Contents

1 Elastic Beam Element Under Static Loading

1.1 Simulate the cantilever by 1 elastic beam element

This is a simple beam example under static loading in three directions. The diagram below shows the loading in one bending direction.

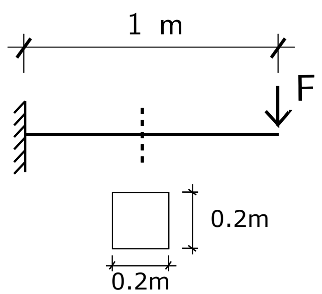


Figure 1: The cantilever model

ESSI model fei file:

```
model name "beam_1element" ;
// define the node coordinates
add node # 1 at (  0.0*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 2 at (  1.0*m ,   0.0*m,    0.0*m) with 6 dofs;
// Geometry: width and height. Help the beam definition.
b=0.2*m;
h=0.2*m;
I=b*h^3/12.0;
// define the beam element
add element # 1 type beam_elastic with nodes (1,2)
  cross_section = b*h
  elastic_modulus = 1e9*N/m^2
  shear_modulus = 5e8*N/m^2
  torsion_Jx = 0.33*b*h^3
  bending_Iy = I
  bending_Iz = I
  mass_density = 0*kg/m^3
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
// add boundary condition
fix node #    1 dofs all;
// axial loading
new loading stage "axial";
add load # 1 to node # 2 type linear Fx = 1*N;
define load factor increment 1;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 1 steps using static algorithm;
// bending in one direction
new loading stage "bending1";
remove load # 1;
add load # 2 to node # 2 type linear Fy = 1*N;
```

```
define load factor increment 1;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 1 steps using static algorithm;
// bending in the other direction
new loading stage "bending2";
remove load # 2;
add load # 3 to node # 2 type linear Fz = 1*N;
define load factor increment 1;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

2 Elastic Beam Element under Dynamic Loading

2.1 Cantilever, One Elastic Beam Element

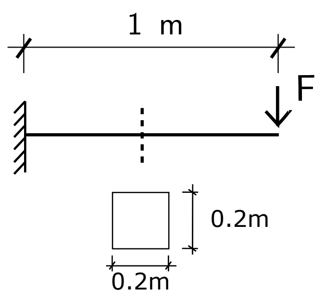


Figure 2: The cantilever model.

Problem description:

ESSI model fei file:

```
model name "beam_1element" ;

// add node
add node # 1 at (  0.0*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 2 at (  1.0*m ,   0.0*m,    0.0*m) with 6 dofs;
  // Geometry: width and height
b=0.2*m;
h=0.2*m;
// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
L=1*m;
rho  = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;
// add elements
add element # 1 type beam_elastic with nodes (1,2)
  cross_section = b*h
  elastic_modulus = elastic_constant
  shear_modulus = elastic_constant/2/(1+possion_ratio)
  torsion_Jx = 0.33*b*h^3
  bending_Iy = b*h^3/12
  bending_Iz = b*h^3/12
  mass_density = rho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);

// add boundary condition
```

```
fix node #    1 dofs all;

// // ---------------------------------------------------------------------------
// // --slowLoading--------------------------------------------------------------
// // add load in 180 seconds. (Slow)
// // ---------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "slowLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // ---------------------------------------------------------------------------
// // --fastLoading--------------------------------------------------------------
// // add load in 0.6 seconds (Fast)
// // ---------------------------------------------------------------------------
// remove load # 1;
// new loading stage "fastLoading";
// add load # 2 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "fastLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // ---------------------------------------------------------------------------
// // --freeVibration------------------------------------------------------------
// // add a load and then release to free vibration
// // ---------------------------------------------------------------------------
// remove load # 2;
new loading stage "freeVibration";
add load # 3 to node # 2 type path_time_series
  Fz =  1.*N
  series_file = "freeVibration.txt" ;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 2000 steps using transient algorithm
  time_step = 0.01*s;

bye;
```

Displacement Results The ESSI model fei files for this example can be downloaded here.

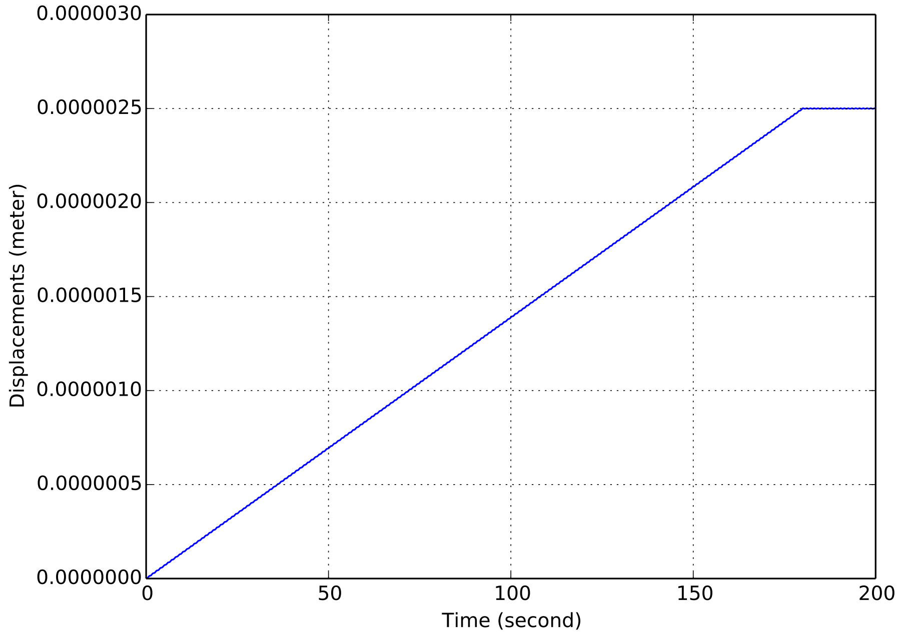


Figure 3: Slow loading condition, vertical displacements or the cantilever tip.

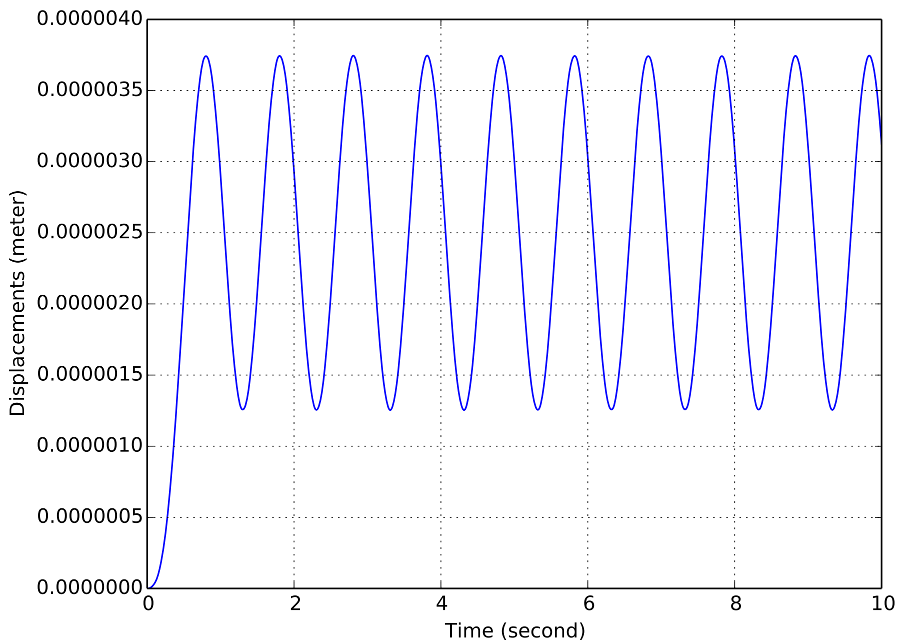


Figure 4: Fast loading condition, vertical displacements of the cantilever tip.

2.2 Cantilever, 5 Elastic Beam Elements

Problem description:

ESSI model fei file:

```
model name "beam_5element" ;
```

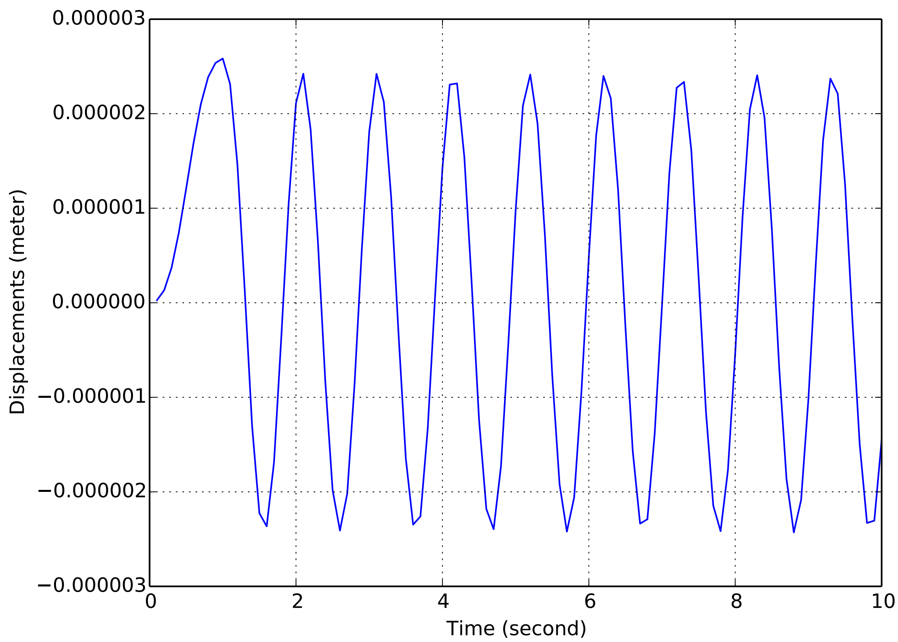


Figure 5: Free vibration, vertical displacements of the cantilever tip.

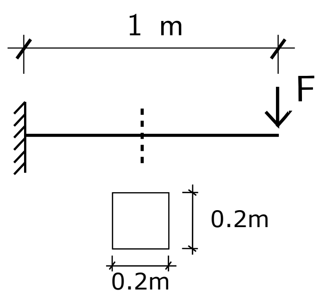


Figure 6: The cantilever model.

```

// add node
add node # 1 at (  0.0*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 2 at (  0.2*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 3 at (  0.4*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 4 at (  0.6*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 5 at (  0.8*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 6 at (  1.0*m ,   0.0*m,    0.0*m) with 6 dofs;

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
```

```
A=b*h;
L=1*m;
rho   = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;

// Cross section geometry: width and height
b=0.2*m;
h=0.2*m;

// add elements
ii=1;
while (ii<6) {
  add element # ii type beam_elastic with nodes (ii,ii+1)
    cross_section = b*h
    elastic_modulus = elastic_constant
    shear_modulus = elastic_constant/2/(1+possion_ratio)
    torsion_Jx = 0.33*b*h^3
    bending_Iy = b*h^3/12
    bending_Iz = b*h^3/12
    mass_density = rho
    xz_plane_vector = ( 1, 0, 1)
    joint_1_offset = (0*m, 0*m, 0*m)
    joint_2_offset = (0*m, 0*m, 0*m);
  ii+=1;
}

// add boundary condition
fix node #    1 dofs all;

// // ----------------------------------------------------------------------------
// // --slowLoading---------------------------------------------------------------
// // add load in 180 seconds.
// // ----------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # 6 type path_time_series
//  Fz = 1.*N
//  series_file = "slowLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // ----------------------------------------------------------------------------
// // --fastLoading---------------------------------------------------------------
// // add load in 0.6 seconds.
// // ----------------------------------------------------------------------------
// remove load # 1;
// new loading stage "fastLoading";
// add load # 2 to node # 6 type path_time_series
//  Fz = 1.*N
//  series_file = "fastLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
```

```
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // -----------------------------------------------------------------------
// // --freeVibration---------------------------------------------------------
// // add a load and then release for free vibration
// // -----------------------------------------------------------------------
// remove load # 2;
new loading stage "freeVibration";
add load # 3 to node # 6 type path_time_series
  Fz =  1.*N
  series_file = "freeVibration.txt" ;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 100 steps using transient algorithm
  time_step = 0.1*s;

bye;
```

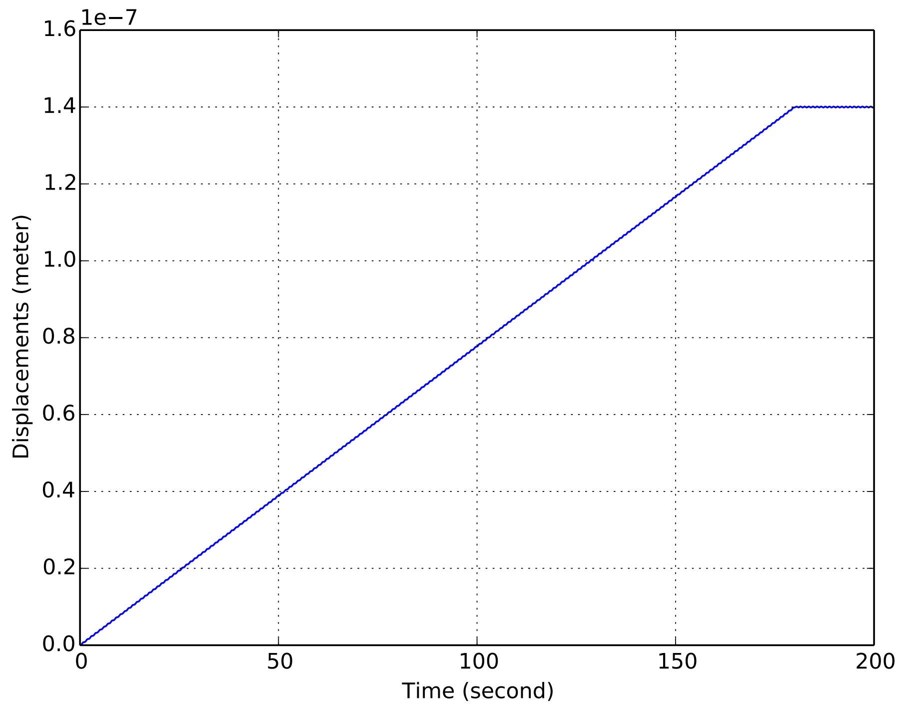


Figure 7: Slow loading condition, vertical displacements of the cantilever tip.

Displacement results The ESSI model fei files for this example can be downloaded here.

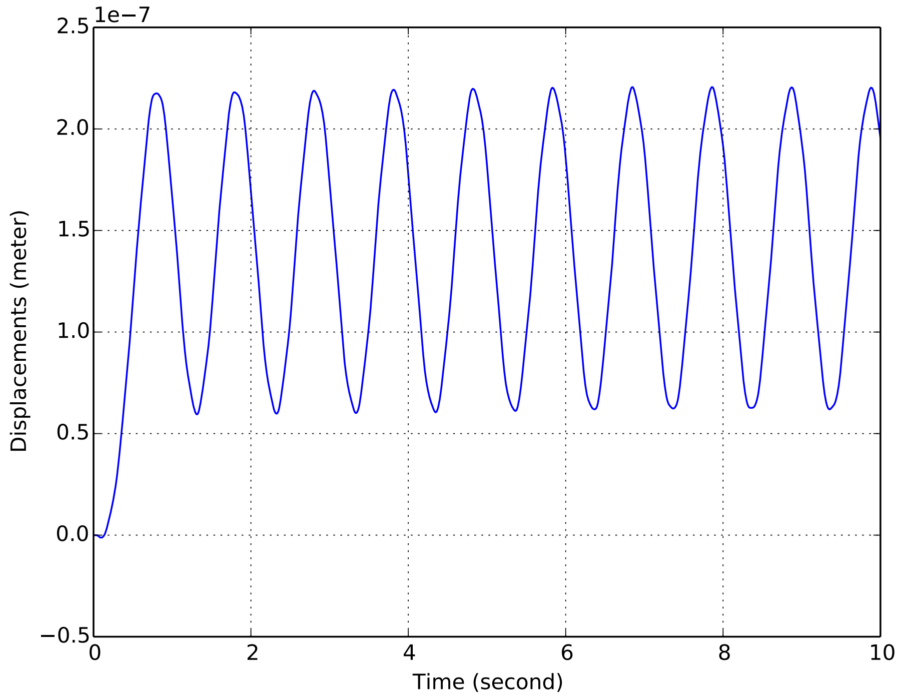


Figure 8: Fast loading condition, vertical displacements of the cantilever tip.

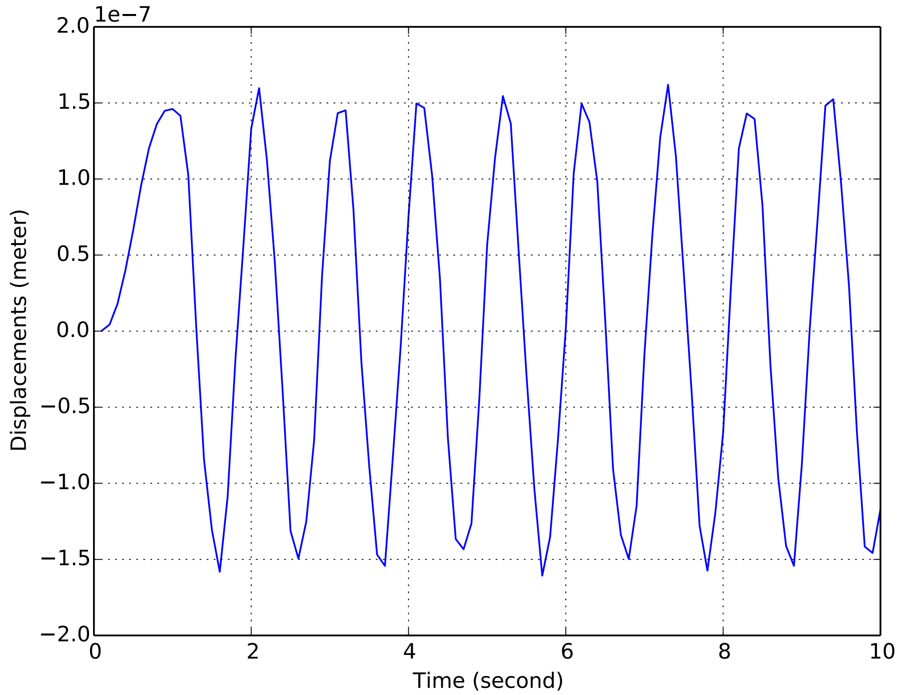


Figure 9: Free vibration condition, vertical displacements of the cantilever tip.

3 Cantilever, one 27 Node Brick Element, Dynamic Loading

3.1 Simulate the cantilever using one 27NodeBrick element

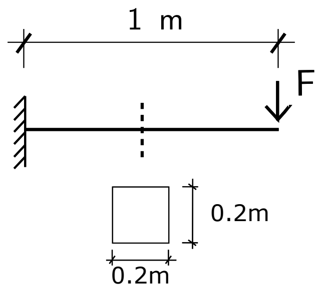


Figure 10: The cantilever model.

Problem description:

ESSI model fei file:

```
model name "brick_1element" ;

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
L=1*m;
rho  = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;


add material # 1 type linear_elastic_isotropic_3d_LT
  mass_density = rho
  elastic_modulus = elastic_constant
  poisson_ratio = possion_ratio;

add node #      1 at (  0.0000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #      2 at (  0.0000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
add node #      3 at (  1.0000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #      4 at (  1.0000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
add node #      5 at (  0.0000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      6 at (  1.0000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      7 at (  1.0000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      8 at (  0.0000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      9 at (  0.0000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
add node #     10 at (  0.5000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #     11 at (  1.0000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
```

```
add node #      12 at (  0.5000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
add node #      13 at (  0.0000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;
add node #      14 at (  0.5000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      15 at (  1.0000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;
add node #      16 at (  0.5000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      17 at (  0.0000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      18 at (  0.0000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      19 at (  1.0000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      20 at (  1.0000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      21 at (  0.5000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      22 at (  0.0000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      23 at (  0.5000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      24 at (  1.0000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      25 at (  0.5000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      26 at (  0.5000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
add node #      27 at (  0.5000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;

add element #       1 type 27NodeBrickLT with nodes(    2,       1,       3,       4,       5,
         8,       7,       6,       9,      10,      11,      12,      13,      14,      15,      16,
        17,      18,      19,      20,      21,      22,      23,      24,      25,      26,      27)
    use material #      1;

fix node # 1 dofs all;
fix node # 2 dofs all;
fix node # 5 dofs all;
fix node # 8 dofs all;
fix node # 9 dofs all;
fix node # 13 dofs all;
fix node # 17 dofs all;
fix node # 18 dofs all;
fix node # 22 dofs all;


// // -------------------------------------------------------------------------------
// // --slowLoading-------------------------------------------------------------------
// // -------------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # 4 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 2 to node # 6 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 3 to node # 3 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 4 to node # 7 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 5 to node # 20 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 6 to node # 11 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 7 to node # 15 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 8 to node # 19 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 9 to node # 24 type path_time_series Fz=4/9.0*N series_file = "slowLoading.txt"
    ;
```

```
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // ---------------------------------------------------------------------------
// // --fastLoading--------------------------------------------------------------
// // ---------------------------------------------------------------------------
// new loading stage "fastLoading";
// add load # 101 to node # 4 type path_time_series Fz=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 102 to node # 6 type path_time_series Fz=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 103 to node # 3 type path_time_series Fz=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 104 to node # 7 type path_time_series Fz=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 105 to node # 20 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 106 to node # 11 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 107 to node # 15 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 108 to node # 19 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 109 to node # 24 type path_time_series Fz=4/9.0*N series_file =
    "fastLoading.txt" ;
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // ---------------------------------------------------------------------------
// // --freeVibration-------------------------------------------------------------
// // ---------------------------------------------------------------------------
new loading stage "freeVibration";
add load # 201 to node # 4 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 202 to node # 6 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 203 to node # 3 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 204 to node # 7 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 205 to node # 20 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 206 to node # 11 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 207 to node # 15 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 208 to node # 19 type path_time_series Fz=1/9.0*N series_file =
```

```
    "freeVibration.txt" ;
add load # 209 to node # 24 type path_time_series Fz=4/9.0*N series_file =
    "freeVibration.txt" ;
// add algorithm and solver
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 10000 steps using transient algorithm
  time_step = 0.001*s;

// end
bye;
```

Displacement results against time series

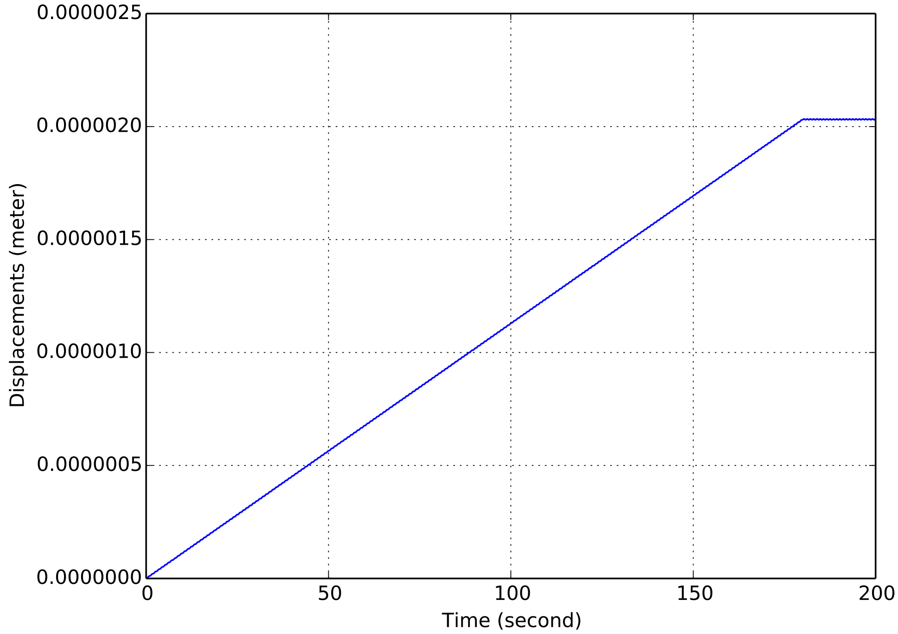


Figure 11: Slow loading condition, vertical displacements of the cantilever tip.

The ESSI model fei files for this example can be downloaded here.

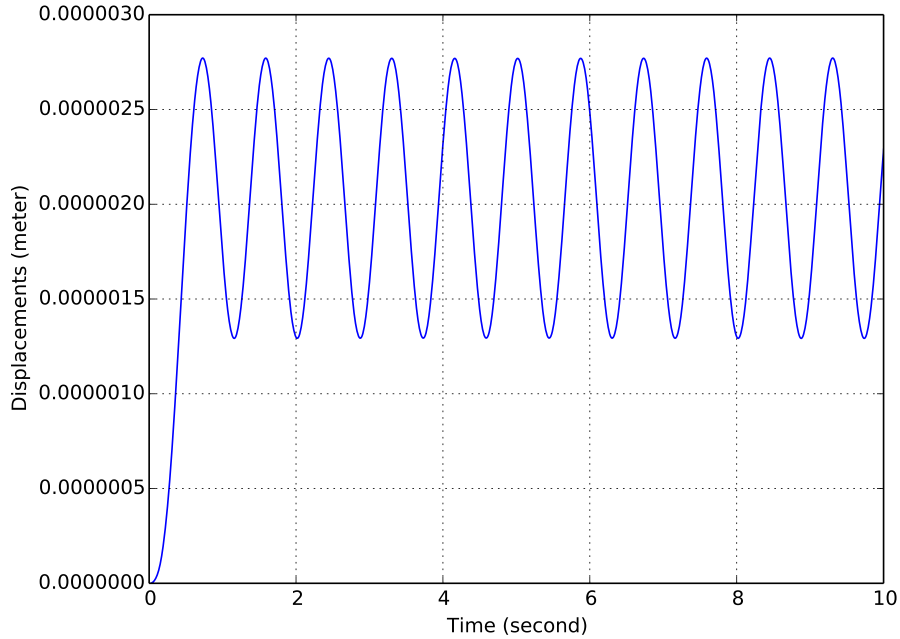


Figure 12: Fast loading condition, vertical displacements of the cantilever tip.

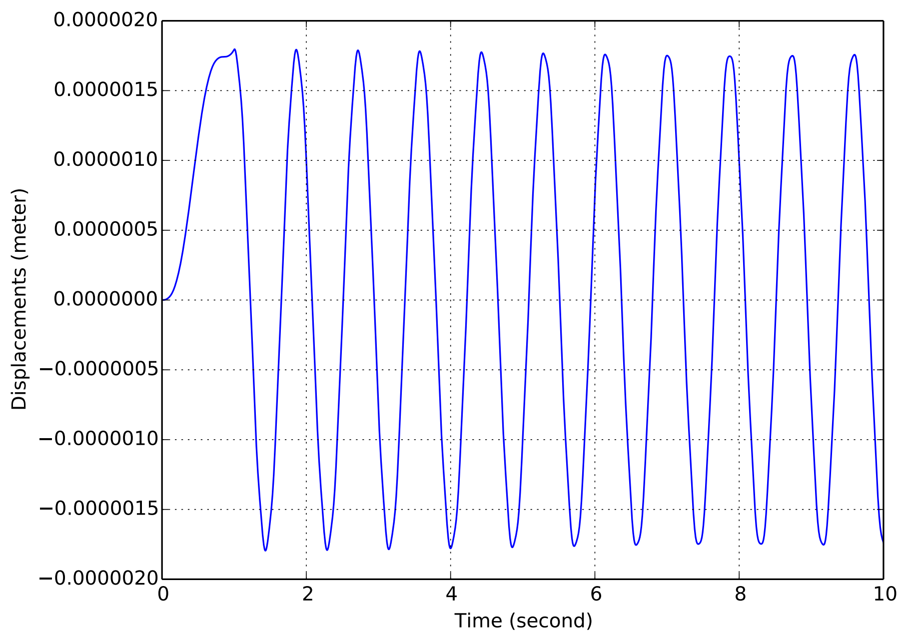


Figure 13: Free vibration condition, vertical displacements of the cantilever tip.

3.2 Simulate Cantilever Using five 27 Node Brick Elements

Problem description:

ESSI model fei file:

```
model name "brick_5element" ;
```

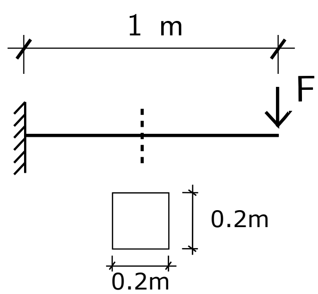


Figure 14: The cantilever model.

```

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
L=1*m;
rho  = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;


add material # 1 type linear_elastic_isotropic_3d_LT
  mass_density = rho
  elastic_modulus = elastic_constant
  poisson_ratio = possion_ratio;

add node # 1 at (0.0*m, 0.0*m , 0.0*m) with 3 dofs;
add node # 2 at (0.1*m, 0.0*m , 0.0*m) with 3 dofs;
add node # 3 at (0.2*m, 0.0*m , 0.0*m) with 3 dofs;
add node # 4 at (0.0*m, 0.1*m , 0.0*m) with 3 dofs;
add node # 5 at (0.1*m, 0.1*m , 0.0*m) with 3 dofs;
add node # 6 at (0.2*m, 0.1*m , 0.0*m) with 3 dofs;
add node # 7 at (0.0*m, 0.2*m , 0.0*m) with 3 dofs;
add node # 8 at (0.1*m, 0.2*m , 0.0*m) with 3 dofs;
add node # 9 at (0.2*m, 0.2*m , 0.0*m) with 3 dofs;

fix node No 1 dofs ux uy uz;
fix node No 2 dofs ux uy uz;
fix node No 3 dofs ux uy uz;
fix node No 4 dofs ux uy uz;
fix node No 5 dofs ux uy uz;
fix node No 6 dofs ux uy uz;
fix node No 7 dofs ux uy uz;
fix node No 8 dofs ux uy uz;
fix node No 9 dofs ux uy uz;
e = 0;
```

```
hh = 0*m;
NBricks=5;
dz = 0.2*m;
while ( e < NBricks)
{
  hh += dz;
  add node # 10+18*e at (0.0*m, 0.0*m , hh - 0.5*dz) with 3 dofs;
  add node # 11+18*e at (0.1*m, 0.0*m , hh - 0.5*dz) with 3 dofs;
  add node # 12+18*e at (0.2*m, 0.0*m , hh - 0.5*dz) with 3 dofs;
  add node # 13+18*e at (0.0*m, 0.1*m , hh - 0.5*dz) with 3 dofs;
  add node # 14+18*e at (0.1*m, 0.1*m , hh - 0.5*dz) with 3 dofs;
  add node # 15+18*e at (0.2*m, 0.1*m , hh - 0.5*dz) with 3 dofs;
  add node # 16+18*e at (0.0*m, 0.2*m , hh - 0.5*dz) with 3 dofs;
  add node # 17+18*e at (0.1*m, 0.2*m , hh - 0.5*dz) with 3 dofs;
  add node # 18+18*e at (0.2*m, 0.2*m , hh - 0.5*dz) with 3 dofs;

  add node # 19+18*e at (0.0*m, 0.0*m , hh) with 3 dofs;
  add node # 20+18*e at (0.1*m, 0.0*m , hh) with 3 dofs;
  add node # 21+18*e at (0.2*m, 0.0*m , hh) with 3 dofs;
  add node # 22+18*e at (0.0*m, 0.1*m , hh) with 3 dofs;
  add node # 23+18*e at (0.1*m, 0.1*m , hh) with 3 dofs;
  add node # 24+18*e at (0.2*m, 0.1*m , hh) with 3 dofs;
  add node # 25+18*e at (0.0*m, 0.2*m , hh) with 3 dofs;
  add node # 26+18*e at (0.1*m, 0.2*m , hh) with 3 dofs;
  add node # 27+18*e at (0.2*m, 0.2*m , hh) with 3 dofs;

  add element # e+1 type 27NodeBrickLT with nodes
    (
      21+18*e,
      27+18*e,
      25+18*e,
      19+18*e,

       3+18*e,
       9+18*e,
       7+18*e,
       1+18*e,

      24+18*e,
      26+18*e,
      22+18*e,
      20+18*e,

       6+18*e,
       8+18*e,
       4+18*e,
       2+18*e,

      12+18*e,
      18+18*e,
      16+18*e,
      10+18*e,

      14+18*e,
      15+18*e,
```

```
      17+18*e,
      13+18*e,
      11+18*e,
      23+18*e,
       5+18*e
    )
    use material # 1;

  e += 1;
};


e = e -1;


// // ------------------------------------------------------------------------------
// // --slowLoading-------------------------------------------------------------------
// // add the 1 Newton load in 180 seconds.
// // ------------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # (19+18*e) type path_time_series Fx=1/36.0*N series_file =
    "slowLoading.txt";
// add load # 2 to node # (20+18*e) type path_time_series Fx=1/9.0*N series_file =
    "slowLoading.txt";
// add load # 3 to node # (21+18*e) type path_time_series Fx=1/36.0*N series_file =
    "slowLoading.txt";
// add load # 4 to node # (22+18*e) type path_time_series Fx=1/9.0*N series_file =
    "slowLoading.txt";
// add load # 5 to node # (23+18*e) type path_time_series Fx=4/9.0*N series_file =
    "slowLoading.txt";
// add load # 6 to node # (24+18*e) type path_time_series Fx=1/9.0*N series_file =
    "slowLoading.txt";
// add load # 7 to node # (25+18*e) type path_time_series Fx=1/36.0*N series_file =
    "slowLoading.txt";
// add load # 8 to node # (26+18*e) type path_time_series Fx=1/9.0*N series_file =
    "slowLoading.txt";
// add load # 9 to node # (27+18*e) type path_time_series Fx=1/36.0*N series_file =
    "slowLoading.txt";
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // ------------------------------------------------------------------------------
// // --fastLoading-------------------------------------------------------------------
// // add the 1 Newton load in 0.6 seconds.
// // ------------------------------------------------------------------------------
// new loading stage "fastLoading";
// add load # 101 to node # (19+18*e) type path_time_series Fx=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 102 to node # (20+18*e) type path_time_series Fx=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 103 to node # (21+18*e) type path_time_series Fx=1/36.0*N series_file =
```

```
    "fastLoading.txt" ;
// add load # 104 to node # (22+18*e) type path_time_series Fx=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 105 to node # (23+18*e) type path_time_series Fx=4/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 106 to node # (24+18*e) type path_time_series Fx=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 107 to node # (25+18*e) type path_time_series Fx=1/36.0*N series_file =
    "fastLoading.txt" ;
// add load # 108 to node # (26+18*e) type path_time_series Fx=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 109 to node # (27+18*e) type path_time_series Fx=1/36.0*N series_file =
    "fastLoading.txt" ;
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // ---------------------------------------------------------------------------
// // --freeVibration-------------------------------------------------------------
// // add a load and then release to free vibration
// // ---------------------------------------------------------------------------
new loading stage "freeVibration";
add load # 201 to node # (19+18*e) type path_time_series Fx=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 202 to node # (20+18*e) type path_time_series Fx=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 203 to node # (21+18*e) type path_time_series Fx=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 204 to node # (22+18*e) type path_time_series Fx=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 205 to node # (23+18*e) type path_time_series Fx=4/9.0*N series_file =
    "freeVibration.txt" ;
add load # 206 to node # (24+18*e) type path_time_series Fx=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 207 to node # (25+18*e) type path_time_series Fx=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 208 to node # (26+18*e) type path_time_series Fx=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 209 to node # (27+18*e) type path_time_series Fx=1/36.0*N series_file =
    "freeVibration.txt" ;
// add algorithm and solver
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 100 steps using transient algorithm
  time_step = 0.1*s;

// end
bye;
```

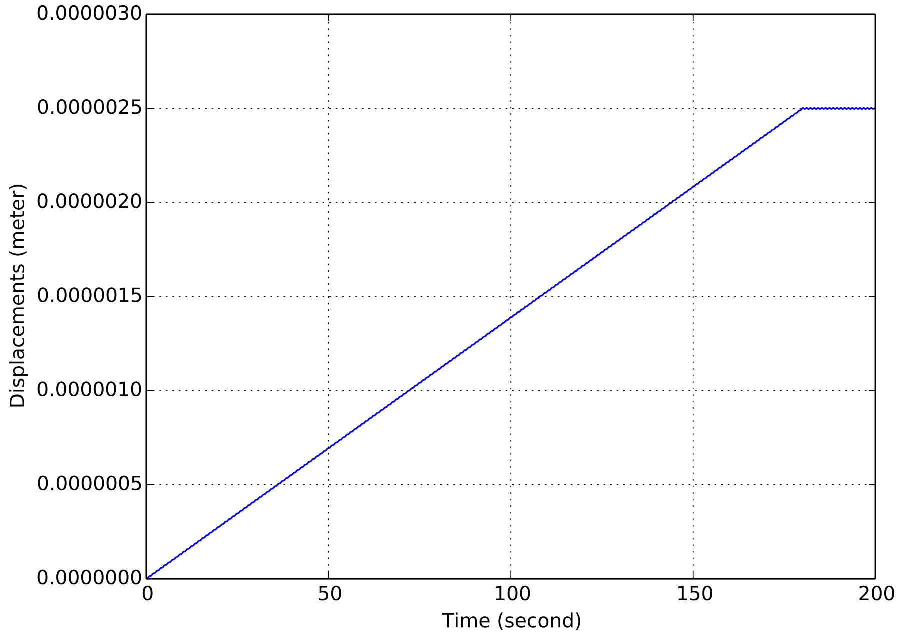


Figure 15: Slow loading condition, vertical displacements of the cantilever tip.

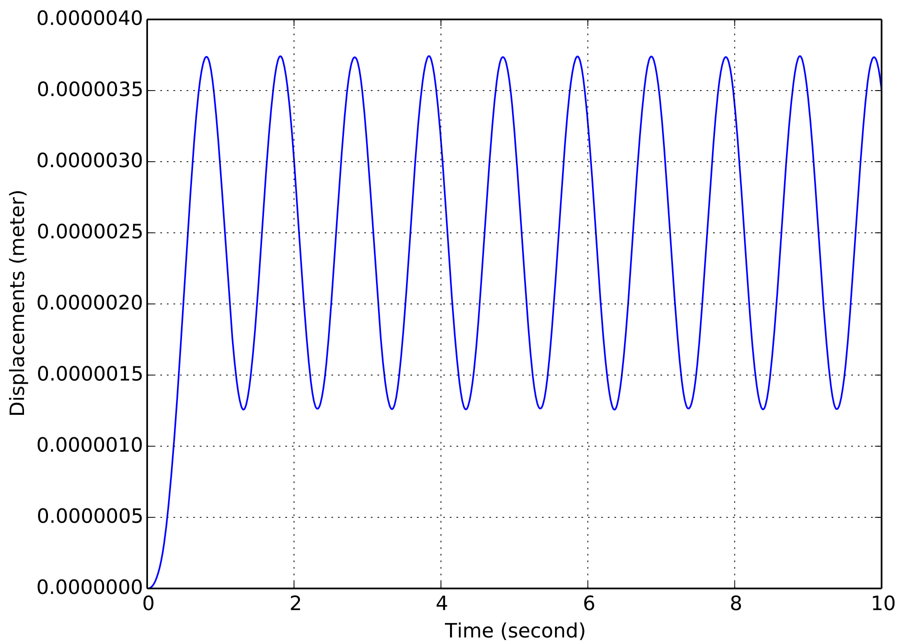


Figure 16: Fast loading condition, vertical displacements of the cantilever tip.

Displacement Results. The ESSI model fei files for this example can be downloaded here.

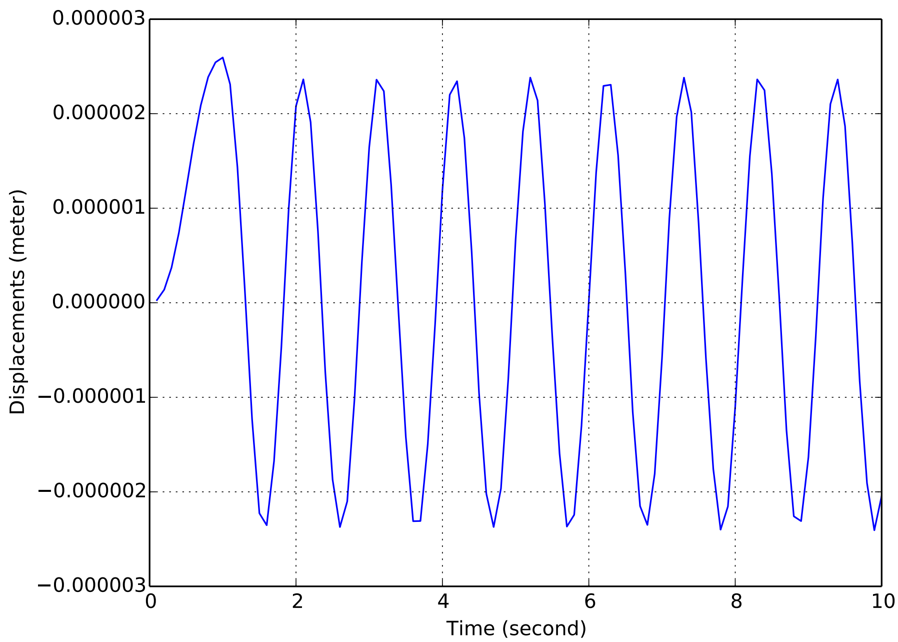


Figure 17: Free vibration condition, vertical displacements of the cantilever tip.

4 Elastic Beam Element under Dynamic Loading with concentrated mass

4.1 Simulate Cantilever Using 1 Elastic Beam Element

Problem description:

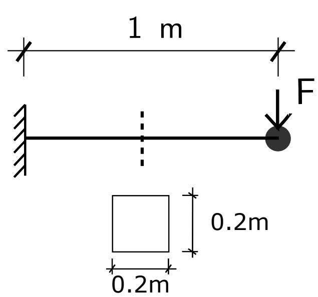


Figure 18: The cantilever-mass model.

ESSI model fei file:

```
model name "beam-mass_1element" ;

// add node
add node # 1 at (  0.0*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 2 at (  1.0*m ,   0.0*m,    0.0*m) with 6 dofs;
```

```

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
L=1*m;
rho  = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;

// add elements
add element # 1 type beam_elastic with nodes (1,2)
  cross_section = b*h
  elastic_modulus = elastic_constant
  shear_modulus = elastic_constant/2/(1+possion_ratio)
  torsion_Jx = 0.33*b*h^3
  bending_Iy = b*h^3/12
  bending_Iz = b*h^3/12
  mass_density = rho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);

// add boundary condition
fix node #    1 dofs all;

// add mass
beamMass=rho*A*L;
add mass to node # 2
  mx = beamMass
  my = beamMass
  mz = beamMass
  Imx = 0*beamMass*L^2
  Imy = 0*beamMass*L^2
  Imz = 0*beamMass*L^2;

// // ------------------------------------------------------------------------------
// // --slowLoading-------------------------------------------------------------------
// // ------------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "slowLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // ------------------------------------------------------------------------------
```

```
// // --fastLoading-------------------------------------------------------------
// // --------------------------------------------------------------------------
// remove load # 1;
// new loading stage "fastLoading";
// add load # 2 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "fastLoading.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // --------------------------------------------------------------------------
// // --freeVibration-----------------------------------------------------------
// // --------------------------------------------------------------------------
// remove load # 2;
new loading stage "freeVibration";
add load # 3 to node # 2 type path_time_series
  Fz =  1.*N
  series_file = "freeVibration.txt" ;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 1000 steps using transient algorithm
  time_step = 0.01*s;

bye;
```

Displacement results against time series

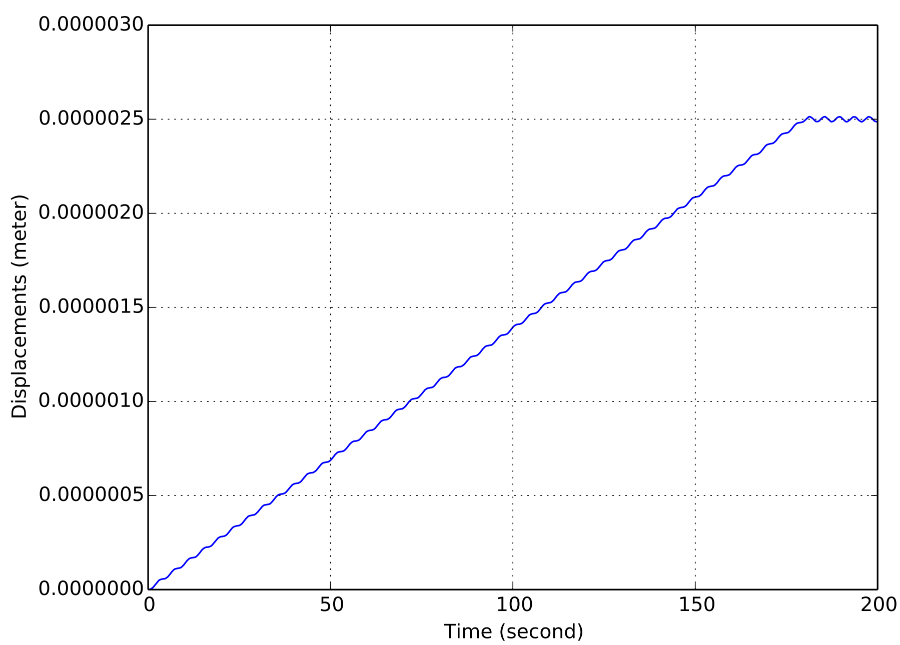


Figure 19: Slow loading condition, vertical displacements of the cantilever tip.

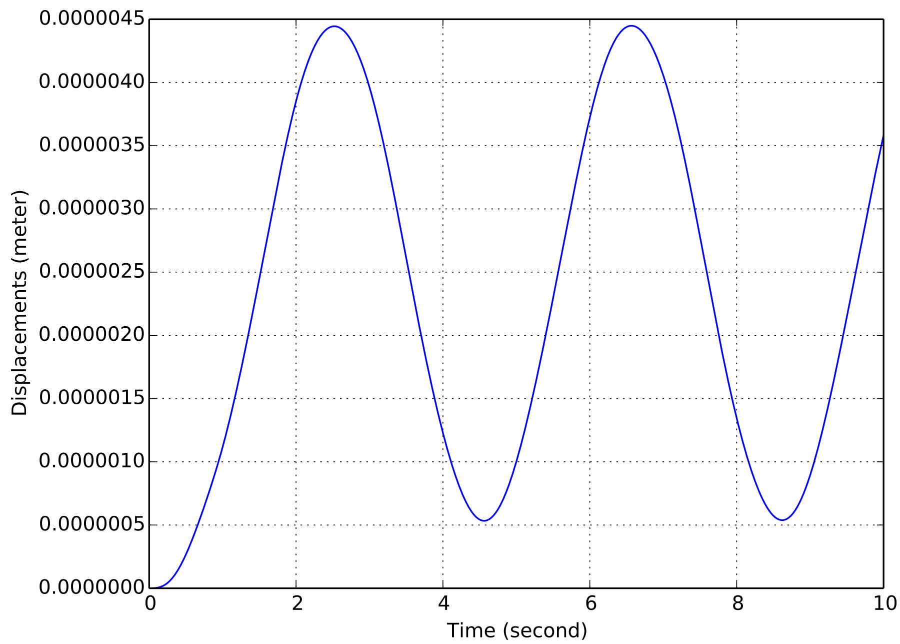


Figure 20: Fast loading condition, vertical displacements of the cantilever tip.

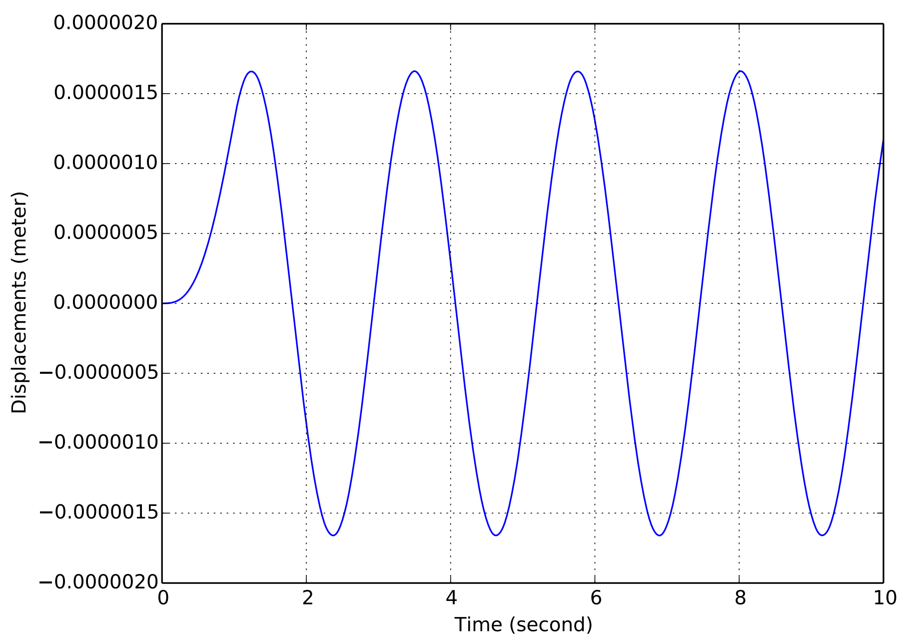


Figure 21: Free vibration condition, vertical displacements of the cantilever tip.

The ESSI model fei files for this example can be downloaded here.

5 Elastic Beam, 27 Node Brick Model With Concentrated Mass

5.1 Simulate the cantilever by one 27NodeBrick element

Problem description:

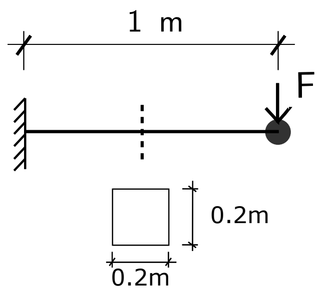


Figure 22: The cantilever-mass model.

ESSI model fei file:

