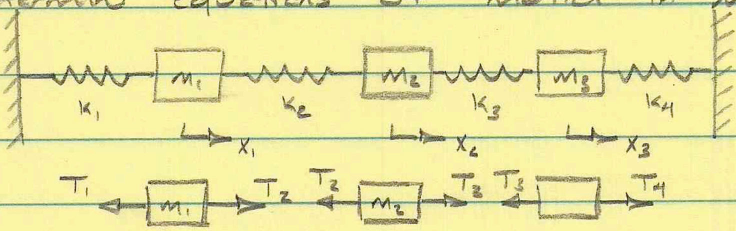
(20 min) 23. a. Three masses are constrained between a wall by 4 springs: determin equations of motion in matrix form.



Linear Momentum Balance for each mass:

① $\sum F_1 = m_1 \dot{v}_1$

$-k_1 x_1 + k_2(x_2 - x_1) = m_1 \ddot{x}_1$

② $-k_2(x_2 - x_1) + k_3(x_3 - x_2) = m_2 \ddot{x}_2$

③ $-k_3(x_3 - x_2) + k_4 x_3 = m_3 \ddot{x}_3$

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \ddot{\vec{x}} + \begin{bmatrix} k_1 + k_2 & -k_2 & 0 \\ -k_2 & k_2 + k_3 & -k_3 \\ 0 & -k_3 & k_3 + k_4 \end{bmatrix} \vec{x} = \vec{0} \rightarrow mI\ddot{\vec{x}} + \begin{bmatrix} 2k & -k & 0 \\ -k & 2k & -k \\ 0 & -k & 2k \end{bmatrix} \vec{x} = \vec{0}$$

b. Using intuition, one normal mode consists of the center mass starting at rest with no initial velocity, and the flanking masses moving away at equal speeds in either direction.

✓

$$\vec{u} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

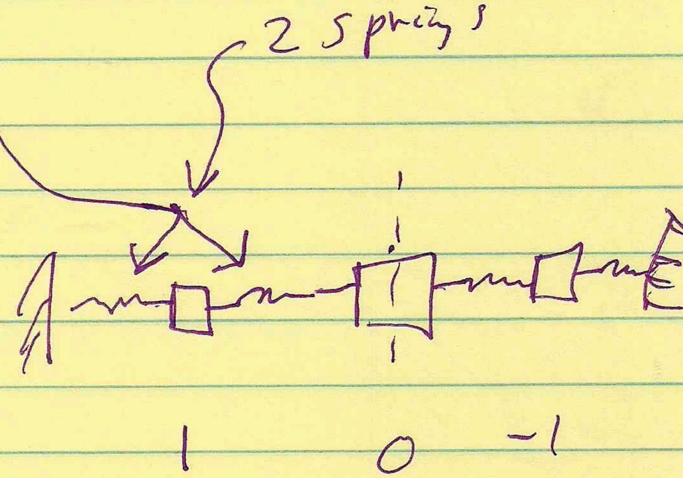
c. See attached MATLAB code.

$$D = \begin{bmatrix} 0.5 & -0.7071 & -0.5 \\ 0.7071 & 0 & 0.7071 \\ 0.5 & 0.7071 & -0.5 \end{bmatrix}$$

$$\lambda = \begin{bmatrix} 0.5858 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3.4142 \end{bmatrix}$$

thats the soln from prev. page.

d. See attached MATLAB code



Contents


```
function HW23 ()
```

Problem Statement

Three masses are in a line between two rigid walls, connected via four springs to one another and the walls.

Problem Setup

```
clear all
close all
clc

% Define parameters
m = 1;
k = 1;
```

Section (a)

```
% See attached
```

Section (b)

