RMIT University

OENG-1116: MODELLING & SIMUALTION OF ENGINEERING SYSTEMS

Week 3a

Assignment-1 businesses. Introduction into FEM. Plotting Surfaces with MATLAB.

Prof Pavel M. Trivailo

pavel.trivailo@rmit.edu.au

FEEDBACK:

QUESTIONS
HOME WORK
OBSERVATIONS
GENERAL DISCUSSIONS

MATLAB ONLINE

In order to access MATLAB from your computers you may wish to try this solution :

https://matlab.mathworks.com/

It should bring you to MATLAB ONLINE.

On my request, several students confirmed that this solution worked well for them and suited their needs.

If you have other effective solutions, which can be useful in the Course, please, let me know and I will share them with the class. Thank you!

USEFUL REFERENCE, available to students for download from RMIT Library:

Trivailo, P.M., Khayyam, H., Jazar, R. (2019). Illustrated Guidelines for Modelling and Dynamic Simulation of Linear and Non-Linear Deterministic Engineering Systems (Book Chapter 6). – In: Nonlinear Approaches in Engineering Applications 6, Edited by Liming Dai, Reza N. Jazar, Springer, Cham. – pp.171-272. DOI: https://doi.org/10.1007/978-3-030-18963-1_6.
Print ISBN: 978-3-030-18962-4; Online ISBN: 978-3-030-18963-1.

RMIT Library web REFERENCE TO THIS CHAPTER, which should work for students to access the whole Book (**with Chapter-6 only needed** for this Course):

https://primo-direct-apac.hosted.exlibrisgroup.com/permalink/f/1d27kpc/RMIT_ALMA51235014800001341

TUTORIAL-1:

Next Week, ON-LINE ONLY:

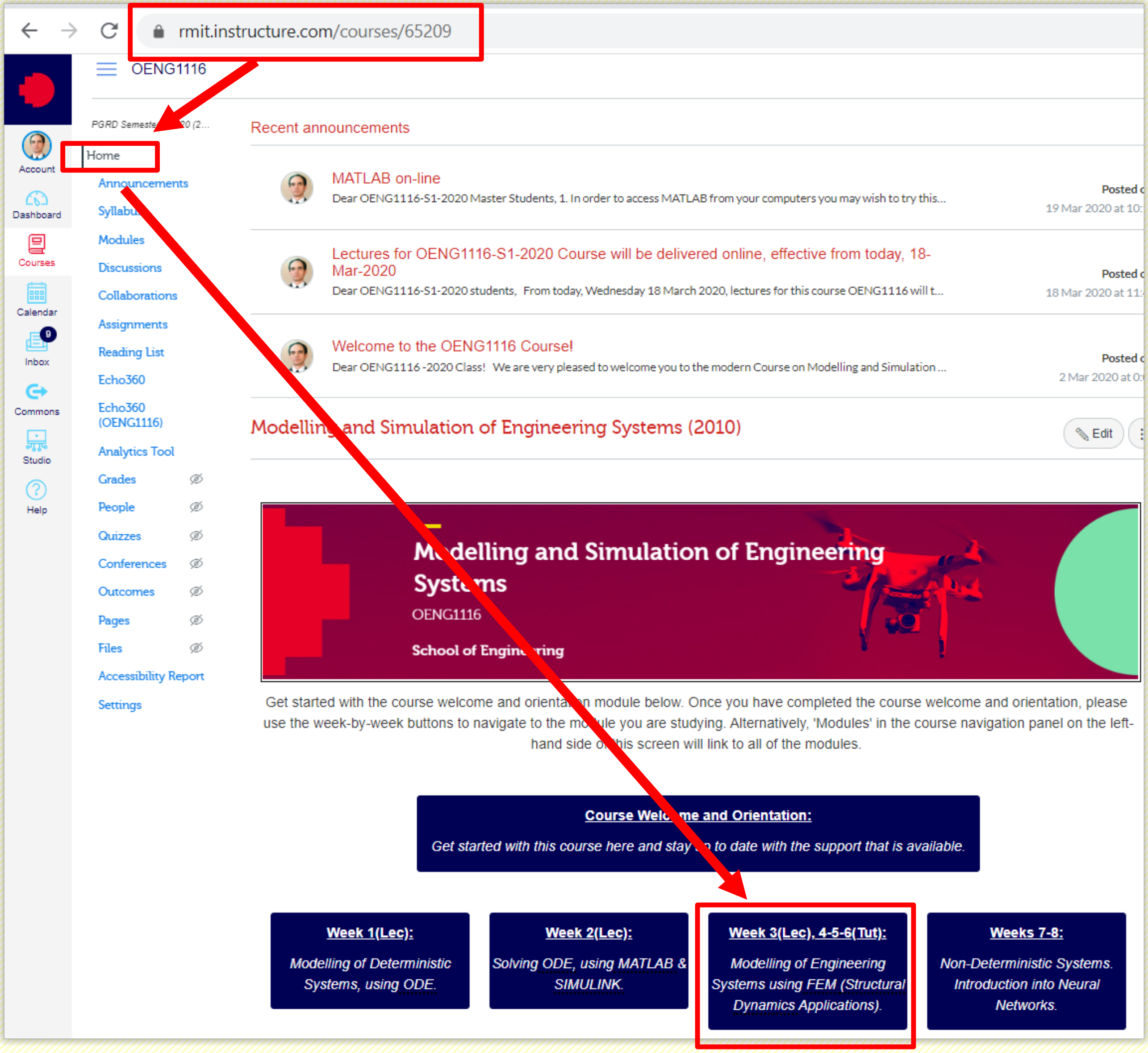
(1) Recordings on Canvas day before scheduled Tutorial;

(2) Tutors to be available via Skype during 19:30-21:20 on 26/03,

More details are in the announcement on Canvas next week

CANVAS Wk3 Notes + Tutorial-Wks4-5-6 Materials:

CANVAS HOME (TOP) PAGE



CANVAS Wk3 Notes + Tutorial-Wk6 Materials:

rmit.instructure.com/courses/65209/pages/weeks-5-6-modelling-of-engineering-systems-using-fem-structural-dynamics-applications

Account | Dashboard | Courses | Calendar | Inbox | Commons | Studio | Help

Quizzes | Conferences | Outcomes | Pages | Files | Accessibility Report | Settings

- **Trivailo, P.M.**, Khayyam, H., Jazar, R. (2019). Illustrated Guidelines for Modelling and Dynamic Simulation of Linear and Non-Linear Deterministic Engineering Systems (Book Chapter 6). – In: Nonlinear Approaches in Engineering Applications 6, Edited by Liming Dai, Reza N. Jazar, Springer, Cham. – pp.171-272. DOI: https://doi.org/10.1007/978-3-030-18963-1_6 . Print ISBN: 978-3-030-18962-4; Online ISBN: 978-3-030-18963-1.
- RMIT Library web REFERENCE TO THIS CHAPTER, which should work for students to access the whole Book (with Chapter-6 only needed for this Course): https://primo-direct-apac.hosted.exlibrisgroup.com/permalink/f/1d27kpc/RMIT_ALMA51235014800001341

Purpose

By completing the essential set text reading and reviewing the recorded lecture to ensure you have understood the key concepts, you will be able to begin work on Project, Part-1.

Lecture-Wk3b Notes by PMT:

https://drive.google.com/open?id=1wBfPtwPql4xZd7PBfSyT6qohtY7alLXw

Lecture-Wk3a Notes by PMT:

https://drive.google.com/open?id=1-pTR1zOEU9iEQbn2DkKfdVMy_-Or2GmG

Prof P.M. Trivailo's Notes on the FEM:

https://drive.google.com/file/d/1-6Sh6mctXu4kYy5Sp8oOBTo-hd2u1mdU/view?usp=sharing

Tutorial-3 (Wk.6) Slides

https://drive.google.com/open?id=1uD8bIHJwMUvYsXRw9CNxoGutDqKeb5zH

Tutorial-2 (Wk.6) MATLAB examples in a single file

https://drive.google.com/open?id=1ywhqfFDyejzKPdc5lpGwy_CtPolvMgrl

Note: each example is framed as a separate cell (section). You may wish to run example after example using the following: (a) open the MATLAB examples file in the MATLAB Editor; (b) select the interested section and press (simultaneously) Ctrl+Shift+ENTER. Pressing Ctrl+Shift+ENTER would execute the next example, etc.

If you are in the MATLAB Command Window and would like to return to the MATLAB Editor Window, press (simultaneously) Ctrl+Shift+ZERO.

Tutorial-1 Slides

https://drive.google.com/open?id=1rtkdvRh3RU7icoFdofyT3BBTRgy60Ogd

Tutorial-1 MATLAB examples in a single file

https://drive.google.com/open?id=1TW5OfXAShETS1fRAAUM_Sf7a_7nw-v9C

Note: each example is framed as a separate cell (section). You may wish to run example after example using the following: (a) open the MATLAB examples file in the MATLAB Editor; (b) select the interested section and press (simultaneously) Ctrl+Shift+ENTER. Pressing Ctrl+Shift+ENTER would execute the next example, etc.

If you are in the MATLAB Command Window and would like to return to the MATLAB Editor Window, press (simultaneously) Ctrl+Shift+ZERO.

ASSIGNMENT-1 Pt1

Released &
Sent via Individual Student Emails

Deadline for Project Proposal:

The **AMENDED** deadline for the submission of the Individual Assignment-1-Pt1 (OENG1116) is end of **Week-6 (Tuesday, 07 April, 2020) 23:59pm AEST.**

The aim of this submission IS:

for each Master to receive a feedback from the Lecturer by the end of Week-6 to bring you certainty, peace of mind and to multiply your confidence!

MAIN LECTURE TOPIC:

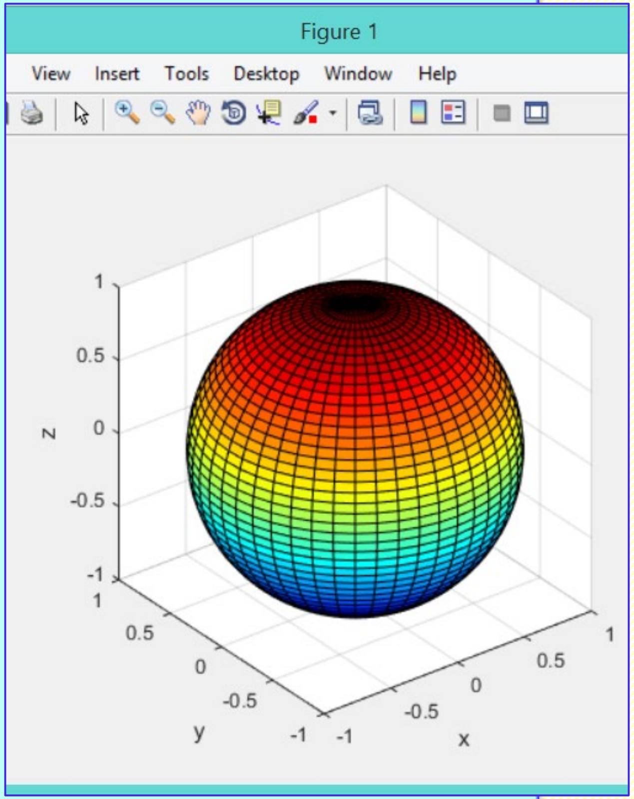
MATAB: PLOTTING SURFACES
FEM: RODS (AXIAL VIBRATIONS)

PLOTTING SURFACES WITH MATLAB:

Useful Examples

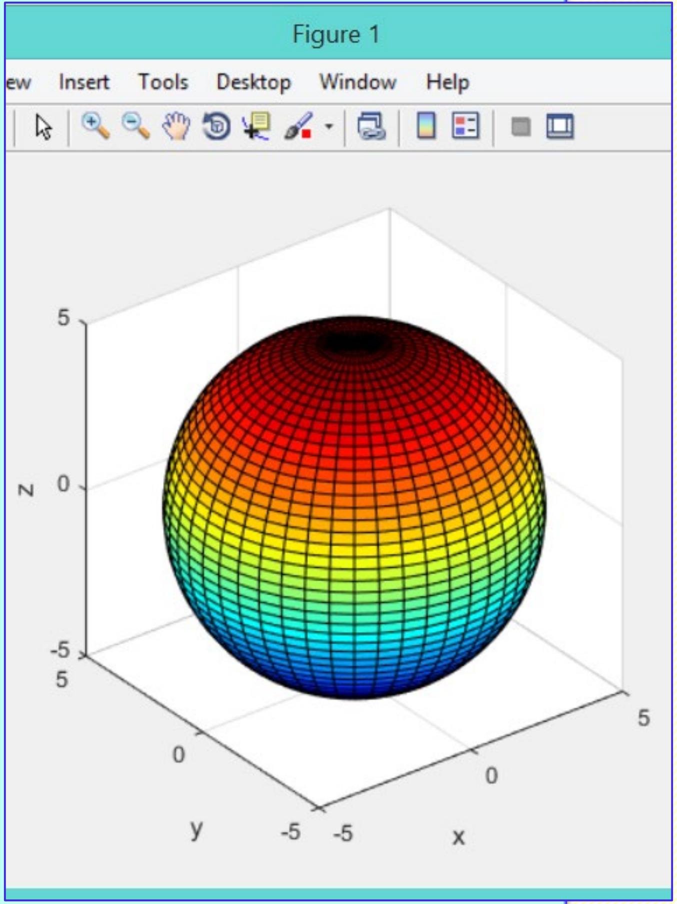
MATLAB: Simple SPHERE with R=1;

```
%% SIMPLE SPHERE
quality=48;

[x,y,z]=sphere(quality);

surf(x,y,z);
axis equal;
colormap jet;

xlabel('x'); ylabel('y'); zlabel('z');
```

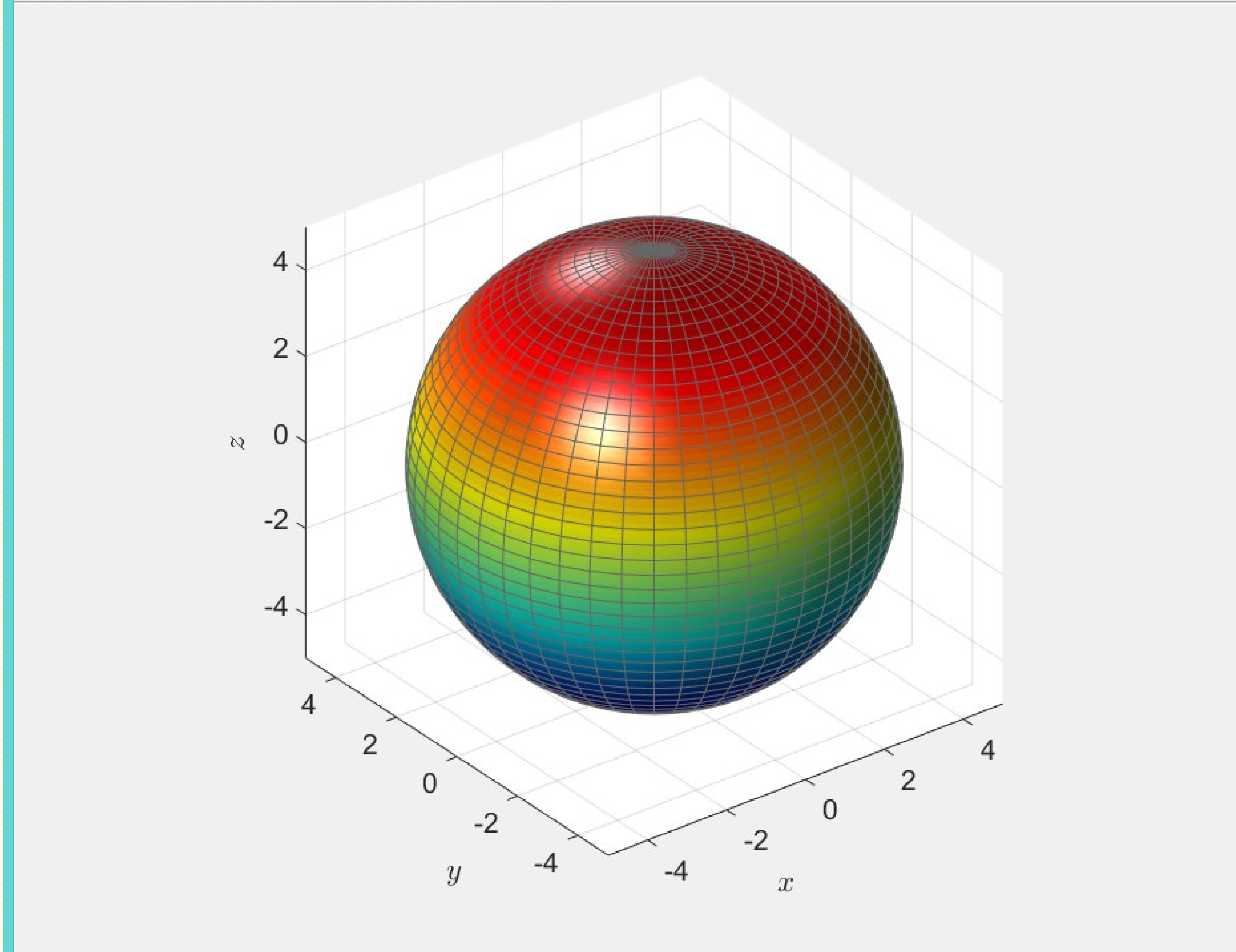


MATLAB: Simple SPHERE with R=5;

```
%% SIMPLE SPHERE
R=5; quality=48;

[x,y,z]=sphere(quality);

x=x*R; y=y*R; z=z*R;

surf(x,y,z);
axis equal;
colormap jet;

xlabel('x'); ylabel('y'); zlabel('z');
```

