Contents

Discrete model of circular plate

According to given initial conditions structure is circle width some amount of distributed load in center. Due to the symmetry of our structure, has to be studied only half of structure(fig.1).

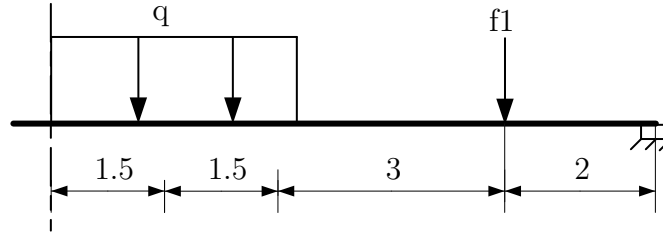Figure 1: Schematic representation for half of structure

Developed discrete model of given structure consist 4 finite elements(fig.2), local displacements and moments of each element shown on figure 3, while global ones are on figure 4.
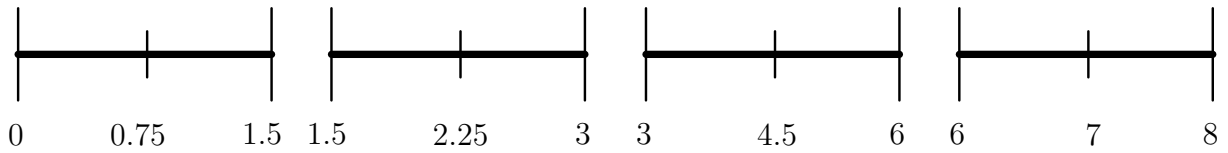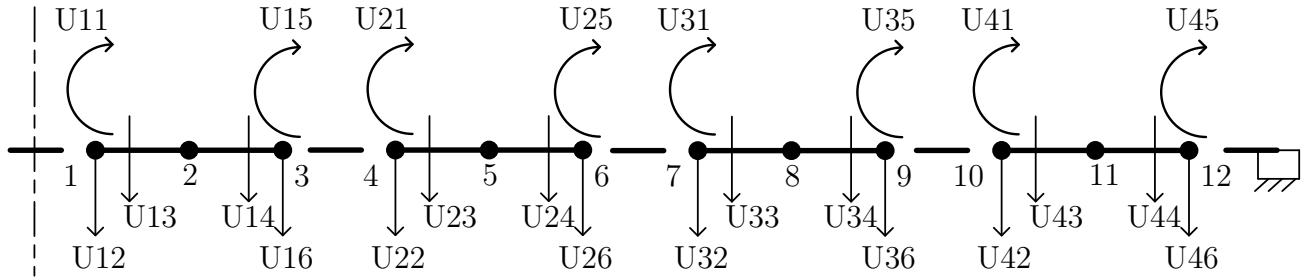

Figure 2: Coordinate scheme o finite elements

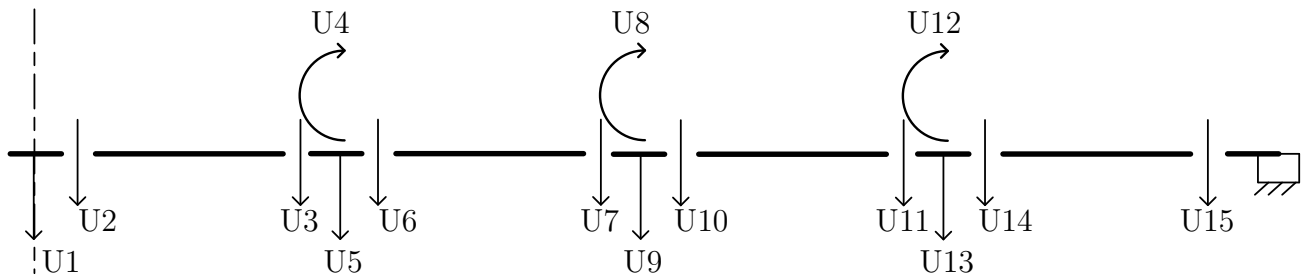

Figure 3: Local displacements and moments of each element

Figure 4: Global displacements and moments of each element

Compatibility matrix of displacements

It's based on the discrete model. It represents a relationship among the local and global displacements. We used 4 finite elements for each one has 6 local displacement and 5 nodes have total number of global displacements m=16. The final compatibility matrix of displacements are shown on figure 5, where rows are global displacements and columns are local displacements.

