

---

# Homework 04 - Steve Mazza

## Table of Contents

Problem 1 .....	1
Problem 2 .....	2
Problem 3 .....	3
Problem 4 .....	5

## Problem 1

```
close all;
clc;
clear all;

syms x y;
p_fun = 3*x*sin(x)-2*x;

% Plot the function
fplot(char(p_fun),[-2*pi 2*pi]);
hold on;

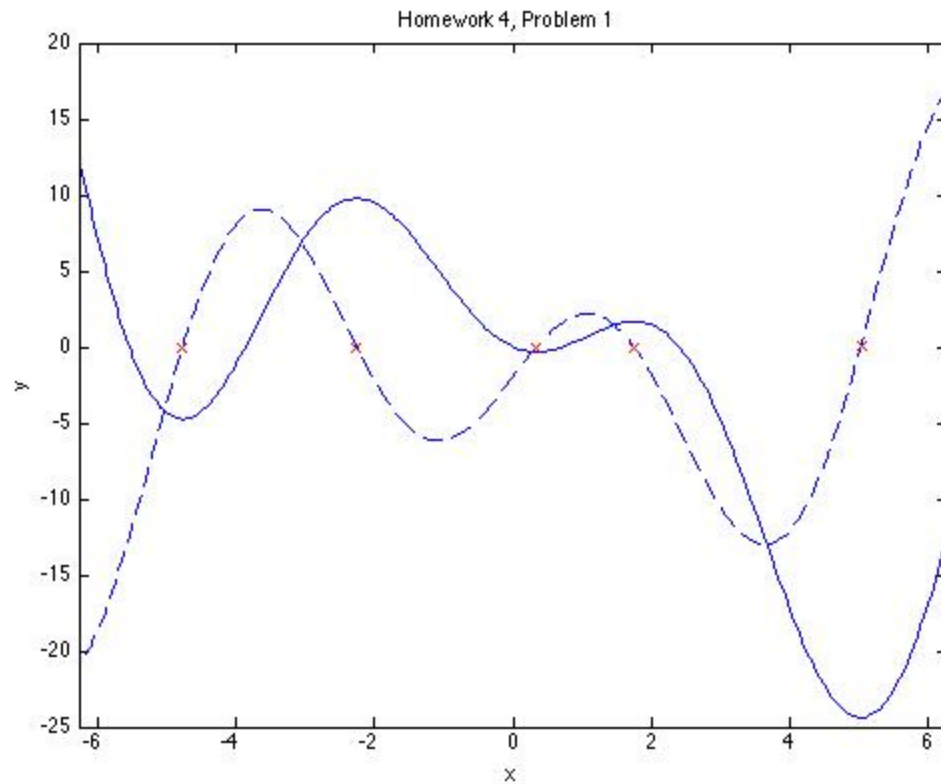
% Compute the derivative by hand
pd_fun = diff(p_fun);

% Add the derivative to the same plot
fplot(char(pd_fun),[-2*pi 2*pi], '--');

% Provide title, axis labels, and legend
title('Homework 4, Problem 1');
xlabel('x');
ylabel('y');

% Use fzero() to find all zeros of the drivative at -4, -2, 0, 2, 4.
f_zero = zeros(1,5);
f_zero(1) = fzero(char(pd_fun),-4);
f_zero(2) = fzero(char(pd_fun),-2);
f_zero(3) = fzero(char(pd_fun),0);
f_zero(4) = fzero(char(pd_fun),2);
f_zero(5) = fzero(char(pd_fun),4);

% Add zeros to the plot as markers
for i=1:5
    x_ = f_zero(i);
    y_ = subs(pd_fun,x,x_);
    plot(x_,y_, 'rx');
end
```



## Problem 2

```

close all;
clc;
clear all;

% NOTE: We derive an equation for distance as a function of time using the
% Pythagorean theorem. Assuming the ships travel along the x- and y- axes
% simplifies our equation to:
% d = sqrt((14-8*t)^2 + (-25+16*t)^2)
%

% =====
% User configurable variables (feel free to play)
t_0 = 0;           % start time
t_n = 4;           % end time
vis = 8;           % visibility distance
speed_a = -8;      % speed in mph and direction of ship a
start_a = 14;      % initial location of ship a on the y-axis
speed_b = 16;      % speed in mph and direction of ship b
start_b = -25;     % initial location of ship b on the x-axis
% Let's not change anything below this line.
% =====

syms t;