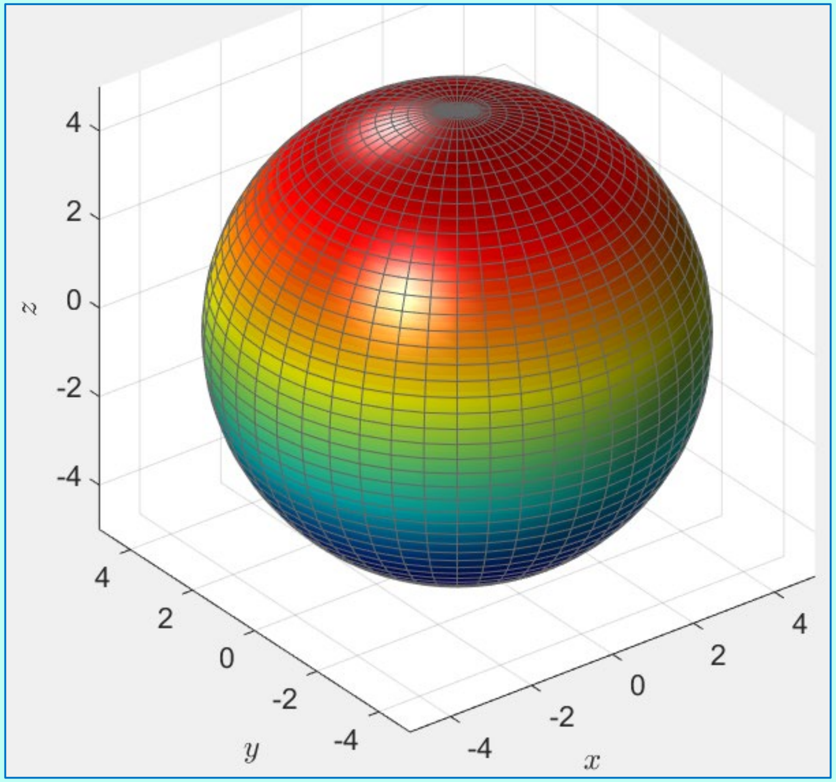


MATLAB: “Advanced” plotting of a Sphere



MATLAB: Script for Advanced Sphere

```
%% ADVANCED PLOTTING OF A SPHERE
% Designed by Prof P.M.Trivailo
clc; clear; close all
R=5; quality=48;

[x,y,z]=sphere(quality);
x=x*R; y=y*R; z=z*R;

surf(x,y,z,'FaceLighting','phong',...
           'FaceColor','interp',...
           'EdgeColor',[.4 .4 .4],...
           'BackFaceLighting','lit');
axis equal;
light('Position',[1 3 2]);
light('Position',[-3 -1 3]);
colormap jet; material shiny;
```




```
xl=xlabel('$x$'); yl=ylabel('$y$'); zl=zlabel('$z$');
set([xl, yl, zl],'Interpreter','LaTeX');

ticks=[-6:2:6];
set(gca,'FontSize',16,'Xtick',ticks,'Ytick',ticks,'Ztick',ticks)
set(gcf,'Position',[488    43    930    71]);
```

MATLAB “meshgrid”:

Essential Command for Plotting 3D Surfaces

MATLAB: MESHGRID

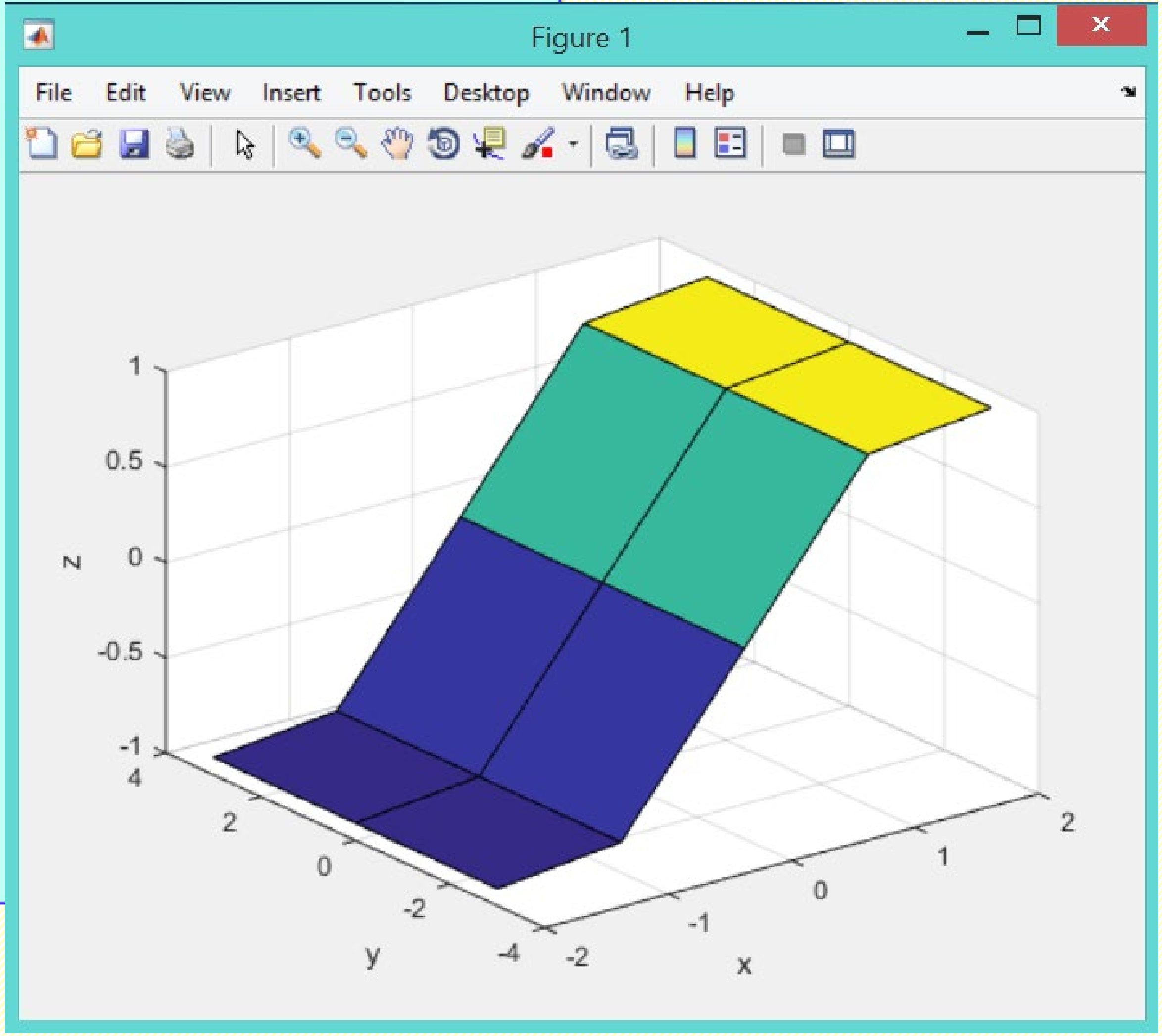
Understanding inputs and outputs of meshgrid

```
>> [X,Y] = meshgrid(-2:1:2, -3:3:3)

X =

    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2


Y =

    -3    -3    -3    -3    -3
     0     0     0     0     0
     3     3     3     3     3
```

```
>> size(X)

ans =

     3     5

>> size(Y)

ans =

     3     5
```

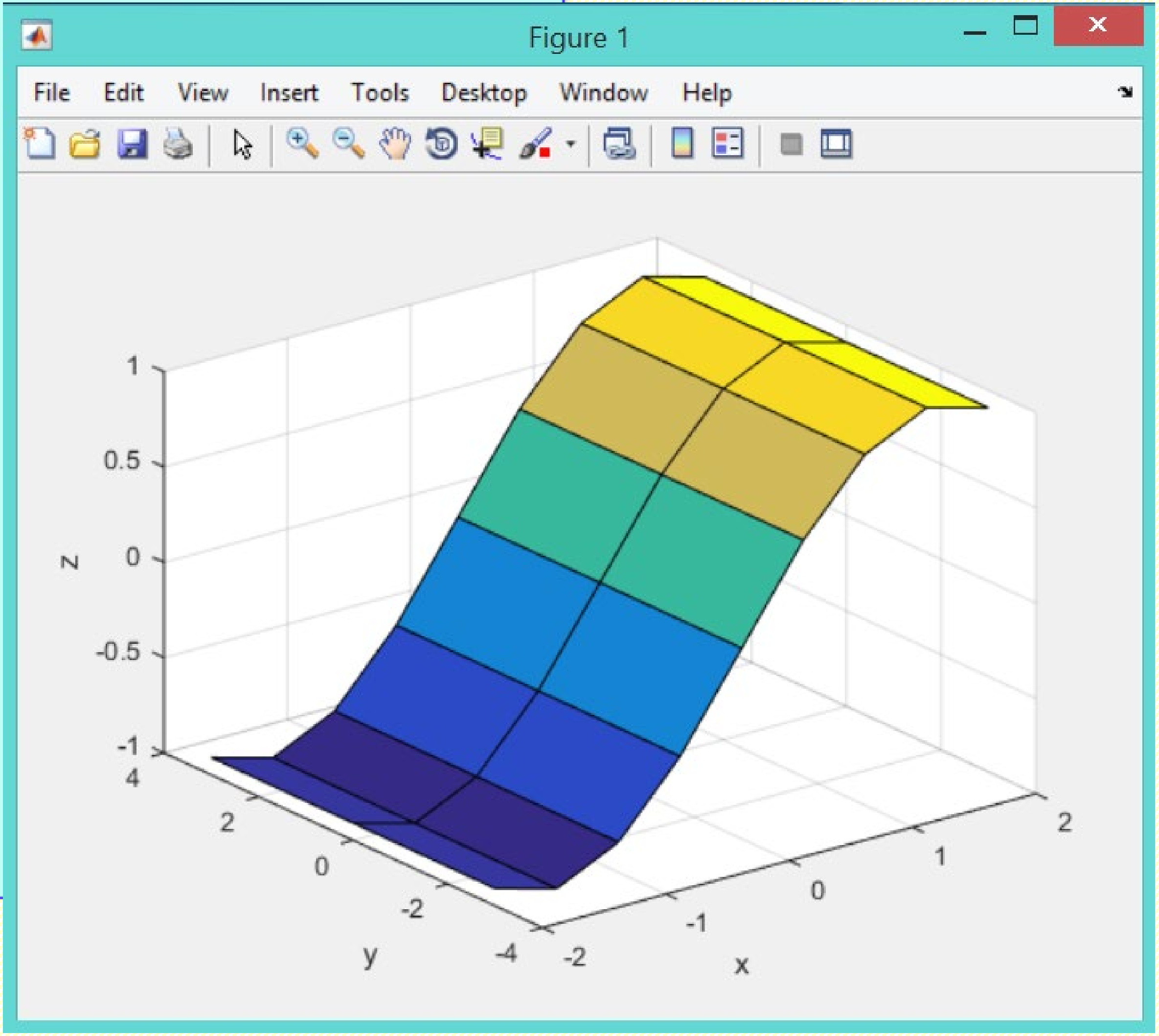
MATLAB: MESHGRID: DISCUSSION

```
[X,Y] = meshgrid(-2: 1 : 2,     -3:3:3);

Z = sin(X);

surf(X,Y,Z);

xlabel('x');
ylabel('y');
zlabel('z');
```



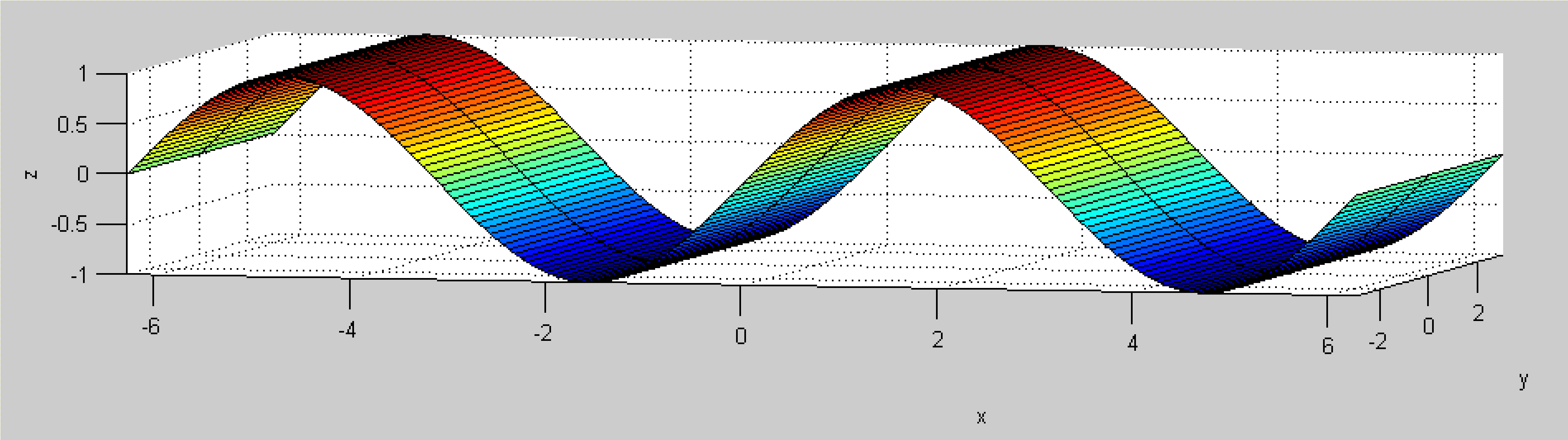
MATLAB: MESHGRID: DISCUSSION

```
[X,Y] = meshgrid(-2: 0.5 : 2,   -3:3:3);

Z = sin(X);

surf(X,Y,Z);

xlabel('x');
ylabel('y');
zlabel('z');
```



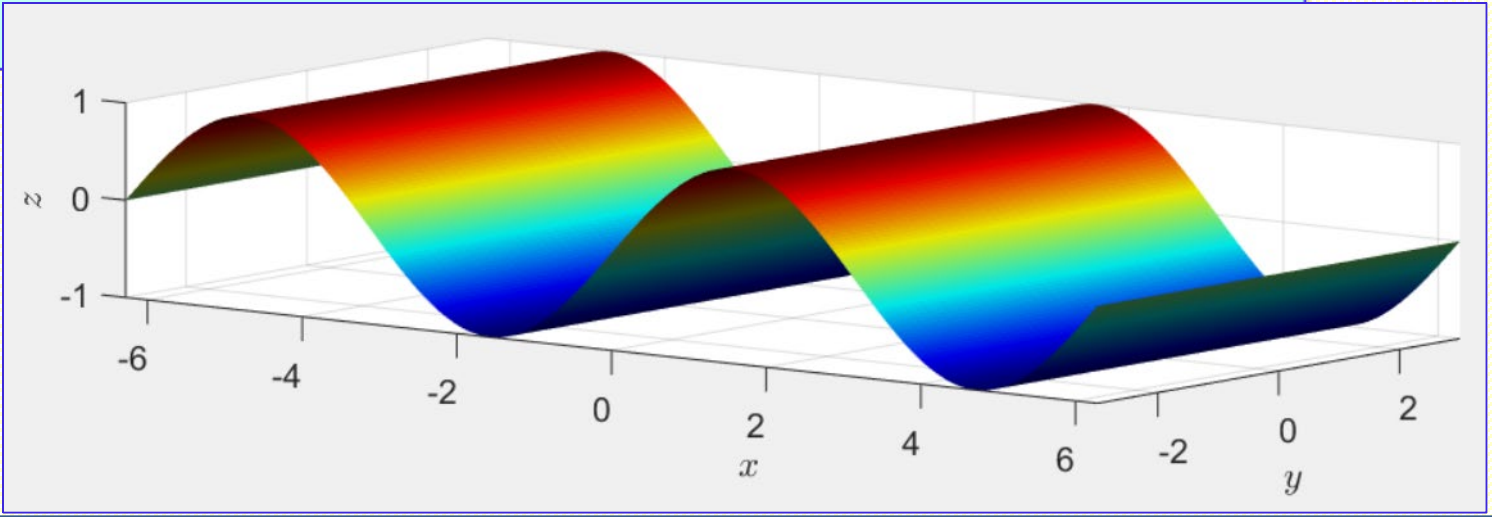
MATLAB: MESHGRID: BASIC PLOT

```
[X,Y] = meshgrid([-1:0.01:1]*2*pi, -3:3:3);
Z=sin(X);
surf(X,Y,Z);
xlabel('x'); ylabel('y'); zlabel('z');
axis([-1 1 -1 1 -1 1]*5); axis equal
view([14, 4])
set(gcf,'Position', [38  294  1074  405])
```



