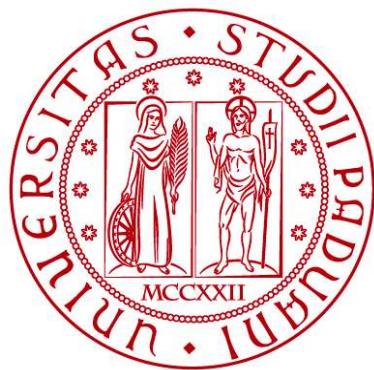


UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI INGEGNERIA CIVILE, EDILE E AMBIENTALE
Department of Civil, Environmental and Architectural Engineering



Corso di Laurea: Ingegneria Civile

Indirizzo: Strutture

MECCANICA COMPUTAZIONALE

Report Mecomp – “CODE”

Docenti: Ing. Mazzucco Giuseppe

Ing. Pomaro Beatrice

Ing. De Marchi Nicolò

Realizzato da: Enrico Viero (1230529)

Massimo Bonato (1227438)

Moscardo Marco(1239111)

Data di Consegna: 07-07-2020

ANNO ACCADEMICO 2019-2020

Sommario

INTRODUZIONE	3
BASI TEORICHE	4
DISCRETIZZAZIONE DELLA STRUTTURA.....	4
TEORIA DI EULERO BERNOULLI	5
MATRICE DI RIGIDEZZA – LOCALE A GLOBALE.....	5
SVILUPPO DEL CODICE	9
TEST	16
PROCEDURA DI CONTROLLO.....	16
STRUTTURA RETICOLARE – TEST SULLA MATRICE DI ROTAZIONE	16
STRUTTURA A TELAIO CON CARICHI ORIZZONTALI	20
TEST SULLE STRUTTURE COMPOSTE (BEAM -TRUSS)	25
PIEDISTALLO	25
PONTE STRALLATO	28
LIBRI DI TESTO CONSULTATI	35

INTRODUZIONE

Con la seguente relazione si intende fornire un report relativo allo sviluppo del codice in ambiente MATLAB per la creazione di un algoritmo di calcolo necessario alla risoluzione di sistemi strutturali tramite Elementi Finiti in ambiente di sviluppo Tridimensionale e tramite l'uso dell'interfaccia GID, per i valori di input.

Verrà posta particolare attenzione all'algoritmo risolutivo necessario al calcolo degli spostamenti e le rotazioni di un elemento BEAM, secondo la teoria di Eulero-Bernoulli e alle operazioni di verifica del corretto funzionamento dello stesso.

BASI TEORICHE

Il metodo numerico agli elementi finiti è largamente diffuso per la risoluzione di problemi di carattere ingegneristico associati a diversi campi di applicazione comprendendo sia problemi di carattere strutturale che di carattere biomolecolare o economico (ex: problemi geotecnici).

Il metodo si basa sulla risoluzione di equazioni differenziali.

Tipicamente, in campo strutturale, vengono ricercate le incognite in campo di spostamento o tensione associate ad una struttura in equilibrio e sottoposta a forze di carattere statico o dinamico.

Nel campo dello studio degli elementi FEM esistono due tipi di approccio distinti per la risoluzione e la determinazione delle incognite: il metodo delle forze e il metodo degli spostamenti.

Il primo, usa le forze interne come incognita del problema attraverso l'imposizione delle equazioni di equilibrio.

Il secondo metodo (stiffness method), assume gli spostamenti e le rotazioni come incognita del problema andando ad imporre la continuità strutturale e la non compenetrazione tra elementi imponendo prima le condizioni al contorno e successivamente ricercando gli spostamenti nodali usando le equazioni di equilibrio, nel seguito si persegirà quest'ultimo metodo per la creazione del codice di calcolo.

DISCRETIZZAZIONE DELLA STRUTTURA

La discretizzazione della struttura avviene dapprima suddividendo la struttura secondo le sue caratteristiche principali e nel caso aumentando il grado di dettaglio andando a migliorare la discretizzazione stessa e quindi il numero di nodi calcolati.

La conseguente scelta dell'elemento da adottare per la discretizzazione deve premettere una rappresentazione ingegneristicamente fedele non solo nella forma geometrica ma anche nel comportamento strutturale associato al differente comportamento dei vari elementi usati per la discretizzazione.

Si riporta ad esempio la rappresentazione di un'asta (trave) soggetta a forze applicate non solo assialmente, se si andasse a rappresentare il comportamento strutturale dell'asta attraverso un elemento TRUSS si otterrebbe una perfetta rappresentazione geometrica di quest'ultima ma non verrebbero rappresentati gli stati tensionali associati ai carichi non assiali, rendendo non realisticamente rilevante l'analisi FEM.

Deve inoltre essere prestata particolare attenzione alla discretizzazione in elementi finiti del corpo, andando a caratterizzare in modo significativo particolari strutturali (es: fori nelle piastre) che possano mettere in luce trasmissioni tensionali particolari.

La discretizzazione deve quindi essere perseguita ricercando la corretta rappresentazione strutturale e dello stato tensionale presupposto ma anche prestando attenzione a non appesantire troppo la Mesh per evitare costi computazionali troppo elevati.

TEORIA DI EULERO BERNOULLI

Vengono utilizzate le equazioni differenziali della linea elastica per passare dalla forma debole alla forma forte (integrale) del problema, secondo le seguenti ipotesi:

- Ipotesi di piccoli spostamenti e piccole deformazioni.
- Le sezioni piane e ortogonali lungo l'asse baricentro si mantengono costanti
- Materiale Omogeneo e Isotropo

MATRICE DI RIGIDEZZA – LOCALE A GLOBALE

Nello spazio tridimensionale, l'elemento Beam associato ai due punti nodali ha 3 spostamenti e 3 rotazioni associate a ciascun nodo.

Le 6 incognite nodali (degree of freedom) portano a 12 incognite per ogni elemento andando a comporre una matrice di rigidezza 12x12.

Attraverso il passaggio da forma debole a forte è possibile definire la matrice di rigidezza andando ad utilizzare il Teorema dei Lavori Virtuali eguagliando lavoro interno ed esterno:

$$\begin{aligned}\int_V \partial \underline{\varepsilon} \partial \underline{\sigma} dV &= \int_V \partial \underline{u} F dV + \int_S \partial \underline{u} f dS \\ \int_V \partial \underline{\varepsilon} \partial \underline{\sigma} dV &= \int_V \partial \underline{u} F dV + \int_S \partial \underline{u} f dS = \sum \partial \underline{u} f^e\end{aligned}$$

Il lavoro virtuale esterno viene espresso come sommatoria di forze, esclusivamente nodali, moltiplicate per i rispettivi spostamenti nodali.

Per solidi composti da più elementi il lavoro interno si esprime come:

$$\sum_{i=1}^n \int_V \partial \underline{\varepsilon} \underline{\sigma} dV$$

Dove il legame costitutivo tensione – deformazione e le relazioni di congruenza deformazioni – spostamenti sono espressi come:

- $\underline{\sigma} = \underline{\underline{D}} \underline{\varepsilon}$
- $\underline{\varepsilon} = \underline{\underline{L}} \underline{u}$
- $\underline{\varepsilon}_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$
- $\underline{\underline{L}}$ è l'operatore di derivazione

Si può relazionare quindi l'incognita del problema, gli spostamenti, con gli spostamenti Nodali attraverso le funzioni di forma associate al' elemento Beam.

Tali funzioni devono rispettare la condizione C^1 di continuità, ovvero lo spostamento e la sua derivata devono essere continui.

$$\begin{aligned}\underline{u} &= \underline{\underline{N}} \underline{a} \\ \underline{\varepsilon} &= \underline{\underline{L}} \underline{u} = \underline{\underline{\underline{L}}} \underline{\underline{N}} \underline{a} = \underline{\underline{B}} \underline{a} \\ \partial \underline{\varepsilon} &= \underline{\underline{B}} \partial \underline{a}\end{aligned}$$

Da cui:

$$\underline{u} = \begin{bmatrix} \underline{\underline{N}}_1 & \dots & \dots & \dots & \underline{\underline{N}}_N \end{bmatrix} \begin{bmatrix} \underline{a}_1 \\ \underline{a}_2 \\ \vdots \\ \underline{a}_N \end{bmatrix}$$

$$\underline{u} = \begin{bmatrix} N_1 & 0 & 0 & : & N_N & 0 & 0 \\ 0 & N_1 & 0 & : & 0 & N_N & 0 \\ 0 & 0 & N_1 & : & 0 & 0 & N_N \end{bmatrix} \begin{bmatrix} a_{1x} \\ a_{1y} \\ a_{1z} \\ \vdots \\ a_{nx} \\ a_{ny} \\ a_{nz} \end{bmatrix}$$

Definiti ora:

$$\begin{aligned}\partial \underline{\varepsilon} &= \underline{\underline{B}} \partial \underline{a} \\ \underline{\sigma} &= \underline{\underline{D}} \underline{\underline{B}} \underline{\varepsilon}\end{aligned}$$

Si ottiene:

$$\left(\int_V \underline{\underline{B}}^T \underline{\underline{D}} \underline{\underline{B}} dV \right) \underline{a} = \underline{f}^e$$

Da cui:

$$\underline{\underline{K}} \underline{a} = \underline{f}^e$$

In coordinate locali, la matrice di rigidezza è data da:

$$[k'] = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 & 0 & 0 & -\frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{GJ}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GJ}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ \hline -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} \\ 0 & 0 & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & -\frac{GJ}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix}$$

Dove:

- E: modulo di Young
- A: Area della Sezione
- L: Lunghezza dell'elemento
- Iz: Inerzia lungo l'asse Z
- Iy: Inerzia lungo l'asse Y
- G: Modulo di Elasticità Tangenziale

- J: Fattore di Rigidezza Torsionale

Si deve quindi procedere a roto-traslare la matrice di rigidezza per allinearla al sistema globale di riferimento, andando a definire una matrice di rotazione legata ai coseni direttori formati tra il sistema di riferimento globale X,Y,Z ed il sistema di riferimento locale x,y,z.

Definendo quindi:

$$T = \begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix}$$

Dove λ è una sottomatrice formata dai coseni direttori tra gli assi globali e locali:

$$\lambda = \begin{bmatrix} C_{xx} & C_{yx} & C_{zx} \\ C_{xy} & C_{yy} & C_{zy} \\ C_{xz} & C_{yz} & C_{zz} \end{bmatrix}$$

e:

- $C_{xi} = \cos\theta_{xi}$ con $i = x, y, z$
- $C_{yi} = \cos\theta_{yi}$ con $i = x, y, z$
- $C_{zi} = \cos\theta_{zi}$ con $i = x, y, z$

Mentre:

$\theta_{xi}, \theta_{yi}, \theta_{zi}$ sono gli angoli misurati rispettivamente tra gli assi globali X,Y,Z e gli assi locali associati al elemento: x,y,z

È quindi possibile ricavare la matrice

$$K_{global} = T^T K' T$$

I due nodi hanno quindi rispettivamente 6 gradi di libertà ciascuno ed è quindi possibile ricavare:

$$F_{global} = K U$$

SVILUPPO DEL CODICE

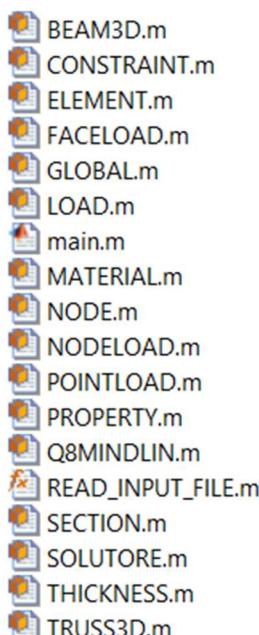
Il codice è stato sviluppato attraverso una programmazione a classi (oggetti) contenenti i vari script di riferimento per la definizione delle varie caratteristiche associate ad ogni elemento finito:

Tramite il MAIN vengono richiamate volta per volta le sottoclassi:

- GLOBAL: contiene delle function utili che possono essere chiamate in ogni punto del codice, in particolare nel calcolo della lunghezza dell'elemento
- NODE: una classe che va a prendersi dal file di INPUT il vettore posizione dei nodi e il loro ID (numero globale)
- ELEMENT: questa classe gestisce tutti i tipi di elementi (TRUSS, BEAM E-B, BEAM T-S, PLATE...), le loro proprietà associate e il loro ID
- LOAD: classe che gestisce i CARICHI NODALI e DISTRIBUITI
- BOUNDARY: dal file di INPUT si va a prendere i le condizioni al contorno e i vincoli
- SOLVER: Il solutore del codice
- PROPERTY: classe che a sua volta gestisce le classi

Attraverso le function get va a prendersi:

- MATERIAL: con proprietà E, v e densità
- SECTION: con proprietà AREA A e inerzie Ix, ly, J e G.