```
model name "brick-mass_1element" ;

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
L=1*m;
rho  = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;

add material # 1 type linear_elastic_isotropic_3d_LT
  mass_density = rho
  elastic_modulus = elastic_constant
  poisson_ratio = possion_ratio;

add node #      1 at (  0.0000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #      2 at (  0.0000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
add node #      3 at (  1.0000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #      4 at (  1.0000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
add node #      5 at (  0.0000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      6 at (  1.0000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      7 at (  1.0000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      8 at (  0.0000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      9 at (  0.0000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
add node #     10 at (  0.5000 *m,  0.2000 *m, 0.0000 *m) with 3 dofs;
add node #     11 at (  1.0000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
add node #     12 at (  0.5000 *m,  0.0000 *m, 0.0000 *m) with 3 dofs;
```

```
add node #      13 at (  0.0000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;
add node #      14 at (  0.5000 *m,  0.2000 *m, 0.2000 *m) with 3 dofs;
add node #      15 at (  1.0000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;
add node #      16 at (  0.5000 *m,  0.0000 *m, 0.2000 *m) with 3 dofs;
add node #      17 at (  0.0000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      18 at (  0.0000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      19 at (  1.0000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      20 at (  1.0000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      21 at (  0.5000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      22 at (  0.0000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      23 at (  0.5000 *m,  0.2000 *m, 0.1000 *m) with 3 dofs;
add node #      24 at (  1.0000 *m,  0.1000 *m, 0.1000 *m) with 3 dofs;
add node #      25 at (  0.5000 *m,  0.0000 *m, 0.1000 *m) with 3 dofs;
add node #      26 at (  0.5000 *m,  0.1000 *m, 0.0000 *m) with 3 dofs;
add node #      27 at (  0.5000 *m,  0.1000 *m, 0.2000 *m) with 3 dofs;

add element #       1 type 27NodeBrickLT with nodes(    2,       1,       3,       4,       5,
         8,       7,       6,       9,      10,      11,      12,      13,      14,      15,      16,
        17,      18,      19,      20,      21,      22,      23,      24,      25,      26,      27)
    use material #      1;

fix node # 1 dofs all;
fix node # 2 dofs all;
fix node # 5 dofs all;
fix node # 8 dofs all;
fix node # 9 dofs all;
fix node # 13 dofs all;
fix node # 17 dofs all;
fix node # 18 dofs all;
fix node # 22 dofs all;


// Mapping from 3 dofs to 6 dofs.
add node #     1003 at (  1.0000 *m,  0.2000 *m, 0.0000 *m) with 6 dofs;
add node #     1004 at (  1.0000 *m,  0.0000 *m, 0.0000 *m) with 6 dofs;
add node #     1006 at (  1.0000 *m,  0.0000 *m, 0.2000 *m) with 6 dofs;
add node #     1007 at (  1.0000 *m,  0.2000 *m, 0.2000 *m) with 6 dofs;
// And connect the nodes at the same location.
add constraint equal dof with master node # 3 and slave node # 1003 dof to constrain ux uy uz;
add constraint equal dof with master node # 4 and slave node # 1004 dof to constrain ux uy uz;
add constraint equal dof with master node # 6 and slave node # 1006 dof to constrain ux uy uz;
add constraint equal dof with master node # 7 and slave node # 1007 dof to constrain ux uy uz;

add mass to node # 24 mx = rho*A*L my = rho*A*L mz = rho*A*L;

// add 6 beams to connect the mass
smallb=0.01*m;
smallh=0.01*m;
smallE = 1e9*N/m^2;
smallnu=0.3;
smallrho=0*kg/m^3;
smallI=smallb*smallh^3/12.0;
add element # 11 type beam_elastic with nodes (1003,1004)
  cross_section = smallb*smallh
  elastic_modulus = smallE
```

```
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
add element # 12 type beam_elastic with nodes (1003,1006)
  cross_section = smallb*smallh
  elastic_modulus = smallE
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
add element # 13 type beam_elastic with nodes (1003,1007)
  cross_section = smallb*smallh
  elastic_modulus = smallE
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
add element # 14 type beam_elastic with nodes (1004,1006)
  cross_section = smallb*smallh
  elastic_modulus = smallE
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
add element # 15 type beam_elastic with nodes (1004,1007)
  cross_section = smallb*smallh
  elastic_modulus = smallE
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);
add element # 16 type beam_elastic with nodes (1006,1007)
  cross_section = smallb*smallh
  elastic_modulus = smallE
```

```
  shear_modulus = smallE/2/(1+smallnu)
  torsion_Jx = 0.33*smallb*smallh^3
  bending_Iy = smallI
  bending_Iz = smallI
  mass_density = smallrho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);


// // -----------------------------------------------------------------------------
// // --slowLoading------------------------------------------------------------------
// // add the 1 Newton load in 180 seconds.
// // -----------------------------------------------------------------------------
// new loading stage "slowLoading";
// add load # 1 to node # 4 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 2 to node # 6 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 3 to node # 3 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 4 to node # 7 type path_time_series Fz=1/36.0*N series_file = "slowLoading.txt"
    ;
// add load # 5 to node # 20 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 6 to node # 11 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 7 to node # 15 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 8 to node # 19 type path_time_series Fz=1/9.0*N series_file = "slowLoading.txt"
    ;
// add load # 9 to node # 24 type path_time_series Fz=4/9.0*N series_file = "slowLoading.txt"
    ;
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 2000 steps using transient algorithm
//  time_step = 0.1*s;

// // -----------------------------------------------------------------------------
// // --fastLoading------------------------------------------------------------------
// // add the 1 Newton load in 0.6 seconds.
// // -----------------------------------------------------------------------------
// new loading stage "fastLoading";
// add load # 101 to node # 4 type path_time_series Fz=1/36.0*N series_file =
     "fastLoading.txt" ;
// add load # 102 to node # 6 type path_time_series Fz=1/36.0*N series_file =
     "fastLoading.txt" ;
// add load # 103 to node # 3 type path_time_series Fz=1/36.0*N series_file =
     "fastLoading.txt" ;
// add load # 104 to node # 7 type path_time_series Fz=1/36.0*N series_file =
     "fastLoading.txt" ;
// add load # 105 to node # 20 type path_time_series Fz=1/9.0*N series_file =
     "fastLoading.txt" ;
```

```
// add load # 106 to node # 11 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 107 to node # 15 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 108 to node # 19 type path_time_series Fz=1/9.0*N series_file =
    "fastLoading.txt" ;
// add load # 109 to node # 24 type path_time_series Fz=4/9.0*N series_file =
    "fastLoading.txt" ;
// // add algorithm and solver
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 1000 steps using transient algorithm
//  time_step = 0.01*s;

// // -----------------------------------------------------------------------
// // --freeVibration---------------------------------------------------------
// // -----------------------------------------------------------------------
new loading stage "freeVibration";
add load # 201 to node # 4 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 202 to node # 6 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 203 to node # 3 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 204 to node # 7 type path_time_series Fz=1/36.0*N series_file =
    "freeVibration.txt" ;
add load # 205 to node # 20 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 206 to node # 11 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 207 to node # 15 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 208 to node # 19 type path_time_series Fz=1/9.0*N series_file =
    "freeVibration.txt" ;
add load # 209 to node # 24 type path_time_series Fz=4/9.0*N series_file =
    "freeVibration.txt" ;
// add algorithm and solver
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 100 steps using transient algorithm
  time_step = 0.1*s;

// end
bye;
```

Displacement Results. The ESSI model fei files for this example can be downloaded here.

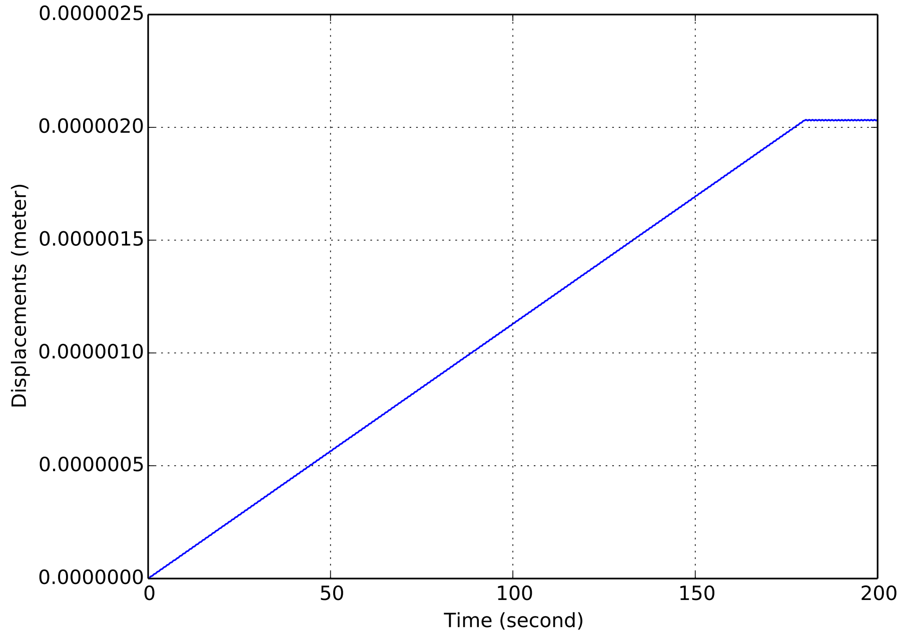


Figure 23: Slow loading condition, vertical displacements of the cantilever tip.

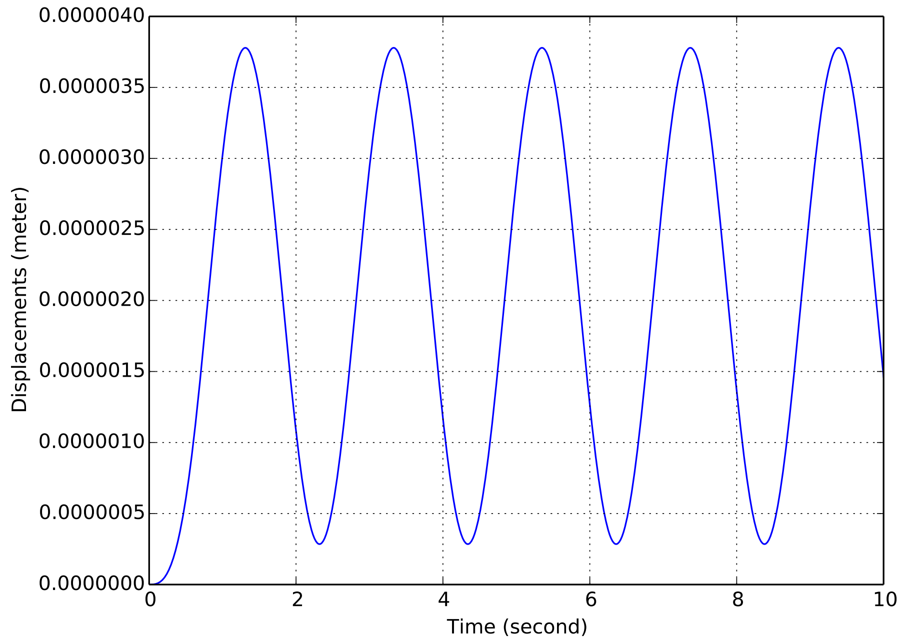


Figure 24: Fast loading condition, vertical displacements of the cantilever tip.

6 Elastic Beam Element, Dynamic Loading, Viscous (Rayleigh/Caughey) and Numerical (Newmark/HHT) Damping

6.1 Single Beam Element

Problem description:

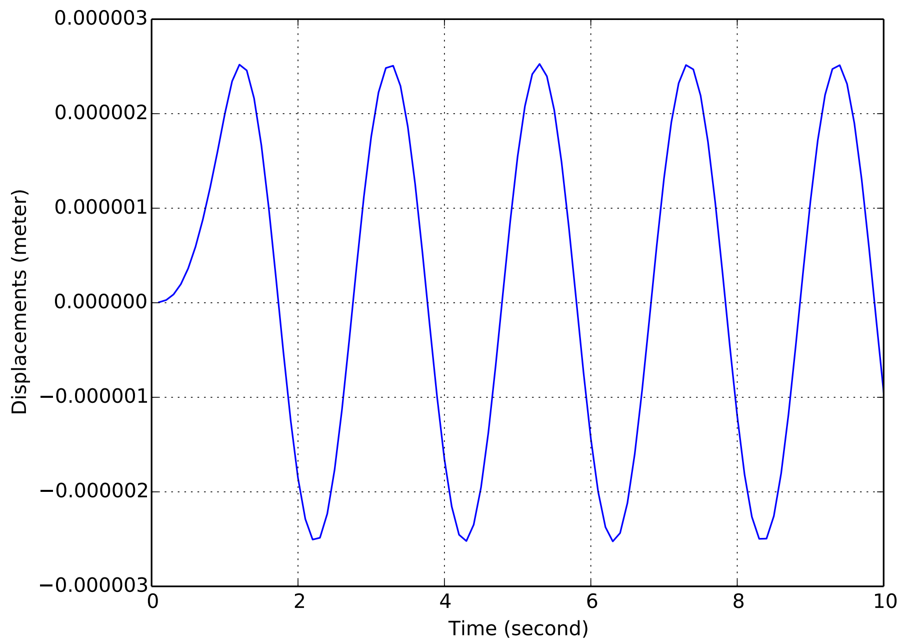


Figure 25: Free vibration condition, vertical displacements of the cantilever tip.

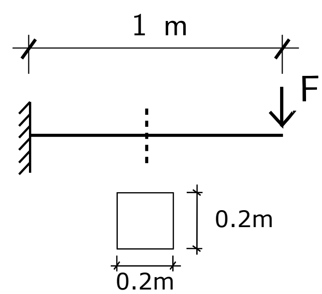


Figure 26: The cantilever-mass model.

ESSI model fei file:

```
model name "beam_1element" ;

// add node
add node # 1 at (  0.0*m ,   0.0*m,    0.0*m) with 6 dofs;
add node # 2 at (  1.0*m ,   0.0*m,    0.0*m) with 6 dofs;

// Geometry: width and height
b=0.2*m;
h=0.2*m;

// Materials: properties
natural_period  = 1*s;
natural_frequency = 2*pi/natural_period;
elastic_constant = 1e9*N/m^2;
I=b*h^3/12.0;
A=b*h;
```

```
L=1*m;
rho   = (1.8751)^4*elastic_constant*I/(natural_frequency^2*L^4*A);
possion_ratio=0.3;

// add elements
add element # 1 type beam_elastic with nodes (1,2)
  cross_section = b*h
  elastic_modulus = elastic_constant
  shear_modulus = elastic_constant/2/(1+possion_ratio)
  torsion_Jx = 0.33*b*h^3
  bending_Iy = b*h^3/12
  bending_Iz = b*h^3/12
  mass_density = rho
  xz_plane_vector = ( 1, 0, 1)
  joint_1_offset = (0*m, 0*m, 0*m)
  joint_2_offset = (0*m, 0*m, 0*m);

// add boundary condition
fix node #    1 dofs all;

// // ------------------------------------------------------------------------------
// // --no-damping---------------------------------------------------------------
// // ------------------------------------------------------------------------------
// new loading stage "no-damping";
// add load # 1 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "freeVibration.txt" ;
// define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 100 steps using transient algorithm
//  time_step = 0.1*s;

// // ------------------------------------------------------------------------------
// // --Newmark-damping-------------------------------------------------------------
// // ------------------------------------------------------------------------------
// remove load # 2;
// new loading stage "Newmark-damping";
// add load # 3 to node # 2 type path_time_series
//  Fz =  1.*N
//  series_file = "freeVibration.txt" ;
// define dynamic integrator Newmark with gamma = 0.6 beta = 0.3025;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 100 steps using transient algorithm
//  time_step = 0.1*s;
// // ------------------------------------------------------------------------------
// // --HHT-damping----------------------------------------------------------------
// // ------------------------------------------------------------------------------
// remove load # 3;
// new loading stage "HHT-damping";
// add load # 4 to node # 6 type path_time_series
//  Fz =  1.*kN
//  series_file = "freeVibration.txt" ;
// define dynamic integrator Hilber_Hughes_Taylor with alpha = -0.20;
```

```
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 300 steps using transient algorithm
//  time_step = 0.1*s;
// // ---------------------------------------------------------------------------
// // --Rayleigh-damping------------------------------------------------------------
// // ---------------------------------------------------------------------------
// remove load # 4;
// simulate using eigen algorithm number_of_modes = 2;
f1=0.996807/s;
f2=0.996807/s;
w1 = 2*pi*f1;
w2 = 2*pi*f2;
xi=0.05;
rayl_a1 = 2*xi/(w1 + w2);
rayl_a0 = rayl_a1*w1*w2;

add damping # 1 type Rayleigh with
  a0 = rayl_a0
  a1 = rayl_a1
  stiffness_to_use = Initial_Stiffness;
add damping # 1 to element # 1;

new loading stage "Rayleigh-damping";
add load # 5 to node # 2 type path_time_series
  Fz = 1.*N
  series_file = "freeVibration.txt" ;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check ;
define solver ProfileSPD;
simulate 100 steps using transient algorithm
  time_step = 0.1*s;

// // ---------------------------------------------------------------------------
// // --Caughey3rd-damping-------------------------------------------------------
// // ---------------------------------------------------------------------------
// add damping # 2 type Caughey3rd with
//  a0 = 0.560523/s
//  a1 = 0.0730746*s
//  a2 = 0.000361559*s^3
//  stiffness_to_use = Last_Committed_Stiffness;
// kk=1;
// while (kk<6) {
//  add damping # 2 to element # kk;
//  kk+=1;
// }
// new loading stage "Caughey3rd-damping";
// add load # 6 to node # 6 type path_time_series
//  Fz = 10.*kN
//  series_file = "freeVibration.txt" ;
// define dynamic integrator Newmark with gamma = 0.6 beta = 0.3025;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 100 steps using transient algorithm
//  time_step = 0.2*s;
```

```


// // ---------------------------------------------------------------------------
// // --Caughey4th-damping------------------------------------------------------
// // ---------------------------------------------------------------------------
// add damping # 2 type Caughey4th with
//  a0 = 0.560523/s
//  a1 = 0.0756472*s
//  a2 = 0.000517195*s^3
//  a3 = 1.20005*10^(-6)*s^5
//  stiffness_to_use = Last_Committed_Stiffness;
// kk=1;
// while (kk<6) {
//  add damping # 2 to element # kk;
//  kk+=1;
// }
// new loading stage "Caughey4th-damping";
// add load # 6 to node # 6 type path_time_series
//  Fz = 10.*kN
//  series_file = "freeVibration.txt" ;
// define dynamic integrator Newmark with gamma = 0.6 beta = 0.3025;
// define algorithm With_no_convergence_check ;
// define solver ProfileSPD;
// simulate 100 steps using transient algorithm
//  time_step = 0.2*s;

bye;
```

Displacement results against time series

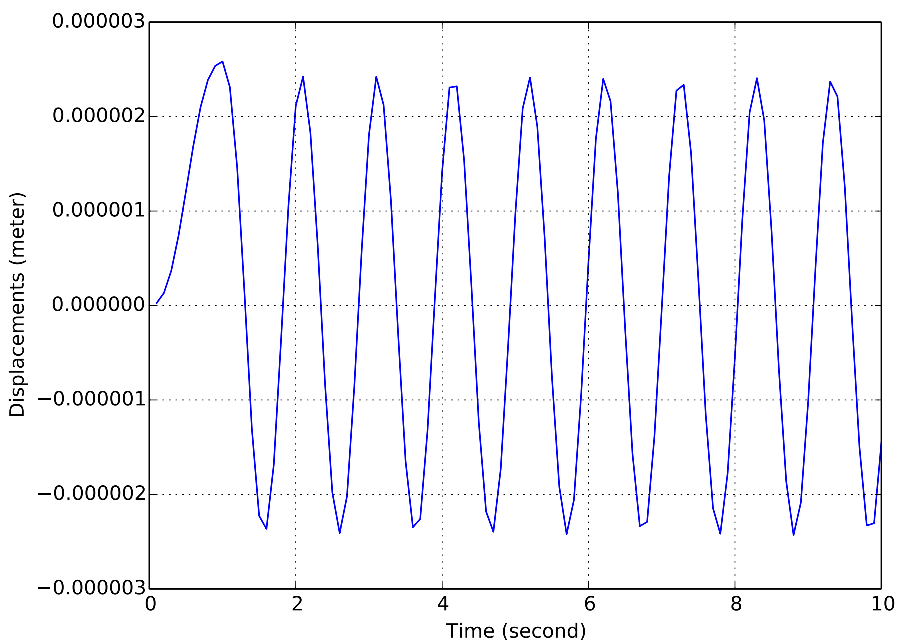


Figure 27: Free vibration condition, no damping, vertical displacements of the cantilever tip.

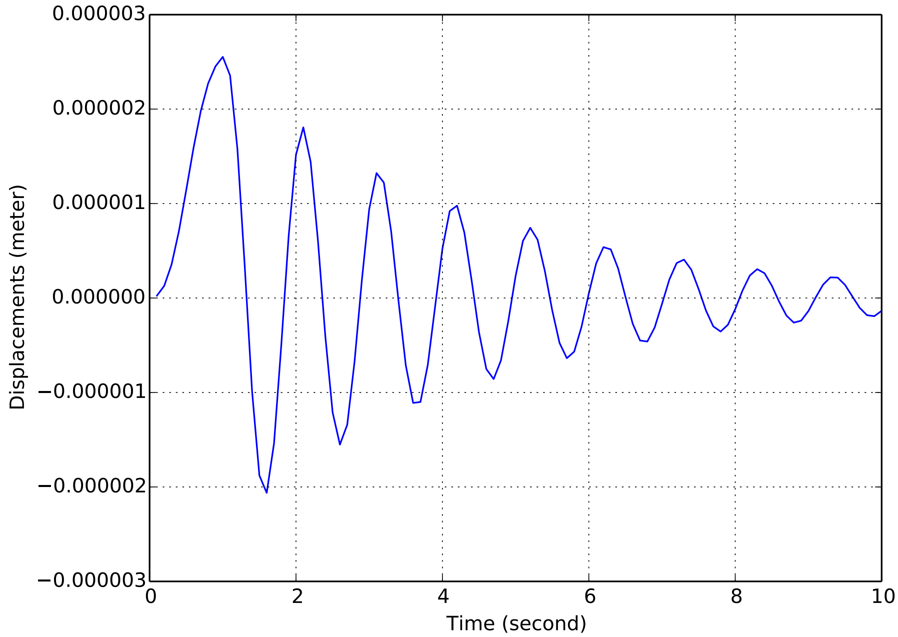


Figure 28: Free vibration condition, viscous (Rayleigh) damping, vertical displacements of the cantilever tip.

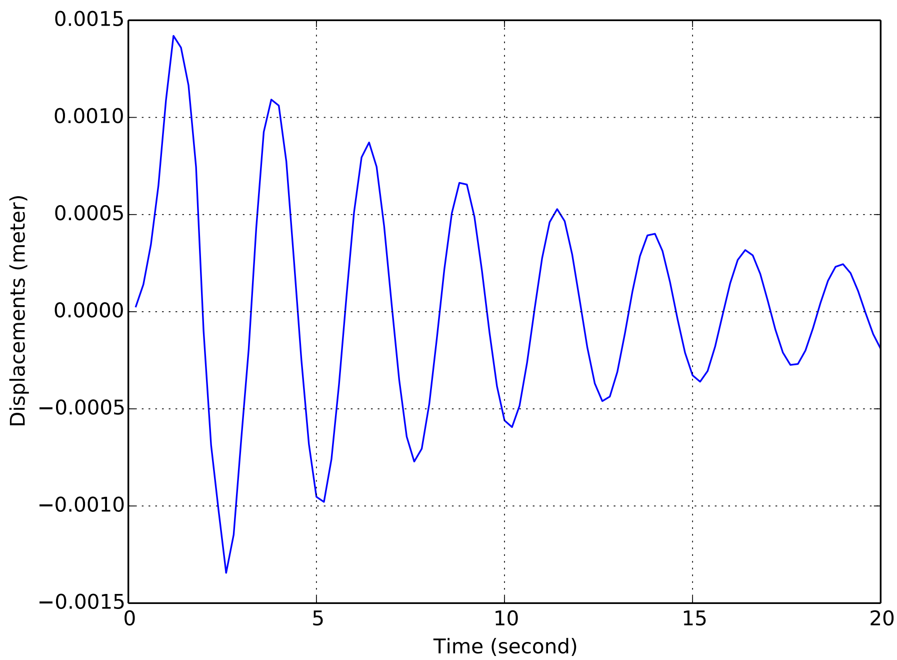


Figure 29: Free vibration condition, viscous (Caughey3rd) damping, vertical displacements of the cantilever tip.

The ESSI model fei files for this example can be downloaded here.

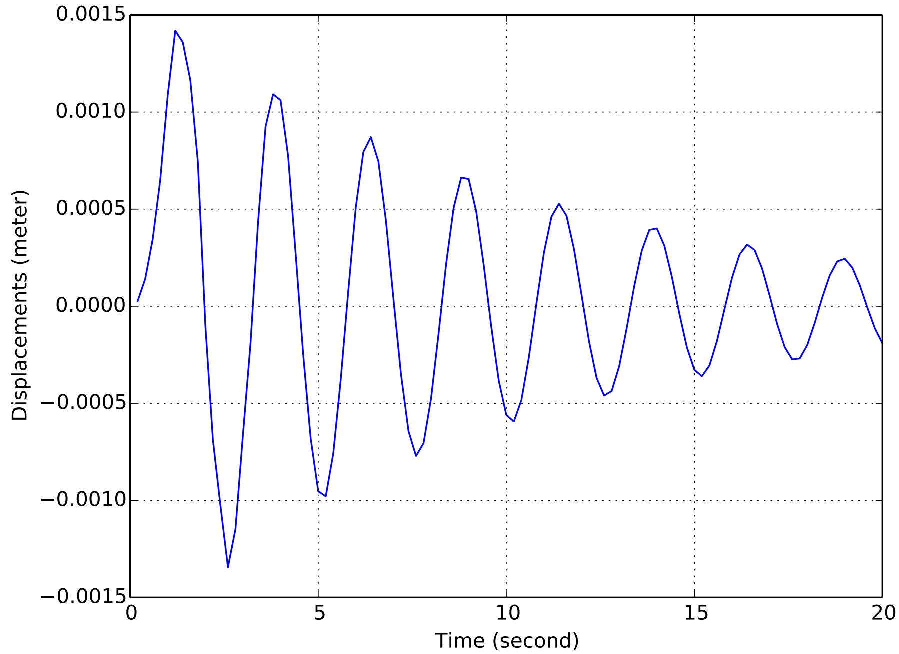


Figure 30: Free vibration condition, viscous (Caughey4th) damping, vertical displacements of the cantilever tip.

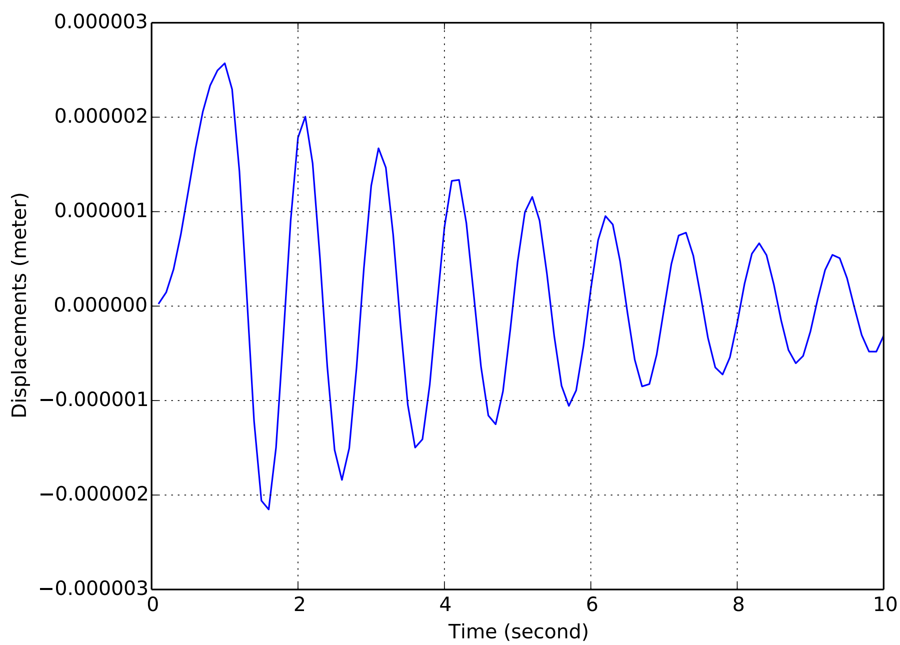


Figure 31: Free vibration condition, numerical (Newmark) damping, vertical displacements of the cantilever tip.

7 Elastic Beam Element for a Simple Frame Structure

Problem Description

- Dimensions: hidth=$6m$, height=$6m$, force=$100N$

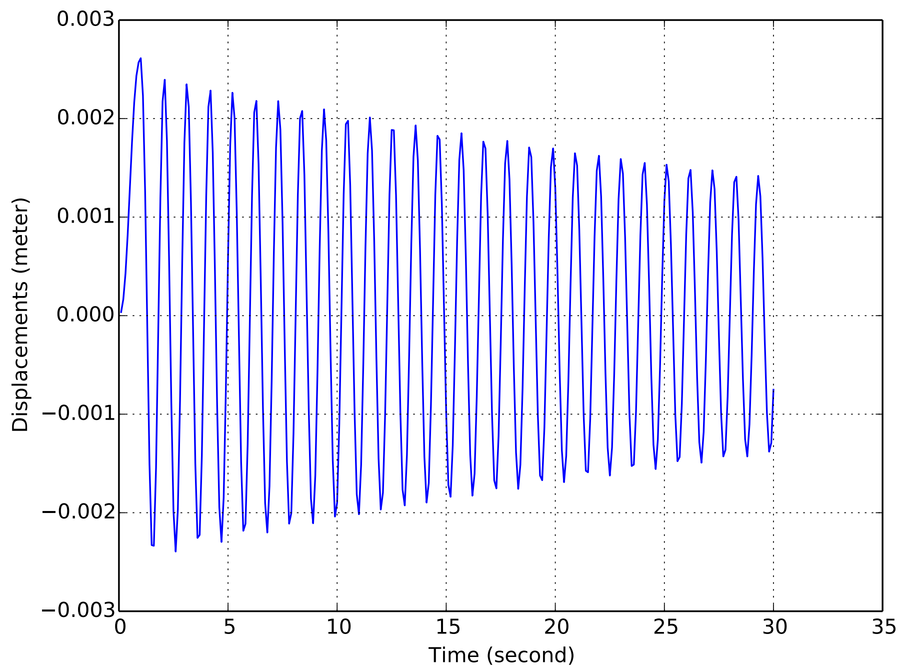


Figure 32: Free vibration condition, numerical (HHT) damping, vertical displacements of the cantilever tip.

- Element dimensions: length=6m, cross section width=1m, cross section height=1m, mass density $\rho = 0.0 \text{kN/m}^3$, Young's modulus $E = 1E8$ Pa, Poisson's ratio $\nu = 0.0$.

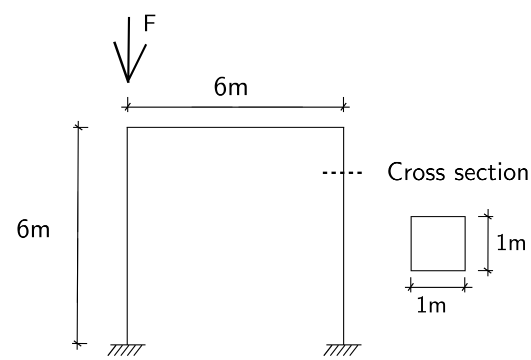


Figure 33: Elastic frame with *beam_ elastic* elements.

ESSI model fei file:

```
model name "beam_element_presentation" ;

add node # 1 at ( 0.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 2 at ( 0.00*m, 0.00*m, 6.00*m) with 6 dofs;
add node # 3 at ( 6.00*m, 0.00*m, 6.00*m) with 6 dofs;
```

```
add node # 4 at ( 6.00*m, 0.00*m, 0.00*m) with 6 dofs;

elastic_constant = 1e8*N/m^2;
b=1*m;
h=1*m;
rho   = 0*kg/m^3;    // Mass density

add element # 1 type beam_elastic with nodes (1, 2)
 cross_section = b*h  elastic_modulus = elastic_constant
 shear_modulus = elastic_constant/2
 torsion_Jx = 0.33*b*h^3 bending_Iy = b*h^3/12 bending_Iz = h*b^3/12
 mass_density = rho  xz_plane_vector = (1, 0, 1 )
   joint_1_offset = (0*m, 0*m, 0*m ) joint_2_offset = (0*m, 0*m, 0*m );

add element # 2 type beam_elastic with nodes (2,3)
 cross_section = b*h elastic_modulus = elastic_constant
 shear_modulus = elastic_constant/2
 torsion_Jx = 0.33*b*h^3 bending_Iy = b*h^3/12 bending_Iz = h*b^3/12
 mass_density = rho  xz_plane_vector = (1, 0, 1 )
   joint_1_offset = (0*m, 0*m, 0*m ) joint_2_offset = (0*m, 0*m, 0*m );

add element # 3 type beam_elastic with nodes (3,4)
 cross_section = b*h elastic_modulus = elastic_constant
 shear_modulus = elastic_constant/2
 torsion_Jx = 0.33*b*h^3 bending_Iy = b*h^3/12 bending_Iz = h*b^3/12
 mass_density = rho  xz_plane_vector = (1, 0, 1 )
   joint_1_offset = (0*m, 0*m, 0*m ) joint_2_offset = (0*m, 0*m, 0*m );

fix node #1 dofs all;
fix node #4 dofs all;

new loading stage "Fz";

add load # 1 to node # 2 type linear Fz=50*N;

define algorithm With_no_convergence_check;
define solver ProfileSPD;
define load factor increment 1;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

8 27NodeBrick Cantilever Beam for the static load

Problem description:

Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. The force direction is shown in Figure (112).

Numerical model:

The 27NodeBrick elements for cantilever beams is shown in Figure (113):

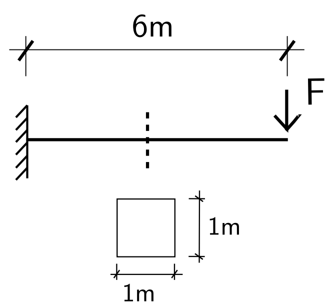


Figure 34: Problem description for cantilever beam.

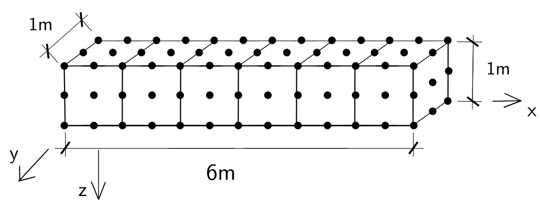


Figure 35: 27NodeBrick elements for cantilever beams made of solid elements.

ESSI model fei file:

```
model name "6meter_cantilever_27brick" ;

add material # 1 type linear_elastic_isotropic_3d
  mass_density = 0*kg/m^3
  elastic_modulus = 1e8*N/m^2
  poisson_ratio = 0.0;

add node # 1 at (  0.00 *m,  1.00 *m, 0.00 *m) with 3 dofs;
add node # 2 at (  0.00 *m,  0.00 *m, 0.00 *m) with 3 dofs;
add node # 3 at (  6.00 *m,  1.00 *m, 0.00 *m) with 3 dofs;
add node # 4 at (  5.00 *m,  1.00 *m, 0.00 *m) with 3 dofs;
add node # 5 at (  4.00 *m,  1.00 *m, 0.00 *m) with 3 dofs;
add node # 6 at (  3.00 *m,  1.00 *m, 0.00 *m) with 3 dofs;
...
...
add node #117 at ( 5.50 *m,  0.50 *m, 1.00 *m) with 3 dofs;

add element # 1 type 27NodeBrickLT with nodes( 2, 10, 8, 1, 15, 17, 28, 23, 29, 30, 31,
    32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47) use material # 1;
add element # 2 type 27NodeBrickLT with nodes( 10, 11, 7, 8, 17, 18, 27, 28, 48, 49, 50,
    30, 51, 52, 53, 34, 38, 54, 55, 39, 56, 57, 58, 59, 43, 60, 61) use material # 1;
add element # 3 type 27NodeBrickLT with nodes( 11, 12, 6, 7, 18, 19, 26, 27, 62, 63, 64,
    49, 65, 66, 67, 52, 54, 68, 69, 55, 70, 71, 72, 73, 58, 74, 75) use material # 1;
add element # 4 type 27NodeBrickLT with nodes( 12, 13, 5, 6, 19, 20, 25, 26, 76, 77, 78,
    63, 79, 80, 81, 66, 68, 82, 83, 69, 84, 85, 86, 87, 72, 88, 89) use material # 1;
add element # 5 type 27NodeBrickLT with nodes( 13, 14, 4, 5, 20, 21, 24, 25, 90, 91, 92,
    77, 93, 94, 95, 80, 82, 96, 97, 83, 98, 99, 100, 101, 86, 102, 103) use material # 1;
```

```
add element # 6 type 27NodeBrickLT with nodes( 14, 9, 3, 4, 21, 16, 22, 24, 104, 105, 106,
    91, 107, 108, 109, 94, 96, 110, 111, 97, 112, 113, 114, 115, 100, 116, 117) use material #
    1;

fix node # 1 dofs all;
fix node # 2 dofs all;
fix node # 15 dofs all;
fix node # 23 dofs all;
fix node # 32 dofs all;
fix node # 36 dofs all;
fix node # 37 dofs all;
fix node # 40 dofs all;
fix node # 45 dofs all;

new loading stage "Fz";
add load # 1 to node # 13 type linear Fz=2.777778*N;
add load # 2 to node # 24 type linear Fz=2.777778*N;
add load # 3 to node # 3 type linear Fz=2.777778*N;
add load # 4 to node # 34 type linear Fz=2.777778*N;
add load # 5 to node # 182 type linear Fz=11.111111*N;
add load # 6 to node # 177 type linear Fz=11.111111*N;
add load # 7 to node # 180 type linear Fz=11.111111*N;
add load # 8 to node # 183 type linear Fz=11.111111*N;
add load # 9 to node # 186 type linear Fz=44.444444*N;

define algorithm With_no_convergence_check ;
define solver UMFPack;
define load factor increment 1;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

9 4NodeANDES Cantilever Beams Under the Force Perpendicular to Plane

Problem description:

Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$.

Numerical model:

For a force direction perpendicular to the plane, only the bending deformation is present. The model is shown in Figure (166).

ESSI model fei file:

```
 model name "6meter_cantilever_4NodeANDES" ;

add material # 1 type linear_elastic_isotropic_3d
  mass_density = 0*kg/m^3
  elastic_modulus = 1e8*N/m^2
```

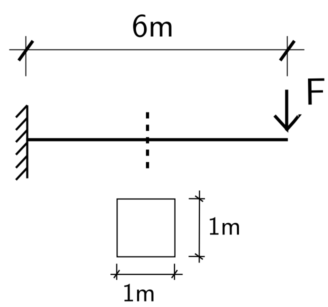


Figure 36: Cantilever beams

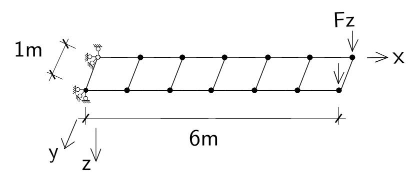


Figure 37: 4NodeANDES elements for cantilever beams under force perpendicular to plane.

```
  poisson_ratio = 0.0;

add node # 1 at ( 0.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 2 at ( 6.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 3 at ( 1.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 4 at ( 2.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 5 at ( 3.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 6 at ( 4.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 7 at ( 5.0*m, 0.0*m, 0.0*m) with 6 dofs;
add node # 8 at ( 6.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 9 at ( 0.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 10 at ( 5.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 11 at ( 4.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 12 at ( 3.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 13 at ( 2.0*m, 1.0*m, 0.0*m) with 6 dofs;
add node # 14 at ( 1.0*m, 1.0*m, 0.0*m) with 6 dofs;

h = 1*m;
add element # 1 type 4NodeShell_ANDES with nodes (1,3,14,9) use material # 1 thickness = h ;
add element # 2 type 4NodeShell_ANDES with nodes (3,4,13,14) use material # 1 thickness = h ;
add element # 3 type 4NodeShell_ANDES with nodes (4,5,12,13) use material # 1 thickness = h ;
add element # 4 type 4NodeShell_ANDES with nodes (5,6,11,12) use material # 1 thickness = h ;
add element # 5 type 4NodeShell_ANDES with nodes (6,7,10,11) use material # 1 thickness = h ;
add element # 6 type 4NodeShell_ANDES with nodes (7,2,8,10) use material # 1 thickness = h ;

fix node # 1 dofs all   ;
fix node # 9 dofs all   ;

new loading stage "Fz";
add load # 1 to node # 8 type linear Fz=50*N;
```

```
add load # 2 to node # 2 type linear Fz=50*N;

define algorithm With_no_convergence_check ;
define solver ProfileSPD;
define load factor increment 1;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

10 4NodeANDES Cantilever Beams under the In-Plane Force

Problem description:

Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$.

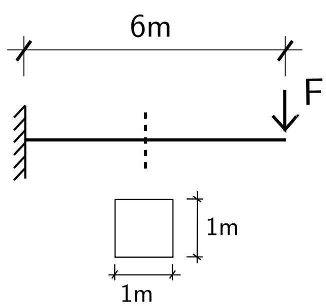


Figure 38: Problem description for cantilever beams with in plane force

Numerical model:

The 4NodeANDES elements under in-plane force is shown in Figure (167).