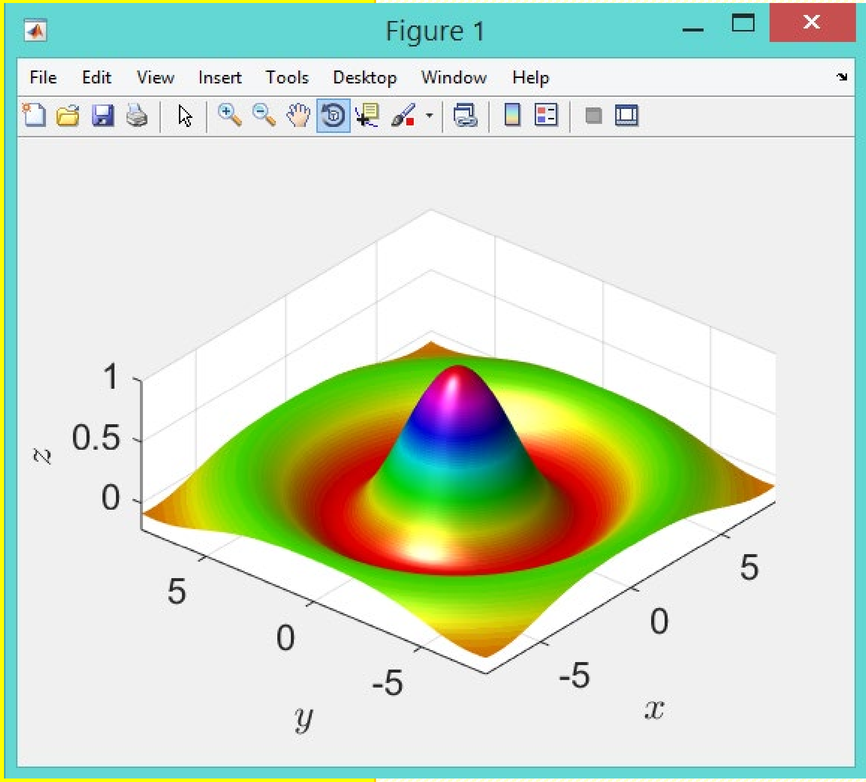
MATLAB: MESHGRID: ADVANCED VER

```
[X,Y] = meshgrid([-1:0.01:1]*2*pi, -3:3:3);
Z=sin(X); surf(X,Y,Z);
xlabel('$x$','Interpreter','LaTeX');
ylabel('$y$','Interpreter','LaTeX');
zlabel('$z$','Interpreter','LaTeX');
colormap jet
axis([-1 1 -1 1 -1 1]*5); axis equal
view([38, 8]); %view([14, 4])
set(gcf,'Position', [38  294  1074  405])
set(gca,'FontSize',16);
shading interp; light; lighting phong;
rotate3d on
```



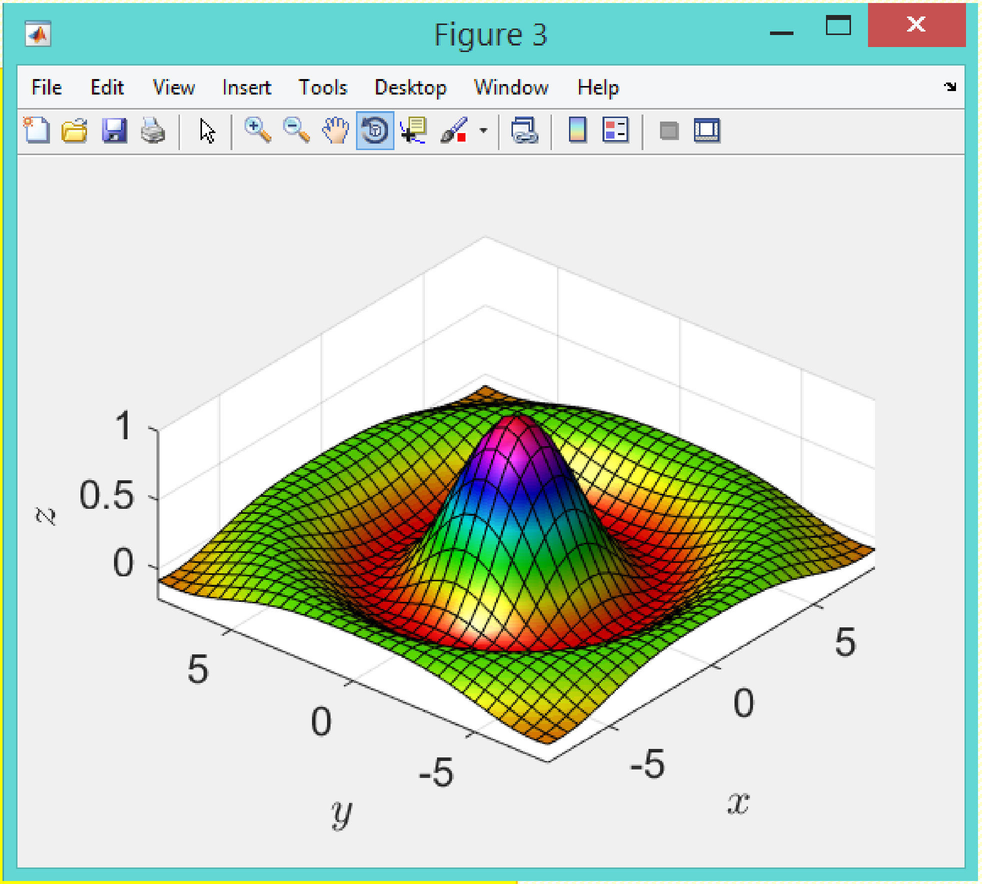
MANIPULATIONS WITH SURFACES: Class Exercises (lighting, rotate3d)

```
clear; close a[X,Y] = meshgrid(-8:.1:8);
R = sqrt(X.^2 + Y.^2) + eps;
Z = sin(R)./R;

figure
colormap hsv
surf(X,Y,Z,'FaceColor','interp',...
    'EdgeColor','none',...
    'FaceLighting','gouraud')
daspect([5 5 1])
axis tight
view(-50,30)
camlight left
xl=xlabel('$x$');
yl=ylabel('$y$');
zl=zlabel('$z$');
set([xl,yl,zl],'Interpreter','LaTeX');
set(gca,'FontSize',18);
rotate3d on
```



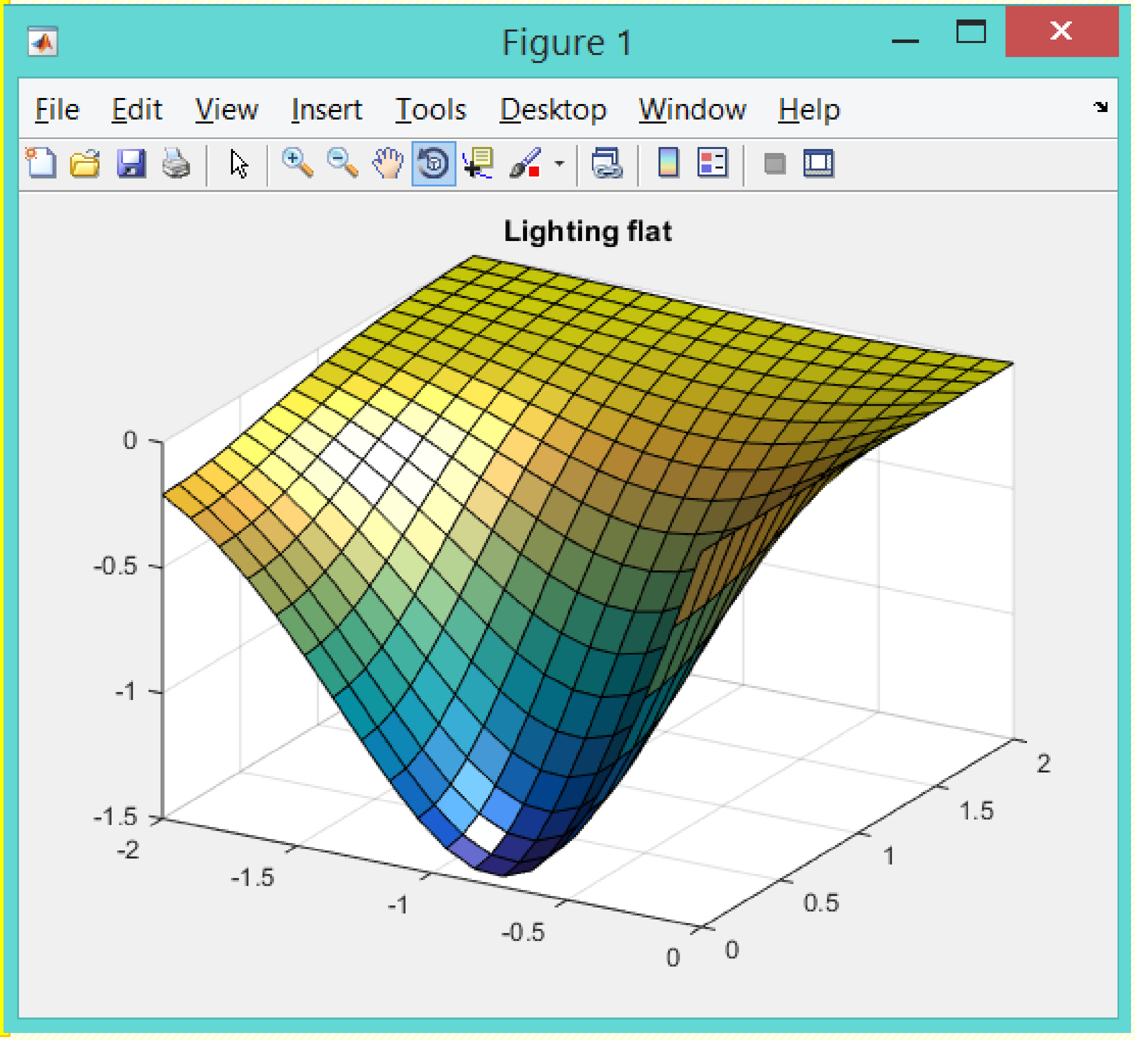
MANIPULATIONS WITH SURFACES: Class Exercises (lighting, rotate3d)

```
figure
colormap hsv
surf(X,Y,Z,…
   'FaceColor','interp',...
   'FaceLighting','gouraud')
%   'EdgeColor','none',...

daspect([5 5 1])
axis tight
view(-50,30)
camlight left
xl=xlabel('$x$');
yl=ylabel('$y$');
zl=zlabel('$z$');
set([xl,yl,zl],'Interpreter','LaTeX');
set(gca,'FontSize',18);
rotate3d on
```



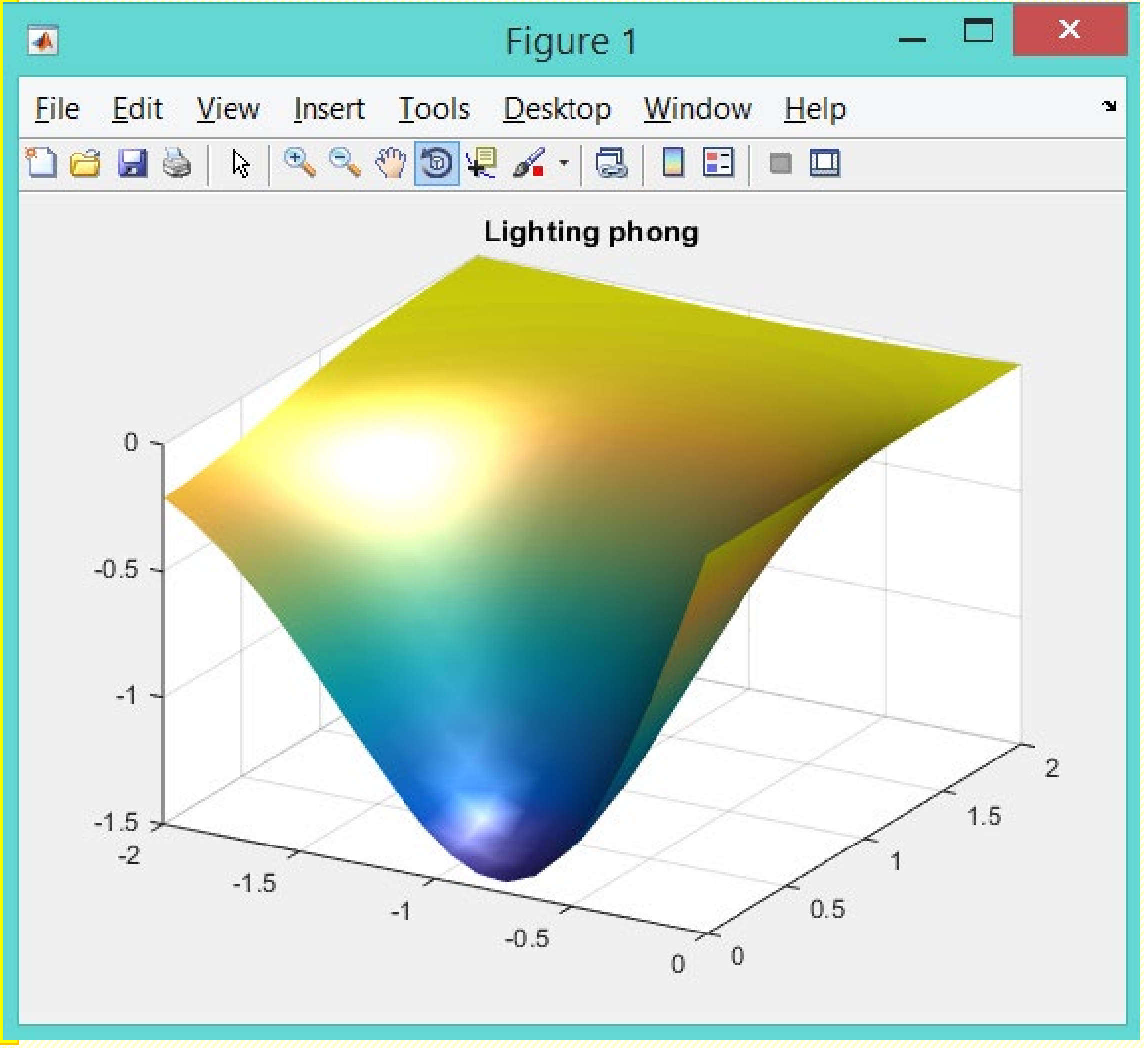
MANIPULATIONS WITH SURFACES: Class Exercises (Lighting)

```
% Create a grid of x and y points
points = linspace(-2, 0, 20);
[X, Y] = meshgrid(points, -points);

% Define the function Z = f(X,Y)
Z = 2./exp((X-.5).^2+Y.^2)- ...
    2./exp((X+.5).^2+Y.^2);

% "flat" lighting is good
% for faceted surfaces
surf(X, Y, Z);
view(30, 30);
shading faceted;
light; lighting flat;
title('Lighting flat');
rotate3d on;
```



MANIPULATIONS WITH SURFACES: Class Exercises (Lighting)

```
% Create a grid of x and y points
points = linspace(-2, 0, 20);
[X, Y] = meshgrid(points, -points);

% Define the function Z = f(X,Y)
Z = 2./exp((X-.5).^2+Y.^2)-...
     2./exp((X+.5).^2+Y.^2);

% "phong" lighting is good for
% curved, interpolated surfaces.
% "gouraud“ is also good for
% curved surfaces
surf(X, Y, Z); view(30, 30);
shading interp
light; lighting phong;
title('Lighting phong')
rotate3d on
```

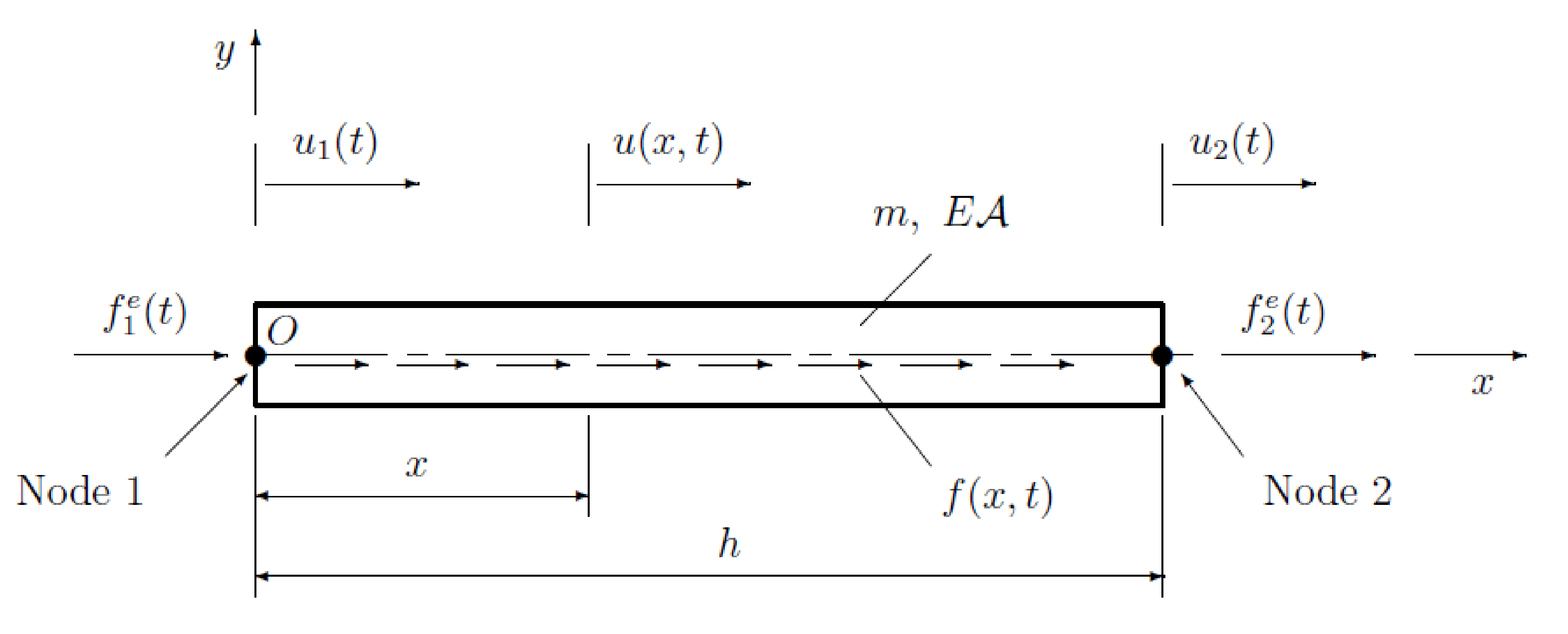


FEM:

Please, refer to the Separate File

https://drive.google.com/open?id=1-6Sh6mctXu4kYy5Sp8oOBTo-hd2u1mdU

FEM: TRUSS ELEMENT



$$u(x,t) = \left(1 - \frac{x}{h}\right) u_1(t) + \frac{x}{h} u_2(t).$$

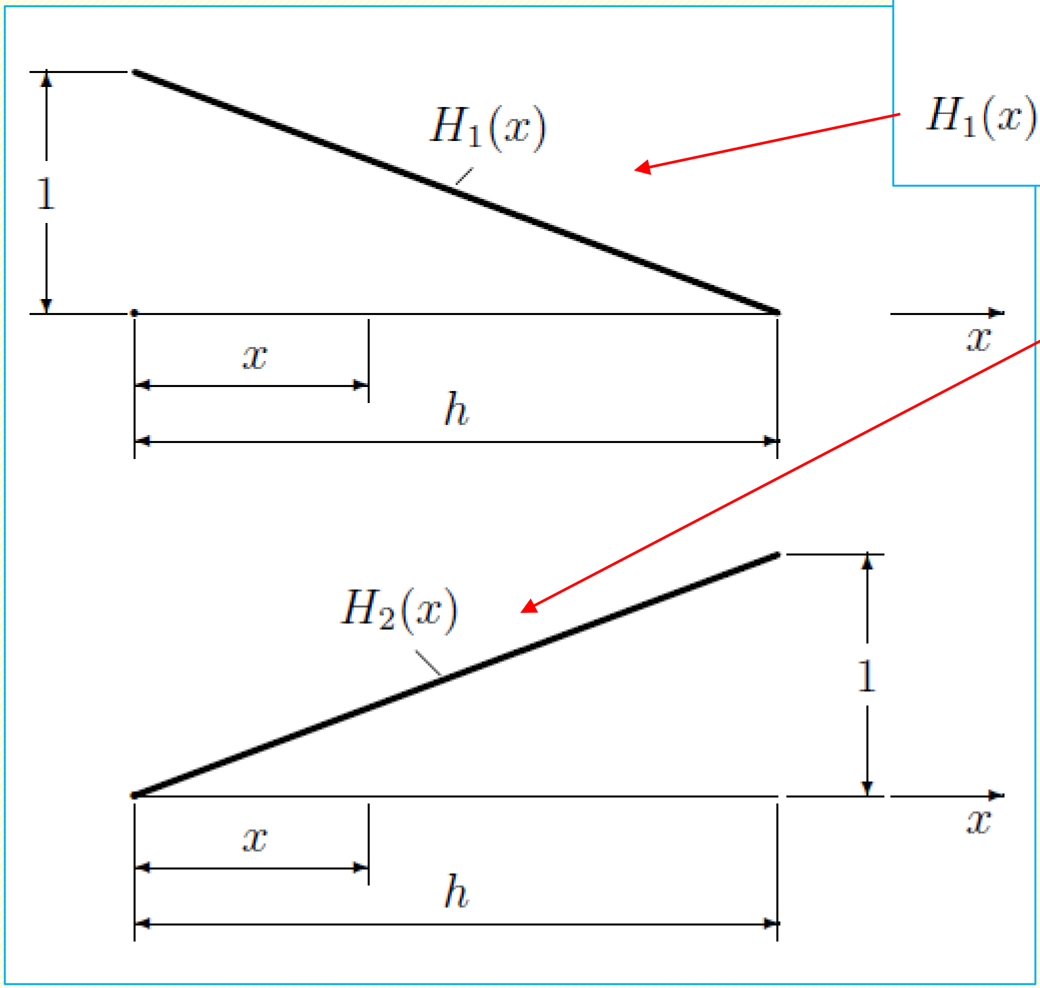
$$u(x,t) = H_1(x)\, u_1(t) + H_2(x)\, u_2(t), \qquad \text{where}$$

$$H_1(x) = 1 - \frac{x}{h}; \qquad H_2(x) = \frac{x}{h}$$

FEM: SHAPE FUNCTIONS for TRUSS ELEMENT

$$u(x,t) = H_1(x)\,u_1(t) + H_2(x)\,u_2(t), \qquad \text{where}$$

$$H_1(x) = 1 - \frac{x}{h}; \qquad H_2(x) = \frac{x}{h}$$



$$u(x,t) = \left(1 - \frac{x}{h}\right) u_1(t) + \frac{x}{h}\, u_2(t).$$

FEM: STIFFNESS & MASS MATRICES

Stiffness Matrix for a 1D Truss FE:
(Global and FE Local Coordinate Systems)

$$[k^e] = \frac{E\mathcal{A}}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Consistent Mass Matrix for a 1D Truss FE:
(Global and FE Local Coordinate Systems)

$$[m^e] = \frac{mh}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

FEM: NODAL FORCES

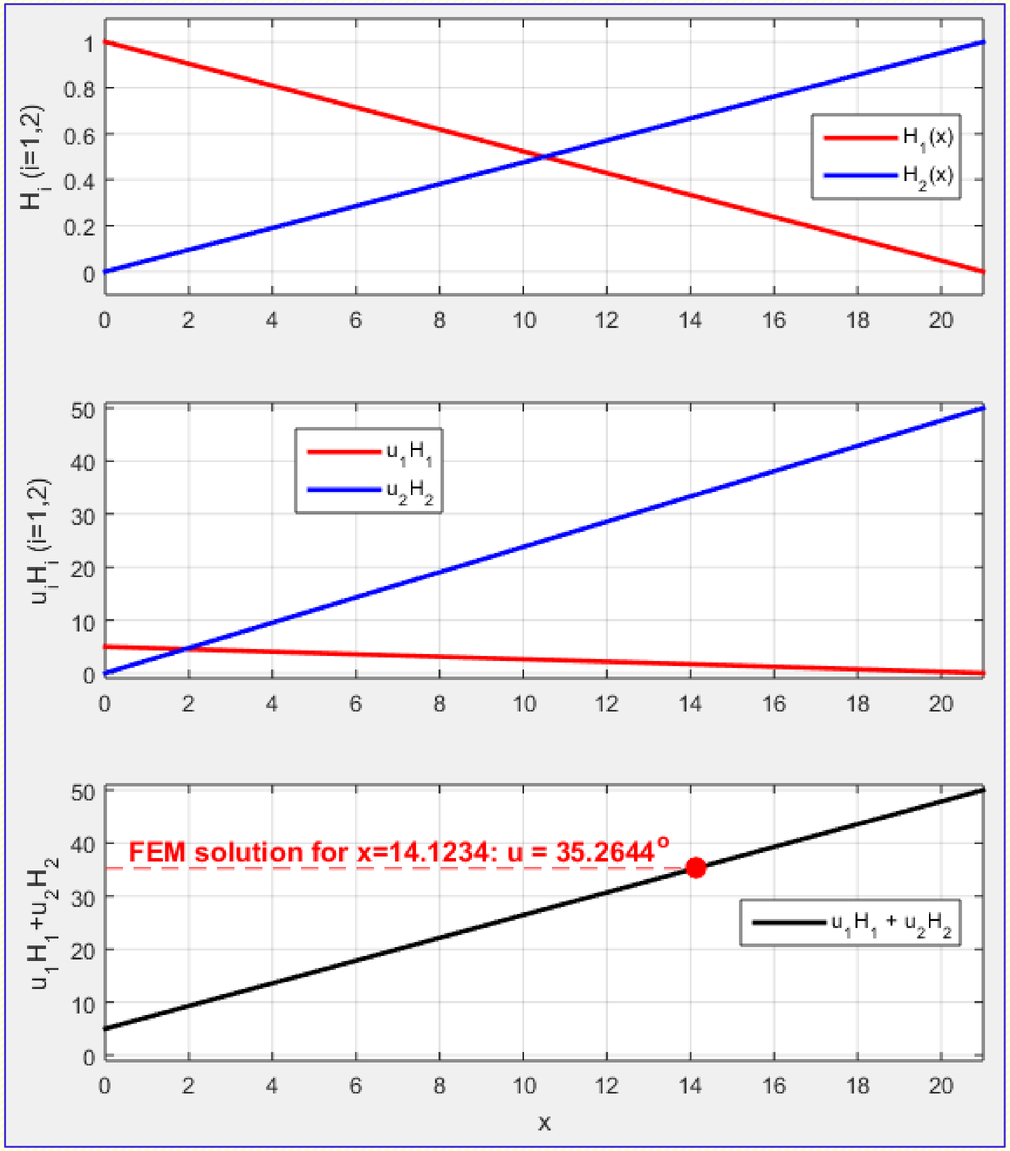
Nodal Forces for a 1D Truss FE:

$$f_1^e(t) = \int_0^h f(x,t)\left(1 - \frac{x}{h}\right)\,dx = \int_0^h f(x,t)\,H_1(x)\,dx$$

$$f_2^e(t) = \int_0^h f(x,t)\left(\frac{x}{h}\right)\,dx = \int_0^h f(x,t)\,H_2(x)\,dx$$

FEM: EXAMPLE with TRUSS ELEMENT

$$h = 21 \quad [\mathrm{m}]$$
$$u_1 = 5 \quad [\mathrm{m}]$$
$$u_2 = 50 \quad [\mathrm{m}]$$



FEM: MATLAN Script for the EXAMPLE above

```
%% OENG1116-S1-2020
% Designed by Prof P.M.Trivailo (C) 2020
% FEM example for the 2DOF Axial Rod
%---------------------------

clear; clc; close('all')
h=21; u1=5; u2=50; % m

x=[0:0.01:1]*h;

H1x = 1-x/h; H2x = x/h;

figure;
%--------------------------
subplot(3,1,1)
plot(x,H1x,'r', x,H2x,'b','LineWidth',2);
h_legend=legend('H_1(x)','H_2(x)','Location','East');
ylabel('H_i (i=1,2)')
grid on
axis([0 h -.1 1.1])
```

```
% Continuation of the Script
%--------------------------
subplot(3,1,2)
plot(x,u1*H1x,'r',  x,u2*H2x,'b','LineWidth',2);
legend('u_1H_1','u_2H_2','Location','best');
ylabel('u_iH_i (i=1,2)')
grid on
axis([0 h -1 51])
%--------------------------
subplot(3,1,3)
u = u1*H1x + u2*H2x;
plot(x,u,'k','LineWidth',2);
legend('u_1H_1 + u_2H_2','Location','East');
grid on
axis([0 h -1 51])
ylabel('u_1H_1+u_2H_2')
xlabel('x')
xC=14.1234; %Just an example: we would like to know u at this point
yC=interp1(x,u,xC);
line('XData',xC, 'YData',yC,'Marker','o','MarkerSize',8,'Color',[1 0 0])
set(gcf,'Position', [488    47   649   733])
```

APPENDIX:

Useful Examples, supporting FEM

EXAMPLE-1: COUNT IN A LOOP

PROBLEM: Let us create an array of randomly generated **N** integers (in the range **0-Nmax**) and determine the number **COUNT** of elements in this array, which are divisible by integer number **DIV.** To make it more practical, we will also aim to display the information on the screen and to ***WRITE RESULTS IN THE FILE.***

We first clear all variables and screen, then describe the simulation case and with **randi** command generate array **[a]**. Then we are using **uigetfile** gui interface to open the file for appending the data to the selected file. *Note:* the comment lines show experiments & discussions in class and lines are kept as a reminder of other options.

```
%%
%----- EXAMPLE-1: COUNTING NUMBERS IN THE CYCLE
clear; clc; close('all');
Nmax=1000; N=30; DIV=3;
a=randi(Nmax,1,N)
%a=[1:5];
%
[ff,dd]=uigetfile('*.txt', 'Select the File to Write your Counts:');
%fid=fopen('oeng1116_count.txt','w');
fid=fopen(ff, 'w');
```

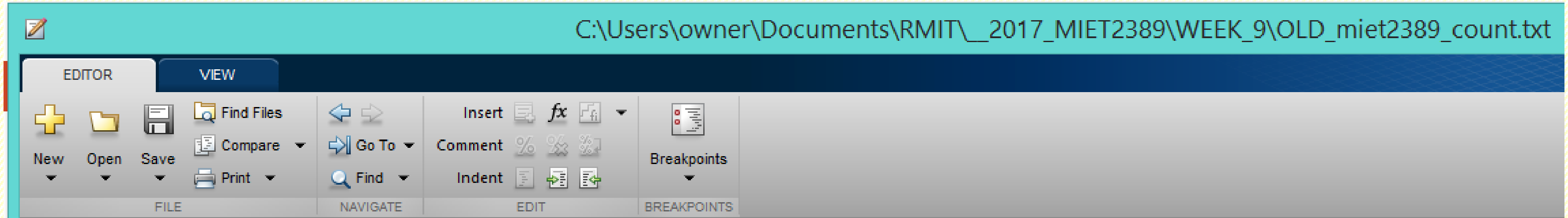
Then we print the generated array **[a]** into the file and set the **COUNTER** to zero:

```
fprintf(fid,'==========MY ARRAY [a] IS:======= \n')
fprintf(fid,'%4i %4i %4i %4i %4i %4i %4i %4i %4i %4i \n',a);
COUNT=0;
```

At last, we design the loop to inspect every element **a[ii]** in the array and to calculate the modulus of the division of the current element by DIV. If the number **a[ii]** is divisible, then we increase the value of the counter COUNT, display the result on the screen and print it to the file. At the end of the segment, the file is closed and the command window (with findings) is displayed:

```
for ii=1:length(a)
   if mod(a(ii),DIV)==0,
      COUNT=COUNT+1;
      disp(sprintf('%2i. Number %3i discovered: a[%2i]=%3i',COUNT,a(ii),ii,a(ii)))
      fprintf(fid,'%2i. Number %3i discovered: a[%2i]=%3i \n',COUNT,a(ii),ii,a(ii))
   end
end
disp(sprintf('There are %i numbers in the array [a]',COUNT))
fclose(fid)
commandwindow
```

EXAMPLE-1: FILE OUTPUT



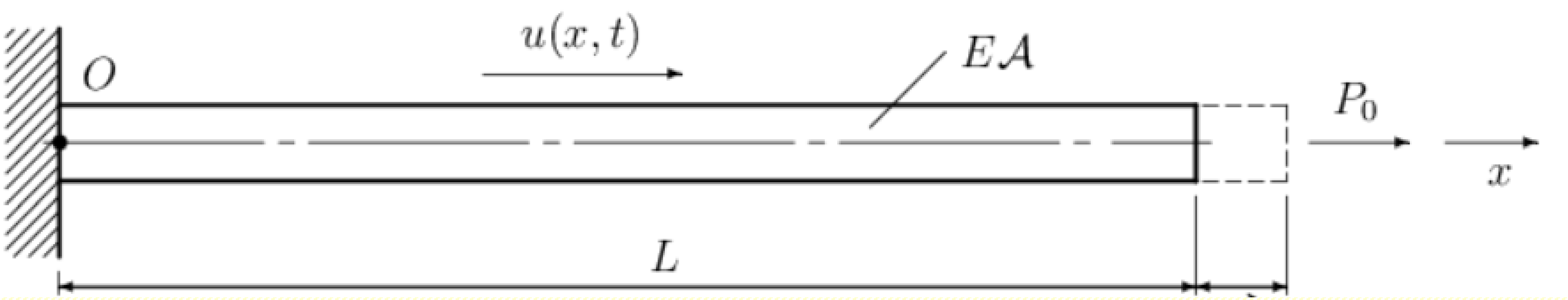

```
==========MY ARRAY [a] IS:=======
  65  437  827  395  614  819  887  932  191  259
 898  594  504  613  820  532  203  454  428  967
 621  696  721  347  517  557  157  563  695  427
 1. Number 819 discovered: a[ 6]=819
 2. Number 594 discovered: a[12]=594
 3. Number 504 discovered: a[13]=504
 4. Number 621 discovered: a[21]=621
 5. Number 696 discovered: a[22]=696
```

Comment: If **fid=fopen(ff, 'w');** command is used in the main script, the existing write-data file will be destroyed. In case you wish to accumulate the data for a few experiments, you may need to change your command to **fid=fopen(ff, ‘a');**

