POLITECNICO
MILANO 1863

School of Civil, Environmental and Land Management Engineering
M.Sc program in Civil Engineering for Risk Mitigation
Computational Mechanics

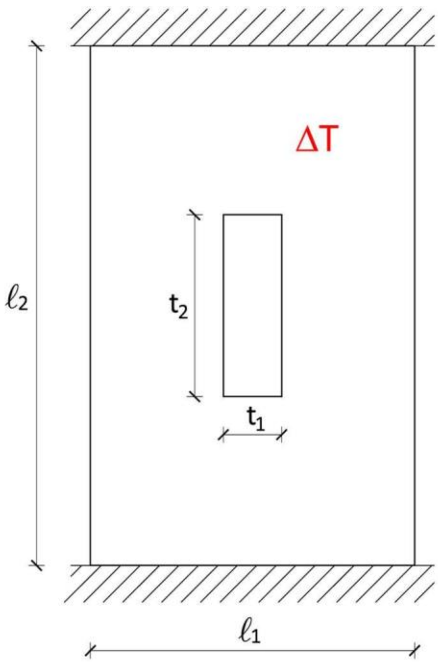
Prof. Gabriella Bolzon
Presented by: Jasem Avaz Nasab

1st Lab
A.Y:21-22

First computing lab - plane elements

STUDENT IDENTIFICATION NUMBER: **<u>9 9 6 3 1 0</u>**
a b c d e f

Consider the rectangular steel plate with a central rectangular hole schematized below, clamped along two sides and subjected to the uniform temperature increase ΔT=30°C.
Assume ℓ_1=(250+5·f) mm, ℓ_2=(400-5·e) mm, thickness h=5 mm for the dimensions of the plate, t_1=(45+10·d) mm and t_2=(140-10·d) mm for the hole sides, where d, e and f coincide with the last digits of your student id number.



Consider for the material the typical properties of a structural steel: elastic modulus E=210000 N/mm^2, Poisson coefficient ν=0.30, thermal expansion coefficient α=1.2·10^{-5} °C^{-1}.
Compare the displacement and stress distributions resulting from two different meshes (one rough, one finer) made of plane stress elements.
Modify the Matlab code in order to introduce the variation of temperature.

MAIN RESULTS
Deliver <u>only</u> the main results listed on pages 2 and 3, filling in the indicated fields, and further provide
- for each discretization:
1) ONE FIGURE REPRESENTING THE STRUCTURAL SCHEME (with legible numbers of nodes and elements)
2) ONE FIGURE REPRESENTING THE DEFORMED SHAPE OF THE STRUCTURE (with legible amplification factor)
3) THE MAP OF THE HORIZONTAL NORMAL STRESS (with legible scale)
4) THE MAP OF THE VERTICAL NORMAL STRESS (with legible scale)
5) THE MAP OF SHEAR STRESS (with legible scale)
6) THE MAP OF VON MISES STRESS (with legible scale)
- for the comparison:
7) ONE FIGURE WITH THE GRAPHICAL COMPARISON OF THE DEFORMED SHAPE OF THE 2 MESHES (with only the external sides in view)
- for modifications referred to the variation of temperature:
8) THE MODIFIED PARTS OF MATLAB CODE

Specify the adopted sign conventions in your graphs and numerical results.

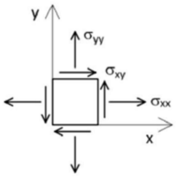
The delivery mode has to follow the instructions in the published document "Delivery Deadlines".

First computing lab - MODULE OF RESULTS

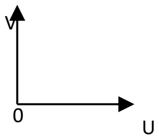
SURNAME: **AVAZ NASAB** NAME: **JASEM**

STUDENT IDENTIFICATION NUMBER: <u>**9 9 6 3 1 0**</u> PERSON CODE: **10753087**

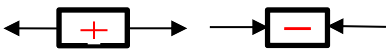
a b c d e f



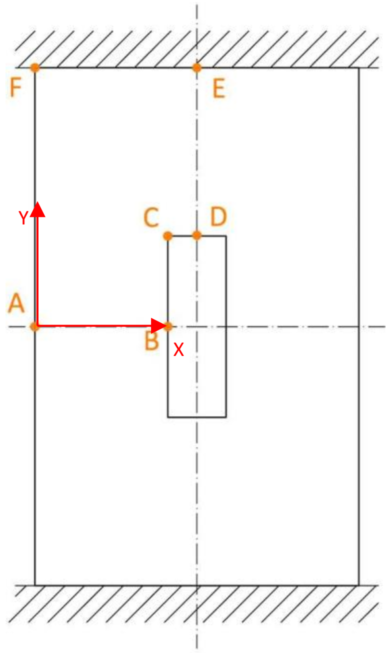
Sign convention: stress



Sign convention: displacements



Sign convention: strain



ROUGH MESH

9) REACTION FORCES (all + their sum)

- Unit of measure of reactions is NEWTON (N).
- On the Direction(X) **rightward** and on the Direction(Y) **upward** are positive.

