

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

Module - 02
Lecture - 15
Computer methods of dynamic analysis

(Refer Slide Time: 00:17)

- 
- Matlab code for influence coefficient method
 - Stodola's method and Rayleigh ritz method

The third program is influence coefficient method program we also did a problem of the same kind in the last lecture in influence coefficient method, but we derive the influence coefficient matrix directly in the last lecture.

(Refer Slide Time: 00:39)

Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2)
Prof. Srinivasan Chandrasekaran

INFLUENCE COEFFICIENT METHOD

15

MATLAB code:

```

% Program for finding frequency and mode shape using Influence Coefficient Method
% Enter dof
dof=3;
%Enter Mass matrix
M=[1 0 0 1 0 0 0 1]; % mass in kg
%Enter Stiffness Matrix
K=[2 -1 0; -1 2 -1; 0 -1 1]; % Stiffness in N/m
% Influence Coefficient Matrix
A=inv(K);
fprintf('Mass Matrix:\n');
disp(M);
fprintf('Stiffness Matrix:\n');
disp(K);
fprintf('Influence Coefficient Matrix:\n');
disp(A);
    
```

INPUT

M=[1 0 0 1 0 0 0 1]; % mass in kg

K=[2 -1 0; -1 2 -1; 0 -1 1]; % Stiffness in N/m

- ❖ The Inputs given for calculating the frequency and mode shape of MDOF system are
 - ❖ Degrees of Freedom
 - ❖ Mass Matrix
 - ❖ Stiffness Matrix
- ❖ Check the size of the matrices while giving the input (Here, it is a 3x3 matrix).
- ❖ The Mass values are given in 'kg' and stiffness values in 'N/m'.

So, the mass matrix and stiffness matrix are again given an input, so it is standard for all the programs. It inverts the mass matrix sorry the stiffness matrix and gets the influence coefficient matrix one can directly also get the influence coefficient matrix which we did in the last lecture. I think you can verify back the solution procedure in the last lecture. For the completion sake let us try to do that in this example.

(Refer Slide Time: 01:17)

File Edit View Insert Actions Tools Help
Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2)
Prof. Srinivasan Chandrasekaran

```

Eigen Solver Method
% Program for finding frequency and mode shape using EIGEN SOLVER METHOD
% Includes check for orthogonality and normalization
clear;
% Enter dof
dof=3;
%Enter Mass matrix
M=[1 0 0 1 0 0 0 1]; % mass in kg
%Enter Stiffness Matrix
K=[2 -1 0; -1 2 -1; 0 -1 1]; % Stiffness in N/m
% Influence Coefficient Matrix
A=inv(K);
fprintf('Mass Matrix:\n');
disp(M);
fprintf('Stiffness Matrix:\n');
disp(K);
fprintf('Influence Coefficient Matrix:\n');
disp(A);

% Eigen solver and eigen method
[omega, M_modes] = eig(M\K);
omega = sqrt(omega);
% Sort the eigenvalues
[omega, ind] = sort(omega);
% Sort the modes
M_modes = M_modes(ind,:);
% Normalize the modes
for i=1:dof
    M_modes(i,:) = M_modes(i,:)/norm(M_modes(i,:),2);
end
% Check for orthogonality
for i=1:dof
    for j=i+1:dof
        orthogonality = norm(M_modes(i,:)-M_modes(j,:),2);
        fprintf('Frequency: %f = %f Hz order %d, %d\n', omega(i), omega(i)*1000/(2*pi), i, j);
    end
end
% The modes are orthogonal to each other.
% The modes are not orthogonal to each other.

% Normalization and check
for i=1:dof
    M_modes(i,:) = M_modes(i,:)/norm(M_modes(i,:),2);
end
% Check for orthogonality
for i=1:dof
    for j=i+1:dof
        orthogonality = norm(M_modes(i,:)-M_modes(j,:),2);
        fprintf('Frequency: %f = %f Hz order %d, %d\n', omega(i), omega(i)*1000/(2*pi), i, j);
    end
end
% The modes are orthogonal to each other.
% The modes are not orthogonal to each other.

% Normalized Modal Matrix
M_modes = M_modes(ind,:);
% Mode shape plot
subplot(2,1,1);
hold on;
for i=1:dof
    plot(omega(i), M_modes(i,:), 'b');
end
axis([0 1 0 1]);
title('Normalized Modal Matrix');
% Mode shape plot
subplot(2,1,2);
hold on;
for i=1:dof
    plot(omega(i), M_modes(i,:), 'r');
end
axis([0 1 0 1]);
title('Mode shape plot');
    
```