C	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 5: Compatibility matrix of displacements

Coefficient matrix of equilibrium equations

The matrix of the coefficients of the equilibrium equations for a ring element of a round plate is given in the table.

$$[\overline{A_k}] = 2\pi \begin{bmatrix}
\rho_{k,1} & 0 & 0 & 0 & 0 & 0 \\
1.5\dfrac{\rho_{k,1}}{b_k} - 1 & 1 & -2\dfrac{\rho_{k,1}}{b_k} & 0 & \dfrac{\rho_{k,1}}{2b_k} & 0 \\
-\dfrac{\rho_{k,2}}{b_k} + 2 & -\dfrac{5}{6} & 2\dfrac{\rho_{k,2}}{b_k} - 2 & \dfrac{2}{3} & -\dfrac{\rho_{k,2}}{b_k} & \dfrac{1}{6} \\
-\dfrac{\rho_{k,2}}{b_k} & -\dfrac{1}{6} & 2\dfrac{\rho_{k,2}}{b_k}2 & -\dfrac{2}{3} & -\dfrac{\rho_{k,2}}{b_k} - 2 & \dfrac{5}{6} \\
0 & 0 & 0 & 0 & -\rho_{k,3} & 0 \\
\dfrac{\rho_{k,3}}{2b_k} & 0 & -2\dfrac{\rho_{k,3}}{b_k} & 0 & 1 + 1.5\dfrac{\rho_{k,3}}{b_k} & -1
\end{bmatrix} \tag{1}$$

$A = [C]^T[\overline{A}]$ - equilibrium equation matrix, where:

$[C]^T$ - is transpose of compatibility matrix of displacements $[C]$

$[A]$ - is the matrix above in table

Calculation of the equilibrium equation matrix from Matlab

```
A =
 Columns 1 through 9
  -6.2832    6.2832         0         0         0         0         0         0         0
   6.2832   -5.2360         0    4.1888   -6.2832    1.0472         0         0         0
  -6.2832   -1.0472   25.1327   -4.1888  -18.8496    5.2360         0         0         0
        0         0         0         0   -9.4248         0    9.4248         0         0
   6.2832         0  -25.1327         0   25.1327   -6.2832   12.5664    6.2832  -25.1327
        0         0         0         0         0         0   -6.2832   -5.2360   25.1327
        0         0         0         0         0         0  -18.8496   -1.0472   50.2655
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0   12.5664         0  -50.2655
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
 Columns 10 through 18
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0         0         0         0         0         0         0         0         0
        0    6.2832         0         0         0         0         0         0         0
   4.1888  -18.8496    1.0472         0         0         0         0         0         0
  -4.1888  -31.4159    5.2360         0         0         0         0         0         0
        0  -18.8496         0   18.8496         0         0         0         0         0
        0   43.9823   -6.2832   12.5664    6.2832  -25.1327         0    6.2832         0
        0         0         0   -6.2832   -5.2360   25.1327    4.1888  -18.8496    1.0472
        0         0         0  -18.8496   -1.0472   50.2655   -4.1888  -31.4159    5.2360
        0         0         0         0         0         0         0  -37.6991         0
        0         0         0   12.5664         0  -50.2655         0   43.9823   -6.2832
        0         0         0         0         0         0         0         0         0
```

```
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0
Columns 19 through 24
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
   37.6991         0         0         0         0         0
   50.2655    6.2832  -75.3982         0   18.8496         0
  -31.4159   -5.2360   75.3982    4.1888  -43.9823    1.0472
  -43.9823   -1.0472  100.5310   -4.1888  -56.5487    5.2360
         0         0         0         0         0         0
```

Flexibility matrix

To reduce the size of the matrix and ease of placing it in the report, the equations were replaced by j1, j2, j3, j4.

