Table of Contents

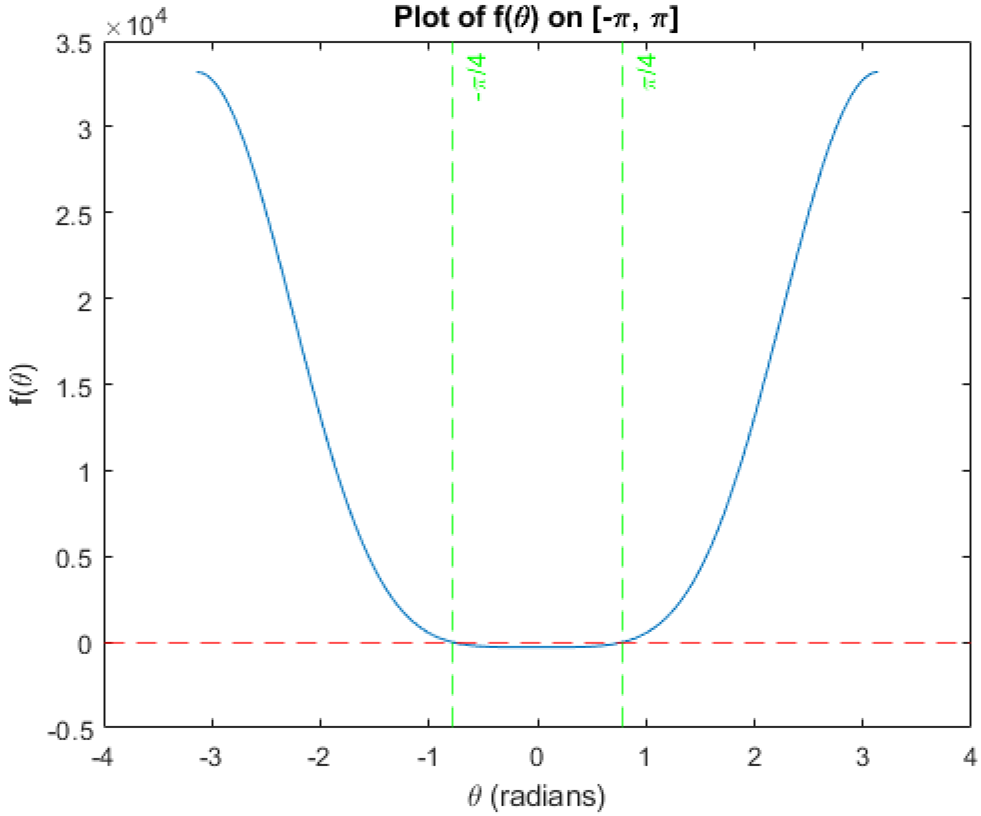

```
clear;
```

QUESTION 1:

```
% Function is at the bottom in the supporting code section
% function out = f(theta)

% Testing theta = pi/4
f(pi/4)

% Testing theta = -pi/4
f(-pi/4)

% Both are close to 0, so we are good
```

QUESTION 2:

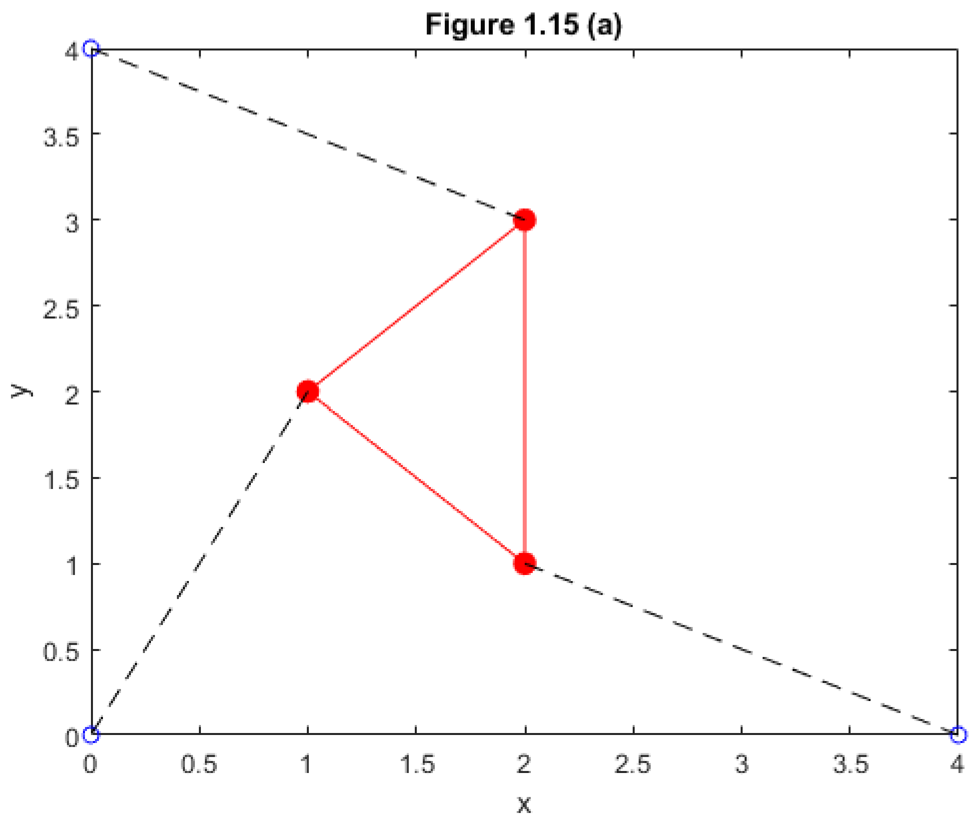
```
% Plotting f(theta) on [-pi, pi]
theta_vals = -pi:0.01:pi;

f_vals = f(theta_vals);

figure(1)
plot(theta_vals, f_vals)
xlabel('\theta (radians)')
ylabel('f(\theta)')
title('Plot of f(\theta) on [-\pi, \pi]')
yline(0, '--r');
xline(pi/4, '--g', '\pi/4');
xline(-pi/4, '--g', '-\pi/4');
drawnow;

% Plot clearly shows that there are roots at +/- pi/4
```