Il file di input, per la definizione della discretizzazione della struttura, viene ricavato tramite software GID.

GID ci consente, attraverso uno script interno, di ricreare la geometria, la posizione dei nodi, il materiale e le caratteristiche intrinseche alla struttura e formattare tali proprietà attraverso un file dati (.dat) che verrà letto dal codice attraverso la function READ_INPUT_FILE.m tramite il main.

```
%-----
function [nodes, ...
    elements, ...
    properties, ...
    boundaries, ...
    endRelease, ...
    loads, ...
    filePath]=READ_INPUT_FILE()

%-----
nodes      = NODE.empty;
elements   = ELEMENT.empty;
properties = PROPERTY.empty;
loads      = LOAD.empty;
boundaries = CONSTRAINT.empty;
endRelease = zeros(1,1);
%tables    = zeros(1,1);
%-----
[stPath, pathName] = uigetfile('*.dat');
stPath = strcat(pathName,stPath);
```

In particolare vengono richiamate le diverse proprietà, e trasformate in forma vettoriale tramite:

```
%----- INPUT
[nodes,elements,properties,boundaries,endRelease,loads, ...
    stFilePath]=READ_INPUT_FILE();

lenPath = length(stFilePath);
if lenPath>4
    st      = strsplt(stFilePath,'.inp');
    stFilePath=st{1};
else
    stFilePath = 'resBEAM2D';
end
%----- END INPUT
```

Classificati gli elementi delle mesh tramite ID identificativo e vengono reindirizzati alla classe ELEMENT per essere associati al corretto script risolutivo (TRUSS – BEAM – Q8MINDLIN)

```

function elementInitialization(this, nodes_id, prop_id)
switch this.type_id
    case 1 %TRUSS
        this.el = TRUSS3D(nodes_id, prop_id);
    case 2 %BEAM EB
        this.el = BEAM3D(nodes_id, prop_id);
    case 3 %Q8 MINDLIN
        this.el = Q8MINDLIN(nodes_id, prop_id);
    case 4 %PLATE
        %under construction

case 'elements'
stLine = fgetl(unitIN);
while (~feof(unitIN))&&(strcmp(stLine,'end')==0)
    e = e +1;
    st = strsplit(stLine);
    st(strcmp(' ',st)) = [];
    dim = size(st,2);
    id = str2double(st(1));
    elemType = str2double(st(2));
    idmat = str2double(st(dim));
    idnodes = str2double(st(3:dim-1));
    elements(e)=ELEMENT(id, elemType, idnodes, idmat);
    stLine = fgetl(unitIN);
end

```

Viene quindi eseguita un'inizializzazione delle proprietà che andranno a caratterizzare la matrice di rigidezza K locale e successivamente K globale

```

function [kglobal]=localstiffnes(this,prop,X)
this.X = X;
x1 = X(1,1);
y1 = X(1,2);
z1 = X(1,3);
x2 = X(2,1);
y2 = X(2,2);
z2 = X(2,3);
E = prop.mat.E;
nu = prop.mat.nu;
G = E / (2 * (1+nu));
A = prop.sec.A;
Iy = prop.sec.I11;
Iz = prop.sec.I22;
J = prop.sec.J;
gl = GLOBAL();
L = gl.distance(X(1,:), X(2,:));

k=zeros(this.DIMDOF^2,this.DIMDOF^2);

```

Successivamente vengono calcolate le posizioni interne alla matrice K locale

```
%-----
k11=E*A/L;
k22=12*E*Iz/L^3;
k33=12*E*Iy/L^3;
k44=G*J/L;
k55=4*E*Iy/L;
k66=4*E*Iz/L;
k26=6*E*Iz/L^2;
k35=6*E*Iy/L^2;
k552=2*E*Iy/L;
k662=2*E*Iz/L;
%-----
```

E viene inizializzata la matrice:

```
ka=[k11 0 0 0 0 0
    0 k22 0 0 0 k26
    0 0 k33 0 -k35 0
    0 0 0 k44 0 0
    0 0 -k35 0 k55 0
    0 k26 0 0 0 k66];
kb=[-k11 0 0 0 0 0
    0 -k22 0 0 0 k26
    0 0 -k33 0 -k35 0
    0 0 0 -k44 0 0
    0 0 k35 0 k552 0
    0 -k26 0 0 0 k662];
kc=[-k11 0 0 0 0 0
    0 -k22 0 0 0 -k26
    0 0 -k33 0 k35 0
    0 0 0 -k44 0 0
    0 0 -k35 0 k552 0
    0 k26 0 0 0 k662];
kd=[k11 0 0 0 0 0
    0 k22 0 0 0 -k26
    0 0 k33 0 k35 0
    0 0 0 k44 0 0
    0 0 k35 0 k55 0
    0 -k26 0 0 0 k66];
%-----
%--k locale del singolo elemento, assemblaggio sottomatrici--
%-----
k=[ka kb
   kc kd];
%
```

La matrice K locale deve ora essere rotata con direzionalità riferita al sistema di riferimento Globale per un suo successivo assemblaggio tramite solutore:

```

CXX = (x2-x1)/L;
CYX = (y2-y1)/L;
CZX = (z2-z1)/L;
D = sqrt(CXX*CXX + CYX*CYX);
CXy = -CYX/D;
CYy = CXX/D;
CZy = 0;
CXz = -CXX*CZX/D;
CYz = -CYX*CZX/D;
CZz = D;
lambda = [CXX CYX CZX ;CXy CYy CZy ;CXz CYz CZz];

end

%-----
%--- assemblo T -----
%-----
```

$$T = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

Possono verificarsi due casistiche in cui la direzionalità al sistema di riferimento Globale sia già garantita o perfettamente invertita.

Nel caso in cui l'asse locale x' coincida con l'asse globale Z allora l'elemento si posiziona in modo parallelo all'asse globale (Z) e l'asse y' risulta nel caso indeterminato. Il problema viene risolto andando ad associare l'asse locale y' come asse di riferimento globale Y.

La matrice di rotazione lambda assumerà quindi una forma equivalente a:

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Nel caso il problema sussistesse per orientamento opposto dell'asse locale x' la matrice di rotazione risulterà pari a:

$$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

All'interno dell'algoritmo il seguente problema viene controllato da uno script che va ad analizzare l'orientamento degli assi e nel caso la matrice di rotazione da andare a considerare distinguendo tra l'assemblaggio della matrice di rotazione T tramite l'ottenimento di lambda da coseni direttori o lambda da check sugli allineamenti degli assi

```

if (x1==x2) && (y1==y2)
    if z2>z1
        lambda=[0 0 1
                 0 1 0
                 -1 0 0];
    else
        lambda=[0 0 -1
                 0 1 0
                 1 0 0];
    end
else

```

Può essere quindi rotata la matrice di rotazione locale per ottenere quella Globale che successivamente verrà assemblata all'interno del solutore:

```

%-----
%-----rotazione k locale a k globale -----
%-----

kglobal= T'*k*T;
end

function write(this)
fprintf('nodes: [%i, %i]; ',this.nodes_id(1),this.nodes_id(2));
fprintf('property ID: %i\n', this.prop_id);

```

Fatto ciò si ritorna nel main con i dati della struttura creata in GiD, nel caso in figura una sorta di tre-piede con un'asta inclinata e incastrato al suolo, per verificare che la rotazione della matrice K fosse corretta anche per aste inclinate:

```

1 - clear;
2 - clc;
3 - clf;
4 - fprintf('-----\n');
5 - fprintf('-----CODE\n');
6 - fprintf('-----MECOMP 2019-20\n');
7 - fprintf('-----\n');
8 - fprintf('-----\n');
9 - fprintf('START\n');
10 -
11 -
12 -
13 - [nodes,elements,properties,boundaries,endRelease,loads, ...
14 -                                         stFilePath]=READ_INPUT_FILE();

```

	boundaries	1x3 CONSTRAINT
	elements	1x3 ELEMENT
	endRelease	0
	loads	1x1 LOAD
	nodes	1x4 NODE
	properties	1x1 PROPERTY

E il MAIN chiama il SOLUTORE:

```
24 %-----  
25 -> sol=SOLUTORE(elements,nodes,properties, loads, boundaries); -----SOLVER  
26 - k=sol.assemblyK();  
27 - [n,n]=size(k);  
28 - k=zeros(n,n)+k;  
29 - [u,f]=sol.solutore();  
30 %----- -----END SOLVER
```

Successivamente assembra la matrice di rigidezza K, passando per ogni elemento attraverso la classe ELEMENT che chiama la function “stiffness” e quindi BEAM3D se del caso.

Passa per GLOBAL anche BEAM3D per trovarsi il valore L che corrisponde alla distanza tra nodi e lunghezza di ogni elemento.

E avviene la costruzione della matrice e la sua rotazione nei modi precedentemente descritti.

Fatto questo il solutore calcola la grandezza della matrice K, nel caso di BEAM3D 12x12 visto i 6 gdl per nodo, e per ognuno dei nodi trova i vettori coordinate per la matrice K locale e globale e assembra.

TEST

Per verificare il corretto funzionamento dell'algoritmo di calcolo e contestualmente eseguire un controllo sulla precisione degli spostamenti e delle rotazioni, sono stati eseguiti dei test di confronto con un programma di calcolo commerciale (MIDAS GEN).