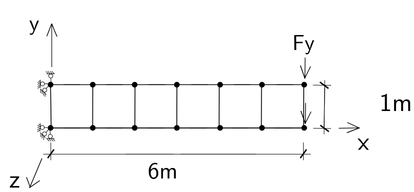


Figure 39: 4NodeANDES elements for cantilever beams under in-plane force

ESSI model fei file:

```
model name "6meter_cantilever_4NodeANDES" ;

add material # 1 type linear_elastic_isotropic_3d
```

```
  mass_density = 0*kg/m^3
  elastic_modulus = 1e8*N/m^2
  poisson_ratio = 0.0;

add node #  1 at ( 0.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  2 at ( 6.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  3 at ( 1.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  4 at ( 2.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  5 at ( 3.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  6 at ( 4.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  7 at ( 5.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node #  8 at ( 6.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node #  9 at ( 0.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node # 10 at ( 5.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node # 11 at ( 4.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node # 12 at ( 3.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node # 13 at ( 2.00*m, 1.00*m, 0.00*m) with 6 dofs;
add node # 14 at ( 1.00*m, 1.00*m, 0.00*m) with 6 dofs;

h    = 1*m;
add element # 1 type 4NodeShell_ANDES with nodes (1,3,14,9) use material # 1 thickness = h ;
add element # 2 type 4NodeShell_ANDES with nodes (3,4,13,14) use material # 1 thickness = h ;
add element # 3 type 4NodeShell_ANDES with nodes (4,5,12,13) use material # 1 thickness = h ;
add element # 4 type 4NodeShell_ANDES with nodes (5,6,11,12) use material # 1 thickness = h ;
add element # 5 type 4NodeShell_ANDES with nodes (6,7,10,11) use material # 1 thickness = h ;
add element # 6 type 4NodeShell_ANDES with nodes (7,2,8,10) use material # 1 thickness = h ;

fix node # 1 dofs all;
fix node # 9 dofs all;

new loading stage "Fy";
add load # 1 to node # 8 type linear Fy=50*N;
add load # 2 to node # 2 type linear Fy=50*N;

define algorithm With_no_convergence_check ;
define solver ProfileSPD;
define load factor increment 1;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

11 27NodeBrick Cantilever Beams for Dynamic Input

Problem description:

Length=20m, Width=1m, Height=1m, E=504MPa, $\nu = 0.4$.
All degree of freedoms at the bottom nodes are fixed.
The load is a self weight with a dynamic displacement of supports.

Numerical model:

The numerical model applied 27NodeBrick to simulate the 1D motion.

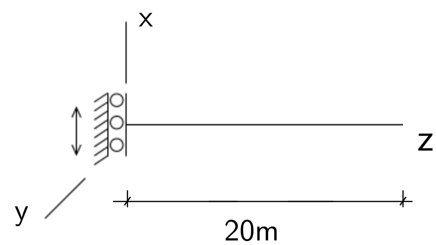


Figure 40: Problem description for one simple dynamic example

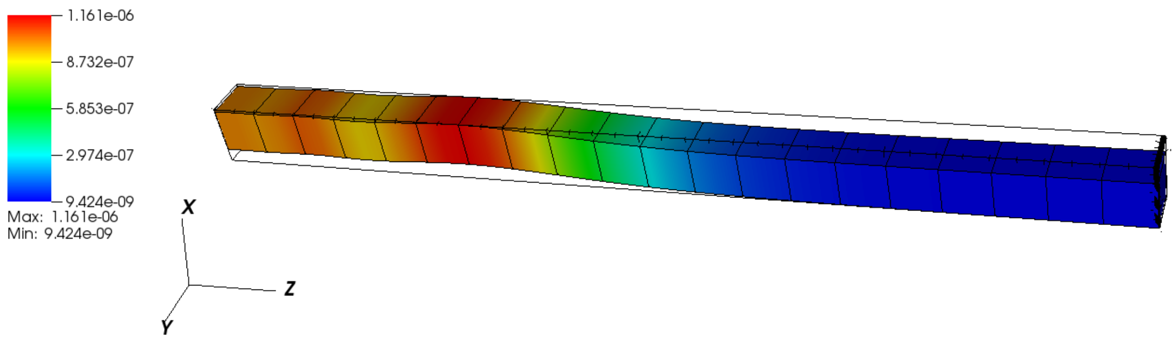


Figure 41: Numerical model for one simple dynamic example

ESSI model fei file:

```
model name "dynamic_example";

add material # 1 type linear_elastic_isotropic_3d_LT
 mass_density   = 2000*kg/m^3
 elastic_modulus = 504000000.00*Pa
 poisson_ratio  = 0.4;

add node No 1 at (0*m, 0*m, 0*m) with 3 dofs;
add node No 2 at (0*m, 0.5*m, 0*m) with 3 dofs;
add node No 3 at (0*m, 1*m, 0*m) with 3 dofs;
add node No 4 at (0.5*m, 0*m, 0*m) with 3 dofs;
add node No 5 at (0.5*m, 0.5*m, 0*m) with 3 dofs;
add node No 6 at (0.5*m, 1*m, 0*m) with 3 dofs;
...
...
add node No 369 at (1*m, 1*m, 20*m) with 3 dofs;

add element # 1 type 27NodeBrickLT with nodes
    (27,21,19,25,9,3,1,7,24,20,22,26,6,2,4,8,18,12,10,16,14,15,11,13,17,23,5) use material # 1
    ;
add element # 2 type 27NodeBrickLT with nodes
    (45,39,37,43,27,21,19,25,42,38,40,44,24,20,22,26,36,30,28,34,32,33,29,31,35,41,23) use
    material # 1 ;
```

```
add element # 3 type 27NodeBrickLT with nodes (63,57,55,61,45,39,37,43,60,56,58,62,42,38,40,44,54,48,46,52,50,51,47,49,53,59,41) use material # 1 ;
add element # 4 type 27NodeBrickLT with nodes (81,75,73,79,63,57,55,61,78,74,76,80,60,56,58,62,72,66,64,70,68,69,65,67,71,77,59) use material # 1 ;
add element # 5 type 27NodeBrickLT with nodes (99,93,91,97,81,75,73,79,96,92,94,98,78,74,76,80,90,84,82,88,86,87,83,85,89,95,77) use material # 1 ;
...
...
add element # 20 type 27NodeBrickLT with nodes (369,363,361,367,351,345,343,349,366,362,364,368,348,
 344,346,350,360,354,352,358,356,357,353,355,359,365,347) use material # 1 ;

add acceleration field # 1 ax = 0*g ay = 0*g az = -1*g ;
add load # 1 to element # 1 type self_weight use acceleration field # 1;
add load # 2 to element # 2 type self_weight use acceleration field # 1;
add load # 3 to element # 3 type self_weight use acceleration field # 1;
add load # 4 to element # 4 type self_weight use acceleration field # 1;
add load # 5 to element # 5 type self_weight use acceleration field # 1;
add load # 6 to element # 6 type self_weight use acceleration field # 1;
...
...
add load # 20 to element # 20 type self_weight use acceleration field # 1;

fix node No 1 dofs  uy uz;
fix node No 2 dofs  uy uz;
fix node No 3 dofs  uy uz;
fix node No 4 dofs  uy uz;
fix node No 5 dofs  uy uz;
fix node No 6 dofs  uy uz;
...
...
fix node No 369 dofs  uy uz;

zeta = 0.0166667;
fq1  = 3.75;
fq2  = 11.25;
omega1 = 2*pi*fq1;
omega2 = 2*pi*fq2;
zeta1 = zeta;
zeta2 = zeta;
alpha1 = 2*omega1*omega2*(zeta1*omega2-zeta2*omega1)/(omega2*omega2-omega1*omega1);
beta1  = 2*            (zeta2*omega2-zeta1*omega1)/(omega2*omega2-omega1*omega1);
add damping # 1
   type Rayleigh
   with
   a0 = alpha1/s
   a1 = beta1*s
   stiffness_to_use = Initial_Stiffness;

add damping # 1 to element # 1;
add damping # 1 to element # 2;
add damping # 1 to element # 3;
```

```
add damping # 1 to element # 4;
add damping # 1 to element # 5;
add damping # 1 to element # 6;
...
...
add damping # 1 to element # 20;

new loading stage "impose_motion";

add imposed motion # 1001 to node # 1 dof ux
 displacement_scale_unit = 1*m   displacement_file  = "dis.txt"
 velocity_scale_unit   = 1*m/s    velocity_file      = "vel.txt"
 acceleration_scale_unit = 1*m/s^2 acceleration_file = "acc.txt";

add imposed motion # 1002 to node # 2 dof ux
 displacement_scale_unit = 1*m     displacement_file  = "dis.txt"
 velocity_scale_unit   = 1*m/s      velocity_file      = "vel.txt"
 acceleration_scale_unit = 1*m/s^2 acceleration_file  = "acc.txt";

add imposed motion # 1003 to node # 3 dof ux
 displacement_scale_unit = 1*m     displacement_file = "dis.txt"
 velocity_scale_unit   = 1*m/s      velocity_file     = "vel.txt"
 acceleration_scale_unit = 1*m/s^2 acceleration_file = "acc.txt";
...
...
add imposed motion # 1009 to node # 9 dof ux
 displacement_scale_unit = 1*m    displacement_file = "dis.txt"
 velocity_scale_unit   = 1*m/s     velocity_file     = "vel.txt"
 acceleration_scale_unit = 1*m/s^2 acceleration_file = "acc.txt";

define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
define algorithm With_no_convergence_check;
define solver ProfileSPD;
simulate 50 steps using transient algorithm time_step = 0.005*s;

bye;
```

The ESSI model fei files for this example can be downloaded here.

12 4NodeANDES Square Plate with Four Edges Clamped

Problem description:

Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.
The four edges are **clamped**.
The load is a self weight.

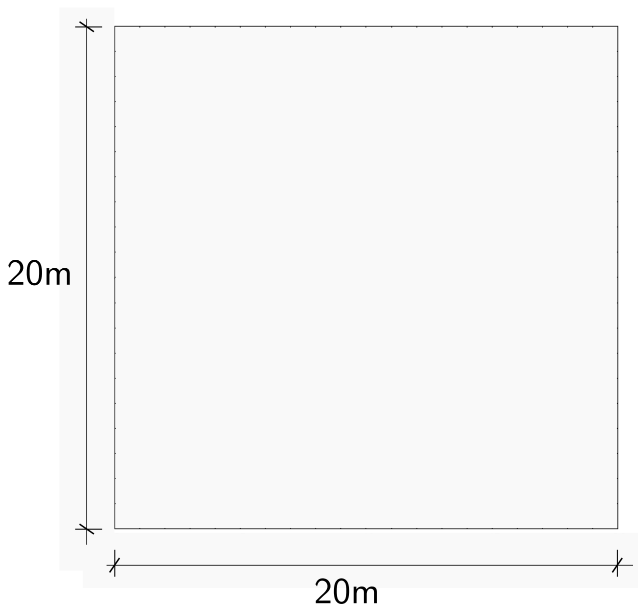


Figure 42: Square plate with four edges clamped

Numerical model:

The element side length is 1 meter.

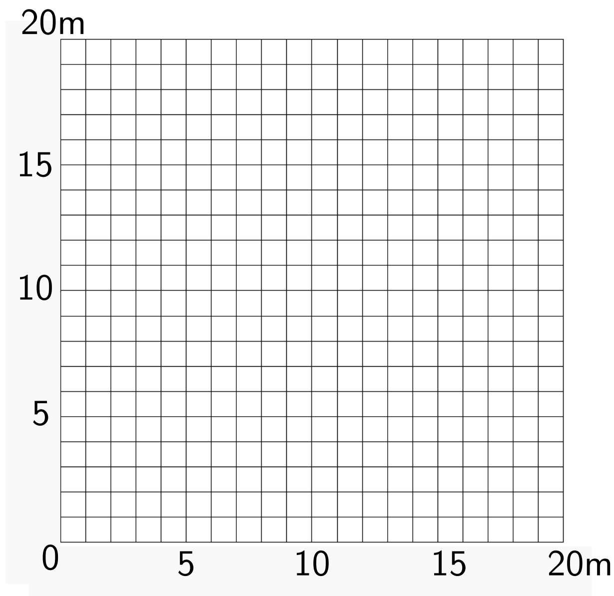


Figure 43: 4NodeANDES edge clamped square plate with element side length 1m

ESSI model fei file:

```
model name "square_plate" ;

add material # 1 type linear_elastic_isotropic_3d
  mass_density = 1e2*kg/m^3 elastic_modulus = 1e8*N/m^2 poisson_ratio = 0.3;

add node # 1 at (  0.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 2 at ( 20.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 3 at (  1.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 4 at (  2.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 5 at (  3.00*m, 0.00*m, 0.00*m) with 6 dofs;
add node # 6 at (  4.00*m, 0.00*m, 0.00*m) with 6 dofs;
...
...
add node # 441 at ( 19.00*m, 19.00*m, 0.00*m) with 6 dofs;

h    = 1*m;
add element #     1 type 4NodeShell_ANDES with nodes( 1, 3, 81, 80) use material # 1
    thickness=h;
add element #     2 type 4NodeShell_ANDES with nodes( 3, 4, 100, 81) use material # 1
    thickness=h;
add element #     3 type 4NodeShell_ANDES with nodes( 4, 5, 119, 100) use material # 1
    thickness=h;
add element #     4 type 4NodeShell_ANDES with nodes( 5, 6, 138, 119) use material # 1
    thickness=h;
```

```
add element #     5 type 4NodeShell_ANDES with nodes( 6, 7, 157, 138) use material # 1
    thickness=h;
add element #     6 type 4NodeShell_ANDES with nodes( 7, 8, 176, 157) use material # 1
    thickness=h;
...
...
add element #   400 type 4NodeShell_ANDES with nodes( 441, 41, 22, 43) use material # 1
    thickness=h;


fix node #     1 dofs all   ;
fix node #     2 dofs all   ;
fix node #     3 dofs all   ;
fix node #     4 dofs all   ;
fix node #     5 dofs all   ;
fix node #     6 dofs all   ;
...
...
fix node #    80 dofs all   ;


new loading stage "self_weight";
add acceleration field # 1 ax = 0*g  ay =  0*g   az =  1*m/s^2;
add load # 1 to element # 1 type self_weight use acceleration field # 1;
add load # 2 to element # 2 type self_weight use acceleration field # 1;
add load # 3 to element # 3 type self_weight use acceleration field # 1;
add load # 4 to element # 4 type self_weight use acceleration field # 1;
add load # 5 to element # 5 type self_weight use acceleration field # 1;
add load # 6 to element # 6 type self_weight use acceleration field # 1;
...
...
add load # 400 to element # 400 type self_weight use acceleration field # 1;


define algorithm With_no_convergence_check ;
define solver ProfileSPD;
define load factor increment 1;
simulate 1 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

13 One Dimensional DRM Model

Problem description:

A simple 1D DRM model is shown in Fig.(44). The "DRM element", "Exterior node" and "Boundary node" are required to be designated in the DRM HDF5 input. The format and script for the HDF5 input is available in DSL/input manual.

Numerical model:

ESSI model fei file:

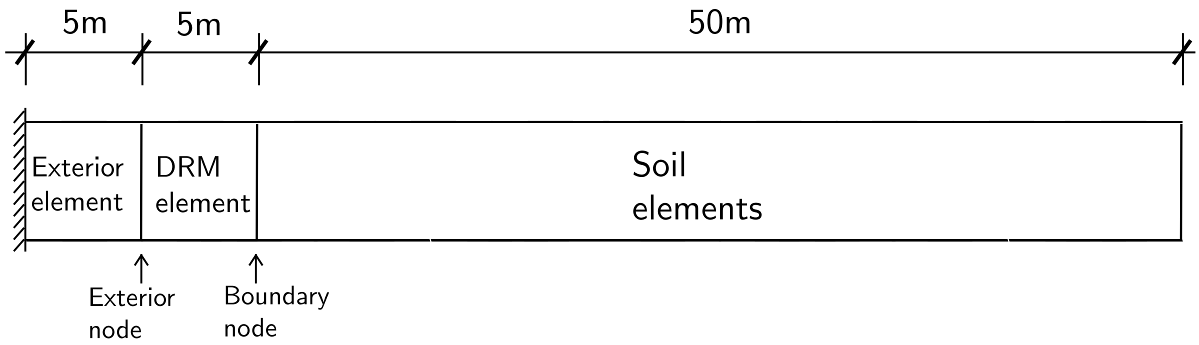


Figure 44: 1D DRM model.

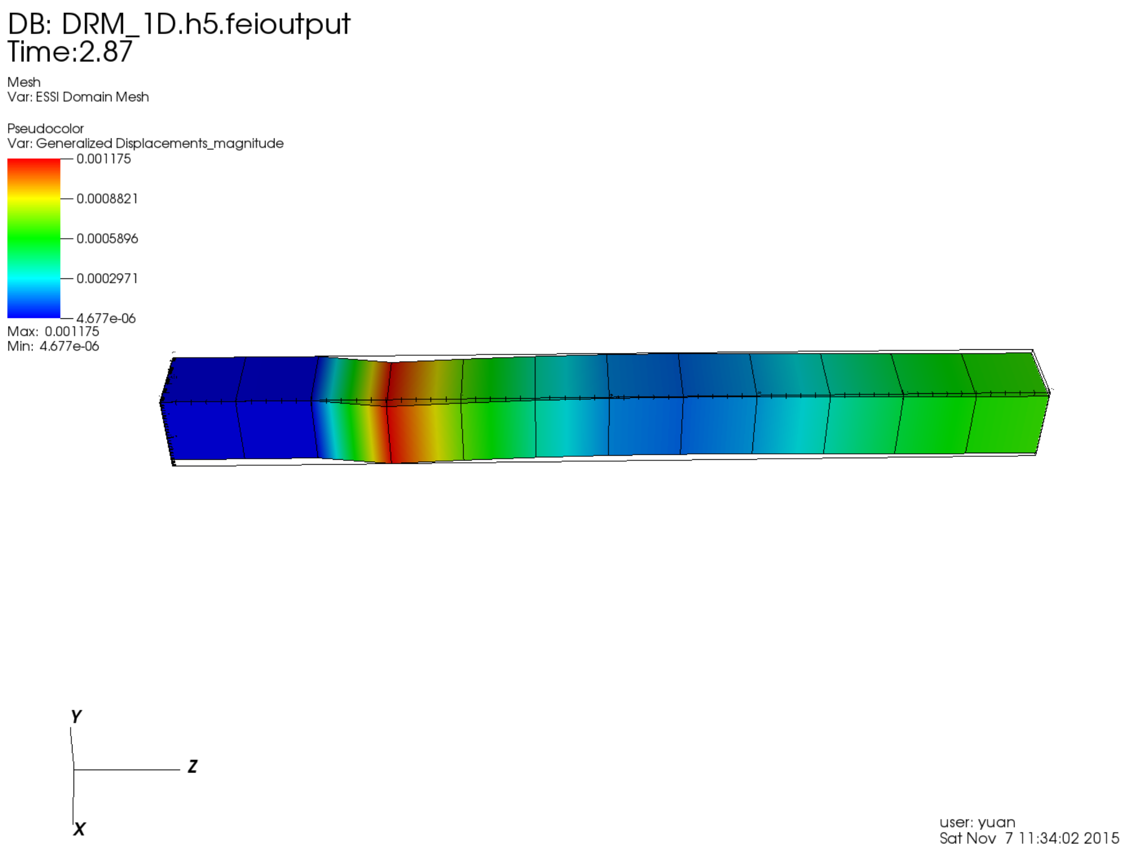


Figure 45: 1D DRM model.

```
model name "DRM" ;

//Material for soil
add material # 1 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;

```

```
//Material for DRM layer
add material # 2 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;

//Material for exterior layer
add material # 3 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;
//
add node # 1 at ( 0.00*m, 0.00*m, 0.00*m) with 3 dofs;
add node # 2 at ( 5.00*m, 0.00*m, 0.00*m) with 3 dofs;
add node # 3 at ( 5.00*m, 5.00*m, 0.00*m) with 3 dofs;
add node # 4 at ( 0.00*m, 5.00*m, 0.00*m) with 3 dofs;
add node # 5 at ( 5.00*m, 0.00*m, 50.00*m) with 3 dofs;
add node # 6 at ( 5.00*m, 0.00*m, 5.00*m) with 3 dofs;
...
...
add node # 52 at ( 0.00*m, 5.00*m, -5.00*m) with 3 dofs;

//
add element #     1 type 8NodeBrickLT with nodes( 1, 4, 3, 2, 24, 44, 34, 6) use material # 1;
add element #     2 type 8NodeBrickLT with nodes( 24, 44, 34, 6, 23, 43, 33, 7) use material
    # 1;
...
add element #    12 type 8NodeBrickLT with nodes( 48, 47, 45, 46, 52, 51, 49, 50) use
    material # 3;

//
fix node # 1 dofs uy ;
fix node # 1 dofs uz ;
fix node # 2 dofs uy ;
fix node # 2 dofs uz ;
fix node # 3 dofs uy ;
fix node # 3 dofs uz ;
fix node # 4 dofs uy ;
fix node # 4 dofs uz ;
...
fix node # 51 dofs ux ;


new loading stage "1D";
add domain reduction method loading # 1
  hdf5_file = "input.hdf5";

define algorithm With_no_convergence_check ;
define solver ProfileSPD;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;
simulate 999 steps using transient algorithm time_step = 0.01*s;

bye;
```

The ESSI model fei files for this example can be downloaded here.

The same model for this example with 27NodeBrickLT can be downloaded here.

Long 1D DRM model 1000:1

To show the wave propagation explicitly, a long 1D model (1000:1) similar to the 1D DRM model above was made in this section.

The model description is same to Fig.(44) except this model use far more soil elements.

The general view is shown in Fig.(46) below.

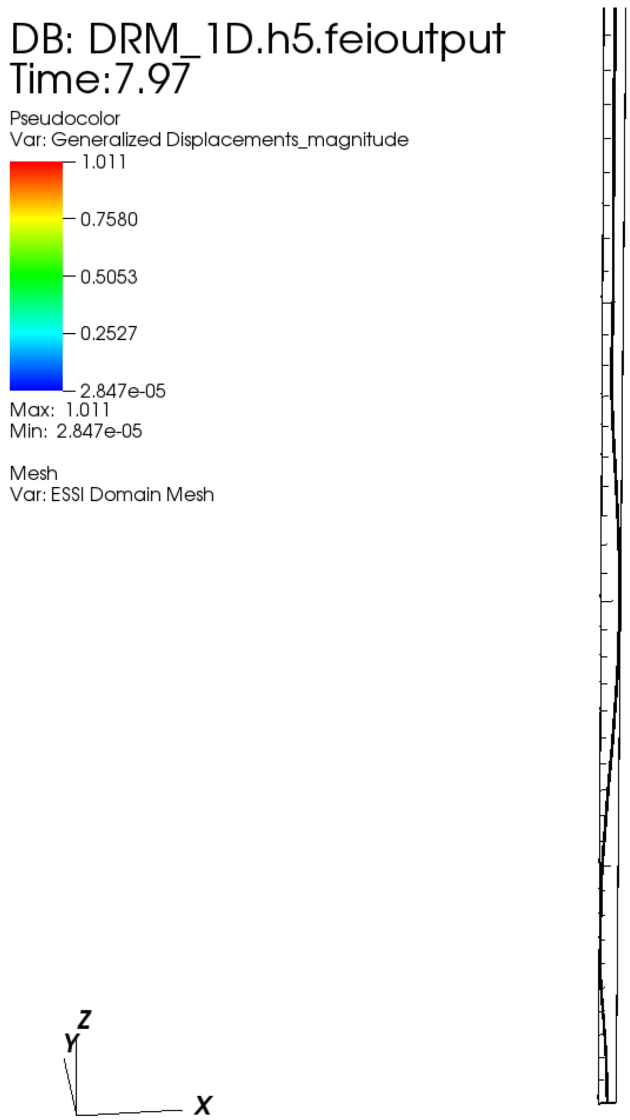


Figure 46: Long 1D DRM model

There is still now outgoing waves at the exterior layers, which is shown in Fig(47).

The ESSI model fei files for this example can be downloaded here.

The results can also be seen from this video.

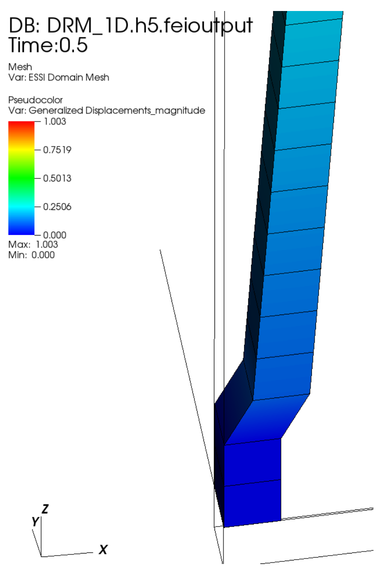


Figure 47: Long 1D DRM model: exterior layer

14 Three Dimensional DRM Model

Problem description:

As shown in Fig.(48), the DRM layer is used to add the earthquake motion.

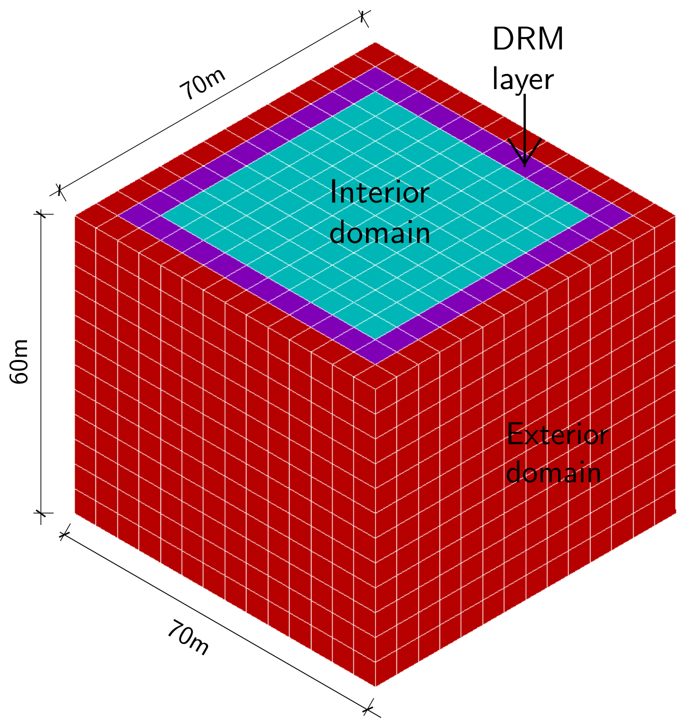


Figure 48: The diagram for 3D Domain Reduction Method example.

Numerical result:

ESSI model fei file:

```
model name "DRM" ;

//Material for soil
add material # 1 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;

//Material for DRM layer
add material # 2 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;

//Material for exterior layer
add material # 3 type linear_elastic_isotropic_3d_LT
  mass_density = 2000*kg/m^3
```

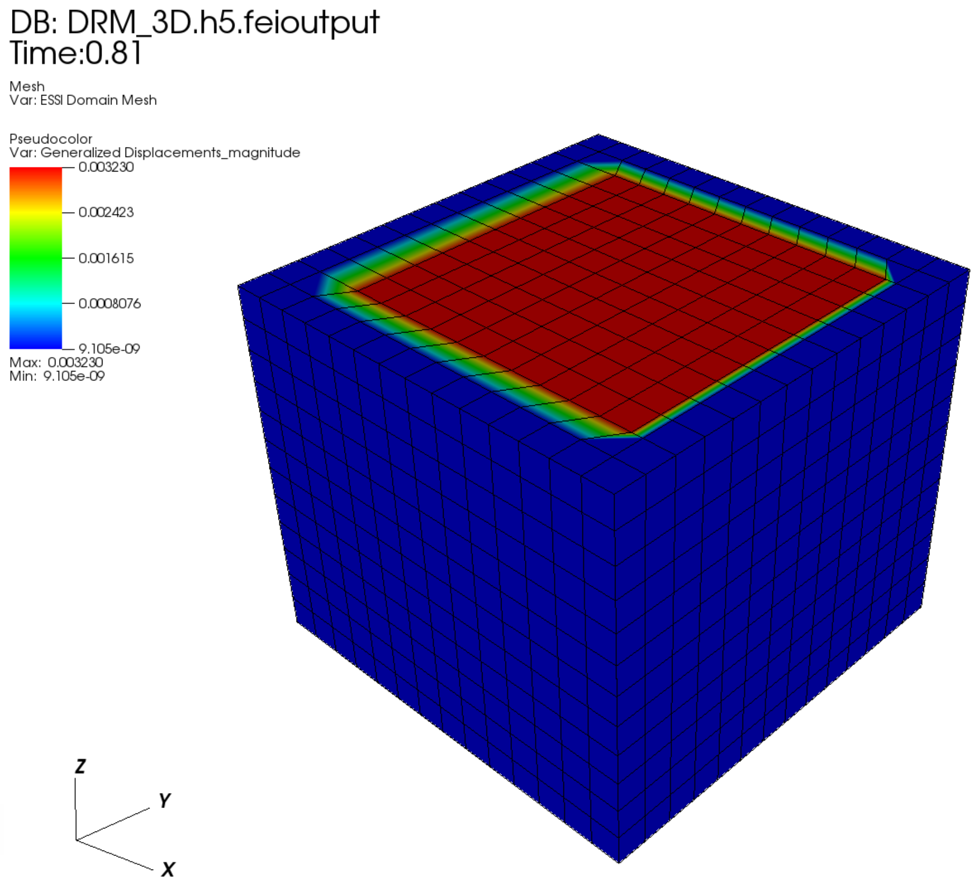


Figure 49: Diagram for the 3D DRM model.

```
  elastic_modulus = 1300*MPa
  poisson_ratio = 0.3;

//
add node # 1 at (  0.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 2 at (  50.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 3 at (  5.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 4 at (  10.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 5 at (  15.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 6 at (  20.00*m,  0.00*m,  0.00*m) with 3 dofs;
add node # 7 at (  25.00*m,  0.00*m,  0.00*m) with 3 dofs;
...
...
add node # 2925 at ( 55.00*m,  55.00*m,  -5.00*m) with 3 dofs;

//
add element # 1 type 8NodeBrickLT with nodes( 1, 40, 41, 3, 150, 441, 603, 151) use material
    # 1;
add element # 2 type 8NodeBrickLT with nodes( 3, 41, 50, 4, 151, 603, 684, 160) use material
    # 1;
...
add element # 2352 type 8NodeBrickLT with nodes( 2925, 2924, 2922, 2923, 2921, 2920, 2918,
    2919) use material # 3;
```

```

//
fix node # 1332 dofs all ;
fix node # 1334 dofs all ;
...
...
fix node # 2924 dofs all ;

new loading stage "3D";
add domain reduction method loading # 1
  hdf5_file = "input.hdf5";

define algorithm With_no_convergence_check ;
define solver ProfileSPD;
define dynamic integrator Newmark with gamma = 0.5 beta = 0.25;

simulate 999 steps using transient algorithm time_step = 0.01*s;

bye;
```

The ESSI model fei files for this example can be downloaded here.
The same model for this example with 27NodeBrickLT can be downloaded here.

15 ShearBeam Element for Pisano Materials

Problem description:

In the element type "ShearBeamLT", only one Gauss point exists. ShearBeamLT element was used here to test the Pisanó material model.

Vertical force F_z was used to apply confinement to the element. Then, cyclic force F_x is used to load. point.

Results

Resulting stress-strain relationship is shown in Fig.(51).

ESSI model fei file:

```
model name "pisanoLT";

add node # 1 at (0*m,0*m,0*m) with 3 dofs;
add node # 2 at (0*m,0*m,1*m) with 3 dofs;

fix node # 1 dofs all;
fix node # 2 dofs uy;

add material # 1 type New_PisanoLT
 mass_density = 2000*kg/m^3
 elastic_modulus_1atm = 325*MPa poisson_ratio = 0.3
 M_in = 1.4  kd_in = 0.0  xi_in = 0.0  h_in = 700  m_in = 0.7
 initial_confining_stress = 0*kPa n_in = 0 a_in = 0.0 eplcum_cr_in = 1e-6;

add element # 1 type ShearBeamLT with nodes (1, 2) \
    cross_section = 1*m^2 use material # 1;
```

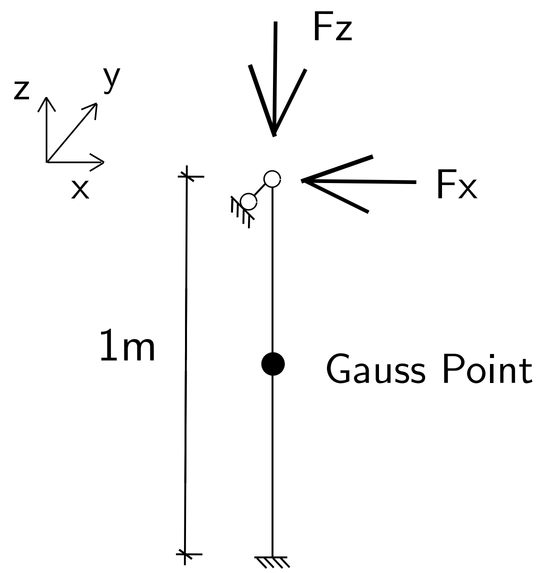


Figure 50: ShearBeam element.

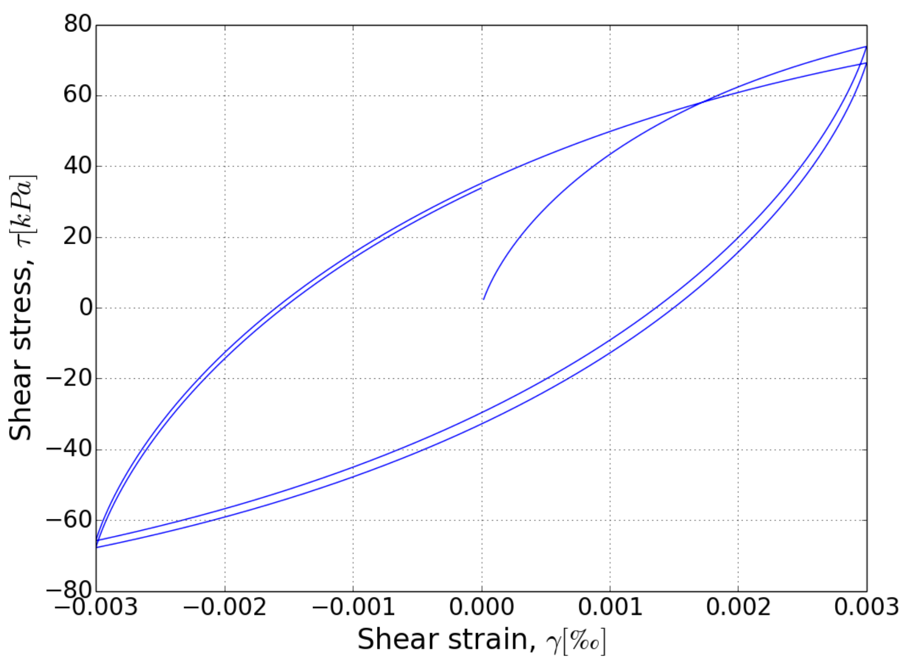


Figure 51: Shear stress-strain response.

```

new loading stage "confinement";

add load # 1 to node # 2 type linear Fz = -200*kN;
```

```
define load factor increment 0.01;
define algorithm With_no_convergence_check ;
define solver UMFPack;
simulate 100 steps using static algorithm;

new loading stage "test01";
gamma_max = 3e-3;
add imposed motion # 2 to node # 2 dof ux
 displacement_scale_unit = gamma_max*m displacement_file = "input_sine.txt"
 velocity_scale_unit = gamma_max*m/s velocity_file = "input_sine.txt"
 acceleration_scale_unit = gamma_max*m/s^2 acceleration_file = "input_sine.txt";

define load factor increment 0.0005;
define algorithm With_no_convergence_check;
define solver UMFPack;
simulate 2000 steps using static algorithm;

bye;
```

The ESSI model fei files for this example can be downloaded here.

16 8NodeBrickLT Element for Drucker Prager Armstrong Federick Material

Problem description:

This example is used to test the materials properties, such as G/Gmax against strains. The element type is 8NodeBrickLT. And there are two stages of loading. The first loading stage is confinement and the second loading stage is shearing.

The boundary condition is specially designed such that each Gauss point has the same stress state.

Results

Resulting stress-strain relationship is shown in Fig.(52).

ESSI model fei file:

```
// Drucker Prager Armstrong Frederick
// This model is created by Jose.
model name "druckeraf";

// Parameters:
phi   = 5;
ha      = 1000;
cr      = 973;
gam        = 0.01;
Ncyc       = 5;
Nsteps     = 1000;
H=1;
vp=1000*m/s;
vs=500*m/s;
rho=2000*kg/m^3;
p0 = 250*kPa;
G = rho*vs^2;
```

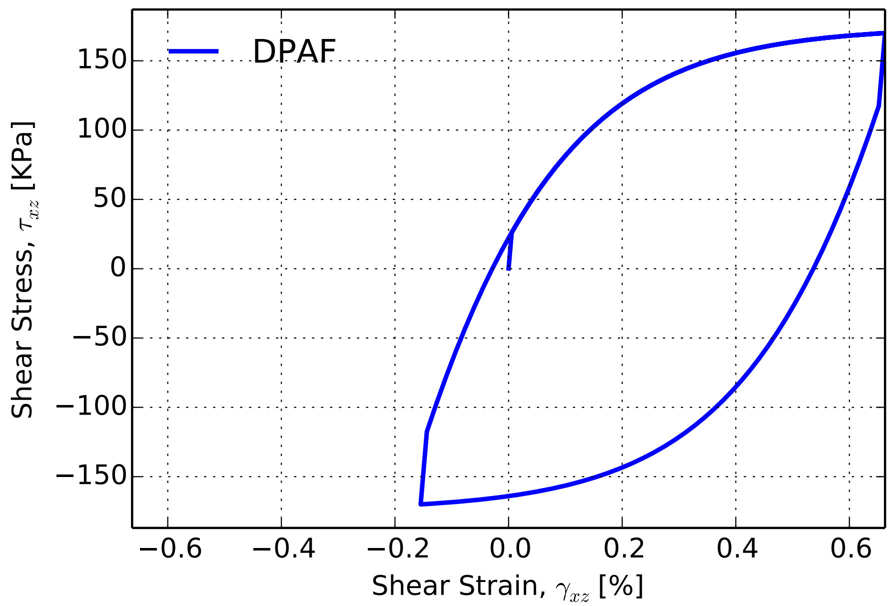


Figure 52: Shear stress-strain response.

```
M = rho*vp^2;

//From wiki (https://en.wikipedia.org/wiki/Elastic_modulus)
E = G*(3*M-4*G)/(M-G);
nu = (M-2*G)/(2*M-2*G);

K0 = 1.0;
phirad = pi*phi/180;
M = 6*sin(phirad)/(3-sin(phirad));

// Define the material:
add material # 1 type DruckerPragerArmstrongFrederickLT
    mass_density = 0*kg/m^3
    elastic_modulus = E
    poisson_ratio = nu
    druckerprager_k = M
    armstrong_frederick_ha = ha*Pa
    armstrong_frederick_cr = cr*Pa
    isotropic_hardening_rate = 0*E
    initial_confining_stress = 1*Pa;

// define the node:
add node # 1 at (0*m,0*m,1*m) with 3 dofs;
add node # 2 at (1*m,0*m,1*m) with 3 dofs;
add node # 3 at (1*m,1*m,1*m) with 3 dofs;
add node # 4 at (0*m,1*m,1*m) with 3 dofs;

add node # 5 at (0*m,0*m,0*m) with 3 dofs;
add node # 6 at (1*m,0*m,0*m) with 3 dofs;
add node # 7 at (1*m,1*m,0*m) with 3 dofs;
add node # 8 at (0*m,1*m,0*m) with 3 dofs;
```

```
// add equal degree of freedom in three directions
add constraint equal dof with master node # 2 and slave node # 3 dof to constrain ux;
add constraint equal dof with master node # 2 and slave node # 6 dof to constrain ux;
add constraint equal dof with master node # 2 and slave node # 7 dof to constrain ux;

add constraint equal dof with master node # 3 and slave node # 4 dof to constrain uy;
add constraint equal dof with master node # 3 and slave node # 8 dof to constrain uy;
add constraint equal dof with master node # 3 and slave node # 7 dof to constrain uy;

add constraint equal dof with master node # 1 and slave node # 2 dof to constrain uz;
add constraint equal dof with master node # 1 and slave node # 3 dof to constrain uz;
add constraint equal dof with master node # 1 and slave node # 4 dof to constrain uz;

// Define the element.
add element # 1 type 8NodeBrickLT with nodes (1, 2,3 , 4, 5, 6,7, 8) use material # 1;

new loading stage "confinement";
fix node # 1 dofs ux uy;
fix node # 2 dofs uy;
fix node # 4 dofs ux;

fix node # 5 dofs ux uy uz;
fix node # 6 dofs uy uz;
fix node # 7 dofs uz;
fix node # 8 dofs ux uz;

sigma_z = -3*p0/(1+2*K0);
sigma_x = K0*sigma_z;
sigma_y = K0*sigma_z;

//Z-face
add load # 1 to node # 1 type linear Fz = sigma_z*m^2/4;
add load # 2 to node # 2 type linear Fz = sigma_z*m^2/4;
add load # 3 to node # 3 type linear Fz = sigma_z*m^2/4;
add load # 4 to node # 4 type linear Fz = sigma_z*m^2/4;

//X-face
add load # 5 to node # 2 type linear Fx = sigma_x*m^2/4;
add load # 6 to node # 6 type linear Fx = sigma_x*m^2/4;
add load # 7 to node # 7 type linear Fx = sigma_x*m^2/4;
add load # 8 to node # 3 type linear Fx = sigma_x*m^2/4;

add load # 9 to node # 3 type linear Fy = sigma_y*m^2/4;
add load # 10 to node # 7 type linear Fy = sigma_y*m^2/4;
add load # 11 to node # 8 type linear Fy = sigma_y*m^2/4;
add load # 12 to node # 4 type linear Fy = sigma_y*m^2/4;

Nsteps_static=100;
define load factor increment 1/Nsteps_static;

define solver UMFPack;
define convergence test Norm_Displacement_Increment
    tolerance = 1e-6
    maximum_iterations = 100
```

```
    verbose_level = 4;
define algorithm Newton ;

define NDMaterialLT constitutive integration algorithm Euler_One_Step
    yield_function_relative_tolerance = 0.002
    stress_relative_tolerance = 0.002
    maximum_iterations = 1000;

simulate Nsteps_static steps using static algorithm;


new loading stage "shearing";
compute reaction forces;
add load # 13 to node # 1 type from_reactions;
add load # 14 to node # 4 type from_reactions;

free node # 1 dofs ux;
free node # 4 dofs ux;
fix node # 3 dofs uy;
fix node # 6 dofs ux;
fix node # 7 dofs ux uy;
fix node # 8 dofs uy;

add constraint equal dof with master node # 1 and slave node # 3 dof to constrain ux;
add constraint equal dof with master node # 1 and slave node # 4 dof to constrain ux;
add constraint equal dof with master node # 1 and slave node # 2 dof to constrain ux;
remove constraint equaldof node # 6;
remove constraint equaldof node # 7;
remove constraint equaldof node # 8;

n = 1;
while(n<=1)
{
    add load # 14+n to node # n type path_time_series
     Fx = 170.*kN
     series_file = "path.txt";
    n+=1;
}

define load factor increment 1/Nsteps;

define solver UMFPack;
define convergence test Norm_Displacement_Increment
    tolerance = 1e-5
    maximum_iterations = 100
    verbose_level = 4;
define algorithm Newton ;

define NDMaterialLT constitutive integration algorithm Euler_One_Step
    yield_function_relative_tolerance = 0.0002
    stress_relative_tolerance = 0.002
    maximum_iterations = 1000;

simulate Ncyc*Nsteps steps using static algorithm;

```

```
bye;
```

The ESSI model fei files for this example can be downloaded here.

17 Beam theory

Problem description: Length=6m, Width=1m, Height=1m, F=100N, E=1E8Pa, $\nu = 0.0$. The force direction was shown in Figure (53).

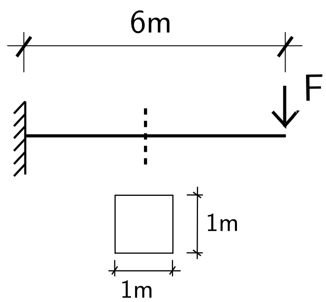


Figure 53: Problem description for cantilever beams

The basic idea to calculate the shear deformation of a beam is

$$\delta = \frac{FL}{GA_v} \tag{1}$$

where A_v is the not the gross cross sectional area of the beam. A_v should be the shear area. Thus,

$$\kappa = \frac{A}{A_v} \tag{2}$$

where κ is the form factor, shear correction factor or shear deformation coefficient, A is the gross sectional area and A_v is the shear area of the section.

The history of κ value is long.

1. Timoshenko (1940) [1] define the form factor for rectangular section is 1.5.
2. Cowper (1970) [2] gave the formula for the form factor:

$$\kappa = \frac{12 + 11\nu}{10(1+\nu)} \tag{3}$$

3. Renton (1991) [3] provided a closed form solution for shear area of rectangular sections. For a rectangular section of depth $2a$ and breadth $2b$.

$$\kappa = \frac{6}{5} + (\frac{\nu}{1+\nu})^2 \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} \frac{144(b/a)^4}{\pi^6 (2m+1)^2 n^2 [(2m+1)^2 (b/2a)^2 + n^2]} \tag{4}$$

[1] Strength of materials, Timoshenko, Krieger Pub Co, 1940

[2] Cowper, G. R. "The shear coefficient in Timoshenko's beam theory." Journal of applied mechanics 33.2 (1966): 335-340.

[3] Renton, J. D. "Generalized beam theory applied to shear stiffness." International Journal of Solids and Structures 27.15 (1991): 1955-1967.

For square cross section, $b = a$, therefore,

$$\kappa = \frac{6}{5} + (\frac{\nu}{1+\nu})^2 \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} \frac{144}{\pi^6 (2m+1)^2 n^2 [(2m+1)^2 (1/2)^2 + n^2]} \tag{5}$$

The summation of the series are very hard. $Matlab$ and $Mathematica$ cannot solve it directly. According to the Renton (1991), the intermediate values are given by

$$\kappa = \frac{6}{5} + C_1 (\frac{\nu}{1+\nu})^2 (\frac{b}{a})^4 \tag{6}$$

When $b = a$, the equation becomes

$$\kappa = \frac{6}{5} + 0.1392 (\frac{\nu}{1+\nu})^2 \tag{7}$$

18 Verification of 8NodeBrick elements

18.1 Verification of 8NodeBrick cantilever beams

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (54).

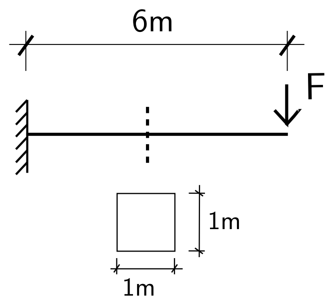
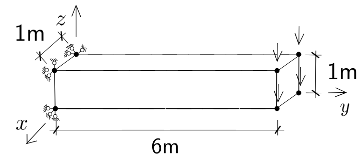


Figure 54: Problem description for cantilever beams

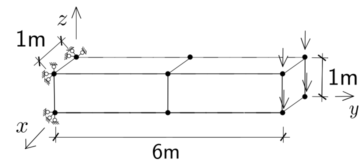
Theoretical displacement (bending and shear deformation):

$$
\begin{aligned}
d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\
&= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\
&= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\
&= 8.784 \times 10^{-4}\ m
\end{aligned}
\tag{8}
$$

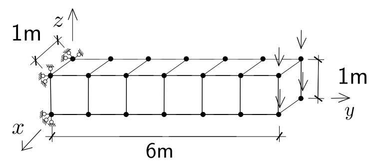
Numerical model:
The 8NodeBrick elements were shown in Figure (55).



(a) One 8NodeBrick element



(b) Two 8NodeBrick elements



(c) Six 8NodeBrick elements

Figure 55: 8NodeBrick elements for cantilever beams

An example ESSI script is shown below.
All the ESSI results were listed in Table (1). The theoretical solution is 8.784E-04 m.

Table 1: Results for 8NodeBrick cantilever beams of different element numbers

Element number	1	2	6
8NodeBrick	4.61E-05 m	1.59E-04 m	5.84E-04 m
Error	94.75%	81.87%	33.52%

The errors were plotted in Figure (56).

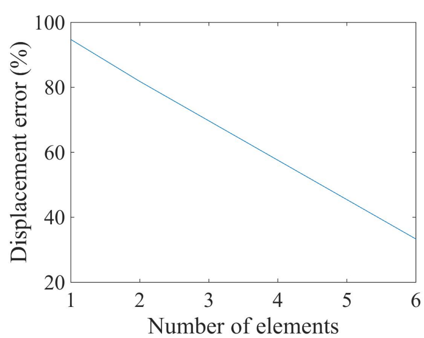


Figure 56: 8NodeBrick cantilever beam for different element number
Displacement error versus Number of elements

The ESSI model fei files for the table above are here

18.2 Verification of 8NodeBrick cantilever beam for different Poisson's ratio

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0 - 0.49$. The force direction was shown in Figure (57).

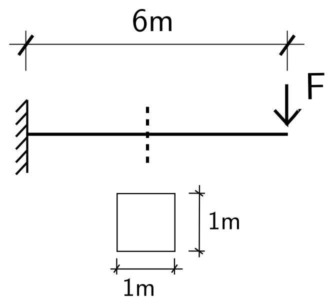


Figure 57: Problem description for cantilever beams of different Poisson's ratios

The theoretical solution for $\nu = 0.0$ was calculated below, while the solution for other Poisson's ratio were calculated by the similar process.

Theoretical displacement (bending and shear deformation):

$$\begin{aligned} d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\ &= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\ &= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\ &= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\ &= 8.784 \times 10^{-4}\ m \end{aligned} \tag{9}$$

The rotation angle at the end:

$$\theta = \frac{FL^2}{2EI} = \frac{100N \times 6^2 m^2}{2 \times 10^8 N/m^2 \times \frac{1}{12} m^4} = 2.16 \times 10^{-4}\ rad = 0.0124° \tag{10}$$

The 8NodeBrick elements for cantilever beams of different Poisson's ratios were shown in Figure (58):

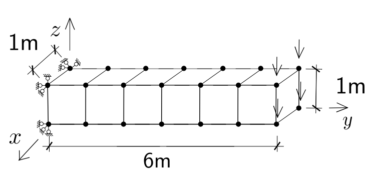


Figure 58: 8NodeBrick elements for cantilever beams of different Poisson's ratios

All the displacement results were listed in Table (2) - (4).

Table 2: ***Displacement*** results for 8NodeBrick cantilever beams with element side length 1 m

Poisson's ratio	8NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	5.840E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	33.52%
0.05	5.924E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	32.62%
0.10	5.969E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	32.16%
0.15	5.971E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	32.20%
0.20	5.922E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	32.81%
0.25	5.814E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	34.09%
0.30	5.634E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	36.19%
0.35	5.364E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	39.29%
0.40	4.970E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	43.80%
0.45	4.353E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	50.82%
0.49	3.142E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	64.52%

Then, in the same geometry, the element side length was cut into 0.5m.

Table 3: ***Displacement*** results for 8NodeBrick cantilever beams with element side length 0.5 m

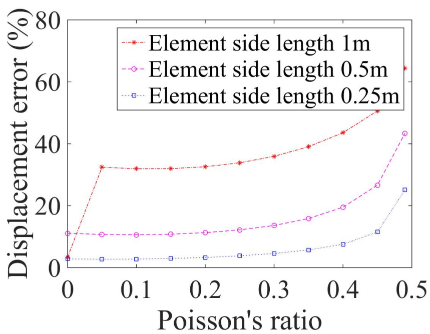
Poisson's ratio	8NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	7.787E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	11.35%
0.05	7.824E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	11.00%
0.10	7.839E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	10.91%
0.15	7.829E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	11.09%
0.20	7.790E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	11.61%
0.25	7.717E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	12.51%
0.30	7.597E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	13.95%
0.35	7.406E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	16.18%
0.40	7.089E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	19.84%
0.45	6.466E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	26.95%
0.49	4.990E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	43.66%

Finally, in the same geometry, the element side length was cut into 0.25m.

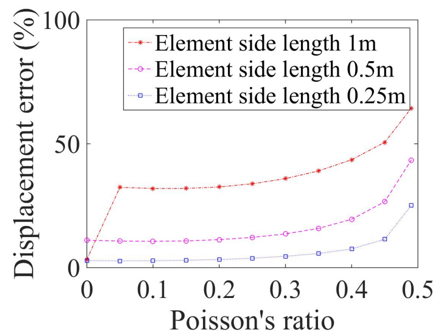
Table 4: ***Displacement*** results for 8NodeBrick cantilever beams with element side length 0.25 m

Poisson's ratio	8NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.511E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	3.11%
0.05	8.525E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	3.03%
0.10	8.527E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	3.09%
0.15	8.518E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	3.27%
0.20	8.494E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	3.62%
0.25	8.455E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	4.15%
0.30	8.393E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	4.93%
0.35	8.299E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	6.08%
0.40	8.141E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	7.94%
0.45	7.801E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	11.86%
0.49	6.603E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	25.45%

The errors were plotted in Figure (59).



(a) Error scale 0% - 80%



(b) Error scale 0% - 100%

Figure 59: 8NodeBrick cantilever beam for different Poisson's ratio ***Displacement error*** versus Poisson's ratio

The angle results were listed in Table (5).

Table 5: ***Rotation angle*** results for 8NodeBrick cantilever beams with element side length 1 m

Poisson's ratio	8NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	8.25E-03	1.24E-02	33.46%
0.05	8.36E-03	1.24E-02	32.55%
0.10	8.42E-03	1.24E-02	32.08%
0.15	8.42E-03	1.24E-02	32.10%
0.20	8.35E-03	1.24E-02	32.67%
0.25	8.20E-03	1.24E-02	33.90%
0.30	7.95E-03	1.24E-02	35.89%
0.35	7.59E-03	1.24E-02	38.83%
0.40	7.07E-03	1.24E-02	43.00%
0.45	6.30E-03	1.24E-02	49.21%
0.49	4.93E-03	1.24E-02	60.20%

Then, in the same geometry, element side length was cut into 0.5m. The angle results were listed in Table (6).

