FM 132

Foundations of Physical World

Spring 2023

MECHANICS PROJECT

TRUSS ANALYSIS

Team:

1. Utkarsh Agarwal - U20220091
2. Ayush Bhatnagar - U20220025
3. Arya Lamba - U20220023

PART 1: Trusses on computers

Description of the code

This code allows us to find the internal tension in each rod along with the support reactions on the truss when it is acted upon by an external force (only at joints).
For the input file, we need -

1. The geometry of the truss. We need to know the coordinates of each joint and which rods are connected to it.
2. The location of the hinge joint and the roller joint.
3. Where are the external forces acting on the truss - their Fx and Fy components.
4. The program consists of three functions - TrussPlotter, TrussAnalyser, Results Visualiser

Associated functions

- TrussPlotter: This function reads the input file that contains the geometry of the truss, its support conditions, and the external loads (allowed only at joints); and plots the truss to show the supports and the applied loads.

- TrussAnalyser: This function creates the matrix equation that contains all joint equilibrium equations and solves the equation to find all internal forces in the truss members as well as support reactions. The output from the function is the tension in all bars and support reactions.

- ResultsVisualizer: This function graphically displays the truss and clearly shows which bars are under tension and which bars are under compression, along with the support reactions in appropriate directions.

Logic used to form the coefficient matrix

First we created a matrix A whose no. of rows = no. of joints and no. of columns = no. of unknown forces (Internal tensions + reaction forces). Each row referred to a joint, and each column to the respective unknown force. This matrix signified which forces were acting on which joints.

Then for each joint, we iterated through the columns for matrix A. If the value of an element was 1, it meant that the tension in the respective rod was acting on the respective joint. Then we found the direction coefficients of every force acting on the joint - the cos component for Fx and the sin component for Fy. This way we formed the coefficient matrix B where each column referred to an unknown force and each joint had 2 rows associated with it - one for Fx and one for Fy.

Then we used the Gauss Jordan Elimination method to find out the unknown forces matrix.

- TrussPlotter

```
function truss_plotter(rods,joints,reactions,load)
axis off
axis equal
xlim auto
ylim auto
hold on
set(gcf,'color','w');
title("Truss Plotter","FontSize",15)
% Plotting the truss
for i=1:size(rods,1)
   x = [];
   y = [];
   x(1) = joints(rods(i,2),2);
   x(2) = joints(rods(i,3),2);
   y(1) = joints(rods(i,2),3);
   y(2) = joints(rods(i,3),3);
   plot(x,y,"Color","k","LineWidth",1.3)
end
% labelling
for i=1:size(joints,1)
   x = joints(i,2);
   y = joints(i,3);
   a = 0.1;
   if x<=y
       a = a*-1;
   end
   text(x+a,y,num2str(joints(i,1)))
end
% Plotting the support reactions
for i=1:size(reactions,1)
   joint = reactions(i,1);
   x = joints(joint,2);
   y = joints(joint,3);
   lines(x,y,i,reactions,"b","R"+i)
end
% Plotting external loads
for i=1:size(load,1)
   joint = load(i,1);
   x = joints(joint,2);
   y = joints(joint,3);
   lines(x,y,i,load,"r","F"+i)
end
hold off
figure
% Function to plot external lines
function lines(x,y,i,reactions,k,l)
   a = 0.2; b = 1;
   if reactions(i,2)>0
       if x==0
           a = -0.2; b = -1;
       end
```

```
        plot([x+a,x+b],[y,y],"Color",k,"LineWidth",1.5)
        text((x+b)-2*a,y-0.1,l+"x")
    end
    a = 0.2; b = 1;
    if reactions(i,3)>0
        if y==0
            a = -0.2; b = -1;
        end
        plot([x,x],[y+a,y+b],"Color",k,"LineWidth",1.5)
        text(x-0.3,(y+b)-2*a,l+"y")
    end
```

