MAESTRÍA EN INGENIERÍA ESTRUCTURAL Y DE CONSTRUCCIÓN

UNIVERSITAT POLITÉCNICA DE CATALUNYA

TRABAJO N°06: Bending of Beams

Student:

Elvis Roberto Gomez Quispe

Assignment

a) Program In Mat Lab the Timoshenko 2 Nodes Beam element with reduce integration for the shear stiffness matrix

$$\mathbf{K}_b^{(e)} = \left(\frac{EI}{l}\right)^{(e)} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$ (The point interpolation is exact for $\mathbf{K}_b^{(e)}$)

$$\mathbf{K}_s^{(e)} = \left(\frac{GA^*}{l}\right)^{(e)} \begin{bmatrix} 1 & \frac{l^{(e)}}{2} & -1 & \frac{l^{(e)}}{2} \\ \cdots & \frac{(l^{(e)})^2}{4} & -\frac{l^{(e)}}{2} & \frac{(l^{(e)})^2}{4} \\ & \cdots & 1 & -\frac{l^{(e)}}{2} \\ \text{Simetr.} & & \cdots & \frac{(l^{(e)})^2}{4} \end{bmatrix}$$ (Reduced integration)

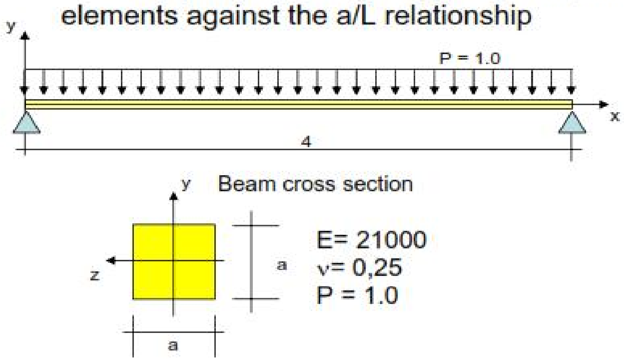
Hint: For stress evaluation make gaus1 = gaus2 = 0.0

Assignment

b) Solve the following problem with a 64 element mesh with the

2 nodes Euler Bernulli element
2 nodes Timoshenko Full Integrate element
2 nodes Timoshenko Reduce Integration element.

Compare maximum displacements, moments and shear for the 3 elements against the a/L relationship



a = 0,001
a = 0,005
a = 0,010
a = 0,020
a = 0,050
a = 0,100
a = 0,200
a = 0,400

Discuses the results observed.

```
%% 2 Nodes Beam using Timoshenko Theory with reduce integration for the shear stiffness matrix
%
% Clear memory and variables.
  clear

% The variables are readed as a MAT-fem subroutine
% young =   Young Modulus+0
% poiss =   Poission Ratio
% thick =   thickness
% denss =   density
% coordinates = [ x , y ] coordinate matrix nnode x ndime (2)
% elements    = [ inode, jnode, knode ] element conectivities matrix
%               nelem x nnode; nnode = 3
% fixdesp     = [node number, dimension, fixed value] matrix with
%               dirichlet restrictions.
% pointload   = [node number, dimension, load value] matrix with
%               nodal loads.

  file_name = input('Enter the file name :','s');

  tic;                   % Start clock
  ttim = 0;              % Initialize time counter
  eval (file_name);      % Read input file

% Finds basics dimentions
  npnod  = size(coordinates,1);        % Number of nodes
  nelem  = size(elements,1);           % Number of elements
  nnode  = size(elements,2);           % Number of nodes per element
  nndof  = npnod*2;                   % Number of total DOF

  ttim = timing('Time needed to read the input file',ttim); %Reporting time

% Dimension the global matrices.
  StifMat  = sparse ( nndof , nndof );  % Create the global stiffness matrix
  force    = sparse ( nndof , 1 );      % Create the global force vector
  Str      = zeros  ( nelem , 3 );      % Create array for streses
  u        = zeros (nndof, 1);         % Nodal variables

%  Material properties (Constant over the domain).
  dmatf = young*inercia;
  dmats = young/(2*(1+poiss))*area*5/6;

  ttim = timing('Time needed to  set initial values',ttim); %Reporting time

%  Element cycle.
  for ielem = 1 : nelem

    lnods_i = elements(ielem,1);
    lnods_j = elements(ielem,2);

    x_i = coordinates(lnods_i); % Elem. coordinates
    x_j = coordinates(lnods_j); % Elem. coordinates
    len = x_j - x_i;

