Complementi 16 - L'approccio numerico

[Ultima revisione: 21 febbraio 2009]

Per travi a sezione retta irregolare non sono piu' ipotizzabili soluzioni analitiche, e spesso e' necessario ricorrere ad approcci numerici, che possano fornire una risposta approssimata ma affidabile al problema del calcolo della funzione di torsione, del quadro tensionale e del momento torcente. In questa Lezione si illustra il classico approccio agli elementi finiti, utilizzando le piu' basse leggi di interpolazione e la piu' semplice geometria.

La discretizzazione del problema

Come si e' visto, l'analisi della torsione puo' ricondursi, tramite l'introduzione della funzione di Prandtl, allo studio di una equazione di Poisson:

$$\frac{\partial^2 \phi}{\partial x_1^2} + \frac{\partial^2 \phi}{\partial x_2^2} = -2\,G\,\Theta \tag{1}$$

valida sulla sezione retta Σ, assieme alle condizioni che annullano ϕ sul contorno Γ:

$$\phi\,(\Gamma) = 0 \tag{2}$$

Alternativamente a questa formulazione differenziale, si e' anche visto che la funzione di torsione ϕ puo' ricavarsi dal principio di minimo dell'energia potenziale totale, che nel caso in esame si scrive come:

$$E_t = \frac{L}{2\,G} \int_\Sigma \left(\left(\frac{\partial \phi}{\partial x_1} \right)^2 + \left(\frac{\partial \phi}{\partial x_2} \right)^2 \right) d\Sigma - 2\,\Theta'\,L \int_\Sigma \phi\, d\Sigma \tag{3}$$

E' questa la formulazione integrale dello stesso problema, gia' utilizzata per ottenere le soluzioni semianalitiche, nella Lezione precedente.

■ Il calcolo delle quantita' elementari

Per giungere ad una conoscenza approssimata della funzione ϕ, si inizi col ricoprire la sezione retta Σ con un insieme di triangoli, della forma piu' regolare possibile, identificando in tal modo M triangoli e T nodi, intendendo con tal termine i vertici dei triangoli. (cfr. Figura 1). Su ciascuno di questi triangoli si supporra' che la funzione ϕ vari con legge lineare, sicche' si potra' porre:

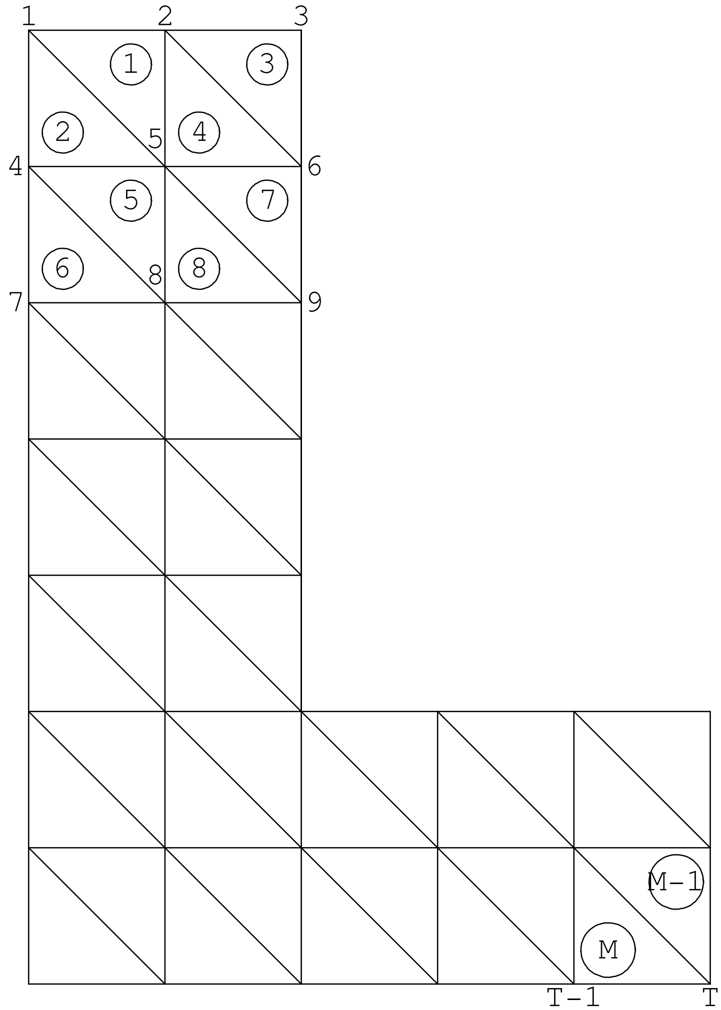

Figura 1 - Una discretizzazione triangolare di una sezione ad L

$$\phi = \phi\,(x_1,\ x_2) = A_0 + A_1\, x_1 + A_2\, x_2 = \boldsymbol{\alpha}\, \mathbf{A} \tag{4}$$

$\boldsymbol{\alpha}$ e' un vettore riga di monomi, mentre $\mathbf{A}$ e' un vettore colonna di costanti, dette incognite lagrangiane:

$$\boldsymbol{\alpha} = [1,\ x_1,\ x_2];\ \mathbf{A} = \begin{pmatrix} A_0 \\ A_1 \\ A_2 \end{pmatrix} \tag{5}$$

Focalizzando ora l'attenzione sul singolo triangolo, come riportato in Figura 2, siano (x_{a1}, x_{a2}), (x_{b1}, x_{b2}) ed (x_{c1}, x_{c2}) le coordinate dei suoi tre vertici, sicche' dalla (4) potremo dedurre una relazione tra le incognite lagrangiane e le incognite nodali:

$$\begin{aligned} \phi_A &= \phi\,(x_{a1},\ x_{a2}) = A_0 + A_1\, x_{a1} + A_2\, x_{a2} \\ \phi_B &= \phi\,(x_{b1},\ x_{b2}) = A_0 + A_1\, x_{b1} + A_2\, x_{b2} \\ \phi_C &= \phi\,(x_{c1},\ x_{c2}) = A_0 + A_1\, x_{c1} + A_2\, x_{c2} \end{aligned} \tag{6}$$

o matricialmente:

$$\mathbf{d} = \mathbf{CA} \tag{7}$$

dove si e' definito il vettore **d** delle incognite nodali, e la matrice di correlazione **C**:

$$\mathbf{d} = \begin{pmatrix} \phi_A \\ \phi_B \\ \phi_C \end{pmatrix}; \quad \mathbf{C} = \begin{pmatrix} 1 & x_{a1} & x_{a2} \\ 1 & x_{b1} & x_{b2} \\ 1 & x_{c1} & x_{c2} \end{pmatrix} \tag{8}$$

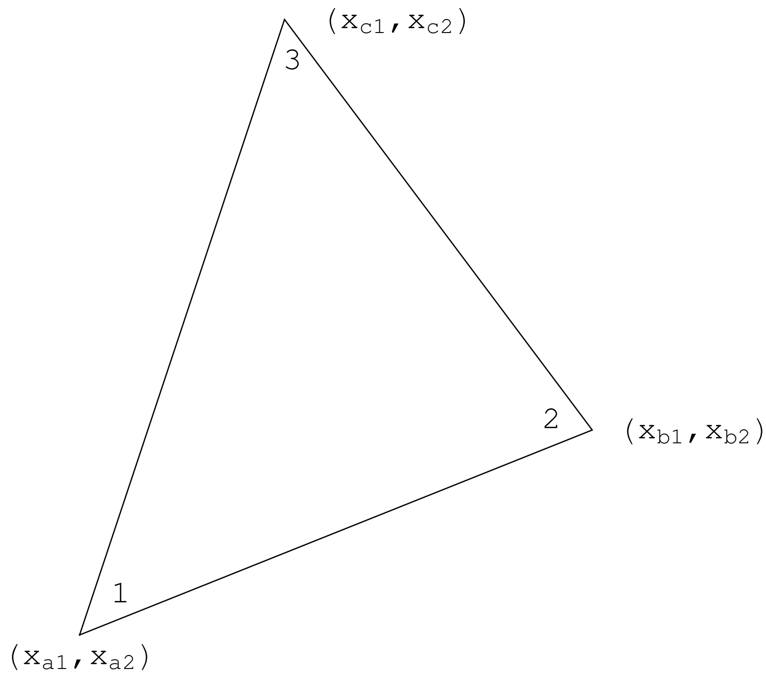

Figura 2 - Un elemento finito triangolare

Dalla (7) puo' ricavarsi $\mathbf{A} = \mathbf{C}^{-1}\,\mathbf{d}$, e dalla (4) si ottiene la funzione ϕ in termini di incognite nodali:

$$\phi = \boldsymbol{\alpha}\,\mathbf{A} = \boldsymbol{\alpha}\,\mathbf{C}^{-1}\,\mathbf{d} = \mathbf{Nd} \tag{9}$$