- TrussAnalyser

```
function d = truss_analyser(rods,joints,reactions,load)
% Finding out which forces are acting on which joints
A = zeros(size(joints,1));
for i=1:size(joints,1)
    for j=1:size(rods,1)
        if rods(j,2) == i || rods(j,3) == i
            A(i,rods(j,1)) = 1;
% Tensions in the rods
        end
    end
end
aa = size(A,2);
for i=1:size(reactions,1)
% Support reactions
    A(reactions(i,1),aa+i) = reactions(i,2);
    A(reactions(i,1),aa+2+i) = reactions(i,3);
end
% Forming co-efficient matrix
B = [];
for k=1:size(A,1)
    current = [joints(k,2),joints(k,3)];
% Storing value of current joint
    for i=1:size(A,2)-4
        if A(k,i)==1
            x1 = joints(rods(i,2),2);
            y1 = joints(rods(i,2),3);
            x2 = joints(rods(i,3),2);
            y2 = joints(rods(i,3),3);
            if x1==current(1) && y1==current(2)
% Storing values of the joints associated with rod
                second = [x2,y2];
            else
                second = [x1,y1];
            end
            dist = sqrt((x1-x2)^2 + (y1-y2)^2);
% Calculating distance between two points
            B(k*2-1,i) = (second(1)-current(1))/dist;
```

```
% Calculating direction components
            B(k*2,i) = (second(2)-current(2))/dist;
        end
    end
    for i=size(A,2)-3:size(A,2)-2
% Support reactions
        B(k*2-1,i) = A(k,i);
    end
    for i=size(A,2)-1:size(A,2)
        B(k*2,i) = A(k,i);
    end
end
disp(B)
% Forming result matrix
b = zeros(size(B,1),1);
for i=1:size(load,1)
    b(2*load(i,1)-1) = load(i,2);
    b(2*load(i,1)) = load(i,3);
end
% Calculating tensions using Gauss Jordan Elimination method
[~,jmax] = size(B);
aug = B;
aug(:,jmax+1) = b(:,1);
% Forming augmented matrix
f2 = rref(aug);
d = f2(:,jmax+1);
% Printing
fprintf("\n Rod No.\tTension\n\n")
unknowns = size(d,1)-3;
for i=1:unknowns
    if abs(d(i))<0.000001
        d(i)=0;
    end
    fprintf("  "+num2str(i)+"\t\t"+num2str(d(i))+"\n")
end
a = "x";
fprintf("\nSupport Reactons: \n\n")
for i=2:size(reactions,2)
    if i==3
        a = "y";
    end
    for j=1:size(reactions,1)
        if reactions(j,i)==1
            fprintf("R"+j+a+" = "+num2str(d(unknowns+1))+"\n")
            unknowns = unknowns+1;
        end
    end
end
```

- ResultsVisualiser

```
function truss_plotter(rods,joints,reactions,load)
axis off
axis equal
xlim auto
ylim auto
hold on
set(gcf,'color','w');
title("Truss Plotter","FontSize",15)
% Plotting the truss
for i=1:size(rods,1)
   x = [];
   y = [];
   x(1) = joints(rods(i,2),2);
   x(2) = joints(rods(i,3),2);
   y(1) = joints(rods(i,2),3);
   y(2) = joints(rods(i,3),3);
   plot(x,y,"Color","k","LineWidth",1.3)
end
% labelling
for i=1:size(joints,1)
   x = joints(i,2);
   y = joints(i,3);
   a = 0;b = 0.1;
   if x>y
       a = 0;
       b = b*-1;
   end
   text(x+a,y+b,num2str(joints(i,1)))
end
% Plotting the support reactions
for i=1:size(reactions,1)
   joint = reactions(i,1);
   x = joints(joint,2);
   y = joints(joint,3);
   lines(x,y,i,reactions,"b","R"+i)
end
% Plotting external loads
for i=1:size(load,1)
   joint = load(i,1);
   x = joints(joint,2);
   y = joints(joint,3);
   lines(x,y,i,load,"r","F"+i)
end
hold off
figure
% Function to plot external lines
function lines(x,y,i,reactions,k,l)
   a = 0.2; b = 1;
   if reactions(i,2)>0
       if x==0
```

```
            a = -0.2; b = -1;
        end
        plot([x+a,x+b],[y,y],"Color",k,"LineWidth",1.5)
        text((x+b)-2*a,y-0.1,l+"x")
    end
    a = 0.2; b = 1;
    if reactions(i,3)>0
        if y==0
            a = -0.2; b = -1;
        end
        plot([x,x],[y+a,y+b],"Color",k,"LineWidth",1.5)
        text(x-0.3,(y+b)-2*a,l+"y")
    end
```