Sample Output

```

Mass Matrix
1 0 0
0 1 0
0 0 1

Stiffness Matrix
2 -1 0
-1 2 -1
0 -1 1

Frequency: 0.445 rad/s
Frequency: 0.817 rad/s
Frequency: 1.102 rad/s

Mode Matrix
1.0000 1.0000 1.0000
0.8618 0.4450 -1.2470
-2.2470 -0.8618 0.8618

The modes are orthogonal to each other.
    
```

So, this is the coding for the classical Eigen solver which I just now showed you. So, the complete coding it continues from here, the coding continues after this it continues here.

I am sorry for the alignment of the code because it cannot come on the same page. So, please continue the code from here to here after this line this is the next line please continue the code. Sample input is given sample output is also known and you can see here the plot which has been using just now discussed.

(Refer Slide Time: 01:52)

The screenshot shows a presentation slide titled "2 Dunkerley's Method". The code defines a function for finding the fundamental frequency using Dunkerley's method. It includes parameters for mass, stiffness, and damping coefficients, and calculates the fundamental frequency. The sample output shows the mass matrix, stiffness matrix, and the resulting fundamental frequency.

```

2 Dunkerley's Method
% Program for finding frequency using Dunkerley's Method
clear;
close;
def=1;
% Dunkerley def
% Dunkerley Mass Matrix
M=[1 0 0 0 0 0 0 0;
  0 1 0 0 0 0 0 0;
  0 0 1 0 0 0 0 0;
  0 0 0 1 0 0 0 0;
  0 0 0 0 1 0 0 0;
  0 0 0 0 0 1 0 0;
  0 0 0 0 0 0 1 0;
  0 0 0 0 0 0 0 1];
% Dunkerley Stiffness Matrix
K=[10 0 0 0 0 0 0 0;
  0 10 0 0 0 0 0 0;
  0 0 10 0 0 0 0 0;
  0 0 0 10 0 0 0 0;
  0 0 0 0 10 0 0 0;
  0 0 0 0 0 10 0 0;
  0 0 0 0 0 0 10 0;
  0 0 0 0 0 0 0 10];
% Dunkerley Damping Matrix
D=[0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0];
% Dunkerley Fundamental Frequency (Dunkerley)
omega=1000;
% Dunkerley Fundamental Frequency (Dunkerley)
omega=1000;
end

Sample Output:
Mass Matrix
1 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 0 1 0 0
0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 1

Stiffness Matrix
10 0 0 0 0 0 0 0
0 10 0 0 0 0 0 0
0 0 10 0 0 0 0 0
0 0 0 10 0 0 0 0
0 0 0 0 10 0 0 0
0 0 0 0 0 10 0 0
0 0 0 0 0 0 10 0
0 0 0 0 0 0 0 10

Damping Coefficient Matrix
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

Fundamental Frequency (Dunkerley)
omega = 0.1000 rad/s
  
```

Similarly, the coding for Dunkerley is also shown on the screen now we have already explained different segments of this code and we also have the sample output which is been plotted from the screen.

(Refer Slide Time: 02:06)