```

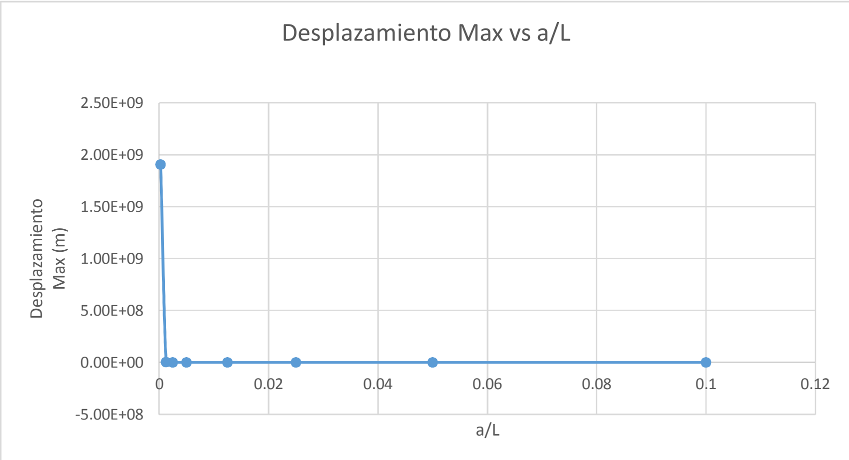
```
    const = dmatf/len;

    K_flex  = [ 0 ,  0 ,  0 ,  0 ;
                0 ,  1 ,  0 , -1 ;
                0 ,  0 ,  0 ,  0 ;
                0 , -1 ,  0 ,  1 ];

    K_flex  = K_flex * const;

    const =   dmats/len;

    K_shear = [   1    ,   len/2 ,   -1    ,   len/2 ;
                len/2 , len^2/4 , -len/2 , len^2/4 ;
                 -1    ,  -len/2 ,    1    ,  -len/2 ;
                len/2 , len^2/4 , -len/2 ,  len^2/4 ];

    K_shear = K_shear * const;

    K_elem = K_flex + K_shear;


    f       = denss*len/2;
    ElemFor = [-f, 0,-f, 0];

% Finds the equation number list for the i-th element
    eqnum(1) = (lnods_i-1)*2+1 ;     % Build the equation number list
    eqnum(2) = (lnods_i-1)*2+2 ;
    eqnum(3) = (lnods_j-1)*2+1 ;
    eqnum(4) = (lnods_j-1)*2+2 ;

% Assamble the force vector and the stiffnes matrix
    for i = 1 : 4
      ipos = eqnum(i);
      force (ipos) = force(ipos)  + ElemFor(i);
      for j = 1 : 4
         jpos = eqnum(j);
         StifMat (ipos,jpos) = StifMat (ipos,jpos) + K_elem(i,j);
      end
    end

  end  % End element cicle

  ttim = timing('Time to assamble the global system',ttim); %Reporting time

%  Add point loads conditions to the force vector
  for i = 1 : size(pointload,1)
    ieqn = (pointload(i,1)-1)*2+pointload(i,2);       % Finds eq. number
    force(ieqn) = force(ieqn) + pointload(i,3);       % add the force
  end

  ttim = timing('Time for apply side and point load',ttim); %Reporting time

 %  Applies the Dirichlet conditions and adjust the right hand side.

  j = 0;
```

