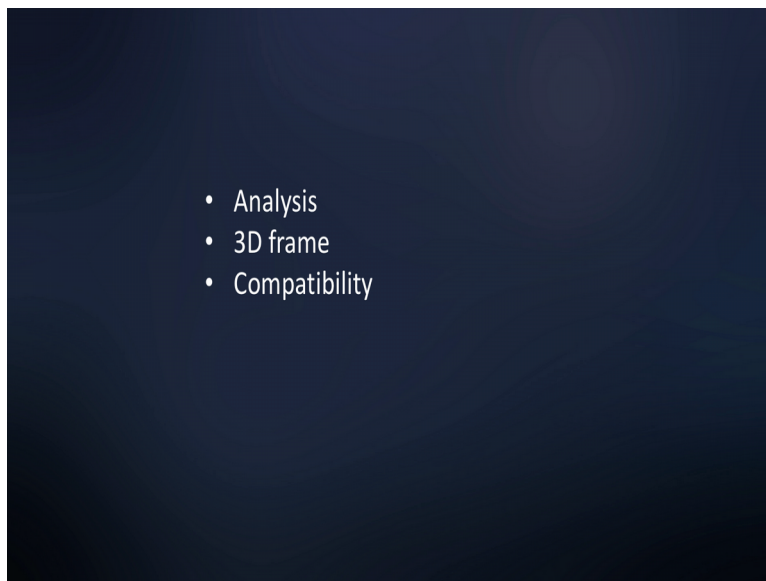


Computer Methods of Analysis of Offshore Structures
Prof. Srinivasan Chandrasekaran
Department of Ocean Engineering
Indian Institute of Technology, Madras

Module - 01
Lecture - 27
3d Analysis Example – 2 (Part – 2)

(Refer Slide Time: 00:16)



(Refer Slide Time: 00:22)

```
File Edit View Insert Actions Tools Help Mod-1 Lec-27- 3d Analysis Example - 2 (Part - 2) Prof. Srinivasan Chandrasekaran
3D analysis of space frame - Example 2
MATLAB program:
%% 3D analysis of space frame
clear
Note Title clear; 11/13/2017
%% Input
n = 5; % number of members
EI = [1 1 1 1]; %Flexural rigidity
EIz = EI;
EIy = EI;
EIx = EI;
GI = [0.25 0.25 0.25 0.25 0.25]; %EI; %Torsional constant
EA = [0.25 0.25 0.25 0.25 0.25]; %EI; %Axial rigidity
T = [4 4 4 4]; %Tangential twist
nJ = n-1; % Number of Joints
coordm = [0 6 0; 4 6 0; 7 6 0; 0 0 0; 4 0 0; 7 0 0]; %Coordinate wrt X,Y,Z; size(m),3
dc = [1 0 0; 1 0 0; 1 0 0; 1 0 0; 1 0 0; 1 0 0]; % Direction cosines for each member
tytr = [1 1 1 2]; % Type of Transformation for each member 1 = yz horizontal 2 = zy horizontal
psi = [0 0 0 90]; % Psi angle in degrees for each member
% C matrix
c1 = [1 0 0; 0 1 0; 0 0 1]; % C matrix for member 1
c2 = [1 0 0; 0 1 0; 0 0 1]; % C matrix for member 2
c3 = [1 0 0; 0 1 0; 0 0 1]; % C matrix for member 3
c4 = [1 0 0; 0 1 0; 0 0 1]; % C matrix for member 4
c5 = [0 1 0; 0 0 1; 1 0 0]; % C matrix for member 5
uu = 12; % Number of unrestrained degrees of freedom
ur = 24; % Number of restrained degrees of freedom
uul = [1 2 3 4 5 6 7 8 9 10 11 12]; % global labels of unrestrained dof
url = [13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36]; % global labels of restrained dof
l1 = [13 14 15 16 17 18 1 2 3 4 5 6]; % Global labels for member 1
l2 = [1 2 3 4 5 6 19 20 21 22 23 24]; % Global labels for member 2
l3 = [25 26 27 28 29 30 7 8 9 10 11 12]; % Global labels for member 3
l4 = [1 8 9 10 11 12 31 32 33 34 35 36]; % Global labels for member 4
l5 = [7 8 9 10 11 12 1 2 3 4 5 6]; % Global labels for member 5
l = [l1; l2; l3; l4; l5];
NPTEL dof = uu + ur; % Degrees of freedom
kTotal = zeros (dof);
```

You can see here, there are 5 members in the problem. Flexural rigidity is taken as 1, $E I_y$ and $E I_z$ as a same as $E I$. $G I$ as Torsional constant as 0.25 of $E I$, $E A$ is again 0.25 of $E I$, the length of the member you see here we already said that 4 3 4 3 6 we have the same thing here 4 3 4 3 6 ok.