Il vettore riga $\mathbf{N} = \boldsymbol{\alpha}\,\mathbf{C}^{-1}$ gioca un ruolo fondamentale nella teoria degli elementi finiti, ed e' noto come vettore delle *funzioni di forma.*

L'energia potenziale totale puo' scriversi come somma dei contributi dovuti ai singoli triangoli, sicche' per il generico triangolo si avra', utilizzando la (9):

$$E_t^{(el)} = \frac{\mathbf{d}^T}{2}\left(\int_\Delta\left(\left(\frac{\partial \mathbf{N}}{\partial x_1}\right)^T\left(\frac{\partial \mathbf{N}}{\partial x_1}\right) + \left(\frac{\partial \mathbf{N}}{\partial x_2}\right)^T\left(\frac{\partial \mathbf{N}}{\partial x_2}\right)\right) d\Delta\right)\mathbf{d} - 2\,G\,\theta'\left(\int_\Delta \mathbf{N}\, d\Delta\right)\mathbf{d} \tag{10}$$

Per definizione, si puo' porre:

$$E_t^{(el)} = \frac{1}{2}\mathbf{d}^T \mathbf{k}_e \mathbf{d} - \mathbf{f}_e^T \mathbf{d} \tag{11}$$

dove la *matrice di rigidezza elementare* $\mathbf{k}_e$ e' fornita da:

$$\mathbf{k}_e = \int_\Delta \left(\left(\frac{\partial \mathbf{N}}{\partial x_1}\right)^T \left(\frac{\partial \mathbf{N}}{\partial x_1}\right) + \left(\frac{\partial \mathbf{N}}{\partial x_2}\right)^T \left(\frac{\partial \mathbf{N}}{\partial x_2}\right)\right) d\Delta \tag{12}$$

ed *il vettore dei termini noti* e' dato da:

$$\mathbf{f}_e^T = 2\,G\,\Theta' \int_\Delta \mathbf{N}\, d\Delta \tag{13}$$

Una formulazione piu' esplicita e' possibile, nel caso in esame di interpolazione lineare, perche' e' immediato realizzare che le derivate delle funzioni di forma sono costanti, e quindi possono essere estratte dall'integrale. Ne segue che la matrice di rigidezza puo' scriversi, se Δ e' l'area del triangolo:

$$\mathbf{k}_e = \Delta \begin{pmatrix} N_{1,x_1}^2 + N_{1,x_2}^2 & N_{1,x_1} N_{2,x_1} + N_{1,x_2} N_{2,x2} & N_{1,x_1} N_{3,x_1} + N_{1,x_2} N_{3,x2} \\ N_{1,x_1} N_{2,x_1} + N_{1,x_2} N_{2,x2} & N_{2,x_1}^2 + N_{2,x_2}^2 & N_{2,x_1} N_{3,x_1} + N_{2,x_2} N_{3,x2} \\ N_{1,x_1} N_{3,x_1} + N_{1,x_2} N_{3,x2} & N_{2,x_1} N_{3,x_1} + N_{2,x_2} N_{3,x2} & N_{3,x_1}^2 + N_{3,x_2}^2 \end{pmatrix} \tag{14}$$

Ipotizzando infine di porre l'origine del riferimento nel baricentro del triangolo, sara' anche possibile scrivere:

$$\mathbf{f}_e = 2\,G\,\Theta'\,\Delta \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix} \tag{15}$$

■ L'assemblaggio

Come si e' detto, la sezione retta Σ e' stata suddivisa in M triangoli, identificando T vertici. In alcuni di essi, ossia nei vertici appartenenti al contorno Γ, la funzione ϕ dovra' annullarsi. Se esistono V vertici di frontiera, interessa conoscere i valori della funzione incognita nei restanti N=T-V vertici, identificando il vettore globale delle incognite, di dimensione N:

$$\mathbf{D}^T = (\phi_1, \phi_2, \ldots, \phi_N) \tag{16}$$

L'energia potenziale totale dell'intera trave potra' scriversi allora come:

$$E_t = L\left(\frac{1}{2}\mathbf{D}^T \mathbf{K}_g \mathbf{D} - \mathbf{F}_g^T \mathbf{D}\right) \tag{17}$$

dove la matrice globale di rigidezza $\mathbf{K}_g$, quadrata di dimensione N, ed il vettore globale dei termini noti $\mathbf{F}_g$, dovranno essere assemblati a partire dalle quantita' locali gia' calcolate. A tal fine, si consideri il generico elemento r-mo, e siano (i,j,k) i suoi tre vertici, supposti per il momento non appartenenti alla frontiera. In tale ipotesi, l'assemblaggio della matrice di rigidezza elementare secondo il seguente schema:

$$\begin{aligned} k_{e11} &\longrightarrow K_{gii} = K_{gii} + k_{e11} \\ k_{e12} &\longrightarrow K_{gij} = K_{gij} + k_{e12} \\ k_{e13} &\longrightarrow K_{gik} = K_{gik} + k_{e13} \end{aligned} \tag{18}$$

$$\begin{aligned} k_{e21} &\longrightarrow K_{gji} = K_{gji} + k_{e21} \\ k_{e22} &\longrightarrow K_{gjj} = K_{gjj} + k_{e22} \\ k_{e23} &\longrightarrow K_{gjk} = K_{gjk} + k_{e23} \end{aligned} \tag{19}$$

$$\begin{aligned} k_{e31} &\longrightarrow K_{gki} = K_{gki} + k_{e31} \\ k_{e32} &\longrightarrow K_{gkj} = K_{gkj} + k_{e32} \\ k_{e33} &\longrightarrow K_{gkk} = K_{gkk} + k_{e33} \end{aligned} \tag{20}$$

$$\begin{aligned} f_{e1} &\longrightarrow F_{gi} = F_{gi} + f_{e1} \\ f_{e2} &\longrightarrow F_{gj} = F_{gj} + f_{e2} \\ f_{e3} &\longrightarrow F_{gk} = F_{gk} + f_{e3} \end{aligned} \tag{21}$$

Se invece un vertice del triangolo, ad esempio il primo, appartiene alla frontiera, e' sufficiente tralasciare l'assemblaggio dei termini della prima riga e della prima colonna della matrice elementare.

■ La soluzione

Ottenute le matrici globali di rigidezza ed il corrispondente vettore dei termini noti, il teorema di minimo dell'energia poitenziale totale porta immediatamente al sistema lineare di N equazione nelle N incognite ϕ_i:

$$\mathbf{K}_g \mathbf{D} = \mathbf{F}_g \tag{22}$$

la cui soluzione puo' formalmente porsi come:

$$\mathbf{D} = \mathbf{K}_g^{-1} \mathbf{F}_g \tag{23}$$

In realta', l'inversione della matrice $\boldsymbol{K_g}$ risulta spesso inutile, preferendosi risolvere il sistema di equazioni con il metodo di Gauss-Jordan, o con metodi iterativi, e comunque sfruttando la particolare e tipica forma a banda della matrice. Di cio' non puo' dirsi oltre, rimandando ai testi specializzati.

L'implementazione numerica per una sezione rettangolare

Si consideri la sezione retta rettangolare di Figura 3, di base $L_x = 6$, ed altezza $L_y = 8$. Al fine di ricoprire tale sezione con una griglia triangolare si suddivida la base in n_x quantita' uguali, e l'altezza in n_y quantita' uguali, identificando quindi $n_x\, n_y$ rettangoli di base $\Delta x = \frac{L_x}{n_x}$ ed altezza $\Delta y = \frac{L_y}{n_y}$. Cio' fatto, si suddivida ciascun rettangolo tracciando una linea dal vertice in alto a sinistra al vertice in basso a destra. Si ottiene in tal modo una griglia di $2\, n_x\, n_y$ triangoli, che verranno numerati come in Figura, ossia da sinistra a destra, e dall'alto in basso. I vertici, in numero di $(n_x+1)(n_y+1)$, possono essere numerati con la stessa logica, giungendo alla discretizzazione di figura.