$$[D_k] = \frac{2\pi b_k}{15K_k(1-\nu_k^2)} \begin{bmatrix} j_2 & -\nu_k j_2 & 2j_1 & -2\nu_k j_1 & -\rho_{k,2} & \nu_k \rho_{k,2} \\ & j_2 & -2\nu_k j_1 & 2j_1 & \nu_k \rho_{k,2} & -\rho_{k,2} \\ & & 16\rho_{k,2} & -16\nu_k \rho_{k,2} & 2j_3 & -2\nu_k j_3 \\ & & & 16\rho_{k,2} & -2\nu_k j_3 & 2j_3 \\ & & & & j_4 & -\nu_k j_4 \\ symm. & & & & & j_4 \end{bmatrix} \tag{2}$$

Where:

$$K_k = \frac{E_k t_k^3}{12(1-\nu_k^2)} \quad j_1 = \rho_{k,2} - b_k$$
$$j_2 = 4\rho_{k,2} - 3b_k$$
$$j_3 = \rho_{k,2} + b_k$$
$$j_4 = 4\rho_{k,2} + 3b_k$$

Calculation of the flexibility matrix from Matlab

```
D =
  Columns 1 through 9
     0.0001   -0.0000        0        0   -0.0001    0.0000        0        0        0
    -0.0000    0.0001        0        0    0.0000   -0.0001        0        0        0
          0         0    0.0017   -0.0005    0.0004   -0.0001        0        0        0
          0         0   -0.0005    0.0017   -0.0001    0.0004        0        0        0
    -0.0001    0.0000    0.0004   -0.0001    0.0008   -0.0002        0        0        0
     0.0000   -0.0001   -0.0001    0.0004   -0.0002    0.0008        0        0        0
          0         0         0        0        0        0    0.0010   -0.0003    0.0004
          0         0         0        0        0        0   -0.0003    0.0010   -0.0001
          0         0         0        0        0        0    0.0004   -0.0001    0.0052
          0         0         0        0        0        0   -0.0001    0.0004   -0.0016
          0         0         0        0        0        0   -0.0003    0.0001    0.0009
          0         0         0        0        0        0    0.0001   -0.0003   -0.0003
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
  Columns 10 through 18
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
          0         0         0        0        0        0        0        0        0
```

```
         0         0         0         0         0         0         0         0         0
   -0.0001   -0.0003    0.0001         0         0         0         0         0         0
    0.0004    0.0001   -0.0003         0         0         0         0         0         0
   -0.0016    0.0009   -0.0003         0         0         0         0         0         0
    0.0052   -0.0003    0.0009         0         0         0         0         0         0
   -0.0003    0.0016   -0.0005         0         0         0         0         0         0
    0.0009   -0.0005    0.0016         0         0         0         0         0         0
         0         0         0    0.0039   -0.0012    0.0017   -0.0005   -0.0013    0.0004
         0         0         0   -0.0012    0.0039   -0.0005    0.0017    0.0004   -0.0013
         0         0         0    0.0017   -0.0005    0.0207   -0.0062    0.0034   -0.0010
         0         0         0   -0.0005    0.0017   -0.0062    0.0207   -0.0010    0.0034
         0         0         0   -0.0013    0.0004    0.0034   -0.0010    0.0065   -0.0019
         0         0         0    0.0004   -0.0013   -0.0010    0.0034   -0.0019    0.0065
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0
         0         0         0         0         0         0         0         0         0

Columns 19 through 24

         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
    0.0048   -0.0014    0.0023   -0.0007   -0.0013    0.0004
   -0.0014    0.0048   -0.0007    0.0023    0.0004   -0.0013
    0.0023   -0.0007    0.0214   -0.0064    0.0031   -0.0009
   -0.0007    0.0023   -0.0064    0.0214   -0.0009    0.0031
   -0.0013    0.0004    0.0031   -0.0009    0.0059   -0.0018
    0.0004   -0.0013   -0.0009    0.0031   -0.0018    0.0059
```

External load vector

Vector of external forces for node described as:

$$\{F_k\} = \frac{2\pi b_k \rho_k}{3} \left\{ \begin{array}{c} 3\rho_{k,2} - b_k \\ 3\rho_{k,2} + b_k \end{array} \right\} = \{\eta_k\}\rho_k \tag{3}$$