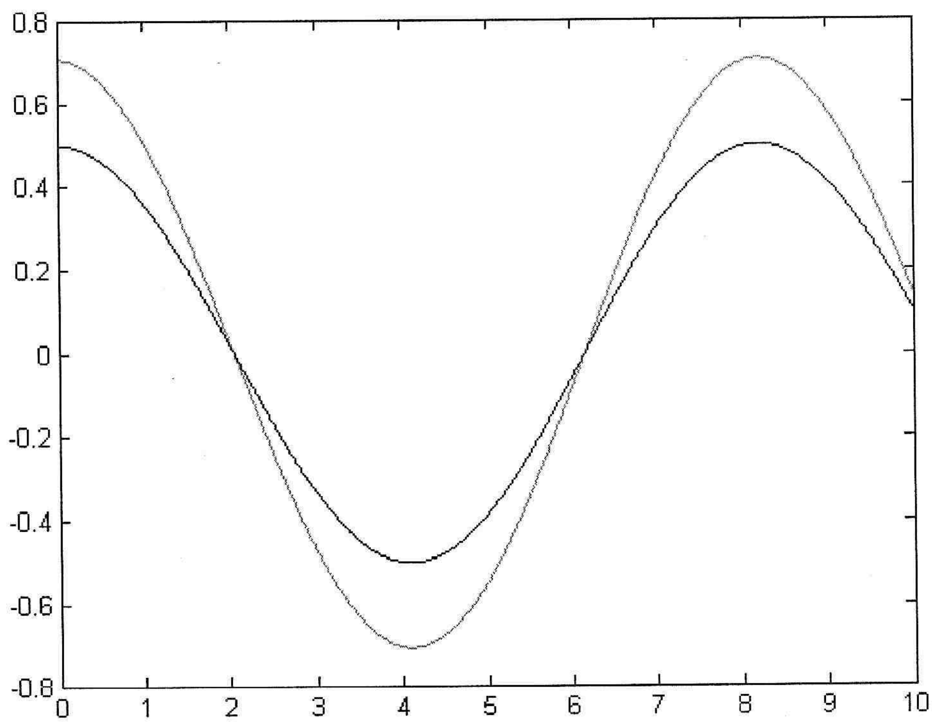
```
% See attached
```

Section (c)

```
% Using the eig function, determine the three normal modes of the system

M = [m    0    0;...       %Mass matrix
     0    m    0;...
     0    0    m];

K = [2*k  -k   0;...
     -k   2*k  -k;...
     0    -k   2*k];       %Stiffness matrix

[p, lambda] = eig(M^-1*K);

eVect1 = p(:,1);
lambda1 = lambda(1);

eVect2 = p(:,2);
lambda2 = lambda(2);

eVect3 = p(:,3);
lambda3 = lambda(3);
```

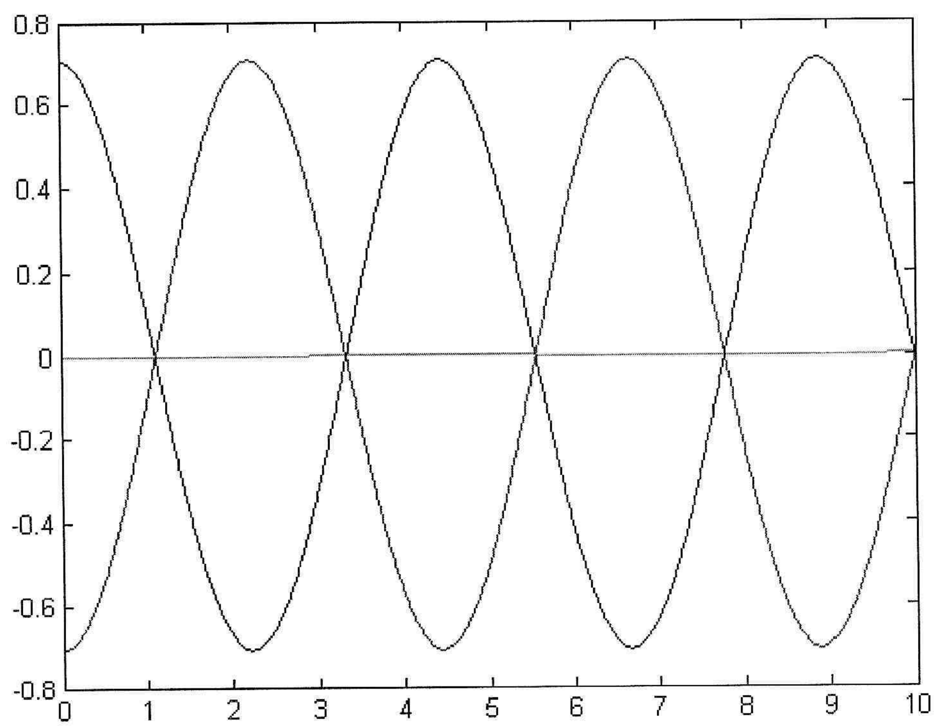
Section (d)

```
% Using numerical integration, with masses released from rest, show that
% normal modes produce synchronous oscillations.

% Initial conditions
x0 = [eVect1(1)      eVect1(2)       eVect1(3)]';
v0 = [0    0     0]';
zzero = [x0;v0];

n = 1000;
tspan = linspace(0,10,n);

% Using ODE45 solving method, calculate and display the solution to the
% normal modes of the system

[t zmatrix] = ode45(@method, tspan, zzero, [], M, K);

x1 = zmatrix(:,1);
x2 = zmatrix(:,2);
x3 = zmatrix(:,3);

plot(t,x1,'r',t,x2,'g',t,x3,'b')
```