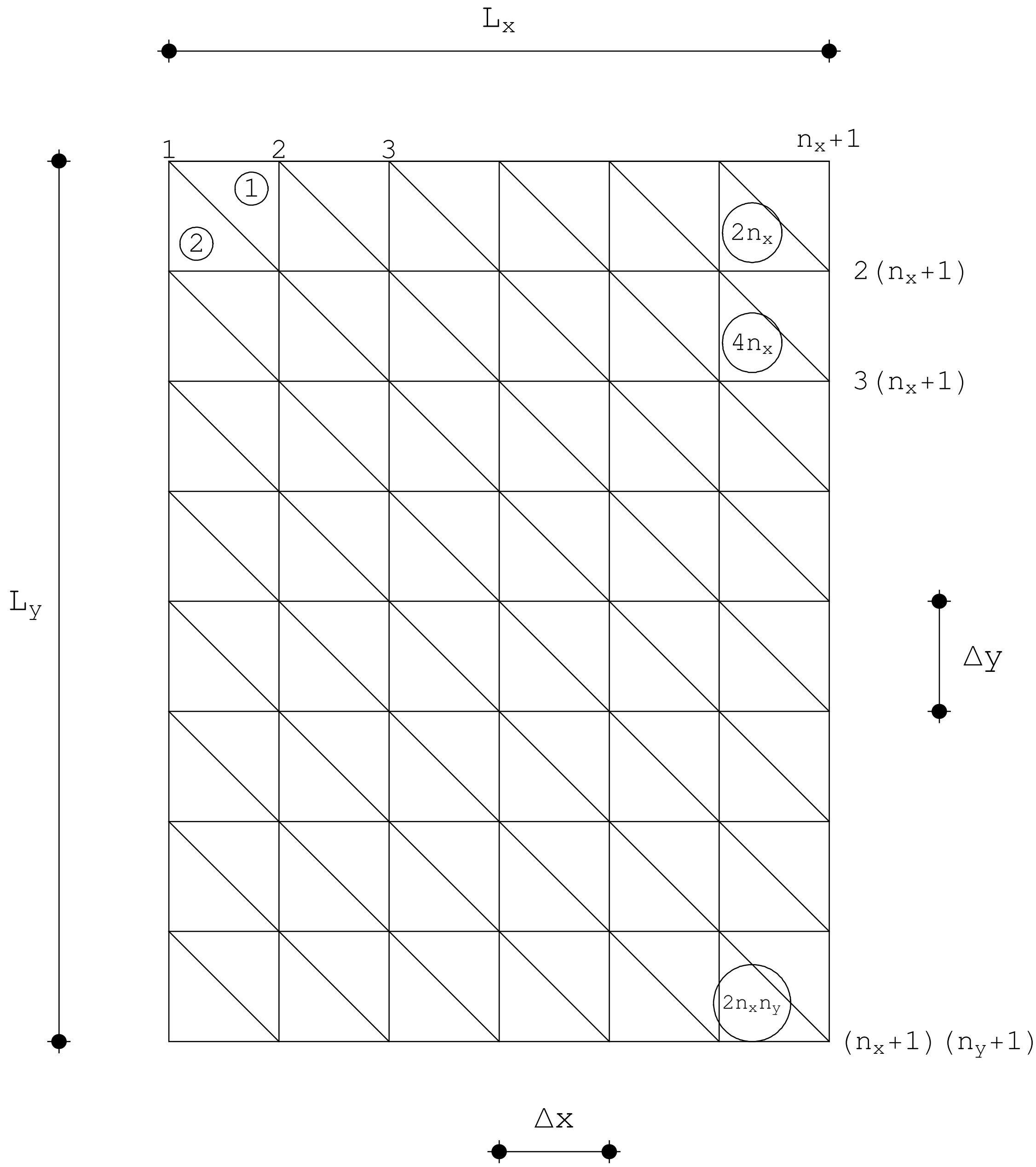

Figura 3 - Una sezione rettangolare suddivisa in $2\, n_x\, n_y$ elementi triangolari

Per descrivere la *geometria* della sezione retta cosi' discretizzata occorre conoscere le coordinate dei nodi. In un riferimento con origine nel nodo in alto a sinistra, e' possibile osservare che le coordinate del generico nodo i-mo, posto all'intersezione dell' orizzontale k-ma e della verticale j-ma sono pari a:

$$\begin{aligned} x_{i1} &= (j-1)\,\Delta x \\ x_{i2} &= -(k-1)\,\Delta y \end{aligned} \tag{24}$$

Queste quantita' possono essere memorizzate in una matrice Coord, con Nnode righe, quante sono i vertici, e 2 colonne.

Per descrivere la *topologia* della struttura e' poi necessario conoscere, per ciascun triangolo, la numerazione dei suoi tre vertici. Mentre e' sempre possibile trarre tali dati per ispezione diretta, riempiendo manualmente una matrice Lnods, con Nelem righe, quante sono i triangoli, e tre colonne, e' ovviamente preferibile dedurre le regole generali che permettono la generazione automatica dei dati. Cio' e' possibile in questo caso a geometria molto semplice, e permette lo studio immediato di differenti livelli di discretizzazione.

Si prendano in esame prima i triangoli segnati in Figura XX con una croce. Il generico triangolo i-mo a distanza j dalla verticale di sinistra avra' vertici forniti da:

$$\begin{aligned}
\mathrm{Lnods}_{i,1} &= \frac{k}{2}\,(n_x+1)+j\\
\mathrm{Lnods}_{i,2} &= \frac{k}{2}\,(n_x+1)+j+n_x+2\\
\mathrm{Lnods}_{i,1} &= \frac{k}{2}\,(n_x+1)+j+1
\end{aligned}\qquad (25)$$

con k variabile da 0 (prima riga) a 2 (n_y-1) (ultima riga) con passo 2. Si noti che in tal modo i vertici del triangolo sono numerati in senso antiorario. I triangoli segnati con un punto possono invece numerarsi con le seguenti formule:

$$\begin{aligned}
\mathrm{Lnods}_{i,1} &= \frac{k}{2}\,(n_x+1)+j\\
\mathrm{Lnods}_{i,2} &= \frac{k}{2}\,(n_x+1)+j+n_x+1\\
\mathrm{Lnods}_{i,1} &= \frac{k}{2}\,(n_x+1)+j+n_x+2
\end{aligned}\qquad (26)$$

Per descrivere infine il contorno della sezione, e quindi per imporre le *condizioni ai limiti*, e' opportuno introdurre la matrice NodFre dei gradi di liberta', costituita da Nnode righe ed una sola colonna. A ciascun nodo di frontiera viene assegnato un valore nullo, agli altri nodi un numero intero variabile da 1 a Ndofn, quantita' positiva che rappresenta il numero di gradi di liberta' del problema, e quindi la dimensione della matrice globale di rigidezza. Anche questa matrice, nel presente esempio, puo' agevolmente riempirsi in modo automatico.