The screenshot shows a presentation slide titled "3 Influence Coefficient Method". The code defines a function for finding the fundamental frequency using the influence coefficient method. It includes parameters for mass, stiffness, and damping coefficients, and calculates the fundamental frequency. The sample output shows the mass matrix, stiffness matrix, and the resulting fundamental frequency.

```

3 Influence Coefficient Method
% Program for finding frequency using Influence Coefficient Method
clear;
close;
def=1;
% Influence def
% Influence Mass Matrix
M=[1 0 0 0 0 0 0 0;
  0 1 0 0 0 0 0 0;
  0 0 1 0 0 0 0 0;
  0 0 0 1 0 0 0 0;
  0 0 0 0 1 0 0 0;
  0 0 0 0 0 1 0 0;
  0 0 0 0 0 0 1 0;
  0 0 0 0 0 0 0 1];
% Influence Stiffness Matrix
K=[10 0 0 0 0 0 0 0;
  0 10 0 0 0 0 0 0;
  0 0 10 0 0 0 0 0;
  0 0 0 10 0 0 0 0;
  0 0 0 0 10 0 0 0;
  0 0 0 0 0 10 0 0;
  0 0 0 0 0 0 10 0;
  0 0 0 0 0 0 0 10];
% Influence Damping Matrix
D=[0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0;
  0 0 0 0 0 0 0 0];
% Influence Fundamental Frequency (Influence)
omega=1000;
% Influence Fundamental Frequency (Influence)
omega=1000;
end

Sample Output:
Mass Matrix
1 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 0 1 0 0
0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 1

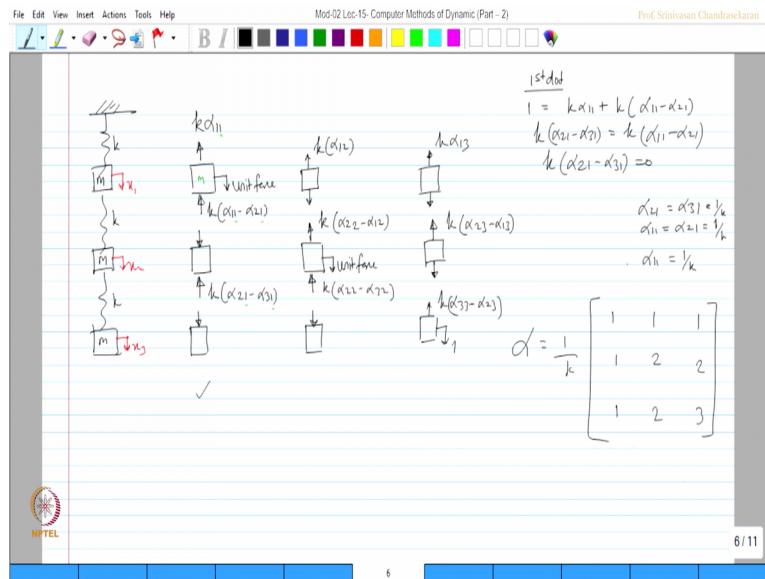
Stiffness Matrix
10 0 0 0 0 0 0 0
0 10 0 0 0 0 0 0
0 0 10 0 0 0 0 0
0 0 0 10 0 0 0 0
0 0 0 0 10 0 0 0
0 0 0 0 0 10 0 0
0 0 0 0 0 0 10 0
0 0 0 0 0 0 0 10

Damping Coefficient Matrix
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

Fundamental Frequency and mode shape
omega = 0.1000 rad/s
mode =
0.1000
0.1000
0.1000
0.1000
0.1000
0.1000
0.1000
0.1000
  
```

Now the influence coefficient method coding is available, but I am interested to estimate the influence coefficient matrix without inverting the stiffness matrix. This code actually inverts the stiffness matrix it inverts the stiffness matrix to find the influence coefficient matrix, but I can directly find the influence coefficient by the first principles which I am going to do now.

(Refer Slide Time: 02:36)



So, this is the problem let us write down the problem this is m and these are all k. The degrees of freedom are this is x_1 , this is x_2 and this is x_3 . So, influence coefficient is nothing, but the flexibility. So, flexibility is give unit force and find the responses. So, I say $k\alpha_{11}$ where α is the influence coefficients in the given matrix and this is going to be $k\alpha_{11}$ minus 21 . Similarly, for the next the arrow is reversed and this spring is going to compress which will be $k\alpha_{21}$ minus 31 and again this arrow will be reversed. So, one can notice very well here.

The second subscript in all will be one indicating we are given unit force in the first degree and we are getting the first column of the flexibility matrix. Let us do for the second case to give unit force here so obviously, this spring will try to pull it back. So, k times of α_{22} minus 12 and there was reversed and this is going to be k times of α_{12} and this spring will try to compress therefore this is going to be k times of α_{22} minus 32 and this spring is reversed.