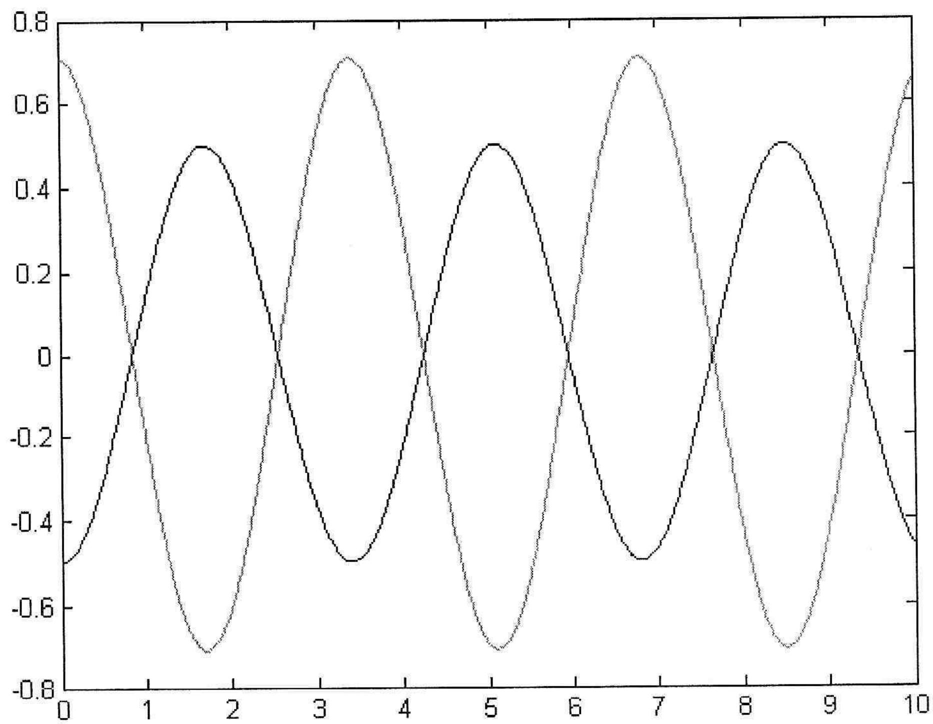

```
end

function zdot = method(t, z, M, K)
x = z(1:3);
v = z(4:6);

xdot = v;
vdot = -M^-1*K*x;

zdot = [xdot; vdot];
```

```
% Using numerical integration, with masses released from rest, show that
% normal modes produce synchronous oscillations.

% Initial conditions
x0 = [eVect2(1)      eVect2(2)      eVect2(3)]';
v0 = [0    0    0]';
zzero = [x0;v0];

n = 1000;
tspan = linspace(0,10,n);

% Using ODE45 solving method, calculate and display the solution to the
% normal modes of the system

[t zmatrix] = ode45(@method, tspan, zzero, [], M, K);

x1 = zmatrix(:,1);
x2 = zmatrix(:,2);
x3 = zmatrix(:,3);

plot(t,x1,'r',t,x2,'g',t,x3,'b')
```




```
end

function zdot = method(t, z, M, K)
x = z(1:3);
v = z(4:6);

xdot = v;
vdot = -M^-1*K*x;

zdot = [xdot; vdot];
```

```
% Using numerical integration, with masses released from rest, show that
% normal modes produce synchronous oscillations.

% Initial conditions
x0 = [eVect3(1)      eVect3(2)      eVect3(3)]';
v0 = [0    0    0]';
zzero = [x0;v0];

n = 1000;
tspan = linspace(0,10,n);

% Using ODE45 solving method, calculate and display the solution to the
% normal modes of the system

[t zmatrix] = ode45(@method, tspan, zzero, [], M, K);

x1 = zmatrix(:,1);
x2 = zmatrix(:,2);
x3 = zmatrix(:,3);

plot(t,x1,'r',t,x2,'g',t,x3,'b')
```




```
end

function zdot = method(t, z, M, K)
x = z(1:3);
v = z(4:6);

xdot = v;
vdot = -M^-1*K*x;

zdot = [xdot; vdot];
```