Testing Code on Sample 6.4

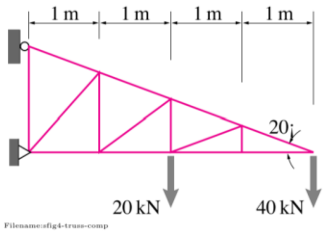


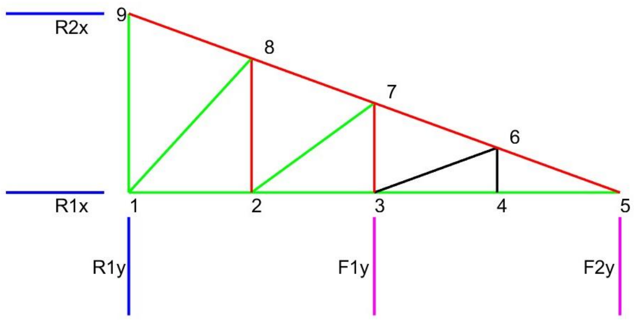
Figure 6.22:

SAMPLE 6.4 The truss shown in the figure has four horizontal bays, each of length 1 m. The top bars make 20° angle with the horizontal. The truss carries two loads of 40 kN and 20 kN as shown. Find the forces in each bar. In particular, find the bars that carry the maximum tensile and compressive forces.

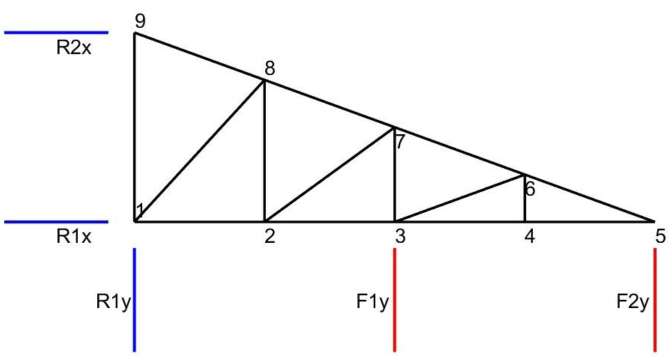
Input code

```
rods = [1 1 2; 2 2 3; 3 3 4; 4 4 5; 5 4 6; 6 3 6; 7 3 7; 8 2 7; 9 2 8; 10 1
8; 11 1 9; 12 8 9; 13 7 8; 14 6 7; 15 5 6];
joints = [1 0 0; 2 1 0; 3 2 0; 4 3 0; 5 4 0; 6 3 0.364; 7 2 0.364*2; 8 1
0.364*3; 9 0 0.364*4];
reactions = [1 1 1; 9 1 0];
load = [3 0 20;5 0 40];
truss_plotter(rods,joints,reactions,load);
sol = truss_analyser(rods,joints,reactions,load);
results_visualizer(rods,joints,sol,reactions,load)
```

Results Visualizer



Truss Plotter



Columns 1 through 11

1.0000	0	0	0	0	0	0	0	0	0.6754	0
0	0	0	0	0	0	0	0	0	0.7375	1.0000
-1.0000	1.0000	0	0	0	0	0	0.8085	0	0	0
0	0	0	0	0	0	0	0.5886	1.0000	0	0
0	-1.0000	1.0000	0	0	0.9397	0	0	0	0	0
0	0	0	0	0	0.3420	1.0000	0	0	0	0
0	0	-1.0000	1.0000	0	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0	0
0	0	0	-1.0000	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	-0.9397	0	0	0	0	0
0	0	0	0	-1.0000	-0.3420	0	0	0	0	0
0	0	0	0	0	0	0	-0.8085	0	0	0
0	0	0	0	0	0	-1.0000	-0.5886	0	0	0
0	0	0	0	0	0	0	0	0	-0.6754	0
0	0	0	0	0	0	0	0	-1.0000	-0.7375	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-1.0000