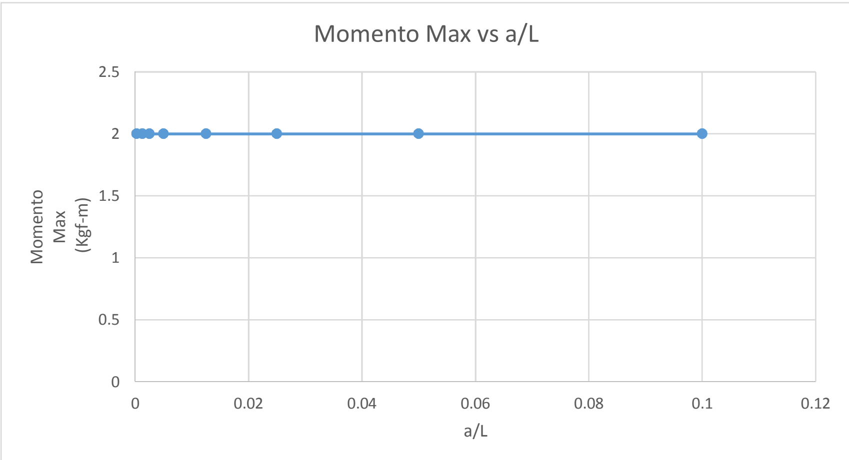
```
  for i = 1 : size(fixdesp,1)
    ieqn = (fixdesp(i,1)-1)*2+fixdesp(i,2);       % Finds eq. number
    u (ieqn) = fixdesp(i,3);                 %and store the solution in u
    j = j + 1;
    fix(j) = ieqn;                           % and mark the eq as a fix value
  end

  force  = force  - StifMat  * u;       % adjust the rhs with the known values

%  Compute the solution by solving StifMat * u = force for the
%  remaining unknown values of u.
  FreeNodes = setdiff ( 1:nndof, fix ); % Finds the free node list and

  u(FreeNodes) = StifMat(FreeNodes,FreeNodes) \ force(FreeNodes);

  ttim = timing('Time  to solve the stifness matrix',ttim); %Reporting time

 %  Compute the reactions on the fixed nodes as a R = StifMat * u - F
  reaction(fix) = StifMat(fix,1:nndof) * u(1:nndof) - force(fix);

  ttim = timing('Time to solve the  nodal reactions',ttim); %Reporting time

% Compute the stresses

%  Element cycle.
  for ielem = 1 : nelem

    lnods_i = elements(ielem,1);
    lnods_j = elements(ielem,2);

    eqnum(1) = (lnods_i-1)*2+1 ;     % Build the equation number list
    eqnum(2) = (lnods_i-1)*2+2 ;
    eqnum(3) = (lnods_j-1)*2+1 ;
    eqnum(4) = (lnods_j-1)*2+2 ;

    u_elem(1:4)=u(eqnum(1:4));

    lnods_i = elements(ielem,1);
    lnods_j = elements(ielem,2);

    x_i = coordinates(lnods_i); % Elem. coordinates
    x_j = coordinates(lnods_j); % Elem. coordinates
    len = x_j - x_i;

    gaus1 = 0;
    gaus2 = 0;

    bmat_f=[ 0, -1/len, 0, 1/len];

    bmat_s1=[-1/len,-(1-gaus1)/2, 1/len,-(1+gaus1)/2];
    bmat_s2=[-1/len,-(1-gaus2)/2, 1/len,-(1+gaus2)/2];

    Str(ielem,1) = dmatf*(bmat_f *transpose(u_elem));
    Str(ielem,2) = dmats*(bmat_s1*transpose(u_elem));
    Str(ielem,3) = dmats*(bmat_s2*transpose(u_elem));
```

```

  end

  ttim = timing('Time to  solve the  nodal stresses',ttim); %Reporting time

% Graphic representation.
  ToGiD_VigaD (file_name,u,reaction,Str);

  ttim = timing('Time  used to write  the  solution',ttim); %Reporting time
  itim = toc;                                               %Close last tic
  fprintf(1,'\n Total running time %12.6f \n',ttim);  %Reporting final time
```

