** (estensione dei quadrilateri alla terza dimensione)

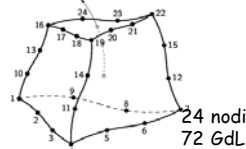
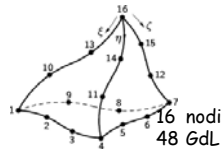
lineari



quadratici



cubici



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

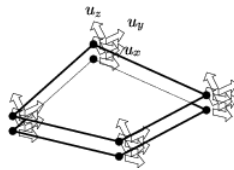
Claudio Borri  
Luca Salvatori

(rev. 19/05/2007) Capitolo 4: 55/57

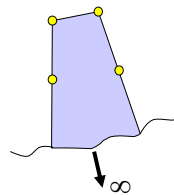


## Elementi speciali

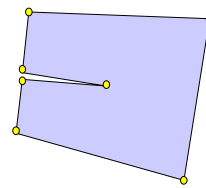
**elementi interfaccia**



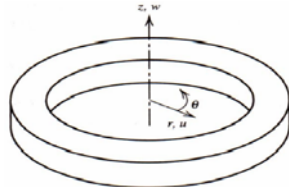
**elementi semi-infiniti**



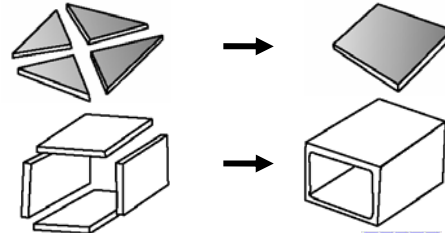
**elementi con frattura pre-esistente**



**elementi assial-simmetrici**



**macro-elementi**



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori

(rev. 19/05/2007) Capitolo 4: 56/57



## Nel prossimo capitolo

### Modellazione delle strutture



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Claudio Borri  
Luca Salvatori  
(rev. 19/05/2007) Capitolo 4: 57/57