Columns 12 through 19

0	0	0	0	1.0000	0	0	0
0	0	0	0	0	0	1.0000	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	-0.9397	0	0	0	0
0	0	0	0.3420	0	0	0	0
0	0	-0.9397	0.9397	0	0	0	0
0	0	0.3420	-0.3420	0	0	0	0
0	-0.9397	0.9397	0	0	0	0	0
0	0.3420	-0.3420	0	0	0	0	0
-0.9397	0.9397	0	0	0	0	0	0
0.3420	-0.3420	0	0	0	0	0	0
0.9397	0	0	0	0	1.0000	0	0
-0.3420	0	0	0	0	0	0	0

Rod No.	Tension
1	-128.2051
2	-109.8901
3	-109.8901
4	-109.8901
5	0
6	0
7	20
8	-22.6543
9	13.3333
10	-13.5595
11	-50
12	146.1797
13	136.4344
14	116.9437
15	116.9437

Support Reactons:

R1x = 137.3626

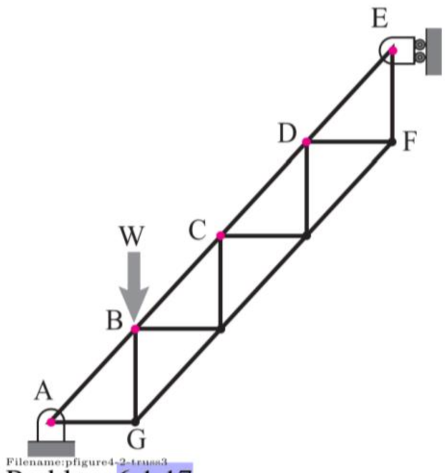
R2x = -137.3626

R1y = 60

Problem 6.1.17

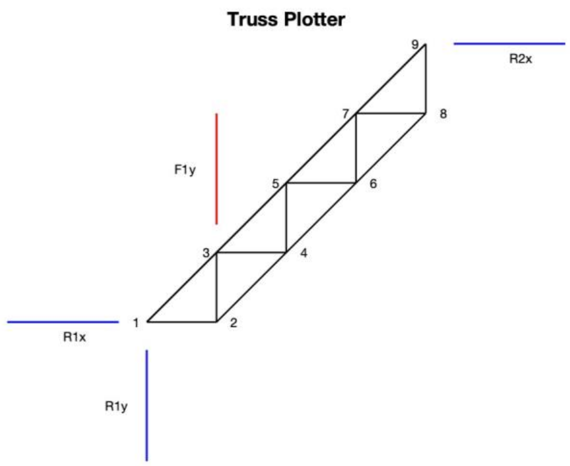
The stair step truss shown in the figure has 500mm long horizontal and vertical bars. Find the support reactions at A and E when a load W = 1kN is applied at