```

```

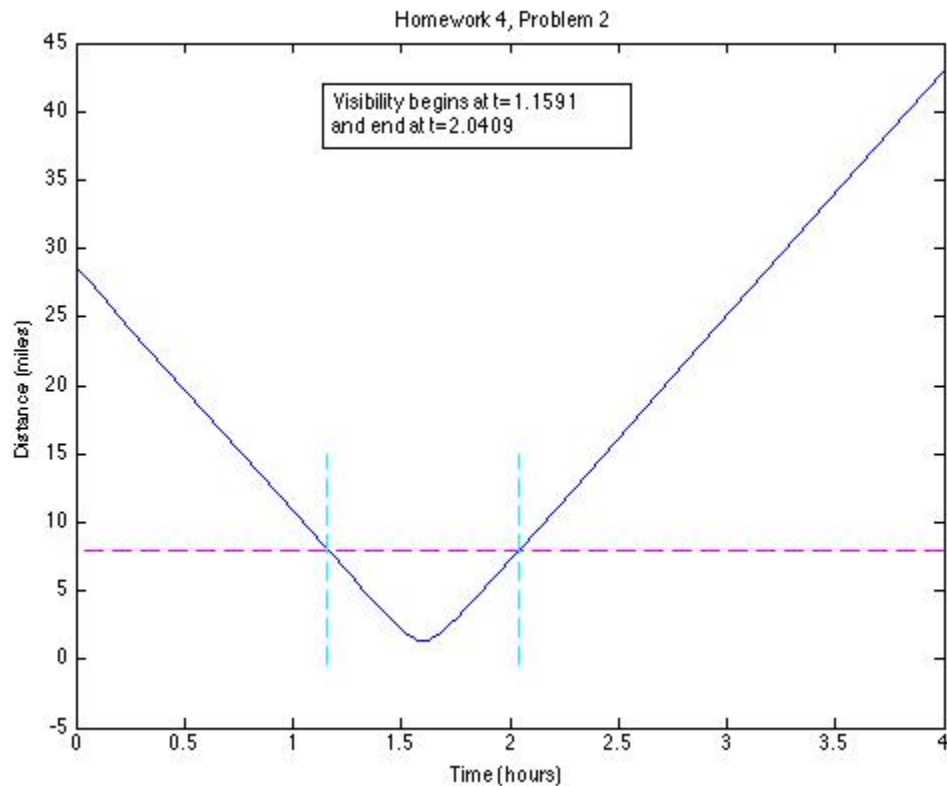
d = sqrt((start_a+speed_a*t)^2 + (start_b+speed_b*t)^2);

% Plot d = f(t) for t=0:4.
fplot(char(d),[t_0 t_n]);

% Calculate t = f(d) for d=8.
v_t = solve(strcat(num2str(vis),strcat('==',char(d))));
v_start_t = eval(char(v_t(2)));
v_end_t = eval(char(v_t(1)));

% Add labels, lines, and annotations.
title('Homework 4, Problem 2');
xlabel('Time (hours)');
ylabel('Distance (miles)');
line([t_0 t_n], [vis vis],'LineStyle','--','Color',[1 0 1]);
line([v_start_t v_start_t], [-1 15],'LineStyle','--','Color',[0 1 1]);
line([v_end_t v_end_t], [-1 15],'LineStyle','--','Color',[0 1 1]);
message_ = strcat(strcat(strcat('Visibility begins at t=',...
    num2str(v_start_t)), ' and end at t=',num2str(v_end_t)));
annotation('textbox',[.35 .8 .275 .075],'String',message_);