Table 6: ***Rotation angle*** results for 8NodeBrick cantilever beams with with element side length 0.5 m

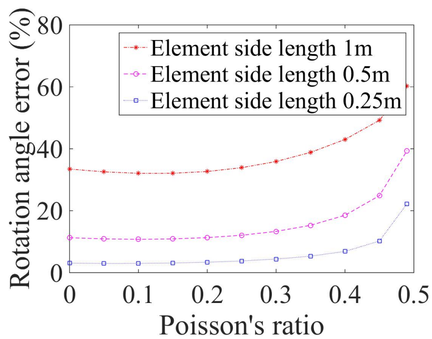
Poisson's ratio	8NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.10E-02	1.24E-02	11.28%
0.05	1.10E-02	1.24E-02	10.91%
0.10	1.11E-02	1.24E-02	10.78%
0.15	1.10E-02	1.24E-02	10.90%
0.20	1.10E-02	1.24E-02	11.32%
0.25	1.09E-02	1.24E-02	12.09%
0.30	1.07E-02	1.24E-02	13.33%
0.35	1.05E-02	1.24E-02	15.29%
0.40	1.01E-02	1.24E-02	18.53%
0.45	9.32E-03	1.24E-02	24.87%
0.49	7.52E-03	1.24E-02	39.35%

Finally, in the same geometry, element side length was cut into 0.25m. The angle results were listed in Table (7).

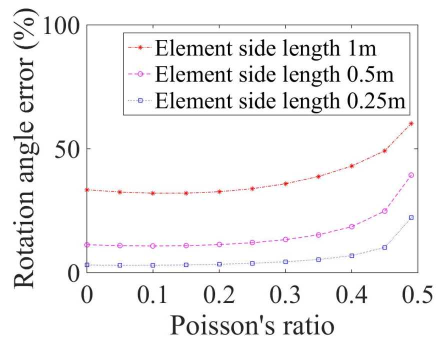
Table 7: ***Rotation angle*** results for 8NodeBrick cantilever beams with with element side length 0.25 m

Poisson's ratio	8NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.20E-02	1.24E-02	3.06%
0.05	1.20E-02	1.24E-02	2.97%
0.10	1.20E-02	1.24E-02	2.99%
0.15	1.20E-02	1.24E-02	3.12%
0.20	1.20E-02	1.24E-02	3.38%
0.25	1.19E-02	1.24E-02	3.79%
0.30	1.19E-02	1.24E-02	4.40%
0.35	1.17E-02	1.24E-02	5.33%
0.40	1.15E-02	1.24E-02	6.87%
0.45	1.11E-02	1.24E-02	10.22%
0.49	9.64E-03	1.24E-02	22.23%

The errors were plotted in Figure (60).



(a) Error scale 30% - 70%



(b) Error scale 0% - 100%

Figure 60: 8NodeBrick cantilever beam for different Poisson's ratio ***Rotation angle error*** versus Poisson's ratio

The ESSI model fei files for the table above are here

18.3 Test of irregular shaped 8NodeBrick cantilever beams

Cantilever model was used as an example. Three different shapes were tested.

In the first test, the upper two nodes of each element were moved one half element size along the $y - axis$, while the lower two nodes were kept at the same location. The element shape was shown in Figure (61).

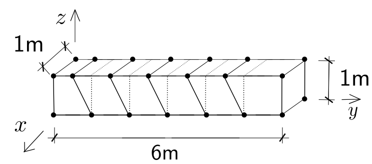


Figure 61: 8NodeBrick cantilever beams for irregular ***Shape 1***

In the second test, the upper two nodes of each element were moved 90% element size along the $y-axis$, while the lower two nodes were moved 90% element size in the other direction along the $y - axis$. The element shape was shown in Figure (62).

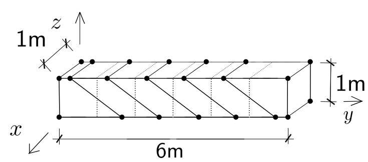


Figure 62: 8NodeBrick cantilever beams for irregular ***Shape 2***

In the third test, the upper two nodes of each element were moved one half element size with different directions along the $y - axis$, while the lower two nodes were kept at the same location. The element shape was shown in Figure (63).

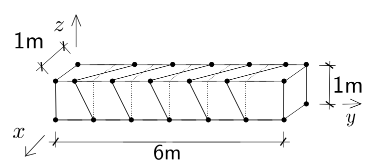
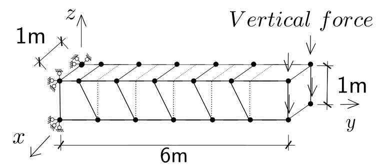
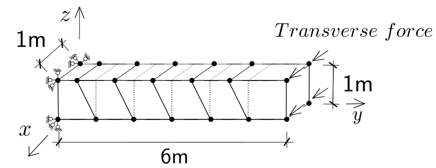


Figure 63: 8NodeBrick cantilever beams for irregular ***Shape 3***

The boundary conditions were shown in Figure (64), (65) and (66) .

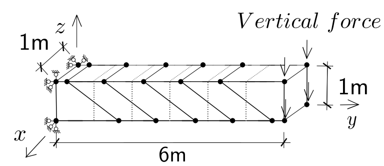


(a) Veritical force

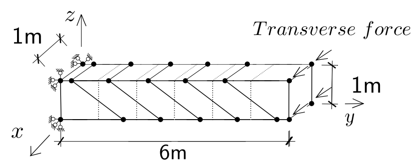


(b) Horizontal force

Figure 64: 8NodeBrick cantilever beam boundary conditions for irregular ***Shape 1***

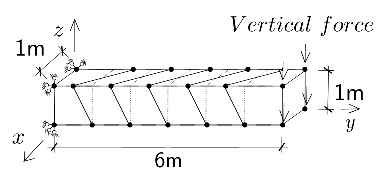


(a) Veritical force

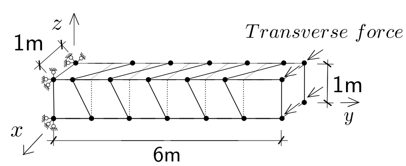


(b) Horizontal force

Figure 65: 8NodeBrick cantilever beam boundary conditions for irregular ***Shape 2***



(a) Veritical force



(b) Horizontal force

Figure 66: 8NodeBrick cantilever beam boundary conditions for irregular ***Shape 3***

The ESSI results were listed in Table (8).

Table 8: Results for 8NodeBrick cantilever beams of irregular shapes

Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
8NodeBrick	Vertical (z)	5.840E-04 m	5.751E-04 m	2.959E-04 m	3.883E-04 m
8NodeBrick	Transverse (y)	5.840E-04 m	4.529E-04 m	1.390E-04 m	4.744E-04 m
Theoretical	-	8.784E-04 m	8.784E-04 m	8.784E-04 m	8.784E-04 m

The errors were listed in Table (9) and (10).

Table 9: Errors for irregular shaped 8NodeBrick compared to theoretical solution

Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
8NodeBrick	Vertical (z)	33.52%	34.53%	66.31%	55.79%
8NodeBrick	Transverse (y)	33.52%	48.44%	84.18%	45.99%

Table 10: Errors for irregular shaped 8NodeBrick compared to normal shape

Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
8NodeBrick	Vertical (z)	0.00%	1.52%	49.33%	33.51%
8NodeBrick	Transverse (y)	0.00%	22.45%	76.20%	18.77%

The ESSI model fei files for the table above are here

Then, the irregular beam was divided into small elements.

Problem description: Length=12m, Width=2m, Height=2m, q=400N/m, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (67).

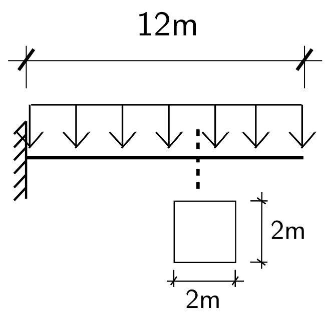


Figure 67: Problem description for cantilever beams under uniform load

Theoretical displacement (bending and shear deformation):

$$
\begin{aligned}
d &= \frac{qL^4}{8EI} + \frac{q\frac{L^2}{2}}{GA_v} \\
&= \frac{qL^4}{8E\frac{bh^3}{12}} + \frac{q\frac{L^2}{2}}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{400N/m \times 12^4 m^4}{8 \times 10^8 N/m^2 \times \frac{2^4}{12} m^4} + \frac{400N/m \times \frac{12^2}{2} m^2}{\frac{10^8}{2} N/m^2 \times 2m \times 2m \times \frac{5}{6}} \\
&= 7.776 \times 10^{-3} m + 1.728 \times 10^{-4} m \\
&= 7.9488 \times 10^{-3} m
\end{aligned}
\tag{11}
$$

The ESSI displacement results were listed in Table (11).

Table 11: Results for 8NodeBrick cantilever beams of irregular shapes with more elements

Element Type	Shape	Force direction	Number of division		
			1	2	4
8NodeBrick	shape1	Vertical (z)	5.37E-03 m	7.08E-03 m	7.71E-03 m
8NodeBrick	shape1	Transverse (y)	4.60E-03 m	6.66E-03 m	7.58E-03 m
8NodeBrick	shape2	Vertical (z)	2.74E-03 m	4.75E-03 m	6.43E-03 m
8NodeBrick	shape2	Transverse (y)	1.46E-03 m	2.72E-03 m	4.63E-03 m
8NodeBrick	shape3	Vertical (z)	9.21E-04 m	6.60E-03 m	7.56E-03 m
8NodeBrick	shape3	Transverse (y)	1.09E-03 m	6.09E-03 m	7.37E-03 m
Theoretical solution			7.95E-03 m	7.95E-03 m	7.95E-03 m

The error were listed in Table (12).

Table 12: Errors for 8NodeBrick cantilever beams of irregular shapes with more elements

Element Type	Shape	Force direction	Number of division		
			1	2	4
8NodeBrick	shape1	Vertical (z)	32.42%	10.95%	3.01%
8NodeBrick	shape1	Transverse (y)	42.16%	16.17%	4.69%
8NodeBrick	shape2	Vertical (z)	65.59%	40.22%	19.05%
8NodeBrick	shape2	Transverse (y)	81.57%	65.76%	41.81%
8NodeBrick	shape3	Vertical (z)	88.42%	16.97%	4.89%
8NodeBrick	shape3	Transverse (y)	86.24%	23.36%	7.28%

The errors were shown in Figure (68), (69) and (70).

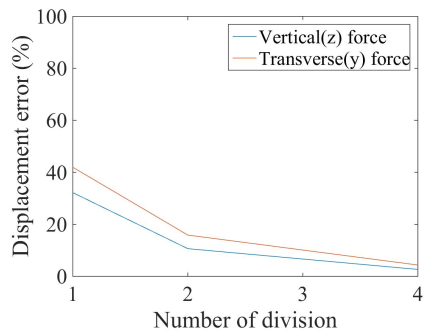


Figure 68: 8NodeBrick cantilever beam for irregular ***Shape 1***
Displacement error versus Number of division

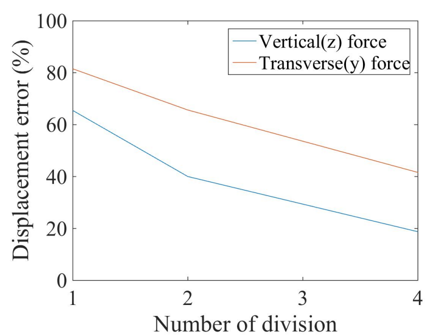


Figure 69: 8NodeBrick cantilever beam for irregular ***Shape 2***
Displacement error versus Number of division

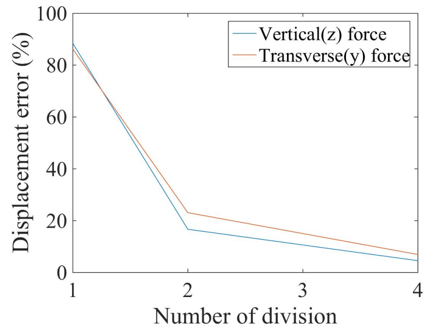


Figure 70: 8NodeBrick cantilever beam for irregular ***Shape 3***
Displacement error versus Number of division

The ESSI model fei files for the table above are here

In this section, the beam was cut into smaller elements with element side length 0.5m and 0.25m respectively. And the element side length of the original models is 1.0m. The numerical models were shown in Figure (71), (72) and (73).

Number of division 1:

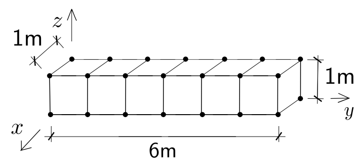


Figure 71: 8NodeBrick clamped beams with element side length 1.0m

Number of division 2:

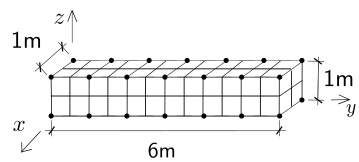


Figure 72: 8NodeBrick clamped beams with element side length 0.5m

Number of division 4:

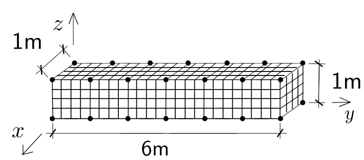


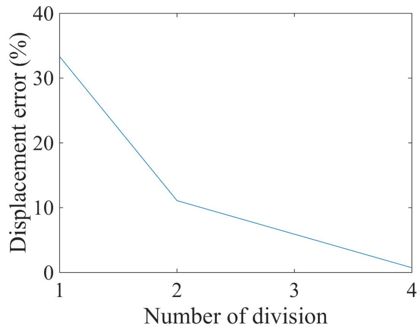
Figure 73: 8NodeBrick clamped beams with element side length 0.25m

The ESSI results were listed in Table (13). The theoretical solution is 1.60E-5 m.

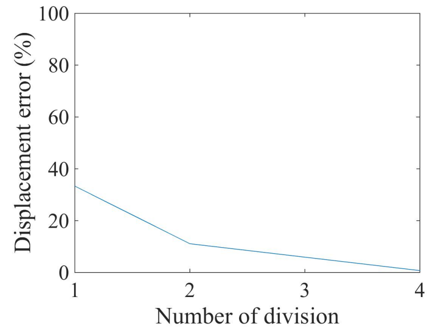
Table 13: Results for 8NodeBrick clamped beams with more elements

Element Type	Element side length		
	1 m	0.5 m	0.25 m
8NodeBrick	1.10E-05 m	1.47E-05 m	1.64E-05 m
Error	33.33%	11.09%	0.73%

The errors were plotted in Figure (74).



(a) Error scale 0% - 40%



(b) Error scale 0% - 100%

Figure 74: 8NodeBrick clamped beam for different element number
Displacement error versus Number of division

The ESSI model fei files for the table above are here

18.4 Verification of 8NodeBrick stress in cantilever beams

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (75).

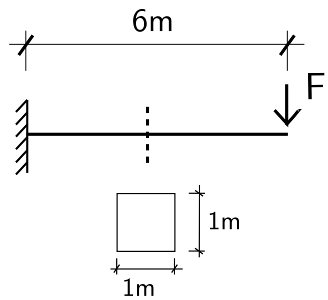


Figure 75: Problem description for cantilever beams of stress verification

The theoretical solution for the stress was calculated below.
The 8NodeBrick elements were shown in Figure (76).

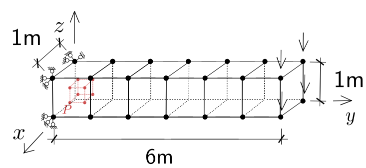


Figure 76: 8NodeBrick for cantilever beams of stress verification

The bending moment at the Gassian Point is

$$M = F(L - P_y) = 100N \times (6 - 0.2113)m = 578.87N \cdot m \tag{12}$$

The bending modulus is

$$I = \frac{bh^3}{12} = \frac{1}{12}m^4 \tag{13}$$

Therefore, the theoretical stress is

$$\sigma = \frac{M \cdot z}{I} = \frac{578.87N \cdot m \times (0.5 - 0.2113)m}{\frac{1}{12}m^4} = 2005Pa \tag{14}$$

To get a better result, the same geometry beam was also cut into small elements. When more elements were used, the theoretical stress was calculated again with the new coordinates. The calculation process is similar to the process above.

The numerical models were shown in Figure (77), (78) and (79).

Number of division 1:

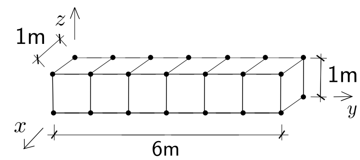


Figure 77: 8NodeBrick stress with element side length 1.0m

Number of division 2:

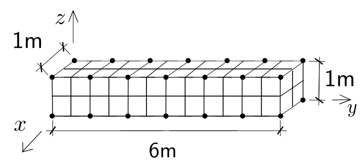


Figure 78: 8NodeBrick stress with element side length 0.5m

Number of division 4:

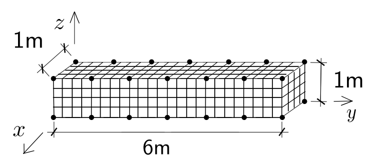


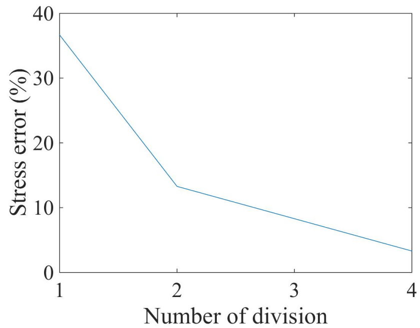
Figure 79: 8NodeBrick stress with element side length 0.25m

All the stress results were listed in Table (14).

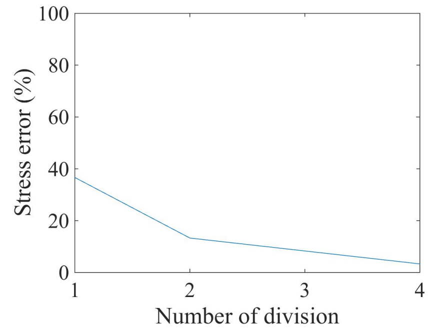
Table 14: Results for 8NodeBrick stress with more elements

Element Type	Element side length		
	1 m	0.5 m	0.25 m
8NodeBrick	1270.17 Pa	2418.60 Pa	3085.48 Pa
Theoretical	2005.26 Pa	2789.23 Pa	3191.27 Pa
Error	36.66%	13.29%	3.31%

The errors were plotted in Figure (80).



(a) Error scale 0% - 40%



(b) Error scale 0% - 100%

Figure 80: 8NodeBrick cantilever beams for stress verification
Stress error versus Number of division

The ESSI model fei files for the table above are here

18.5 Verification of 8NodeBrick square plate with four edges clamped

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{15}$$

The theoretical solution is

$$d = \alpha_c \frac{qa^4}{D} = 0.00406 \times \frac{100 N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \ N \cdot m} = 2.2015 \times 10^{-3} m \tag{16}$$

where α_c is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[4] α_c is 0.00406.

The 8NodeBrick were shown in Figure (81) - (86).

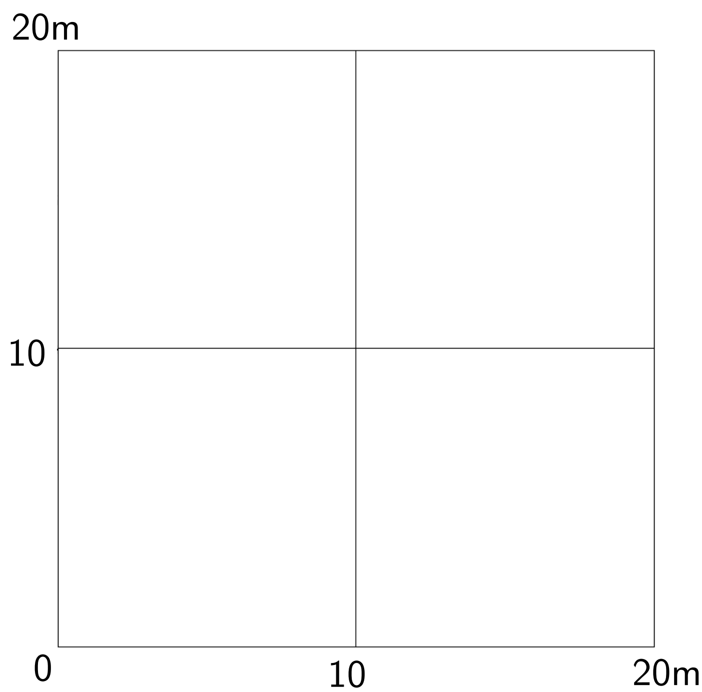


Figure 81: 8NodeBrick edge clamped square plate with element side length 10m

[4]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page120, 1959.

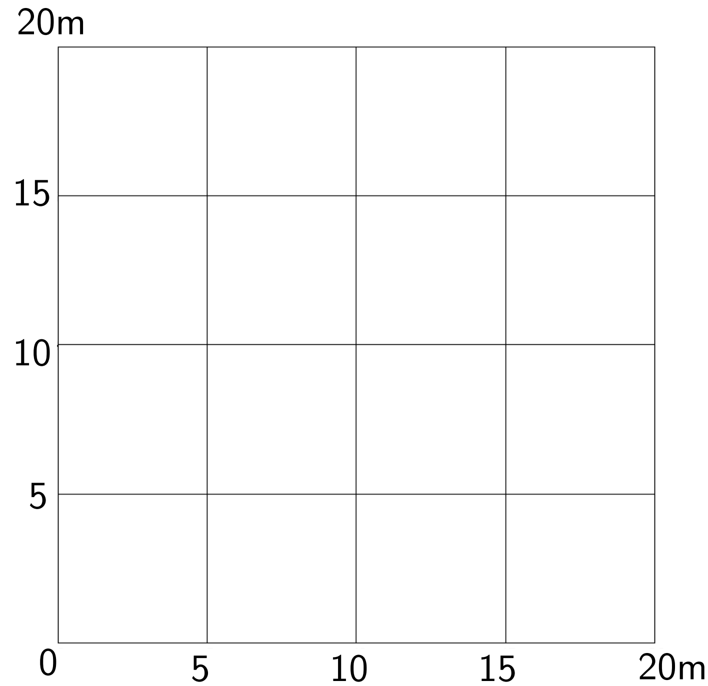


Figure 82: 8NodeBrick edge clamped square plate with element side length 5m

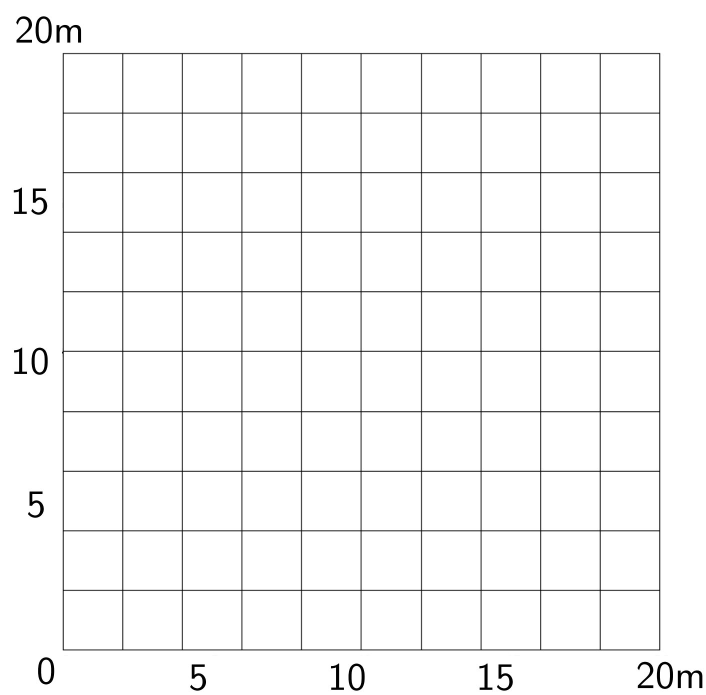


Figure 83: 8NodeBrick edge clamped square plate with element side length 2m

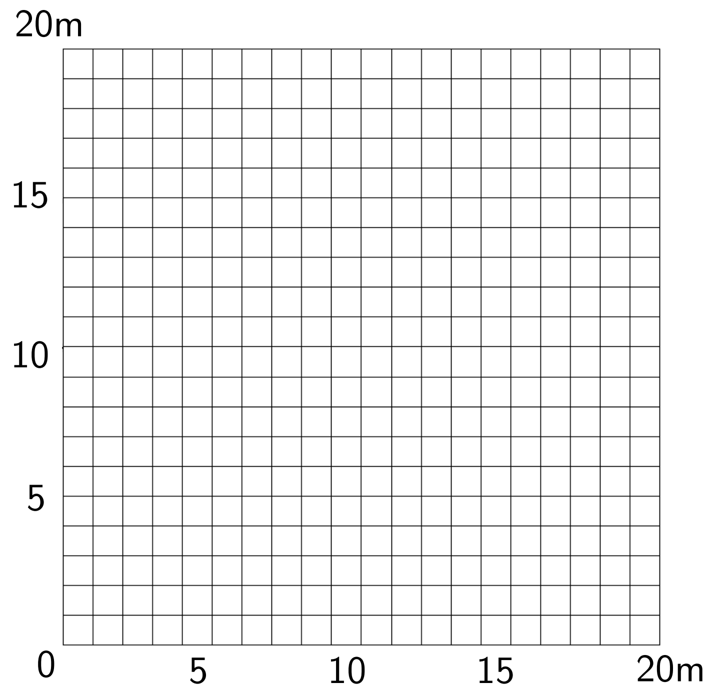


Figure 84: 8NodeBrick edge clamped square plate with element side length 1m

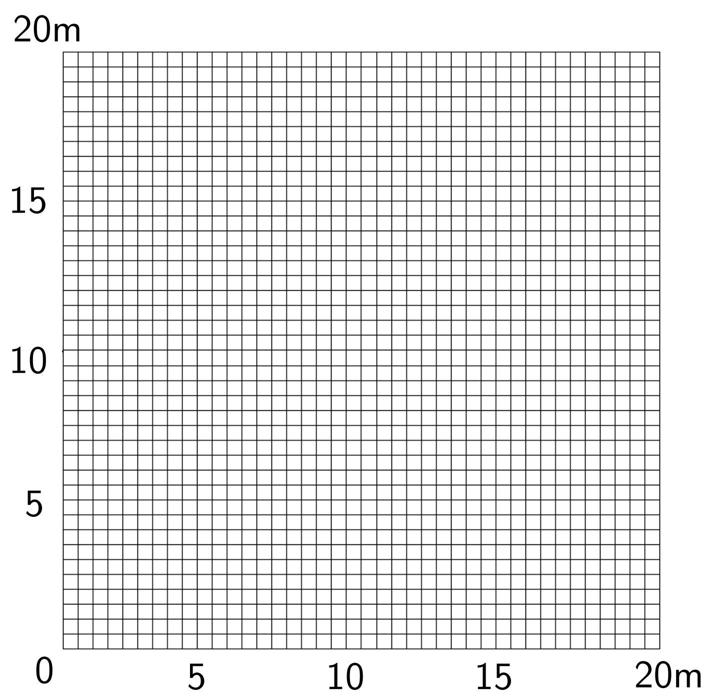


Figure 85: 8NodeBrick edge clamped square plate with element side length 0.5m

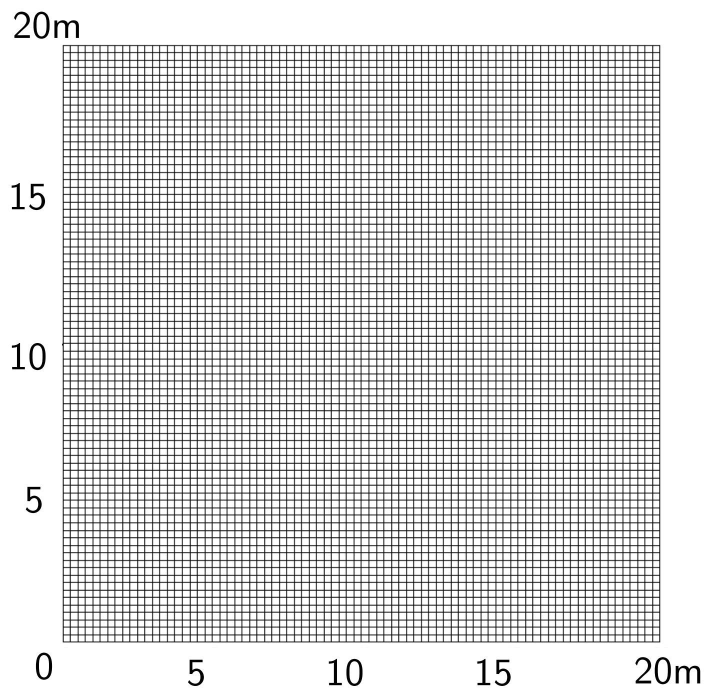


Figure 86: 8NodeBrick edge clamped square plate with element side length 0.25m

The results were listed in Table (15).

Table 15: Results for 8NodeBrick square plate with four edges clamped

Element type	8NodeBrick	8NodeBrick	8NodeBrick	Theoretical displacement
Number of layers	1layer	2layers	4layers	
Element side length	Height:1.00m	Height:0.50m	Height:0.25m	
10m	9.75E-05 m	9.75E-05 m	9.75E-05 m	2.20E-03 m
5m	3.28E-04 m	3.32E-04 m	3.32E-04 m	2.20E-03 m
2m	1.04E-03 m	1.10E-03 m	1.12E-03 m	2.20E-03 m
1m	1.56E-03 m	1.74E-03 m	1.79E-03 m	2.20E-03 m
0.5m	1.80E-03 m	2.30E-03 m	2.12E-03 m	2.20E-03 m
0.25m	1.87E-03 m	2.14E-03 m	2.23E-03 m	2.20E-03 m

The errors were listed in Table (16).

Table 16: Errors for 8NodeBrick square plate with four edges clamped

Element type	8NodeBrick	8NodeBrick	8NodeBrick
Number of layers	1layer	2layers	4layers
Element side length	Height:1.00m	Height:0.50m	Height:0.25m
10m	95.57%	95.57%	95.57%
5m	85.09%	84.94%	84.91%
2m	52.98%	50.09%	49.25%
1m	28.93%	21.17%	18.72%
0.5m	18.26%	4.58%	3.56%
0.25m	15.05%	2.70%	1.37%

The errors were plotted in Figure (87).

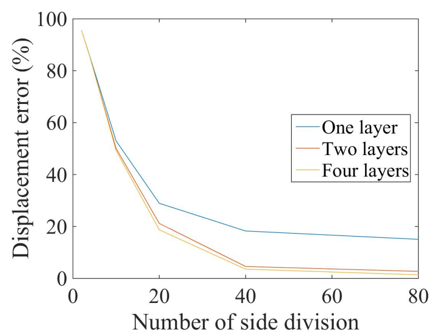


Figure 87: 8NodeBrick square plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

18.6 Verification of 8NodeBrick square plate with four edges simply supported

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{17}$$

The theoretical solution is

$$d = \alpha_s \frac{qa^4}{D} = 0.00126 \times \frac{100N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \ N \cdot m} = 7.0936 \times 10^{-3} m \tag{18}$$

where α_s is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[5] α_s is 0.00126.

The 8NodeBrick were shown in Figure (88) - (93).

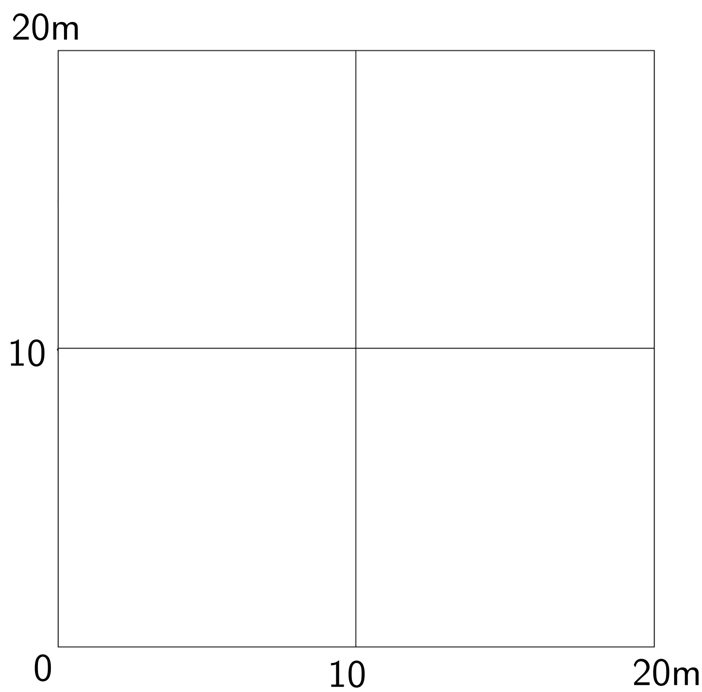


Figure 88: 8NodeBrick edge simply supported square plate with element side length 10m

[5]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page202, 1959.

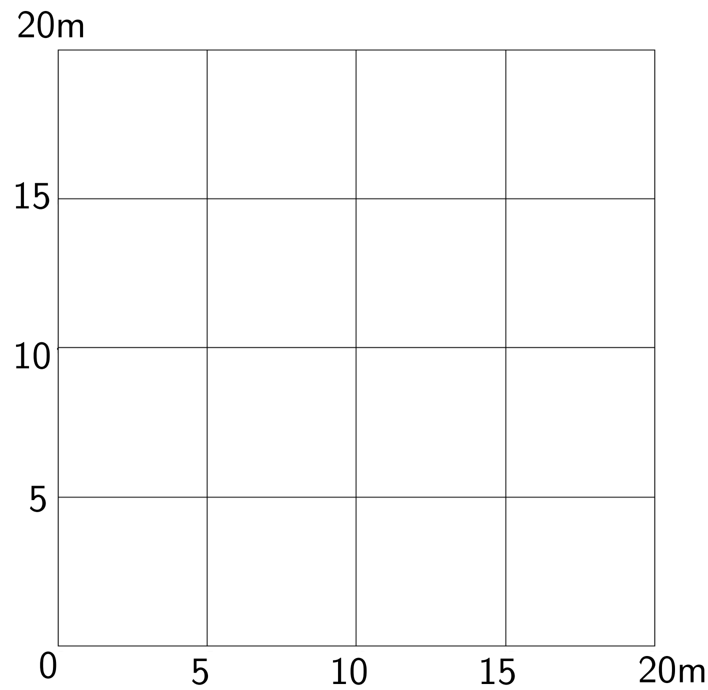


Figure 89: 8NodeBrick edge simply supported square plate with element side length 5m

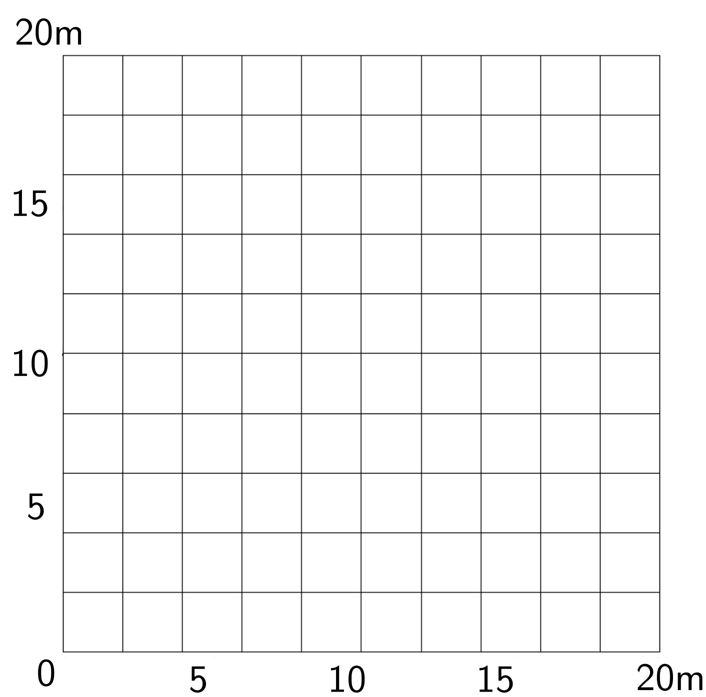


Figure 90: 8NodeBrick edge simply supported square plate with element side length 2m

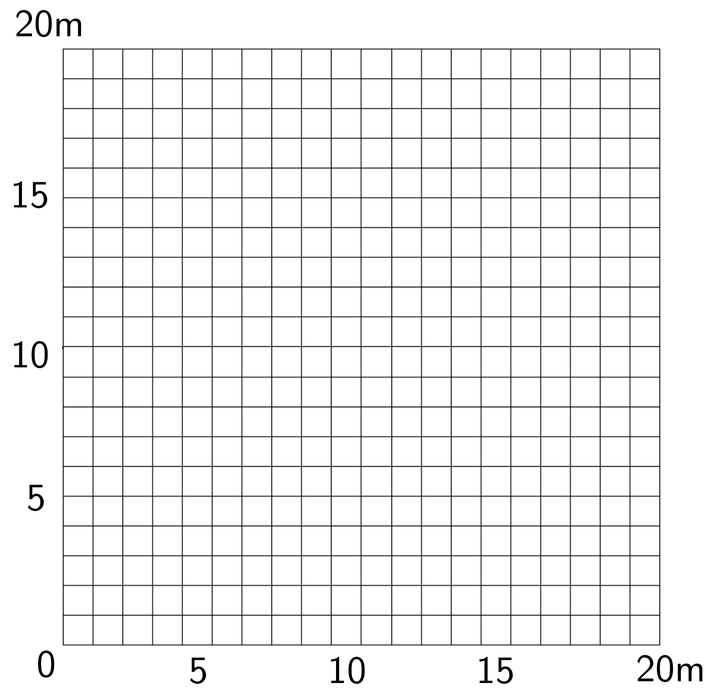


Figure 91: 8NodeBrick edge simply supported square plate with element side length 1m

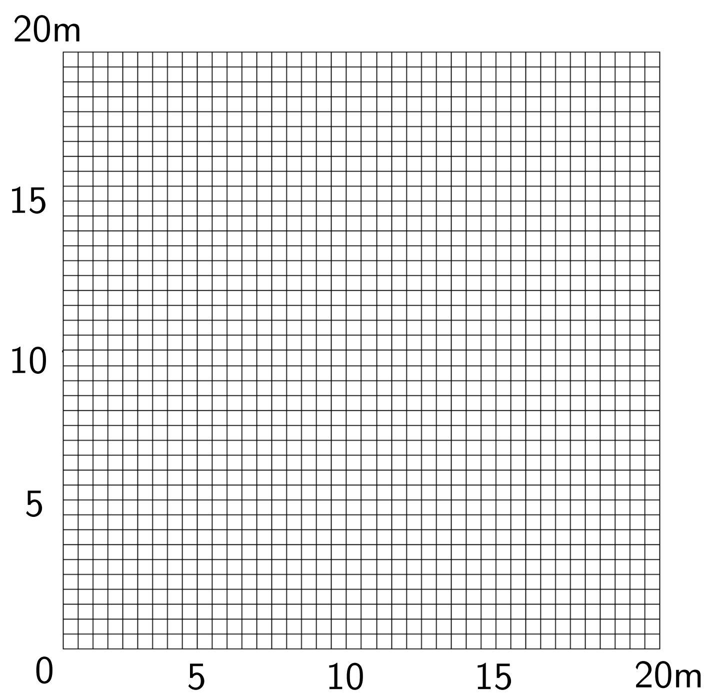


Figure 92: 8NodeBrick edge simply supported square plate with element side length 0.5m

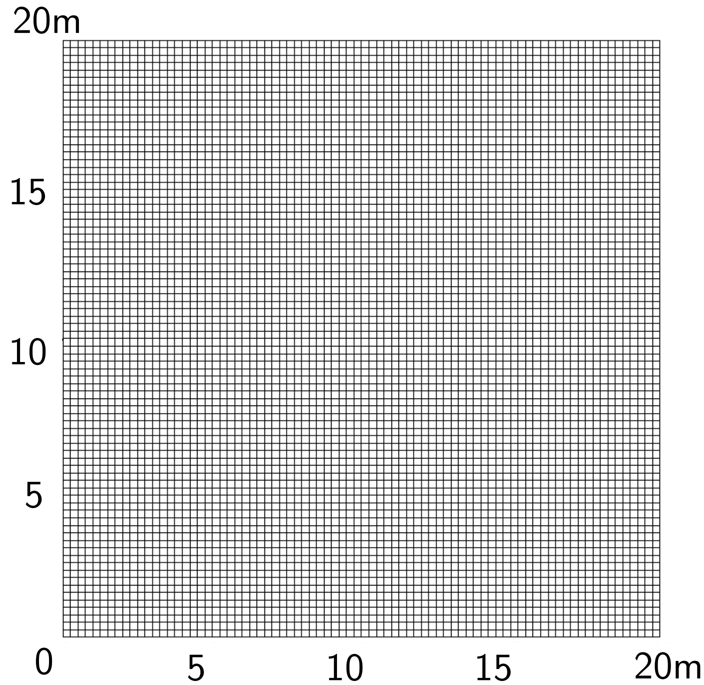


Figure 93: 8NodeBrick edge simply supported square plate with element side length 0.25m

The results were listed in Table (17).

Table 17: Results for 8NodeBrick square plate with four edges simply supported

Element type	8NodeBrick	8NodeBrick	Theoretical displacement
Number of layers	2layers	4layers	
Element side length	Height:0.50m	Height:0.25m	
10m	3.75E-004 m	3.76E-004 m	7.09E-03 m
5m	1.34E-003 m	1.35E-003 m	7.09E-03 m
2m	4.16E-003 m	4.27E-003 m	7.09E-03 m
1m	5.98E-003 m	6.22E-003 m	7.09E-03 m
0.5m	6.75E-003 m	7.04E-003 m	7.09E-03 m
0.25m	8.07E-003 m	7.30E-003 m	7.09E-03 m

The errors were listed in Table (18).

Table 18: Errors for 8NodeBrick square plate with four edges simply supported

Element type	8NodeBrick	8NodeBrick
Number of layers	2layers	4layers
Element side length	Height:0.50m	Height:0.25m
10m	94.72%	94.71%
5m	81.05%	80.91%
2m	41.31%	39.79%
1m	15.64%	12.38%
0.5m	4.88%	0.70%
0.25m	13.74%	2.86%

The errors were plotted in Figure (94).

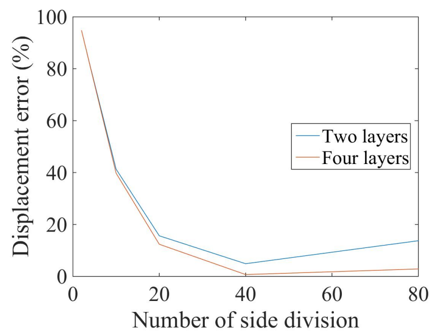


Figure 94: 8NodeBrick square plate with four edges simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

18.7 Verification of 8NodeBrick circular plate with all edges clamped

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{19}$$

The theoretical solution[6] is

$$d = \frac{qa^4}{64D} = \frac{100N/m^2 \times 10^4 m^4}{64 \times 9.1575 \times 10^6 \ N \cdot m} = 1.7106 \times 10^{-3} m \tag{20}$$

The 8NodeBrick were shown in Figure (95) - (100).

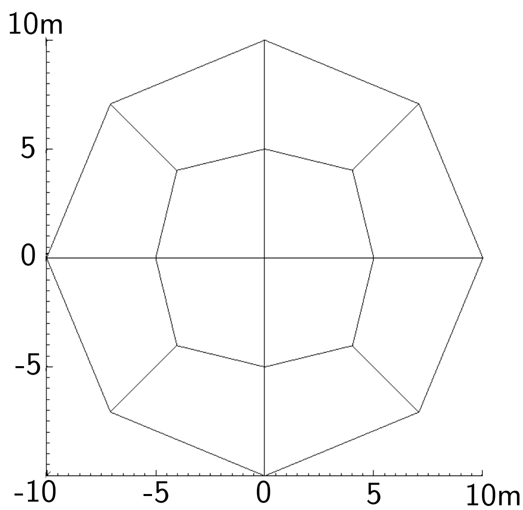


Figure 95: 8NodeBrick edge clamped circular plate with element side length 10m

[6]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

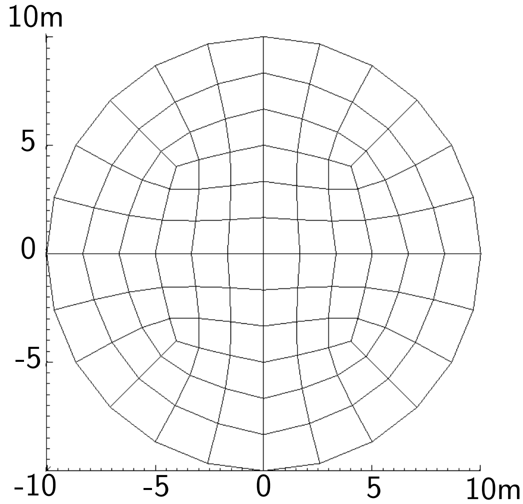


Figure 96: 8NodeBrick edge clamped circular plate with element side length 5m

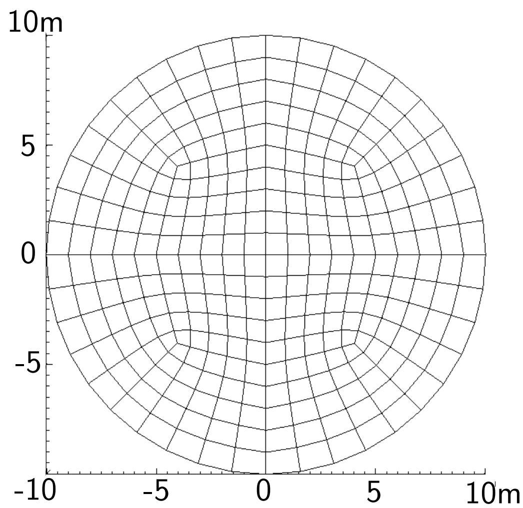


Figure 97: 8NodeBrick edge clamped circular plate with element side length 2m

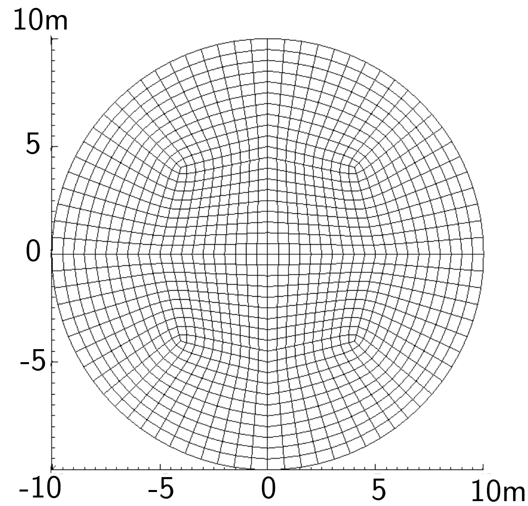


Figure 98: 8NodeBrick edge clamped circular plate with element side length 1m

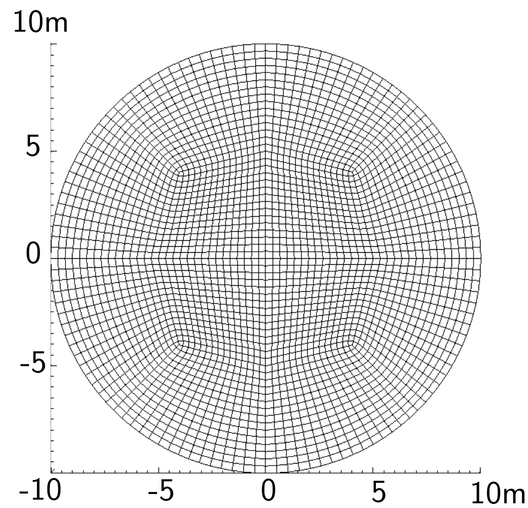


Figure 99: 8NodeBrick edge clamped circular plate with element side length 0.5m

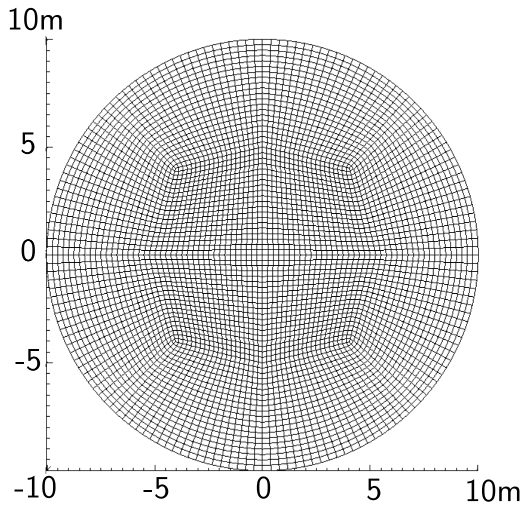


Figure 100: 8NodeBrick edge clamped circular plate with element side length 0.25m

The results were listed in Table (19).

Table 19: Results for 8NodeBrick circular plate with four edges clamped

Element type	8NodeBrick	8NodeBrick	8NodeBrick	Theoretical displacement
Number of layers	1layer	2layers	4layers	
Number of diameter divisions	Height:1.00m	Height:0.50m	Height:0.25m	
4	1.97E-04 m	1.99E-04 m	2.00E-04 m	1.71E-03 m
12	7.95E-04 m	8.47E-04 m	8.62E-04 m	1.71E-03 m
20	1.13E-03 m	1.25E-03 m	1.28E-03 m	1.71E-03 m
40	1.36E-03 m	1.54E-03 m	1.60E-03 m	1.71E-03 m
60	1.41E-03 m	1.62E-03 m	1.68E-03 m	1.71E-03 m
80	1.43E-03 m	1.64E-03 m	1.71E-03 m	1.71E-03 m

The errors were listed in Table (20).