$[F] = [F_o] + [C]^T[F_p]$, where:
$\quad [F_o] = f_1 2\pi R_{f1}$
$\quad f_1$ – value of external force which correspond u_1
$\quad R_{f_1}$ - coordinate where is m_1
$\quad [F_p]$ - represent the equivalent of distributed loads.
$\quad [C]^T$ - is transpose of compatibility matrix of displacements $[C]$

```
F =
         0
   47.1239
   94.2478
         0
         0
  188.4956
  235.6194
         0
         0
         0
         0
         0
  753.9822
         0
         0
         0
```

The results of global displacements are shown as following :

$$[U] = \{[A][D]^{-1}\}^{-1}[A]^T[F] \tag{4}$$

Where:
$\quad A$ - coefficient matrix of equilibrium equations
$\quad [A]^T$ - transpose coefficient matrix of equilibrium equations
$\quad [D]$: flexibility matrix
$\quad [F]$ - External load vector

```
Uglob =
    0.9698
    0.9776
    0.9305
   -0.0618
    0.9228
    0.9294
    0.7959
   -0.1130
    0.7897
    0.7993
    0.3667
   -0.1702
```

```
   0.3509
   0.3488
   0.0026
  -0.1743
```

The results of internal forces are shown as following:

$$[S] = [D]^{-1}[A]^T[U] \tag{5}$$

Where:

[D] - flexibility matrix

$[A]^T$- transpose coefficient matrix of equilibrium equations

[U] - global displacements

Results of internal forces (S) consists of M_ρ and moments M_φ. Schema have 24 values of internal forces (S) it's divided into: 12 values of moment M_ρ and 12 values of moments M_φ. In the matlab command $M_\rho = S(1:2:end)$ means to take the 1st value of (S) , 3rd , 5th , 7 th ..... till end. All those values belong to moment M_ρ. For $M_\varphi = S(2:2:end)$ means to take the 2nd value of (S), 4th , 6th.....till end.

```
M_Ro =
   132.3830
   130.0626
   123.1017
   123.1017
   111.5001
    95.2580
    95.2580
    67.9050
    49.2954
    49.2954
    22.2194
     0.0000


M_fi =
   132.3830
   131.0470
   127.0392
   127.0392
   120.3595
   111.0080
   121.8088
    88.9612
    82.3435
    82.2697
    59.6855
    51.6711
```