```



## Problem 3

```

close all;
clc;

```

```
clear all;

% User defined variables.
R = 1.5;          % radius of the staircase in meters
n = 3;           % number of revolutions per floor
h = 3;           % height of each floor in meters
tbar = 0:0.01:1;% We are showing only 1 floor.  Can change to 2.

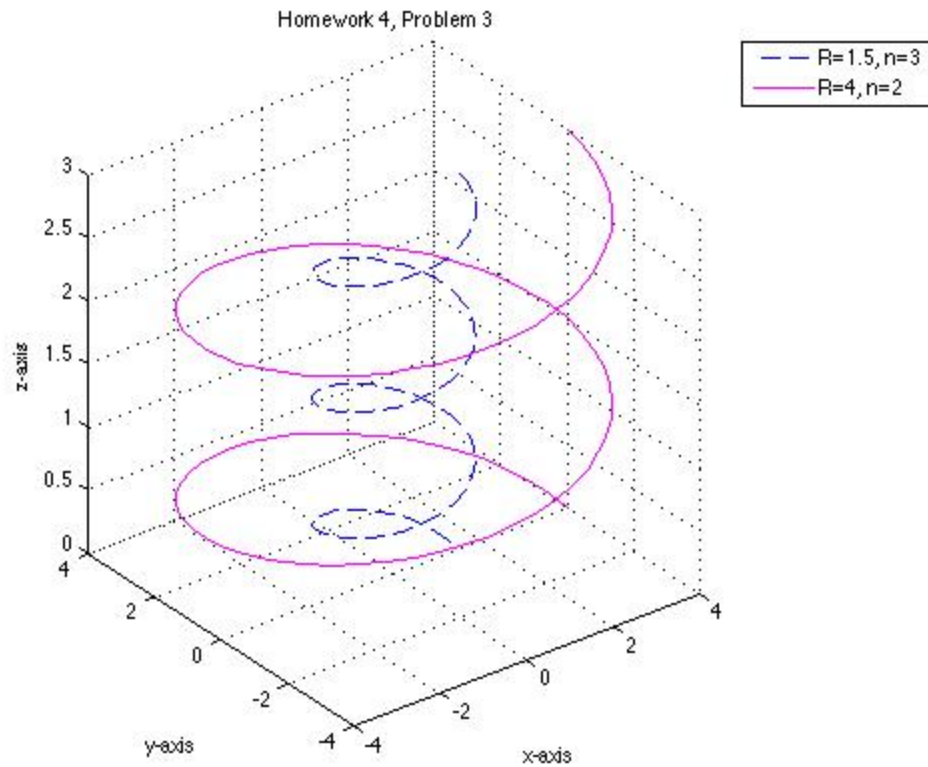
% Parametric equations.
syms x y z;
x = R*cos(2*pi*n*tbar);
y = R*sin(2*pi*n*tbar);
z = tbar*h;

% Plot the first staircase.
plot3(x,y,z,'LineStyle','--');
hold on;
grid on, axis square;

% Now change values and re-calculate.
R = 4;
n = 2;
x = R*cos(2*pi*n*tbar);
y = R*sin(2*pi*n*tbar);

% Plot the second staircase.
plot3(x,y,z,'Color',[1 0 1]);

% Add labels and legends.
title('Homework 4, Problem 3');
xlabel('x-axis');
ylabel('y-axis');
zlabel('z-axis');
legend('R=1.5, n=3', 'R=4, n=2');
```



## Problem 4

```
close all;
clc;
clear all;

freq = 3.8316;

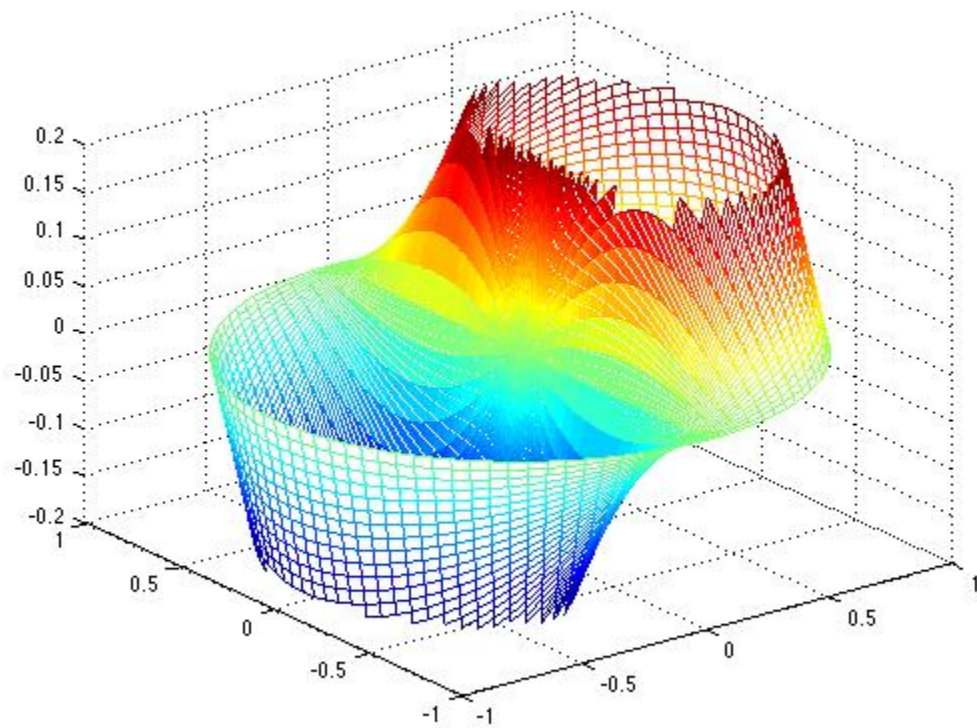
% Hundred-element vectors (101, actually).
r = 0:0.01:1;
phi = 0:pi/50:2*pi;

% Compute the values of x, y, and z.
for i = 1:101
    for j = 1:101
        x(i,j) = r(i)*cos(phi(j));
        y(i,j) = r(i)*sin(phi(j));
        z(i,j) = besselj(1,freq*r(i))*cos(phi(j));
    end
end

% Selectively remove some values of z.
z(find(abs(z) > 0.2)) = NaN;

% Plot the membrane using mesh().
```

```
mesh(x,y,z);
```



*Published with MATLAB® R2013a*