Nodes	Direction(X) [N]	Direction(Y) [N]
1	0.00	5163.65
2	0.00	14347.96
3	0.00	14817.96
4	0.00	6096.55
14	3126.64	0.00
20	743.39	0.00
26	-5780.40	0.00
32	-11408.01	0.00
33	11462.95	-14537.39
34	5458.19	-11505.44
35	2640.68	-5994.97
36	960.29	-3569.95
37	443.95	-3249.40
38	-7647.68	-1568.97
SUM	**0.00**	**0.00**

10) HORIZONTAL NORMAL STRESS, VERTICAL NORMAL STRESS, SHEAR STRESS, VON MISES STRES IN THE HIGHLIGHTED NODES

- Unit of measure of Stresses is (N/mm2).
- Compression is considered (Negative) & Tension is (Positive).

node	σ_{xx}	σ_{yy}	σ_{xy}	σ_{VM}
A	0.17	-55.99	5.92	57.01
B	1.03	-126.26	4.65	127.03
C	1.88	-112.97	-30.43	125.52
D	76.94	3.24	1.93	75.44
E	-87.40	-39.32	0.00	75.81
F	-127.30	-172.34	83.52	211.88

11) NODAL DISPLACEMENTS

- Unit of measure of displacements are (mm).
- On the Direction(X) **rightward** and on the Direction(Y) **upward** are positive.

Nodes	Displacement(X) [mm]	Displacement(Y) [mm]
A	-0.073	0.000
B	-0.030	0.000
C	-0.024	-0.018
D	0.000	-0.037

FINE MESH

12) REACTION FORCES (all + their sum)

- Unit of measure of reactions is NEWTON (N).
- On the Direction(X) **rightward** and on the Direction(Y) **upward** are positive.

NODE	DIRECTION X	DIRECTION Y
1	0.000	2435.710
2	0.000	5861.435
3	0.000	7082.378
4	0.000	8386.694
5	0.000	7411.978
6	0.000	5727.395
7	0.000	3033.965
39	1838.543	0.000
50	2140.102	0.000
61	524.880	0.000
72	-1257.149	0.000
83	-2719.483	0.000
94	-4072.078	0.000
105	-5559.634	0.000
116	-7095.207	0.000
117	6916.967	-9919.587
118	4281.564	-8068.216
119	3426.752	-5444.307
120	2383.141	-4444.641
121	1317.198	-2954.150
122	680.304	-1880.236
123	513.427	-1749.354
124	366.692	-1643.687
125	235.480	-1565.947
126	114.996	-1518.306
127	-4036.494	-751.125
SUM	**0.00**	**0.00**

13) HORIZONTAL NORMAL STRESS, VERTICAL NORMAL STRESS, SHEAR STRESS, VON MISES STRES IN THE HIGHLIGHTED NODES

- Unit of measure of Stresses is (N/mm2).
- Compression is considered (Negative) & Tension is (Positive).

Nodes	σ_{xx} [N/mm2]	σ_{yy} [N/mm2]	σ_{xy} [N/mm2]	σ_{VM} [N/mm2]
A	1.314	-52.777	2.035	53.562
B	-0.700	-129.062	1.486	128.739
C	-13.657	-128.745	-39.842	140.590
D	77.482	3.508	-3.075	75.975
E	-86.120	-35.066	0.000	75.009
F	-145.112	-231.706	108.296	276.236

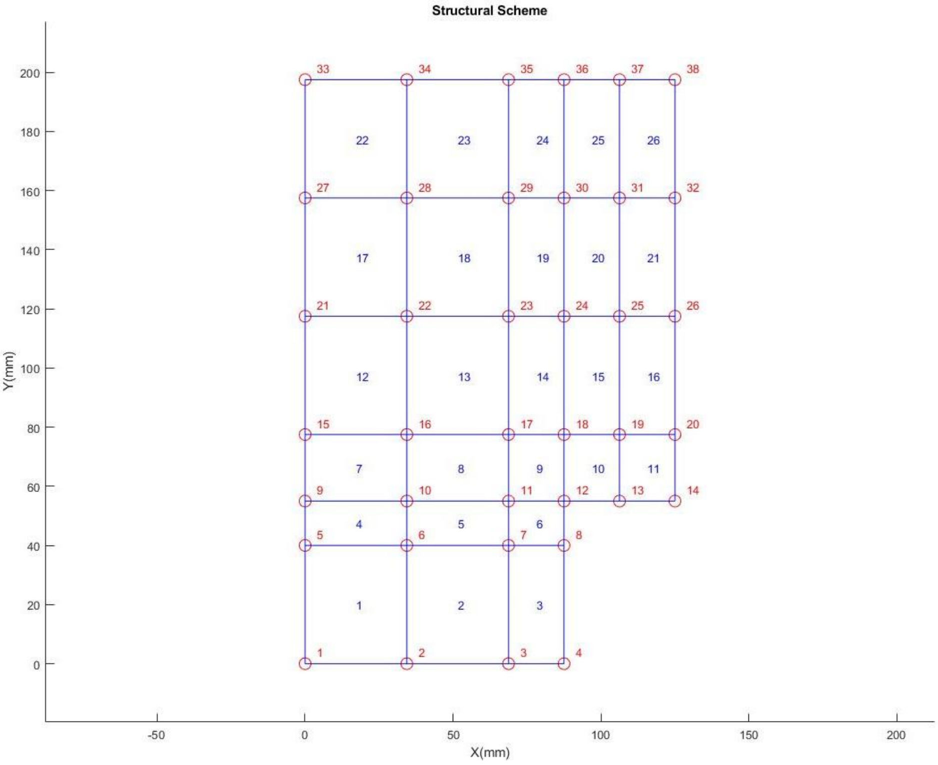
14) NODAL DISPLACEMENTS

- Unit of measure of displacements are (mm).
- On the Direction(X) **rightward** and on the Direction(Y) **upward** are positive