Vediamo infine come puo' costruirsi un semplice programma *Mathematica* per tradurre in pratica quanto finora detto. Si inizia con l'assegnare le quantita' che definiscono la geometria, ossia base ed altezza, e la discretizzazione, ossia il numero di suddivisioni orizzontali n_x e verticali n_y:

```
Lx = 6; Ly = 8; nx = 6; ny = 8;
```

Sono queste le sole quantita' da far variare per poter esaminare altre sezioni a forma rettangolare, ed altri livelli di discretizzazione. Nella cella successiva si definiscono il numero di elementi Nelem, il numero di nodi Nnode, la base e l'altezza dei sub-rettangoli. Poi si inizializzano la matrice Coord, che dovra' descrivere la geometria, la matrice Lnods, che dovra' descrivere la topologia, e la matrice NodFre che dovra' descrivere vincoli e gradi di liberta'. Infine, si definisce anche il vettore α di monomi:

```
Nelem = 2 nx ny; Nnode = (nx + 1) (ny + 1); Δx = Lx / nx; Δy = Ly / ny;
Lnods = Table[0, {i, 1 Nelem}, {j, 1, 3}];
Coord = Table[0, {i, 1, Nnode}, {j, 1, 2}];
NodFre = Table[1, {i, 1, Nnode}];
α = {1, x, y};
```

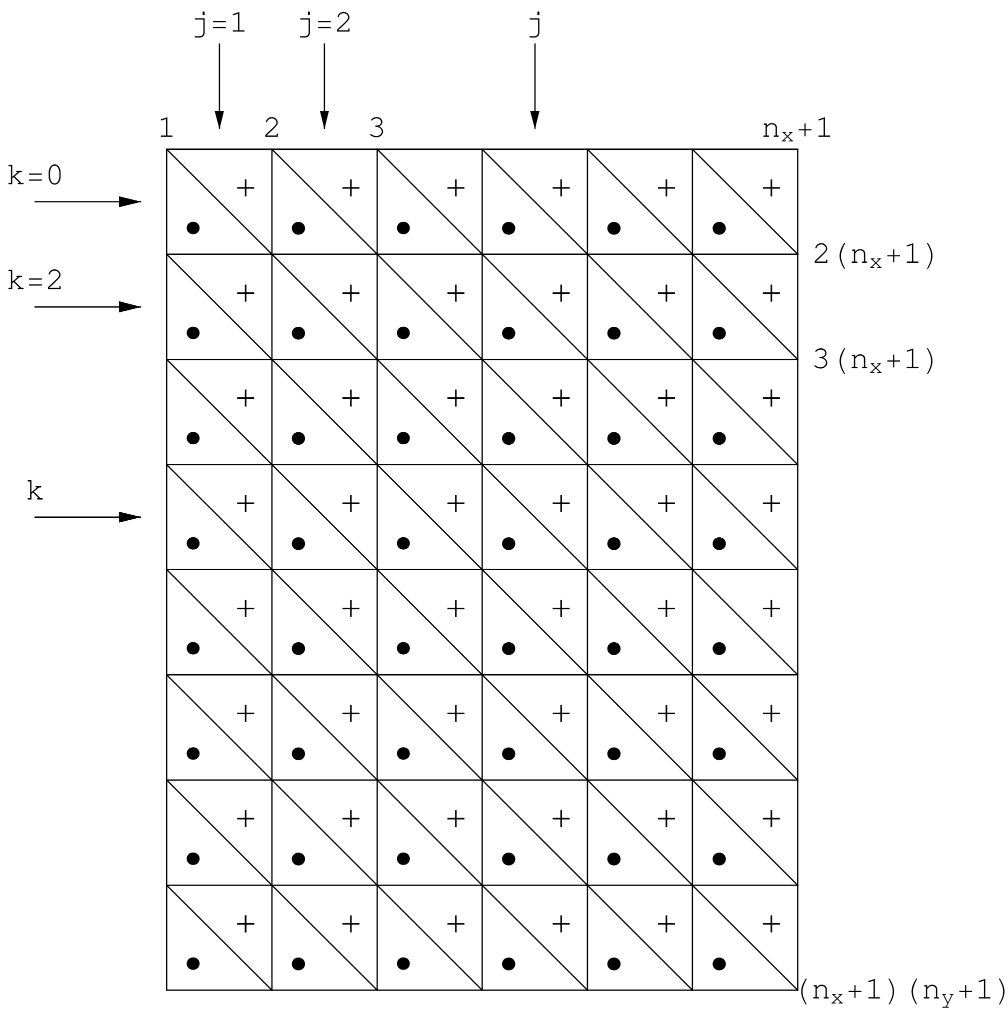

Figura 4 - Lo schema per la numerazione automatica degli elementi e dei nodi

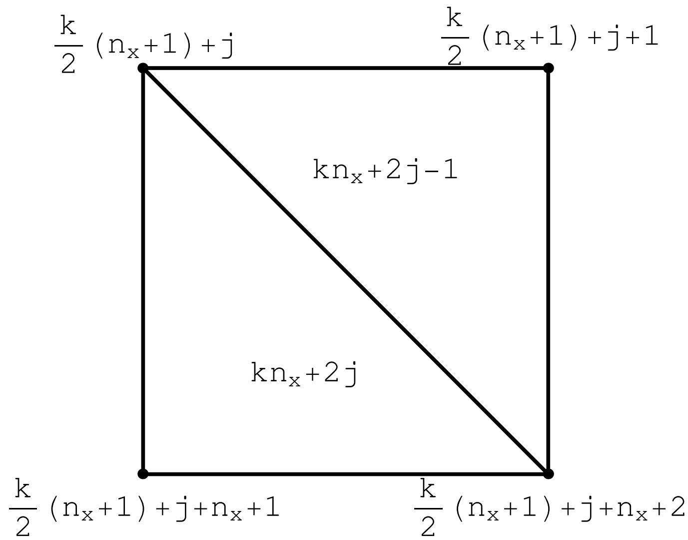

Figura 5 -La generica coppia di elementi finiti triangolari identificati dai numeri j e k

Nelle tre celle successive si generano le matrici Lnods, Coord e NodFre, rispettivamente.

```
Do[
 j = 1;
 Do[
  Lnods[[i, 1]] = k / 2 (nx + 1) + j;
  Lnods[[i, 2]] = k / 2 (nx + 1) + j + nx + 2;
  Lnods[[i, 3]] = k / 2 (nx + 1) + j + 1;
  j = j + 1,
  {i, k nx + 1, (k + 1) nx}];
 j = 1;
 Do[
  Lnods[[i, 1]] = k / 2 (nx + 1) + j;
  Lnods[[i, 2]] = k / 2 (nx + 1) + j + nx + 1;
  Lnods[[i, 3]] = k / 2 (nx + 1) + j + nx + 2;
  j = j + 1,
  {i, (k + 1) nx + 1, (k + 2) nx}],
 {k, 0, 2 (ny - 1), 2}]
```

```
Do[
 j = 1;
 Do[
  Coord[[i, 1]] = (j - 1) Δx;
  Coord[[i, 2]] = - (k - 1) Δy;
  j = j + 1,
  {i, (k - 1) (nx + 1) + 1, k (nx + 1)}],
 {k, 1, ny + 1}]

Do[
  NodFre[[i]] = 0,
  {i, 1, nx + 1}];
Do[
  NodFre[[k (nx + 1)]] = 0;
  NodFre[[k (nx + 1) + 1]] = 0,
  {k, 1, ny}];
Do[
  NodFre[[i]] = 0,
  {i, (nx + 1) ny, (nx + 1) (ny + 1)}];
j = 0;
Do[
 If[NodFre[[i]] == 1, j = j + 1];
 If[NodFre[[i]] == 1, NodFre[[i]] = j],
 {i, 1, Nnode}]
```