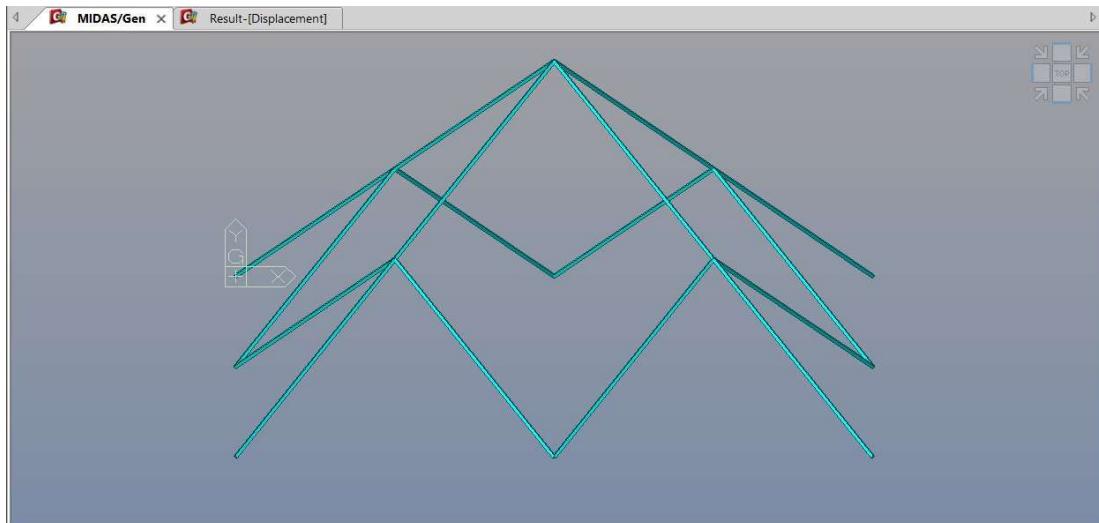
Verranno riportati alcuni degli esempi eseguiti per un telaio a base quadrata di 4 piani con alcune forze orizzontali applicate, una struttura reticolare di forma piramidale per un confronto sul corretto dimensionamento della matrice di rotazione e infine un esempio dove verrà interfacciata la capacità del codice di gestire sia elementi BEAM che elementi TRUSS al suo interno.

PROCEDURA DI CONTROLLO

- Inserimento dei dati in GID e creazione della struttura
- Inserimento dei dati in MIDAS GEN e creazione della struttura
- Controllo del file .dat e delle proprietà inserite
- Controllo del file di analisi MIDAS GEN
- Controllo degli spostamenti massimi
- Controllo delle Rotazioni

STRUTTURA RETICOLARE – TEST SULLA MATRICE DI ROTAZIONE

Successivamente alle basilari operazioni di calcolo di elementi comuni, quali il singolo incastro orientato nello spazio e un portare tridimensionale con alcune forze applicate, per testare il codice ed il comportamento della matrice di rotazione ed il suo corretto assemblaggio nei confronti del solutore si è proceduto al calcolo di alcune strutture più complesse, di seguito elencate.



La base della piramide è inscritta in un quadrato di lato 1000mm con un'altezza massima di 500mm e contiene al suo interno 16 elementi BEAM.

Il lato di ogni singola asta è di 4330,12 e va a formare un angolo con la sua ortogonale di 45°.

Il materiale usato per l'analisi è acciaio S235 con sezione quadrata di lato 50mm.

```

34 end
35 ! ID mat      E          nu          A          I11          I22          J
36 properties
37   1  2.10000e+05  3.00000e-01  2.50000e+03  5.20833e-07  5.20833e-07  8.78906e-07
38 end
  
```

La piramide è stata caricata con un carico nodale puntuale verticale applicato al nodo superiore di forza pari a 100KN.

```

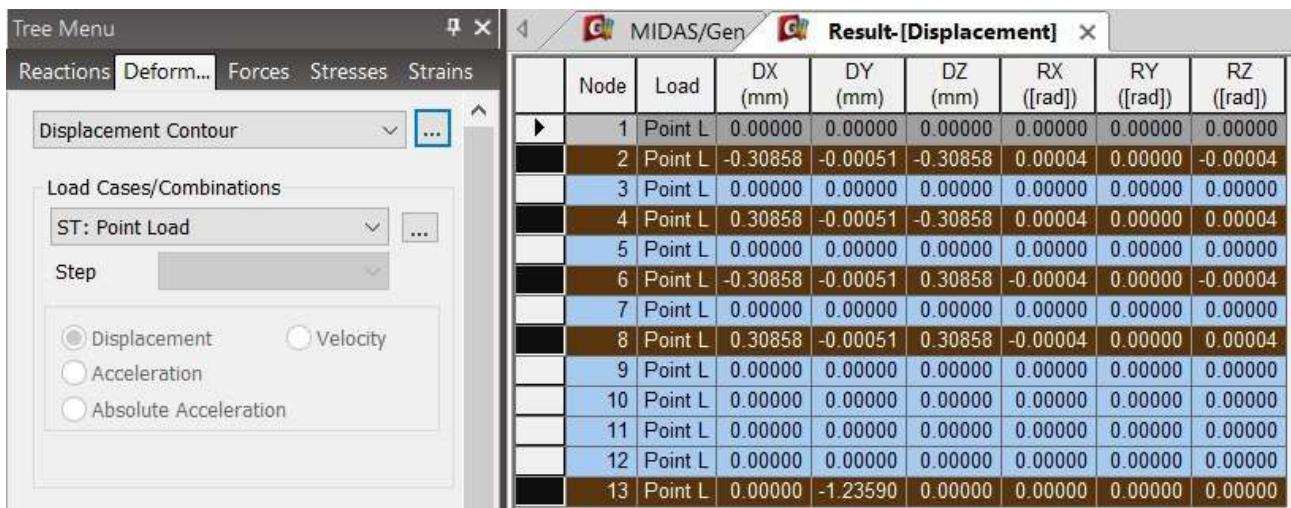
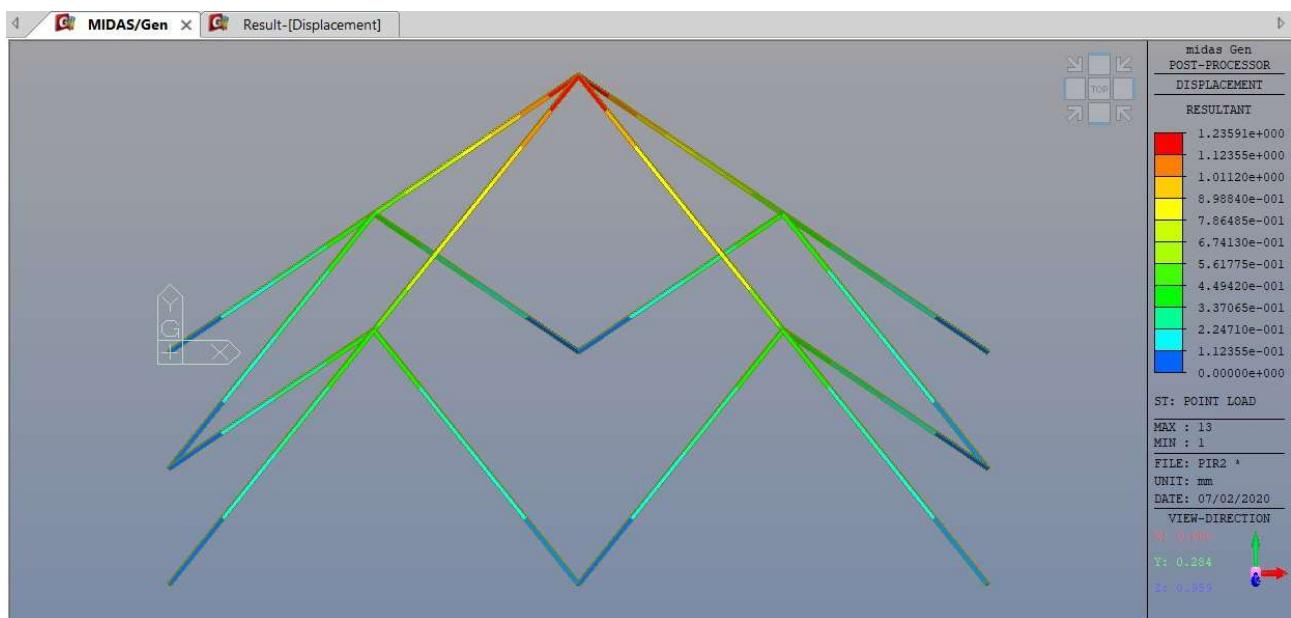
49 loads
50   1  0.0000000e+00  -1.0000000e+05  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  7
51 end
52
  
```

E ne risulta uno spostamento massimo in X e Z: -0.3093 m

E uno spostamento massimo in Y: -1.2372 m

Le rotazioni risultano infinitesimali con valore massimo: 4,29e-5 rad

Viene qui riportata il confronto con MIDAS che ha prodotto risultati concordi a quanto sviluppato col codice Matlab:



Si possono quindi notare l'uguaglianza di spostamenti e rotazioni nelle diverse direzioni con il codice:

The screenshot shows the MATLAB interface with two windows open. On the left is the 'Editor - main.m' window, which contains the following code:

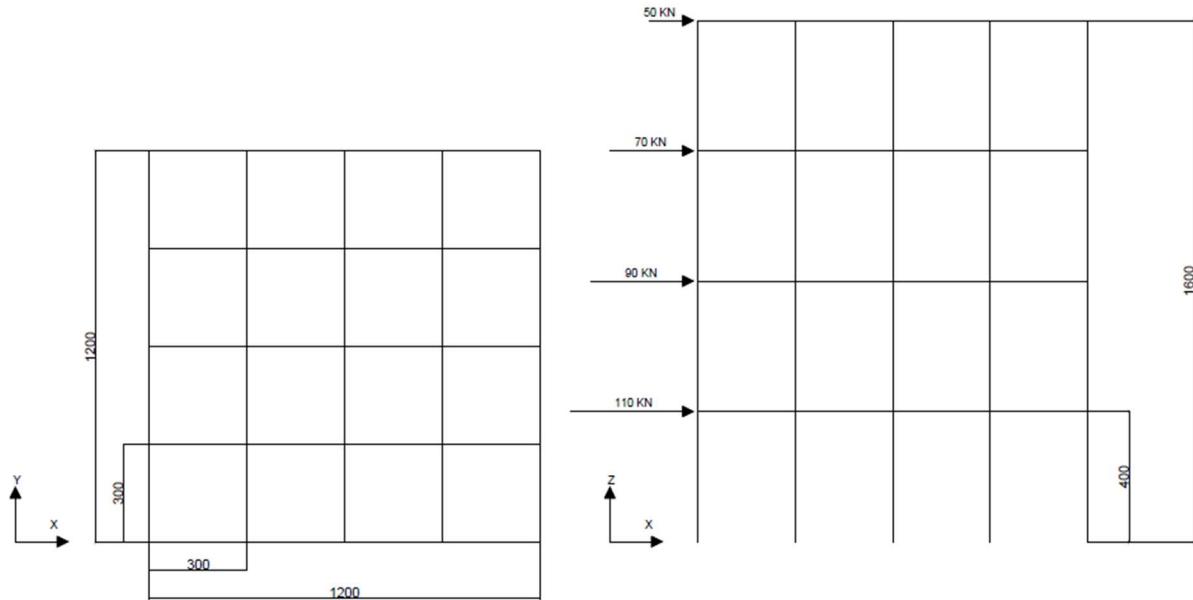
```
19 u = [0; 0; 0; 0; -0.0309; -5.1517e-05; -0.0309; 4.2923e-06; -5.4044e-22; -4.2923e-06; 0.0309; -5.1517e-05; 0.0309; -4.2923e-06; 7.2058e-22; 4.2923e-06; -1.8916e-18; -0.1236; 1.2038e-17; 1.1377e-21; -1.8644e-22];
```

On the right is the 'Variables - u' browser window, which displays the variable 'u' as a 78x1 double matrix. The matrix has 78 rows and 1 column, with the first few elements matching the values defined in the code.

STRUTTURA A TELAIO CON CARICHI ORIZZONTALI

Per verificare il corretto assemblaggio delle proprietà di diverse sezioni e materiali all'interno del codice si è deciso di sviluppare un esempio a telaio, che simuli l'azione di una forza sismica orizzontale applicata con una distribuzione di forze coerenti con i principi dell'analisi statica lineare.