(Refer Slide Time: 00:47)

Member	Li (m)	Ⓐ	Ⓑ	Direction Cosines			Transformation	Psi angle
				Cx	Cy	Cz		
AB	4	Ⓐ	Ⓑ	1	0	0	Y-Z-X	$\psi_1 = 0$
BC	3	Ⓑ	Ⓒ	1	0	0	Y-Z-X	$\psi_2 = 0$
DE	4	Ⓓ	Ⓔ	1	0	0	Y-Z-X	$\psi_3 = 0$
EF	3	Ⓔ	Ⓕ	1	0	0	Y-Z-X	$\psi_4 = 0$
EB	6	Ⓔ	Ⓑ	0	1	0	Z-Y-X	$\psi_5 = 90^\circ$

Number of joints will be; obviously, 6 there are 5 members, then we also enter the coordinates of this matrix, that is; 0 6 0 you can see here 0 6 0; 4 6 0 etcetera, we enter this here.

So, 0 6 0; then the direction cosine matrix for each member there are 5 members. So, let us enter this direction cosine matrix, we already have it here 1 0 0; 1 0 0 and so on. Let us enter that here 1 0 0; 1 0 0 and so on; you can see that here the direction cosines. Then we also ensure the type of transformation; one here represents Y Z X transformation, and two represents Z Y X transformation. So, all are one four members fifth member is 2, then the psi angle is entered, you see here the psi angle is when calculated - 0 0 0 0 and 90 degree.

So, we entered that here 0 0 0 0 90 degree; then the direction cosine matrix C; C matrices for all the members. You know is three by three matrixes. So, we know that we already computed the matrix here.

(Refer Slide Time: 02:09)

$C = C_{ij}$
 $\begin{matrix} x_1 & x_2 & x_3 \\ y & & \\ z & & \end{matrix}$

$C_{AB} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$C_{AB} = C_{BC} = C_{CE} = C_{EF}$

$C_{BC} = \begin{bmatrix} 0 & +1 & 0 \\ 0 & 0 & +1 \\ +1 & 0 & 0 \end{bmatrix}$

1 0 0; 0 1 0; 0 0 1; diagonal once so we entered that here 1 0 0; 0 1 0; for c 1, c 2, c 3, c 4 for c 5 it is different for c 5 it is 0 1 0; 0 0 1; 1 0 0; so 0 1 0; 0 0 1; 1 0 0.

(Refer Slide Time: 02:35)

Unrestrained dof $\{1 \dots 12\}$
 restrained dof $\{13 \dots 36\}$

There are 12 unrestrained degrees, and there are remaining 24 is restrained degree, then the labels are unrestrained degree 1 to 12, then 13 to 36, for each member we introduce the labels. I think this is the same procedure, what we have in the previous example; you can easily follow these labels, then we worked out the total stiffness matrix.

(Refer Slide Time: 03:02)

```
file edit view insert actions tools help Mod-1 Lec-27- 3d Analysis Example - 2 (Part - 2) Prof. Srinivasan Chandrasekaran
% Getting Type of transformation and Psi angle
for i = 1:n
    if lytr(i) == 1
        fprintf('Member Number %d\n', i);
        disp(i);
        fprintf('Type of transformation is Y-Z-X \n');
    else
        fprintf('Member Number %d\n', i);
        disp(i);
        fprintf('Type of transformation is Z-Y-X \n');
    end
    fprintf('Psi angle = %d\n', psi(i));
    disp(psi(i));
end

% Stiffness coefficients for each member
oc1 = EA/Lr;
oc2 = 6*EAz/(L^2);
oc3 = 6*EAy/(L^2);
oc4 = EI/L;
oc5 = 2*EAy/L;
oc6 = 12*EAz/(L^3);
oc7 = 12*EAy/(L^3);
oc8 = 2*EAz/Lr;
```

Then found the transformation matrix for each member.