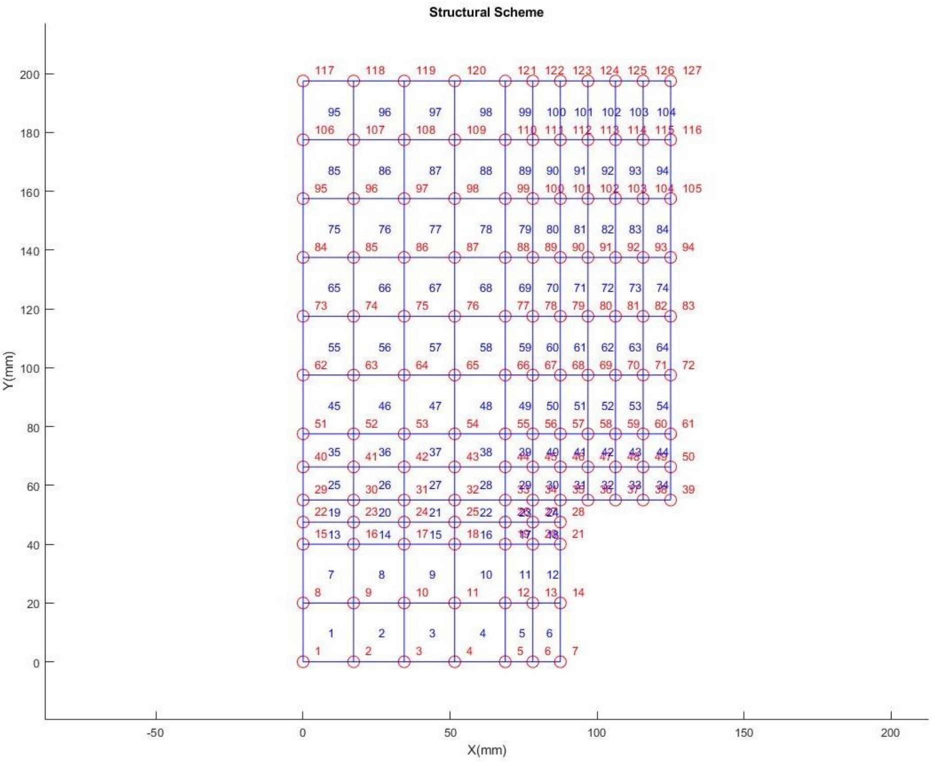
Nodes	Displacement(X) [mm]	Displacement(Y) [mm]
A	-0.073	0.000
B	-0.030	0.000
C	-0.025	-0.018
D	0.000	-0.037

1) Figure representing the structure scheme.