Si è ipotizzata una struttura a telaio, in calcestruzzo armato C28/30 a maglia quadrata con distanza orizzontale tra elementi pari a 3m, si sviluppa in altezza per 4 piani con distanza interpiano di 4m:



Proprietà dei Materiali:

- $E=32308 \text{ N/mm}^2$
- $V=0,2$

Per Travi e Pilastri sono state inserite le seguenti Proprietà all'interno di GID:

P30x30
b = 300 mm
h = 300 mm
A = 90000 mm ²
I ₁₁ = 675000000 mm ⁴
I ₂₂ = 675000000 mm ⁴
J = 1139063000

P40x40
b = 400 mm
h = 400 mm
A = 160000 mm ²
I ₁₁ = 213333333 mm ⁴
I ₂₂ = 213333333 mm ⁴
J = 3600000000

P50x50
b = 500 mm
h = 500 mm
A = 250000 mm ²
I ₁₁ = 5208333333 mm ⁴
I ₂₂ = 5208333333 mm ⁴
J = 8789063000

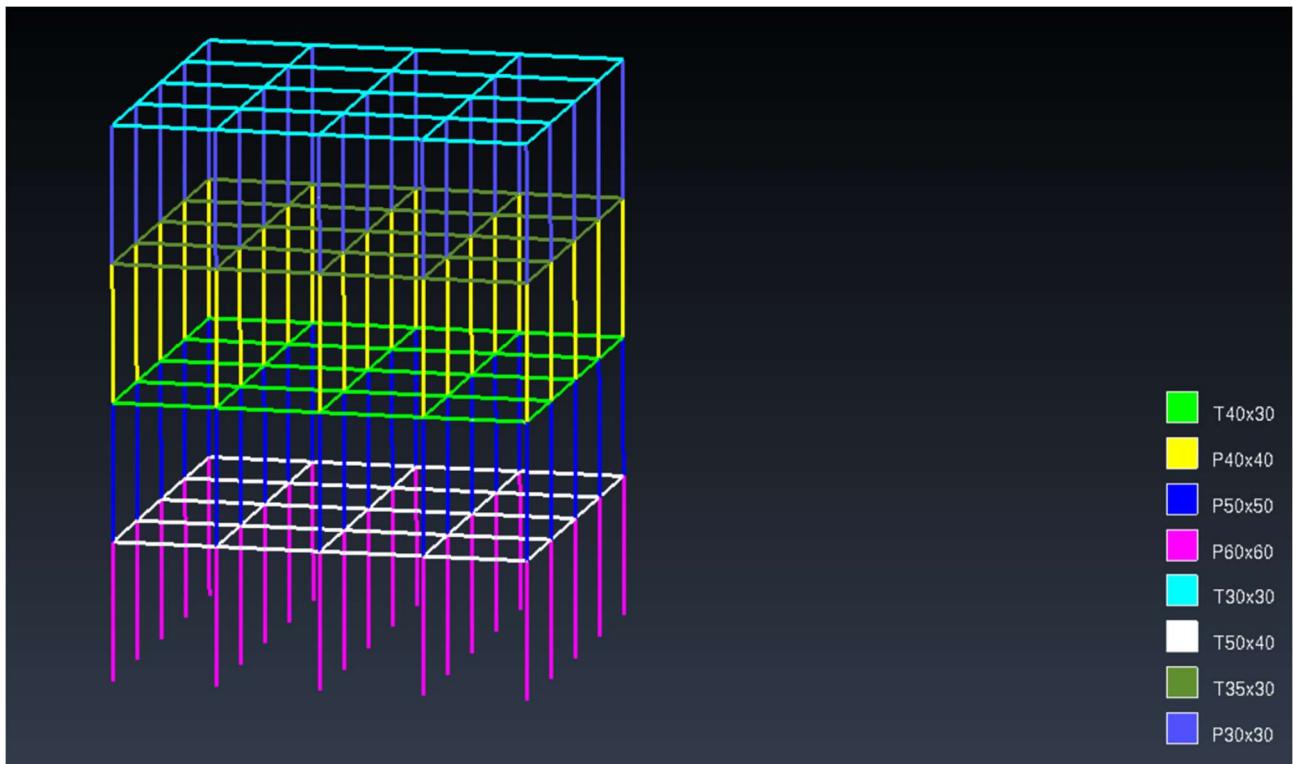
P60x60
b = 600 mm
h = 600 mm
A = 360000 mm ²
I ₁₁ = 10800000000 mm ⁴
I ₂₂ = 10800000000 mm ⁴
J = 18225000000

T30x30
b = 300 mm
h = 300 mm
A = 90000 mm ²
I ₁₁ = 675000000 mm ⁴
I ₂₂ = 675000000 mm ⁴
J = 1139063000

T35x30
b = 300 mm
h = 350 mm
A = 105000 mm ²
I ₁₁ = 1071875000 mm ⁴
I ₂₂ = 787500000 mm ⁴
J = 1525513000

T40x30
b = 300 mm
h = 400 mm
A = 120000 mm ²
I ₁₁ = 1600000000 mm ⁴
I ₂₂ = 900000000 mm ⁴
J = 1943851000

T50x40
b = 400 mm
h = 500 mm
A = 200000 mm ²
I ₁₁ = 4166666667 mm ⁴
I ₂₂ = 2666666667 mm ⁴
J = 5474167000



Controllabili tramite il file .dat:

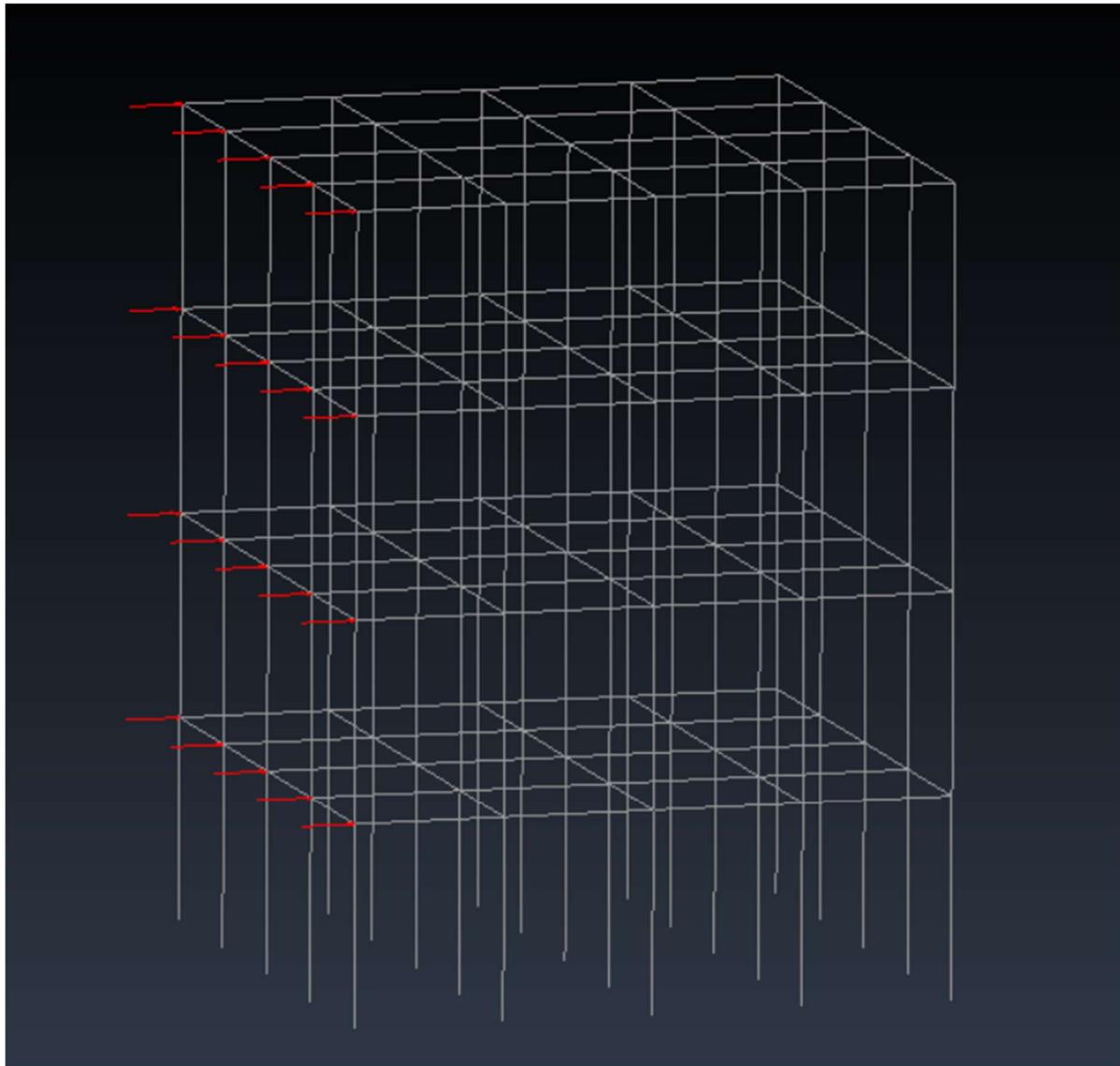
! ID	mat	E	nu	A	I11	I22	J
properties							
1	3.23080e+04	2.00000e-01	1.20000e+05	1.60000e+09	9.00000e+08	1.94385e+09	
2	3.23080e+04	2.00000e-01	1.60000e+05	2.13333e+09	2.13333e+09	3.60000e+09	
3	3.23080e+04	2.00000e-01	2.50000e+05	5.20833e+09	5.20833e+09	8.78906e+09	
4	3.23080e+04	2.00000e-01	3.60000e+05	1.08000e+10	1.08000e+10	1.82250e+10	
5	3.23080e+04	2.00000e-01	9.00000e+04	6.75000e+08	6.75000e+08	1.13906e+09	
6	3.23080e+04	2.00000e-01	2.00000e+05	4.16667e+09	2.66667e+09	5.47417e+09	
7	3.23080e+04	2.00000e-01	1.05000e+05	1.07188e+09	7.87500e+08	1.52551e+09	
8	3.23080e+04	2.00000e-01	9.00000e+04	6.75000e+08	6.75000e+08	1.13906e+09	

La struttura viene incastrata rigidamente alla base e viene simulata una forza sismica agente con direzione X positivo ad ogni impalcato.

Vengono quindi ipotizzate 4 forze parallele ad ogni impalcato e agenti con direzione X positiva.