(Refer Slide Time: 03:14)

```
file edit view insert actions tools help Mod-1 Lec-27- 3d Analysis Example - 2 (Part - 2) Prof. Srinivasan Chandrasekaran
Ktotal = Ktotal + Knew;
if i == 1
    T11 = Tz;
    K11 = Kz;
    fombar1 = T11*fem1;
elseif i == 2
    T12 = Tz;
    K12 = Kz;
    fombar2 = T12*fem2;
elseif i == 3
    T13 = Tz;
    K13 = Kz;
    fombar3 = T13*fem3;
elseif i == 4
    T14 = Tz;
    K14 = Kz;
    fombar4 = T14*fem4;
else
    T15 = Tz;
    K15 = Kz;
    fombar5 = T15*fem5;
end
end
fprintf('Stiffness Matrix of complete structure, [Ktotal] = \n');
disp(Ktotal);
Kurr = zeros(12);
for x=1:uu
    for y=1:uu
        Kurr(x,y) = Ktotal(x,y);
    end
end
```

Then entered the type of transformation then found out the stiffness coefficients.

So, is entered, then we found out del u. So, in this line we got unrestrained displacements. Of course, E I multiplied by is there in all the entire case. We found del u.

(Refer Slide Time: 04:23)

```
file (edit View Insert Actions Look Help) Mod1 Loc27-3d Analysis Example -2 (Part -2) Prof. Srinivasan Chandrasekaran
delbar1 = del1;
mbar1 = (Eg1 * delbar1)/fembar1;
fprintf('Member Number -');
disp (i);
fprintf('Global displacement matrix [DeltaBar] = \n');
disp (delbar1);
fprintf('Global end moment matrix [MBar] = \n');
disp (mbar1);
elseif i == 2
delbar2 = del1;
mbar2 = (Eg2 * delbar2)/fembar2;
fprintf('Member Number -');
disp (i);
fprintf('Global displacement matrix [DeltaBar] = \n');
disp (delbar2);
fprintf('Global end moment matrix [MBar] = \n');
disp (mbar2);
elseif i == 3
delbar3 = del1;
mbar3 = (Eg3 * delbar3)/fembar3;
fprintf('Member Number -');
disp (i);
fprintf('Global displacement matrix [DeltaBar] = \n');
disp (delbar3);
fprintf('Global end moment matrix [MBar] = \n');
disp (mbar3);
elseif i == 4
delbar4 = del1;
mbar4 = (Eg4 * delbar4)/fembar4;
fprintf('Member Number -');
disp (i);
fprintf('Global displacement matrix [DeltaBar] = \n');
disp (delbar4);
fprintf('Global end moment matrix [MBar] = \n');
disp (mbar4);
else
delbar5 = del1;
mbar5 = (Eg5 * delbar5)/fembar5;
fprintf('Member Number -');
disp (i);
fprintf('Global displacement matrix [DeltaBar] = \n');
disp (delbar5);
fprintf('Global end moment matrix [MBar] = \n');
disp (mbar5);
end
%% check
mbar = [mbar1'; mbar2'; mbar3'; mbar4'; mbar5'];
jE = zeros(dof,1);
for a=1:n
for b=1:12 % size of k matrix
d = 1(a,b);
jE(a,d) = mbar(a,b);
jE = jE/jEsum;
end
end
fprintf('Joint forces = \n');
disp (jE);
```

Then we found the member end forces by each member for member 1; member 2; member 3; member 4 and member 5. So, there is a line here; the program continues from here it goes here member 5, then we found the entire joint forces.