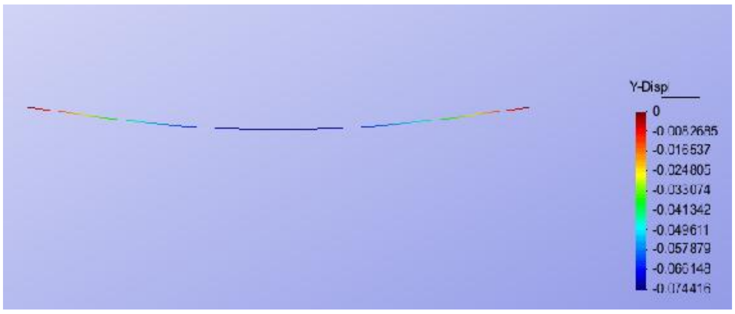
2 Nodes Euler Bernulli Element

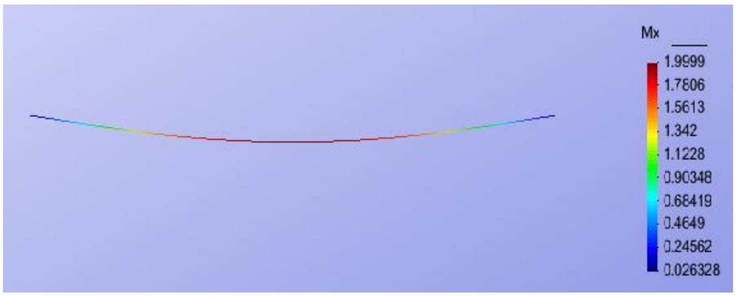
a (m)	Area (m2)	Inercia (m4)	E (kgf/m2)	L (m)	a/L	Desplazamiento Max (m)	Momento Max (Kgf-m)
0.001	0.000001	8.3333333333E-14	21000	4	0.00025	1.90E+09	1.999
0.005	0.000025	5.2083333333E-11	21000	4	0.00125	3.16E+06	1.999
0.01	0.0001	8.3333333333E-10	21000	4	0.0025	1.90E+05	1.999
0.02	0.0004	1.3333333333E-08	21000	4	0.005	1.19E+04	1.999
0.05	0.0025	5.2083333333E-07	21000	4	0.0125	3.05E+02	1.999
0.1	0.01	8.3333333333E-06	21000	4	0.025	1.90E+01	1.999
0.2	0.04	1.3333333333E-04	21000	4	0.05	1.19E+00	1.999
0.4	0.16	2.1333333333E-03	21000	4	0.1	7.44E-02	1.999





Resultados con a=0.4 :