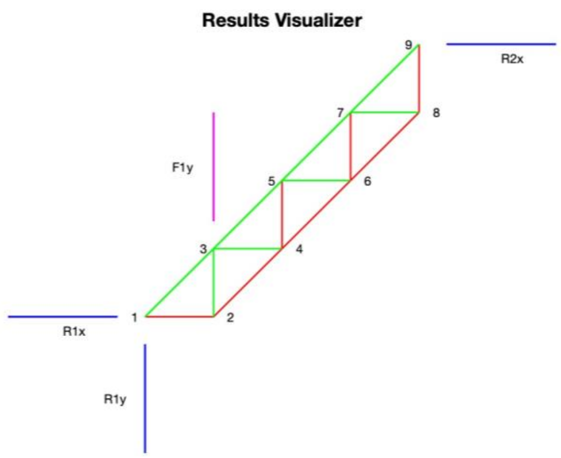
(a) point B (b) Point C (c) Point D

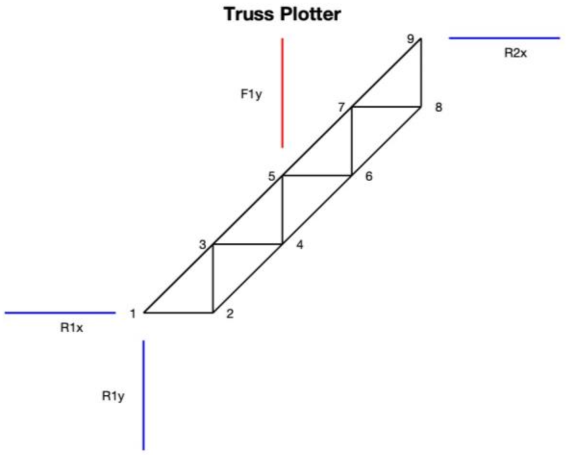


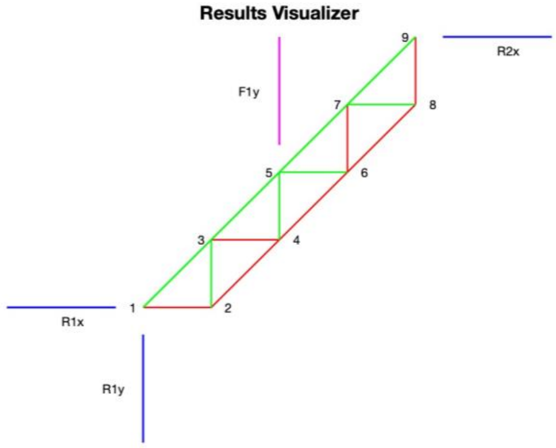
Problem 6.1.17

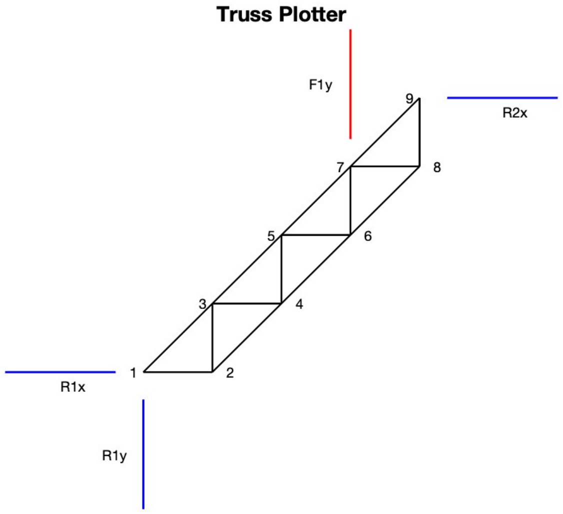
```
rods = [1 1 2; 2 2 4; 3 2 3; 4 1 3; 5 3 4; 6 4 6; 7 4 5; 8 3 5; 9 5 6; 10 6
8; 11 6 7; 12 5 7; 13 7 8; 14 8 9; 15 7 9];
joints = [1 0 0; 2 0.5 0; 3 0.5 0.5; 4 1 0.5; 5 1 1; 6 1.5 1; 7 1.5 1.5; 8
2 1.5; 9 2 2];
reactions = [1 1 1; 9 1 0];
load = [3 0 1];
truss_plotter(rods,joints,reactions,load);
sol = truss_analyser(rods,joints,reactions,load);
results_visualizer(rods,joints,sol,reactions,load)
```