Viene considerata, data la rigidezza costante di ogni impalcato, una suddivisione paritaria della forza sismica applicata per ogni pilastro secondo la seguente distribuzione:

F ,totale	KN	F, pilastro	KN
F ₁	110	F _{1,i}	22
F ₂	90	F _{2,i}	18
F ₃	70	F _{3,i}	14
F ₄	50	F _{4,i}	10



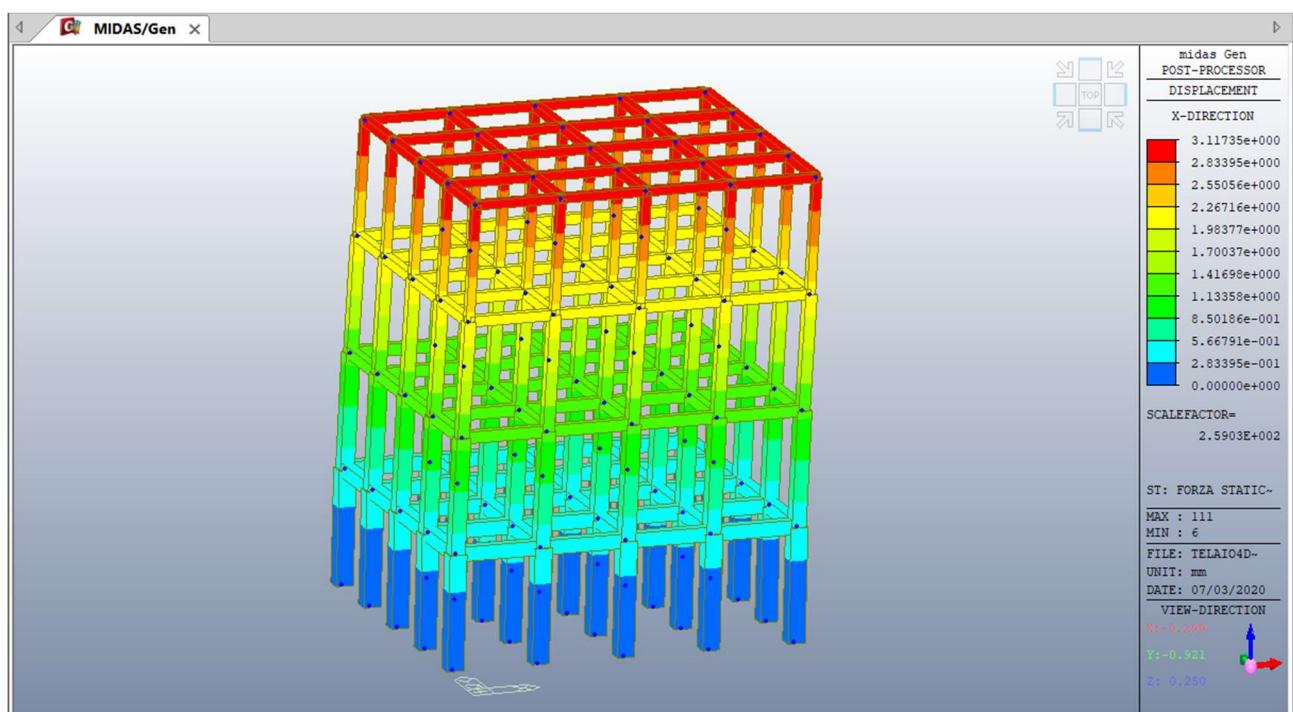
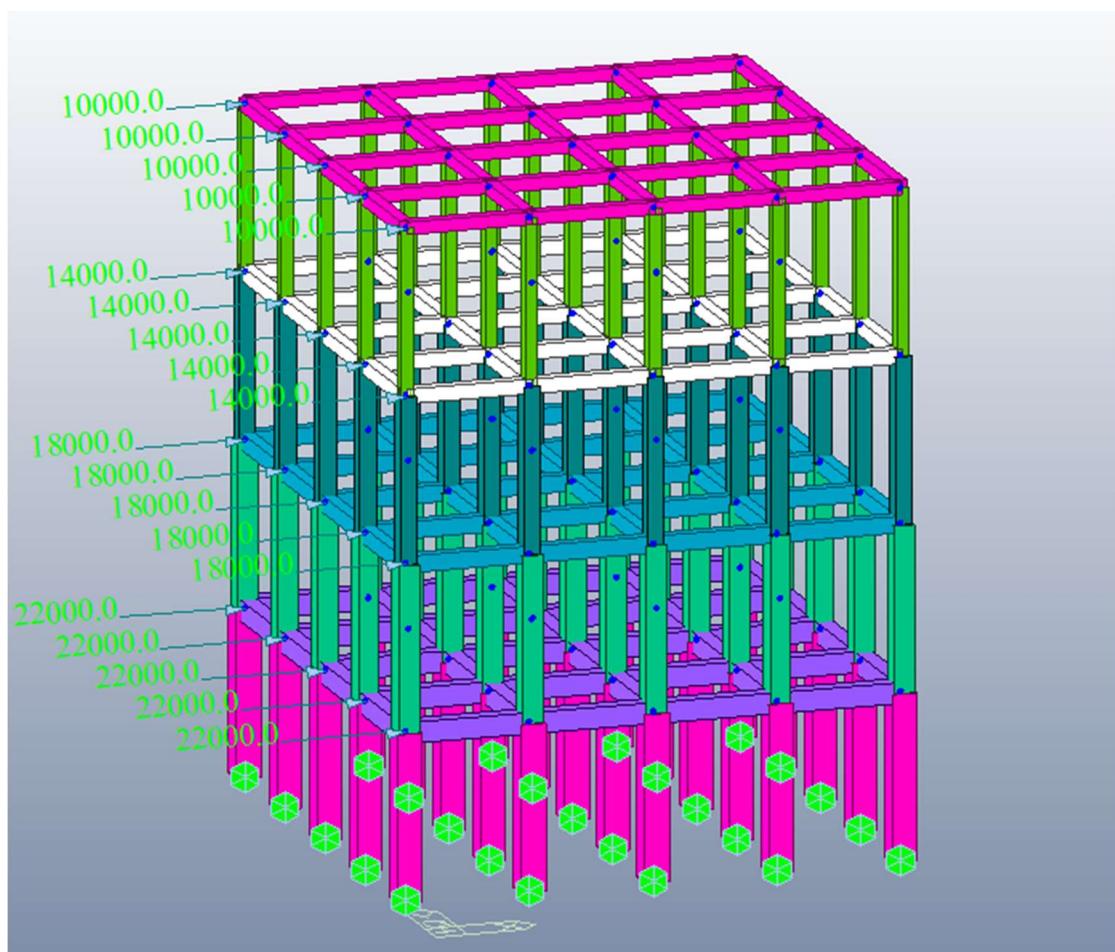
È possibile verificare tale condizione dal file .dat:

```

loads
 1   1.000000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    41
 1   1.000000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    45
 1   1.400000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    51
 1   1.400000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    56
 1   1.000000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    60
 1   1.400000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    66
 1   1.800000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    68
 1   1.800000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    71
 1   1.000000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    76
 1   1.400000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    81
 1   1.800000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    83
 1   1.000000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    91
 1   2.200000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    92
 1   1.800000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    94
 1   2.200000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    97
 1   1.400000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    101
 1   2.200000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    103
 1   1.800000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    110
 1   2.200000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    112
 1   2.200000e+04    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    0.000000e+00    120
end

```

Sono quindi state inserite le medesime proprietà in MIDAS e si è verificata la conformità di risultati



Ne risultano quindi i seguenti spostamenti massimi e rotazioni massime:

	Node	Load	DX (mm)	DY (mm)	DZ (mm)	RX ([rad])	RY ([rad])	RZ ([rad])
	106	forza st	2.19011	0.00000	0.02695	0.00000	0.00017	0.00000
	107	forza st	2.17942	0.00000	-0.00285	0.00000	0.00011	0.00000
	108	forza st	2.17173	0.00000	-0.00003	0.00000	0.00012	0.00000
	109	forza st	2.16701	0.00000	0.00278	0.00000	0.00011	0.00000
	110	forza st	2.16530	0.00000	-0.02684	0.00000	0.00017	0.00000
	111	forza st	3.11734	0.00000	0.02967	0.00000	0.00010	0.00000
	112	forza st	3.10857	0.00000	-0.00332	0.00000	0.00005	0.00000
	113	forza st	3.10221	0.00000	-0.00003	0.00000	0.00005	0.00000
	114	forza st	3.09824	0.00000	0.00325	0.00000	0.00005	0.00000
	115	forza st	3.09669	0.00000	-0.02957	0.00000	0.00010	0.00000
	116	forza st	3.11734	0.00000	0.02967	0.00000	0.00010	0.00000
	117	forza st	3.10857	0.00000	-0.00332	0.00000	0.00005	0.00000
	118	forza st	3.10221	0.00000	-0.00003	0.00000	0.00005	0.00000
	119	forza st	3.09824	0.00000	0.00325	0.00000	0.00005	0.00000
	120	forza st	3.09669	0.00000	-0.02957	0.00000	0.00010	0.00000
	121	forza st	3.11734	0.00000	0.02967	0.00000	0.00010	0.00000
	122	forza st	3.10857	0.00000	-0.00332	0.00000	0.00005	0.00000
	123	forza st	3.10221	0.00000	-0.00003	0.00000	0.00005	0.00000
	124	forza st	3.09824	0.00000	0.00325	0.00000	0.00005	0.00000
	125	forza st	3.09669	0.00000	-0.02957	0.00000	0.00010	0.00000
	126	forza st	3.11734	0.00000	0.02967	0.00000	0.00010	0.00000
	127	forza st	3.10857	0.00000	-0.00332	0.00000	0.00005	0.00000
	128	forza st	3.10221	0.00000	-0.00003	0.00000	0.00005	0.00000
	129	forza st	3.09824	0.00000	0.00325	0.00000	0.00005	0.00000
	130	forza st	3.09669	0.00000	-0.02957	0.00000	0.00010	0.00000
	131	forza st	3.11734	0.00000	0.02967	0.00000	0.00010	0.00000
	132	forza st	3.10857	0.00000	-0.00332	0.00000	0.00005	0.00000
	133	forza st	3.10221	0.00000	-0.00003	0.00000	0.00005	0.00000
	134	forza st	3.09824	0.00000	0.00325	0.00000	0.00005	0.00000
	135	forza st	3.09669	0.00000	-0.02957	0.00000	0.00010	0.00000

\Displacements(Global) / Displacements(Local) / <

Il codice fornisce i seguenti parametri massimi:

Editor - main.m		Variables - u								
	u	1	2	3	4	5	6	7	8	9
		750x1 double								
1		2.9777								
2		1.2759e-15								
3		-0.0296								
4		5.7921e-20								
5		1.0439e-04								
6		1.7123e-18								
7		2.9777								
8		1.2342e-15								
9		-0.0296								
10		7.3190e-20								
11		1.0439e-04								
12		4.4629e-18								
13		2.9792								
14		6.5081e-16								
15		0.0033								
16		2.3117e-20								

Andando ad analizzare lo spostamento massimo in X si verifica che i due valori non sono pienamente concordi, anche se mantengono stesso ordine di grandezza, come si verificava tramite lo sviluppo di strutture contenenti mesh a ridotto numero nodale.

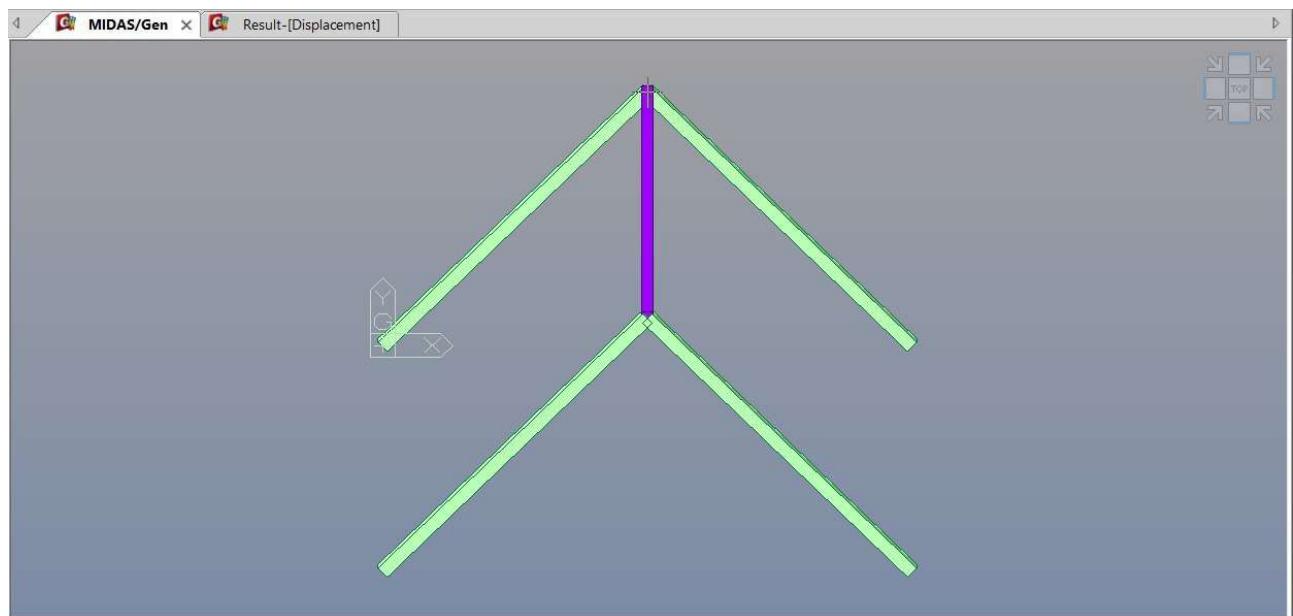
Si verifica infatti che la differenza di risultati, pari a poco più del 3% e concorde alle approssimazioni usate per il dimensionamento della struttura.