Nel caso in esame sara' quindi:

```
Lnods

{{1, 9, 2}, {2, 10, 3}, {3, 11, 4}, {4, 12, 5}, {5, 13, 6}, {6, 14, 7},
 {1, 8, 9}, {2, 9, 10}, {3, 10, 11}, {4, 11, 12}, {5, 12, 13},
 {6, 13, 14}, {8, 16, 9}, {9, 17, 10}, {10, 18, 11}, {11, 19, 12},
 {12, 20, 13}, {13, 21, 14}, {8, 15, 16}, {9, 16, 17}, {10, 17, 18},
 {11, 18, 19}, {12, 19, 20}, {13, 20, 21}, {15, 23, 16}, {16, 24, 17},
 {17, 25, 18}, {18, 26, 19}, {19, 27, 20}, {20, 28, 21}, {15, 22, 23},
 {16, 23, 24}, {17, 24, 25}, {18, 25, 26}, {19, 26, 27}, {20, 27, 28},
 {22, 30, 23}, {23, 31, 24}, {24, 32, 25}, {25, 33, 26}, {26, 34, 27},
 {27, 35, 28}, {22, 29, 30}, {23, 30, 31}, {24, 31, 32}, {25, 32, 33},
 {26, 33, 34}, {27, 34, 35}, {29, 37, 30}, {30, 38, 31}, {31, 39, 32},
 {32, 40, 33}, {33, 41, 34}, {34, 42, 35}, {29, 36, 37}, {30, 37, 38},
 {31, 38, 39}, {32, 39, 40}, {33, 40, 41}, {34, 41, 42}, {36, 44, 37},
 {37, 45, 38}, {38, 46, 39}, {39, 47, 40}, {40, 48, 41}, {41, 49, 42},
 {36, 43, 44}, {37, 44, 45}, {38, 45, 46}, {39, 46, 47}, {40, 47, 48},
 {41, 48, 49}, {43, 51, 44}, {44, 52, 45}, {45, 53, 46}, {46, 54, 47},
 {47, 55, 48}, {48, 56, 49}, {43, 50, 51}, {44, 51, 52}, {45, 52, 53},
 {46, 53, 54}, {47, 54, 55}, {48, 55, 56}, {50, 58, 51}, {51, 59, 52},
 {52, 60, 53}, {53, 61, 54}, {54, 62, 55}, {55, 63, 56}, {50, 57, 58},
 {51, 58, 59}, {52, 59, 60}, {53, 60, 61}, {54, 61, 62}, {55, 62, 63}}
```

```
Coord
```

```
{{0, 0}, {1, 0}, {2, 0}, {3, 0}, {4, 0}, {5, 0}, {6, 0}, {0, -1}, {1, -1},
 {2, -1}, {3, -1}, {4, -1}, {5, -1}, {6, -1}, {0, -2}, {1, -2}, {2, -2},
 {3, -2}, {4, -2}, {5, -2}, {6, -2}, {0, -3}, {1, -3}, {2, -3}, {3, -3},
 {4, -3}, {5, -3}, {6, -3}, {0, -4}, {1, -4}, {2, -4}, {3, -4}, {4, -4},
 {5, -4}, {6, -4}, {0, -5}, {1, -5}, {2, -5}, {3, -5}, {4, -5}, {5, -5},
 {6, -5}, {0, -6}, {1, -6}, {2, -6}, {3, -6}, {4, -6}, {5, -6}, {6, -6},
 {0, -7}, {1, -7}, {2, -7}, {3, -7}, {4, -7}, {5, -7}, {6, -7},
 {0, -8}, {1, -8}, {2, -8}, {3, -8}, {4, -8}, {5, -8}, {6, -8}}
```

```
NodFre
```

```
{0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 3, 4, 5, 0, 0, 6, 7, 8, 9, 10, 0, 0, 11, 12,
 13, 14, 15, 0, 0, 16, 17, 18, 19, 20, 0, 0, 21, 22, 23, 24, 25, 0, 0,
 26, 27, 28, 29, 30, 0, 0, 31, 32, 33, 34, 35, 0, 0, 0, 0, 0, 0, 0, 0}
```

Il numero di gradi di liberta' Ndofn del problema in esame coincide con la massima entrata della matrice NodFre, e puo' essere estratto immediatamente. Poi si possono inizializzare le matrici globali di rigidezza kglob, ed il vettore globale dei termini noti fglob:

```
Ndofn = j;
kglob = Table[0, {i, 1, Ndofn}, {j, 1, Ndofn}];
fglob = Table[0, {i, 1, Ndofn}];
```

La cella successiva esamina la struttura discretizzata elemento per elemento (ciclo di Do per Ielem che va da 1 ad Nelem). Per ciascun elemento si estrae dalla matrice Lnods la numerazione dei suoi tre vertici, chiamandoli Node1, Node2 e Node3, poi dalla matrice Coord si estraggono i valori delle coordinate dei tre vertici, chiamandoli (x_a, y_a) , (x_b, y_b) ed (x_c, y_c) rispettivamente. Successivamente si riempie la matrice C di trasferimento, e si calcola l'area del triangolo Δ, si deducono le funzioni di forma, e le si deriva rispetto ad x e rispetto ad y.

La riga successiva calcola la matrice elementare di rigidezza, utilizzando il comando Outer per generare l'integrando. Cio' fatto, si assembla la matrice elementare nella matrice globale e si calcola direttamente l'aliquota di termine noto dovuto al generico elemento:

```
dof = Table[0, {i, 1, 3}];
Do[
 Node1 = Lnods[[Ielem, 1]];
 Node2 = Lnods[[Ielem, 2]]; Node3 = Lnods[[Ielem, 3]];
 xa = Coord[[Node1, 1]]; ya = Coord[[Node1, 2]];
 xb = Coord[[Node2, 1]]; yb = Coord[[Node2, 2]];
 xc = Coord[[Node3, 1]]; yc = Coord[[Node3, 2]];
 c = {{1, xa, ya}, {1, xb, yb}, {1, xc, yc}}; Δ = Det[c] / 2;
 n = α.Inverse[c]; bx = D[n, x]; by = D[n, y];
 kel = Δ (Outer[Times, bx, bx] + Outer[Times, by, by]);
 dof[[1]] = NodFre[[Node1]];
 dof[[2]] = NodFre[[Node2]]; dof[[3]] = NodFre[[Node3]];
 Do[
  Do[
   If[dof[[i]] dof[[j]] != 0, kglob[[dof[[i]], dof[[j]]]] =
     kglob[[dof[[i]], dof[[j]]]] + kel[[i, j]]],
   {i, 1, 3}],
  {j, 1, 3}];
 Do[
  If[dof[[i]]  != 0, fglob[[dof[[i]]]] = fglob[[dof[[i]]]] + 2 Δ / 3],
  {i, 1, 3}],
 {Ielem, 1, Nelem}]
```

— *General::spell1 : Possible spelling error: new symbol name "Ielem" is similar to existing symbol "Nelem". More…*

Infine, si risolve il sistema di equazioni con il comando LinearSolve:

```
ϕ = LinearSolve[N[kglob], fglob]
```

```
{2.0428, 3.04767, 3.35365, 3.04767, 2.0428, 3.12354, 4.79423,
 5.31924, 4.79423, 3.12354, 3.65712, 5.68647, 6.33487, 5.68647,
 3.65712, 3.81848, 5.95967, 6.64727, 5.95967, 3.81848, 3.65712,
 5.68647, 6.33487, 5.68647, 3.65712, 3.12354, 4.79423, 5.31924,
 4.79423, 3.12354, 2.0428, 3.04767, 3.35365, 3.04767, 2.0428}
```