Internal forces and displacements

The equilibrium of finite element method used to solve the mentioned annular plate. The results were obtained from a MATLAB commands. Based on these calculations, the results shows the internal forces and the a distribution of global displacements along our structure. The results shows that the maximum displacement was $U = 969.8$ mm in the direction gravity. The allowable displacement was $U = L/250 = 64$ mm so, the verification was correct based on the current geometry and material properties. Some parametric analysis were done for plate thickness as a very important parameter to reduce the displacement values.
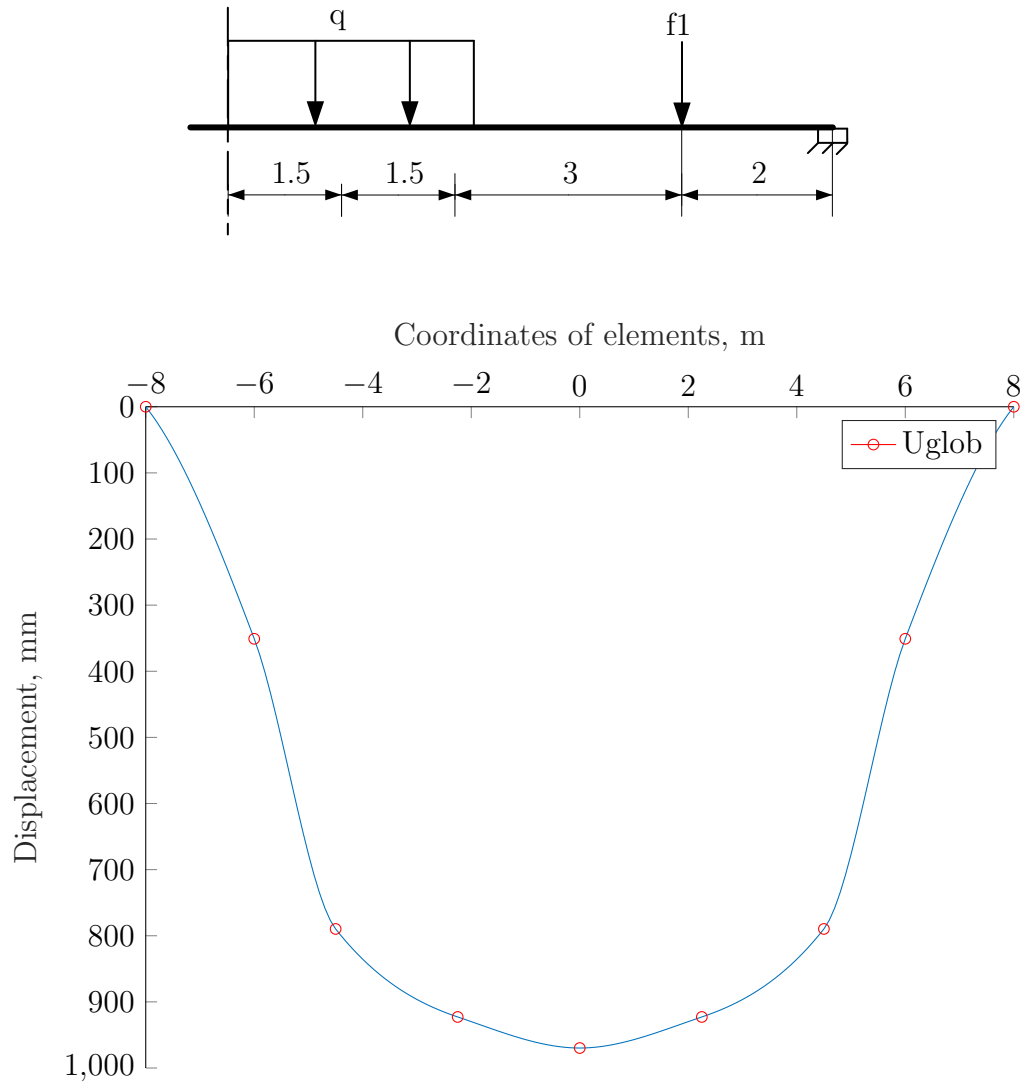
Figure 6: Global displacement across plate

```
um_mm =
   969.7797
   922.7799
   789.6782
   350.9039
         0
```
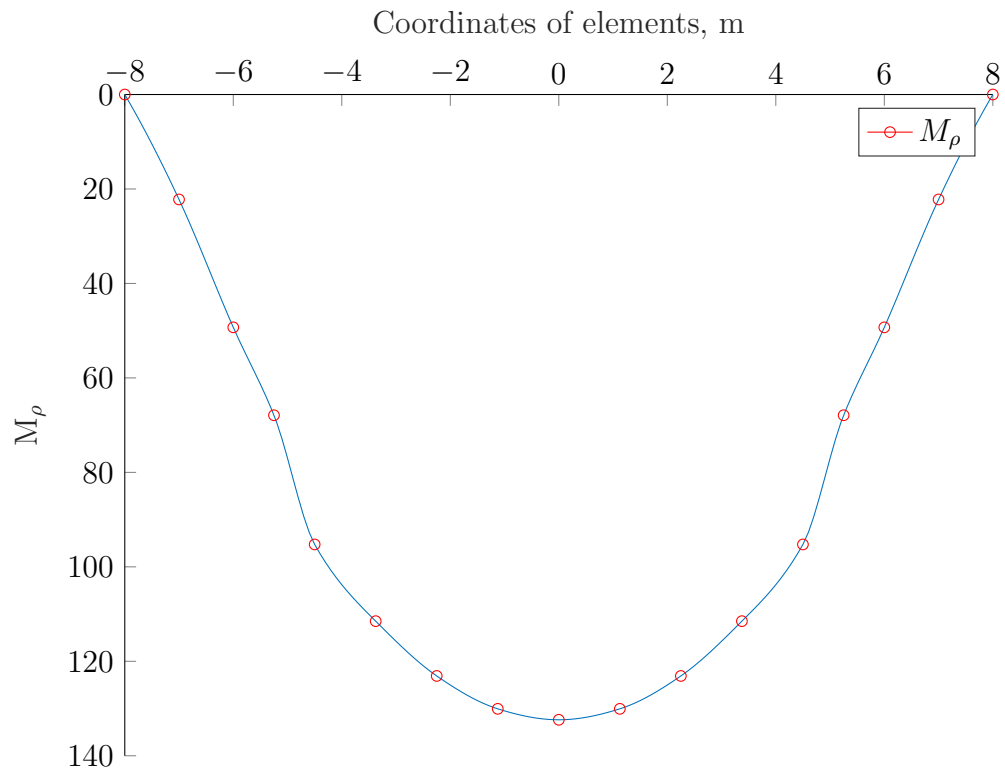
Figure 7: M_ρ across plate

```
M_Ro =
  132.3830
  130.0626
  123.1017
  123.1017
  111.5001
   95.2580
   95.2580
   67.9050
   49.2954
   49.2954
   22.2194
    0.0000
```
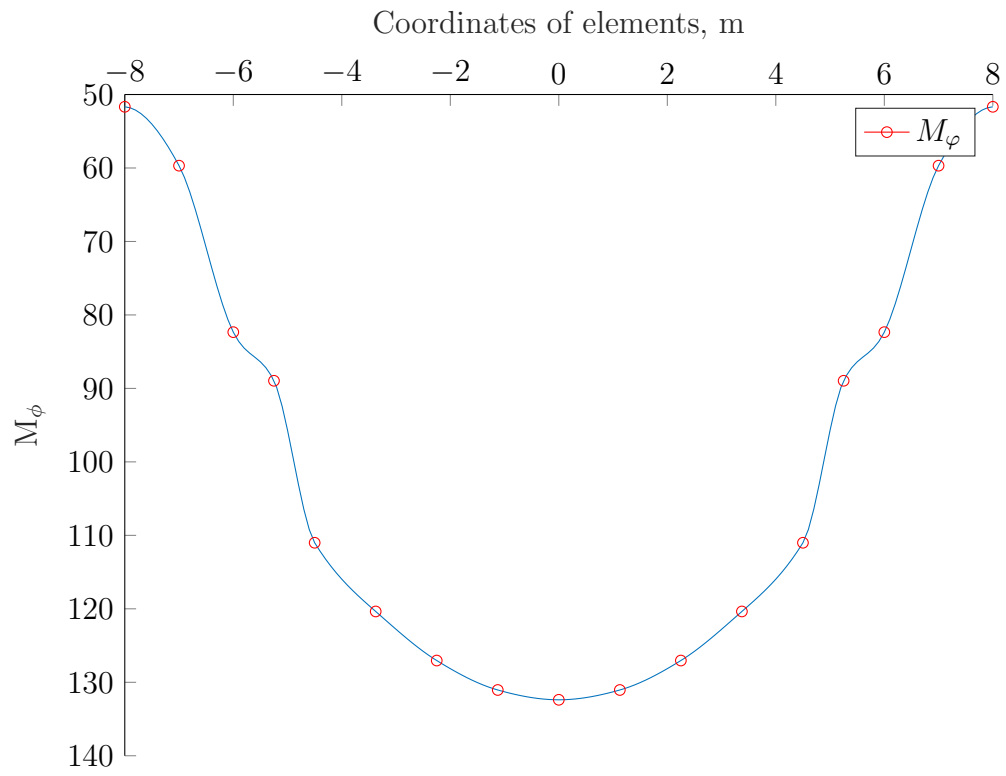
Figure 8: M_φ across plate

```
M_fi =
  132.3830
  131.0470
  127.0392
  127.0392
  120.3595
  111.0080
  121.8088
   88.9612
   82.3435
   82.2697
   59.6855
   51.6711
```