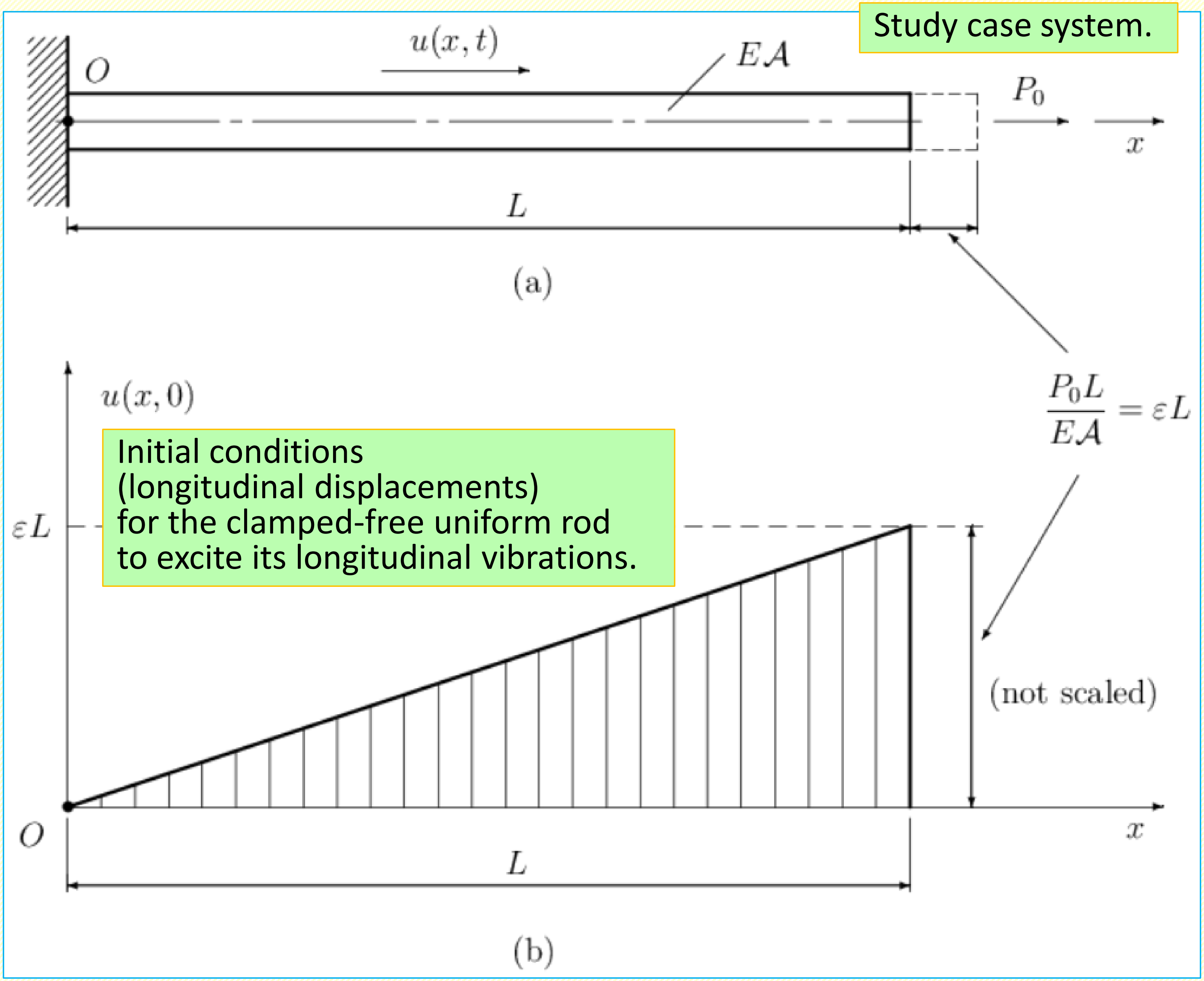
EXAMPLE-2: Axial Vibrations & FEM

PROBLEM: In one of the previous classes, analytical (exact) response of the clamped bar, excited with the initial stretch at the free end, was plotted as a "*t-x-u*" 3D surface. *For completeness, this script and results are presented below.*

TASK: A uniform rod, with one end fixed and other free, is stretched under a static load, as shown in Figure, and suddenly released from rest at time t=0. From these initial conditions, determine the longitudinal displacements u(x,t).



REFERENCE: **Trivailo P.M.** (2008), Vibrations: Theory & Aerospace Applications, Vol.1&2, - (The textbook for senior undergraduate and graduate aerospace students). - Melbourne: RMIT Publisher - 2008. - 247pp +348pp=595 pp., 355 ill, 4 software programs.



Study case system.
$u(x,t)$
O
$E\mathcal{A}$
P_0
x
L
(a)
$\frac{P_0 L}{E\mathcal{A}} = \varepsilon L$
$u(x,0)$
Initial conditions
(longitudinal displacements)
for the clamped-free uniform rod
to excite its longitudinal vibrations.
εL
(not scaled)
O
x
L
(b)

ANALYTICAL SOLUTION

The *initial conditions* are defined by

$$u(x,t)|_{t=0} = \varepsilon x, \qquad \left.\frac{\partial u(x,t)}{\partial t}\right|_{t=0} = 0.$$

Due to the Eq.(8.13) the displacements $u(x,t)$ and velocities $\dot{u}(x,t)$ of a uniform rod vibrating longitudinally are:

$$u(x,t) = \sum_{r=1}^{\infty} \sin\frac{(2r-1)\pi x}{2L}\,(C_r \sin\omega_r t + D_r \cos\omega_r t),$$

$$\dot{u}(x,t) = \sum_{r=1}^{\infty} \omega_r \sin\frac{(2r-1)\pi x}{2L}\,(C_r \cos\omega_r t - D_r \sin\omega_r t).$$

From the second initial condition we derive

$$0 = \sum_{r=1}^{\infty} C_r\,\omega_r \sin\frac{(2r-1)\pi x}{2L} \qquad \text{or} \qquad C_r = 0.$$

From the first initial condition we obtain,

$$\varepsilon x = \sum_{r=1}^{\infty} D_r \sin\frac{(2r-1)\pi x}{2L}.$$

If we multiply this equation by $\sin \frac{(2s-1)\pi x}{2L}$ and integrate from $x = 0$ to $x = L$, all the terms on the right side will be zero, except the term $r = s$.

Thus, we arrive at the result

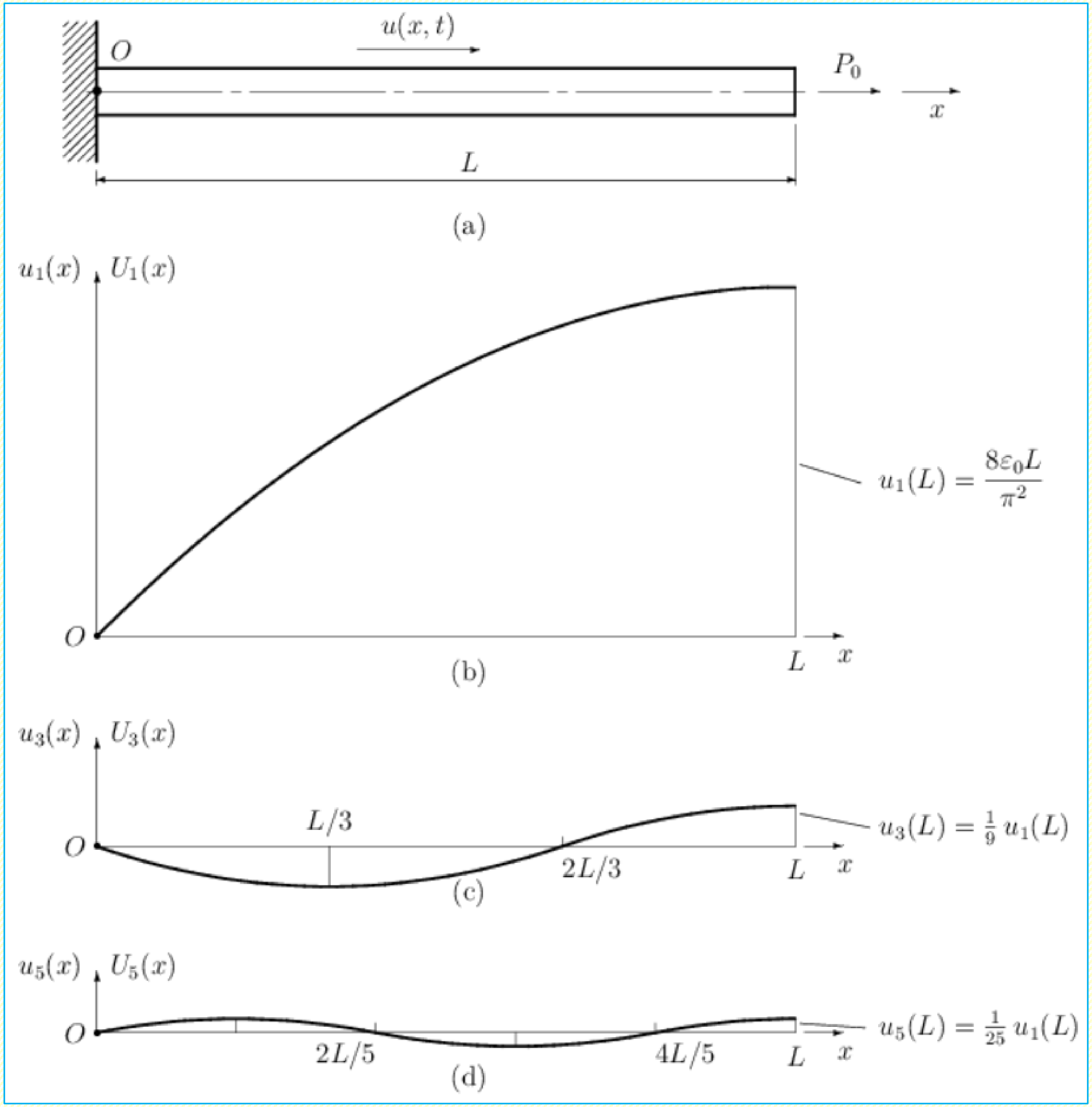
$$
\begin{aligned}
D_s &= \frac{2}{L}\int_0^L \varepsilon x \sin \frac{(2s-1)\pi x}{2L}\, dx = \\
&= \frac{8\varepsilon L}{(2s-1)^2\pi^2}\left[\sin \frac{(2s-1)\pi x}{2L}\right]_0^L = \\
&= \frac{8\varepsilon L}{(2s-1)^2\pi^2}(-1)^{s-1}.
\end{aligned}
$$

The above formula can be obtained after integration by parts of

$$\int x \sin(ax)\, dx = \frac{\sin(ax)}{a^2} - \frac{x \cos(ax)}{a}.$$

The solution of the problem is then:

$$u(x,t) = \frac{8\varepsilon L}{\pi^2}\sum_{r=1}^{\infty}\frac{(-1)^{r-1}}{(2r-1)^2}\sin\frac{(2r-1)\pi x}{2L} \times \cos\frac{(2r-1)\pi ct}{2L}. \tag{8.22}$$



The scaled contribution of the first three modes to the total longitudinal response of the excited bar.

It is Interesting to Know

From an initial condition for the stretched rod :

$$u(L,0) = \varepsilon L.$$

it directly follows from the formula (8.22) an interesting result, which is well known in mathematical literature:

$$\frac{\pi^2}{8} = \frac{1}{1^2} + \frac{1}{3^2} + \frac{1}{5^2} + \frac{1}{7^2} + \frac{1}{9^2} + \dots \tag{8.23}$$

Using (8.23), we can calculate π with a surprising accuracy:

$$\pi = 3,\ 141\,592\,653\,589\,793\,238\,462\,643\ \dots$$

In order to convince the reader, that the formula (8.23) is a really outstanding result, let us briefly remind the history of the figure π:

It is Interesting to Know (Cont’d)

- The estimation of the great Archimedes(287-212 B.C.) for the π is

$$3\tfrac{10}{71} < \quad \pi \quad < 3\tfrac{1}{7}, \quad \text{i.e.}$$
$$3,1408 < \quad \pi \quad < 3,1429$$

- The ratio $\frac{22}{7}$ was used as the π for a long time, in spite of the fact that it had beed found in China in V-th century that

$$\pi \approx \frac{355}{113} = 3,1415929$$

The same reult was rediscovered in Europe in XVI-th century.

- It was believed in India that

$$\pi = \sqrt{10} = 3,1622;$$

- The French mathematician Viet [Fransua Viet (1540-1603)] calculated π in 1579 with the accuracy of 9 digits;

It is Interesting to Know (Cont'd)

- The Holland mathematician Ludolf Van Ceilen calculated π in 1596 with the accuracy of 32 digits;
- The German mathematician Lambert proved in 1767 that π is an irrational figure;
- The German mathematician Lindeman proved in 1882 that π is a transcendential figure.

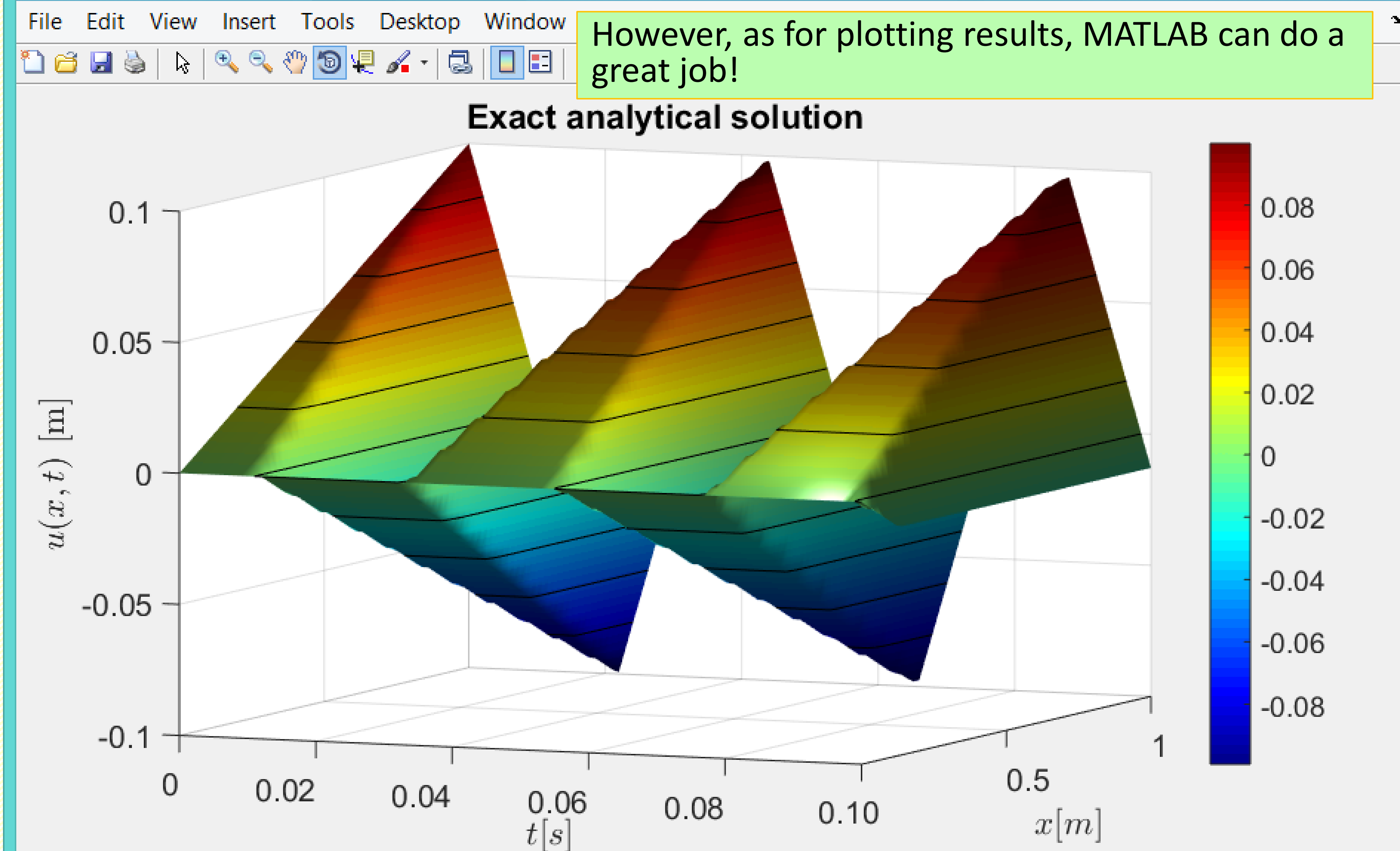
What about US?

```
>> sprintf('PI = %40.38f',pi)

ans =

PI = 3.14159265358979310000000000000000000000
```

Using formatted print, I managed to display 16 digits of **pi** only.

EXAMPLE-2: MATLAB RESULTS

However, as for plotting results, MATLAB can do a great job!



MATLAB SCRIPT (designed & discussed in class):

```
%%
% Designed by Prof P.M.Trivailo; Wk-3-2020-OENG1116
%-------------------------------------------
L=1; e=0.1; E=0.01*10^9; % Pa
rho=1.2*10^3; % kg/m^3
c=sqrt(E/rho);
tmax=0.1; % s
t=[0:0.01:1]*tmax; x=[0:0.02:1]*L;
[T,X]=meshgrid(t,x);
N=100; U=0;
for r=1:N
   U=U+((-1)^(r-1)/(2*r-1)^2)*...
      sin( (2*r-1)*pi*X/(2*L) ).*...
      cos( (2*r-1)*pi*c*T/(2*L));
end
U=U*8*e*L/(pi^2);
```

Note: full script of the file **oeng1116_2018_w5_axial_vibr.m** is available from the Blackboard

Comment: This is the core of the program. Problem parameters are entered. Solution is constructed by adding summation elements in a loop.