Similarly, for the third degree unit force, so this is opposing k of α_{33} minus $2\alpha_{23}$ which will be reversed with this spring this will try to pull the mass back which will be k times of α_{23} minus α_{13} which will be reverse here and this will try to pull this back again which is k times of α_{13} . So, once this is done we can write the equations and try to solve and get the influence coefficient matrix for example, let us do for the first two degree. So, I am referring to this figure. So, I am writing it as 1 is equal to $k\alpha_{11}$ plus $k\alpha_{11}$ minus $2\alpha_{21}$ I can also write $k\alpha_{21}$ $k\alpha_{21}$ minus $3\alpha_{31}$ is equal to $k\alpha_{11}$ minus $2\alpha_{21}$, $k\alpha_{21}$ minus $3\alpha_{31}$ is 0 , so this implies that k cannot be 0 therefore, α_{21} minus $3\alpha_{31}$ is 0 which implies α_{21} , α_{21} is actually equal to $3\alpha_{31}$.

Substituting this back in the second equation the left hand side will become 0 which says that α_{11} is also equal to α_{21} , substituting that back in the first equation you will see that α_{11} becomes 1 by k which are now true for all the cases. So, α_{11} therefore, α_{21} is also 1 by k and α_{31} is also 1 by k . So, that is the first column.

One can similarly do for the second column and third column which we did in the last class last lecture please see that. So, the alpha matrix can be simply said as 1 by k of 1 1 1 that is what we got the first column 1 2 2 , and 1 2 and 3 . So, let us see what did we get from the computer program.

(Refer Slide Time: 08:17)

Mod-02 Lec-15: Computer Methods of Dynamic (Part - 2)
Prof. Srinivasan Chandrasekaran

SAMPLE OUTPUT

```

Mass Matrix
1   0   0
0   1   0
0   0   1

Stiffness Matrix
2  -1   0
-1  2  -1
0  -1   1

Influence Coefficient Matrix:
1.0000  1.0000  1.0000
1.0000  2.0000  2.0000
1.0000  2.0000  3.0000

Fundamental frequency and mode shape: wn = 0.445 rad/s
x =
1.0000
1.8019
2.2470
    
```

❖ MATLAB coding gives only the fundamental frequency and the mode shape.

17

So, one can see here for giving a mass matrix and stiffness matrix I get the influence coefficient matrix as 1 by k multiplier is anyway out. So, 1 by k is anyway here. So, first

column is as same as what we got second column and third column. So, the matrix exactly is same as we obtained from the program.

(Refer Slide Time: 08:46)

Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2) Prof. Srinivasan Chandrasekaran

INFLUENCE COEFFICIENT METHOD - contd

16

```

x=ones(dof,1);% assumed modal matrix
xo=x;
for u=1:10000
    y=A*M*xo;
    xn=y/y(1);
    d=xo-xn;
    ch=1e-6.*x;
    if abs(d)<ch
        x=xn;
        wn1=sqrt(1/y(1));
        fprintf('Fundamental frequency and mode shape: wn = %6.3f rad/s \n x = \n',wn1);
        disp(x);
        break
    else
        xo=xn;
    end
end
    
```

Assumption of first mode

Influence Coefficient Method

NPTEL

So, once we know this then this coding estimates the fundamental frequency and mode shape and the fundamental frequency now is 0.445 and the mode shape is first mode shape is this value which closely compares and agrees with the previous methods.

The next is a Stodola's method which we also solved a problem in the last lecture.

(Refer Slide Time: 09:09)

Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2) Prof. Srinivasan Chandrasekaran

STODOLA'S METHOD

19

MATLAB code:

```

% Program for finding frequency and mode shape using Stodola's Method
clc;
clear;
% Enter dof
dof=3;
% Enter mass values
m=[1 1 1]; % in kg
% Enter stiffness values
k=[1 1 1]; % in N/m
    
```

INPUT

m_1, k_1
 m_2, k_2
 m_3, k_3

$[m]$ $[k]$

- ❖ The Inputs given for calculating the frequency and mode shape of MDOF system are
 - ❖ Degrees of Freedom
 - ❖ Mass values
 - ❖ Stiffness values
- ❖ The Mass values are given in 'kg' and stiffness values in 'N/m'.