L'incapsulamento in una funzione

Come si e' detto, gli unici parametri che occorre definire volta per volta sono la base e l'altezza del rettangolo, ed il numero delle suddivisioni orizzontali e verticali per fissare la mesh. Sorge quindi l'idea di incapsulare tutta la procedura descritta in un modulo, inaccessibile dall'esterno, in modo da semplificare al massimo l'analisi, e soprattutto in modo da annullare la possibilita' di interferenze tra variabili globali e variabili locali. La risultante funzione, Rectangular1, e' presentata nel seguito. E', come detto, una funzione dei quattro argomenti Lx, Ly, nx ed ny, ed in essa vengono definiti localmente tutte le altre variabili. Infine, la funzione restituisce il vettore ϕ contenente il valore della funzione di Prandtl nei vertici. Il listato della funzione e' il seguente:

```
Rectangular1[Lx_, Ly_, nx_, ny_] :=
 Module[{Nelem, Nnode, Δx,  Δy, Lnods, Coord,
   NodFre, α, i, j, k, Ielem, dof, Node1, Node2, Node3, xa,
   ya, xb, yb, xc, yc, c, n, Δ, bx, by, kel, kglob, fglob},
Nelem = 2 nx ny;
  Nnode = (nx + 1) (ny + 1); Δx = Lx / nx; Δy = Ly / ny;
  Lnods = Table[0, {i, 1 Nelem}, {j, 1, 3}];
  Coord = Table[0, {i, 1, Nnode}, {j, 1, 2}];
  NodFre = Table[1, {i, 1, Nnode}]; α = {1, x, y};
Do[
   j = 1;
   Do[
    Lnods[[i, 1]] = k / 2 (nx + 1)  + j;
    Lnods[[i, 2]] = k / 2 (nx + 1)  + j + nx + 2;
    Lnods[[i, 3]] = k / 2 (nx + 1)  + j + 1;
    j = j + 1,
    {i, k nx  + 1, (k + 1) nx}];
   j = 1;
   Do[
    Lnods[[i, 1]] = k / 2 (nx + 1)  + j;
    Lnods[[i, 2]] = k / 2 (nx + 1)  + j + nx + 1;
    Lnods[[i, 3]] = k / 2 (nx + 1)  + j + nx + 2;
    j = j + 1,
    {i, (k + 1) nx  + 1, (k + 2) nx}],
   {k, 0, 2 (ny - 1), 2}];

Do[
   j = 1;
   Do[
    Coord[[i, 1]] = (j - 1) Δx;
    Coord[[i, 2]] = - (k - 1) Δy;
    j = j + 1,
    {i, (k - 1) (nx + 1) + 1, k (nx + 1) }],
   {k, 1, ny + 1}];
Do[
   NodFre[[i]] = 0,
   {i, 1, nx + 1}];
Do[
   NodFre[[k (nx + 1) ]] = 0;
   NodFre[[k (nx + 1) + 1]] = 0,
   {k, 1, ny}];
Do[
   NodFre[[i]] = 0,
   {i, (nx + 1) ny, (nx + 1) (ny + 1) }];
j = 0;
Do[
   If[NodFre[[i]] == 1, j = j + 1];
   If[NodFre[[i]] == 1, NodFre[[i]] = j],
```

```
    {i, 1, Nnode}];
 Ndofn = j;
 kglob = Table[0, {i, 1, Ndofn}, {j, 1, Ndofn}];
   fglob = Table[0, {i, 1, Ndofn}];
 dof = Table[0, {i, 1, 3}];
 Do[
    Node1 = Lnods[[Ielem, 1]];
    Node2 = Lnods[[Ielem, 2]]; Node3 = Lnods[[Ielem, 3]];
    xa = Coord[[Node1, 1]]; ya = Coord[[Node1, 2]];
    xb = Coord[[Node2, 1]]; yb = Coord[[Node2, 2]];
    xc = Coord[[Node3, 1]]; yc = Coord[[Node3, 2]];
    c = {{1, xa, ya}, {1, xb, yb}, {1, xc, yc}}; Δ = Det[c] / 2;
    n = α.Inverse[c]; bx = D[n, x]; by = D[n, y];
    kel = Δ (Outer[Times, bx, bx] + Outer[Times, by, by]);
    dof[[1]] = NodFre[[Node1]];
    dof[[2]] = NodFre[[Node2]]; dof[[3]] = NodFre[[Node3]];
    Do[
     Do[
      If[dof[[i]] dof[[j]] != 0, kglob[[dof[[i]], dof[[j]]]] =
        kglob[[dof[[i]], dof[[j]]]] + kel[[i, j]]],
      {i, 1, 3}],
     {j, 1, 3}];
    Do[
     If[dof[[i]]  != 0, fglob[[dof[[i]]]] = fglob[[dof[[i]]]] + 2 Δ / 3],
     {i, 1, 3}],
    {Ielem, 1, Nelem}];
 ϕ = LinearSolve[N[kglob], fglob];
   Return[ϕ]
  ]
```

Sara' quindi, per l'esempio precedente, ma con un raddoppio di discretizzazione:

```
Rectangular1[6, 8, 2 6, 2 8]
```

```
{0.733104, 1.21386, 1.53849, 1.75029, 1.87061, 1.90969, 1.87061, 1.75029,
 1.53849, 1.21386, 0.733104, 1.21855, 2.08386, 2.68982, 3.09204,
 3.32245, 3.39754, 3.32245, 3.09204, 2.68982, 2.08386, 1.21855,
 1.55724, 2.71321, 3.54487, 4.10563, 4.4296, 4.53558, 4.4296,
 4.10563, 3.54487, 2.71321, 1.55724, 1.79721, 3.16688, 4.17081,
 4.85601, 5.25473, 5.3856, 5.25473, 4.85601, 4.17081, 3.16688,
 1.79721, 1.9647, 3.48631, 4.61547, 5.39287, 5.8477, 5.99738, 5.8477,
 5.39287, 4.61547, 3.48631, 1.9647, 2.07529, 3.69817, 4.9119,
 5.75231, 6.24581, 6.40851, 6.24581, 5.75231, 4.9119, 3.69817,
 2.07529, 2.13831, 3.81916, 5.08166, 5.95865, 6.47473, 6.64505,
 6.47473, 5.95865, 5.08166, 3.81916, 2.13831, 2.15878, 3.85851,
 5.13694, 6.02591, 6.5494, 6.72222, 6.5494, 6.02591, 5.13694,
 3.85851, 2.15878, 2.13831, 3.81916, 5.08166, 5.95865, 6.47473,
 6.64505, 6.47473, 5.95865, 5.08166, 3.81916, 2.13831, 2.07529,
 3.69817, 4.9119, 5.75231, 6.24581, 6.40851, 6.24581, 5.75231,
 4.9119, 3.69817, 2.07529, 1.9647, 3.48631, 4.61547, 5.39287, 5.8477,
 5.99738, 5.8477, 5.39287, 4.61547, 3.48631, 1.9647, 1.79721, 3.16688,
 4.17081, 4.85601, 5.25473, 5.3856, 5.25473, 4.85601, 4.17081,
 3.16688, 1.79721, 1.55724, 2.71321, 3.54487, 4.10563, 4.4296,
 4.53558, 4.4296, 4.10563, 3.54487, 2.71321, 1.55724, 1.21855,
 2.08386, 2.68982, 3.09204, 3.32245, 3.39754, 3.32245, 3.09204,
 2.68982, 2.08386, 1.21855, 0.733104, 1.21386, 1.53849, 1.75029,
 1.87061, 1.90969, 1.87061, 1.75029, 1.53849, 1.21386, 0.733104}
```

Per un quadrato a lato unitario, si avra' invece:

```
Rectangular1[1, 1, 10, 10]
```

```
{0.0256262, 0.0412524, 0.0507943, 0.0559989, 0.0576566, 0.0559989,
 0.0507943, 0.0412524, 0.0256262, 0.0412524, 0.0685891, 0.0859258,
 0.0955446, 0.0986286, 0.0955446, 0.0859258, 0.0685891, 0.0412524,
 0.0507943, 0.0859258, 0.108775, 0.121625, 0.125769, 0.121625, 0.108775,
 0.0859258, 0.0507943, 0.0559989, 0.0955446, 0.121625, 0.136411,
 0.141197, 0.136411, 0.121625, 0.0955446, 0.0559989, 0.0576566,
 0.0986286, 0.125769, 0.141197, 0.146197, 0.141197, 0.125769, 0.0986286,
 0.0576566, 0.0559989, 0.0955446, 0.121625, 0.136411, 0.141197,
 0.136411, 0.121625, 0.0955446, 0.0559989, 0.0507943, 0.0859258,
 0.108775, 0.121625, 0.125769, 0.121625, 0.108775, 0.0859258, 0.0507943,
 0.0412524, 0.0685891, 0.0859258, 0.0955446, 0.0986286, 0.0955446,
 0.0859258, 0.0685891, 0.0412524, 0.0256262, 0.0412524, 0.0507943,
 0.0559989, 0.0576566, 0.0559989, 0.0507943, 0.0412524, 0.0256262}
```

L'utilizzo dell'elemento finito rettangolare

La discretizzazione in triangoli e' largamente diffusa, sia per la semplicita' d'uso, sia soprattutto perche' e' facile ricoprire geometrie complesse con un reticolo di triangoli. Tuttavia, e' possibile anche discretizzare la sezione retta utilizzando elementi finiti di forma rettangolare, come illustrato in Figura 6. Per il generico rettangolo, la funzione di torsione potra' assumersi variabile secondo la legge:

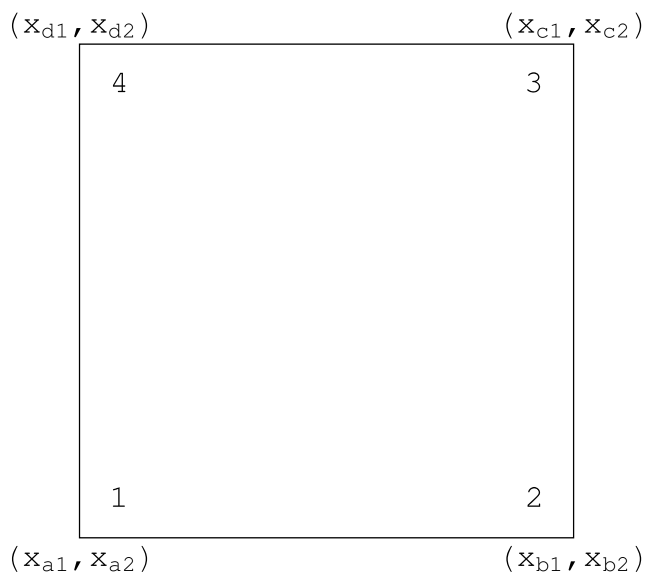

Figura 6 - Un elemento finito rettangolare

$$\phi = \phi\,(x_1,\ x_2) = A_0 + A_1\, x_1 + A_2\, x_2 + A_3\, x_1\, x_2 = \boldsymbol{\alpha}\,\mathbf{A} \qquad (27)$$

dove questa volta α e' un vettore riga di dimensione quattro, cosi' come il vettore colonna $\mathbf{A}$:

$$\boldsymbol{\alpha} = [1,\ x_1,\ x_2,\ x_1\, x_2];\ \mathbf{A} = \begin{pmatrix} A_0 \\ A_1 \\ A_2 \\ A_3 \end{pmatrix} \qquad (28)$$