Table 20: Errors for 8NodeBrick circular plate with four edges clamped

Element type	8NodeBrick	8NodeBrick	8NodeBrick
Number of layers	1layer	2layers	4layers
Number of diameter divisions	Height:1.00m	Height:0.50m	Height:0.25m
4	88.43%	88.32%	88.30%
12	53.43%	50.35%	49.47%
20	33.79%	27.00%	24.93%
40	20.14%	9.47%	6.03%
60	17.11%	5.34%	1.51%
80	16.01%	3.80%	0.19%

The errors were shown in Figure (101).

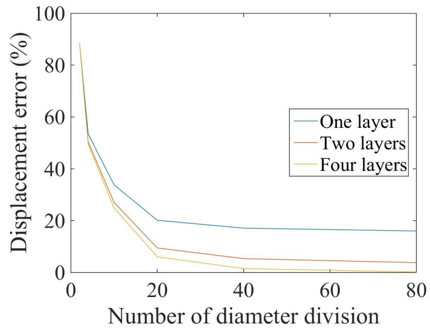


Figure 101: 8NodeBrick circular plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

18.8 Verification of 8NodeBrick circular plate with all edges simply supported

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{21}$$

The theoretical solution[7] is

$$d = \frac{(5+\nu)qa^4}{64(1+\nu)D} = \frac{(5+0.3) \times 100 N/m^2 \times 10^4 m^4}{64 \times (1+0.3) \times 9.1575 \times 10^6 \ N \cdot m} = 6.956 \times 10^{-3} m \tag{22}$$

The 8NodeBrick were shown in Figure (102) - (107).

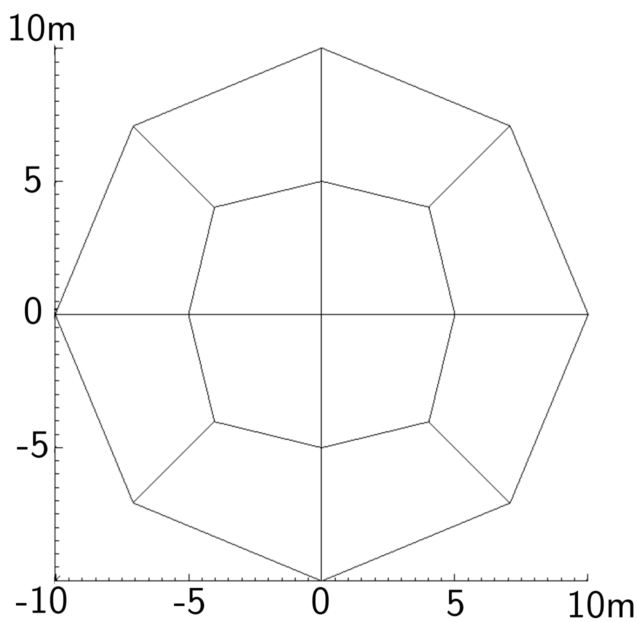


Figure 102: 8NodeBrick edge simply supported circular plate with element side length 10m

[7]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

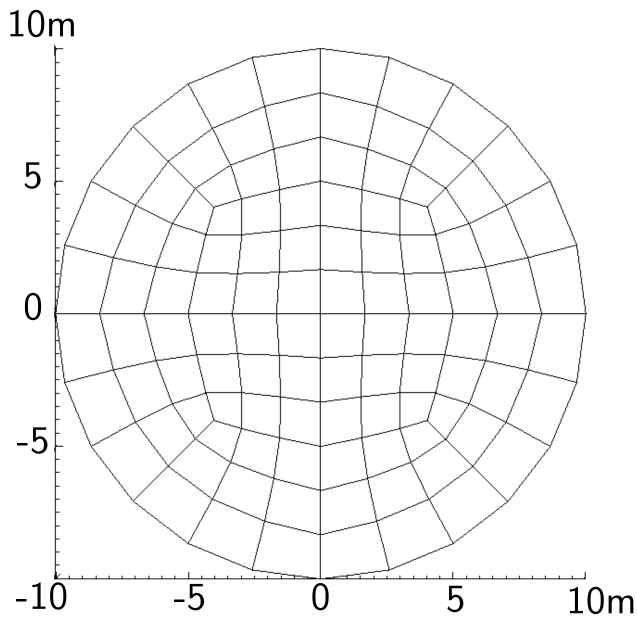


Figure 103: 8NodeBrick edge simply supported circular plate with element side length 5m

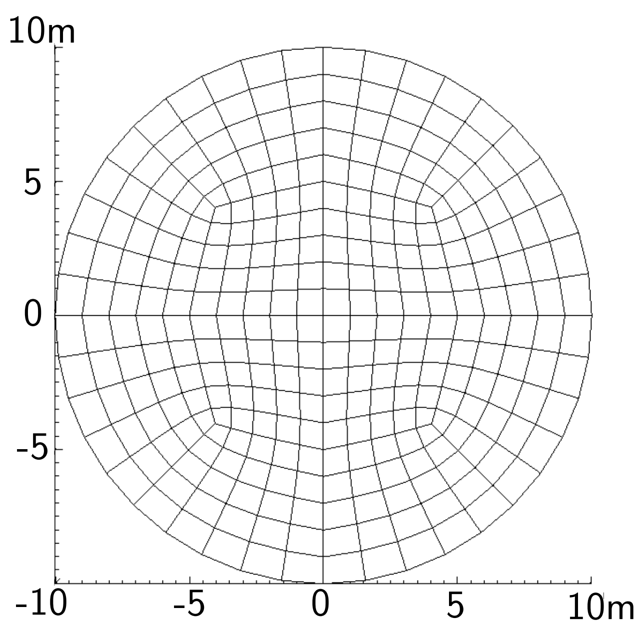


Figure 104: 8NodeBrick edge simply supported circular plate with element side length 2m

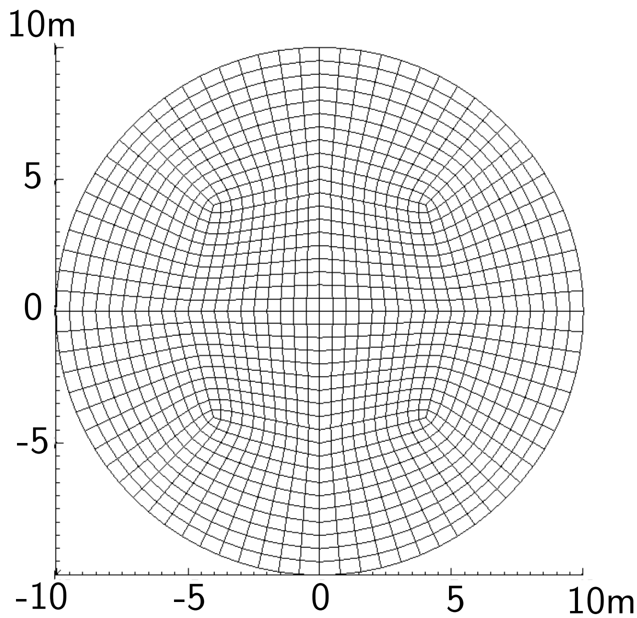


Figure 105: 8NodeBrick edge simply supported circular plate with element side length 1m

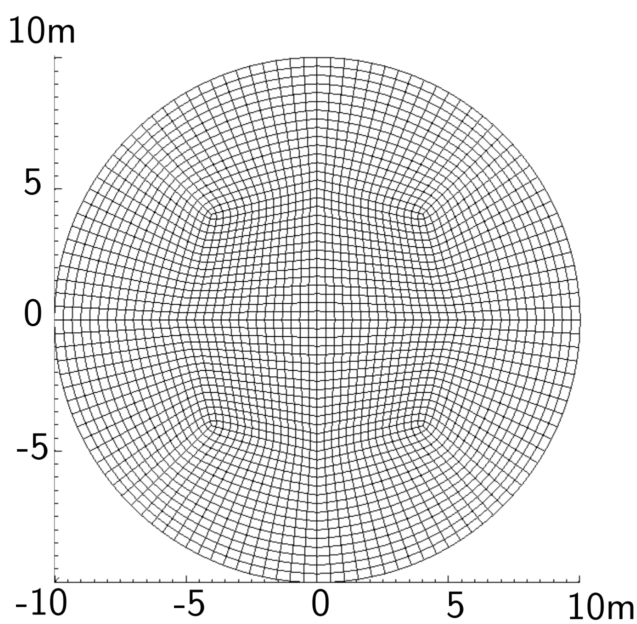


Figure 106: 8NodeBrick edge simply supported circular plate with element side length 0.5m

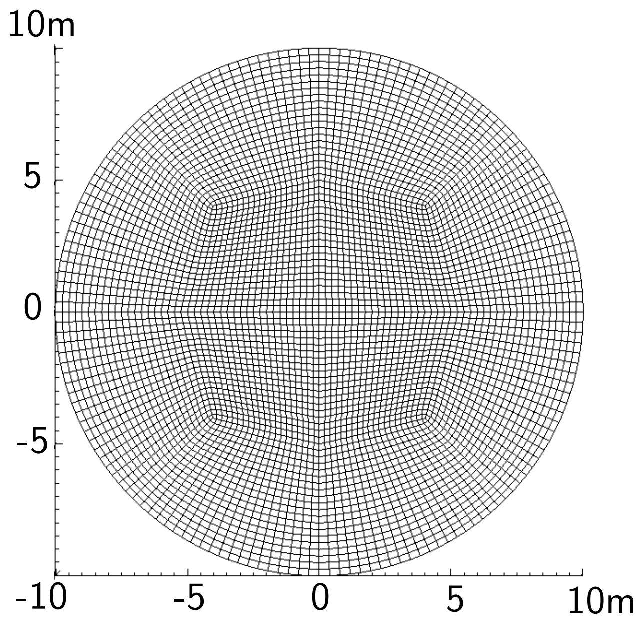


Figure 107: 8NodeBrick edge simply supported circular plate with element side length 0.25m

The results were listed in Table (21).

Table 21: Results for 8NodeBrick cicular plate with four edges simply supported

Element type	8NodeBrick	8NodeBrick	Theoretical displacement
Number of layers	2layers	4layers	
Number of diameter divisions	Height:0.50m	Height:0.25m	
4	6.35E-04 m	6.39E-04 m	6.96E-03 m
12	3.46E-03 m	3.57E-03 m	6.96E-03 m
20	4.96E-03 m	5.18E-03 m	6.96E-03 m
40	6.05E-03 m	6.37E-03 m	6.96E-03 m
60	6.30E-03 m	6.65E-03 m	6.96E-03 m
80	6.39E-03 m	6.76E-03 m	6.96E-03 m

The errors were listed in Table (22).

Table 22: Errors for 8NodeBrick cicular plate with four edges simply supported

Element type	8NodeBrick	8NodeBrick
Number of layers	2layers	4layers
Number of diameter divisions	Height:0.50m	Height:0.25m
4	90.87%	90.82%
12	50.19%	48.65%
20	28.64%	25.47%
40	13.09%	8.40%
60	9.45%	4.36%
80	8.10%	2.85%

The errors were plotted in Figure (108).

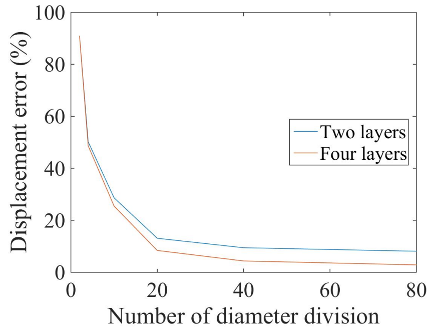


Figure 108: 8NodeBrick circular plate with edge simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here.

19 Verification of 27NodeBrick elements

19.1 Verification of 27NodeBrick cantilever beams

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (109).

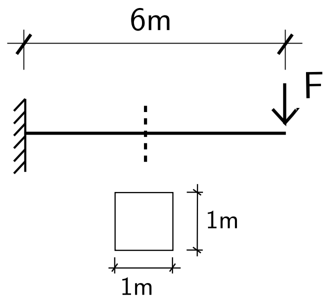
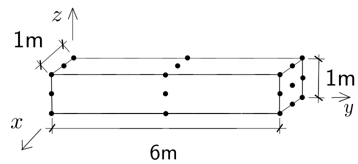


Figure 109: Problem description for cantilever beams

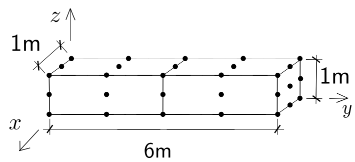
Theoretical displacement (bending and shear deformation):

$$\begin{aligned} d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\ &= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\ &= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\ &= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\ &= 8.784 \times 10^{-4}\ m \end{aligned} \tag{23}$$

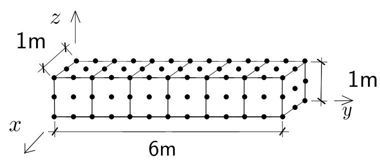
Numerical model:
The 27NodeBrick elements were shown in Figure (110).



(a) One 27NodeBrick element



(b) Two 27NodeBrick elements



(c) Six 27NodeBrick elements

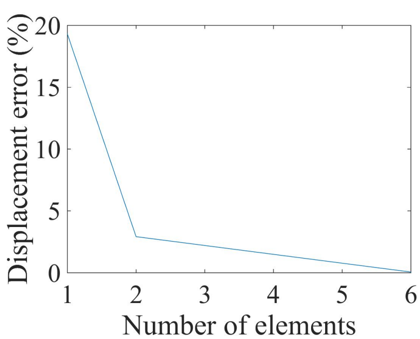
Figure 110: 27NodeBrick elements for cantilever beams

All the ESSI results were listed in Table (23).

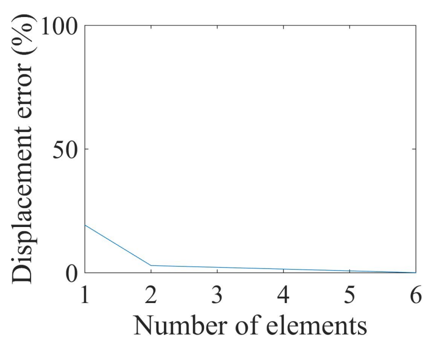
Table 23: Results for 27NodeBrick cantilever beams of different element numbers

Element number	1	2	6
27NodeBrick	7.07E-04 m	8.50E-04 m	8.75E-04 m
Error	19.52%	3.19%	0.34%

The errors were plotted in Figure (111).



(a) Error scale 0% - 20%



(b) Error scale 0% - 100%

Figure 111: 27NodeBrick cantilever beam for different element number
Displacement error versus Number of elements

The ESSI model fei files for the table above are here

19.2 Verification of 27NodeBrick cantilever beam for different Poisson's ratio

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0 - 0.49$. The force direction was shown in Figure (112).

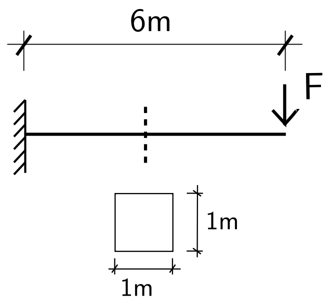


Figure 112: Problem description for cantilever beams of different Poisson's ratios

The theoretical solution for $\nu = 0.0$ was calculated below, while the solution for other Poisson's ratio were calculated by the similar process.

Theoretical displacement (bending and shear deformation):

$$\begin{aligned} d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\ &= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\ &= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\ &= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\ &= 8.784 \times 10^{-4}\ m \end{aligned} \tag{24}$$

The rotation angle at the end:

$$\theta = \frac{FL^2}{2EI} = \frac{100N \times 6^2 m^2}{2 \times 10^8 N/m^2 \times \frac{1}{12} m^4} = 2.16 \times 10^{-4}\ rad = 0.0124° \tag{25}$$

The 27NodeBrick elements for cantilever beams of different Poisson's ratios were shown in Figure (113):

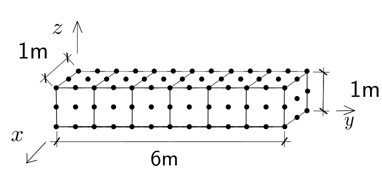


Figure 113: 27NodeBrick elements for cantilever beams of different Poisson's ratios

All the displacement results were listed in Table (24).

Table 24: ***Displacement*** results for 27NodeBrick cantilever beams with element side length 1 m

Poisson's ratio	27NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.755E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.34%
0.05	8.757E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.39%
0.10	8.751E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.54%
0.15	8.735E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.80%
0.20	8.708E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	1.19%
0.25	8.667E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	1.74%
0.30	8.608E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	2.50%
0.35	8.520E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	3.57%
0.40	8.385E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	5.18%
0.45	8.147E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	7.96%
0.49	7.711E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	12.94%

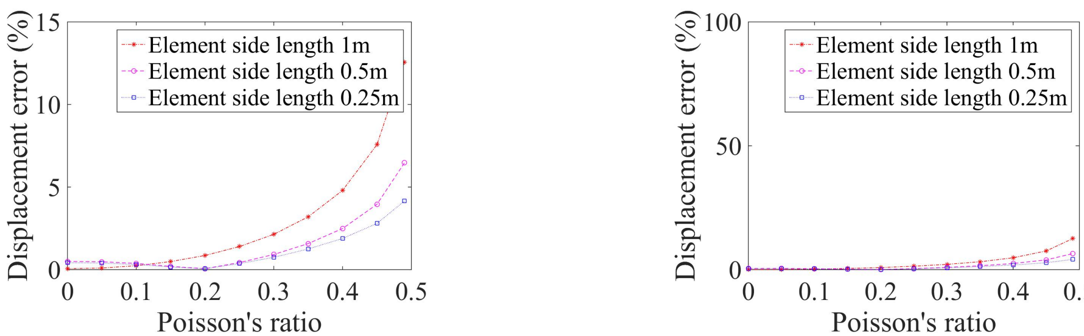
Table 25: ***Displacement*** results for 27NodeBrick cantilever beams with element side length 0.5 m

Poisson's ratio	27NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.804E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.23%
0.05	8.808E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.19%
0.10	8.805E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.08%
0.15	8.796E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.12%
0.20	8.778E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	0.40%
0.25	8.752E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	0.78%
0.30	8.715E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	1.28%
0.35	8.663E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	1.95%
0.40	8.588E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	2.89%
0.45	8.465E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	4.36%
0.49	8.248E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	6.88%

Table 26: ***Displacement*** results for 27NodeBrick cantilever beams
with element side length 0.25 m

Poisson's ratio	27NodeBrick displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.797E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.15%
0.05	8.801E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.11%
0.10	8.799E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.01%
0.15	8.792E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.16%
0.20	8.778E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	0.40%
0.25	8.758E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	0.71%
0.30	8.730E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	1.12%
0.35	8.692E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	1.63%
0.40	8.641E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	2.29%
0.45	8.567E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	3.21%
0.49	8.452E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	4.58%

The errors were plotted in Figure (114).



(a) Error scale 0% - 15% (b) Error scale 0% - 100%

Figure 114: 27NodeBrick cantilever beam for different Poisson's ratio
Displacement error versus Poisson's ratio

The angle results were listed in Table (27).

Table 27: ***Rotation angle*** results for 27NodeBrick cantilever beams with element side length 1 m

Poisson's ratio	27NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.238E-02	1.24E-02	0.19%
0.05	1.237E-02	1.24E-02	0.24%
0.10	1.236E-02	1.24E-02	0.34%
0.15	1.233E-02	1.24E-02	0.53%
0.20	1.230E-02	1.24E-02	0.80%
0.25	1.225E-02	1.24E-02	1.18%
0.30	1.219E-02	1.24E-02	1.70%
0.35	1.210E-02	1.24E-02	2.45%
0.40	1.196E-02	1.24E-02	3.55%
0.45	1.172E-02	1.24E-02	5.47%
0.49	1.130E-02	1.24E-02	8.89%

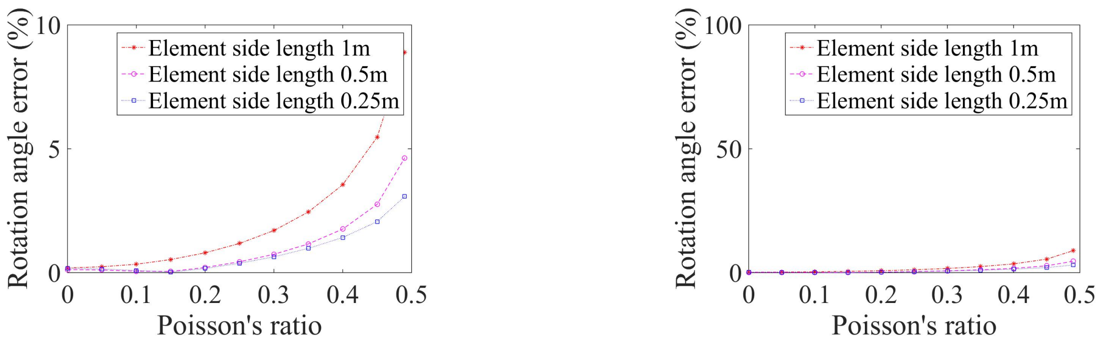
Table 28: ***Rotation angle*** results for 27NodeBrick cantilever beams with element side length 0.5 m

Poisson's ratio	27NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.242E-02	1.24E-02	0.12%
0.05	1.241E-02	1.24E-02	0.11%
0.10	1.241E-02	1.24E-02	0.06%
0.15	1.239E-02	1.24E-02	0.05%
0.20	1.237E-02	1.24E-02	0.21%
0.25	1.235E-02	1.24E-02	0.44%
0.30	1.231E-02	1.24E-02	0.74%
0.35	1.226E-02	1.24E-02	1.16%
0.40	1.218E-02	1.24E-02	1.76%
0.45	1.206E-02	1.24E-02	2.76%
0.49	1.183E-02	1.24E-02	4.63%

Table 29: ***Rotation angle*** results for 27NodeBrick cantilever beams with element side length 0.25 m

Poisson's ratio	27NodeBrick angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.242E-02	1.24E-02	0.17%
0.05	1.242E-02	1.24E-02	0.15%
0.10	1.241E-02	1.24E-02	0.09%
0.15	1.240E-02	1.24E-02	0.02%
0.20	1.238E-02	1.24E-02	0.17%
0.25	1.235E-02	1.24E-02	0.38%
0.30	1.232E-02	1.24E-02	0.64%
0.35	1.228E-02	1.24E-02	0.98%
0.40	1.222E-02	1.24E-02	1.42%
0.45	1.214E-02	1.24E-02	2.06%
0.49	1.202E-02	1.24E-02	3.08%

The errors were plotted in Figure (115).



(a) Error scale 0% - 10% (b) Error scale 0% - 100%

Figure 115: 27NodeBrick cantilever beam for different Poisson's ratio ***Rotation angle error*** versus Poisson's ratio

The ESSI model fei files for the table above are here

Then, different values of elastic modulus were also tried. The errors were plotted below.

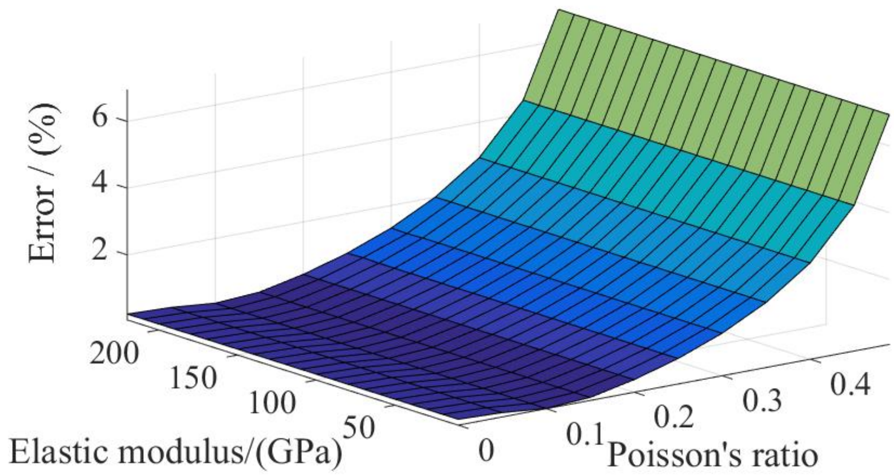


Figure 116: The influence of Poisson's ratio and elastic modulus on the errors

According to Fig.(116)), the different values of elastic modulus will not influence the error.

However, the different Poisson's ratio will influence the error. The error will increase with the Poisson's ratio increase.

19.3 Test of irregular shaped 27NodeBrick cantilever beams

Cantilever model was used as an example. Three different shapes were tested.

In the first test, the upper two nodes of each element were moved one half element size along the $y-axis$, while the lower two nodes were kept at the same location. The element shape was shown in Figure (117).

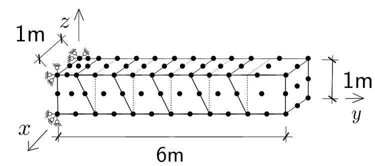


Figure 117: 27NodeBrick cantilever beams for irregular ***Shape 1***

In the second test, the upper two nodes of each element were moved 90% element size along the $y-axis$, while the lower two nodes were moved 90% element size in the other direction along the $y-axis$. The element shape was shown in Figure (118).

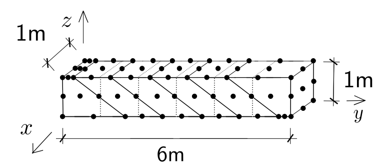


Figure 118: 27NodeBrick cantilever beams for irregular ***Shape 2***

In the third test, the upper two nodes of each element were moved one half element size with different directions along the $y-axis$, while the lower two nodes were kept at the same location. The element shape was shown in Figure (119).

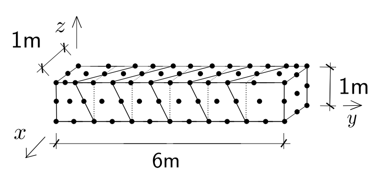
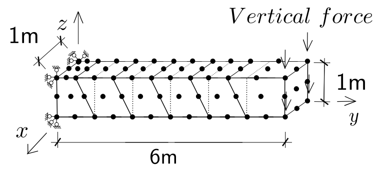
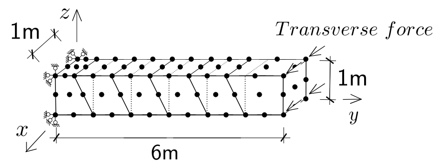


Figure 119: 27NodeBrick cantilever beams for irregular ***Shape 3***

The boundary conditions were shown in Figure (120), (121) and (122) .

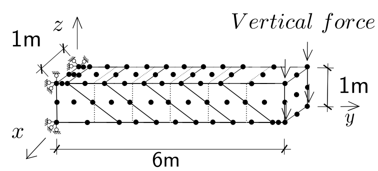


(a) Veritical force

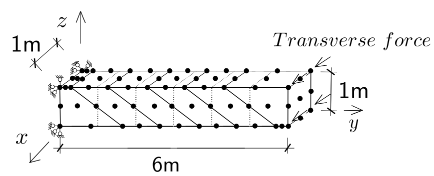


(b) Horizontal force

Figure 120: 27NodeBrick cantilever beam boundary conditions for irregular ***Shape 1***

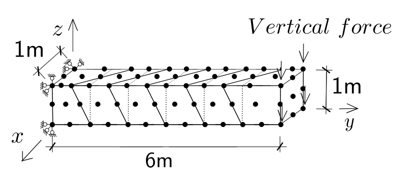


(a) Veritical force

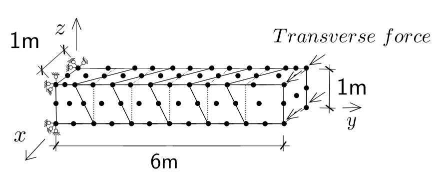


(b) Horizontal force

Figure 121: 27NodeBrick cantilever beam boundary conditions for irregular ***Shape 2***



(a) Veritical force



(b) Horizontal force

Figure 122: 27NodeBrick cantilever beam boundary conditions for irregular ***Shape 3***

The ESSI results were listed in Table (30).

Table 30: Results for 27NodeBrick cantilever beams of irregular shapes

Displacements for irregular shaped element					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
27NodeBrick	Vertical (z)	8.755E-04 m	8.819E-04 m	8.709E-04 m	8.837E-04 m
27NodeBrick	Transverse (y)	8.755E-04 m	8.831E-04 m	8.462E-04 m	8.824E-04 m
Theoretical	-	8.784E-04 m	8.784E-04 m	8.784E-04 m	8.784E-04 m

The errors were listed in Table (31) and (32).

Table 31: Errors for irregular shaped 27NodeBrick compared to theoretical solution

Errors for irregular shaped element, compared to theoretical solutions					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
27NodeBrick	Vertical (z)	0.34%	0.40%	0.85%	0.60%
27NodeBrick	Transverse (y)	0.34%	0.54%	3.67%	0.46%

Table 32: Errors for irregular shaped 27NodeBrick compared to normal shape

Errors for irregular shaped element, compared to normal shape					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
27NodeBrick	Vertical (z)	0.00%	0.74%	0.52%	0.94%
27NodeBrick	Transverse (y)	0.00%	0.87%	3.34%	0.79%

The ESSI model fei files for the table above are here

Then, the beam was divided into small elements.

Problem description: Length=12m, Width=2m, Height=2m, Force=400N/m, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (123).

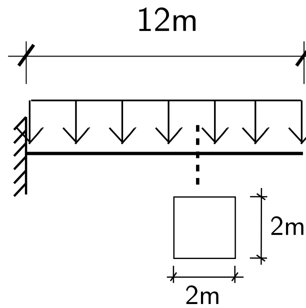


Figure 123: Problem description for cantilever beams under uniform pressure

Theoretical displacement (bending and shear deformation):

$$\begin{aligned}
d &= \frac{qL^4}{8EI} + \frac{q\frac{L^2}{2}}{GA_v} \\
&= \frac{qL^4}{8E\frac{bh^3}{12}} + \frac{q\frac{L^2}{2}}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{400N/m \times 12^4 m^4}{8 \times 10^8 N/m^2 \times \frac{2^4}{12} m^4} + \frac{400N/m \times \frac{12^2}{2} m^2}{\frac{10^8}{2} N/m^2 \times 2m \times 2m \times \frac{5}{6}} \\
&= 7.776 \times 10^{-3} m + 1.728 \times 10^{-4} m \\
&= 7.9488 \times 10^{-3} m
\end{aligned} \tag{26}$$

The ESSI displacement results were listed in Table (33).

Table 33: Results for 27NodeBrick cantilever beams of irregular shapes with more elements

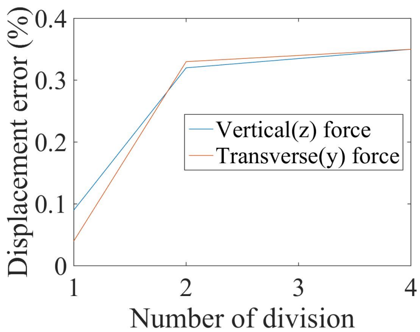
Element Type	Shape	Force direction	Number of division		
			1	2	4
27NodeBrick	shape1	Vertical (z)	7.913E-03 m	7.946E-03 m	7.948E-03 m
27NodeBrick	shape1	Transverse (y)	7.903E-03 m	7.946E-03 m	7.948E-03 m
27NodeBrick	shape2	Vertical (z)	7.741E-03 m	7.930E-03 m	7.947E-03 m
27NodeBrick	shape2	Transverse (y)	7.371E-03 m	7.894E-03 m	7.944E-03 m
27NodeBrick	shape3	Vertical (z)	1.982E-03 m	7.946E-03 m	7.948E-03 m
27NodeBrick	shape3	Transverse (y)	1.979E-03 m	7.947E-03 m	7.948E-03 m
Theoretical solution			7.9488E-03 m	7.9488E-03 m	7.9488E-03 m

The error were listed in Table (34).

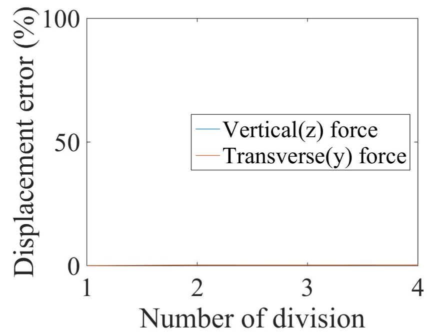
Table 34: Errors for 27NodeBrick cantilever beams of irregular shapes with more elements

Element Type	Shape	Force direction	Number of division		
			1	2	4
27NodeBrick	shape1	Vertical (z)	0.45%	0.04%	0.01%
27NodeBrick	shape1	Transverse (y)	0.32%	0.03%	0.01%
27NodeBrick	shape2	Vertical (z)	2.61%	0.23%	0.03%
27NodeBrick	shape2	Transverse (y)	7.27%	0.69%	0.06%
27NodeBrick	shape3	Vertical (z)	75.06%	0.04%	0.01%
27NodeBrick	shape3	Transverse (y)	75.11%	0.03%	0.01%

The errors were shown in Figure (124), (125) and (126).

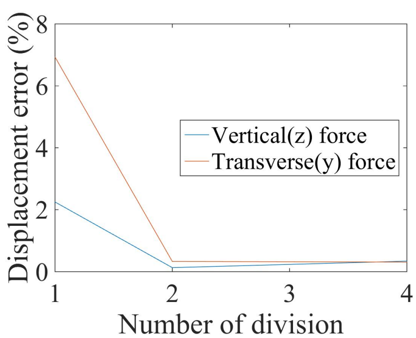


(a) Error scale 0% - 0.4%

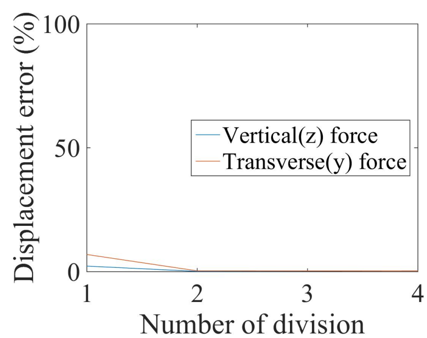


(b) Error scale 0% - 100%

Figure 124: 27NodeBrick cantilever beam for irregular ***Shape 1***
Displacement error versus Number of division

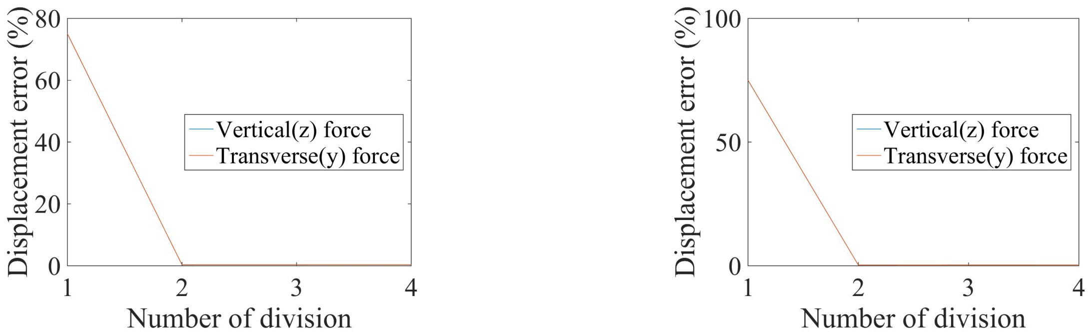


(a) Error scale 0% - 8%



(b) Error scale 0% - 100%

Figure 125: 27NodeBrick cantilever beam for irregular ***Shape 2***
Displacement error versus Number of division



(a) Error scale 0% - 80%

(b) Error scale 0% - 100%

Figure 126: 27NodeBrick cantilever beam for irregular ***Shape 3***
Displacement error versus Number of division

The ESSI model fei files for the table above are here

In this section, the beam was cut into smaller elements with element side length 0.5m and 0.25m respectively. And the element side length of the original models is 1.0m. The numerical models were shown in Figure (127), (128) and (129).

Number of division 1:

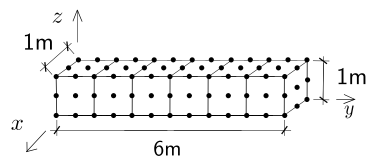


Figure 127: 27NodeBrick clamped beams with element side length 1.0m

Number of division 2:

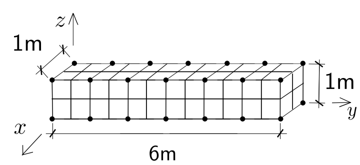


Figure 128: 27NodeBrick clamped beams with element side length 0.5m

Number of division 4:

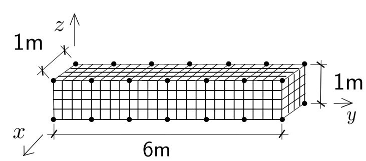


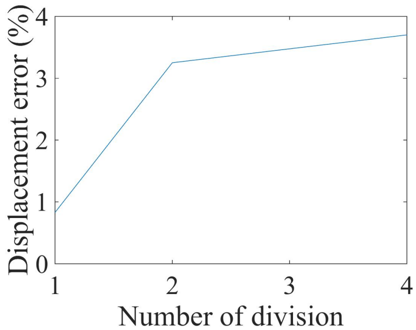
Figure 129: 27NodeBrick clamped beams with element side length 0.25m

The ESSI results were listed in Table (35). The theoretical solution is 1.60E-5 m.

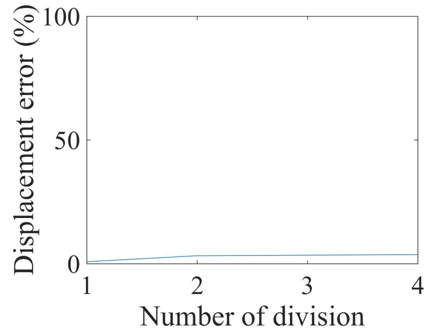
Table 35: Results for 27NodeBrick clamped beams with more elements

Element Type	Element side length		
	1 m	0.5 m	0.25 m
27NodeBrick	1.64E-05 m	1.70E-05 m	1.71E-05 m
Error	0.83%	3.25%	3.70%

The errors were plotted in Figure (130).



(a) Error scale 0% - 4%



(b) Error scale 0% - 100%

Figure 130: 27NodeBrick clamped beam for different element number
Displacement error versus Number of division

The ESSI model fei files for the table above are here

19.4 Verification of 27NodeBrick stress in cantilever beams

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (131).

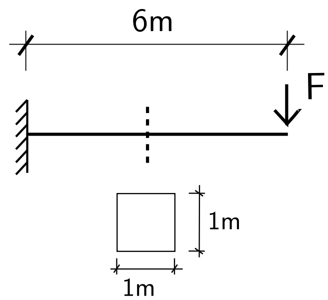


Figure 131: Problem description for cantilever beams of stress verification

The theoretical solution for the stress was calculated below.

The 27NodeBrick elements were shown in Figure (132).

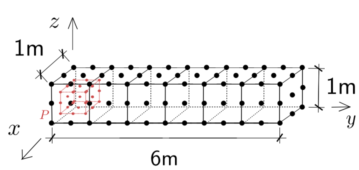


Figure 132: 27NodeBrick for cantilever beams of stress verification

The bending moment at the Gassian Point is

$$M = F(L - P_y) = 100N \times (6 - 0.1127)m = 588.73N \cdot m \tag{27}$$

The bending modulus is

$$I = \frac{bh^3}{12} = \frac{1}{12}m^4 \tag{28}$$

Therefore, the theoretical stress is

$$\sigma = \frac{M \cdot z}{I} = \frac{588.73N \cdot m \times (0.5 - 0.1127)m}{\frac{1}{12}m^4} = 2736Pa \tag{29}$$

To get a better result, the same geometry beam was also cut into small elements. When more elements were used, the theoretical stress was calculated again with the new coordinates. The calculation process is similar to the process above.

The numerical models were shown in Figure (133), (134) and (135).

Number of division 1:

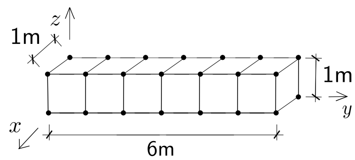


Figure 133: 27NodeBrick stress with element side length 1.0m

Number of division 2:

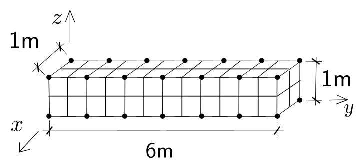


Figure 134: 27NodeBrick stress with element side length 0.5m

Number of division 4:

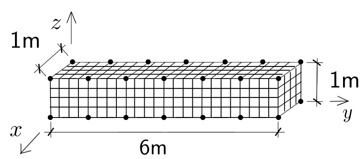
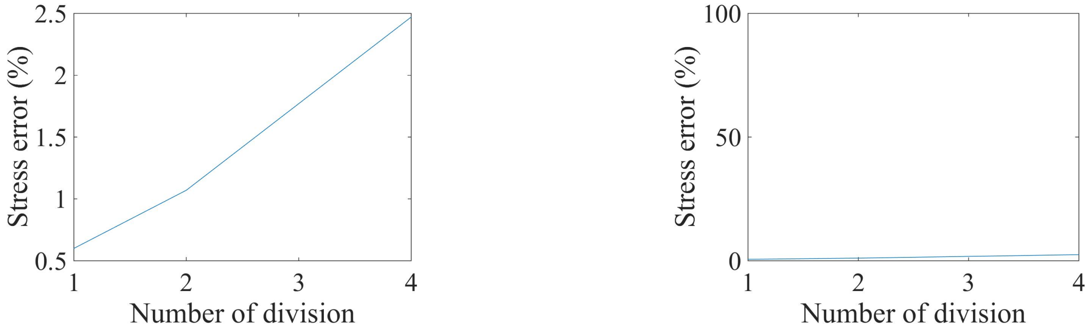


Figure 135: 27NodeBrick stress with element side length 0.25m

All the stress results were listed in Table (36).

Table 36: Results for 27NodeBrick stress with more elements

Element Type	Element side length		
	1 m	0.5 m	0.25 m
27NodeBrick	2719.81 Pa	3198.19 Pa	3464.76 Pa
Theoretical	2736.17 Pa	3164.27 Pa	3381.18 Pa
Error	0.60%	1.07%	2.47%



(a) Error scale 0% - 2.5%

(b) Error scale 0% - 100%

Figure 136: 27NodeBrick cantilever beams for stress verification
Stress error versus Number of division

The ESSI model fei files for the table above are here

19.5 Verification of 27NodeBrick square plate with four edges clamped

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{30}$$

The theoretical solution is

$$d = \alpha_c \frac{qa^4}{D} = 0.00406 \times \frac{100N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \ N \cdot m} = 2.2015 \times 10^{-3} m \tag{31}$$

where α_c is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[8] α_c is 0.00406.

The 27NodeBrick were shown in Figure (137) - (142).

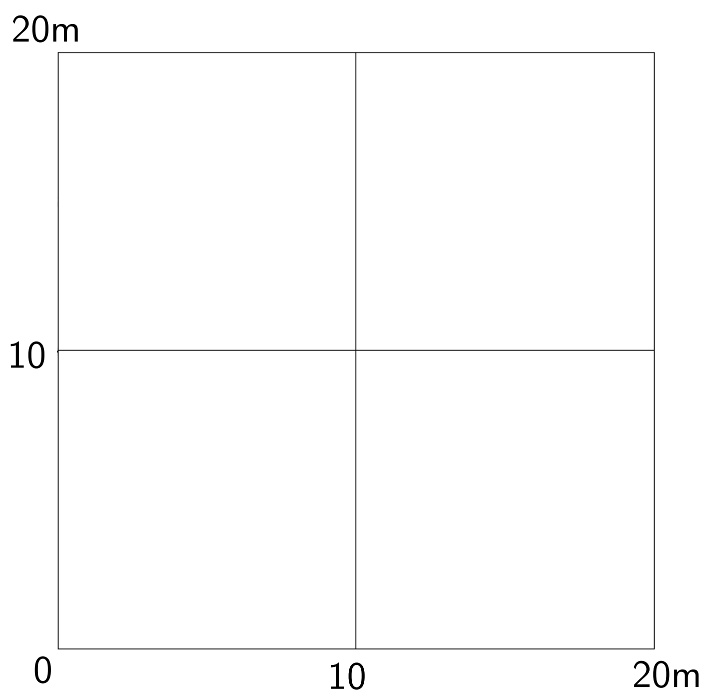


Figure 137: 27NodeBrick edge clamped square plate with element side length 10m

[8]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page120, 1959.

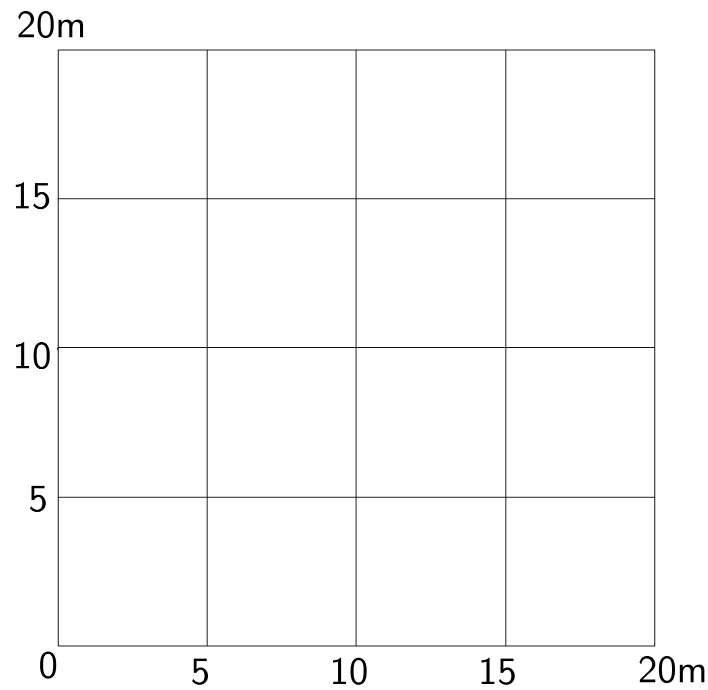


Figure 138: 27NodeBrick edge clamped square plate with element side length 5m

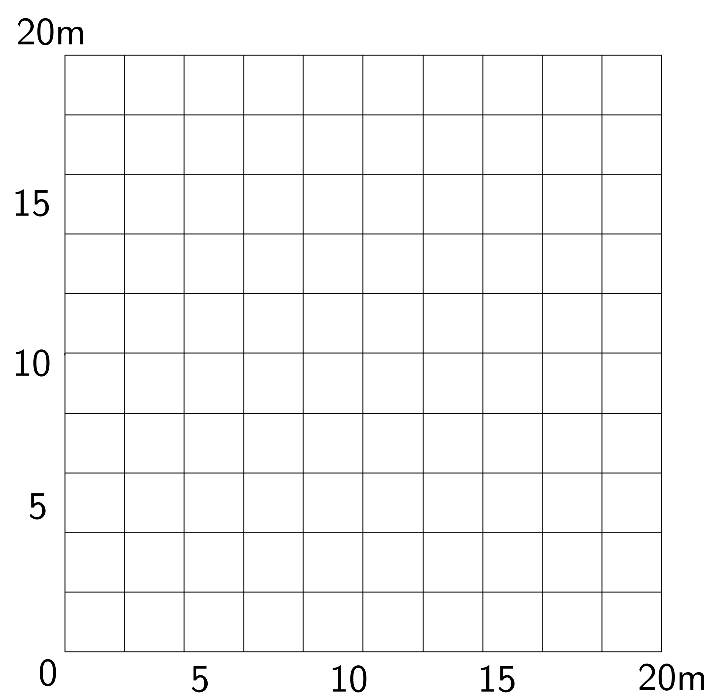


Figure 139: 27NodeBrick edge clamped square plate with element side length 2m

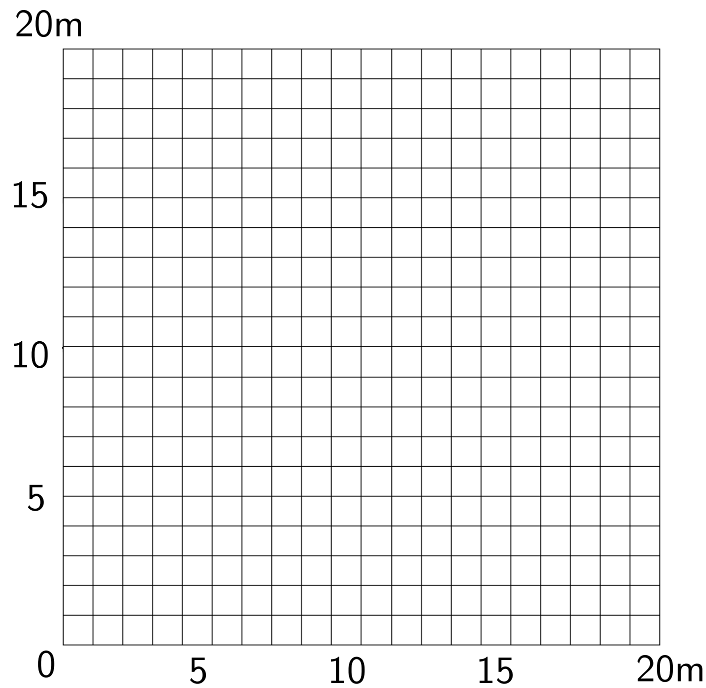


Figure 140: 27NodeBrick edge clamped square plate with element side length 1m

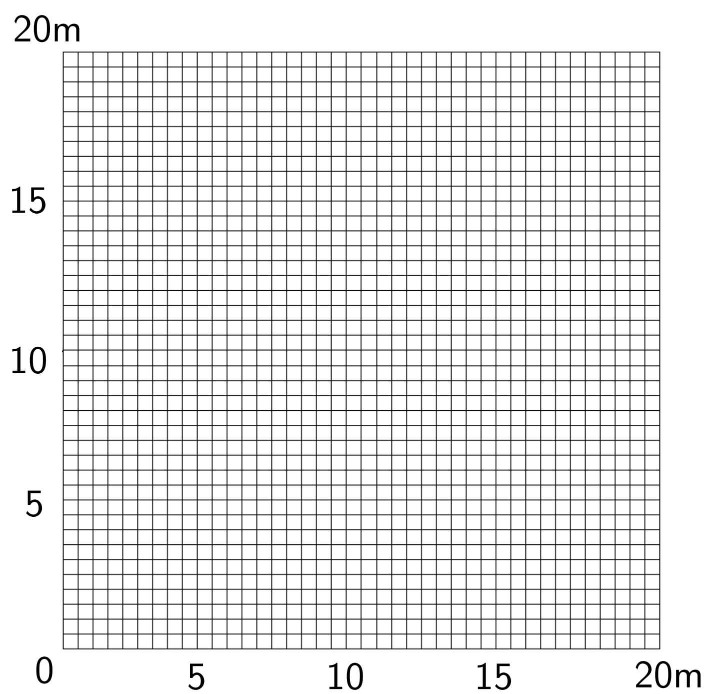


Figure 141: 27NodeBrick edge clamped square plate with element side length 0.5m

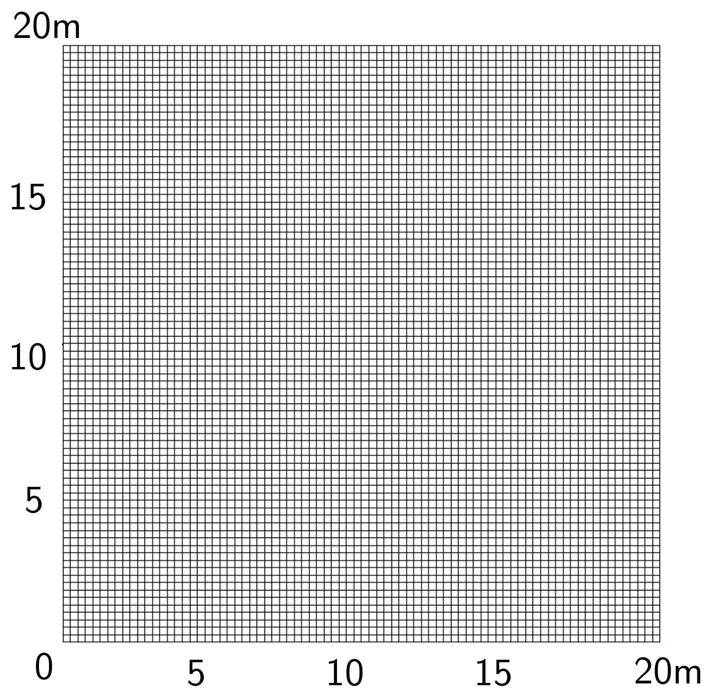


Figure 142: 27NodeBrick edge clamped square plate with element side length 0.25m

The results were listed in Table (37).