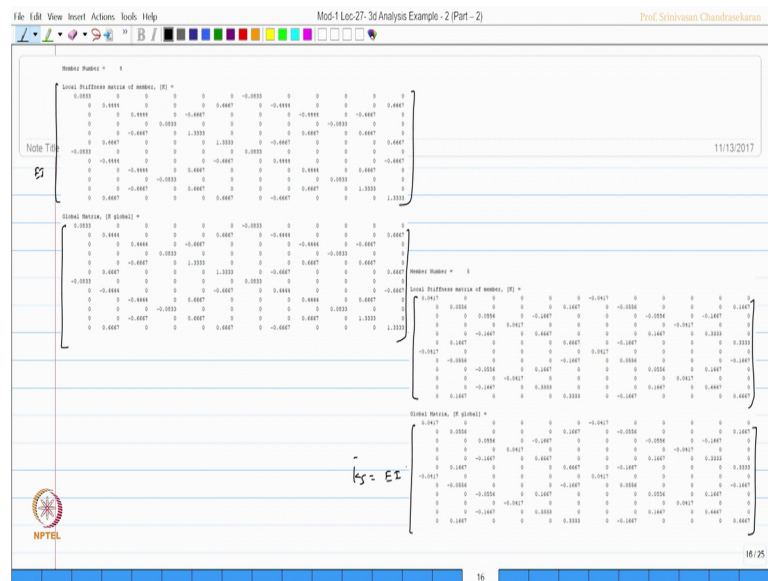
(Refer Slide Time: 04:45)

```
file (edit View Insert Actions Look Help) Mod1 Loc27-3d Analysis Example -2 (Part -2) Prof. Srinivasan Chandrasekaran
Member Number = 1 /
Type of transformation is 3-D-3
Psi angle = 0
Member Number = 2 /
Type of transformation is 3-D-3
Psi angle = 0
Member Number = 3 /
Type of transformation is 3-D-3
Psi angle = 0
Member Number = 4 /
Type of transformation is 3-D-3
Psi angle = 0
Member Number = 5 /
Type of transformation is 3-D-3
Psi angle = 90
Global Matrix, [K global] =
0.0425 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0.1875 0 0 0 0 0 0.3750 0 -0.1875 0 0 0 0 0 0 0 0 0 0
0 0 0.1875 0 -0.3750 0 0 0 0 0 -0.1875 0 0 0 0 -0.3750 0 0 0 0
0 0 0 0.0425 0 0 0 0 0 0 0 -0.0425 0 0 0 0 0 0 0 0
0 0 0.3750 0 0 0 0 1.0000 0 0 0 0 0.3750 0 0 0.5000 0 0
-0.0425 0 0 0 0 0 0 0 0.0425 0 0 0 0 0 0 0 0 0 0
0 -0.1875 0 0 0 0 0 -0.3750 0 0 0 0.1875 0 0 0 -0.3750 0
0 0 0 -0.1875 0 0 0 0 0 0 0 0 0.0425 0 0 0 0 0 0
0 0 0 -0.3750 0 0 0.5000 0 0 0 0 0 0.3750 0 0 1.0000 0 0
0 0.3750 0 0 0 0 0 0.5000 0 0 0 0 -0.3750 0 0 0 1.0000 0 0
```

So, the results are like this; type one: member 1, 2, 3, 4 and 5, we have entered we got the psi angles. We also got the type of transformation and now for each member we have got the local stiffness matrix; there is a multiply E I here.

Then the global stiffness matrix and the labels of course, entered as per the order what we already have for each member the member 1 for member 2 you can see here it is 12 by 12.

(Refer Slide Time: 05:10)



And this is K global, which will be actually K global is T transpose k local T. We obtain this is for member 3 and this is K global for member 3 there is an E I multiplier here K global for member 4 K local for member 5 and K global for member 5. So, now, we have got the global stiffness matrices of the 5 members.

(Refer Slide Time: 06:22)

The screenshot shows a software window titled "Mod-1 Lec-27- 3d Analysis Example - 2 (Part - 2)" with a user name "Prof. Srinivasan Chandrasekaran". The main area displays a large matrix with numerical values. To the right of the matrix, there are handwritten annotations: a box labeled "K" with "36x36" below it, and another box labeled "12x12" with "u_r" and "r_B" next to it. Below these, there are more handwritten labels: "u_r", "r_S", and "36x36". The NPTEL logo is visible in the bottom left corner.

There is an E I multiplier. So now, I have the total K global of the entire which is 36 by 36. Now we know very clearly that this K global will have unrestrained degrees of this unrestrained, this is restrained, unrestrained and restrained this is 12 by 12, this is 36 by 36 the remaining.

