
Assignment 4

Table of Contents

Problem 1: Fixed Point Iteration	1
Problem 2: Newton Rhapson	3
Problem 3	3
Problem 4	4

Problem 1: Fixed Point Iteration

Part a done on paper Part b

```
clc; clear all; close all;
l1 = 7; l2 = 1.8; l3 = 4.2; l4 = 4.8;
p1 = l1/l4; p2 = l1/l2; p3 = (l1^2 + l2^2 + l4^2 - l3^2)/(2*l2*l4);
theta2 = 45; xi = 100; ea_goal = 0.1; ea = 10000; i = 0;
fpi = @(x) acosd((p1*cosd(theta2) - p3 + cosd(theta2 - x))/p2);

while ea > ea_goal
    i = i + 1;
    xil = fpi(xi);
    ea = abs((xil - xi)/xil);
    xi = xil;
end
theta4a = xi;
theta4 = xi;
fprintf('Using fixed point iteration: \n');
fprintf('After %d iterations, theta 4 is %f\n', i, theta4a);

% Part c
x = [0, l2*cosd(theta2), l1 + l4*cosd(theta4), l1, 0];
y = [0, l2*sind(theta2), l4*sind(theta4), 0, 0];
% First figure is for part c
figure(1);
plot(x, y, '-ob');

% Part d: Add a for loop
for theta2 = 0:10:360
    ea = 10000;
    fpi = @(x) acosd((p1*cosd(theta2) - p3 + cosd(theta2 - x))/p2);
    while ea > ea_goal
        i = i + 1;
        xil = fpi(xi);
        ea = abs((xil - xi)/xil);
        xi = xil;
    end
    theta4 = xi;
```