Table 37: Results for 27NodeBrick square plate with four edges clamped

Element type	27NodeBrick	27NodeBrick	27NodeBrick	Theoretical displacement
Number of layers	1layer	2layers	4layers	
Element side length	Height:1.00m	Height:0.50m	Height:0.25m	
10m	4.82E-004 m	4.82E-004 m	4.82E-004 m	2.20E-03 m
5m	1.97E-003 m	1.98E-003 m	1.98E-003 m	2.20E-03 m
2m	2.25E-003 m	2.26E-003 m	2.26E-003 m	2.20E-03 m
1m	2.28E-003 m	2.29E-003 m	2.29E-003 m	2.20E-03 m
0.5m	2.29E-003 m	2.30E-003 m	2.30E-003 m	2.20E-03 m
0.25m	2.29E-003 m	2.30E-003 m	-[9]	2.20E-03 m

The errors were listed in Table (38).

[9] This model run out of memory on machine cml01 (memory: 23.5GB). This model has 233,289 nodes with 3 dofs, which may require 40GB memory.

Table 38: Errors for 27NodeBrick square plate with four edges clamped

Element type	27NodeBrick	27NodeBrick	27NodeBrick
Number of layers	1layer	2layers	4layers
Element side length	Height:1.00m	Height:0.50m	Height:0.25m
10m	78.11%	78.10%	78.10%
5m	10.67%	10.19%	10.16%
2m	2.23%	2.79%	2.83%
1m	3.56%	4.16%	4.22%
0.5m	3.96%	4.58%	4.65%
0.25m	4.08%	4.70%	-

The errors were plotted in Figure (143).

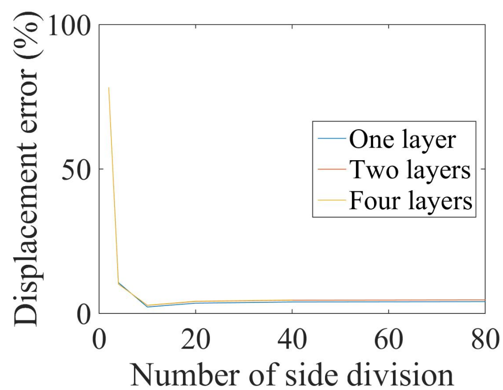


Figure 143: 27NodeBrick square plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

19.6 Verification of 27NodeBrick square plate with four edges simply supported

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{32}$$

The theoretical solution is

$$d = \alpha_s \frac{qa^4}{D} = 0.00126 \times \frac{100N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \ N \cdot m} = 7.0936 \times 10^{-3} m \tag{33}$$

where α_s is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[10] α_s is 0.00126.

The 27NodeBrick were shown in Figure (144) - (149).

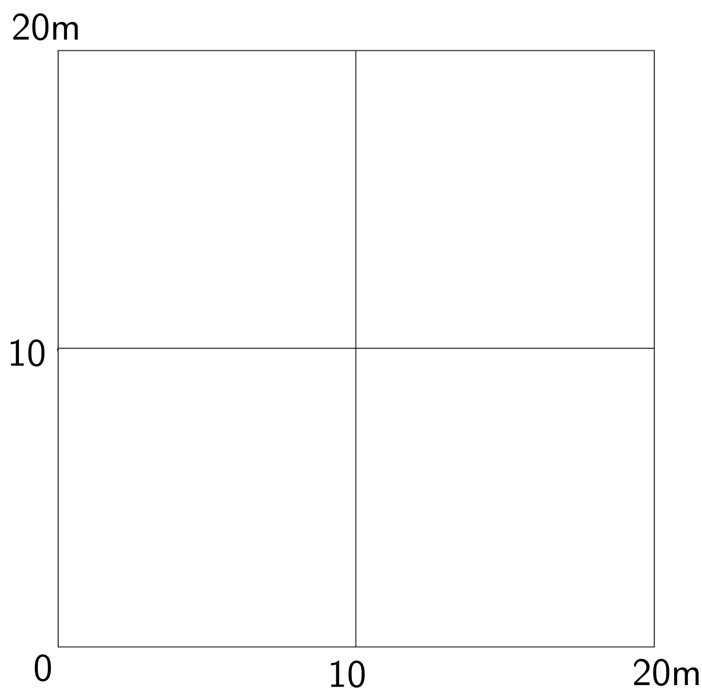


Figure 144: 27NodeBrick edge simply supported square plate with element side length 10m

[10] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page202, 1959.

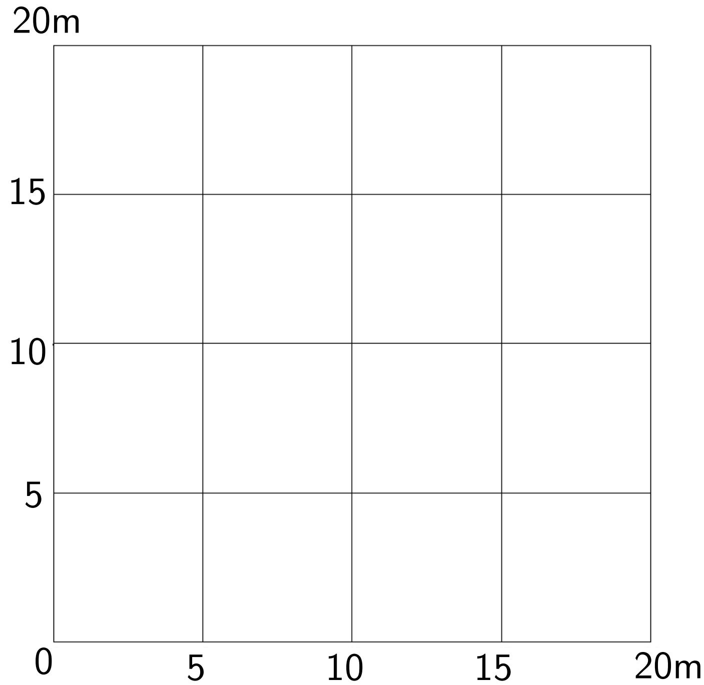


Figure 145: 27NodeBrick edge simply supported square plate with element side length 5m

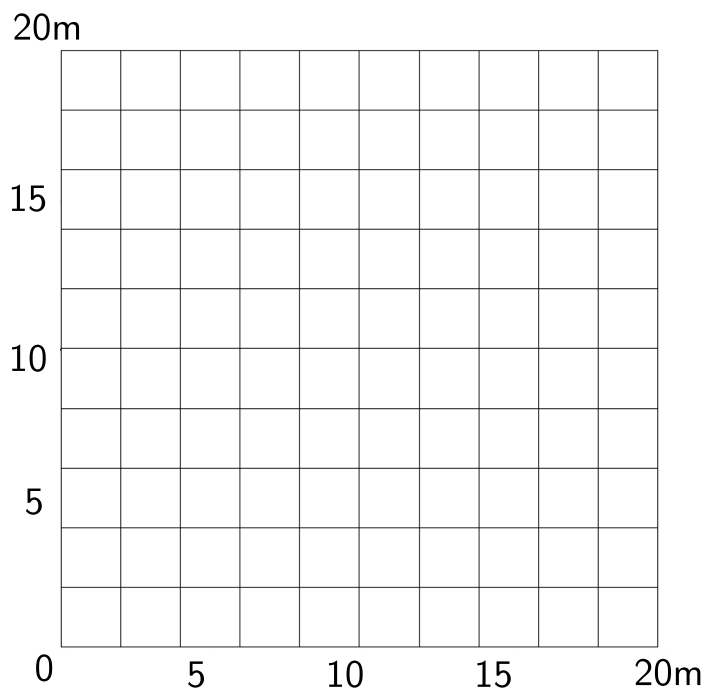


Figure 146: 27NodeBrick edge simply supported square plate with element side length 2m

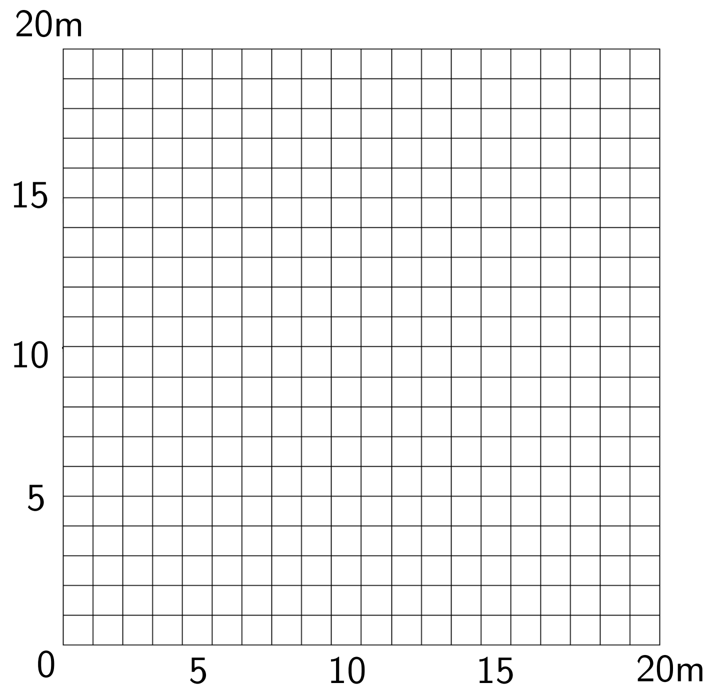


Figure 147: 27NodeBrick edge simply supported square plate with element side length 1m

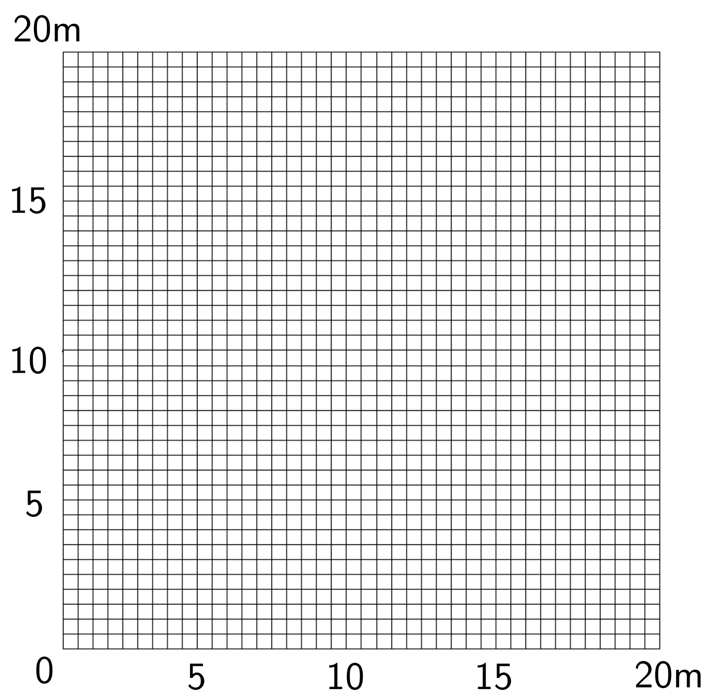


Figure 148: 27NodeBrick edge simply supported square plate with element side length 0.5m

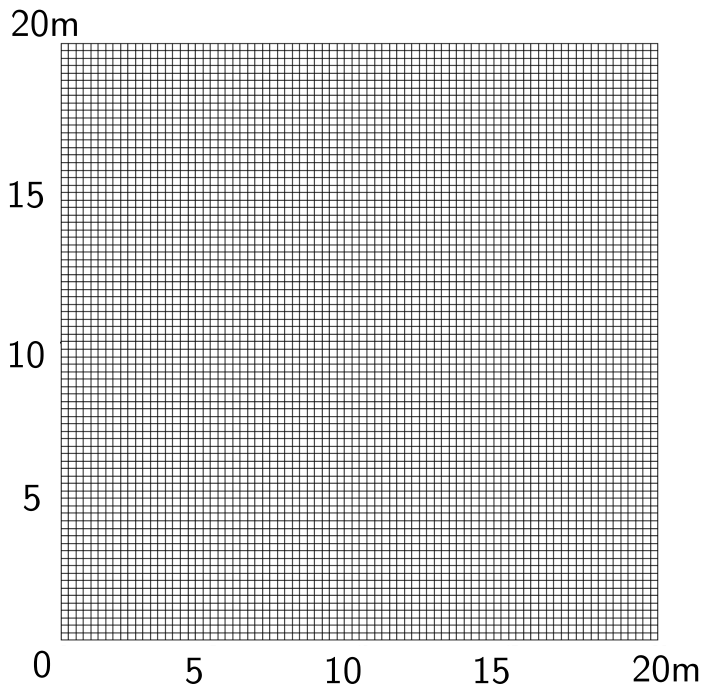


Figure 149: 27NodeBrick edge simply supported square plate with element side length 0.25m

The results were listed in Table (39).

Table 39: Results for 27NodeBrick square plate with four edges simply supported

Element type	27NodeBrick	27NodeBrick	Theoretical displacement
Number of layers	2layers	4layers	
Element side length	Height:0.50m	Height:0.25m	
10m	6.54E-003 m	6.54E-003 m	7.09E-03 m
5m	7.24E-003 m	7.24E-003 m	7.09E-03 m
2m	7.44E-003 m	7.44E-003 m	7.09E-03 m
1m	7.49E-003 m	7.49E-003 m	7.09E-03 m
0.5m	7.50E-003 m	7.50E-003 m	7.09E-03 m
0.25m	7.51E-003 m	-[11]	7.09E-03 m

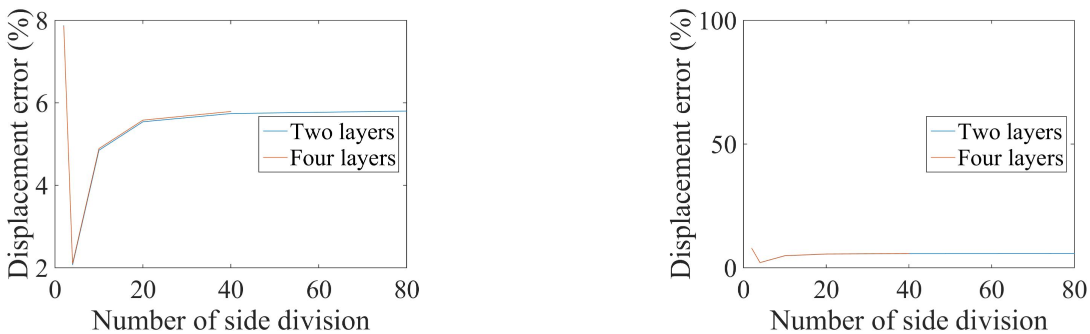
The errors were listed in Table (40).

[11] This model run out of memory on machine cml01 (memory: 23.5GB). This model has 233,289 nodes with 3 dofs, which may require 40GB memory.

Table 40: Errors for 27NodeBrick square plate with four edges simply supported

Element type	27NodeBrick	27NodeBrick
Number of layers	2layers	4layers
Element side length	Height:0.50m	Height:0.25m
10m	7.87%	7.85%
5m	2.07%	2.10%
2m	4.85%	4.89%
1m	5.54%	5.58%
0.5m	5.74%	5.79%
0.25m	5.80%	-

The errors were plotted in Figure (150).



(a) Error scale 0% - 8%

(b) Error scale 0% - 100%

Figure 150: 27NodeBrick square plate with edge simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

19.7 Verification of 27NodeBrick circular plate with all edges clamped

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{34}$$

The theoretical solution[12] is

$$d = \frac{qa^4}{64D} = \frac{100 N/m^2 \times 10^4 m^4}{64 \times 9.1575 \times 10^6 \ N \cdot m} = 1.7106 \times 10^{-3} m \tag{35}$$

The 27NodeBrick were shown in Figure (151) - (156).

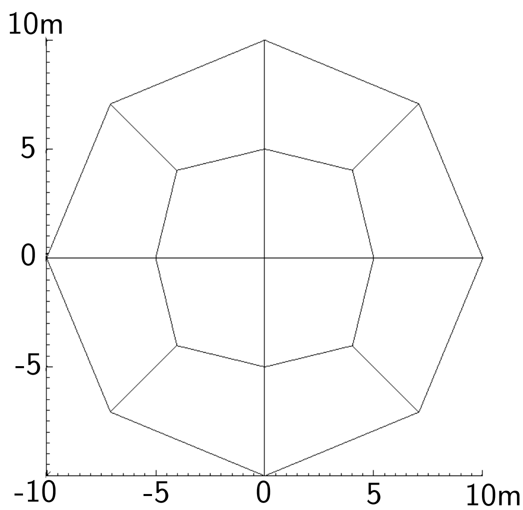


Figure 151: 27NodeBrick edge clamped circular plate with element side length 10m

[12]Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

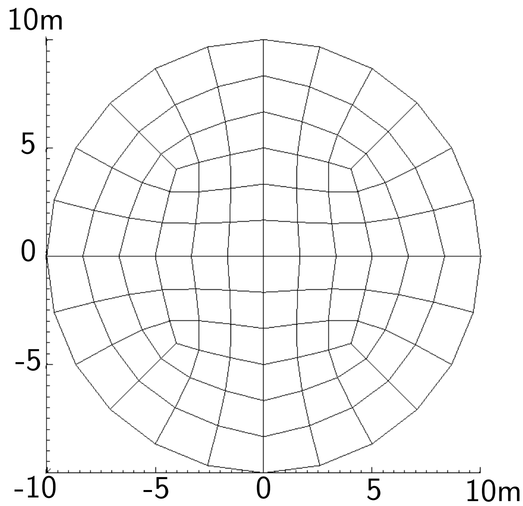


Figure 152: 27NodeBrick edge clamped circular plate with element side length 5m

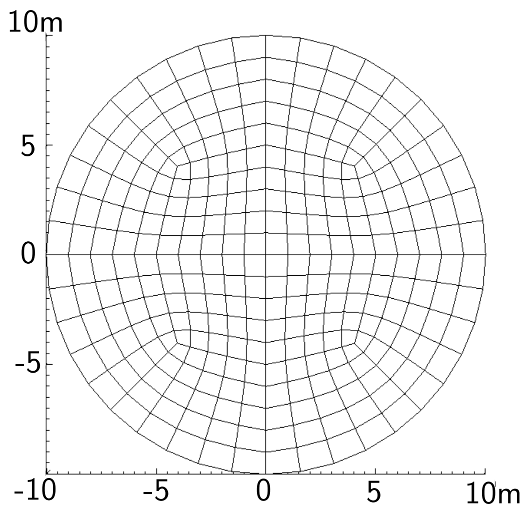


Figure 153: 27NodeBrick edge clamped circular plate with element side length 2m

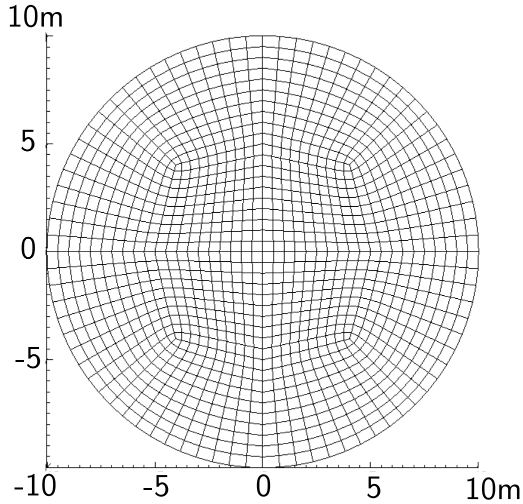


Figure 154: 27NodeBrick edge clamped circular plate with element side length 1m

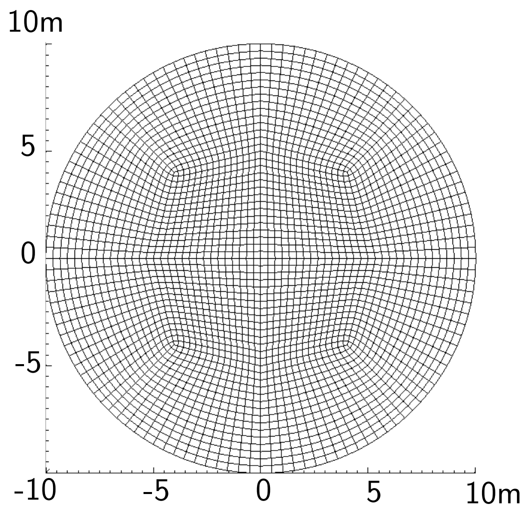


Figure 155: 27NodeBrick edge clamped circular plate with element side length 0.5m

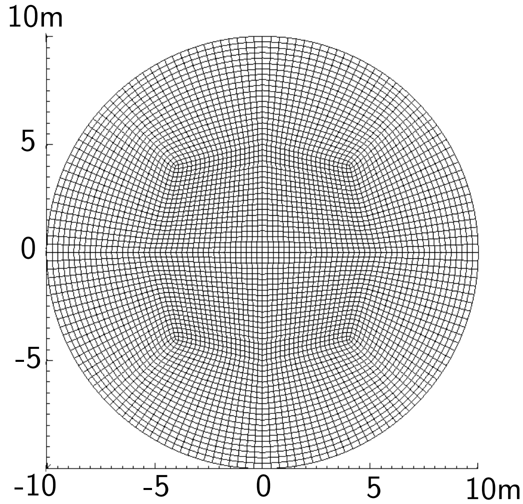


Figure 156: 27NodeBrick edge clamped circular plate with element side length 0.25m

The results were listed in Table (41).

Table 41: Results for 27NodeBrick circular plate with four edges clamped

Element type	27NodeBrick	27NodeBrick	27NodeBrick	Theoretical displacement
Number of layers	1layer	2layers	4layers	
Number of diameter divisions	Height:1.00m	Height:0.50m	Height:0.25m	
4	2.777E-03 m	2.788E-03 m	2.789E-03 m	1.706E-03 m
12	2.772E-03 m	2.786E-03 m	2.787E-03 m	1.706E-03 m
20	2.545E-03 m	2.556E-03 m	2.558E-03 m	1.706E-03 m
40	1.758E-03 m	1.768E-03 m	1.769E-03 m	1.706E-03 m
60	1.762E-03 m	1.772E-03 m	1.773E-03 m	1.706E-03 m
80	1.763E-03 m	1.773E-03 m	1.774E-03 m	1.706E-03 m

The errors were listed in Table (42).

Table 42: Errors for 27NodeBrick circular plate with four edges clamped

Element type	27NodeBrick	27NodeBrick	27NodeBrick
Number of layers	1layer	2layers	4layers
Number of diameter divisions	Height:1.00m	Height:0.50m	Height:0.25m
4	62.75%	63.42%	63.47%
12	62.46%	63.27%	63.34%
20	49.14%	49.82%	49.91%
40	3.03%	3.62%	3.68%
60	3.25%	3.83%	3.91%
80	3.32%	3.91%	3.99%

The errors were shown in Figure (157).

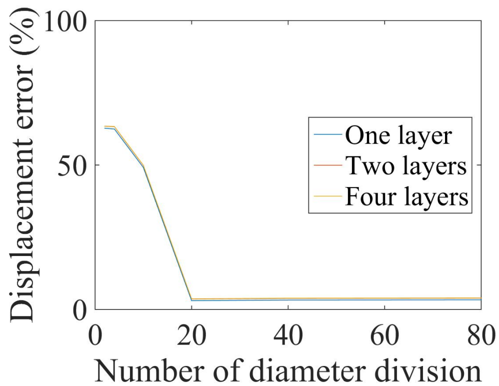


Figure 157: 27NodeBrick circular plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

19.8 Verification of 27NodeBrick circular plate with all edges simply supported

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{36}$$

The theoretical solution[13] is

$$d = \frac{(5+\nu)qa^4}{64(1+\nu)D} = \frac{(5+0.3) \times 100 N/m^2 \times 10^4 m^4}{64 \times (1+0.3) \times 9.1575 \times 10^6 \ N \cdot m} = 6.956 \times 10^{-3} m \tag{37}$$

The 27NodeBrick were shown in Figure (158) - (163).

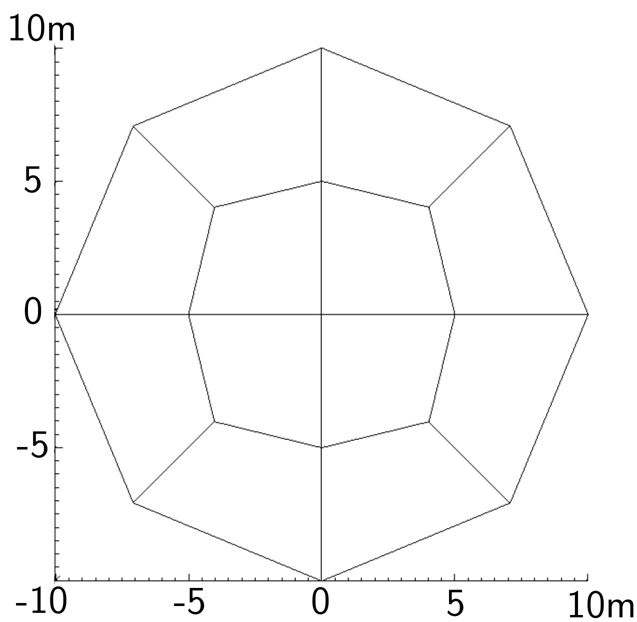


Figure 158: 27NodeBrick edge simply supported circular plate with element side length 10m

[13] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

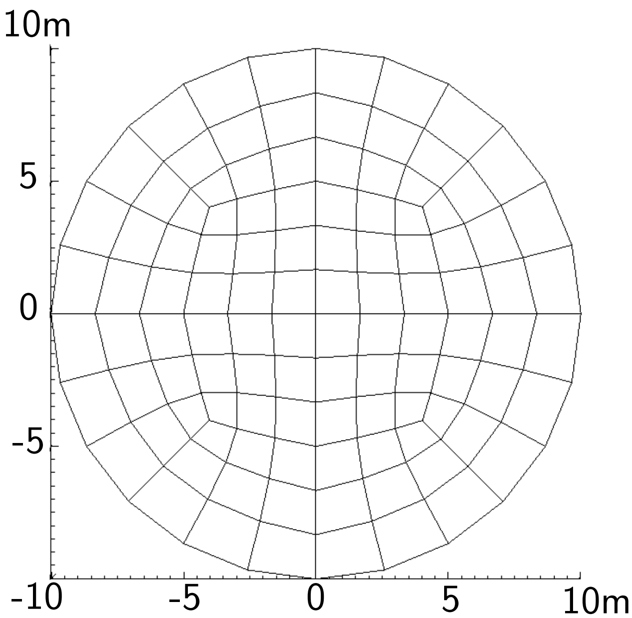


Figure 159: 27NodeBrick edge simply supported circular plate with element side length 5m

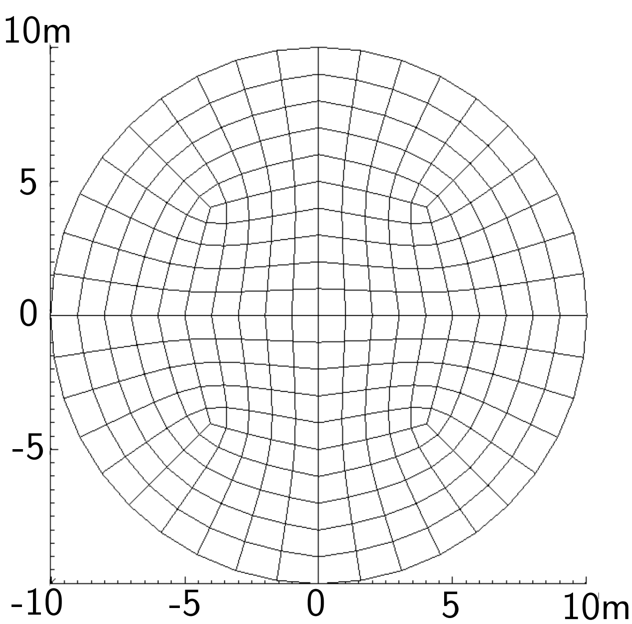


Figure 160: 27NodeBrick edge simply supported circular plate with element side length 2m

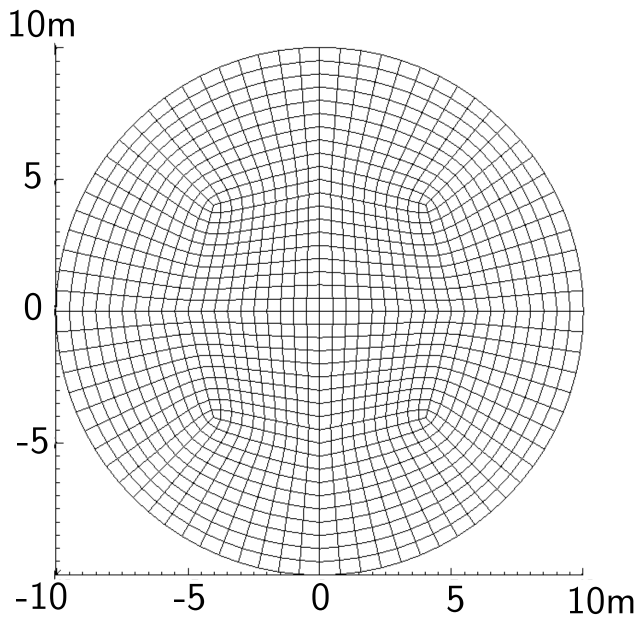


Figure 161: 27NodeBrick edge simply supported circular plate with element side length 1m

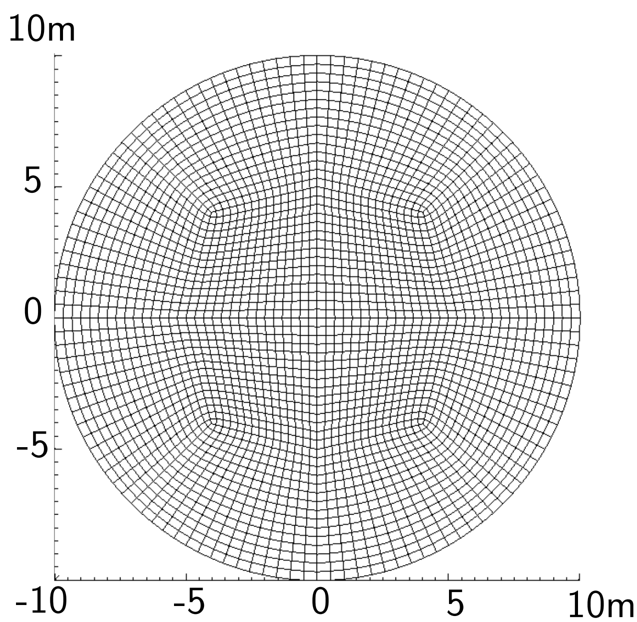


Figure 162: 27NodeBrick edge simply supported circular plate with element side length 0.5m

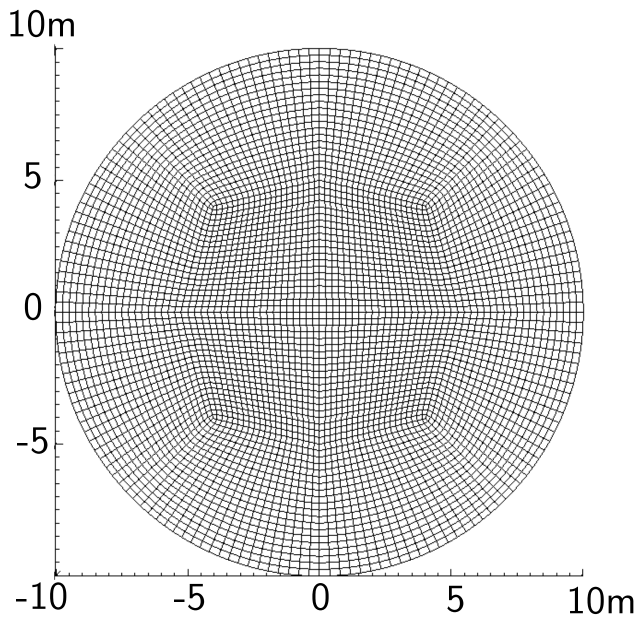


Figure 163: 27NodeBrick edge simply supported circular plate with element side length 0.25m

The results were listed in Table (43).

Table 43: Results for 27NodeBrick cicular plate with four edges simply supported

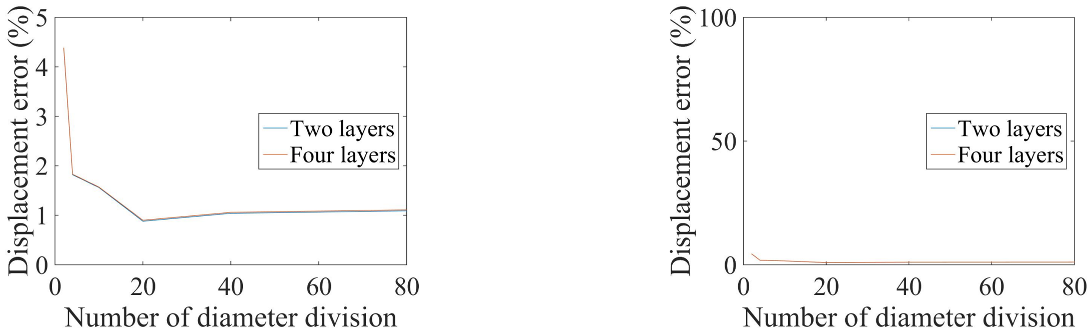
Element type	27NodeBrick	27NodeBrick	Theoretical displacement
Number of layers	2layers	4layers	
Number of diameter divisions	Height:0.50m	Height:0.25m	
4	7.259E-03 m	7.261E-03 m	6.956E-03 m
12	7.083E-03 m	7.084E-03 m	6.956E-03 m
20	7.064E-03 m	7.065E-03 m	6.956E-03 m
40	7.018E-03 m	7.019E-03 m	6.956E-03 m
60	7.029E-03 m	7.030E-03 m	6.956E-03 m
80	7.032E-03 m	7.034E-03 m	6.956E-03 m

The errors were listed in Table (44).

Table 44: Errors for 27NodeBrick cicular plate with four edges simply supported

Element type	27NodeBrick	27NodeBrick
Number of layers	2layers	4layers
Number of diameter divisions	Height:0.50m	Height:0.25m
4	4.36%	4.38%
12	1.82%	1.83%
20	1.56%	1.57%
40	0.88%	0.90%
60	1.04%	1.06%
80	1.09%	1.11%

The errors were plotted in Figure (164).



(a) Error scale 0% - 5%

(b) Error scale 0% - 100%

Figure 164: 27NodeBrick circular plate with edge simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

Verification for 4NodeANDES

20 Verification of 4NodeANDES elements

20.1 Verification of 4NodeANDES cantilever beams

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (165).

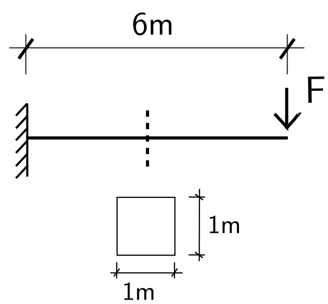


Figure 165: Problem description for cantilever beams

Theoretical displacement (bending and shear deformation):

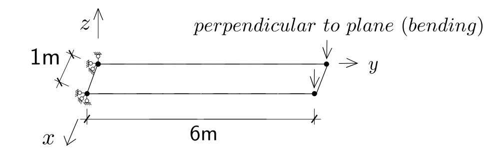
$$
\begin{aligned}
d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\
&= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\
&= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\
&= 8.784 \times 10^{-4}\ m
\end{aligned}
\tag{38}
$$

4NodeANDES element model:

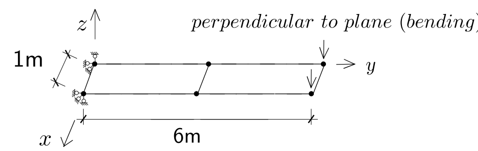
- ***Force direction: perpendicular to plane (bending)***

When the force direction is perpendicular to the plane, only the bending deformation is calculated in 4NodeANDES elements.

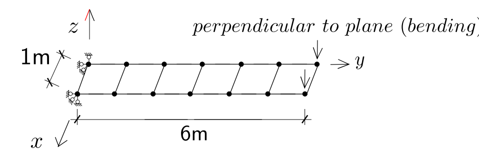
The 4NodeANDES elements were shown in Figure (166).



(a) One 4NodeANDES element



(b) Two 4NodeANDES elements



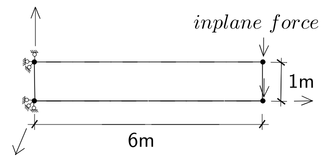
(c) Six 4NodeANDES elements

Figure 166: 4NodeANDES elements for cantilever beams under force perpendicular to plane

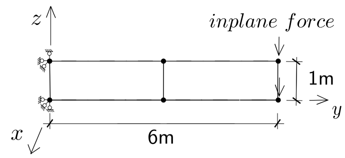
- ***Force direction: inplane force***

When the force direction is inplane, both the bending and shear deformation are calculated in 4NodeANDES elements.

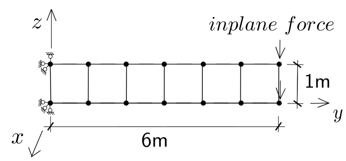
The 4NodeANDES elements under inplane force were shown in Figure (167).



(a) One 4NodeANDES element



(b) Two 4NodeANDES elements



(c) Six 4NodeANDES elements

Figure 167: 4NodeANDES elements for cantilever beams under inplane force

The ESSI results for the force ***perpendicular to plane (bending)*** were listed in Table (45). The theoretical solution is 8.784E-04 m.

Table 45: Results for 4NodeANDES cantilever beams under the force perpendicular to plane (bending)

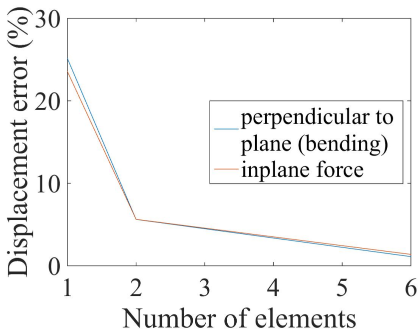
Element number	1	2	6
4NodeANDES	6.56E-04 m	8.27E-04 m	8.86E-04 m
Error	25.34%	5.87%	0.83%

The ESSI results for the ***inplane force*** were listed in Table (46).
The theoretical solution is 8.784E-04 m.

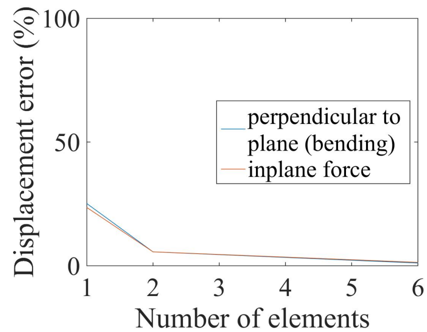
Table 46: Results for 4NodeANDES cantilever beams under the inplane force

Element number	1	2	6
4NodeANDES	6.70E-04 m	8.27E-04 m	8.64E-04 m
Error	23.77%	5.89%	1.65%

The errors were plotted in Figure (168).



(a) Error scale 0% - 30%



(b) Error scale 0% - 100%

Figure 168: 4NodeANDES cantilever beam for different element number
Displacement error versus Number of elements

The ESSI model fei files for the table above are here

20.2 Verification of 4NodeANDES cantilever beam for different Poisson's ratio

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0 - 0.49$. The force direction was shown in Figure (169).

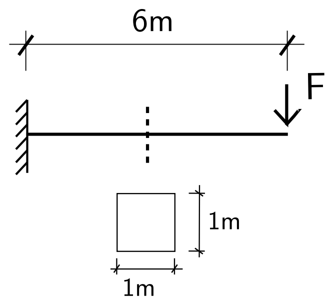


Figure 169: Problem description for cantilever beams of different Poisson's ratios

The theoretical solution for $\nu = 0.0$ was calculated below, while the solution for other Poisson's ratio were calculated by the similar process.

Theoretical displacement (bending and shear deformation):

$$\begin{aligned}
d &= \frac{FL^3}{3EI} + \frac{FL}{GA_v} \\
&= \frac{FL^3}{3E\frac{bh^3}{12}} + \frac{FL}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{100N \times 6^3 m^3}{3 \times 10^8 N/m^2 \times \frac{1}{12} m^4} + \frac{100N \times 6m}{\frac{10}{2} \times 10^7 N/m^2 \times 1m^2 \times \frac{5}{6}} \\
&= 8.64 \times 10^{-4} m + 0.144 \times 10^{-4} m \\
&= 8.784 \times 10^{-4}\ m
\end{aligned} \tag{39}$$

The rotation angle at the end:

$$\theta = \frac{FL^2}{2EI} = \frac{100N \times 6^2 m^2}{2 \times 10^8 N/m^2 \times \frac{1}{12} m^4} = 2.16 \times 10^{-4}\ rad = 0.0124° \tag{40}$$

The 4NodeANDES elements for cantilever beams of different Poisson's ratios were shown in Figure (170) and (171):

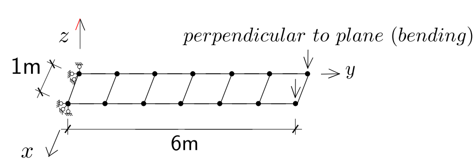


Figure 170: 4NodeANDES elements for different Poisson's ratios under the force perpendicular to plane (bending)

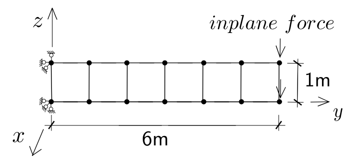


Figure 171: 4NodeANDES elements for different Poisson's ratios under the inplane force

The ESSI results for the force ***perpendicular to plane (bending)*** were listed in Table (47) - (49).

Table 47: ***Displacement error*** results for 4NodeANDES with element side length 1 m under the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.639E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	1.38%
0.05	8.635E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	1.49%
0.10	8.622E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	1.71%
0.15	8.599E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	2.04%
0.20	8.566E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	2.48%
0.25	8.522E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	3.05%
0.30	8.466E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	3.75%
0.35	8.398E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	4.59%
0.40	8.315E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	5.60%
0.45	8.216E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	6.78%
0.49	8.124E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	7.88%

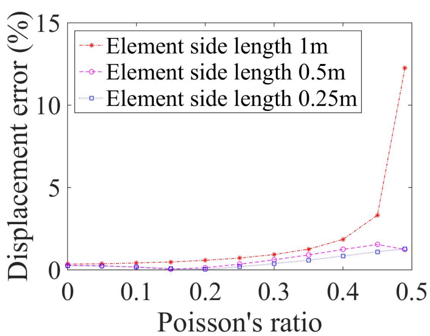
Table 48: ***Displacement error*** results for 4NodeANDES with element side length 0.5 m under the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.724E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.68%
0.05	8.724E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.76%
0.10	8.717E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.93%
0.15	8.703E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	1.17%
0.20	8.682E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	1.49%
0.25	8.652E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	1.91%
0.30	8.615E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	2.42%
0.35	8.569E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	3.02%
0.40	8.514E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	3.73%
0.45	8.449E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	4.54%
0.49	8.388E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	5.30%

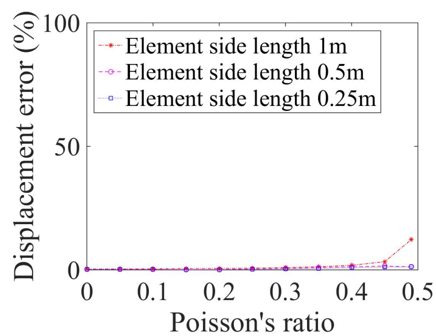
Table 49: ***Displacement error*** results for 4NodeANDES with element side length 0.25 m under the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.640E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	1.64%
0.05	8.637E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	1.75%
0.10	8.627E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	1.95%
0.15	8.611E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	2.21%
0.20	8.588E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	2.56%
0.25	8.559E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	2.97%
0.30	8.523E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	3.46%
0.35	8.480E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	4.03%
0.40	8.429E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	4.69%
0.45	8.370E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	5.44%
0.49	8.316E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	6.11%

The errors were plotted in Figure (172).



(a) Error scale 0% - 15%



(b) Error scale 0% - 100%

Figure 172: 4NodeANDES cantilever beam for force perpendicular to the plane(bending)
Displacement error versus Poisson's ratio

The ESSI results for the ***inplane force*** were listed in Table (50) - (52).