So, this can be now inverted. So, I got the invert of this.

(Refer Slide Time: 06:52)

The screenshot shows the same software window as the previous slide. The main area displays the "Inverse of Unrestrained Stiffness sub-matrix, (MuInverse) =". The matrix contains numerical values. On the right side, there is a list of "Discrete Load vectors, (DL) =". The NPTEL logo is visible in the bottom left corner.

(Refer Slide Time: 06:58)

The screenshot displays a software window titled "Mod-1 Lec-27-3d Analysis Example - 2 (Part - 2)" with the user name "Prof. Srinivasan Chandrasekaran". The main content area shows the following matrices and vectors:

- Unrestrained Stiffness sub-matrix, [Kuu] ***: A 10x10 matrix with values ranging from 0 to 3.0000. Handwritten annotations include "E I" on the left and "Joint Load vector, [FJ]" on the right.
- Inverse of Unrestrained Stiffness sub-matrix, [KuuInverse] ***: A 10x10 matrix with values ranging from 0 to 5.4104. Handwritten annotations include "E I" on the left and "Displacement, [DU]" on the right.
- Joint Load vector, [FJ] ***: A 10x1 column vector with values: 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.
- Displacement, [DU] ***: A 10x1 column vector with values: 0, -16.9797, 241.8182, -8.4933, 0, 0, 0, 0, 0, 0.

Handwritten annotations include "E I" on the left side of the matrices and "δ = 1/EI" on the right side of the displacement vector.

Now, this is unrestrained stiffness matrix which has been taken out. There is an $E I$ multiplier. There is an inverse of this, where 1 by $E I$ will be there, then the joint load vector is displayed here which was an input for the problem, then we have the δu . So, δu will have 1 by $E I$ as a multiplier. Once we get δu , we use a standard equation. Find the global end moments of member 1, member 2, member 3, member 4 and then member 5 ok.

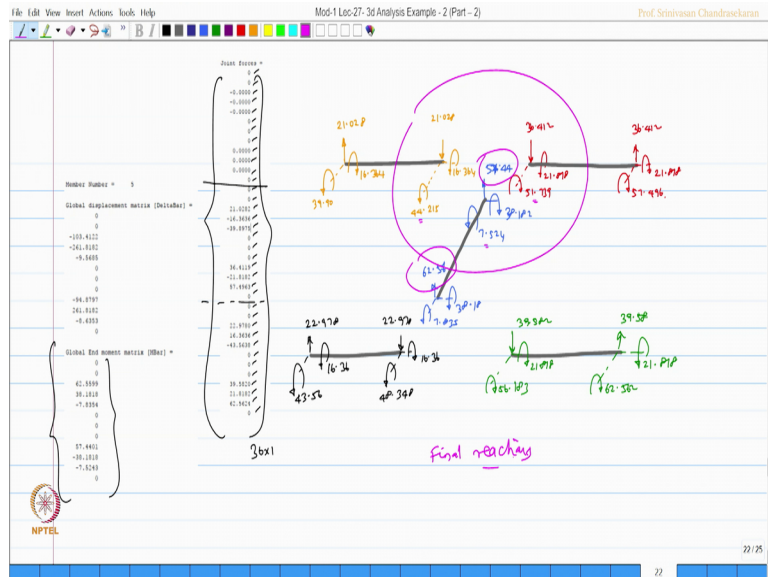
(Refer Slide Time: 07:31)

The screenshot displays a software window titled "Mod-1 Lec-27-3d Analysis Example - 2 (Part - 2)" with the user name "Prof. Srinivasan Chandrasekaran". The main content area shows the following matrices for four members:

- Member Number = 1**: Global displacement matrix [DU1] and Global End moment matrix [DM1].
- Member Number = 2**: Global displacement matrix [DU2] and Global End moment matrix [DM2].
- Member Number = 3**: Global displacement matrix [DU3] and Global End moment matrix [DM3].
- Member Number = 4**: Global displacement matrix [DU4] and Global End moment matrix [DM4].

Handwritten annotations include "E I" on the left side of the matrices and "δ = 1/EI" on the right side of the displacement matrices.