MATLAB SCRIPT (designed & discussed in class):

```
figure('Position',[10+10+10+940   -80   940   -80+600]);
surf(T,X,U);
axis([0 tmax   0 L   -e e]); grid on;
xlabel('$t [s]$','Interpreter','LaTeX');
ylabel('$x [m]$','Interpreter','LaTeX');
zlabel('$u(x,t)$ [m]','Interpreter','LaTeX');
title('Exact analytical solution‘,'FontWeight','bold');
rotate3d on;
hold on;
colormap jet; colorbar;
hh=camlight;
set(hh,'Position',[0.09,0.04,0.03])
view([23, 8]);
lighting phong; shading interp;
contour3(T,X,U,[-0.1:0.02:0.1],'k');
set(gca,'FontSize',16);
```

Comment: This is a “plotting results” segment. Note use of the colormap, colorbar, lighting, shading and contour3, discussed in class.

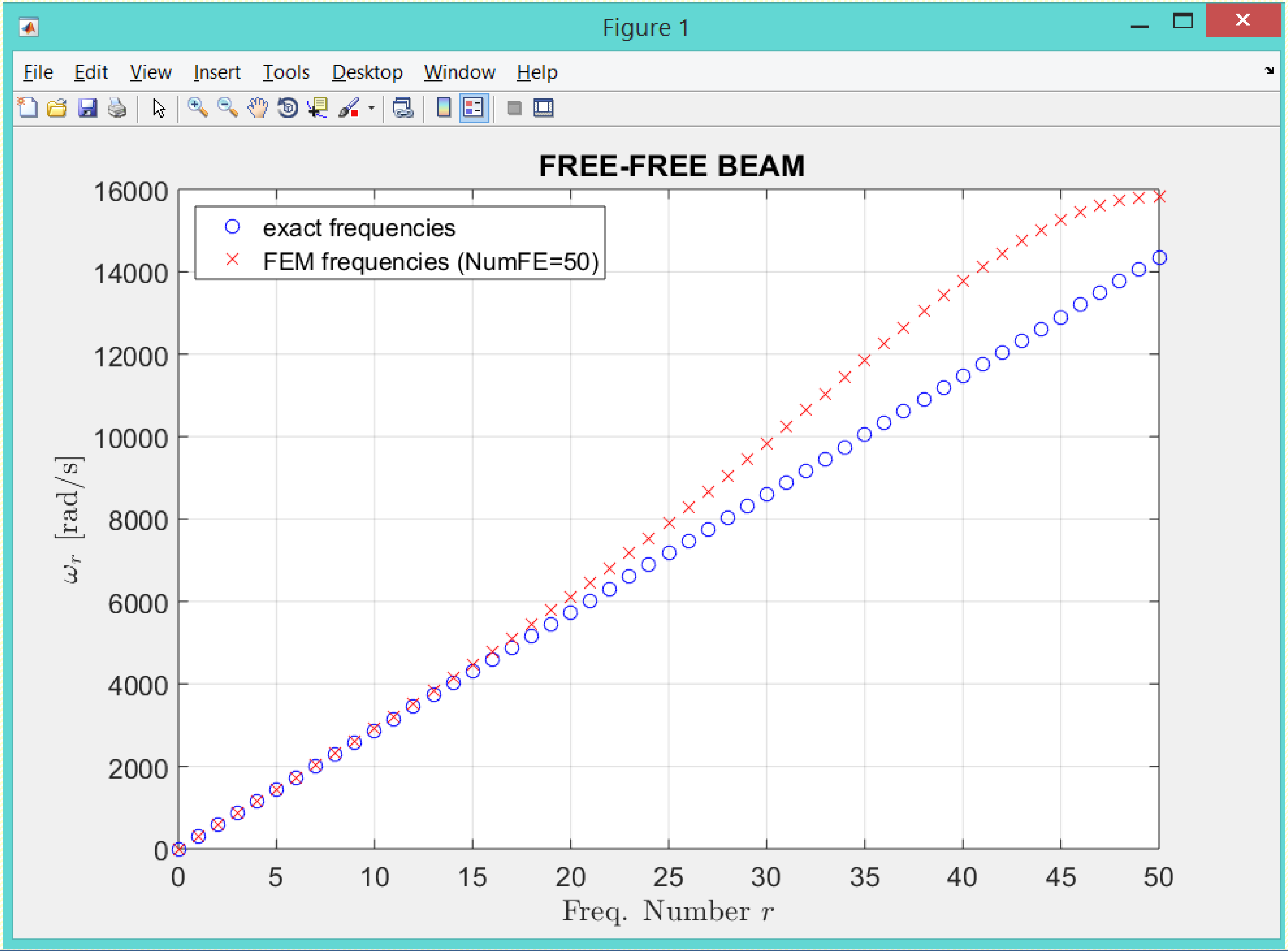


Figure 1
File Edit View Insert Tools Desktop Window Help
FREE-FREE BEAM
exact frequencies
FEM frequencies (NumFE=50)
ω_r [rad/s]
Freq. Number r
0
2000
4000
6000
8000
10000
12000
14000
16000
0
5
10
15
20
25
30
35
40
45
50

EXAMPLE-2: Axial Vibrations & FEM

In this class, we aim **to solve** (ourselves!) response for the same system, using the FEM method and then plot the FEM results as a 3D "*t-x-u*" surface.

Pursuing many educational objectives, I am presenting the following stages in the process:

(1) we formulate and program a **<u>supplementary task (!)</u>** of a FREE-FREE bar;

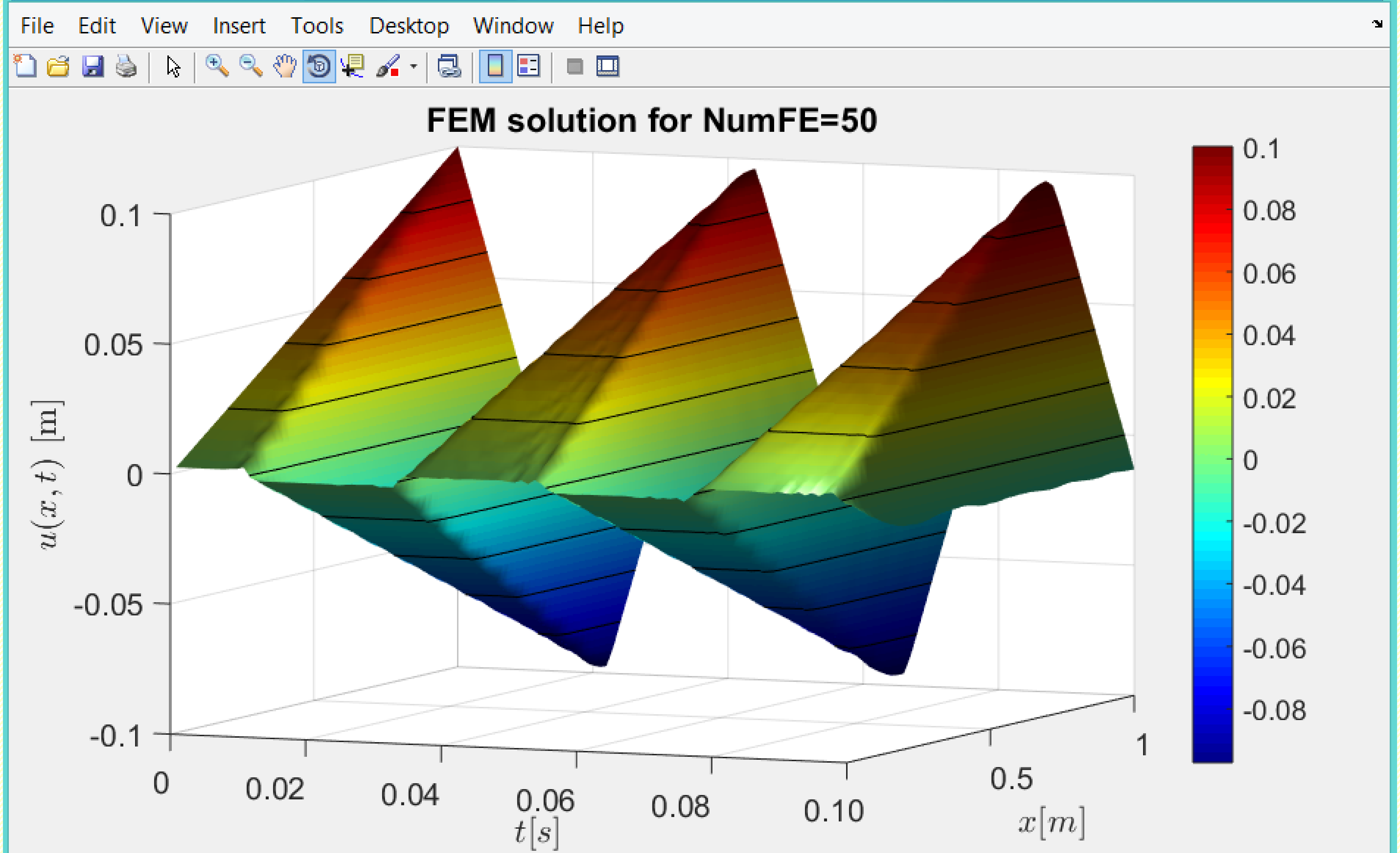
(2) We solve the eigenvalue problem for the supplementary task and experiment with the numbers of FE to compare the exact natural frequencies with the natural frequencies, obtained with FEM;

(3) we convert the supplementary task into the task of interest by applying the boundary conditions and solve the eigenvalue problem;

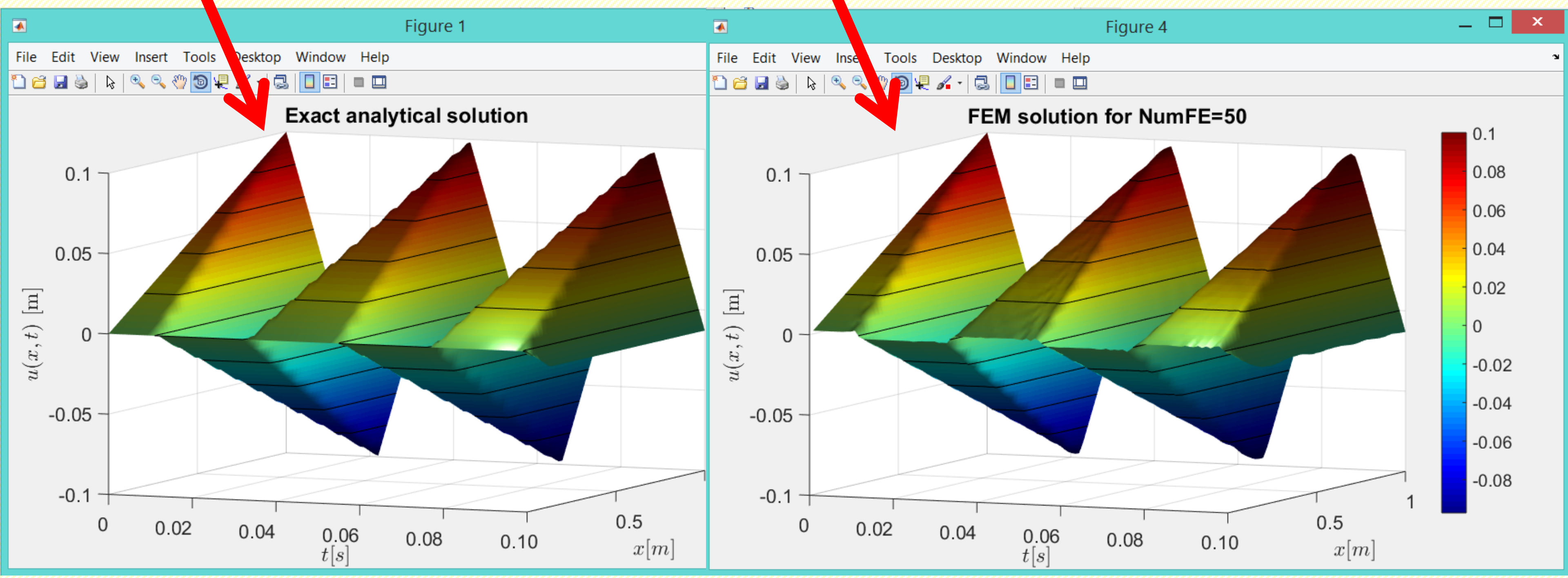
(4) we calculate response of the system using the ode45 procedure.

(Courtesy: MathWorks)

EXAMPLE-2: RESULTS



SIDE BY SIDE COMPARISON: EXACT (left) & FEM (right) SOLUTIONS and their representations



Comment: for comparison we are using the same axis limits and the same lighting

MATLAB SCRIPT (designed & discussed in class):

Note: full script of the file **oeng1116_2018_FEM_example.m will be** available from the Canvas

```
%%
%-----
clear; clc; close('all');
global dof M_gl K_gl C_gl
L=1; e=0.1; E=0.01*10^9; % Pa
rho=1.2*10^3; % kg/m^3
c=sqrt(E/rho); A=(0.08*0.08); EA=E*A;
NumFE=50;
h=L/NumFE; m=A*1*rho;
kk=(EA/h)*[1 -1; -1 1];
mm=(m*h/6)*[2 1; 1 2];
M=zeros([1,1]*(NumFE+1));
K=zeros([1,1]*(NumFE+1));
for ii=1:NumFE
   idx=[1 2]+(ii-1);    M(idx, idx)=M(idx, idx)+mm;    K(idx, idx)=K(idx, idx)+kk;
end
```

Comment: global variables declared; example data is entered; provisions for the global mass **[M]** and stiffness **[K]** matrices for the supplementary(!!!) free-free task are created;
elementary mass **[m^e]** and stiffness matrices **[k^e]** are entered;
global **[M]** and **[K]** matrices (for the FREE-FREE bar) are assembled.
Note: properties for different FEs is assumed to be the same.

MATLAB SCRIPT (continued):

```
[U,D]=eig(K,M);
disp(sprintf('===ANALYTICAL SOLUTIONS FOR FREE=FREE BEAM ARE:======'))
disp(sprintf('w0 = %g \t w1 = %g \t w2 = %g \t w3 = %g \t w4 = %g',pi*[0:4]*c/L))
disp(sprintf('==FEM SOLUTIONS FOR FREE=FREE BEAM (N=%i) ARE:======',NumFE))

str='';
for ii=0:NumFE
   if sqrt(D(ii+1,ii+1))<10^(-4),
      str1=sprintf('w%i = 0 \t ',ii);
   else
      str1=sprintf('w%i = %g \t ',ii,sqrt(diag(D(ii+1,ii+1))));
   end
   str=[str, str1];
end
disp(str)
```

Comment: solving eigenvalue problem for the supplementary(!!!) free-free task; display results in the command window.

MATLAB SCRIPT (continued):

```
%------ PLOTTING THE FREQUESNCIES THEORETICS vs FEM on the single plot--
figure;
plot([0:NumFE],pi*[0:NumFE]*c/L,'ob','MarkerSize',8);
hold on
plot([0:NumFE],diag(sqrt(D)),'xr','MarkerSize',8); grid on;
str=sprintf('FEM frequencies (NumFE=%i)',NumFE);
legend('exact frequencies',str,'Location','NorthWest');
ax1=axis;
xlabel('Freq. Number $r$','Interpreter','LaTeX');
ylabel('$\omega_r$ [rad/s]','Interpreter','LaTeX');
title('\bf FREE-FREE BEAM');
%%
set(gca,'FontSize',16);
```

Comment: plot results in the figure, add legend and axis labels.

MATLAB SCRIPT (continued):

```
%--- CONVERT FREE-FREE FEM into the CLAMPED-FREE FEM MODEL
M(1,:)=[]; M(:,1)=[];
K(1,:)=[]; K(:,1)=[];
[U2,D2]=eig(K,M);
disp(sprintf('===ANALYTICAL SOLUTIONS FOR CLAMPED=FREE BEAM ARE:======'))
disp(sprintf('w1 = %g \t w2 = %g \t w3 = %g \t w4 = %g \t w5 = %g \t',pi*(2*[1:5]-
1)*c/2/L))
disp(sprintf('===      FEM SOLUTIONS FOR CLAMPED=FREE BEAM (N=%i)
ARE:======',NumFE))
str='';
for ii=1:NumFE
    str=[str, sprintf('w%i = %g \t ',ii,sqrt(diag(D2(ii,ii))))];
end
disp(str)
```

Comment: Converting the FREE-FREE FEM model into the CLAMPED-FREE FEM model, solving the eigenvalue problem and programmatically plotting the FEM natural frequencies in command window.

MATLAB SCRIPT (continued):

```
%%
%------ PLOTTING THE FREQUESNCIES THEORETICAL vs FEM on the single plot--
figure;
plot([1:NumFE],pi*(2*[1:NumFE]-1)*c/2/L,'ob','MarkerSize',10);
hold on
plot([1:NumFE],diag(sqrt(D2)),'xr','MarkerSize',10);
str=sprintf('FEM frequencies (NumFE=%i)',NumFE);
legend('exact frequencies',str,'Location','NorthWest');
grid on; axis(ax1);
xlabel('Freq. Number $r$','Interpreter','LaTeX');
ylabel('$\omega_r$ [rad/s]','Interpreter','LaTeX');
title('\bf CLAMPED-FREE BEAM')
set(gca,'FontSize',16);
```

Comment: Plotting all **NumFE** exact (from analytical solution) and FEM natural frequencies for the CLAMPED-FREE bar on the same plot.