Code listing

Main function

```
 1 close all; clc; clear;
 2 format compact;
 3 addpath('../matlab2tikz/');
 4 % initial data
 5 q_LOAD =20; % KN/m
 6 f=20; % KN
 7 L=16; % span length m
 8 h=0.05; % thickness of plate
 9 E=210e6; % modulus of elastisty kPa
10 v=.3; % poisson's ratio
11 %
12 r=L/2; % radius of plate
13 B=[1.5 1.5 3 2];%vector of elements lengths, meters
14 no_FE=length(B); % Number of plate finite elements, as length of vector with
       elements lengths
15 b=B/2; % half of elements lengths
16 coords = zeros(no_FE,3);%filling matrix (number of elements by 3) by zeros
17 coords(1,2)=B(1)/2;% coordinate of half of first element
18 coords(1,3)=B(1);% coordinate of end of first element
19 for i=2:no_FE% loop over coordinates matrix, "i" is current element, "i-1" is
       previous element
20     coords(i,1)=coords(i - 1,3);%start coordinate as end coordinate of previous
           element
21     coords(i,2)=coords(i - 1,3) + B(i)/2;%half coordinate as end coordinate of
           previous element plus half length of current element
22     coords(i,3)= coords(i - 1,3) + B(i);%end coordinate as end coordinate of
           previous element plus length of current element
23 end
24 saveAsFileDairy('coords', coords);
25 no_of_local_dis =6; % Number of local displacements
26 no_of_global_dis =16; % Number of global displacements
27 % 2. Compatipality matrix C
28 % for the first FE
29 c_1stcompatipality_matrix=zeros(no_of_local_dis,no_of_global_dis);
30 c_1stcompatipality_matrix(2:6,1:5)=eye(5);
31 % for the Second FE
32 c_2ndcompatipality_matrix=zeros(no_of_local_dis,no_of_global_dis);
33 c_2ndcompatipality_matrix(1:6,4:9)=eye(6);
34 % for the third FE
35 c_3rdcompatipality_matrix=zeros(no_of_local_dis,no_of_global_dis);
36 c_3rdcompatipality_matrix(1:6,8:13)=eye(6);
37 % for the 4th FE
38 c_4thcompatipality_matrix=zeros(no_of_local_dis,no_of_global_dis);
39 c_4thcompatipality_matrix(1:5,12:16)=eye(5);
40 % the total compatipality matrix of displacements
41 C=[c_1stcompatipality_matrix;c_2ndcompatipality_matrix;c_3rdcompatipality_matrix;
       c_4thcompatipality_matrix];
42 %saveAsFileDairy('C', C);
43 %4. Matrix of equilibrium equantions A
44 for k=1:no_FE
45 A_matrix = getAmtx(coords(k,1), coords(k,2), coords(k,3), b(k));
46 A_(k*6-5:k*6,k*6-5:k*6)=2*pi*A_matrix;
47 end
48 A=C'*A_;
```

```matlab
49 saveAsFileDairy('A', A);
50 %% 5. Flexibility MATRIX OF D
51 for k=1:no_FE
52 Rok2=coords(k,2);
53 bk=b(k);
54 D_matrix = getDmtx(coords(k,2), b(k), v);
55 K.k=E*h^3/(12*(1-v^2));
56 D_(k*6-5:k*6,k*6-5:k*6)=(2*pi*bk/(15*K.k*(1-v^2)))*D_matrix;
57 end
58 saveAsFileDairy('D', D_);
59 %% 6. EXTERNAL LOAD VICTOR F
60 Fo=zeros(no_of_global_dis,1);
61 Rof=6;
62  Fo(13)=f*2*pi*Rof;
63 % Rof coordinate where f
64 % Fkp is nodal external load vector which is equivilant to distributed load of
        the kth element
65  q_Load_vector=[20 20 0 0];
66  for k=1:no_FE
67      bk=b(k);
68      q=q_Load_vector(k);
69      Rok2=coords(k,2);
70      Fk=(2*pi*bk/3)*q*[3*Rok2-bk;3*Rok2+bk];
71      Fp=[0;0;Fk;0;0];
72      Fp_(6*k-5:k*6,1)=Fp;
73  end
74 F=Fo+C'*Fp_;
75 saveAsFileDairy('F', F);
76
77 Uglob=inv(A*inv(D_)*A')*F;
78 saveAsFileDairy('Uglob', Uglob);
79
80 S=inv(D_)*A'*Uglob;
81 saveAsFileDairy('S', S);
82
83 M_Ro=S(1:2:end);
84 saveAsFileDairy('M_Ro', M_Ro);
85
86 M_fi=S(2:2:end);
87 saveAsFileDairy('M_fi', M_fi);
88
89 um_mm = 1000*[Uglob(1:4:end);0];
90 saveAsFileDairy('um_mm', um_mm);
91 % xcoord = [0;coords(1:end,3)];
92 % figure(1);
93 % plot(xcoord,um_mm,'DisplayName','Uglob');
94 % xlabel('Coordinates of elements, m')
95 % ylabel('Displacement, m')
96 % set(gca,'XAxisLocation','top','YAxisLocation','left','ydir','reverse');
97 % matlab2tikz('um_mm.tex','showInfo', false);
98 u_allowable=16/250*1000;
99 saveAsFile('u_allowable', u_allowable);
100 fullCoord = [-8; -6; -4.5; -2.25; 0; 2.25; 4.5; 6; 8];
101 plotU(fullCoord, um_mm, 1, true);
102 plotMfi(fullCoord, M_fi,3, true);
103 plotMro(fullCoord, M_Ro, 2, true);
```