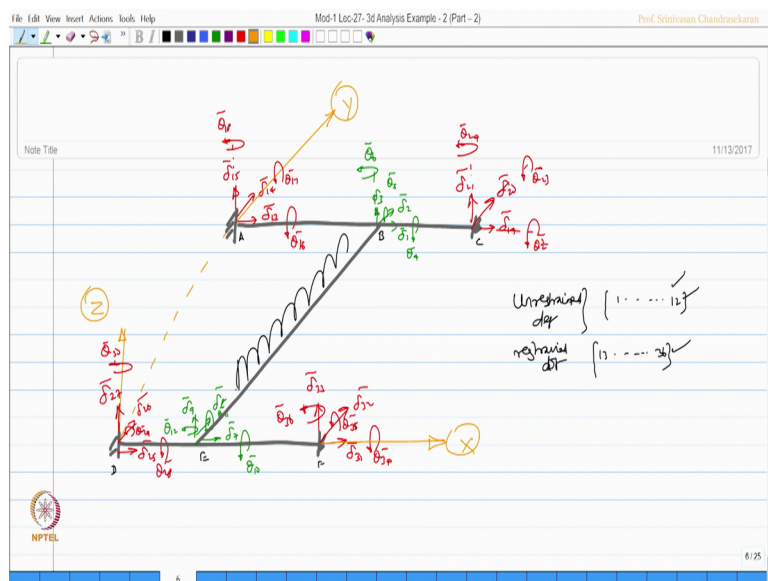
(Refer Slide Time: 07:39)



Once we get this we find the end joint forces of whole structural system from degrees of freedom starting from 1 to 36 ok.

You can see 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, and 36. So, we have a vector of 36. I want to plot this ensure the result. Let us do it for member wise; though this from this member separately and this member separately.

(Refer Slide Time: 08:44)



So, now degree of freedom once starts from here, you can see a degree of freedom starts from here. So, along X along Y and along Z, that is; how we have going to label this, we are going to interpret the results like that. So, along X along Y is 0; I am not marking the zero values, I am only marking the known values. So, let us say; all this member it is going to be 21.028 and this is going to be 16.364 and for this it is going to be 39.90 for this member it is going to be downward 21.028 and this is going to be 16.364 and this is going to be 44.215 for the member.

Next member for this it is going to be 36.412 and here it is 36.412 and this is going to be 21.818 and this is going to be 21.818 and this value is going to be 51.739 and this is 57.496. Let us do for this member this value is going to be 57.44, 38.18 and 7.524.

Similarly, at this joint it is 62.56 and 38.18 and 7.835, so for this member, 22.978, 22.978 and 16.16, 16.36, 43.56, 48.348. And for the last member 39.58, 39.58, 21.818, 21.818, 62.562, 56.183 that's a final end moments. So, now, interestingly the total reaction makes the total downward load and you can see the compatibility for example, take this joint 44 anticlockwise, 7 anticlockwise, that is; about y axis right which makes 51 clockwise and so on. So, one can see the compatibility this is the final end moments we are final reactions.

(Refer Slide Time: 13:44)

Summary

- 3d space frame - computer code
- input data - program
- results
 - understood - synchronization of the results
 - plotted checked for equilibrium compatibility

NPTEL

21

So, friends we have solved a 3d space frame problem using computer code. We are able to use the input data properly to run the program and obtain the results understood, the sign convention of the results and they are plotted and check for compatibility.

So, friends I would argue that you should try to do some more problems on your own. Derive the input data as required from the problem from the local axis alignments and try to solve uses computer program and see how you can solve them very easily and conveniently. So, friends you will realize now that, what all program we have given use in input data, the algorithm the pattern of analysis using stiffness method as never changed for 2d orthogonal frames, 2d non orthogonal, 2d truss members and non orthogonal truss members 3d we have the same algorithm continuing and therefore, there is a complete iterative scheme available. So, that the program can be easily done using MATLAB and you can solve the problems by hand as well as by computer coding.

I hope you enjoy this and you will try this program as a input data for variety of problems which are available in tutorial sheets, in the coming days and weekends for you to solve the problem. We are also giving with the solutions of the problem; I hope you will try to have a new facet of understanding 3d analysis using this lectures.

Thank you very much.