Table 50: ***Displacement error*** results for 4NodeANDES with element side length 1 m under the inplane force

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.790E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.07%
0.05	8.799E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.09%
0.10	8.809E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.12%
0.15	8.821E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.17%
0.20	8.835E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	0.25%
0.25	8.853E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	0.37%
0.30	8.878E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	0.56%
0.35	8.913E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	0.87%
0.40	8.971E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	1.44%
0.45	9.107E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	2.89%
0.49	9.901E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	11.79%

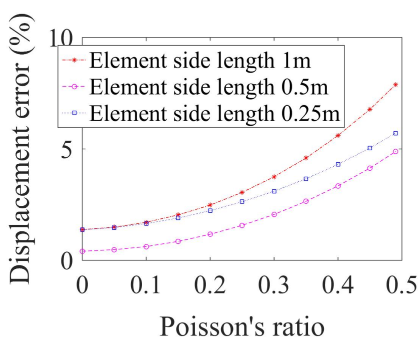
Table 51: ***Displacement error*** results for 4NodeANDES with element side length 0.5 m under the inplane force

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.784E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.00%
0.05	8.788E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.04%
0.10	8.787E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.13%
0.15	8.782E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.27%
0.20	8.772E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	0.47%
0.25	8.759E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	0.70%
0.30	8.742E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	0.98%
0.35	8.722E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	1.29%
0.40	8.699E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	1.63%
0.45	8.679E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	1.94%
0.49	8.709E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	1.67%

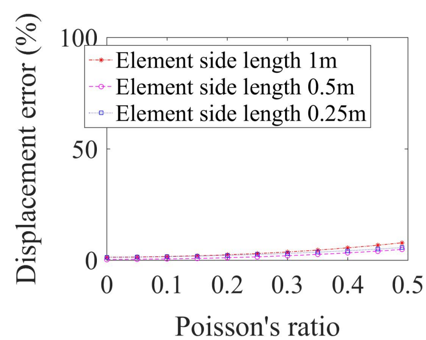
Table 52: ***Displacement error*** results for 4NodeANDES with element side length 0.25 m under the inplane force

Poisson's ratio	4NodeANDES displacement	Theory displacement (bending)	Theory displacement (shear)	Theory displacement(all)	Error
0.00	8.782E-04 m	8.640E-04 m	1.440E-05 m	8.784E-04 m	0.02%
0.05	8.786E-04 m	8.640E-04 m	1.512E-05 m	8.791E-04 m	0.06%
0.10	8.788E-04 m	8.640E-04 m	1.586E-05 m	8.799E-04 m	0.12%
0.15	8.786E-04 m	8.640E-04 m	1.659E-05 m	8.806E-04 m	0.23%
0.20	8.781E-04 m	8.640E-04 m	1.734E-05 m	8.813E-04 m	0.37%
0.25	8.774E-04 m	8.640E-04 m	1.808E-05 m	8.821E-04 m	0.53%
0.30	8.763E-04 m	8.640E-04 m	1.884E-05 m	8.828E-04 m	0.74%
0.35	8.750E-04 m	8.640E-04 m	1.959E-05 m	8.836E-04 m	0.97%
0.40	8.734E-04 m	8.640E-04 m	2.035E-05 m	8.844E-04 m	1.24%
0.45	8.717E-04 m	8.640E-04 m	2.111E-05 m	8.851E-04 m	1.52%
0.49	8.706E-04 m	8.640E-04 m	2.173E-05 m	8.857E-04 m	1.71%

The errors were plotted in Figure (172).



(a) Error scale 0% - 10%



(b) Error scale 0% - 100%

Figure 173: 4NodeANDES cantilever beam for inplane force ***Displacement error*** versus Poisson's ratio

The angle results for the force ***perpendicular to plane (bending)*** were listed in Table (53).

Table 53: ***Rotation angle*** results for element side length 1 m under the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.238E-02	1.240E-02	0.19%
0.05	1.237E-02	1.240E-02	0.23%
0.10	1.236E-02	1.240E-02	0.34%
0.15	1.234E-02	1.240E-02	0.52%
0.20	1.230E-02	1.240E-02	0.78%
0.25	1.226E-02	1.240E-02	1.12%
0.30	1.221E-02	1.240E-02	1.54%
0.35	1.214E-02	1.240E-02	2.07%
0.40	1.206E-02	1.240E-02	2.70%
0.45	1.197E-02	1.240E-02	3.46%
0.49	1.188E-02	1.240E-02	4.16%

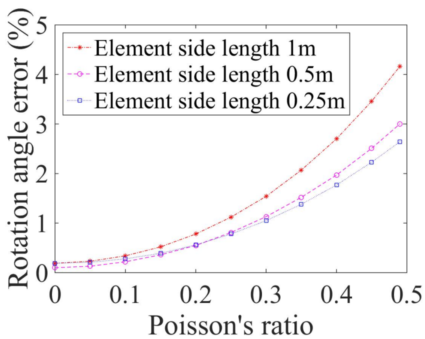
Table 54: ***Rotation angle*** results for element side length 0.5 m the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.239E-02	1.240E-02	0.10%
0.05	1.238E-02	1.240E-02	0.13%
0.10	1.237E-02	1.240E-02	0.22%
0.15	1.236E-02	1.240E-02	0.36%
0.20	1.233E-02	1.240E-02	0.55%
0.25	1.230E-02	1.240E-02	0.81%
0.30	1.226E-02	1.240E-02	1.13%
0.35	1.221E-02	1.240E-02	1.52%
0.40	1.216E-02	1.240E-02	1.97%
0.45	1.209E-02	1.240E-02	2.51%
0.49	1.203E-02	1.240E-02	3.00%

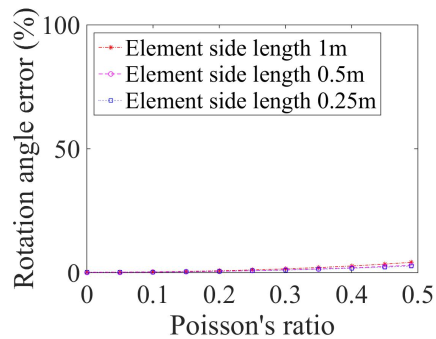
Table 55: ***Rotation angle*** results for element side length 0.25 m under the force perpendicular to plane (bending)

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.238E-02	1.240E-02	0.19%
0.05	1.237E-02	1.240E-02	0.21%
0.10	1.237E-02	1.240E-02	0.28%
0.15	1.235E-02	1.240E-02	0.39%
0.20	1.233E-02	1.240E-02	0.56%
0.25	1.230E-02	1.240E-02	0.78%
0.30	1.227E-02	1.240E-02	1.05%
0.35	1.223E-02	1.240E-02	1.38%
0.40	1.218E-02	1.240E-02	1.77%
0.45	1.212E-02	1.240E-02	2.23%
0.49	1.207E-02	1.240E-02	2.64%

The errors were plotted in Figure (174).



(a) Error scale 0% - 5%



(b) Error scale 0% - 100%

Figure 174: 4NodeANDES cantilever beam for force perpendicular to the plane(bending)
Rotation angle error versus Poisson's ratio

The ESSI results for the ***inplane force*** were listed in Table (56 - (58).

Table 56: ***Rotation angle*** results for element side length 1 m under the inplane force

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.254E-02	1.240E-02	1.14%
0.05	1.255E-02	1.240E-02	1.19%
0.10	1.256E-02	1.240E-02	1.26%
0.15	1.257E-02	1.240E-02	1.35%
0.20	1.258E-02	1.240E-02	1.47%
0.25	1.260E-02	1.240E-02	1.64%
0.30	1.263E-02	1.240E-02	1.89%
0.35	1.269E-02	1.240E-02	2.30%
0.40	1.278E-02	1.240E-02	3.08%
0.45	1.305E-02	1.240E-02	5.28%
0.49	1.506E-02	1.240E-02	21.43%

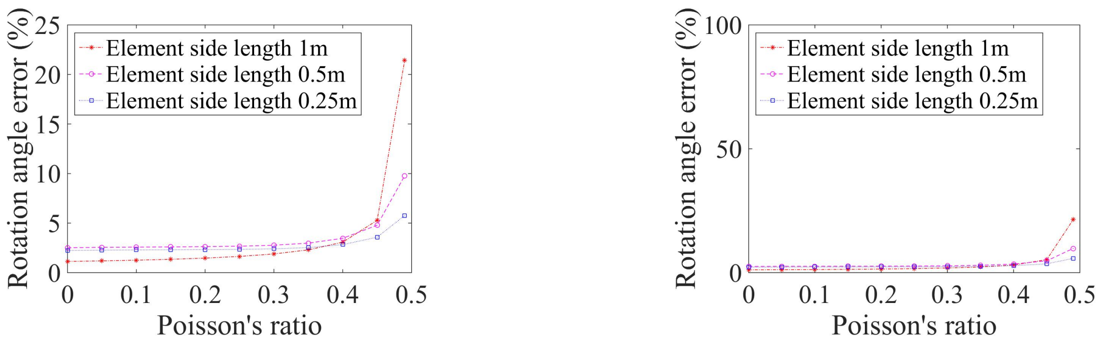
Table 57: ***Rotation angle*** results for element side length 0.5 m under the inplane force

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.271E-02	1.240E-02	2.51%
0.05	1.272E-02	1.240E-02	2.56%
0.10	1.272E-02	1.240E-02	2.58%
0.15	1.272E-02	1.240E-02	2.60%
0.20	1.273E-02	1.240E-02	2.63%
0.25	1.273E-02	1.240E-02	2.67%
0.30	1.274E-02	1.240E-02	2.77%
0.35	1.277E-02	1.240E-02	2.98%
0.40	1.283E-02	1.240E-02	3.47%
0.45	1.299E-02	1.240E-02	4.79%
0.49	1.361E-02	1.240E-02	9.78%

Table 58: ***Rotation angle*** results for element side length 0.25 m under the inplane force

Poisson's ratio	4NodeANDES angle (unit:°)	Theory angle (unit:°)	Error
0.00	1.268E-02	1.240E-02	2.24%
0.05	1.268E-02	1.240E-02	2.27%
0.10	1.268E-02	1.240E-02	2.30%
0.15	1.269E-02	1.240E-02	2.31%
0.20	1.269E-02	1.240E-02	2.33%
0.25	1.269E-02	1.240E-02	2.35%
0.30	1.270E-02	1.240E-02	2.41%
0.35	1.271E-02	1.240E-02	2.53%
0.40	1.275E-02	1.240E-02	2.83%
0.45	1.284E-02	1.240E-02	3.58%
0.49	1.312E-02	1.240E-02	5.77%

The errors were plotted in Figure (174).



(a) Error scale 0% - 25%

(b) Error scale 0% - 100%

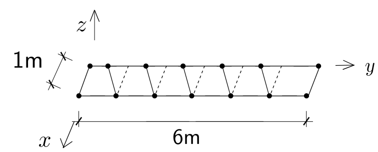
Figure 175: 4NodeANDES cantilever beam for inplane force
Rotation angle error versus Poisson's ratio

The ESSI model fei files for the table above are here

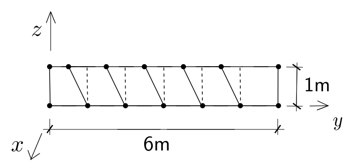
20.3 Test of irregular shaped 4NodeANDES cantilever beams

Cantilever model was used as an example. Three different shapes were tested.

In the ***first*** test, the upper two nodes of each element were moved one half element size along the $y - axis$, while the lower two nodes were kept at the same location. The element shape was shown in Figure (176).



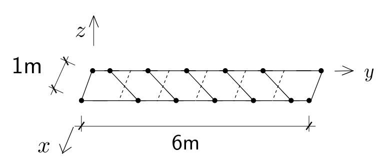
(a) Horizontal plane



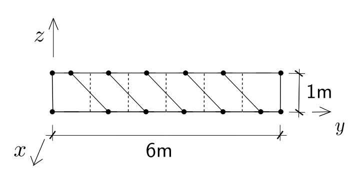
(b) Veritical plane

Figure 176: 4NodeANDES cantilever beam for irregular ***Shape 1***

In the ***second*** test, the upper nodes of each element were moved 50% element size along the $y - axis$, while the lower nodes were moved 50% element size in the other direction along the $y - axis$. The element shape was shown in Figure (177).



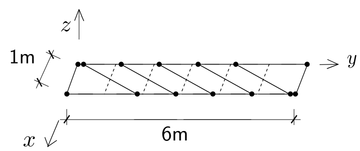
(a) Horizontal plane



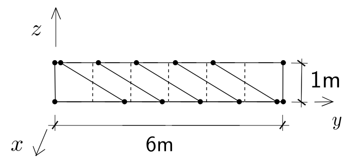
(b) Veritical plane

Figure 177: 4NodeANDES cantilever beam for irregular ***Shape 2***

In the ***third*** test, the upper two nodes of each element were moved 90% element size with different directions along the $y - axis$, while the lower nodes were moved 90% element size in the other direction along the $y - axis$. The element shape was shown in Figure (178).



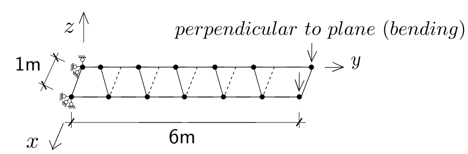
(a) Horizontal plane



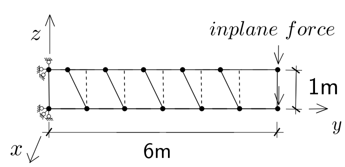
(b) Veritical plane

Figure 178: 4NodeANDES cantilever beam for irregular ***Shape 3***

The boundary conditions were shown in Figure (179), (180) and (181).

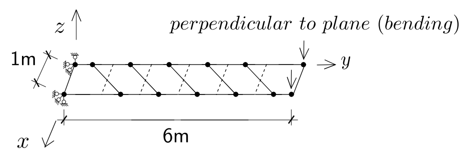


(a) Horizontal plane

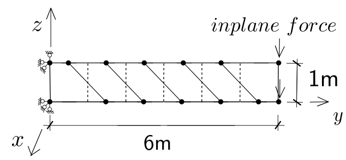


(b) Veritical plane

Figure 179: 4NodeANDES cantilever beam boundary conditions for irregular ***Shape 1***

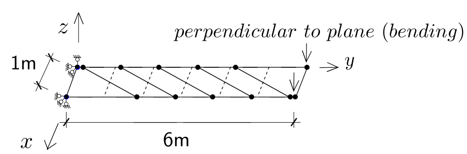


(a) Horizontal plane

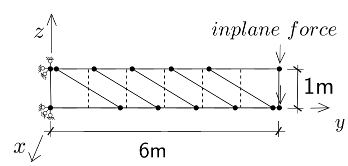


(b) Veritical plane

Figure 180: 4NodeANDES cantilever beam boundary conditions for irregular ***Shape 2***



(a) Horizontal plane



(b) Veritical plane

Figure 181: 4NodeANDES cantilever beam boundary conditions for irregular ***Shape 3***

The ESSI results were listed in Table (59).

Table 59: Results for 4NodeANDES cantilever beams of irregular shapes

Displacements for irregular shaped element					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
4NodeANDES	perpendicular to plane (bending)	8.639E-04 m	8.602E-04 m	8.534E-04 m	7.851E-04 m
4NodeANDES	inplane force	8.857E-04 m	7.036E-04 m	4.263E-04 m	1.909E-04 m
Theoretical	-	8.784E-04 m	8.784E-04 m	8.784E-04 m	8.784E-04 m

The errors were listed in Table (60) and (61).

Table 60: Errors for irregular shaped 4NodeANDES compared to theoretical solution

Errors for irregular shaped element, compared to theoretical solutions					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
4NodeANDES	perpendicular to plane (bending)	1.65%	2.07%	2.85%	10.63%
4NodeANDES	inplane force	0.83%	19.90%	51.47%	78.27%

Table 61: Errors for irregular shaped 4NodeANDES compared to normal shape

Errors for irregular shaped element, compared to normal shape					
Element Type	Force direction	Normal shape	Shape 1	Shape 2	Shape 3
4NodeANDES	perpendicular to plane (bending)	0.00%	0.42%	1.22%	9.12%
4NodeANDES	inplane force	0.00%	20.56%	51.87%	78.45%

The ESSI model fei files for the table above are here

Then, the beam was divided into small elements.

Problem description: Length=6m, Width=1m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.0$. Use the shear deformation coefficient $\kappa = 1.2$. The force direction was shown in Figure (182).

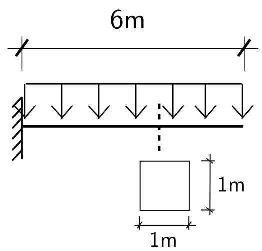


Figure 182: Problem description for cantilever beams under uniform pressure

Theoretical displacement (bending and shear deformation):

$$
\begin{aligned}
d &= \frac{qL^4}{8EI} + \frac{q\frac{L^2}{2}}{GA_v} \\
&= \frac{qL^4}{8E\frac{bh^3}{12}} + \frac{q\frac{L^2}{2}}{\frac{E}{2(1+\nu)}\frac{bh}{\kappa}} \\
&= \frac{400N/m \times 12^4 m^4}{8 \times 10^8 N/m^2 \times \frac{2^4}{12} m^4} + \frac{400N/m \times \frac{12^2}{2} m^2}{\frac{10^8}{2} N/m^2 \times 2m \times 2m \times \frac{5}{6}} \\
&= 7.776 \times 10^{-3} m + 1.728 \times 10^{-4} m \\
&= 7.9488 \times 10^{-3} m
\end{aligned}
\tag{41}
$$

The ESSI displacement results were listed in Table (62).

Table 62: Results for 4NodeANDES cantilever beams of irregular shapes with more elements

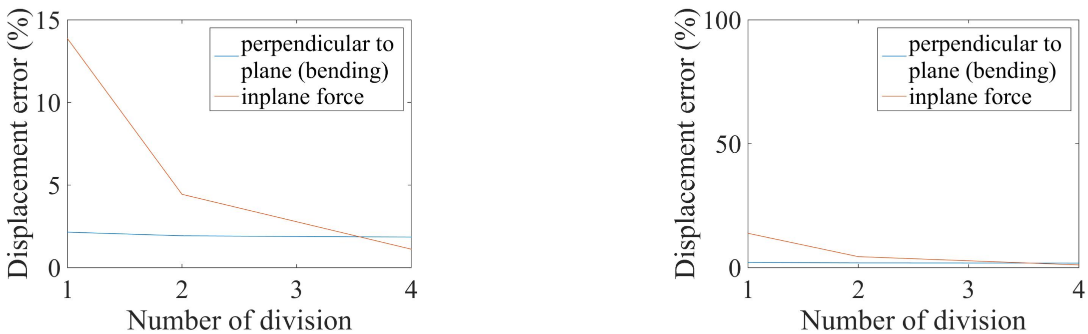
Element Type	Shape	Force direction	Number of division		
			1	2	4
4NodeANDES	shape1	perpendicular to plane (bending)	7.750E-03 m	7.768E-03 m	7.774E-03 m
4NodeANDES	shape1	inplane force	6.822E-03 m	7.569E-03 m	7.832E-03 m
4NodeANDES	shape2	perpendicular to plane (bending)	7.656E-03 m	7.734E-03 m	7.765E-03 m
4NodeANDES	shape2	inplane force	3.875E-03 m	5.855E-03 m	7.074E-03 m
4NodeANDES	shape3	perpendicular to plane (bending)	6.637E-03 m	7.139E-03 m	7.521E-03 m
4NodeANDES	shape3	inplane force	1.555E-03 m	2.424E-03 m	3.896E-03 m
Theoretical solution			7.9488E-03 m	7.9488E-03 m	7.9488E-03 m

The error were listed in Table (63).

Table 63: Errors for 4NodeANDES cantilever beams of irregular shapes with more elements

Element Type	Shape	Force direction	Number of division		
			1	2	4
4NodeANDES	shape1	perpendicular to plane (bending)	2.51%	2.28%	2.20%
4NodeANDES	shape1	inplane force	14.18%	4.78%	1.48%
4NodeANDES	shape2	perpendicular to plane (bending)	3.68%	2.71%	2.31%
4NodeANDES	shape2	inplane force	51.25%	26.34%	11.00%
4NodeANDES	shape3	perpendicular to plane (bending)	16.51%	10.19%	5.38%
4NodeANDES	shape3	inplane force	80.44%	69.51%	50.98%

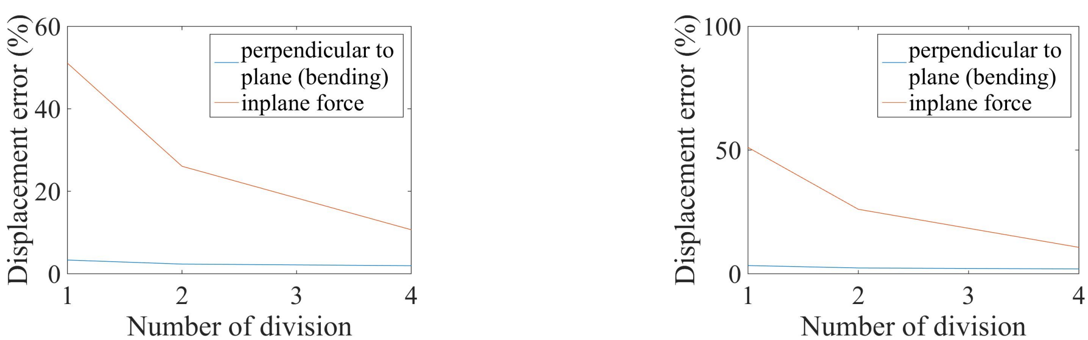
The errors were shown in Figure (183), (184) and (185).



(a) Error scale 0% - 15%

(b) Error scale 0% - 100%

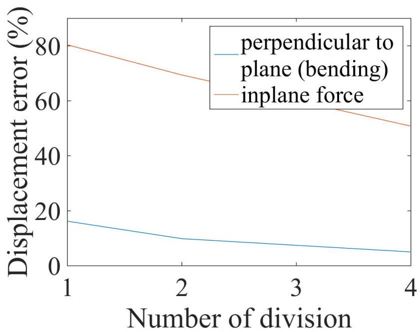
Figure 183: 4NodeANDES cantilever beam for irregular ***Shape 1***
Displacement error versus Number of division



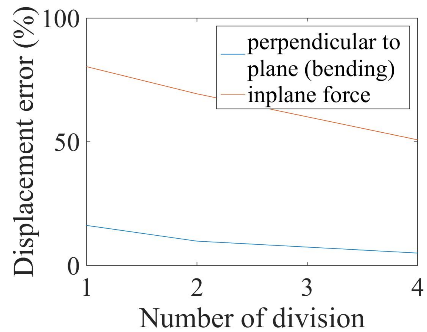
(a) Error scale 0% - 60%

(b) Error scale 0% - 100%

Figure 184: 4NodeANDES cantilever beam for irregular ***Shape 2***
Displacement error versus Number of division



(a) Error scale 0% - 80%



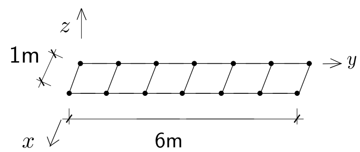
(b) Error scale 0% - 100%

Figure 185: 4NodeANDES cantilever beam for irregular ***Shape 3***
Displacement error versus Number of division

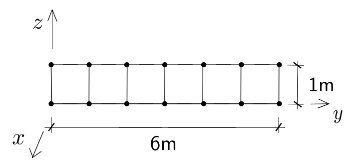
The ESSI model fei files for the table above are here

In this section, the beam was cut into smaller elements with element side length 0.5m and 0.25m respectively. And the element side length of the original models is 1.0m. The numerical models were shown in Figure (186), (187) and (188).

Number of division 1:



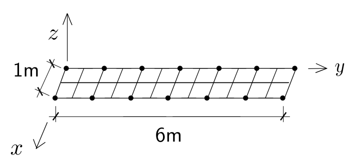
(a) Horizontal plane



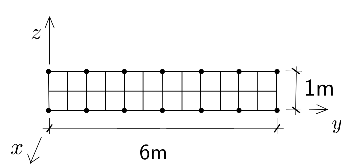
(b) Veritical plane

Figure 186: 4NodeANDES clamped beam with element side length 1.0m

Number of division 2:



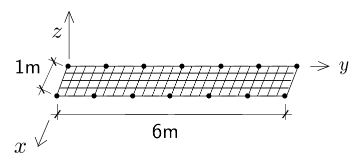
(a) Horizontal plane



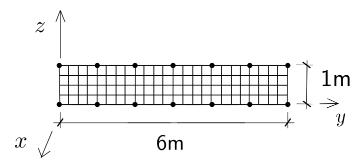
(b) Veritical plane

Figure 187: 4NodeANDES clamped beam with element side length 0.5m

Number of division 4:



(a) Horizontal plane



(b) Veritical plane

Figure 188: 4NodeANDES clamped beam with element side length 0.25m

The ESSI results for the force ***perpendicular to plane (bending)*** were listed in Table (64). The theoretical solution is 1.60E-5 m.

Table 64: Results for 4NodeANDES clamped beams under the force perpendicular to plane (bending)

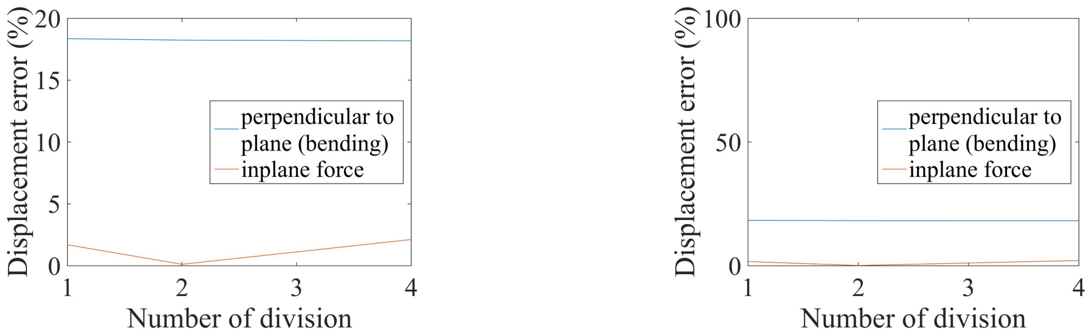
Element Type	Element side length		
	1 m	0.5 m	0.25 m
4NodeANDES	1.347E-05 m	1.35E-05 m	1.35E-05 m
Error	18.36%	18.24%	18.18%

The ESSI results for the ***inplane force*** were listed in Table (65). The theoretical solution is 1.60E-5 m.

Table 65: Results for 4NodeANDES clamped beams under the inplane force

Element Type	Element side length		
	1 m	0.5 m	0.25 m
4NodeANDES	1.62E-05 m	1.65E-05 m	1.69E-05 m
Error	1.70%	0.12%	2.12%

The errors were plotted in Figure (189).



(a) Error scale 0% - 20%

(b) Error scale 0% - 100%

Figure 189: 4NodeANDES clamped beam for different element number
Displacement error versus Number of division

The ESSI model fei files for the table above are here

20.4 Verification of 4NodeANDES square plate with four edges clamped

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{42}$$

The theoretical solution is

$$d = \alpha_c \frac{qa^4}{D} = 0.00406 \times \frac{100 N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \ N \cdot m} = 2.2015 \times 10^{-3} m \tag{43}$$

where α_c is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[14] α_c is 0.00406.

The 4NodeANDES were shown in Figure (190) - (195).

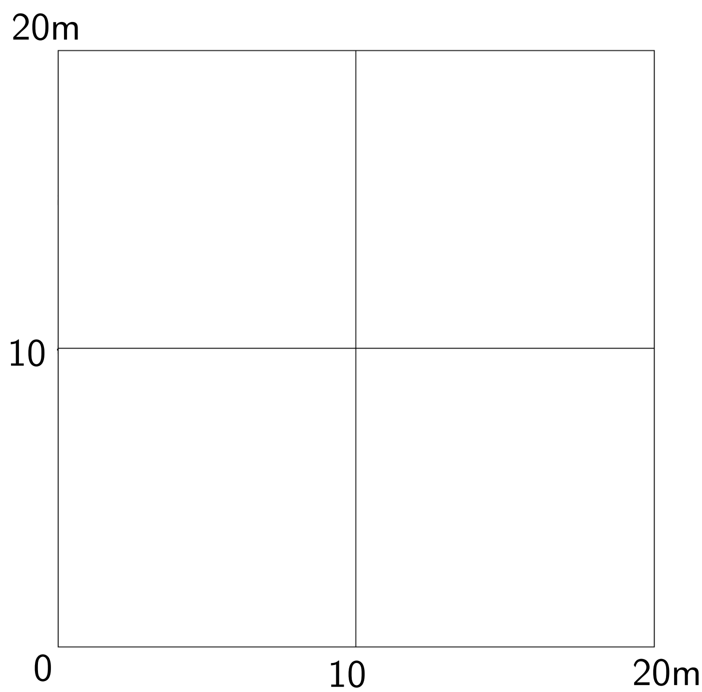


Figure 190: 4NodeANDES edge clamped square plate with element side length 10m

[14] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page120, 1959.

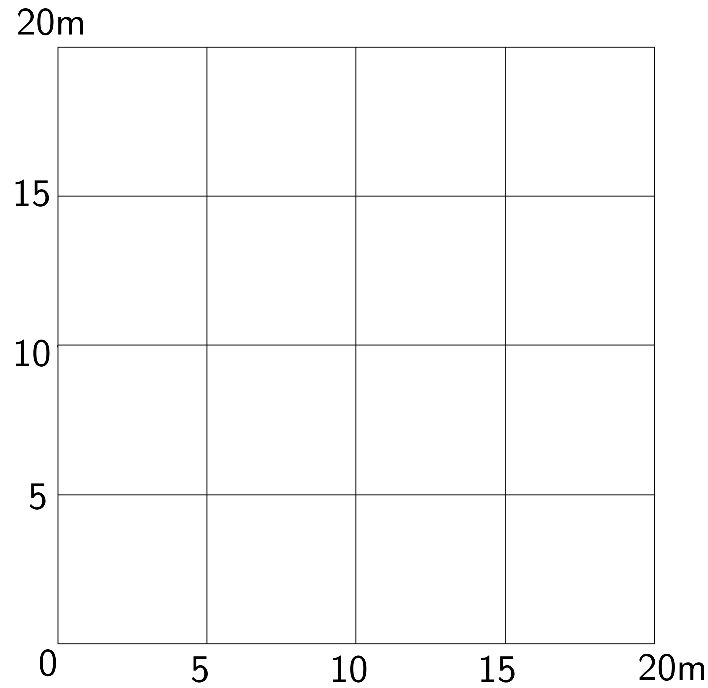


Figure 191: 4NodeANDES edge clamped square plate with element side length 5m

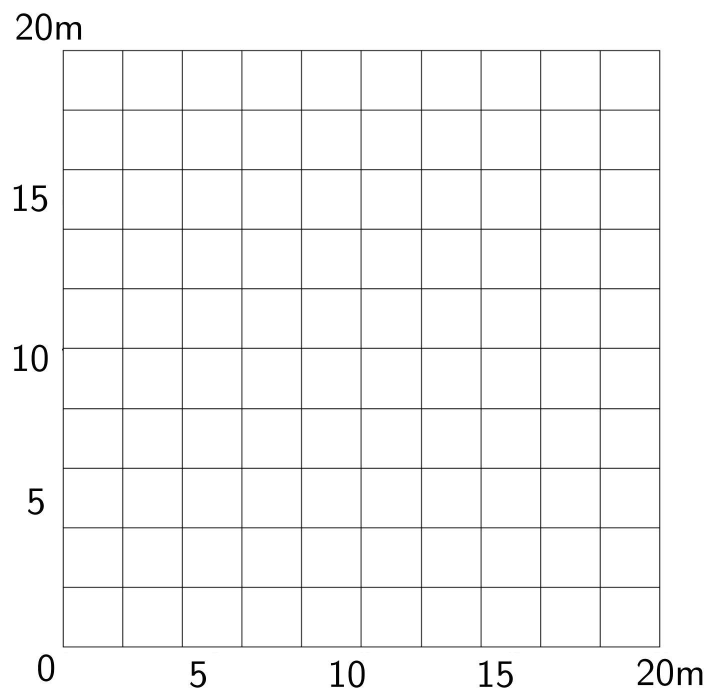


Figure 192: 4NodeANDES edge clamped square plate with element side length 2m

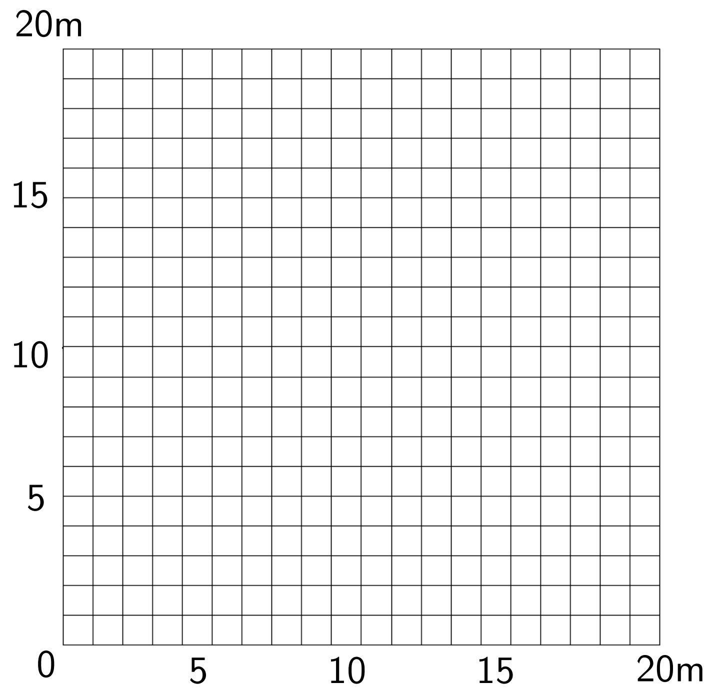


Figure 193: 4NodeANDES edge clamped square plate with element side length 1m

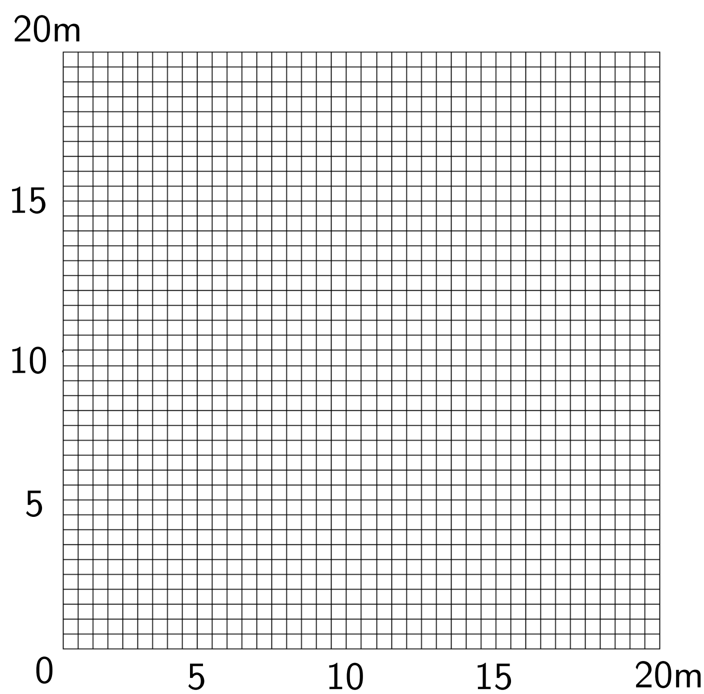


Figure 194: 4NodeANDES edge clamped square plate with element side length 0.5m

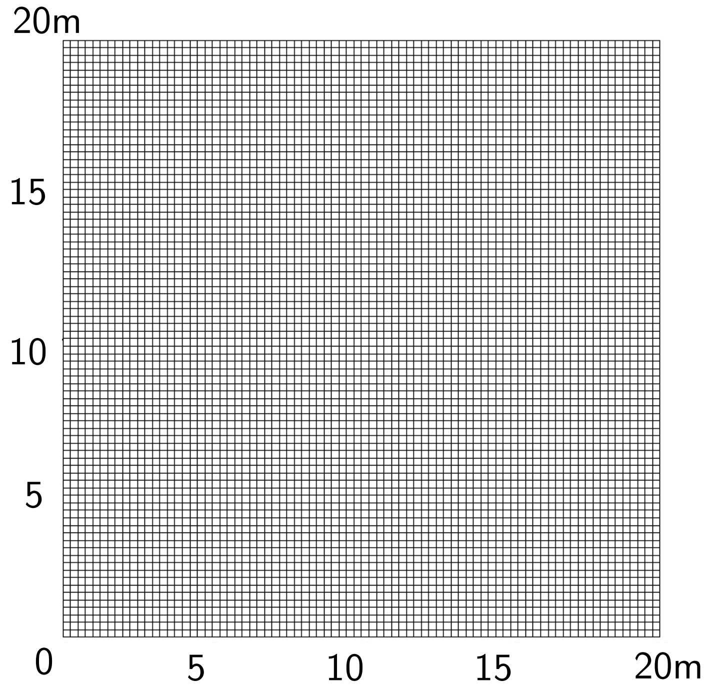


Figure 195: 4NodeANDES edge clamped square plate with element side length 0.25m

The results were listed in Table (66).

Table 66: Results for 4NodeANDES square plate with four edges clamped

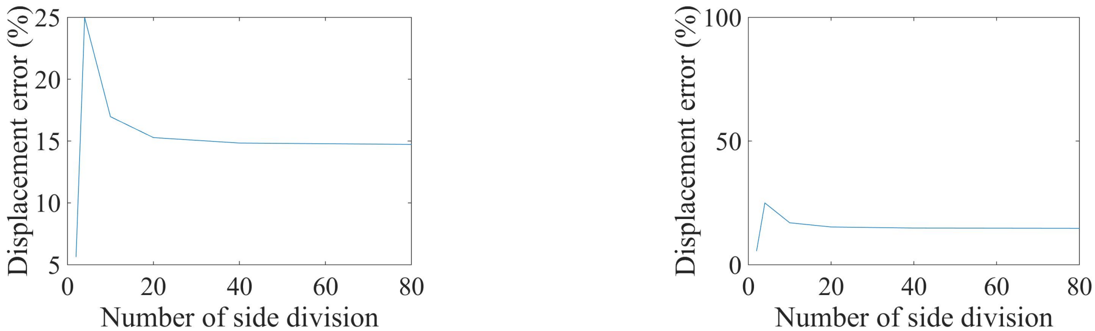
Element type	4NodeANDES	Theoretical
Element side length	Height:1.00m	displacement
10m	2.33E-003 m	2.20E-03 m
5m	2.75E-003 m	2.20E-03 m
2m	2.58E-003 m	2.20E-03 m
1m	2.54E-003 m	2.20E-03 m
0.5m	2.53E-003 m	2.20E-03 m
0.25m	2.53E-003 m	2.20E-03 m

The errors were listed in Table (67).

Table 67: Errors for 4NodeANDES square plate with four edges clamped

Element type	4NodeANDES
Element side length	Height:1.00m
10m	5.65%
5m	24.98%
2m	16.97%
1m	15.28%
0.5m	14.84%
0.25m	14.73%

The errors were plotted in Figure (196).



(a) Error scale 0% - 25%

(b) Error scale 0% - 100%

Figure 196: 4NodeANDES square plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

20.5 Verification of 4NodeANDES square plate with four edges simply supported

Problem description: Length=20m, Width=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \; N \cdot m \tag{44}$$

The theoretical solution is

$$d = \alpha_s \frac{qa^4}{D} = 0.00126 \times \frac{100 N/m^2 \times 20^4 m^4}{9.1575 \times 10^6 \; N \cdot m} = 7.0936 \times 10^{-3} m \tag{45}$$

where α_s is a coefficient, which depends on the ratio of plate length to width. In this problem, the coefficient[15] α_s is 0.00126.

The 4NodeANDES were shown in Figure (197) - (202).

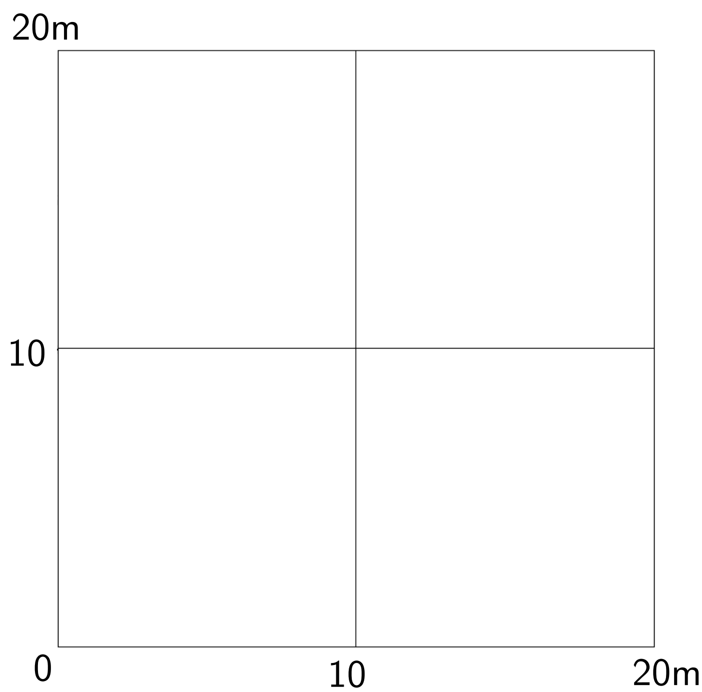


Figure 197: 4NodeANDES edge simply supported square plate with element side length 10m

[15] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page202, 1959.

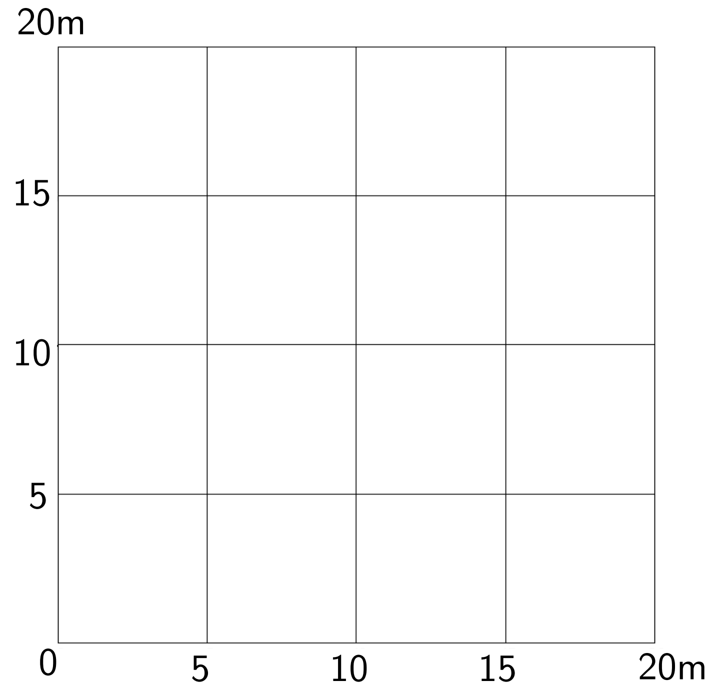


Figure 198: 4NodeANDES edge simply supported square plate with element side length 5m

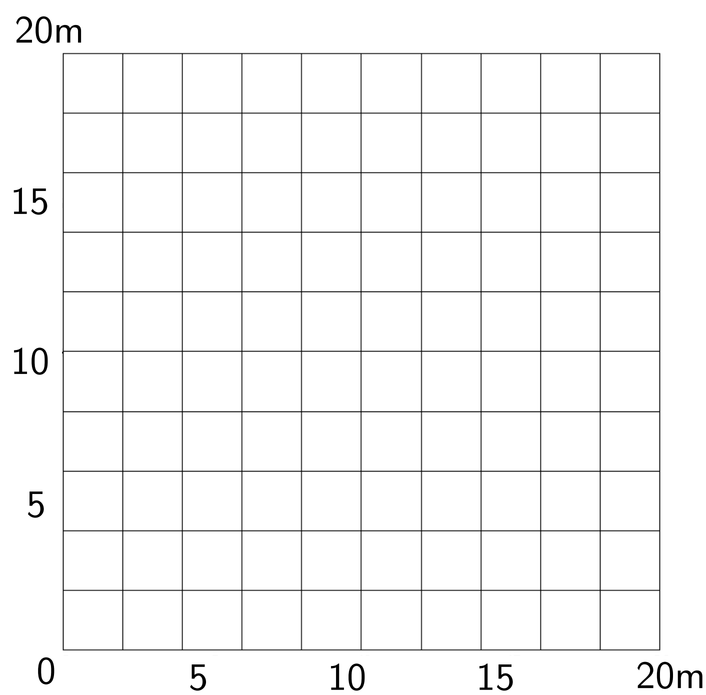


Figure 199: 4NodeANDES edge simply supported square plate with element side length 2m

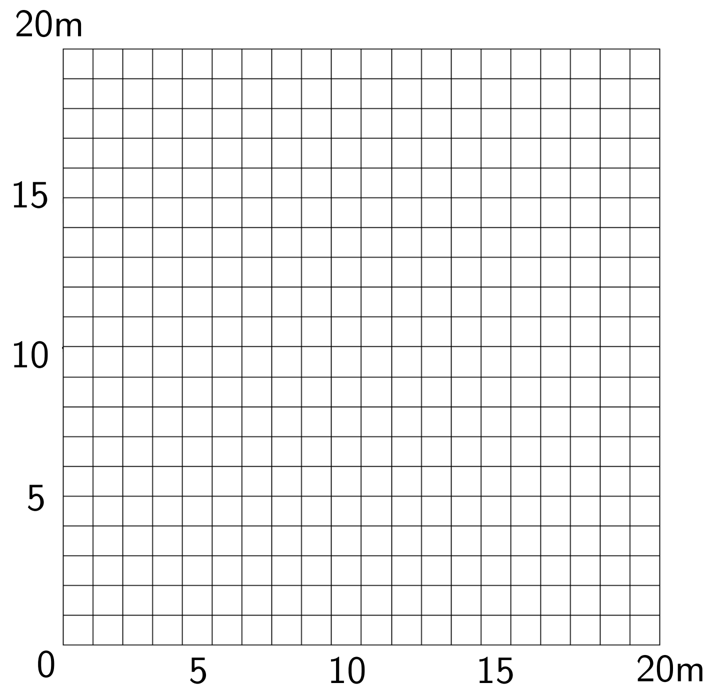


Figure 200: 4NodeANDES edge simply supported square plate with element side length 1m

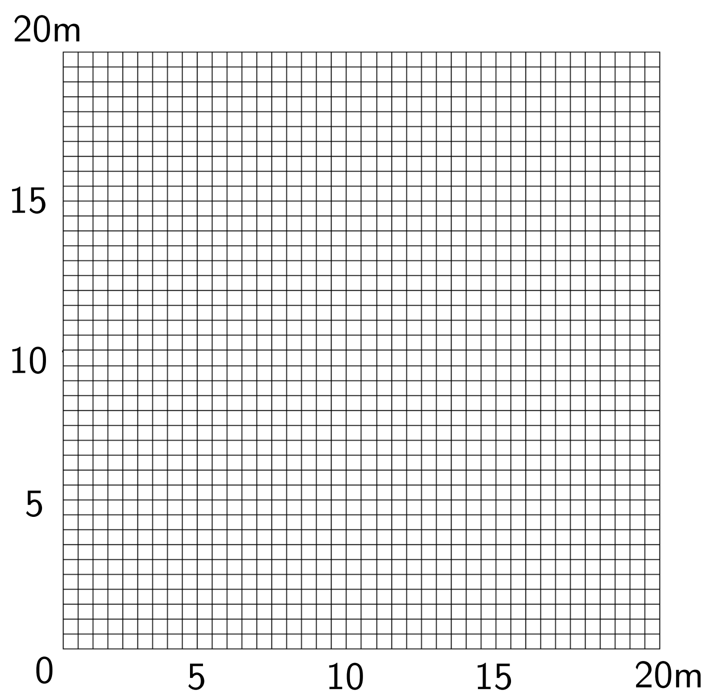


Figure 201: 4NodeANDES edge simply supported square plate with element side length 0.5m

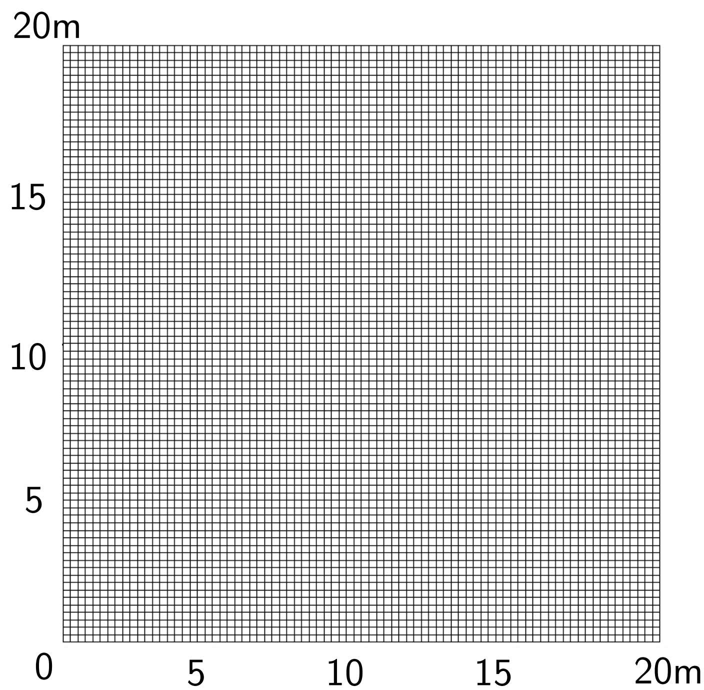


Figure 202: 4NodeANDES edge simply supported square plate with element side length 0.25m

The results were listed in Table (68).