Se (x_{a1}, x_{a2}), (x_{b1}, x_{b2}), (x_{c1}, x_{c2}) e (x_{d1}, x_{d2}) sono le coordinate dei suoi quattro vertici, dalla (27) potremo dedurre una relazione tra le incognite lagrangiane e le incognite nodali:

$$\begin{aligned} \phi_A &= \phi\,(x_{a1},\ x_{a2}) = A_0 + A_1\, x_{a1} + A_2\, x_{a2} + A_3\, x_{a1}\, x_{a2} \\ \phi_B &= \phi\,(x_{b1},\ x_{b2}) = A_0 + A_1\, x_{b1} + A_2\, x_{b2} + A_3\, x_{b1}\, x_{b2} \\ \phi_C &= \phi\,(x_{c1},\ x_{c2}) = A_0 + A_1\, x_{c1} + A_2\, x_{c2} + A_3\, x_{c1}\, x_{c2} \\ \phi_D &= \phi\,(x_{d1},\ x_{d2}) = A_0 + A_1\, x_{d1} + A_2\, x_{d2} + A_3\, x_{d1}\, x_{d2} \end{aligned} \qquad (29)$$

o matricialmente:

$$\mathbf{d} = \mathbf{CA} \tag{30}$$

dove si e' definito il vettore **d** delle incognite nodali, e la matrice di correlazione **C**:

$$\mathbf{d} = \begin{pmatrix} \phi_A \\ \phi_B \\ \phi_C \\ \phi_D \end{pmatrix}; \ \mathbf{C} = \begin{pmatrix} 1 & x_{a1} & x_{a2} & x_{a1}\,x_{a2} \\ 1 & x_{b1} & x_{b2} & x_{b1}\,x_{b2} \\ 1 & x_{c1} & x_{c2} & x_{c1}\,x_{c2} \\ 1 & x_{d1} & x_{d2} & x_{d1}\,x_{d2} \end{pmatrix} \tag{31}$$

Dalla (30) puo' ricavarsi $\mathbf{A} = \mathbf{C}^{-1}\,\mathbf{d}$, e dalla (27) si ottiene la funzione ϕ in termini di incognite nodali:

$$\phi = \alpha\,\mathbf{A} = \alpha\,\mathbf{C}^{-1}\,\mathbf{d} = \mathbf{Nd} \tag{32}$$

Il resto della procedura e' analogo a quella per elementi finiti triangolari. L'unica differenza sostanziale si riscontra nel calcolo della matric elementare di rigidezza. Per essa infatti la formula (12) rimane valida, ma stavolta l'integrando non e' costante, e pertanto l'integrale deve essere effettivamente calcolato. La seguente funzione permette il calcolo della funzione di torsione per una sezione retta rettangolare, discretizzata con elementi finiti rettangolari:

```
Rectangular2[Lx_, Ly_, nx_, ny_] :=
 Module[{Nelem, Nnode, Δx,  Δy, Lnods, Coord,
   NodFre, α, i, j, k, Ielem, dof, Node1, Node2, Node3, xa,
   ya, xb, yb, xc, yc, c, n, Δ, bx, by, kel, kglob, fglob},
Nelem = nx ny; Nnode = (nx + 1) (ny + 1);
  Lnods = Table[0, {i, 1, Nelem}, {j, 1, 4}];
  Coord = Table[0, {i, 1, Nnode}, {j, 1, 2}];
NodFre = Table[1, {i, 1, Nnode}];
Δx = Lx / nx; Δy = Ly / ny; α = {1, x, y, x y};
j = 1;
Do[
   Do[
    Lnods[[j, 1]] = j + k - 1;
    Lnods[[j, 2]] = j + (k - 1) + (nx + 1);
    Lnods[[j, 3]] = j + k + (nx + 1);
    Lnods[[j, 4]] = j + k;
    j = j + 1,
    {i, 1, nx}],
   {k, 1, ny}];
Do[
   j = 1;
   Do[
    Coord[[i, 1]] = (j - 1) Δx;
    Coord[[i, 2]] = - (k - 1) Δy;
    j = j + 1,
    {i, (k - 1) (nx + 1) + 1, k (nx + 1)}],
   {k, 1, ny + 1}];
Do[
   NodFre[[i]] = 0,
   {i, 1, nx + 1}];
```

```
Do[
   NodFre[[k (nx + 1)]] = 0;
   NodFre[[k (nx + 1) + 1]] = 0,
   {k, 1, ny}];
Do[
   NodFre[[i]] = 0,
   {i, (nx + 1) ny, (nx + 1) (ny + 1)}];
j = 0;
Do[
   If[NodFre[[i]] == 1, j = j + 1];
   If[NodFre[[i]] == 1, NodFre[[i]] = j],
   {i, 1, Nnode}];
Ndofn = j; dof = {0, 0, 0, 0};
kglob = Table[0, {i, 1, Ndofn}, {j, 1, Ndofn}];
  fglob = Table[0, {i, 1, Ndofn}];
Do[
   Node1 = Lnods[[Ielem, 1]]; Node2 = Lnods[[Ielem, 2]];
   Node3 = Lnods[[Ielem, 3]]; Node4 = Lnods[[Ielem, 4]];
   xa = Coord[[Node1, 1]]; ya = Coord[[Node1, 2]];
   xb = Coord[[Node2, 1]]; yb = Coord[[Node2, 2]];
   xc = Coord[[Node3, 1]]; yc = Coord[[Node3, 2]];
   xd = Coord[[Node4, 1]]; yd = Coord[[Node4, 2]];
   c = {{1, xa, ya, xa ya},
     {1, xb, yb, xb yb}, {1, xc, yc, xc yc}, {1, xd, yd, xd yd}};
   n = α.Inverse[c]; bx = D[n, x]; by = D[n, y];
   k = Integrate[Outer[Times, bx, bx] + Outer[Times, by, by],
     {x, xa, xc}, {y, yb, yd}];
   floc = Integrate[n, {x, xa, xc}, {y, yb, yd}];
   dof[[1]] = NodFre[[Node1]]; dof[[2]] = NodFre[[Node2]];
   dof[[3]] = NodFre[[Node3]]; dof[[4]] = NodFre[[Node4]];
   Do[
    Do[
     If[dof[[i]] dof[[j]] != 0, kglob[[dof[[i]], dof[[j]]]] =
       kglob[[dof[[i]], dof[[j]]]] + k[[i, j]]],
     {j, 1, 4}],
    {i, 1, 4}];
   Do[
    If[dof[[i]] != 0,
     fglob[[dof[[i]]]] = fglob[[dof[[i]]]] + 2 floc[[i]]],
    {i, 1, 4}],
   {Ielem, 1, Nelem}];
  ϕ = LinearSolve[N[kglob], fglob];
  Return[ϕ]
 ]
```

```
Rectangular2[6, 8, 6, 8]
```

```
{2.17235, 3.16944, 3.47173, 3.16944, 2.17235, 3.24293, 4.96646,
 5.50201, 4.96646, 3.24293, 3.76798, 5.86724, 6.53804, 5.86724,
 3.76798, 3.9264, 6.14078, 6.85332, 6.14078, 3.9264, 3.76798,
 5.86724, 6.53804, 5.86724, 3.76798, 3.24293, 4.96646, 5.50201,
 4.96646, 3.24293, 2.17235, 3.16944, 3.47173, 3.16944, 2.17235}
```