Rough mesh

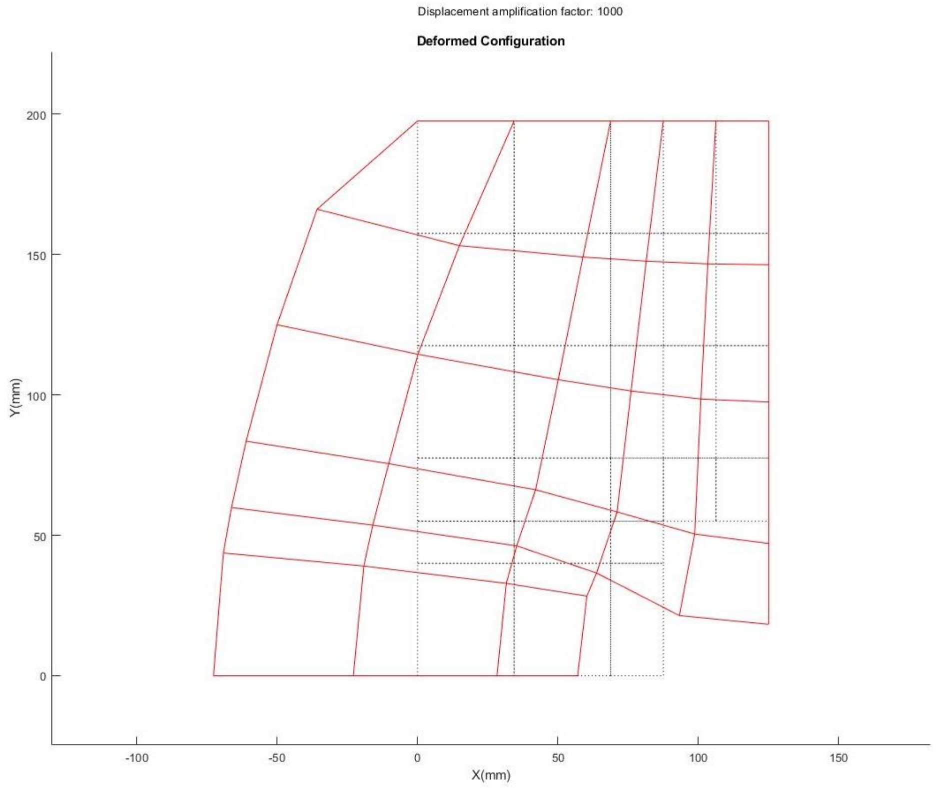


Fine mesh

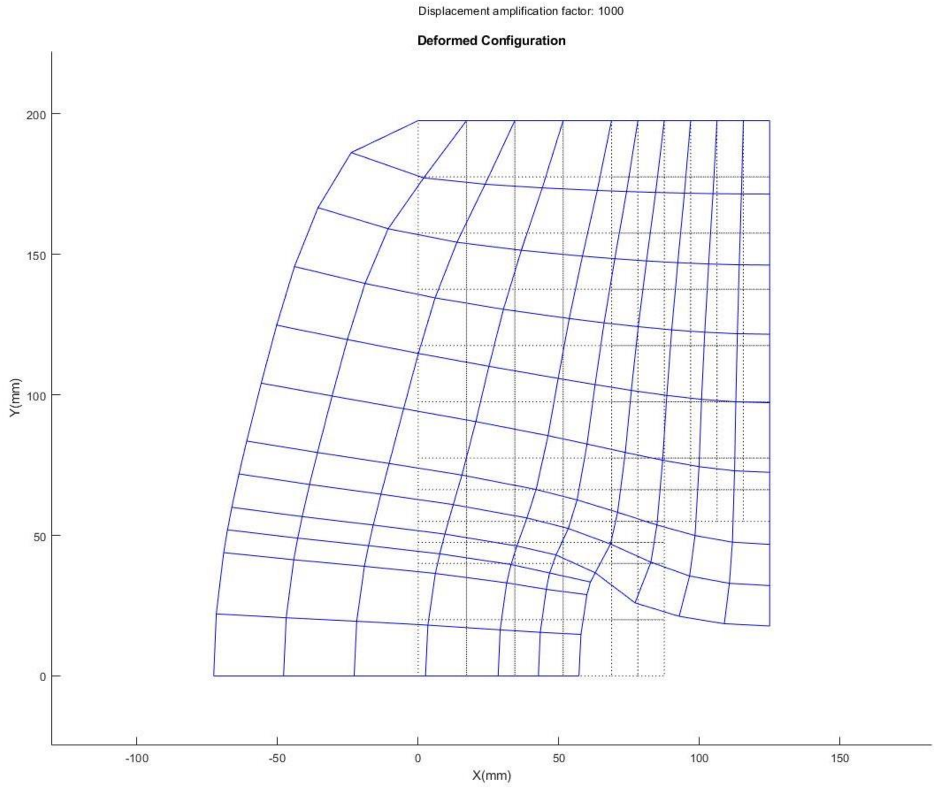


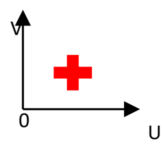
2) Deformed shape of the structure

Rough mesh



Fine mesh

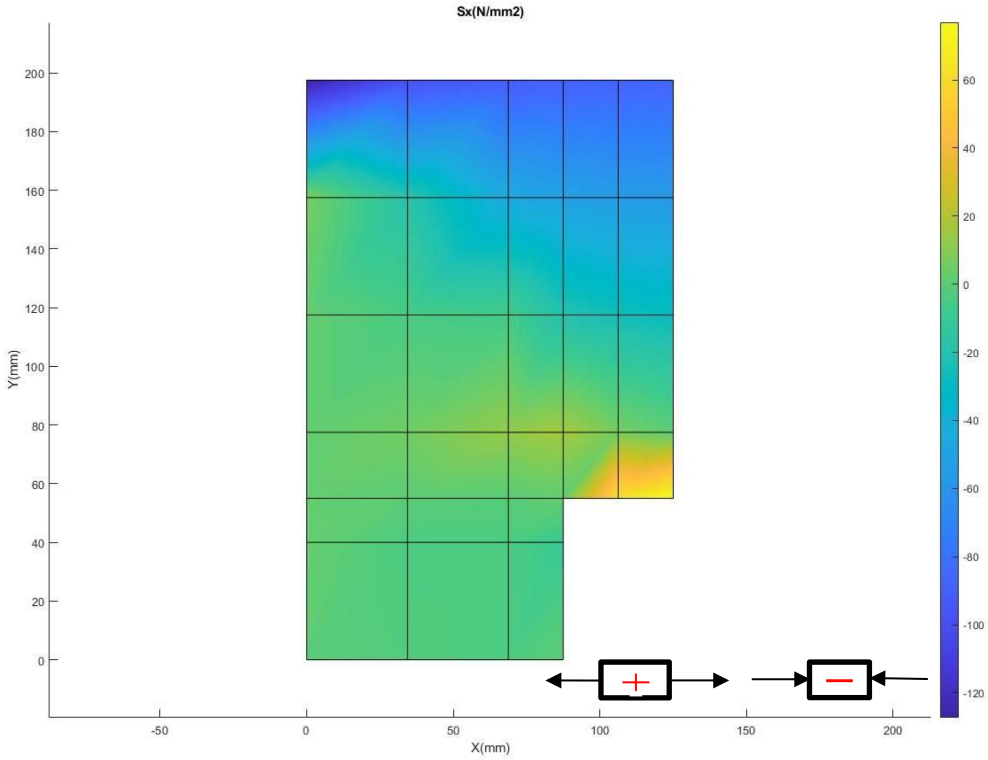




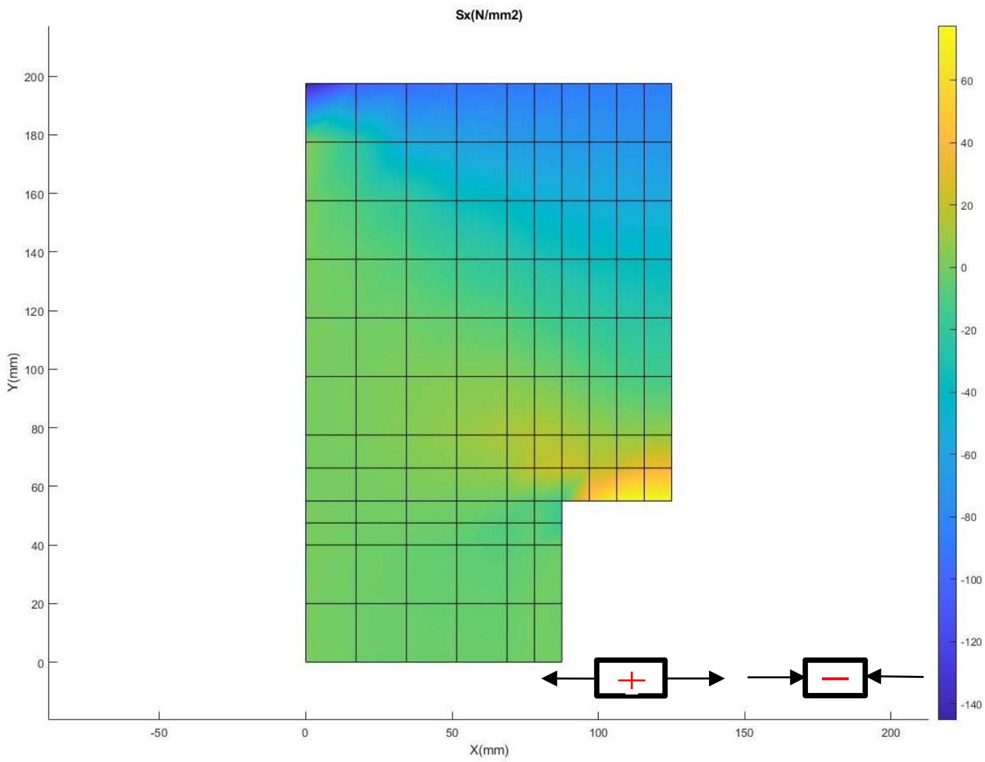
Sign convention: displacements

3) The map of horizontal stresses

Rough mesh

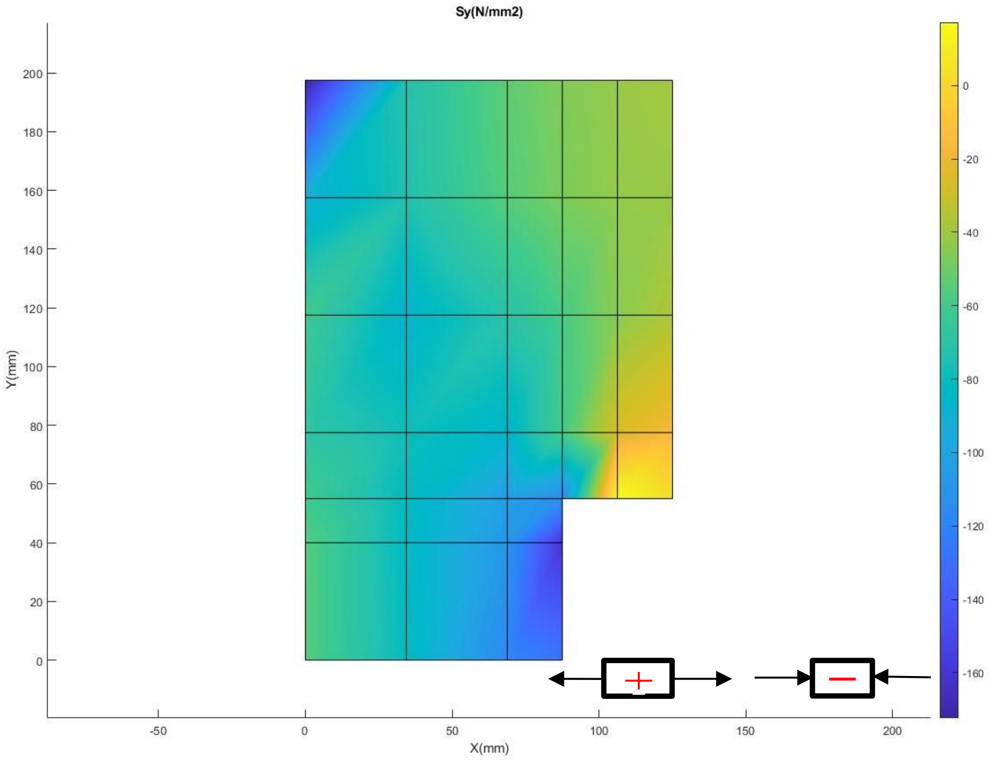


Fine mesh

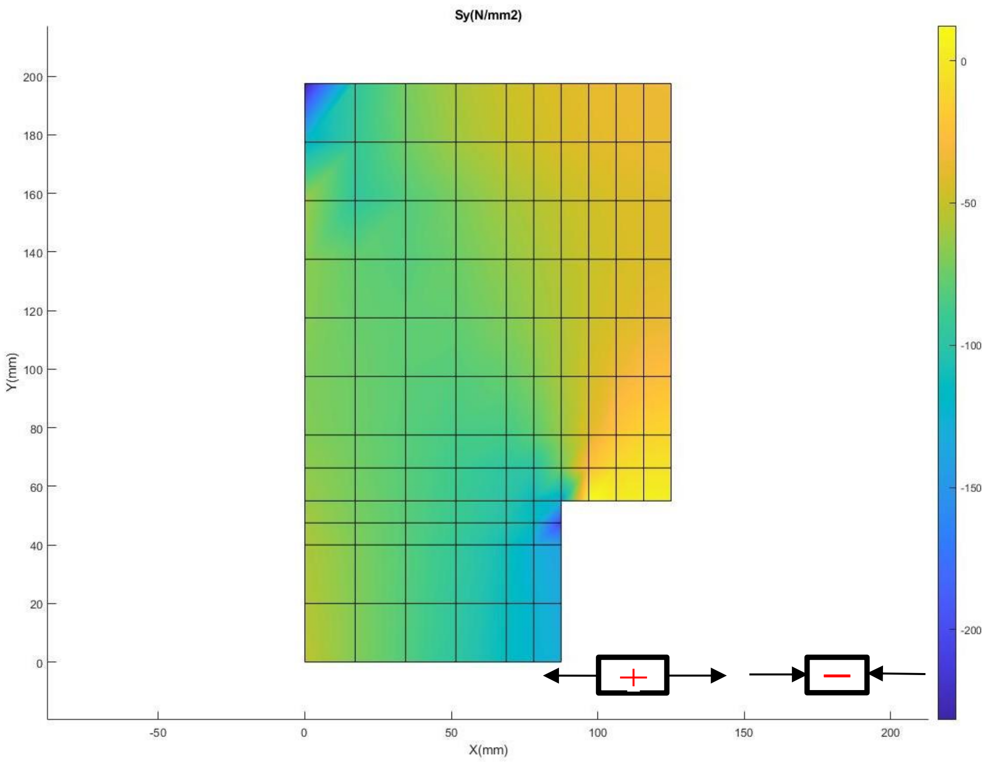


4) The map of the vertical normal stresses

Rough mesh

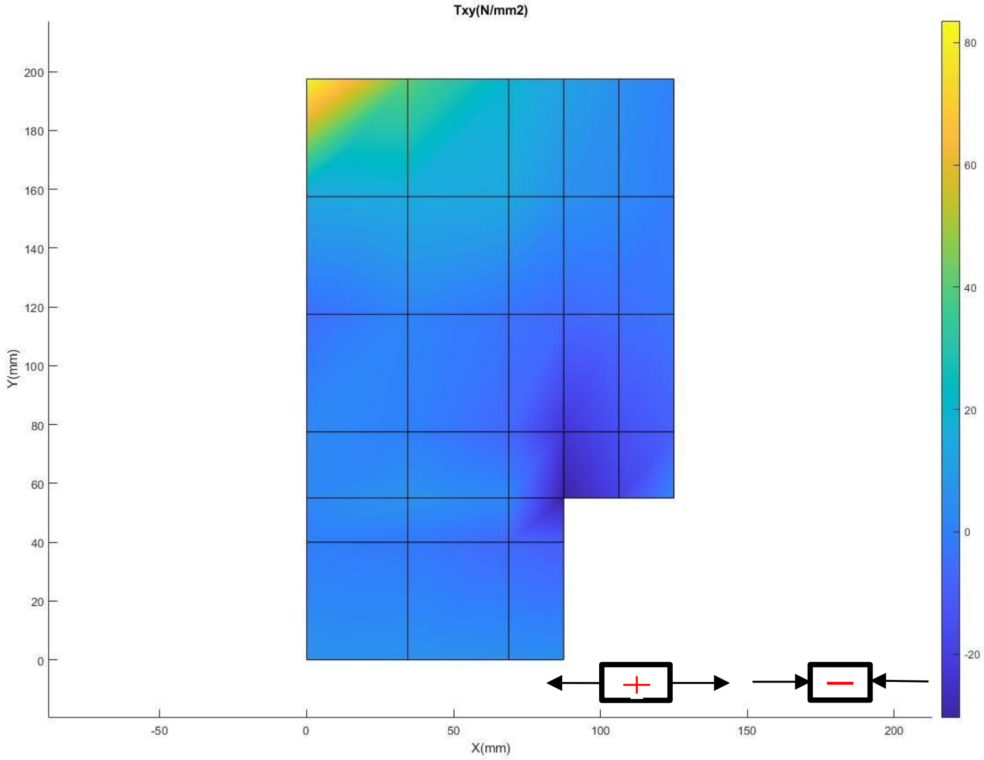


Fine mesh

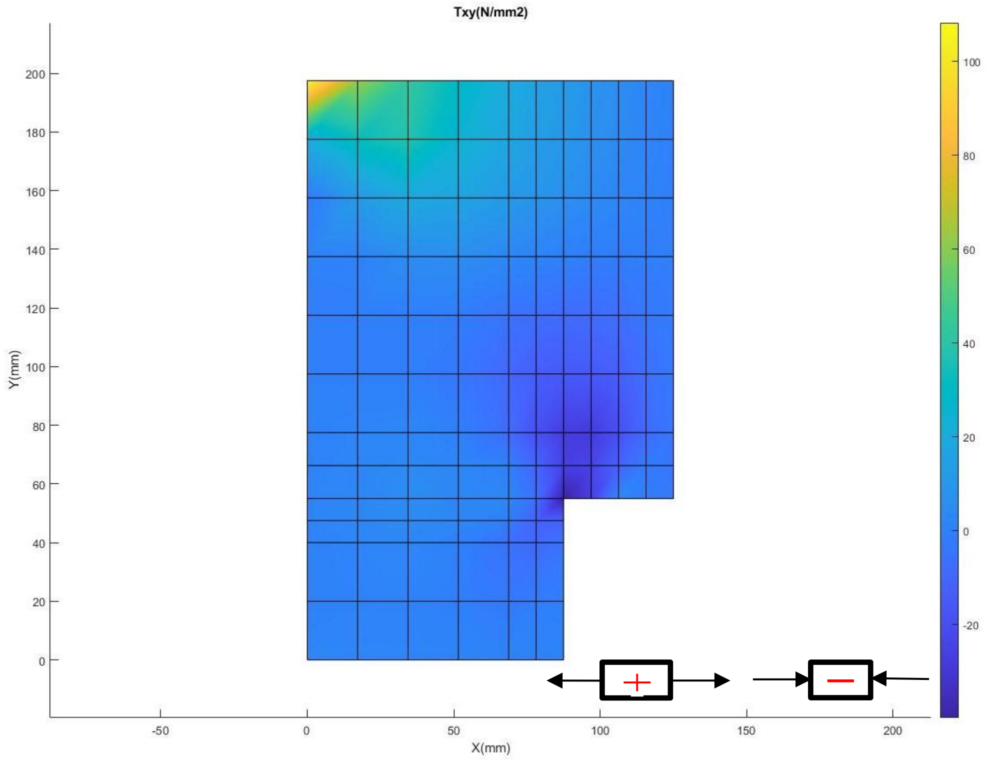


5) The map of shear stresses

Rough mesh

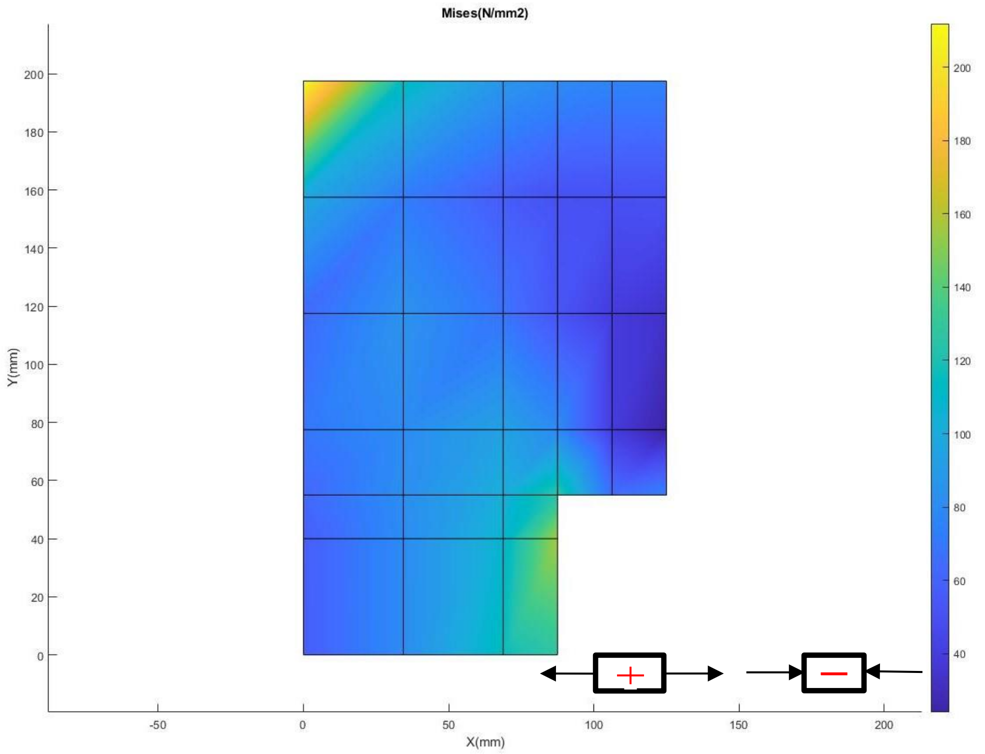


Fine mesh

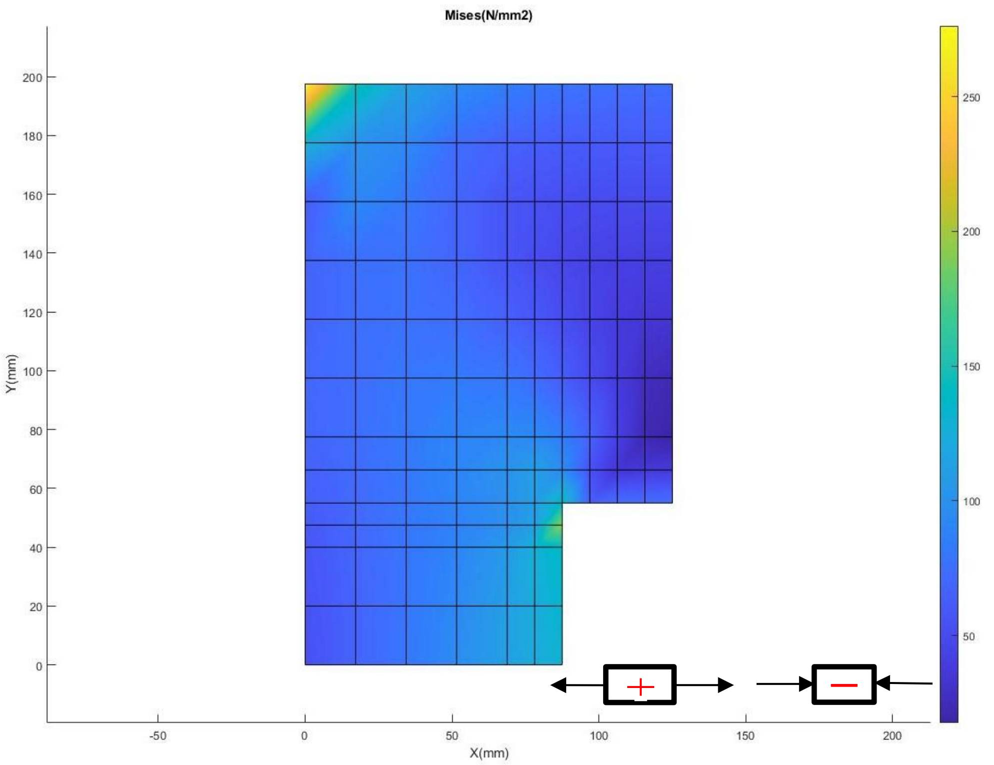


6) The map of VON MISWS stress