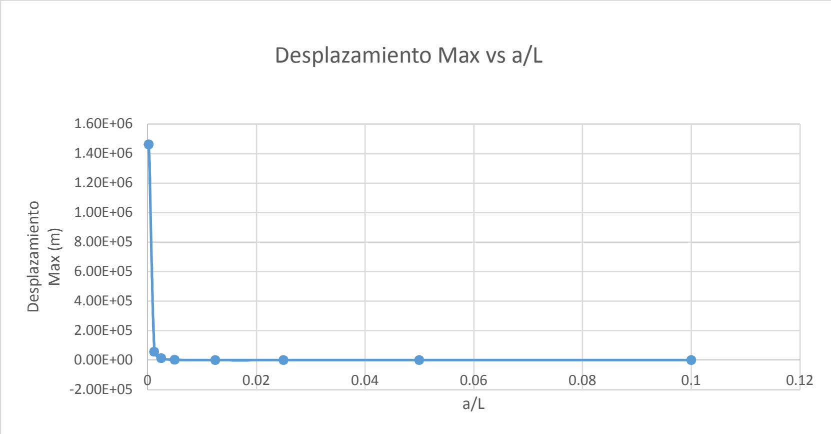


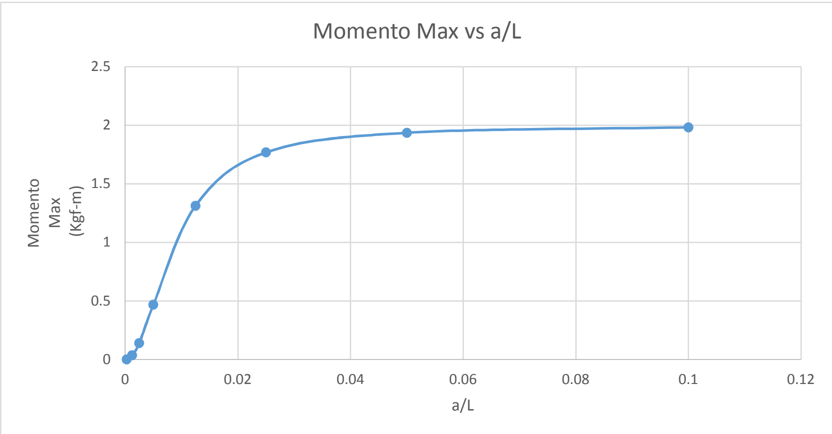


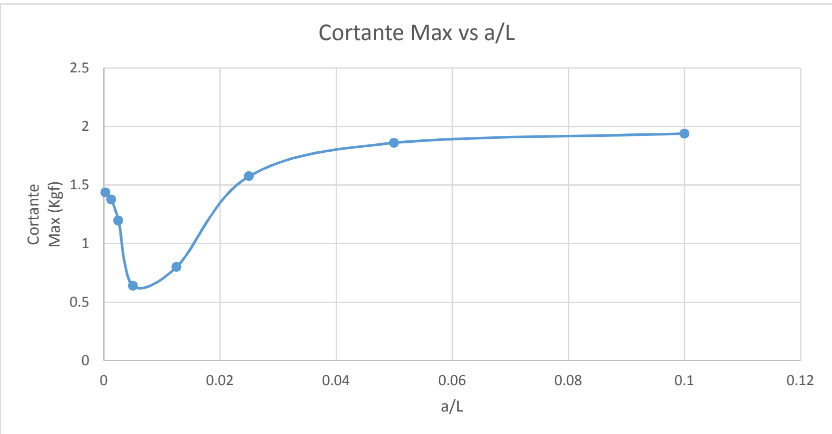
- Se observa que conforme se incrementa la seccion, es decir aumenta de rigidez de la barra, se va reduciendo la deformacion de la viga.
- Se observa que los momentos flectores, se mantienen contantes, independientemente de las modificacion de la seccion transversal.
- Conforme lo indicado en la teoria, no se conforman esfuerzos tangenciales de corte.

2 Nodes Timoshenko Full Integrate Element

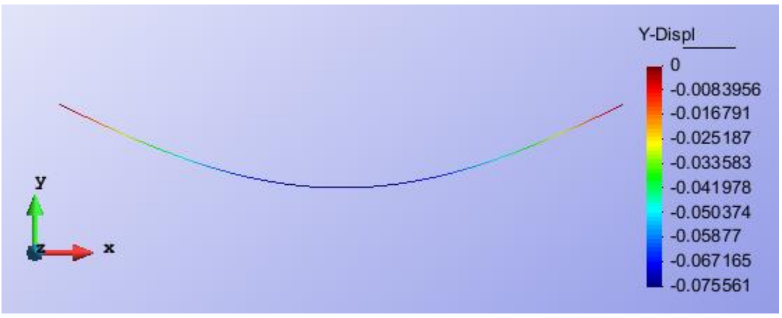
a (m)	Area (m2)	Inercia (m4)	E (kgf/m2)	L (m)	a/L	Desplazamiento Max (m)	Momento Max (Kgf-m)	Cortante Max (Kgf)
0.001	0.000001	8.3333333333E-14	21000	4	0.00025	1.46E+06	0.0015341	1.4386
0.005	0.000025	5.2083333333E-11	21000	4	0.00125	5.74E+04	0.037658	1.377
0.01	0.0001	8.3333333333E-10	21000	4	0.0025	1.36E+04	0.14257	1.198
0.02	0.0004	1.3333333333E-08	21000	4	0.005	2.80E+03	0.46978	0.63986
0.05	0.0025	5.2083333333E-07	21000	4	0.0125	2.00E+02	1.3144	0.80095
0.1	0.01	8.3333333333E-06	21000	4	0.025	1.69E+01	1.7687	1.5759
0.2	0.04	1.3333333333E-04	21000	4	0.05	1.16E+00	1.936	1.8613
0.4	0.16	2.1333333333E-03	21000	4	0.1	7.56E-02	1.9829	1.9412