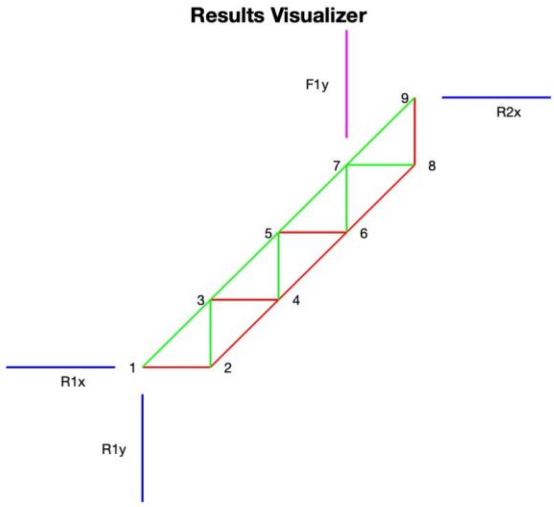












Columns 1 through 13

1.0000	0	0	0.7071	0	0	0	0	0	0	0	0	0
0	0	0	0.7071	0	0	0	0	0	0	0	0	0
-1.0000	0.7071	0	0	0	0	0	0	0	0	0	0	0
0	0.7071	1.0000	0	0	0	0	0	0	0	0	0	0
0	0	0	-0.7071	1.0000	0	0	0.7071	0	0	0	0	0
0	0	-1.0000	-0.7071	0	0	0	0.7071	0	0	0	0	0
0	-0.7071	0	0	-1.0000	0.7071	0	0	0	0	0	0	0
0	-0.7071	0	0	0	0.7071	1.0000	0	0	0	0	0	0
0	0	0	0	0	0	0	-0.7071	1.0000	0	0	0.7071	0
0	0	0	0	0	0	-1.0000	-0.7071	0	0	0	0.7071	0
0	0	0	0	0	-0.7071	0	0	-1.0000	0.7071	0	0	0
0	0	0	0	0	-0.7071	0	0	0	0.7071	1.0000	0	0
0	0	0	0	0	0	0	0	0	0	0	-0.7071	1.0000
0	0	0	0	0	0	0	0	0	0	-1.0000	-0.7071	0
0	0	0	0	0	0	0	0	0	-0.7071	0	0	-1.0000
0	0	0	0	0	0	0	0	0	-0.7071	0	0	0
0	0	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 14 through 19

         0         0    1.0000         0         0         0
         0         0         0         0    1.0000         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0    0.7071         0         0         0         0
         0    0.7071         0         0         0         0
         0         0         0         0         0         0
    1.0000         0         0         0         0         0
         0   -0.7071         0    1.0000         0         0
   -1.0000   -0.7071         0         0         0         0
```

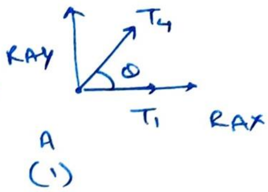
```
Rod No.     Tension

 1      0.75
 2      1.0607
 3      -0.75
 4      -1.4142
 5      -0.25
 6      0.70711
 7      0.25
 8      -1.0607
 9      -0.25
 10         0.35355
 11         0.25
 12         -0.70711
 13         -0.25
 14         0.25
 15         -0.35355
```

```
Support Reactons:

R1x = 0.25
R2x = -0.25
R1y = 1
```

Hand calculation

→ At Joint A,(1)

$$\Sigma x \Rightarrow T_1 + T_4 \cos\theta + R_{AX} = 0 \quad -(1)$$



$$\Sigma y \Rightarrow R_{AY} + T_4 \sin\theta = 0$$

$$1\,kN + T_4 \sin 45^\circ = 0$$

$$T_4 = \frac{-1}{\sin 45^\circ}$$

$$T_4 = \frac{-1}{1} \times \sqrt{2}$$

$$\boxed{T_4 = -\sqrt{2}}$$

$$\boxed{\begin{aligned} R_{AY} &= 1\,kN \\ R_{AX} &= -R_{EX} \end{aligned}}$$

$$\tan\theta = \frac{1}{1}$$

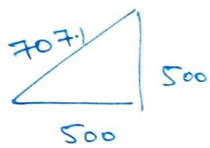
$$\theta = 45^\circ$$

→ At joint A equating the moment,

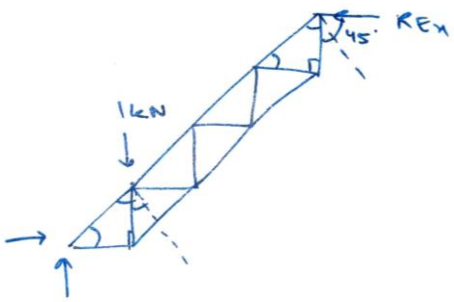
$$R_{EX} \cos 45^\circ \times 707.1 \times 4 = 1 \cos 45^\circ \times 707.1$$

$$R_{EX} \times \cancel{\frac{1}{\sqrt{2}}} \times \cancel{707.1} \times 4 = \cancel{\frac{1}{\sqrt{2}}} \times \cancel{707.1}$$

$$R_{EX} = \frac{1}{4} = 0.25$$

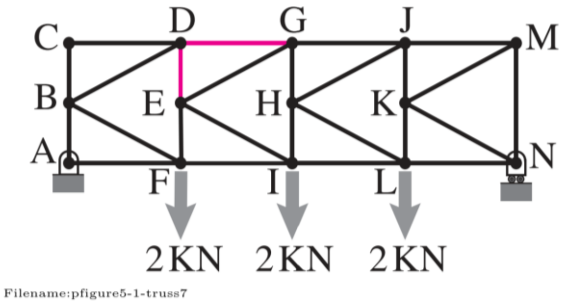


Using the method of joints and drawing the FBD, we can calculate the forces in the rods connected to joint A. These include the tensions in 4 and R_{EX}. **When we compare the results obtained from the method of joints to the results obtained from the MATLAB code (highlighted in yellow), we find that the tension in rod 4 and R_{EX} are the same.**.



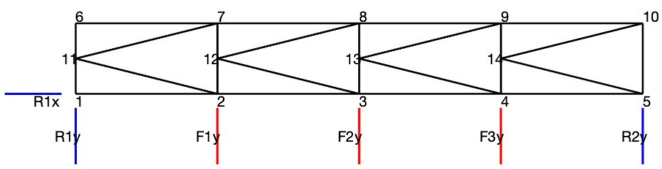
Problem 6.1.20

The truss shown in the figure consists of 4 square bays of a K structure. Each bay has two 2m long horizontal, two 1m long vertical bars, and two diagonal bars. Find the tensions in rods DE and DG.

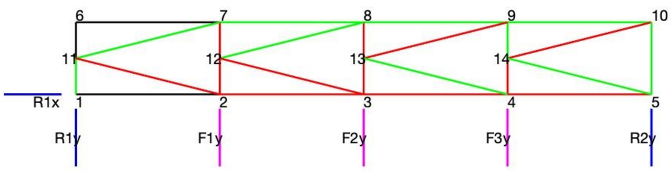