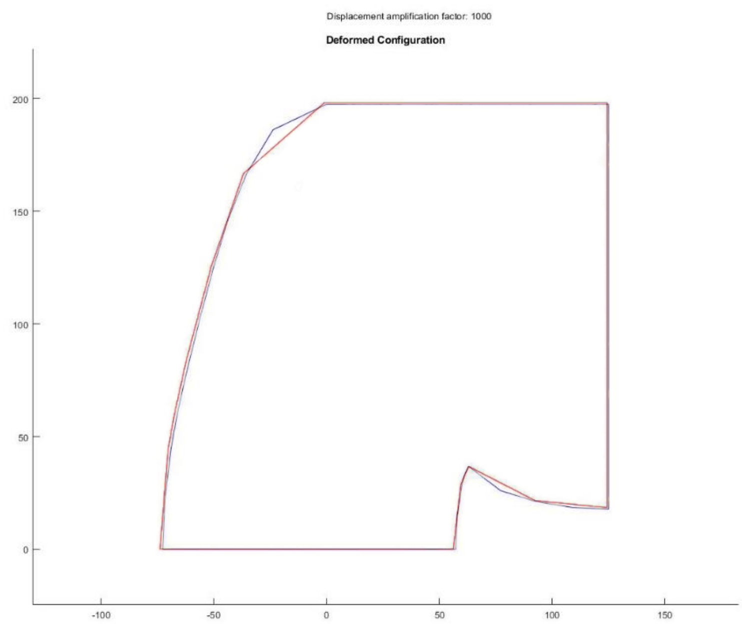
Rough mesh



Fine mesh



7) The figure with the graphical comparison of the deformed shape of 2 meshes



Fine mesh
Rough mesh

8) The modified parts of Matlab code

Mecpar:

```
function [dPar,Alfa]=mecpar

 % Matrix dPar:
 % collects Young's modulus and Poisson's coefficient for the material;
 % dPar=[E, ni]

 Alfa=1.2e-5;
 dE=210000;
 dni=0.3;
 dPar=[dE, dni, Alfa];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Locons :

```
function [nCons,dC,nForce,dF,npq,dpq,dt]=locons

 % Load matrix dF:

 dF=[];

 [nForce,nn]=size(dF);  % nForce=total number of considered loads

 % Constraint matrix dC:


dt=30;
 dC=[33,1,0;
        33,2,0;
        .
        .
        .
         ];

 [nCons,nn]=size(dC);  % nCons=total number of constrained dofs