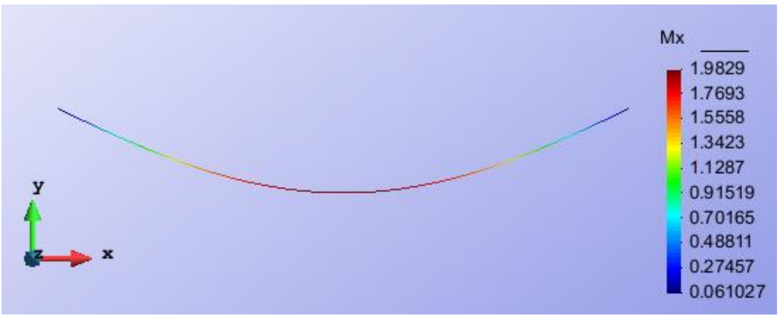


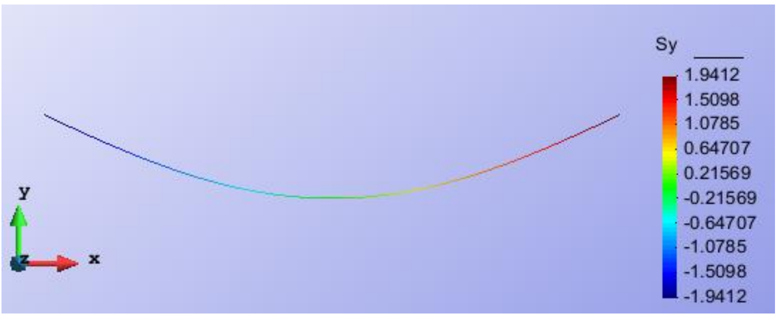




Resultados con a=0.4 :