Table 68: Results for 4NodeANDES square plate with four edges simply supported

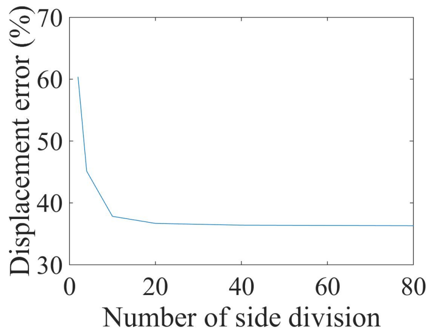
Element type	4NodeANDES	Theoretical
Element side length	Height:1.00m	displacement
10m	1.14E-002 m	7.09E-03 m
5m	1.03E-002 m	7.09E-03 m
2m	9.78E-003 m	7.09E-03 m
1m	9.70E-003 m	7.09E-03 m
0.5m	9.68E-003 m	7.09E-03 m
0.25m	9.67E-003 m	7.09E-03 m

The errors were listed in Table (69).

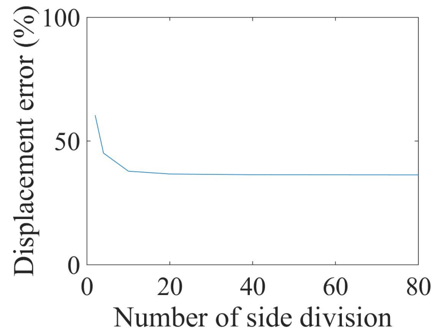
Table 69: Errors for 4NodeANDES square plate with four edges simply supported

Element type	4NodeANDES
Element side length	Height:1.00m
10m	60.34%
5m	45.14%
2m	37.83%
1m	36.69%
0.5m	36.40%
0.25m	36.32%

The errors were plotted in Figure (203).



(a) Error scale 0% - 70%



(b) Error scale 0% - 100%

Figure 203: 4NodeANDES square plate with edge simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

20.6 Verification of 4NodeANDES circular plate with all edges clamped

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are clamped.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{46}$$

The theoretical solution[16] is

$$d = \frac{qa^4}{64D} = \frac{100N/m^2 \times 10^4 m^4}{64 \times 9.1575 \times 10^6 \ N \cdot m} = 1.7106 \times 10^{-3} m \tag{47}$$

The 4NodeANDES were shown in Figure (204) - (209).

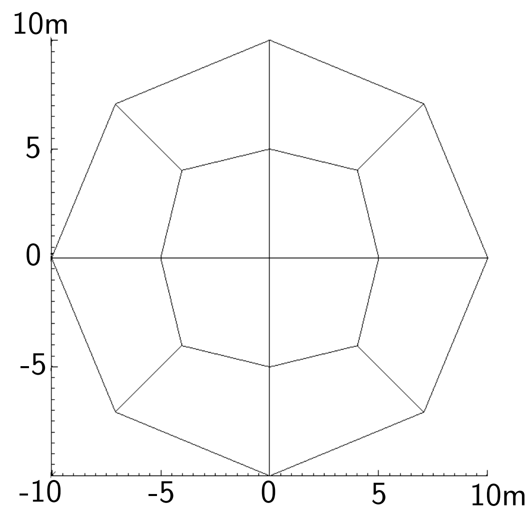


Figure 204: 4NodeANDES edge clamped circular plate with element side length 10m

[16] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

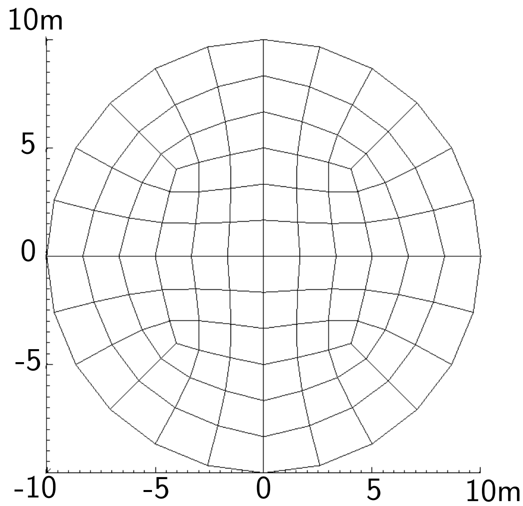


Figure 205: 4NodeANDES edge clamped circular plate with element side length 5m

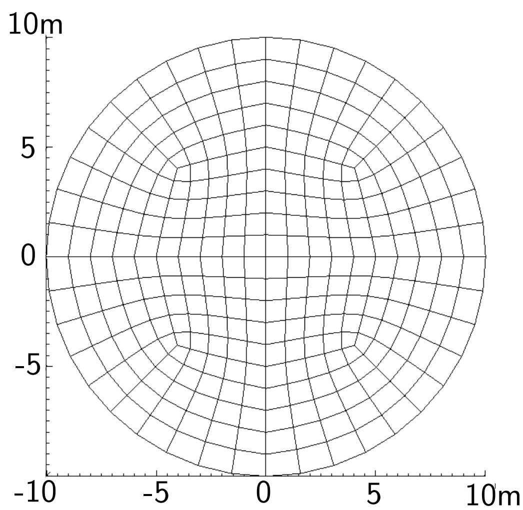


Figure 206: 4NodeANDES edge clamped circular plate with element side length 2m

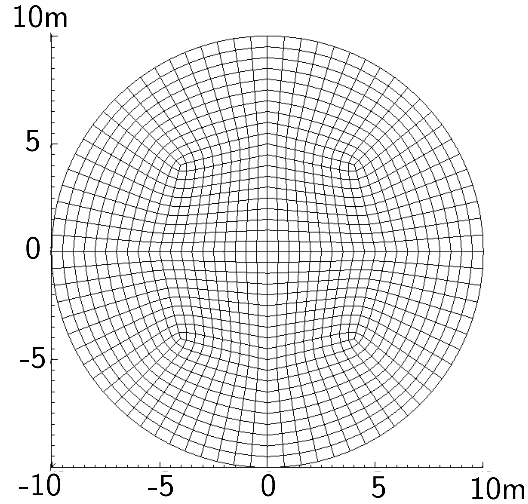


Figure 207: 4NodeANDES edge clamped circular plate with element side length 1m

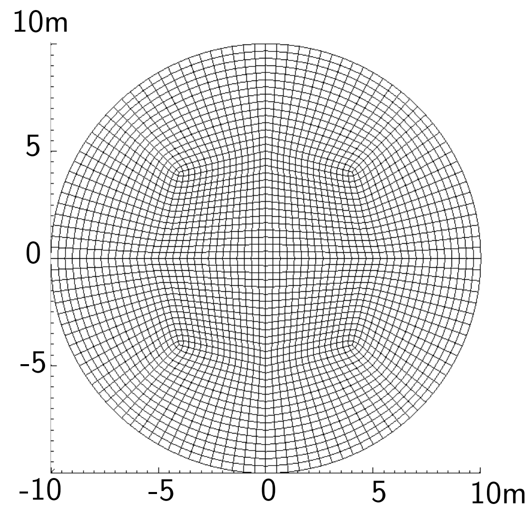


Figure 208: 4NodeANDES edge clamped circular plate with element side length 0.5m

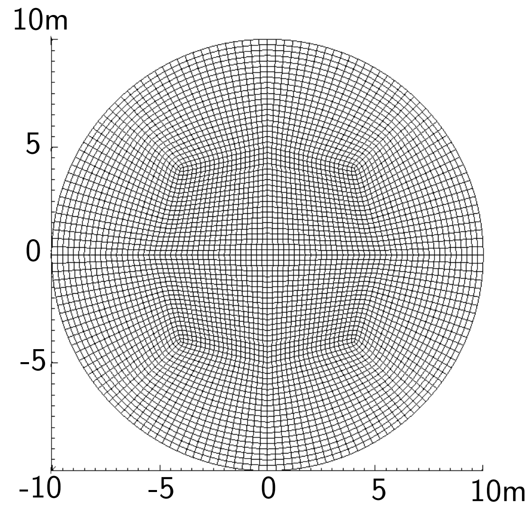


Figure 209: 4NodeANDES edge clamped circular plate with element side length 0.25m

The results were listed in Table (70).

Table 70: Results for 4NodeANDES circular plate with four edges clamped

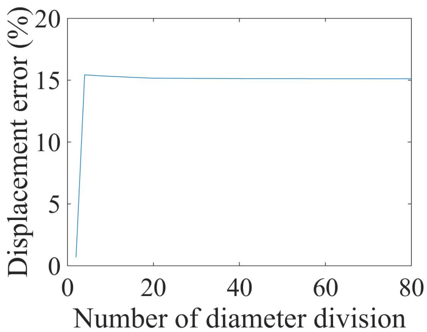
Element type	4NodeANDES	Theoretical
Element side length	Height:1.00m	displacement
10m	1.69E-003 m	1.706E-03 m
5m	1.97E-003 m	1.706E-03 m
2m	1.97E-003 m	1.706E-03 m
1m	1.96E-003 m	1.706E-03 m
0.5m	1.96E-003 m	1.706E-03 m
0.25m	1.96E-003 m	1.706E-03 m

The errors were listed in Table (71).

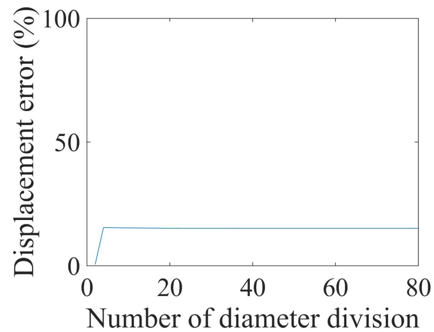
Table 71: Errors for 4NodeANDES circular plate with four edges clamped

Element type	4NodeANDES
Element side length	Height:1.00m
10m	0.71%
5m	15.43%
2m	15.31%
1m	15.16%
0.5m	15.13%
0.25m	15.12%

The errors were shown in Figure (210).



(a) Error scale 0% - 20%



(b) Error scale 0% - 100%

Figure 210: 4NodeANDES circular plate with edge clamped
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

20.7 Verification of 4NodeANDES circular plate with all edges simply supported

Problem description: Diameter=20m, Height=1m, Force=100N, E=1E8Pa, $\nu = 0.3$.

The four edges are simply supported.

The load is the uniform normal pressure on the whole plate.

The plate flexural rigidity is

$$D = \frac{Eh^3}{12(1-\nu^2)} = \frac{10^8 N/m^2 \times 1^3 m^3}{12 \times (1-0.3^2)} = 9.1575 \times 10^6 \ N \cdot m \tag{48}$$

The theoretical solution[17] is

$$d = \frac{(5+\nu)qa^4}{64(1+\nu)D} = \frac{(5+0.3) \times 100 N/m^2 \times 10^4 m^4}{64 \times (1+0.3) \times 9.1575 \times 10^6 \ N \cdot m} = 6.956 \times 10^{-3} m \tag{49}$$

The 4NodeANDES were shown in Figure (211) - (216).

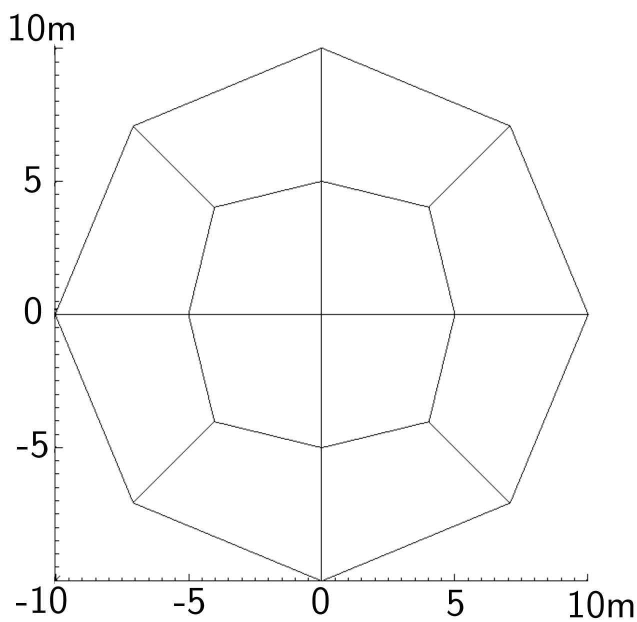


Figure 211: 4NodeANDES edge simply supported circular plate with element side length 10m

[17] Stephen Timoshenko, Theory of plates and shells (2nd edition). MrGRAW-Hill Inc, page55, 1959.

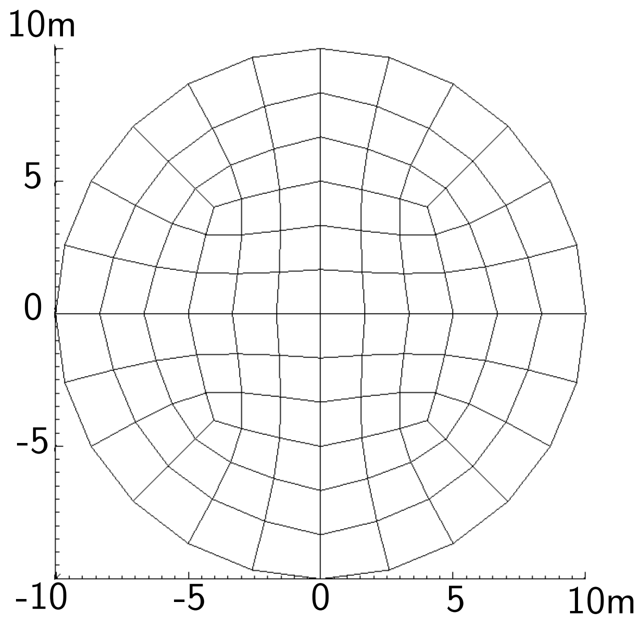


Figure 212: 4NodeANDES edge simply supported circular plate with element side length 5m

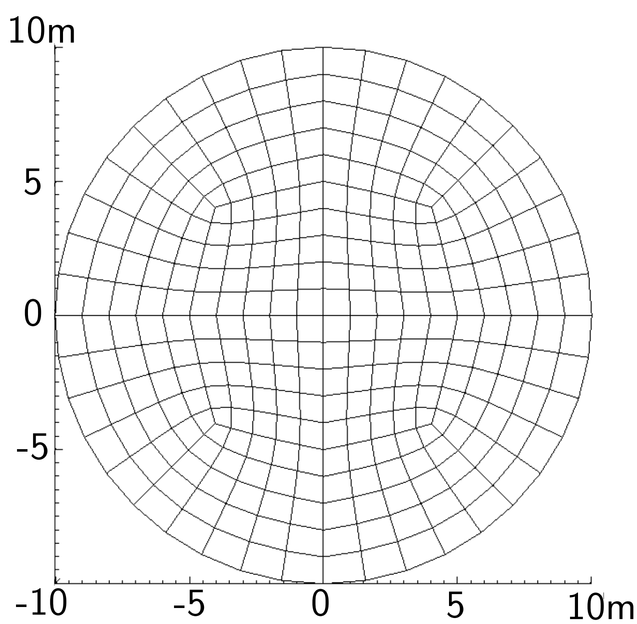


Figure 213: 4NodeANDES edge simply supported circular plate with element side length 2m

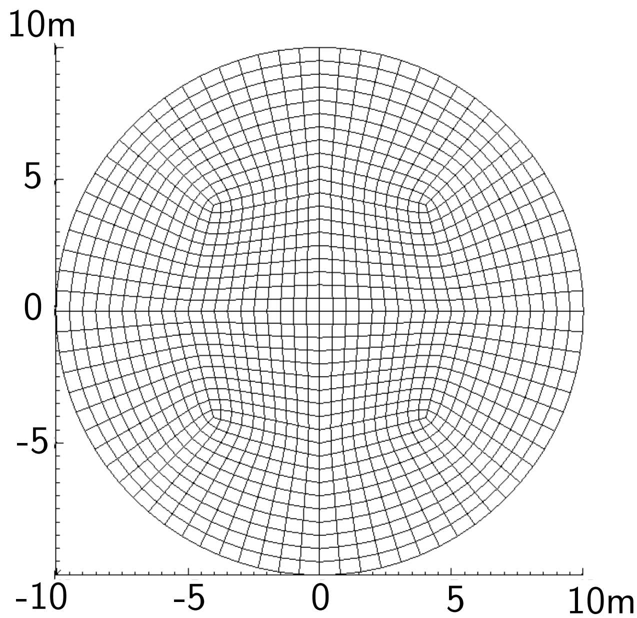


Figure 214: 4NodeANDES edge simply supported circular plate with element side length 1m

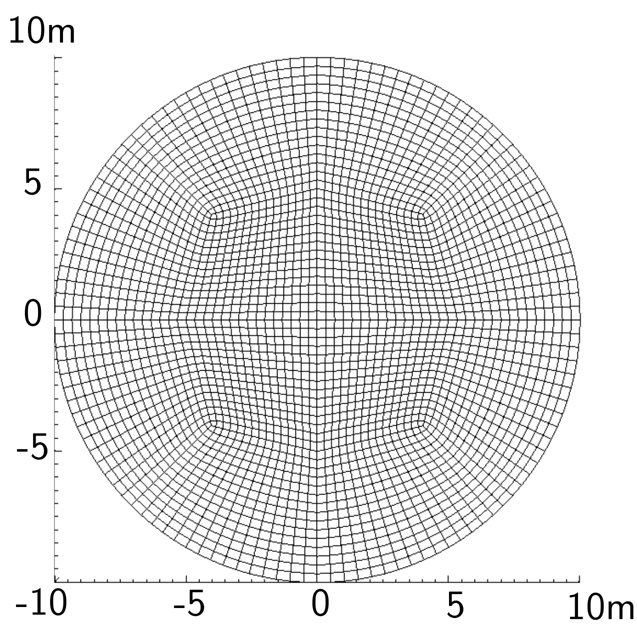


Figure 215: 4NodeANDES edge simply supported circular plate with element side length 0.5m

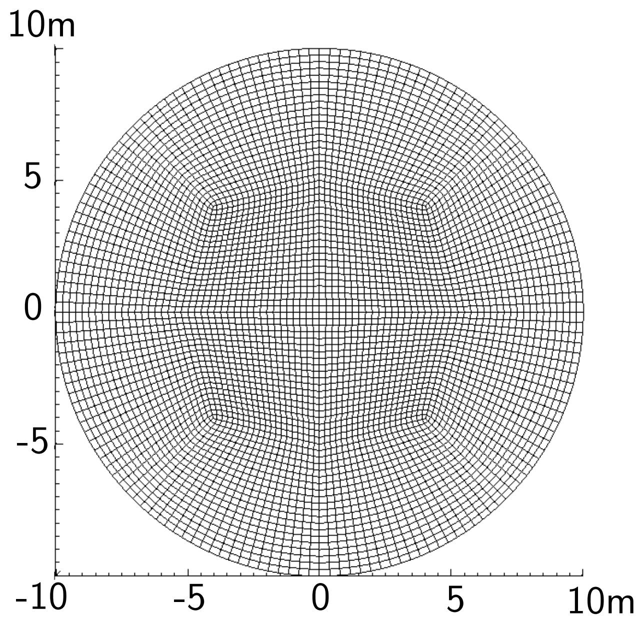


Figure 216: 4NodeANDES edge simply supported circular plate with element side length 0.25m

The results were listed in Table (72).

Table 72: Results for 4NodeANDES cicular plate with four edges simply supported

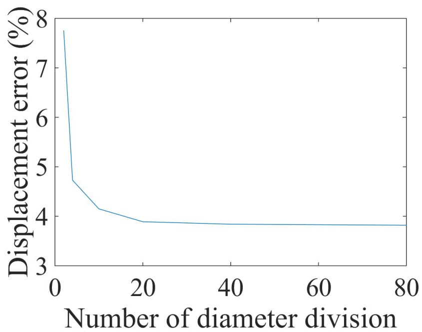
Element type	4NodeANDES	Theoretical
Element side length	Height:1.00m	displacement
10m	7.50E-003 m	6.956E-03 m
5m	7.29E-003 m	6.956E-03 m
2m	7.25E-003 m	6.956E-03 m
1m	7.23E-003 m	6.956E-03 m
0.5m	7.22E-003 m	6.956E-03 m
0.25m	7.22E-003 m	6.956E-03 m

The errors were listed in Table (73).

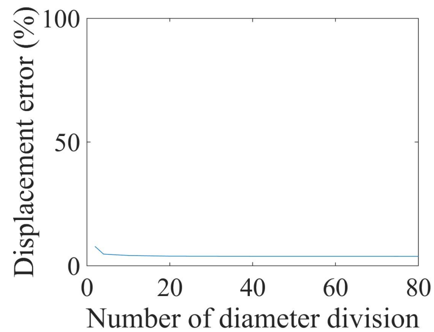
Table 73: Errors for 4NodeANDES cicular plate with four edges simply supported

Element type	4NodeANDES
Element side length	Height:1.00m
10m	7.75%
5m	4.73%
2m	4.15%
1m	3.89%
0.5m	3.84%
0.25m	3.82%

The errors were plotted in Figure (217).



(a) Error scale 0% - 8%



(b) Error scale 0% - 100%

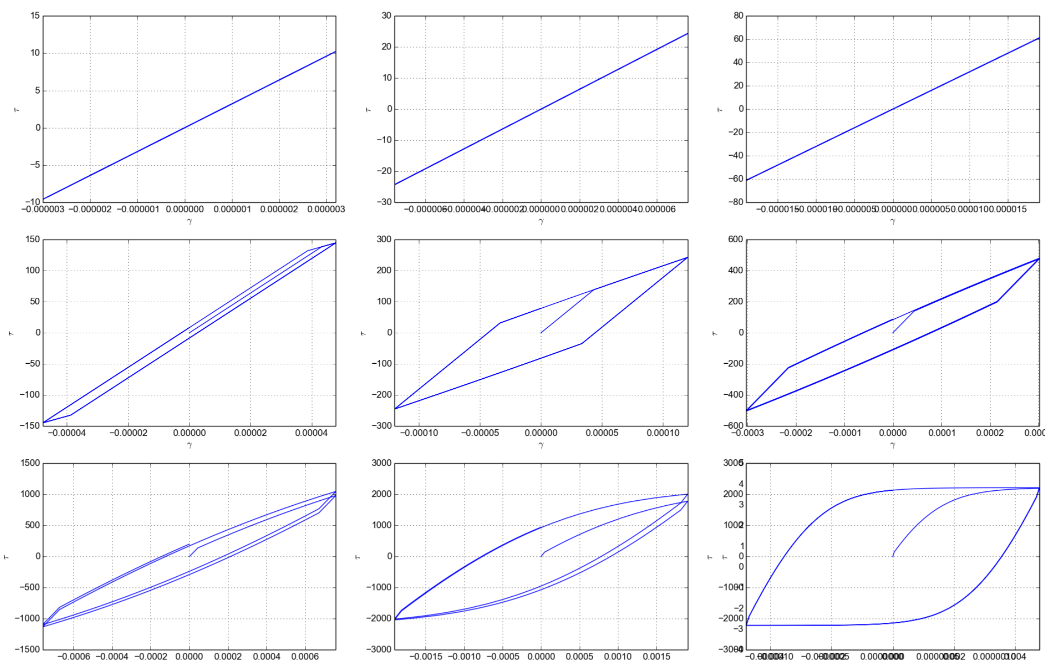
Figure 217: 4NodeANDES circular plate with edge simply supported
Displacement error versus Number of side division

The ESSI model fei files for the table above are here

21 G/Gmax plot by Drucker-Prager Armstrong Frederick

21.1 Plot stress-strain

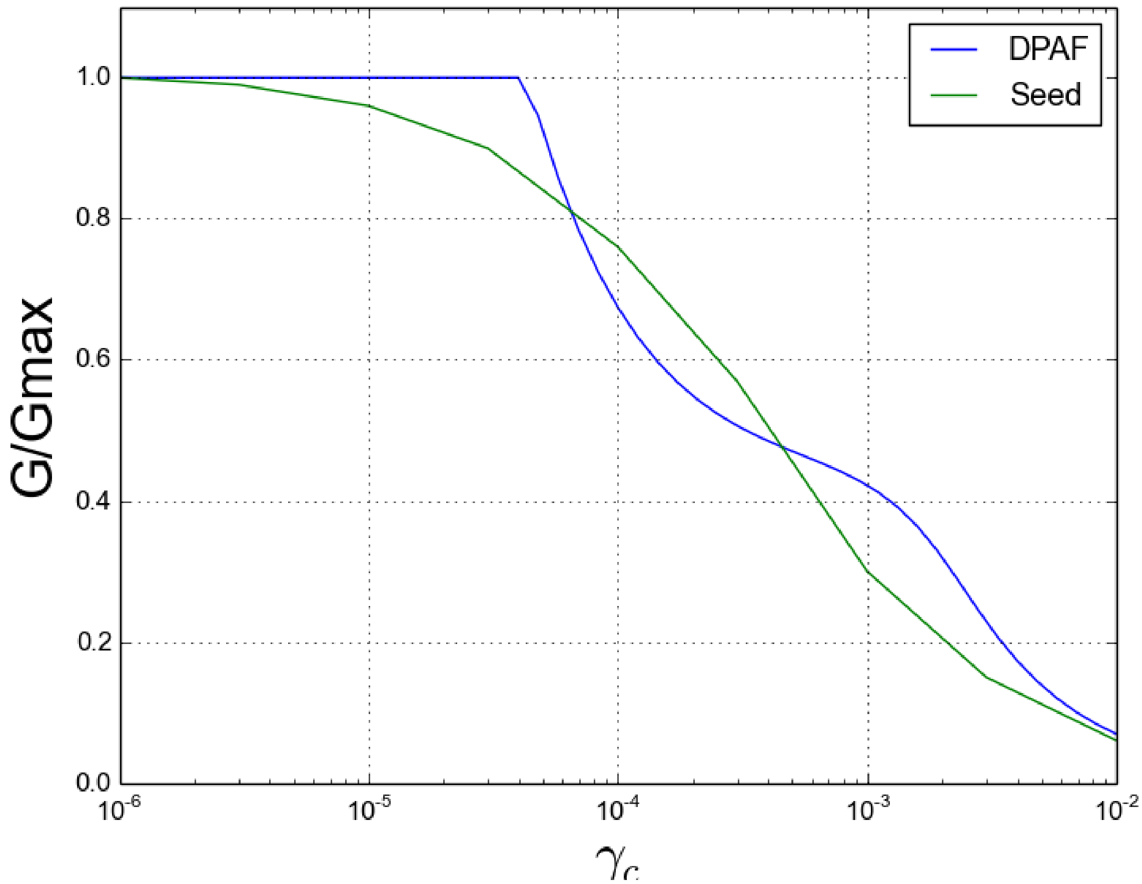
Figure 218: The stress-strain diagram for different γ_c



When γ_c are very small, the materials are in the elastic range.

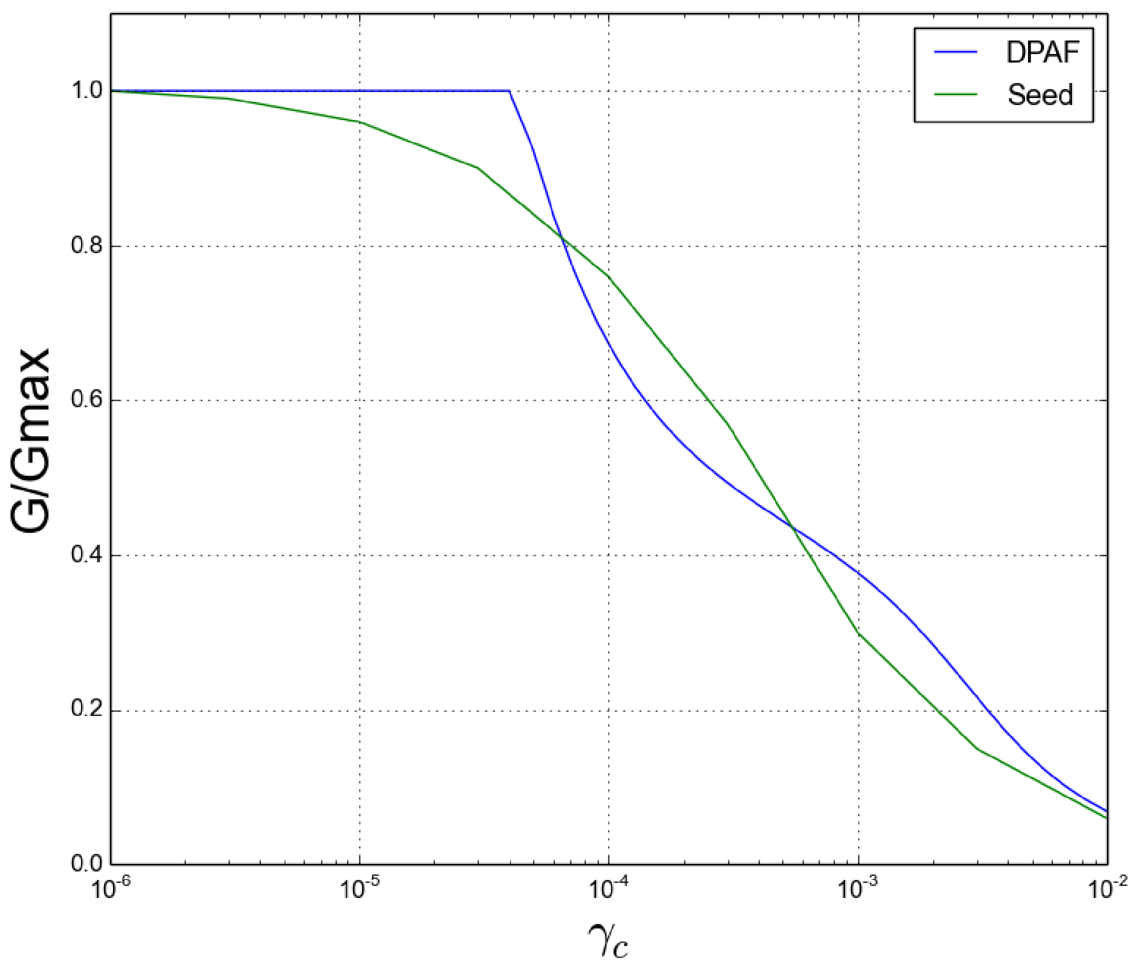
21.2 Plot G/Gmax

Figure 219: The G/Gmax diagram with multiple loops



The $x - axis$ labels are the unitless γ_c, **not** in percent.
So the max shear strain is 1 percent here.

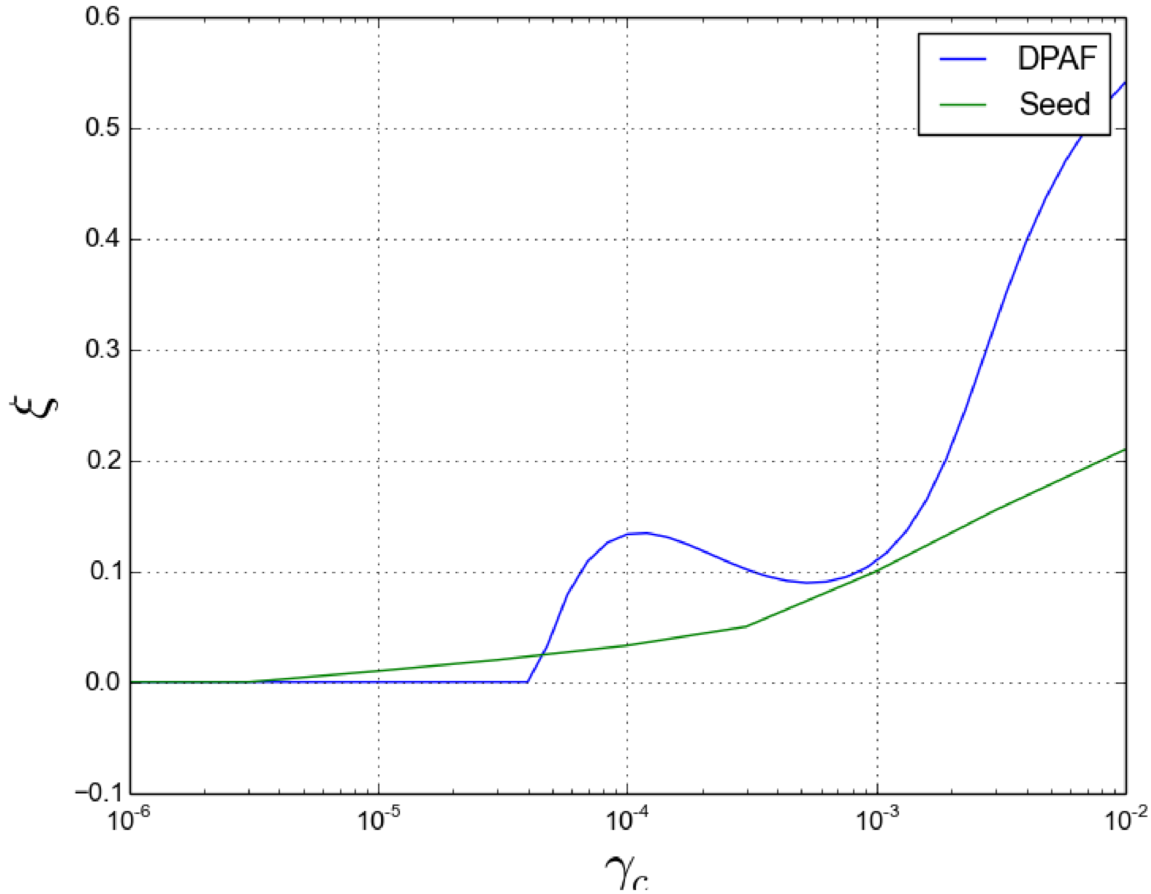
Figure 220: The G/Gmax diagram in one loading



The G/Gmax curve above is plotted by a different method. Only one loading is used to plot G/Gmax. The different plot functions are attached at the end. The curves between different plotting methods are almost the same.

21.3 Plot damping ratio

Figure 221: The damping ratio



The parameters I am using

```
// Yuan is testing different Armstrong Frederick Drucker Prager (AFDP) parameters:
// For Real ESSI, Units are processed by other functions.
// Here, all units should be the SI units.
// initial_von_mises_radius is actually the variable 'k' in the function of yield suface
double initial_von_mises_radius =0.0008;
double kinematic_hardening_AF_ha =12;
double kinematic_hardening_AF_cr =1000;
double isotropic_hardening_rate =0.0;
double elastic_modulus       =4e6;  //4MPa
double poisson_ratio         =0.25;
double density               =0.0;
double initial_confining_stress =3e5; //300kPa
```

In the Appendix (21.4), you would see the whole cpp code I am using to define the materials.

21.4 Appendix (Code)

21.4.1 Framework: A cpp main function to test the materials directly

```

#include "ClassicElastoplasticMaterial.h"
#include "ConstitutiveModels/DruckerPragerArmstrongFrederick.h"
#include <iostream>
#include <fstream>
#include <vector>
#include <math.h>

using namespace std;

void print_this_step(const int &printStep, NDMaterialLT* test_material, ofstream &
    out_strain,ofstream & out_stress){
  // Terminal: print the input strain:
  cout<<"-----------------------------step "<<printStep<<"
      start------------------------------"<<endl;
  auto input_strain = test_material -> getStrainTensor();
  cout<<"The current strain: "<<endl;
  cout<<input_strain<<endl;

  // Terminal: Output the calculated stress
  auto stress_result = test_material -> getStressTensor();
  cout<<"The output stress: "<<endl;
  cout<<stress_result<<endl;
  cout<<"--------------------------------step end-------------------------------"<<endl;

  // output to file as well
  out_strain<<input_strain(0,2)<<endl;
  out_stress<<stress_result(0,2)<<endl;
}

int main(){
  // The corresponding constructor for reference:
    //First constructor, creates a material at its "ground state" from its parameters.
    // DruckerPragerArmstrongFrederick(int tag_in,
    //  double k0_in,
    //  double ha_alpha,
    //  double cr_alpha,
    //  double H_k,
    //  double E,
    //  double nu,
    //  double rho_,
    //  double p0) ;

  // Define and input the material properties.
  int    material_tag            =1;

  // Yuan is testing different Armstrong Frederick Drucker Prager (AFDP) parameters:
```

```
// For Real ESSI, units are processed by other functions.
// Here, all units should be the SI units.
// initial_von_mises_radius is actually the variable 'k' in the function of yield suface
double initial_von_mises_radius =0.0008;
double kinematic_hardening_AF_ha =12;
double kinematic_hardening_AF_cr =1000;
double isotropic_hardening_rate =0.0;
double elastic_modulus        =4e6;  //4MPa
double poisson_ratio          =0.25;
double density                =0.0;
double initial_confining_stress =3e5; //300kPa

// Set the integration rule and tolerance to the material:
int method = (int) NDMaterialLT_Constitutive_Integration_Method::Euler_One_Step;
double f_relative_tol=0.001;
double stress_relative_tol=0.001;
double n_max_iterations=10;

  NDMaterialLT::set_constitutive_integration_method(
    method,
    f_relative_tol,
    stress_relative_tol,
    n_max_iterations);


// generate the uniformly distributed values on the logarithmic axis.
double strain_start=0.0001/100.0;
double strain_end =1.0/100.0;
double Lstart=log10(strain_start);
double Lend=log10(strain_end);
  int num_gammac=51;
std::vector<double> gammac(num_gammac,0.0);

for (int i = 0; i < num_gammac; ++i)
{
  gammac[i] = pow(10.0, Lstart+(Lend-Lstart)*i/(num_gammac-1) );
}

double steplength_loading=strain_start/10.0;

  // the Gammac for the step
double theGammac=0.0;
for (int i_gamma = 0; i_gamma < num_gammac; ++i_gamma)
{
  // gammac is the maximum shear strain for this cyclic loading
  // theGammac=steplength_gammac*(i_gamma+1);
  theGammac=gammac[i_gamma];


  // Declare the new materials
    NDMaterialLT* DPrackerAF_material = new DruckerPragerArmstrongFrederick( material_tag,
      initial_von_mises_radius,
      kinematic_hardening_AF_ha,
      kinematic_hardening_AF_cr,
      isotropic_hardening_rate,
```

```
      elastic_modulus,
      poisson_ratio,
      density,
      initial_confining_stress) ;

    // Output the parameters:
    cout<<"------------------------------Start Testing----------------------------"<<endl;
   cout<<"The input Drucker Prager Armstrong Frederick material parameters:"<<endl;
   cout<<"initial_von_mises_radius : "<<initial_von_mises_radius<<endl;
   cout<<"kinematic_hardening_AF_ha : "<<kinematic_hardening_AF_ha<<endl;
   cout<<"kinematic_hardening_AF_cr : "<<kinematic_hardening_AF_cr<<endl;
   cout<<"isotropic_hardening_rate : "<<isotropic_hardening_rate<<endl;
   cout<<"elastic_modulus        : "<<elastic_modulus<<endl;
   cout<<"poisson_ratio          : "<<poisson_ratio<<endl;
   cout<<"density                : "<<density<<endl;
   cout<<"initial_confining_stress : "<<initial_confining_stress<<endl;
   cout<<"-----------------------------------------------------------------------"<<endl;


   // Print out the initial state of the materials:
   cout<<"The initial state:"<<endl;
   auto initial_strain = DPrackerAF_material -> getStrainTensor();
   cout<<"The initial strain state: "<<endl;
   cout<<initial_strain<<endl;

   auto initial_stress = DPrackerAF_material -> getStressTensor();
   cout<<"The initial stress state: "<<endl;
   cout<<initial_stress<<endl;
   cout<<"-----------------------------------------------------------------------"<<endl;


   // create the output filename here:
   ofstream outfile_strain;
   ofstream outfile_stress;
   string outfilename_strain= "shearStrain"+ to_string(i_gamma+1) +".txt" ;
   string outfilename_stress= "shearStress"+ to_string(i_gamma+1) +".txt" ;

   outfile_strain.open(outfilename_strain);
   outfile_stress.open(outfilename_stress);

   // write the initial state to the file
   outfile_strain<<"0.0"<<endl;
   outfile_stress<<"0.0"<<endl;


   cout<<"------------------------------Start loading------------------------------"<<endl;

   int mystep=0;
   DTensor2 myinputStrain( 3, 3, 0.0 );

   short loop=0;
   for (int loop = 0; loop < 2; ++loop){
     // Input positive shear strain increment each step
     myinputStrain(0,2)=steplength_loading;
     myinputStrain(2,0)=myinputStrain(0,2);
```

```
      for (int i = 0; i*steplength_loading < theGammac ; ++i)
      {
        DPrackerAF_material->setTrialStrainIncr(myinputStrain);
        DPrackerAF_material->commitState();
        print_this_step(++mystep,DPrackerAF_material,outfile_strain,outfile_stress);
      }

      // Input nagative shear strain increment each step
      myinputStrain(0,2)=-steplength_loading;
      myinputStrain(2,0)=myinputStrain(0,2);
      for (int i = 0; i*steplength_loading < theGammac*2; ++i)
      {
        DPrackerAF_material->setTrialStrainIncr(myinputStrain);
        DPrackerAF_material->commitState();
        print_this_step(++mystep,DPrackerAF_material,outfile_strain,outfile_stress);
      }

      // Input positive shear strain increment each step
      myinputStrain(0,2)=steplength_loading;
      myinputStrain(2,0)=myinputStrain(0,2);
      for (int i = 0; i*steplength_loading < theGammac; ++i)
      {
        DPrackerAF_material->setTrialStrainIncr(myinputStrain);
        DPrackerAF_material->commitState();
        print_this_step(++mystep,DPrackerAF_material,outfile_strain,outfile_stress);
      }
    }

    // garbage collection before exit this loop.
    outfile_strain.close();
    outfile_stress.close();
    delete DPrackerAF_material;
  }

  return 0;

}

// previous data:
// calculate 9 values only. The plotted curve are not smooth but this will be fast.
  // double Seed_strain[]={0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1};
    // int num_gammac=sizeof(Seed_strain)/sizeof(double);
  // vector<double> gammac (Seed_strain, Seed_strain + num_gammac);
  // for(auto& g: gammac) g=g/100;
```

21.4.2 The function to plot the results with 51 loops

I did not use LBNL's function to plot the results, since I am not using the HDF5 output. I write my own function plot the results.

```
#!/usr/bin/python

import numpy as np
import matplotlib.pyplot as plt
pi=np.pi

# Open and read files
# fig1 for stress-strain results:
fig1=plt.figure()
G_Gmax=[1.0]
gamma=[0.0]
xi=[0.0]
for i in range(1,10,1):

  filename_strain ="shearStrain"+str(i)+".txt"
  filename_stress ="shearStress"+str(i)+".txt"
  with open(filename_strain) as fStrain:
    shearStrain = fStrain.read()

  with open(filename_stress) as fStress:
    shearStress = fStress.read()

  # split the string to strains.
  strain_spl=shearStrain.split('\n')
  stress_spl=shearStress.split('\n')
  # remove the last blankspace.
  strain_spl.pop()
  stress_spl.pop()
  # convert string to int
  strain=map(float, strain_spl)
  stress=map(float, stress_spl)


  strain_max=max(strain)
  stress_max=max(stress)
  G_sec=stress_max/strain_max

  stress_2nd=stress[1]
  strain_2nd=strain[1]
  G_max=stress_2nd/strain_2nd

  thisG_Gmax=G_sec/G_max
  G_Gmax.append(thisG_Gmax)
  gamma.append(strain_max)

  # ------------------------------------
    # add the subplot to the stress-strain results
  thisFig = fig1.add_subplot(3,3,i)
  thisFig.plot(strain,stress)
```

```
  plt.xlabel(r'$\gamma$')
  plt.ylabel(r'$\tau$')
  plt.grid(True)
  plt.autoscale(enable=True,axis='x',tight=True) #
  # plt.xlim( (-0.1, 0.1) )
  # plt.ylim((-1e6,1e6))


  # -----------------------------------
  # calculate the damping ratio
  # find the number of steps in the one loop
  num_zero_strain=0
  strain_len=len(strain)
  index=strain_len
  while True:
    # the last is zero.
    index=index-1
    # find the second last and third last zeros.
    if abs(strain[index-1]) <1e-15:
      num_zero_strain=num_zero_strain+1
    if num_zero_strain==2:
      break;
  num_steps_perloop=strain_len-index+1
  strain_last_loop=strain[-num_steps_perloop:]
  stress_last_loop=stress[-num_steps_perloop:]

  stress_bottom_last_loop= min(stress_last_loop)
  stress_top_bottom = [i-stress_bottom_last_loop for i in stress_last_loop]


  A_loop=0.0
  for i in range(num_steps_perloop-1):
    A_loop=A_loop+(stress_top_bottom[i]+stress_top_bottom[i+1])/2*(strain_last_loop[i+1]-strain_last_loop

  damping_ratio=A_loop/strain_max/strain_max/2.0/pi/G_sec
  xi.append(damping_ratio)
  print "-------------------------"
  print "G_sec:",G_sec
  print "strain_max",strain_max
  print "A_loop",A_loop
  print "G_Gmax", thisG_Gmax
  print "damping_ratio", damping_ratio
  print "-------------------------"

# For fig1 strain-stress plot
# reduce spacing between subplots to minimize the overlaps.
# plt.tight_layout()

# -----------------------------------
# plot the G_Gmax results
fig2=plt.figure()
plt.semilogx(gamma,G_Gmax,label = "DPAF")
# plot Seed results for comparison
Seed_strain_percent = [0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1.0];
Seed_G = [1, 0.99, 0.96, 0.9, 0.76, 0.57, 0.3, 0.15, 0.06];
```

```
Seed_strain=[x/100.0 for x in Seed_strain_percent];
plt.semilogx(Seed_strain, Seed_G, label="Seed")
plt.legend()
# plt.xlim((0,0.011))
plt.autoscale(enable=True,axis='x',tight=True) #
plt.ylim((0,1.1))
plt.xlabel('$\gamma_c$',fontsize=25)
plt.ylabel('G/Gmax',fontsize=25)
plt.grid(True)
# -----------------------------------
# plot the damping ratio results
fig3=plt.figure()
plt.semilogx(gamma,xi,label = "DPAF")
sameStrain=Seed_strain
Seed_damping=[0.0, 0.0, 0.01, 0.02, 0.033, 0.05, 0.1, 0.155, 0.21]
plt.semilogx(sameStrain,Seed_damping,label="Seed")
plt.legend()

# plt.ylim((0,0.4))

plt.xlabel(r'$\gamma_c$',fontsize=25)
plt.ylabel(r'$\xi$',fontsize=25)
plt.grid(True)
# -----------------------------------
# disp all the results on screen
plt.show()
```

21.4.3 The function to plot the G/Gmax with one loading only

```python
#!/usr/bin/python

import numpy as np
import matplotlib.pyplot as plt

# Open and read files
# open one file only
# 51 is the final loading file only.
i=51

filename_strain ="shearStrain"+str(i)+".txt"
filename_stress ="shearStress"+str(i)+".txt"
with open(filename_strain) as fStrain:
  shearStrain = fStrain.read()

with open(filename_stress) as fStress:
  shearStress = fStress.read()

# split the string to strains.
strain_spl=shearStrain.split('\n')
stress_spl=shearStress.split('\n')
# remove the last blankspace.
strain_spl.pop()
stress_spl.pop()
# convert string to int
strain=map(float, strain_spl)
stress=map(float, stress_spl)

index=0
while True:
  index=index+1
  # find the second zero.
  if strain[index] > strain[index+1]:
    break;

strain_1load=strain[:index]
stress_1load=stress[:index]


stress_2nd=stress[1]
strain_2nd=strain[1]
G_max=stress_2nd/strain_2nd

G_Gmax=[1.0]
gamma=[0.0]

for i in range(index-1):
  G_sec=stress_1load[i+1]/strain_1load[i+1]
  thisG_Gmax=G_sec/G_max
  G_Gmax.append(thisG_Gmax)
  gamma.append(strain_1load[i])
```

```
# ------------------------------------
# plot the G_Gmax results
fig2=plt.figure()
plt.semilogx(gamma,G_Gmax,label = "DPAF")
# plot Seed results for comparison
Seed_strain_percent = [0.0001, 0.0003, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1.0];
Seed_G = [1, 0.99, 0.96, 0.9, 0.76, 0.57, 0.3, 0.15, 0.06];
Seed_strain=[x/100.0 for x in Seed_strain_percent];
plt.semilogx(Seed_strain, Seed_G, label="Seed")
plt.legend()
# plt.xlim((0,0.011))
plt.autoscale(enable=True,axis='x',tight=True) #
plt.ylim((0,1.1))
plt.xlabel('$\gamma_c$',fontsize=25)
plt.ylabel('G/Gmax',fontsize=25)
plt.grid(True)
# ------------------------------------
plt.show()
```

References

[1] GR Cowper. The shear coefficient in timoshenko's beam theory. *Journal of applied mechanics*, 33(2):335–340, 1966.

[2] JD Renton. Generalized beam theory applied to shear stiffness. *International Journal of Solids and Structures*, 27(15):1955–1967, 1991.