 % Distributed load matrix dpq:
```

```
 dpq=[];

 [npq,nn]=size(dpq);  % npq=total number of considered loads

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Stress :

```
function [dSigma]=stress(du,dPar,nInc,nElements,dXY,nGtot,dCsiEtaG)

     .
     .
     .


     dt=30;
     dSigma(ne,4*ng-3:4*ng-1)=(dEmat*(dBne*dune-dPar(3)*dt*[1;1;0]))';
     dSigma(ne,4*ng)=sqrt(dSigma(ne,4*ng-3:4*ng-2)*dSigma(ne,4*ng-3:4*ng-2)'-dSigma(ne,4*ng-3:4*ng-3)*dSigma(ne,4*ng-2:4*ng-
2)+3*dSigma(ne,4*ng-1:4*ng-1)^2);
   end
  end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Assilc :

```
function [nUs,dUs,nUu,dT]=assilc(nInc,nForce,dF,nCons,dC,npq,dpq,dXY,thickness,nDofTot,nGtot,dCsiEtaG,dWG,nEl,Alfa,delT,dEmat,dpar)

    .
    .
    .


%   dBneT=dBne';

%computation of the thermal forces at nodes
 if delT ~=0

    epsT=Alfa*delT*[1;1;0];

     for i=1:1:nEl

     dtherm=zeros([8,1]);
     n14=nInc(i,1:4);
     dXnodes=dXY(n14,1);
     dYnodes=dXY(n14,2);

         for ng=1:nGtot % nodal thermal forces for one element
         dxg=dCsiEtaG(ng,1);
         dyg=dCsiEtaG(ng,2);
         dPhi=[(1-dxg)*(1-dyg); (1+dxg)*(1-dyg); (1+dxg)*(1+dyg); (1-dxg)*(1+dyg)]/4;
         dPhidCsi=[-(1-dyg);  (1-dyg); (1+dyg); -(1+dyg)]/4;
         dPhidEta=[-(1-dxg); -(1+dxg); (1+dxg);  (1-dxg)]/4;

         dQmat=dPhidCsi*dPhidEta'-dPhidEta*dPhidCsi';
         ddJ=dXnodes'*dQmat*dYnodes;

         dE=dpar(1);
         dni=dpar(2);
         dG=dE/2/(1+dni);


         dBne=zeros([3,8]);
         dBne(1,1:2:end)=-dYnodes'*dQmat;
         dBne(2,2:2:end)= dXnodes'*dQmat;