TEST SULLE STRUTTURE COMPOSTE (BEAM -TRUSS)

Per verificare il corretto assemblaggio di elementi truss e beam all'interno del codice si è deciso di verificare una semplice struttura composta da 5 aste e una struttura più complessa che simuli il funzionamento di un ponte strallato e di conseguenza l'assemblaggio di sezioni più complesse da gestire

PIEDISTALLO

Tipologia di struttura analizzata:



ELEMENTI:

-Aste Inclinate:

- Beam
- Sezione Quadrata : 50mm
- Angolo di Inclinazione : 45°

-Asta Orizzontale:

- Truss

- Sezione Quadrata : 40mm

CARICO APPLICATO

- Z ortogonale come da sistema di riferimento : 10000 N

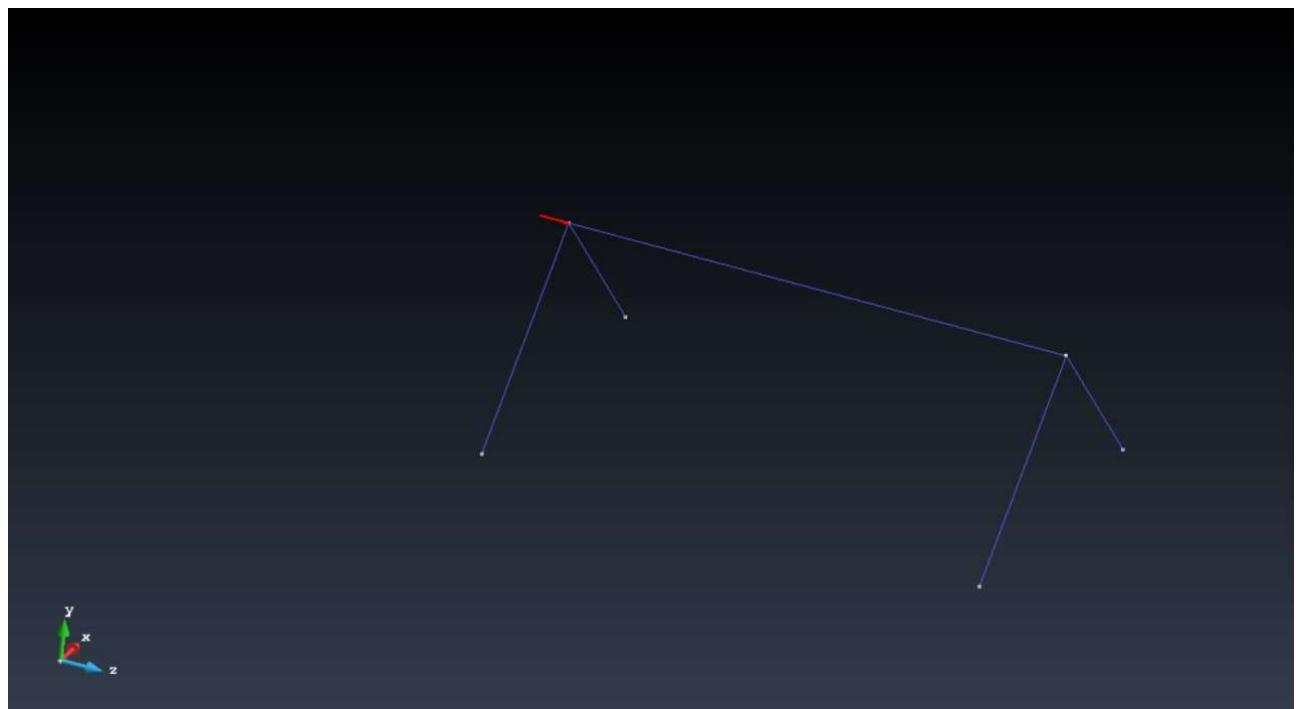
VINCOLI ALLA BASE

- Incastri alla Base

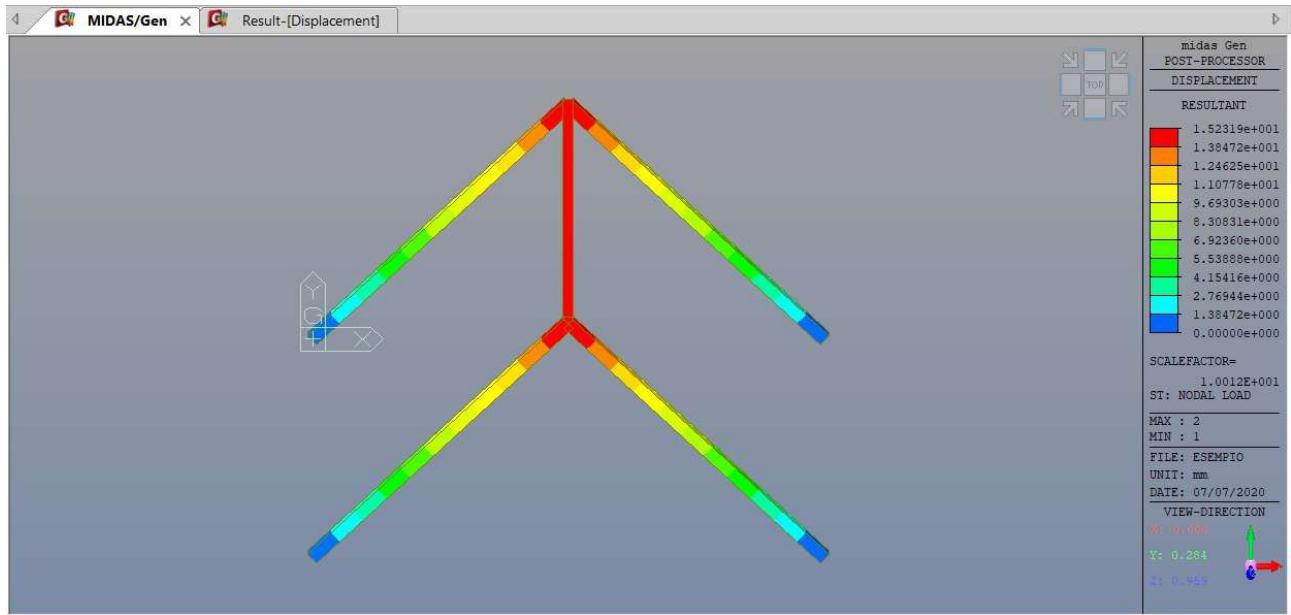
TIPOLOGIA DI MATERIALE

- Acciaio S235
- $E=210000 \text{ N/mm}^2$
- $\nu=0.3$

Inserimento della Struttura (GID)



Calcolo degli spostamenti in MIDAS:



Spostamenti ricavati dal software di calcolo MIDAS:

	Node	Load	DX (mm)	DY (mm)	DZ (mm)	RX ([rad])	RY ([rad])	RZ ([rad])
▶	1	Nodal L	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
◀	2	Nodal L	0.00000	0.00000	15.2319	0.01963	0.00000	0.00000
●	3	Nodal L	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
◆	4	Nodal L	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
◆	5	Nodal L	0.00000	0.00000	15.1873	0.01957	0.00000	0.00000
◆	6	Nodal L	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

Spostamenti Ricavati dal software di calcolo creato in MATLAB:

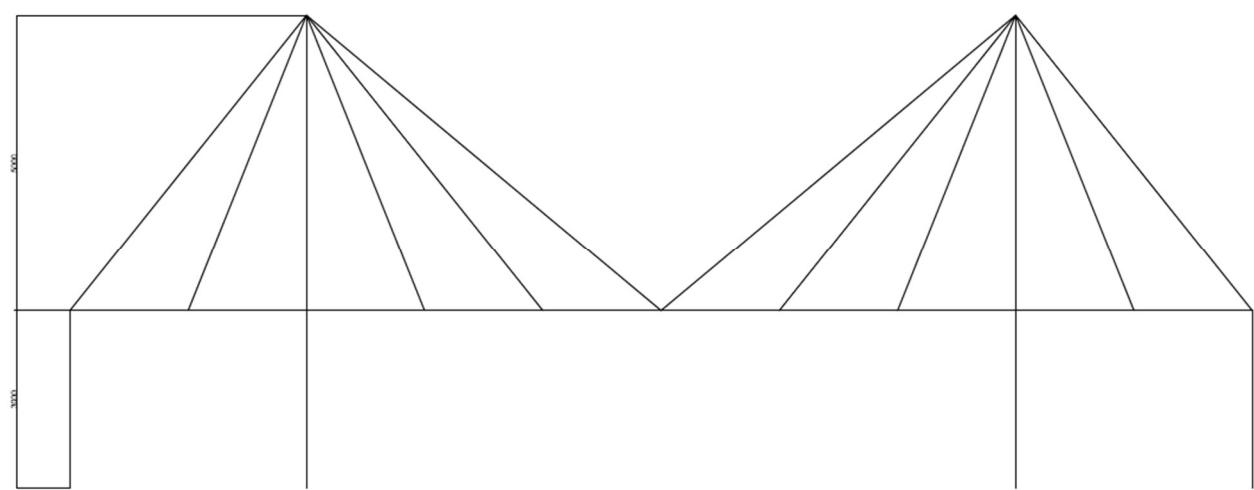
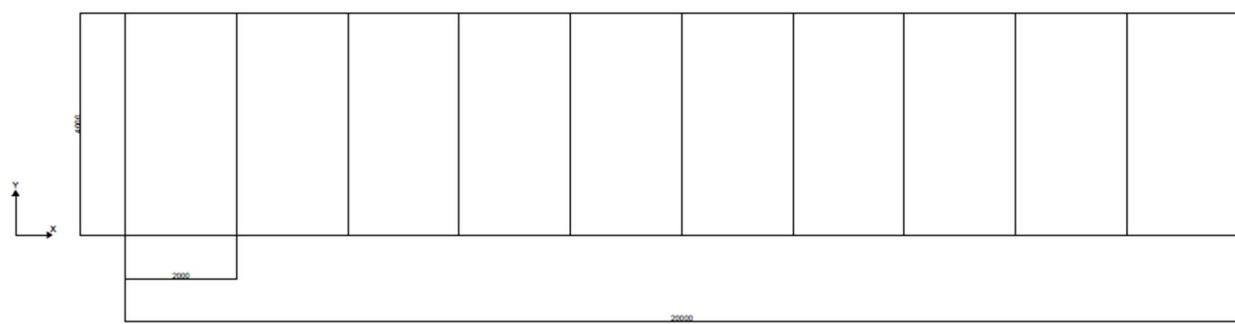
u	
	36x1 double
1	
7	0
8	0
9	15.1663
10	0.0196
11	-6.1815e-18
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0
21	0
22	0
23	0
24	0
25	0
26	0
27	15.2109
28	0.0196
29	0
30	0