NPTEL

We maintain the same degree of freedom, mass value and stiffness values as entered in this case, but please note here the entire mass matrix and stiffness matrices need not be given, you want to give only the value of m_1 , m_2 and m_3 ; then k_1 , k_2 and k_3 . We do not have to give the mass matrix and k matrix.

Please note in the earlier case we are supposed to give the mass matrix completely and the k matrix completely row wise. But in this coding it does not require you to give the mass matrix and k matrix you are to only give the mass values and k values with a space bar in each row.

(Refer Slide Time: 10:15)

```

%% Stodola's Method
ad=ones(1,dof);% Assumed deflection
ado=ad;
for u=1:10000
    fi=m.*ado;% Inertia force
    fs=flip(cumsum(flip(fi)));%spring force
    sd=fs./k;% Cumulative deflection
    cd=cumsum(sd);% Cumulative deflection
    r=cd/cd(1);%ratio Mode Shape
    d=abs(ado-r);
    ch=1e-6.*ado;
end

if d<ch
    ad=r;
    wn2=sqrt(sum(r)/sum(cd));
    fprintf('Stodola method:\nFundamental
    frequency and mode shape: wn = %6.3f rad/s \n
    x = \n',wn2);
    disp(r);
    break
else
    ado=r;
end
end
    
```

This is the mass value this is a k value once you do this the code actually identifies and computes accumulated deflection finds the mode shape plus iteration and then from the iterated value it picks up ω_n^2 the fundamental frequency and then also picks up the mode shape. Let us see the sample output what we got from this code it is 0.445 and this is my vector. You can see this value closely agrees with what we have in influence coefficient method.

(Refer Slide Time: 10:34)

Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2) Prof. Srinivasan Chandrasekaran

SAMPLE OUTPUT

```
Stodola method:  
Fundamental frequency and mode shape: wn = 0.445 rad/s  
x =  
1.0000  
1.8019  
2.2470
```

❖ This method gives only the fundamental frequency and the mode shape.

NPTEL

We have 0.445 and the vector is 1, 1.8, 2.25 you can see here, this is again 0.445, 1, 1.8, 2.25. So, the values are exactly matching and surprisingly you will also notice we add exactly obtained the same values close to this by hand calculation in the last lecture. So, you can compare them. So, the computer program what we explained to you is exactly in the same line as the computer methods what we discussed by hand in the last lecture.

(Refer Slide Time: 11:24)

Mod-02 Lec-15 Computer Methods of Dynamic (Part - 2) Prof. Srinivasan Chandrasekaran

RAYLEIGH RITZ METHOD

MATLAB code:

```
% Program for finding frequency and mode shape using Rayleigh's Method  
clc;  
clear;  
% Enter dof  
doF=3;  
% Enter mass values  
m=[1 1 1]; % in kg  
% Enter stiffness values  
k=[1 1 1]; % in N/m
```

INPUT

M_1, k_1
 M_2, k_2
 M_3, k_3
 $M_1 = M_2 = M_3 = M$
 $k_1 = k_2 = k_3 = k$
 $(M), (k)$

❖ The Inputs given for calculating the frequency and mode shape of MDOF system are
❖ Degrees of Freedom
❖ Mass values
❖ Stiffness values
❖ The Mass values are given in 'kg' and stiffness values in 'N/m'.

NPTEL

The last one is the Rayleigh Ritz method which we also explained in the last lecture we solved a problem using Rayleigh method. Similarly please note friends in this method

also we have to give the value of m_1 , m_2 and m_3 similarly k_1 , k_2 and k_3 . In this problem all m s that is m_1 , m_2 and m_3 are actually equal to m , so they are taken as unity. Similarly k_1 , k_2 and k_3 in this problem are taken as k and we have entered 1 1 1. You do not have to give the mass matrix and k matrix as an input for this problem, you have to give only the values of independent stiffness and mass the row wise as you see the screen here.