MATLAB SCRIPT (continued):

```
%---RESPONSE
M_gl =M;
K_gl =K;
C_gl =0*M;
dof=size(M_gl,1);
tmax=0.1;
tstep=0.001;
t=0:tstep:tmax;
e=0.1;
x0=[interp1([0 L], [0 e],[1:NumFE]/NumFE), zeros(1,dof)]';
[tt,xx] = ode45('FEM_xdot',t,x0);
figure;
plot(tt,xx(:,NumFE));
grid on;
ylabel('$u_{tip}$','Interpreter','LaTeX'); xlabel('$t$','Interpreter','LaTeX');
set(gca,'FontSize',16);
```

Comment: Calculating response of the system due to the initial excitation, using OED45 and plotting response of the tip of the bar.

Note: one of the most critical steps is formulation of the vector of the initial conditions **x0 !!!**

MATLAB SCRIPT (continued):

```
%---RESPONSE
M_gl =M;
K_gl =K;
C_gl =0*M;
dof=size(M_gl,1);
tmax=0.1;
tstep=0.001;
t=0:tstep:tmax;
e=0.1;
x0=[interp1([0 L], [0 e],[1:NumFE]/NumFE), zeros(1,dof)]';
[tt,xx] = ode45('FEM_xdot',t,x0);
figure;
plot(tt,xx(:,NumFE));
grid on;
ylabel('$u_{tip}$','Interpreter','LaTeX'); xlabel('$t$','Interpreter','LaTeX');
set(gca,'FontSize',16);
```

Comment: Plotting results as 3D surface plot.

MATLAB SCRIPT (continued):

Comment: Plotting results as 3D surface plot.

```
[TT,XX]=meshgrid(tt,[1:NumFE]/NumFE);
figure('Position',[10   -80   940   -80+600]);
surf(TT,XX,xx(:,1:NumFE)');
axis([0 tmax 0 L -e e]);
grid on;
xlabel('$t [s]$','Interpreter','LaTeX');
ylabel('$x [m]$','Interpreter','LaTeX');
zlabel('$u(x,t)$ [m]','Interpreter','LaTeX');
str=sprintf('FEM solution for NumFE=%i',NumFE);
title(str,'FontWeight','bold');
rotate3d on;
hold on;
colormap jet; colorbar; hh=camlight;
set(hh,'Position',[0.09,0.04,0.03]); view([23, 8]);
lighting phong; shading interp;
contour3(TT,XX,xx(:,1:NumFE)',[-0.1:0.02:0.1],'k'); set(gca,'FontSize',16);
```

MATLAB SCRIPT (continued):

```
x=xx(:,dof);
m = length(x);         % Window length
n = pow2(nextpow2(m));  % Transform length
y = fft(x,n);          % DFT
fs=1/(tstep);
f = (0:n-1)*(fs/n);        % Frequency range
power = y.*conj(y)/n;       % Power of the DFT
x1=xx(:,dof); m1 = length(x1);   n = pow2(nextpow2(m1));
%---Displacement at the tip---------
y1 = fft(x1,n);            % DFT
f = (0:n-1)*(fs/n);        % Frequency range
power1 = y1.*conj(y1)/n;        % Power of the DFT
%---Displacement at the middle of the span (if NumFE is even number!) -----
x2=xx(:,dof/2); y2 = fft(x2,n);   power2 = y2.*conj(y2)/n;        % Power of the DFT
figure; gg=plot(f,power1, f, power2); ax=axis; delete(gg); w1=sqrt(D2(1,1));
plot(f,power1, 'r-', f, power2, 'b-', [1 1+eps]*w1/2/pi,ax(3:4),'k--');
str3=sprintf('analytical solution: f=%7.4f Hz',w1/2/pi);
grid on; legend('FEM: node at the tip of the bar','FEM: middle node',str3);
xlabel('Frequency (Hz)'); ylabel('Power'); title(sprintf('{\\bf Periodogram:}  N=%i',n));
set(gca,'FontSize',16); ax=axis; axis([ax(1) ax(2)/2 ax(3) ax(4)])
```

Comment: processing responses (displacements) of the tip and mid-points, using FFT.

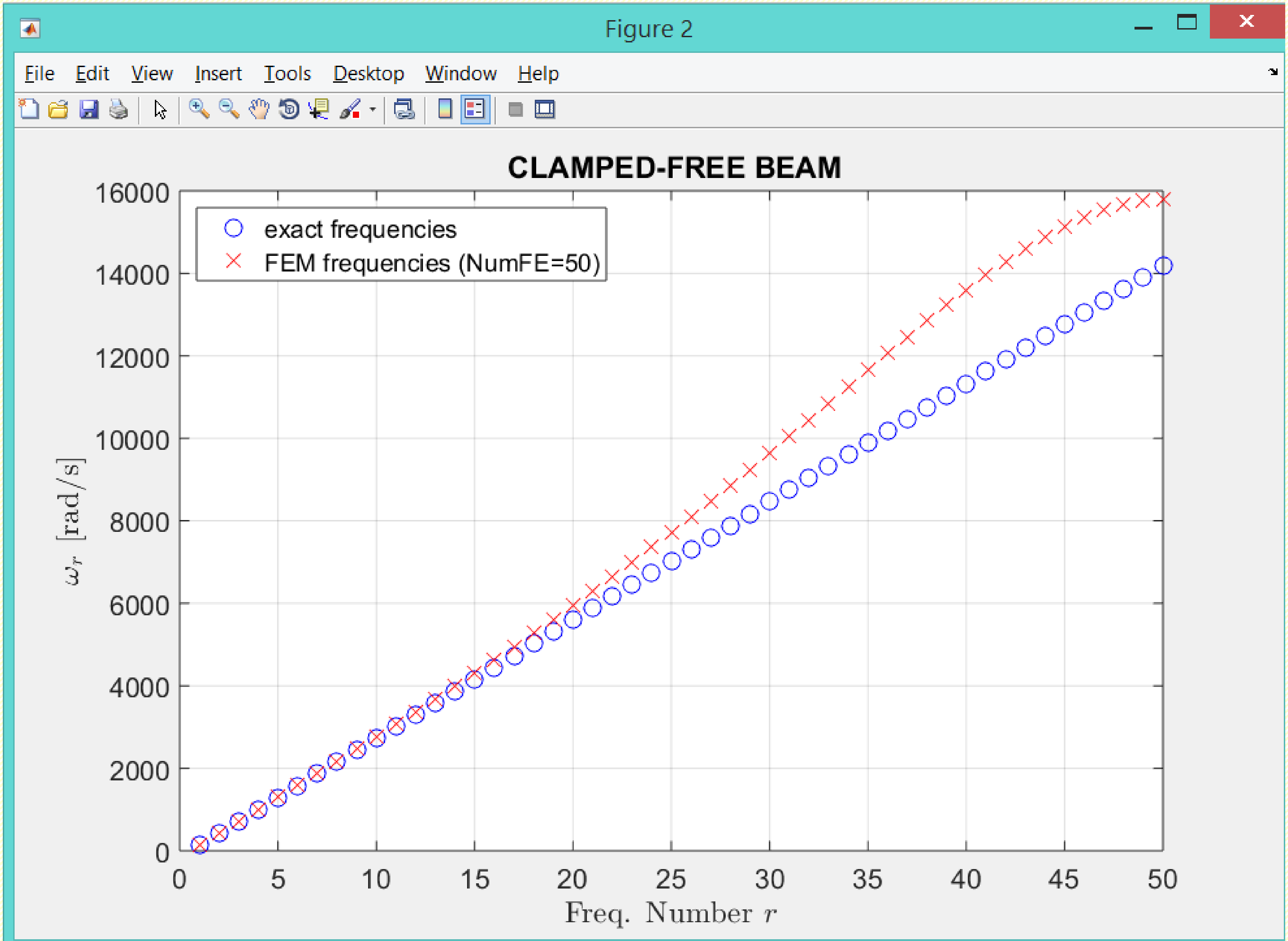
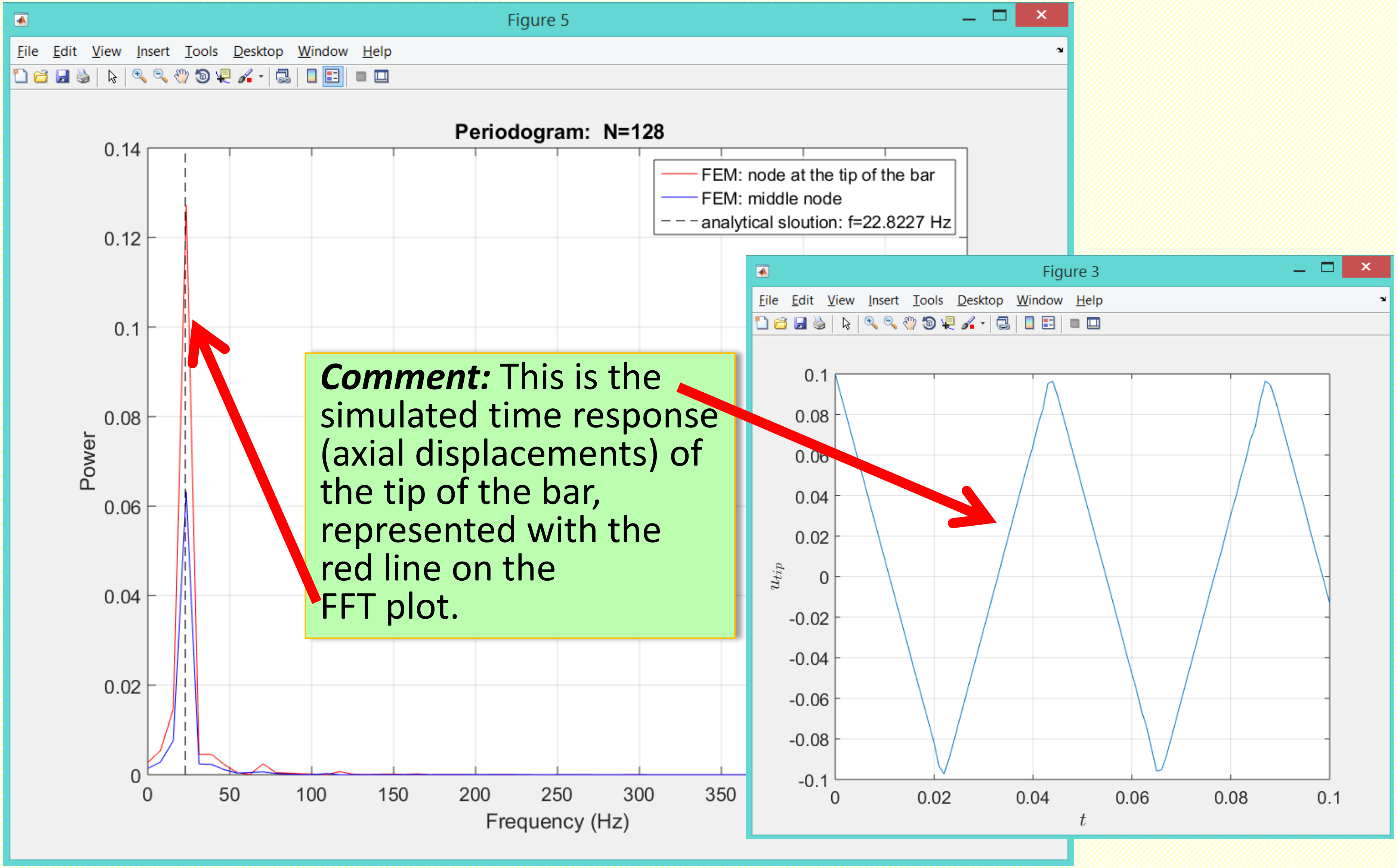


Figure 2
File Edit View Insert Tools Desktop Window Help
CLAMPED-FREE BEAM
exact frequencies
FEM frequencies (NumFE=50)
16000
14000
12000
10000
8000
6000
4000
2000
0
ω_r [rad/s]
0
5
10
15
20
25
30
35
40
45
50
Freq. Number r

EXAMPLE-2: FEM FFT RESULTS



Comment: This is the simulated time response (axial displacements) of the tip of the bar, represented with the red line on the FFT plot.

APPENDIX: Extracted Relevant Pages from my Text

REFERENCE: **Trivailo P.M.** (2008), Vibrations: Theory & Aerospace Applications, Vol.1&2, - (The textbook for senior undergraduate and graduate aerospace students). - Melbourne: RMIT Publisher - 2008. - 247pp +348pp=595 pp., 355 ill, 4 software programs.

RODS IN AXIAL VIBRATIONS

Partial Differential Equation
of the Longitudinal Vibration of Rods:

$$\frac{\partial}{\partial x}\left[E\mathcal{A}(x)\,\frac{\partial u(x,t)}{\partial x}\right] + f(x,t) = m(x)\,\frac{\partial^2 u(x,t)}{\partial t^2}, \quad 0 < x < L. \tag{8.1}$$

This equation represents the *partial differential equation of the longitudinal vibration of rods.*

Equation of motion and the boundary conditions of the problem constitute what is referred to as a *boundary-value problem.*

8.3 THE WAVE EQUATION

For the case of a uniform rod with constant axial stiffness, when $f(x,t) = 0$ Eq.(8.1) reduces to

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{E\mathcal{A}}{m}\frac{\partial^2 u(x,t)}{\partial x^2}, \qquad \text{or}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2}, \qquad \text{where} \tag{8.2}$$

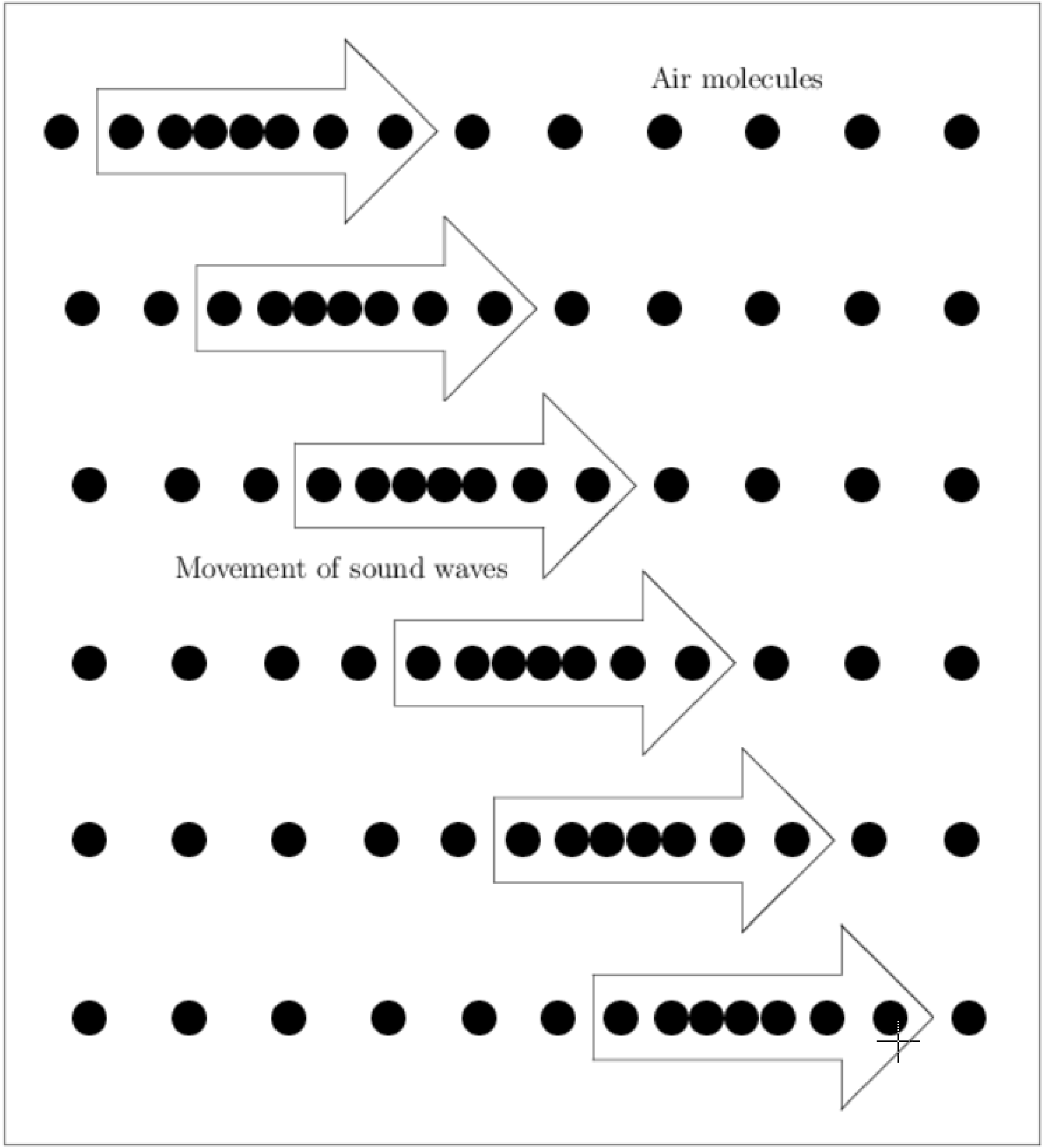


Figure 8.3: Analogy with a sound wave.