Rectangular2[6, 8, 2 6, 2 8]

```
{0.761862, 1.23738, 1.5581, 1.7677, 1.88695, 1.92572, 1.88695, 1.7677,
 1.5581, 1.23738, 0.761862, 1.24202, 2.1155, 2.72153, 3.12271,
 3.35234, 3.42716, 3.35234, 3.12271, 2.72153, 2.1155, 1.24202,
 1.57667, 2.74472, 3.58173, 4.14439, 4.46896, 4.57508, 4.46896,
 4.14439, 3.58173, 2.74472, 1.57667, 1.81421, 3.19695, 4.20913,
 4.89889, 5.2998, 5.43132, 5.2998, 4.89889, 4.20913, 3.19695, 1.81421,
 1.98027, 3.51504, 4.65386, 5.4375, 5.89574, 6.0465, 5.89574, 5.4375,
 4.65386, 3.51504, 1.98027, 2.09002, 3.72593, 4.94994, 5.79751,
 6.29517, 6.45923, 6.29517, 5.79751, 4.94994, 3.72593, 2.09002,
 2.15259, 3.84634, 5.11936, 6.00394, 6.52456, 6.69638, 6.52456,
 6.00394, 5.11936, 3.84634, 2.15259, 2.17292, 3.8855, 5.1745, 6.07119,
 6.59933, 6.77371, 6.59933, 6.07119, 5.1745, 3.8855, 2.17292,
 2.15259, 3.84634, 5.11936, 6.00394, 6.52456, 6.69638, 6.52456,
 6.00394, 5.11936, 3.84634, 2.15259, 2.09002, 3.72593, 4.94994,
 5.79751, 6.29517, 6.45923, 6.29517, 5.79751, 4.94994, 3.72593,
 2.09002, 1.98027, 3.51504, 4.65386, 5.4375, 5.89574, 6.0465,
 5.89574, 5.4375, 4.65386, 3.51504, 1.98027, 1.81421, 3.19695,
 4.20913, 4.89889, 5.2998, 5.43132, 5.2998, 4.89889, 4.20913,
 3.19695, 1.81421, 1.57667, 2.74472, 3.58173, 4.14439, 4.46896,
 4.57508, 4.46896, 4.14439, 3.58173, 2.74472, 1.57667, 1.24202,
 2.1155, 2.72153, 3.12271, 3.35234, 3.42716, 3.35234, 3.12271,
 2.72153, 2.1155, 1.24202, 0.761862, 1.23738, 1.5581, 1.7677,
 1.88695, 1.92572, 1.88695, 1.7677, 1.5581, 1.23738, 0.761862}
```

Rectangular2[6, 8, 3 6, 3 8]

```
{0.39581, 0.664941, 0.866545, 1.02091, 1.13845, 1.22525, 1.28504,
 1.32008, 1.33163, 1.32008, 1.28504, 1.22525, 1.13845, 1.02091,
 0.866545, 0.664941, 0.39581, 0.66586, 1.16901, 1.55564, 1.8552,
 2.08474, 2.25485, 2.37226, 2.44117, 2.4639, 2.44117, 2.37226, 2.25485,
 2.08474, 1.8552, 1.55564, 1.16901, 0.66586, 0.870221, 1.56023,
 2.10617, 2.53541, 2.86709, 3.11412, 3.28516, 3.38573, 3.41892,
 3.38573, 3.28516, 3.11412, 2.86709, 2.53541, 2.10617, 1.56023,
 0.870221, 1.0301, 1.86991, 2.54827, 3.08902, 3.51051, 3.82619,
 4.04553, 4.17477, 4.21746, 4.17477, 4.04553, 3.82619, 3.51051,
 3.08902, 2.54827, 1.86991, 1.0301, 1.15685, 2.11692, 2.90382,
 3.53802, 4.03621, 4.41134, 4.6729, 4.82734, 4.87842, 4.82734, 4.6729,
 4.41134, 4.03621, 3.53802, 2.90382, 2.11692, 1.15685, 1.25749,
 2.31374, 3.18852, 3.8995, 4.4616, 4.88683, 5.18427, 5.36025, 5.4185,
 5.36025, 5.18427, 4.88683, 4.4616, 3.8995, 3.18852, 2.31374, 1.25749,
 1.33673, 2.46903, 3.41386, 4.18663, 4.80067, 5.26696, 5.59401,
```

```
5.78784, 5.85206, 5.78784, 5.59401, 5.26696, 4.80067, 4.18663,
3.41386, 2.46903, 1.33673, 1.39787, 2.58902, 3.58833, 4.40947,
5.06444, 5.56329, 5.91393, 6.12204, 6.19103, 6.12204, 5.91393,
5.56329, 5.06444, 4.40947, 3.58833, 2.58902, 1.39787, 1.44327,
2.67821, 3.71819, 4.57558, 5.26137, 5.78485, 6.1534, 6.37237, 6.445,
6.37237, 6.1534, 5.78485, 5.26137, 4.57558, 3.71819, 2.67821,
1.44327, 1.47461, 2.7398, 3.80794, 4.6905, 5.39775, 5.93843, 6.31951,
6.54611, 6.62129, 6.54611, 6.31951, 5.93843, 5.39775, 4.6905,
3.80794, 2.7398, 1.47461, 1.49299, 2.77594, 3.86061, 4.75799, 5.4779,
6.02873, 6.41723, 6.64834, 6.72504, 6.64834, 6.41723, 6.02873,
5.4779, 4.75799, 3.86061, 2.77594, 1.49299, 1.49905, 2.78785,
3.87798, 4.78025, 5.50433, 6.05852, 6.44948, 6.68208, 6.75928,
6.68208, 6.44948, 6.05852, 5.50433, 4.78025, 3.87798, 2.78785,
1.49905, 1.49299, 2.77594, 3.86061, 4.75799, 5.4779, 6.02873, 6.41723,
6.64834, 6.72504, 6.64834, 6.41723, 6.02873, 5.4779, 4.75799,
3.86061, 2.77594, 1.49299, 1.47461, 2.7398, 3.80794, 4.6905, 5.39775,
5.93843, 6.31951, 6.54611, 6.62129, 6.54611, 6.31951, 5.93843,
5.39775, 4.6905, 3.80794, 2.7398, 1.47461, 1.44327, 2.67821, 3.71819,
4.57558, 5.26137, 5.78485, 6.1534, 6.37237, 6.445, 6.37237, 6.1534,
5.78485, 5.26137, 4.57558, 3.71819, 2.67821, 1.44327, 1.39787,
2.58902, 3.58833, 4.40947, 5.06444, 5.56329, 5.91393, 6.12204,
6.19103, 6.12204, 5.91393, 5.56329, 5.06444, 4.40947, 3.58833,
2.58902, 1.39787, 1.33673, 2.46903, 3.41386, 4.18663, 4.80067,
5.26696, 5.59401, 5.78784, 5.85206, 5.78784, 5.59401, 5.26696,
4.80067, 4.18663, 3.41386, 2.46903, 1.33673, 1.25749, 2.31374,
3.18852, 3.8995, 4.4616, 4.88683, 5.18427, 5.36025, 5.4185, 5.36025,
5.18427, 4.88683, 4.4616, 3.8995, 3.18852, 2.31374, 1.25749, 1.15685,
2.11692, 2.90382, 3.53802, 4.03621, 4.41134, 4.6729, 4.82734,
4.87842, 4.82734, 4.6729, 4.41134, 4.03621, 3.53802, 2.90382,
2.11692, 1.15685, 1.0301, 1.86991, 2.54827, 3.08902, 3.51051,
3.82619, 4.04553, 4.17477, 4.21746, 4.17477, 4.04553, 3.82619,
3.51051, 3.08902, 2.54827, 1.86991, 1.0301, 0.870221, 1.56023,
2.10617, 2.53541, 2.86709, 3.11412, 3.28516, 3.38573, 3.41892,
3.38573, 3.28516, 3.11412, 2.86709, 2.53541, 2.10617, 1.56023,
0.870221, 0.66586, 1.16901, 1.55564, 1.8552, 2.08474, 2.25485,
2.37226, 2.44117, 2.4639, 2.44117, 2.37226, 2.25485, 2.08474,
1.8552, 1.55564, 1.16901, 0.66586, 0.39581, 0.664941, 0.866545,
1.02091, 1.13845, 1.22525, 1.28504, 1.32008, 1.33163, 1.32008,
1.28504, 1.22525, 1.13845, 1.02091, 0.866545, 0.664941, 0.39581}
```

```
Max[ϕ]
```

```
6.75928
```

Grafici