         dBne(3,1:2:end)=dBne(2,2:2:end);
         dBne(3,2:2:end)=dBne(1,1:2:end);
         dBne=dBne/ddJ;
         dBneT=dBne';


         dQmat=dPhidCsi*dPhidEta'-dPhidEta*dPhidCsi';
         ddJ=dXnodes'*dQmat*dYnodes;
```

```
        dtherm(1:2:end,1)=dtherm(1:2:end,1)+dWG(ng)*thickness*dBneT(1:2:end,:)*dEmat*epsT*abs(ddJ); %x
        dtherm(2:2:end,1)=dtherm(2:2:end,1)+dWG(ng)*thickness*dBneT(2:2:end,:)*dEmat*epsT*abs(ddJ); %y


        end
   nVne=nInc(i,5:12);
   dT(nVne,1)=dT(nVne,1)+dtherm;
   end
 end

%  dBneT=dBne';
 if npq>0
 % Distributed (uniform) loads
     .
     .
     .
 end
 % dThermic(1:end,1)=dThermic(1:1:end,1)+dWG(ng)*thickness*dBneT*dEmat*epsT*abs(ddJ);
 %         dEmat=inv([   1/dE, -dni/dE,    0;
%             -dni/dE,    1/dE,    0;
%             0,       0, 1/dG]);
```