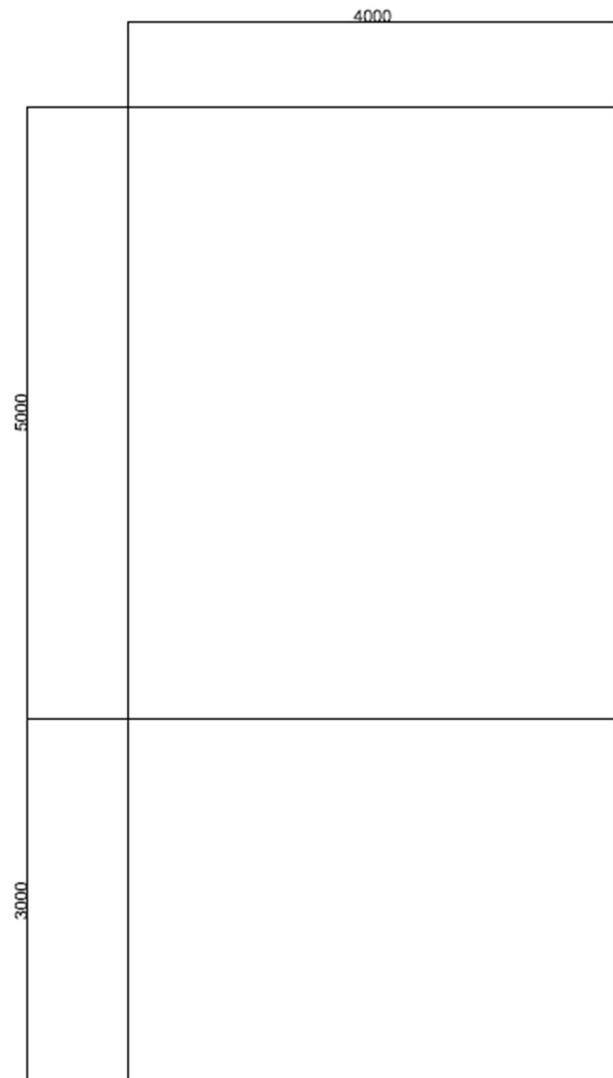
PONTE STRALLATO

Viene ora proposta la risoluzione di un ponte strallato, dove gli stralli vengono realizzati con elementi truss agenti come tiranti per 10 campate realizzate.

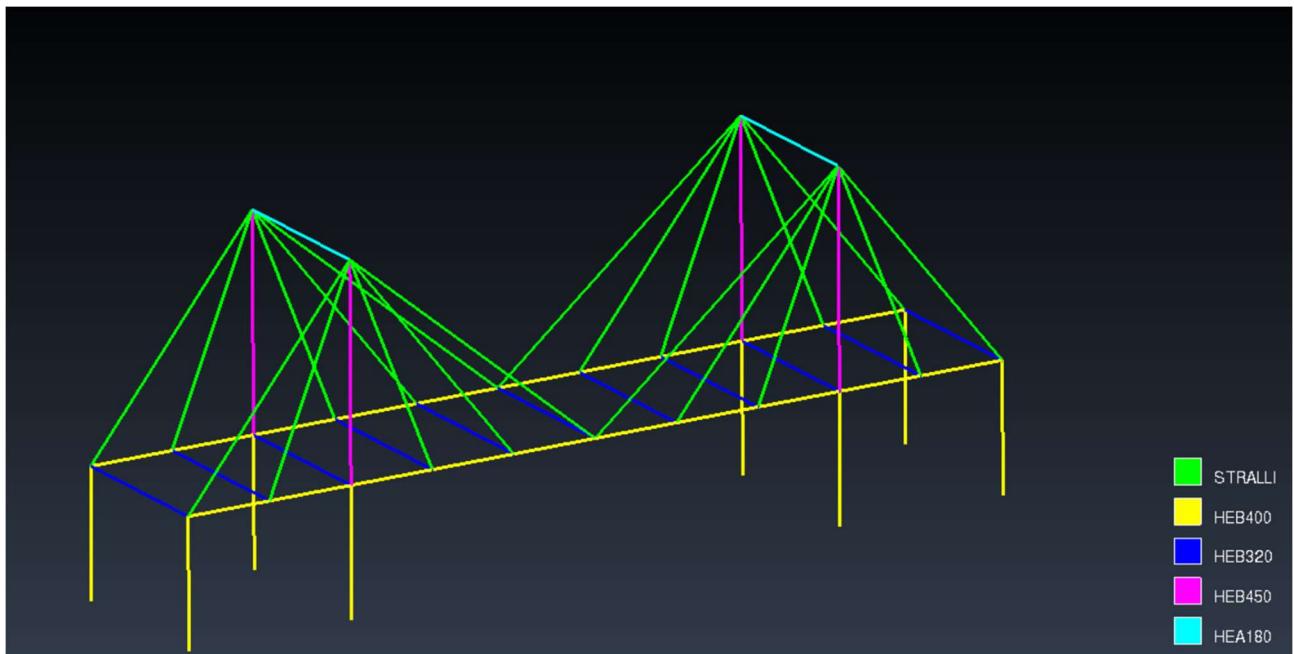
La risoluzione, non essendo ancora implementati i carichi distribuiti, non può ovviamente essere completa ma funge da prova di validazione per verificare il comportamento del codice in condizioni di elevata complessità di sezione ed elementi assemblati.

GEOMETRIA:





Realizzazione della geometria in GID:



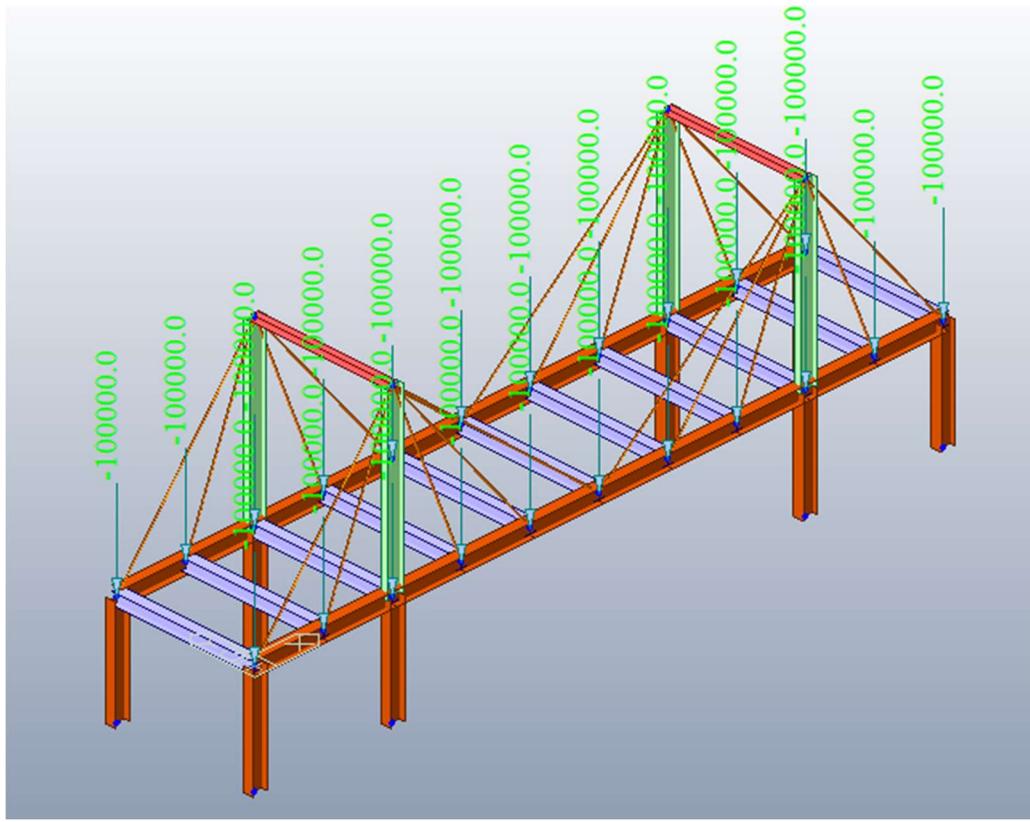
PROPRIETÀ DELLE SEZIONI:

HEB400	A =	19800 mm ²	HEB320	A =	16100 mm ²	HEB450	A =	9880 mm ²	STRALLI	A =	1960 mm ²	HEA180	A =	4530 mm ²
	I11 =	3070000 mm ⁴		I11 =	1870000 mm ⁴		I11 =	515000 mm ⁴		I11 =	307000 mm ⁴		I11 =	115000 mm ⁴
	I22 =	577000000 mm ⁴		I22 =	308000000 mm ⁴		I22 =	337000000 mm ⁴		I22 =	307000 mm ⁴		I22 =	25100000 mm ⁴
	J =	108000000 mm ⁴		J =	92400000 mm ⁴		J =	16800000 mm ⁴		J =	614000 mm ⁴		J =	9250000 mm ⁴

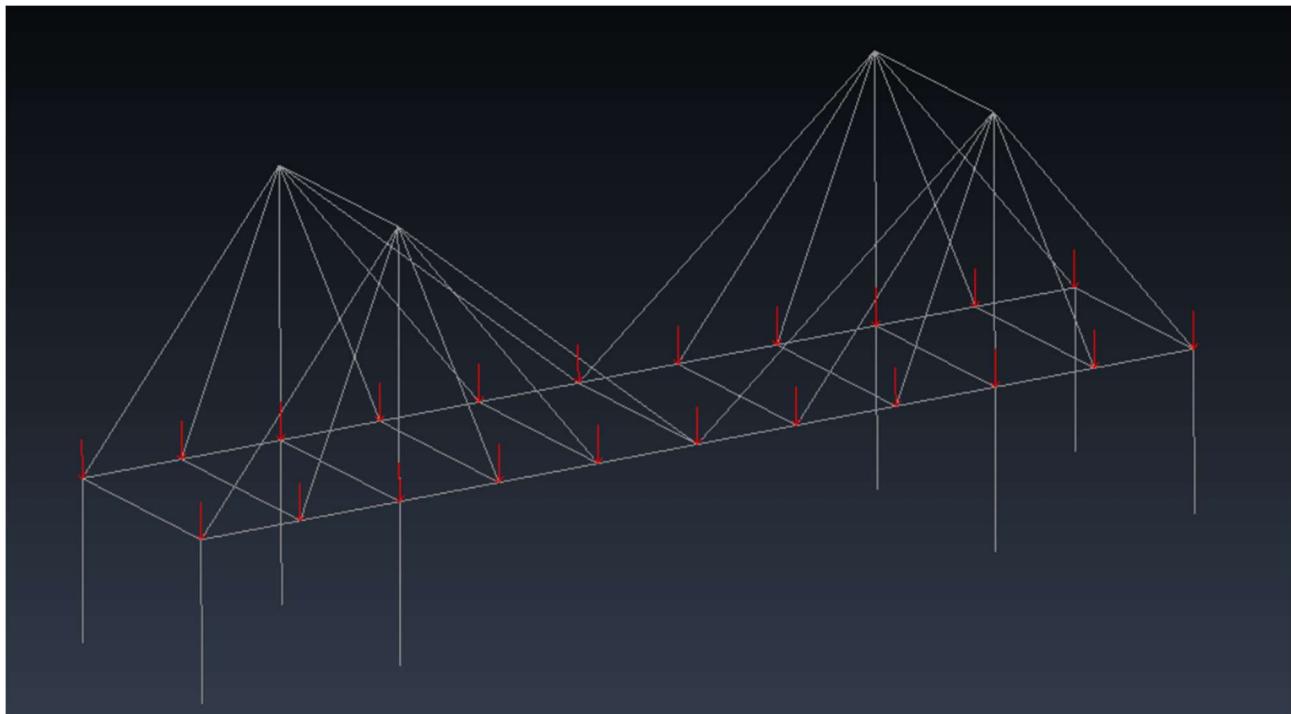
```
! ID mat      E          nu        A           I11        I22        J
properties
  1  2.10000e+05  3.00000e-01  1.96000e+03  3.07000e+05  3.07000e+05  6.14000e+05
  2  2.10000e+05  3.00000e-01  1.98000e+04  3.07000e+06  5.77000e+08  1.08000e+08
  3  2.10000e+05  3.00000e-01  1.61000e+04  1.87000e+06  3.08000e+08  9.24000e+07
  4  2.10000e+05  3.00000e-01  9.88000e+03  5.15000e+05  3.37000e+08  1.68000e+07
  5  2.10000e+05  3.00000e-01  4.53000e+03  1.15000e+05  2.51000e+07  9.25000e+06
end
boundaries
  1  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
  4  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
  5  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
  9  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
 23  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
 25  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
 31  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
 33  1   1   1   1   1   1   0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00  0.0000000e+00
```