```
rods = [1 1 2; 2 2 3; 3 3 4; 4 4 5; 5 6 7; 6 7 8; 7 8 9; 8 9 10; 9 6 11; 10
1 11; 11 7 12; 12 2 12; 13 8 13; 14 3 13; 15 9 14; 16 4 14; 17 7 11; 18 2
11; 19 8 12; 20 3 12; 21 9 13; 22 4 13; 23 10 14; 24 5 14; 25 5 10];
joints = [1 0 0; 2 2 0; 3 4 0; 4 6 0; 5 8 0; 6 0 1; 7 2 1; 8 4 1; 9 6 1; 10
8 1; 11 0 0.5; 12 2 0.5; 13 4 0.5; 14 6 0.5];
reactions = [1 1 1; 5 0 1];
load = [2 0 2; 3 0 2; 4 0 2];
truss_plotter(rods,joints,reactions,load);
sol = truss_analyser(rods,joints,reactions,load);
results_visualizer(rods,joints,sol,reactions,load)
```

Truss Plotter



Results Visualizer



Columns 1 through 13

1.0000	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1.0000	0	0	0
-1.0000	1.0000	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1.0000	0
0	-1.0000	1.0000	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1.0000	1.0000	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-1.0000	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1.0000	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	-1.0000	0	0	0	0
0	0	0	0	-1.0000	1.0000	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-1.0000	0	0
0	0	0	0	0	-1.0000	1.0000	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	-1.0000
0	0	0	0	0	0	-1.0000	1.0000	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	-1.0000	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1.0000	-1.0000	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1.0000	-1.0000	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1.0000
0	0	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 14 through 26

0	0	0	0	0	0	0	0	0	0	0	0	1.0000
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	-0.9701	0	0	0	0	0	0	0	0
0	0	0	0	0.2425	0	0	0	0	0	0	0	0
0	0	0	0	0	0	-0.9701	0	0	0	0	0	0
1.0000	0	0	0	0	0	0.2425	0	0	0	0	0	0
0	0	0	0	0	0	0	0	-0.9701	0	0	0	0
0	0	1.0000	0	0	0	0	0	0.2425	0	0	0	0
0	0	0	0	0	0	0	0	0	0	-0.9701	0	0
0	0	0	0	0	0	0	0	0	0	0.2425	1.0000	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-0.9701	0	0	0	0	0	0	0	0	0
0	0	0	-0.2425	0	0	0	0	0	0	0	0	0
0	0	0	0	0	-0.9701	0	0	0	0	0	0	0
0	0	0	0	0	-0.2425	0	0	0	0	0	0	0
0	0	0	0	0	0	0	-0.9701	0	0	0	0	0
0	-1.0000	0	0	0	0	0	-0.2425	0	0	0	0	0
0	0	0	0	0	0	0	0	0	-0.9701	0	0	0
0	0	0	0	0	0	0	0	0	-0.2425	0	-1.0000	0
0	0	0	0.9701	0.9701	0	0	0	0	0	0	0	0
0	0	0	0.2425	-0.2425	0	0	0	0	0	0	0	0
0	0	0	0	0	0.9701	0.9701	0	0	0	0	0	0
0	0	0	0	0	0.2425	-0.2425	0	0	0	0	0	0
0	0	0	0	0	0	0	0.9701	0.9701	0	0	0	0
-1.0000	0	0	0	0	0	0	0.2425	-0.2425	0	0	0	0
0	0	0	0	0	0	0	0	0	0.9701	0.9701	0	0
0	1.0000	-1.0000	0	0	0	0	0	0	0.2425	-0.2425	0	0

Columns 27 through 29

0	0	0
0	1.0000	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	1.0000
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Rod No.	Tension
1	0
2	6
3	8
4	6
5	0
6	-6
7	-8
8	-6
9	0
10	-3
11	1.5
12	0.5
13	0.5
14	1.5
15	-0.5
16	2.5
17	-6.1847
18	6.1847
19	-2.0616
20	2.0616
21	2.0616
22	-2.0616
23	6.1847
24	-6.1847
25	-1.5

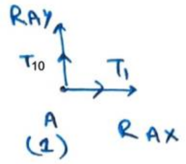
```
Support Reactons:

R1x = 0
R1y = 3
R2y = 3
```

→ Joint A(1)

$\Sigma F_y \rightarrow R_{AY} = 3\,kN$

$R_{NY} = 3\,kN$

$\Sigma F_x \rightarrow R_{AX} = 0$



At Joint A(1)

$R_{AX} + T_1 = 0 \Rightarrow T_1 = -R_{AX}$

$\boxed{T_1 = 0}$

$R_{AY} + T_2 = 0 \Rightarrow T_{10} = -R_{AY}$

$\boxed{T_{10} = -3\,kN}$

Using the method of joints and drawing the FBD, we can calculate the forces in the rods connected to joint A. These include the tensions in rod 1 and rod 10, which are both connected to joint A. **When we compare the results obtained from the method of joints to the results obtained from the MATLAB code (highlighted in yellow), we find that the tensions in rod 1 and rod 10 are the same,** which confirms the validity of the MATLAB code. This shows that the matrix method used in the MATLAB code is consistent with the principles of equilibrium used in the method of joints.