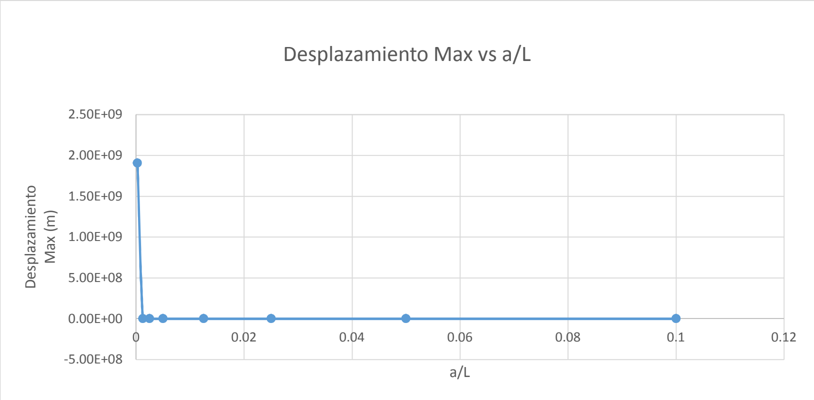


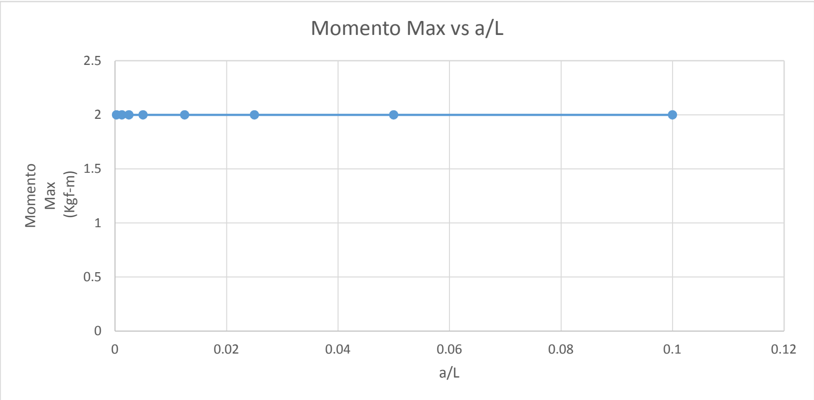


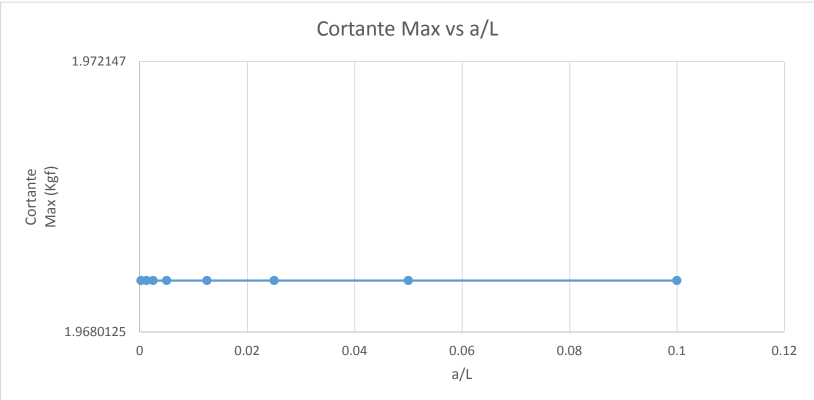
- Se observa que conforme se incrementa la seccion, es decir aumenta de rigidez de la barra, se va reduciendo la deformacion de la viga.
- Los valores obtenidos van aproximandose a los resultados obtenidos con los otros metodos, conforme aumenta la seccion de la viga.
- Se observa que respecto a los momentos flectores, a diferencia del Metodo de Euler-.Bernoulli, estos valores no se mantienen contantes, apreciandose que la precision del momento maximo mejora, conforme aumenta la seccion de la viga.
- Se observa que respecto a las cortantes, a diferencia del Metodo de Euler-.Bernoulli, estos valores se presentan, apreciandose que la precision del momento maximo mejora, conforme se aumenta la seccion de la viga.

2 Nodes Timoshenko Reduce Integration Element

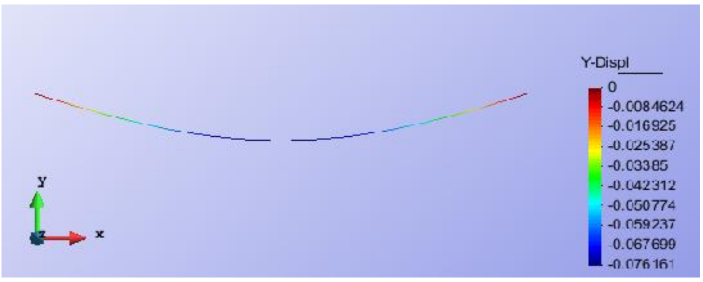
a (m)	Area (m2)	Inercia (m4)	E (kgf/m2)	L (m)	a/L	Desplazamiento Max (m)	Momento Max (Kgf-m)	Cortante Max (Kgf)
0.001	0.000001	8.3333333333E-14	21000	4	0.00025	1.90E+09	1.999	1.9688
0.005	0.000025	5.2083333333E-11	21000	4	0.00125	3.05E+06	1.999	1.9688
0.01	0.0001	8.3333333333E-10	21000	4	0.0025	1.90E+05	1.999	1.9688
0.02	0.0004	1.3333333333E-08	21000	4	0.005	1.19E+04	1.999	1.9688
0.05	0.0025	5.2083333333E-07	21000	4	0.0125	3.05E+02	1.999	1.9688
0.1	0.01	8.3333333333E-06	21000	4	0.025	1.91E+01	1.999	1.9688
0.2	0.04	1.3333333333E-04	21000	4	0.05	1.20E+00	1.999	1.9688
0.4	0.16	2.1333333333E-03	21000	4	0.1	7.62E-02	1.999	1.9688







Resultados con a=0.4 :



- Se observa que conforme se incrementa la seccion, es decir aumenta de rigidez de la barra, se va reduciendo la deformacion de la viga.
- Los valorse son muy similares a los obtenidos con el metodo Euler-Bernoulli.
- Se observa que respecto a los momentos flectores, a diferencia del Metodo de Timoshenko sin Modificar, estos valores se mantienen contantes, conforme se aumenta la seccion de la viga.
- Se observa que respecto a las cortantes, a diferencia del Metodo de Timoshenko sin Modificar , se aprecia que los valores de