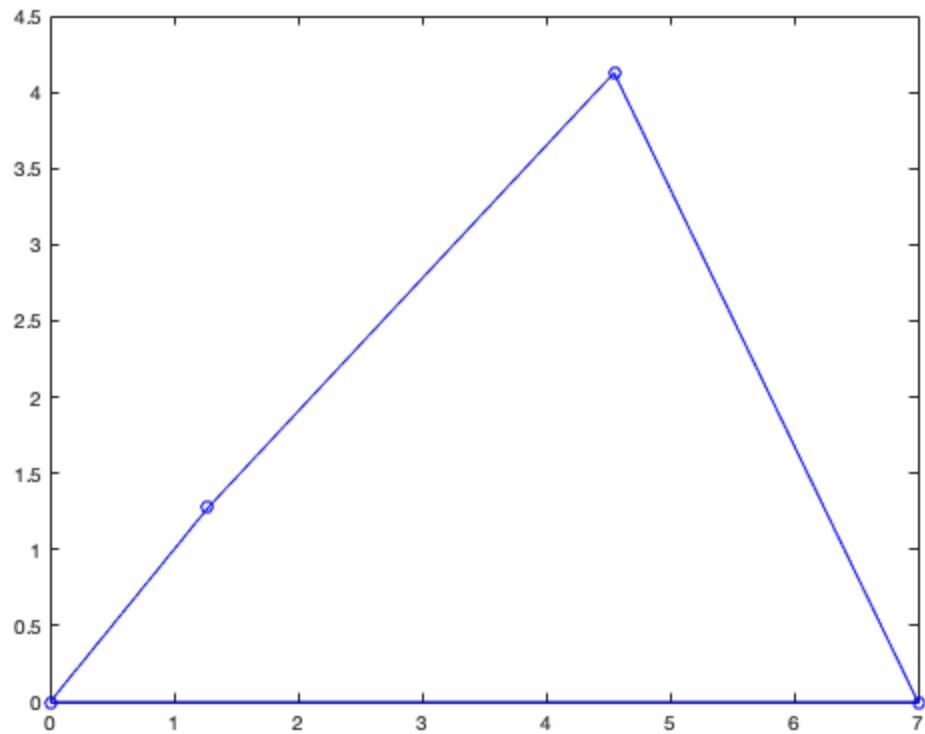
```

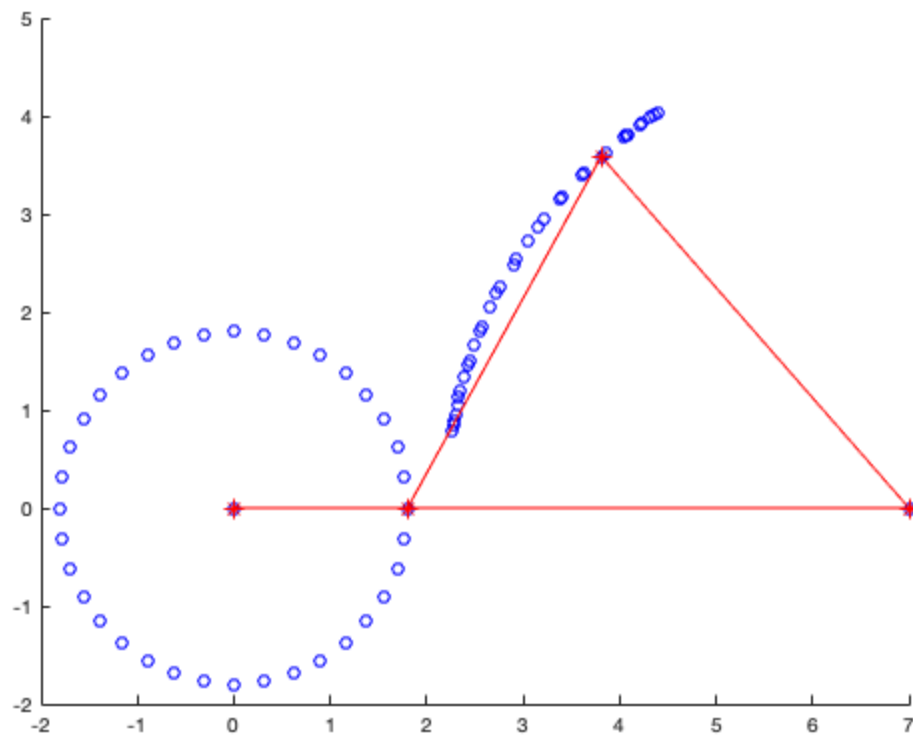
x = [0, 12*cosd(theta2), 11 + 14*cosd(theta4), 11, 0];
y = [0, 12*sind(theta2), 14*sind(theta4), 0, 0];
figure(2);
scatter(x, y, 'ob');
hold on
end
plot(x, y, '-*r');
% Second figure is for part d

```

Using fixed point iteration:

After 2 iterations, theta 4 is 120.708904





Problem 2: Newton Raphson

```
close all;
theta2 = 45; ea = 10000; i = 0; xi = theta4a;
nri = @(x) cosd(theta2 - x) + p1*cosd(theta2) - p3 -p2*cosd(x);
nril = @(x) sind(theta2 - x) + p2*sind(x);

while ea > ea_goal
    i = i + 1;
    xil = xi - nri(xi)/nril(xi);
    ea = abs((xil - xi)/xil);
    xi = xil;
end
theta4 = xil;
fprintf('Using Newton-Raphson Method: \n');
fprintf('After %d iterations, theta 4 is %f\n', i, theta4);
```

*Using Newton-Raphson Method:
After 1 iterations, theta 4 is 120.739074*

Problem 3

Work done on paper

Problem 4

```
clc; close all; clear all;
max = -10000000; xmax = -100000; ymax = -1000000; z = -1000000;
for i = 1:1000
    x = -2 + 4*rand;
    y = -2 + 4*rand;
    z = opt(x, y);
    if z > max
        max = z;
        xmax = x;
        ymax = y;
    end
end

fprintf('\nUsing 2D Optimization: \n');
fprintf('The x value is: %f\n The y value is: %f\n The optimum value\n\n is: %f\n', xmax, ymax, max);

function z = opt(x, y)
z = sin(4*x + 3*y + 2*x^2 - x^4 - 3*x*y - 2*y^2);
end
```

```
Using 2D Optimization:
The x value is: 0.551037
The y value is: 1.166155
The optimum value is: 1.000000
```

Published with MATLAB® R2020a