- It is easy to send longitudinal waves along a coiled spring: pull end of spring and out.

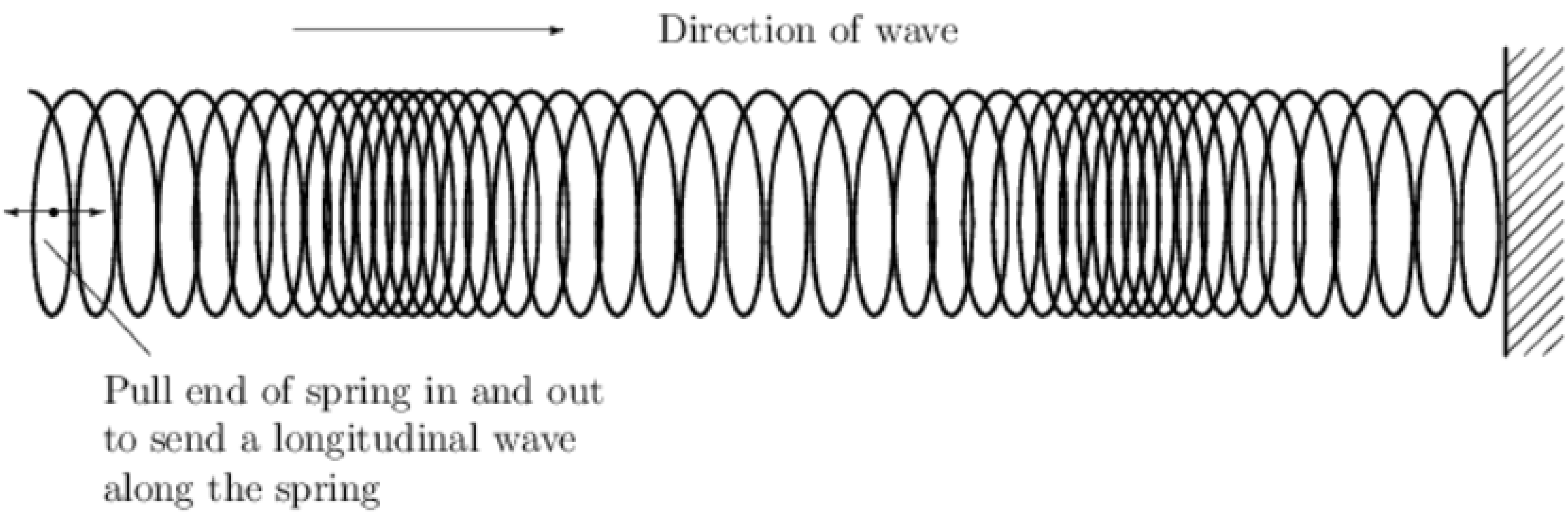


Figure 8.4: Analogy with transverse wave in a spring.

8.6 FREE VIBRATION. EIGENVALUE PROBLEM

Let us consider the vibrating rod. In the case of free vibration, namely, when the distributed force is zero, $f(x,t)=0$, the boundary-value problem reduces to the differential equation:

$$\frac{\partial}{\partial x}\left[E\mathcal{A}(x)\,\frac{\partial u(x,t)}{\partial x}\right] = m(x)\,\frac{\partial^2 u(x,t)}{\partial t^2}, \qquad 0 < x < L. \tag{8.6}$$

The solution of (8.6) is assumed in the form:

$$u(x,t) = U(x)\cdot F(t),$$

where $U(x)$ represents the general rod deformation and depends on the spatial variable x alone, and where $F(t)$ indicates the type of motion of rod configuration executes with time and depends on t alone.

$$\frac{1}{m(x)\,U(x)}\,\frac{d}{dx}\left[E\mathcal{A}(x)\,\frac{dU(x)}{dx}\right] = \frac{1}{F(t)}\,\frac{d^2F(t)}{dt^2}.$$

Since the left side of this equation is independent of t, whereas the right side is independent of x, it follows that each side must be a constant.

$$\frac{d^2F(t)}{dt^2} + \omega^2\,F(t) = 0, \tag{8.7}$$

$$-\frac{d}{dx}\left[E\mathcal{A}(x)\,\frac{dU(x)}{dx}\right] = \omega^2\,m(x)\,U(x), \qquad 0 < x < L. \tag{8.8}$$

The problem of determining the values of the parameter ω^2 for which nontrivial solutions $U(x)$ exist, where the solutions are subject to *boundary conditions*, is called the characteristic-value, or *eigenvalue problem* (from German *eigen*, characteristic).

Examples of boundary conditions are:

- $u(0,t) = u(L,t) = 0$ clamped-clamped rod
- $u'(0,t) = u'(L,t) = 0$ free-free rod

The differential equation (8.8) possesses space-dependent coefficients, so that in general no closed-form solution can be expected. A closed-form solution can be obtained in the special case of a *uniform rod* with $m(x) = m = const$, $E\mathcal{A}(x) = E\mathcal{A} = const$. Considering that case, Eq.(8.8) reduces to

$$\frac{d^2U(x)}{dx^2} + \beta^2\,U(x) = 0, \qquad \beta^2 = \omega^2\,\frac{m}{E\mathcal{A}} = \left(\frac{\omega}{c}\right)^2, \tag{8.9}$$

which must be satisfied over the domain $0 < x < L$.

The solution of the equation (8.9) has the form

$$U(x) = \left(A\,\sin\frac{\omega}{c}x + B\,\cos\frac{\omega}{c}x\right) \tag{8.10}$$

where the arbitrary constants A, B depend on the *boundary conditions*.

The general solution of the Eq.(8.7) is

$$F(t) = (C\,\sin\omega t + D\,\cos\omega t), \tag{8.11}$$

where the arbitrary constants C, D depend on the *initial conditions*.

Combining Eq.(8.10) and (8.11) we can write the general solution for $u(x,t)$ in the following form

$$u(x,t) = \left(A\,\sin\frac{\omega}{c}x + B\,\cos\frac{\omega}{c}x\right)\times(C\,\sin\omega t + D\,\cos\omega t). \tag{8.12}$$

8.7 TAKING INTO ACCOUNT BOUNDARY CONDITIONS

8.7.1 Example: Longitudinal Vibration of a Free-Free Uniform Rod

Solve the eigenvalue problem associated with a uniform rod, vibrating longitudinally, with both ends free (*see* Fig. 8.8).

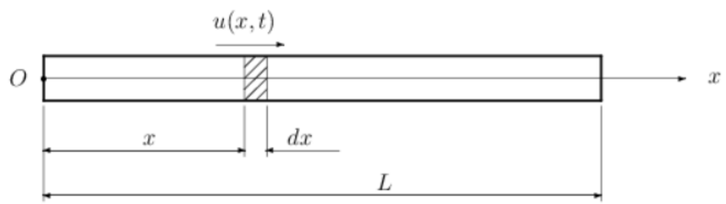


Figure 8.8: A free-free uniform rod in longitudinal vibration.

Solution

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2},$$

$$u(x,t) = \left(A\,\sin\frac{\omega}{c}x + B\,\cos\frac{\omega}{c}x\right) \times (C\,\sin\omega t + D\,\cos\omega t).$$

The arbitrary constants A, B, C, D depend on the *boundary conditions* and the *initial conditions*.

Since the bar has free ends, *the axial force*, which is proportional to dU/dx, *must be zero at each extremity*. Thus the boundary conditions for this problem may be written as

$$EA\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=0} = 0, \qquad EA\left.\frac{\partial u(x,t)}{\partial x}\right|_{x=L} = 0.$$

The first boundary condition will require that $A = 0$, so

$$u(x,t) = B\cos\frac{\omega}{c}x\,(C\,\sin\omega t + D\,\cos\omega t).$$

The second boundary condition then leads to the *characteristic equation*:

$$\sin\frac{\omega L}{c} = 0, \quad \text{or} \quad \frac{\omega_r L}{c} = r\pi, \quad r = 1, 2, 3, \ldots,$$

to which corresponds the infinite set of *eigenfunctions*:

$$U_r(x) = B_r\,\cos\frac{r\pi x}{L}.$$

The first natural modes are plotted in Figure 8.9, where the modes have been normalized by letting $B_r = 1$. We note that the first mode has one node, the second has two nodes and the third has three nodes. In general the r-th mode has r nodes ($r = 1, 2, \ldots$).

The system natural frequencies are:

$$\omega_r = \frac{r\pi c}{L} = r\pi\sqrt{\frac{E}{\rho L^2}}, \qquad r = 1, 2, 3, \ldots.$$

FREE-FREE ROD:

In the more general case of free vibration initiated in any manner, the solution will contain many of the normal modes:

$$u(x,t) = \sum_{r=1}^{\infty}\cos\frac{r\pi x}{L}\,(C_r\,\sin\omega_r t + D_r\,\cos\omega_r t)$$

$$\boxed{\omega_r = \frac{r\pi c}{L}, \quad r = 1, 2, 3, \ldots.}$$

The arbitrary constants C_r, D_r depend on the *initial conditions*.

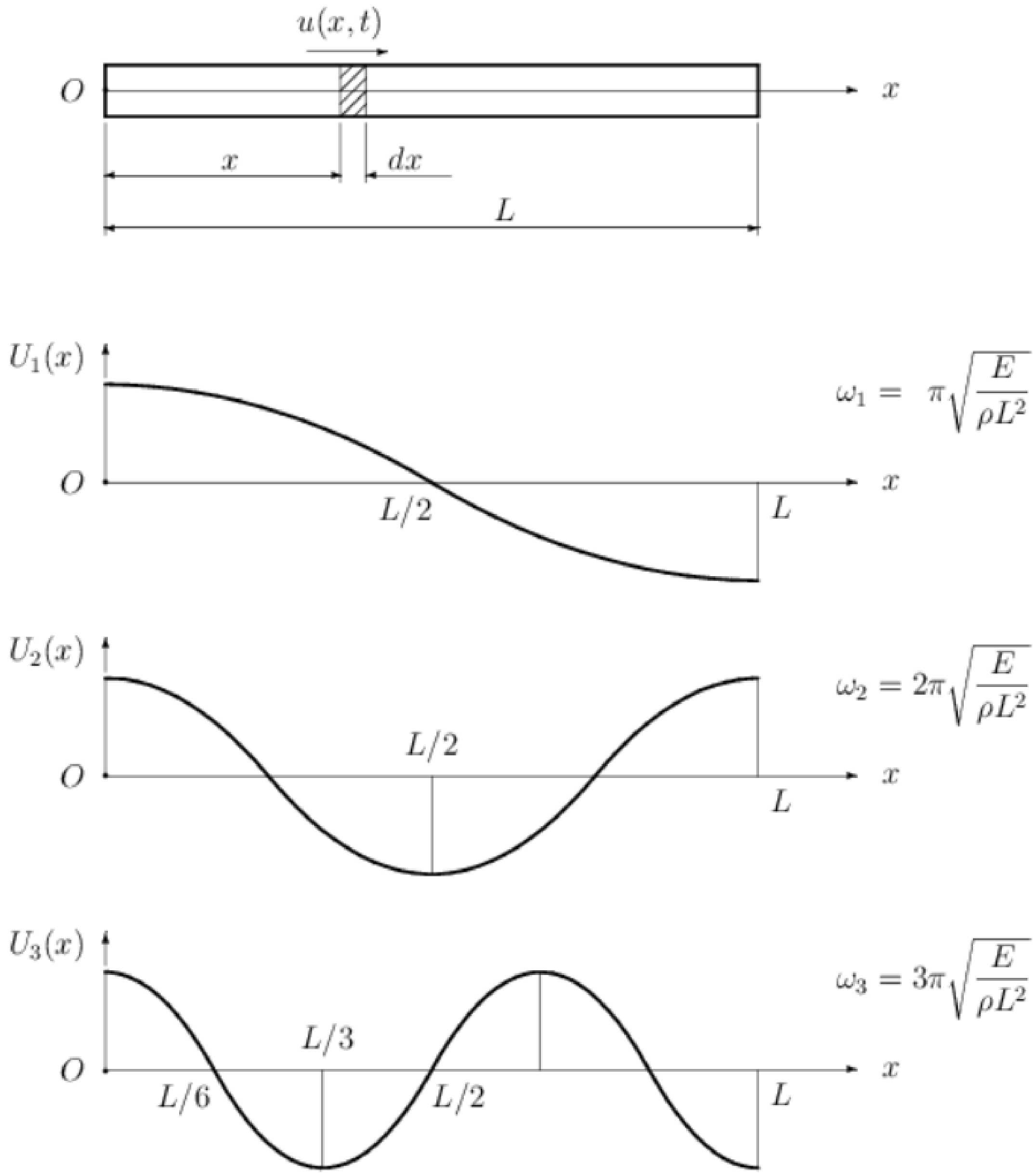


Figure 8.9: The first three natural modes of longitudinal vibration for a free-free bar.

8.7.2 Example: Axial Vibration of a Clamped-Free Uniform Rod

Derive an expression for the free longitudinal vibration of a uniform bar of length L, one end of which is fixed and the other end free (it see Fig. 8.10).

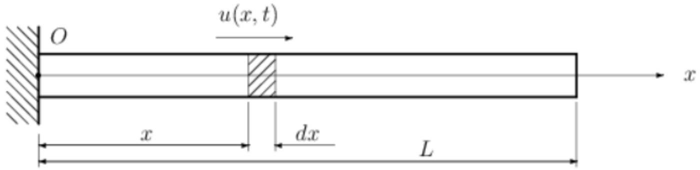


Figure 8.10: A clamped-free rod in longitudinal vibration.

Solution

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2},$$

The general solution for the free longitudinal vibration of uniform bars is given as

$$u(x,t) = \left(A\,\sin\frac{\omega}{c}x + B\,\cos\frac{\omega}{c}x\right) \times (C\,\sin\omega t + D\,\cos\omega t).$$

The arbitrary constants A, B, C, D depend on the *boundary conditions* and the *initial conditions*.

The *tensile force at the free end of this bar is equal to zero* while the *displacement of the fixed end of the bar is also equal to zero*, i.e. the boundary conditions of the problem are:

$$\begin{aligned} EA\,\frac{\partial u(x,t)}{\partial x}\bigg|_{x=L} &= 0, \\ u(x,t)\big|_{x=0} &= 0. \end{aligned}$$

The condition that $u(0,t) = 0$ will require that $B = 0$, so

$$u(x,t) = A\sin\frac{\omega}{c}x\,(C\,\sin\omega t + D\,\cos\omega t).$$

The condition $\frac{\partial u(L,t)}{\partial x} = 0$ then leads to the *characteristic equation*:

$$\cos\frac{\omega L}{c} = 0, \quad \text{or} \quad \frac{\omega_r L}{c} = \frac{(2r-1)\pi}{2}, \quad r = 1, 2, 3, \ldots,$$

to which corresponds the infinite set of *eigenfunctions*:

$$U_r(x) = A_r\,\sin\frac{(2r-1)\pi x}{2L}.$$

We note that the first mode, presented in Fig. 8.11, has no nodes, the second has one node and the third has two nodes. In general the r-th mode has $r-1$ nodes ($r = 1, 2, \ldots$).

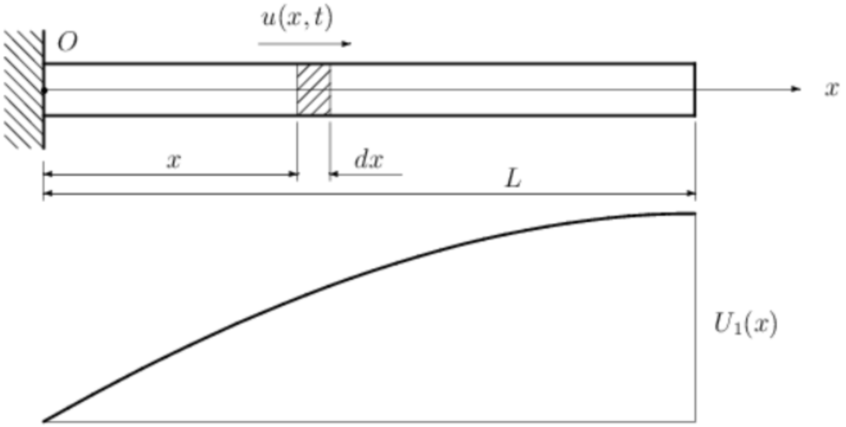


Figure 8.11: The fundamental mode of vibration of a clamped-free rod.

CLAMPED-FREE ROD:

The system natural frequencies are:

$$\boxed{\omega_r = \frac{(2r-1)\pi c}{2L} = \frac{(2r-1)\pi}{2}\sqrt{\frac{E}{\rho L^2}}, \qquad r = 1, 2, 3, \ldots .}$$

In the more general case of free vibration initiated in any manner, the solution will contain many of the normal modes:

$$\begin{aligned} u(x,t) &= \sum_{r=1}^{\infty} \sin\frac{(2r-1)\pi x}{2L}\left(C_r \sin\omega_r t + D_r \cos\omega_r t\right) \\ \omega_r &= \frac{(2r-1)\pi c}{2L}, \quad r = 1, 2, 3, \ldots . \end{aligned} \tag{8.13}$$

The arbitrary constants C_r, D_r depend on the *initial conditions.*

END OF LECTURE-3a SLIDES