(Refer Slide Time: 12:09)

```

Mod 02 Lec 15 Computer Methods of Dynamic (Part - 2) Prof. Srinivasan Chandrasekaran
RAYLEIGH RITZ METHOD - contd 24

%% Rayleigh's Method
ad=ones(1,dof);% Assumed Initial mode shape
ado=ad;
for u=1:10000
    fr=m.*ado;
    cd=zeros(1,dof);
    sfr=sum(fr);
    cd(1)=sfr/k(1);
    for i=2:dof
        sfr(i)=sum(fr(i:dof));
        cd(i)=(sfr(i)/k(i))+cd(i-1);
    end
    scd=cd(1);
    r=cd./scd;%mode shape Mode Shape
    d=abs(ado-r);
    ch=1e-6.*ado;

if d<ch
    ad=r;
    fro=fr.*r;
    sfr0=sum(fro);
    mro2=m.*r.*r;
    smro2=sum(mro2);
    wn2=sqrt(sfr0/(scd*smro2));
    fprintf('Rayleigh Ritz Method:\nFundamental frequency
    and mode shape: wn = %6.3f rad/s \n x = \n',wn2);
    disp(r);
    break
else
    ado=r;
end
end
end
  
```

Once you do this it assumes an initial mode shape here because we are looking for a fundamental frequency all are taken as positive.

Then it calculates the mode shape then it calculates the fundamental frequency let us see the output. So, you get 0.445 and this is the vector, so eigen vector of the first mode shape which is exactly same as we obtained by the previous methods and this value also matches with what we have worked out by hand in the last lecture. This is the coding for Stodola method, which is shown on the screen now there is a sample output and this is the coding for Rayleigh Ritz method which is shown on the screen now and that is the sample output.

(Refer Slide Time: 12:41)

The screenshot shows a presentation slide titled "4. Stodola's Method". The slide is divided into two columns: "Stodola's Method" and "Sample Output".

Stodola's Method:

```

% Program for finding frequency and mode shape using Stodola's Method
clear;
% Define mat
m=1;
% Define mass matrix
m=[1 0; 0 1];
% Define stiffness matrix
k=[10 0; 0 10];
% Define force vector
f=[1; 1];
% Define initial guess
phi=[1; 1];
% Define iteration count
n=10;
% Define tolerance
tol=1e-6;
% Define iteration counter
i=1;
% Define iteration loop
while i <= n
    % Calculate displacement
    u=inv(m)*f;
    % Calculate mode shape
    phi=phi+u;
    % Calculate frequency
    omega=1/sqrt(phi'*m*phi);
    % Calculate mode shape
    phi=phi/omega;
    % Check convergence
    if abs(omega(i)-omega(i-1)) < tol
        break;
    end
    i=i+1;
end
% Display results
disp('Natural frequency (rad/s) = ');
disp(omega);
disp('Mode shape = ');
disp(phi);

```

Sample Output:

```

Natural frequency (rad/s) =
    1.0000
    1.0000
Mode shape =
    1.0000
    1.0000

```

We have compared the results and they were all in good agreement.

(Refer Slide Time: 12:59)

The screenshot shows a presentation slide with handwritten text. The text is as follows:

Summary

- Computer methods for determining (ω_n, ϕ_n) of MDOF
 - Eigenvalue method
 - Dunkerley
 - Influence coeff method
 - Stodola
 - Rayleigh Ritz method
- Computer coding
 - sample output || good agreement with hand calculation - by calculator

→ $[M], [K], [C]$ (ω, ϕ) all d.o.f ✓
out come

So, let us make the summary. So, the summary says we learned computer methods of determining natural frequency and mode shape of multi degree freedom system models. In dynamic systems the methods learnt were classical Eigen solver method, Dunkerley method, Influence coefficient method, Stodola method and Rayleigh method. We picked our sample problems, we also gave you the computer coding, we showed you the sample

outputs and they were found to be in good agreement with hand calculations what we showed in last lectures.

I think friends now if I am able to obtain the mass matrix for offshore structural system and stiffness matrix for offshore structural system I should be able to obtain omega and phi for the dynamic system in all degrees of freedom. That is the outcome of this particular set of programs what we did for dynamic systems. In the next lecture we will start discussing about the damping matrix which is also one of the important issue as far as offshore structures are concerned.

Thank you very much.