getAmtx function

```
1 function A_matrix = getAmtx(Rok1, Rok2, Rok3, bk)
2      A_matrix = zeros(6,6);
3      A_matrix(1,1)=Rok1;
4      A_matrix(2,1)=1.5*Rok1/bk-1;
5      A_matrix(2,2)=1;
6      A_matrix(2,3)=-2*Rok1/bk;
7      A_matrix(2,5)=Rok1/(2*bk);
8      A_matrix(3,1)=-Rok2/bk+2;
9      A_matrix(3,2)=-5/6;
10     A_matrix(3,3)=2*Rok2/bk-2;
11     A_matrix(3,4)=2/3;
12     A_matrix(3,5)=-Rok2/bk;
13     A_matrix(3,6)=1/6;
14     A_matrix(4,1)=-Rok2/bk;
15     A_matrix(4,2)=-1/6;
16     A_matrix(4,3)=2*Rok2/bk+2;
17     A_matrix(4,4)=-2/3;
18     A_matrix(4,5)=-Rok2/bk-2;
19     A_matrix(4,6)=5/6;
20     A_matrix(5,5)=-Rok3;
21     A_matrix(6,1)=Rok3/(2*bk);
22     A_matrix(6,3)=-2*Rok3/bk;
23     A_matrix(6,5)=1+1.5*Rok3/bk;
24     A_matrix(6,6)=-1;
25 end
```

getDmtx function

```
1  function D_matrix = getDmtx(Rok2, bk, v)
2      j1=Rok2-bk;
3      j2=4*Rok2-3*bk;
4      j3=Rok2+bk;
5      j4=4*Rok2+3*bk;
6      D_matrix(1,1)=j2;
7      D_matrix(1,2)=-v*j2;
8      D_matrix(1,3)=2*j1;
9      D_matrix(1,4)=-2*v*j1;
10     D_matrix(1,5)=-Rok2;
11     D_matrix(1,6)=v*Rok2;
12     D_matrix(2,1)=D_matrix(1,2);
13     D_matrix(2,2)=j2;
14     D_matrix(2,3)=-2*v*j1;
15     D_matrix(2,4)=2*j1;
16     D_matrix(2,5)=v*Rok2;
17     D_matrix(2,6)=-Rok2;
18     D_matrix(3,1)=D_matrix(1,3);
19     D_matrix(3,2)=D_matrix(2,3);
20     D_matrix(3,3)=16*Rok2;
21     D_matrix(3,4)=-16*v*Rok2;
22     D_matrix(3,5)=2*j3;
23     D_matrix(3,6)=-2*v*j3;
24     D_matrix(4,1)=D_matrix(1,4);
25     D_matrix(4,2)=D_matrix(2,4);
26     D_matrix(4,3)=D_matrix(3,4);
27     D_matrix(4,4)=16*Rok2;
28     D_matrix(4,5)=-2*v*j3;
29     D_matrix(4,6)=2*j3;
30     D_matrix(5,1)=D_matrix(1,5);
31     D_matrix(5,2)=D_matrix(2,5);
32     D_matrix(5,3)=D_matrix(3,5);
33     D_matrix(5,4)=D_matrix(4,5);
34     D_matrix(5,5)=j4;
35     D_matrix(5,6)=-v*j4;
36     D_matrix(6,1)=D_matrix(1,6);
37     D_matrix(6,2)=D_matrix(2,6);
38     D_matrix(6,3)=D_matrix(3,6);
39     D_matrix(6,4)=D_matrix(4,6);
40     D_matrix(6,5)=D_matrix(5,6);
41     D_matrix(6,6)=j4;
42  end
```