Le sezioni Verticali verranno incastrate alla base, come è visibile sul file.dat

REALIZZAZIONE DELLA GEOMETRIA IN MIDAS E CARICHI APPLICATI:



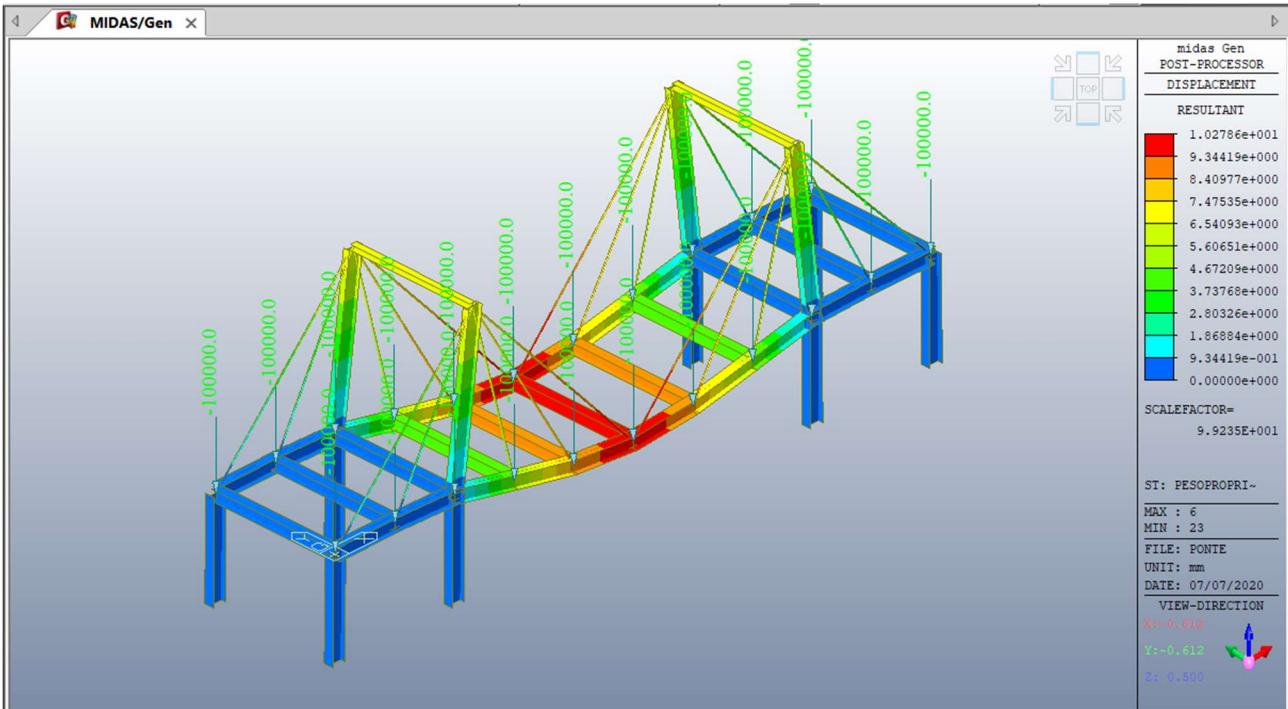
Ad ogni nodo della sezione nel piano X-Y verrà applicato un carico di 100KN che ci permetterà poi di verificare spostamenti apprezzabili e confrontabili all'interno del codice.



CONTROLLO DEGLI SPOSTAMENTI:

- Spostamenti Ricavati da Midas:

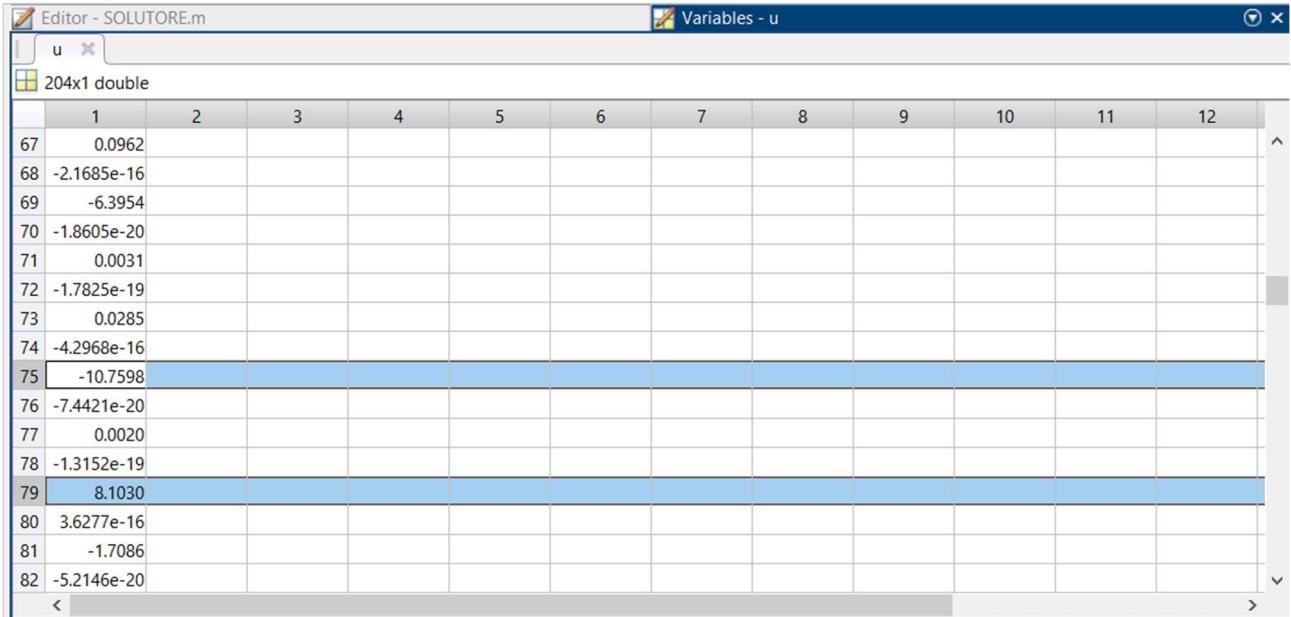
Massimo spostamento DZ:10,27mm



	Node	Load	DX (mm)	DY (mm)	DZ (mm)	RX ([rad])	RY ([rad])	RZ ([rad])
	5	pesopr	0.01352	0.00000	-8.60356	0.00000	0.00148	0.00000
	6	pesopr	0.00000	0.00000	-10.2786	0.00000	0.00000	0.00000
	7	pesopr	-0.01352	0.00000	-8.60356	0.00000	-0.00148	0.00000
	8	pesopr	-0.06171	0.00000	-4.62152	0.00000	-0.00220	0.00000
	9	pesopr	-0.12035	0.00000	-0.43410	0.00000	-0.00123	0.00000
	10	pesopr	-0.17902	0.00000	0.21612	0.00000	0.00000	0.00000
	11	pesopr	-0.22646	0.00000	0.03728	0.00000	0.00001	0.00000
	12	pesopr	0.17902	0.00000	0.21612	0.00000	-0.00000	0.00000
	13	pesopr	0.12035	0.00000	-0.43410	0.00000	0.00123	0.00000
	14	pesopr	0.06171	0.00000	-4.62152	0.00000	0.00220	0.00000
	15	pesopr	0.01352	0.00000	-8.60356	0.00000	0.00148	0.00000
►	16	pesopr	0.00000	0.00000	-10.2786	0.00000	0.00000	0.00000
	17	pesopr	-0.01352	0.00000	-8.60356	0.00000	-0.00148	0.00000
	18	pesopr	-0.06171	0.00000	-4.62152	0.00000	-0.00220	0.00000
	19	pesopr	-0.12035	0.00000	-0.43410	0.00000	-0.00123	0.00000
	20	pesopr	-0.17902	0.00000	0.21612	0.00000	0.00000	0.00000
	21	pesopr	-0.22646	0.00000	0.03728	0.00000	0.00001	0.00000
	22	pesopr	0.22646	0.00000	0.03728	0.00000	-0.00001	0.00000
	23	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	24	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	25	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	26	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	27	pesopr	6.24413	0.00000	-1.32547	0.00000	0.00122	0.00000
	28	pesopr	6.24413	0.00000	-1.32547	0.00000	0.00122	0.00000
	29	pesopr	-6.24413	0.00000	-1.32547	0.00000	-0.00122	0.00000
	30	pesopr	-6.24413	0.00000	-1.32547	0.00000	-0.00122	0.00000
	31	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	32	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	33	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
	34	pesopr	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

- Spostamenti Ricavati dal Solutore:

Massimo Spostamento DZ: 10,75mm



The screenshot shows the MATLAB Editor window with the title 'Editor - SOLTORE.m'. The variable 'u' is selected in the workspace. The data is displayed as a 204x1 double matrix. The maximum value in the matrix is highlighted in blue and has a value of 10.7598.

	1	2	3	4	5	6	7	8	9	10	11	12
67	0.0962											
68	-2.1685e-16											
69	-6.3954											
70	-1.8605e-20											
71	0.0031											
72	-1.7825e-19											
73	0.0285											
74	-4.2968e-16											
75	-10.7598											
76	-7.4421e-20											
77	0.0020											
78	-1.3152e-19											
79	8.1030											
80	3.6277e-16											
81	-1.7086											
82	-5.2146e-20											

Come per la risoluzione del telaio a 4 piani la risoluzione del codice, seppur approssimata di un 3.5%, risulta concorde con quanto risolto dal codice commerciale.

LIBRI DI TESTO CONSULTATI

- A first course in finite elements (Fish Belytschko)
- A First Course in the Finite Element Method (Daryl; L.Logan)
- Sviluppo di Elementi Finiti con Non Linearità (Tesi di Gabriele Attanasi-anno2004_2005)
- Introduction to finite element analysis using matlab and abaqus (Amar Khennane)
- MATLAB based Finite Element Programming in Electromagnetic Modeling (O.Ozgun ; M.Kuzuoglu)
- Matlab codes for finite element analysis (A.J.M. Ferreira)
- Structural Analysis with the Finite Element Method _volume1 (E. Onate)
- Structural Analysis with the Finite Element Method _volume2 (E. Onate)