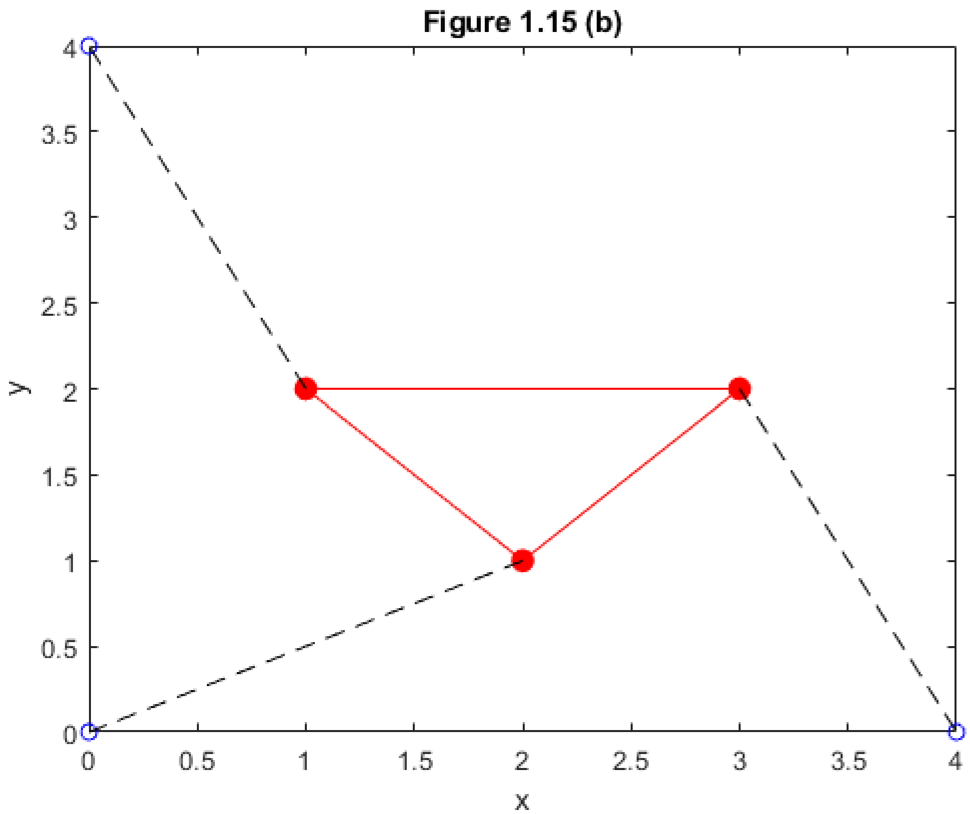


QUESTION 3:

```
% Pose from Figure 1.15 (a)
u1 = 1; u2 = 2; u3 = 2;
v1 = 2; v2 = 1; v3 = 3;
x1 = 4; x2 = 0; y2 = 4;

figure(2)
plot([u1 u2 u3 u1], [v1 v2 v3 v1], 'r'); hold on          % Platform triangle
plot([0 x1 x2], [0 0 y2], 'bo')                           % Base anchors
plot([u1 u2 u3], [v1 v2 v3], 'ro', 'MarkerSize', 8, 'MarkerFaceColor', 'r')
% Platform joints
plot([u1 0], [v1 0], 'k--')     % Strut 1
plot([u2 x1], [v2 0], 'k--')    % Strut 2
plot([u3 x2], [v3 y2], 'k--')   % Strut 3
title('Figure 1.15 (a)')
xlabel('x')
ylabel('y')
drawnow;

% Pose from Figure 1.15 (b)
u1 = 2; u2 = 3; u3 = 1;
v1 = 1; v2 = 2; v3 = 2;
x1 = 4; x2 = 0; y2 = 4;
```

```
figure(3)
plot([u1 u2 u3 u1], [v1 v2 v3 v1], 'r'); hold on          % Platform triangle
plot([0 x1 x2], [0 0 y2], 'bo')                          % Base anchors
plot([u1 u2 u3], [v1 v2 v3], 'ro', 'MarkerSize', 8, 'MarkerFaceColor', 'r')
% Platform joints
plot([u1 0], [v1 0], 'k--')     % Strut 1
plot([u2 x1], [v2 0], 'k--')    % Strut 2
plot([u3 x2], [v3 y2], 'k--')   % Strut 3
title('Figure 1.15 (b)')
xlabel('x')
ylabel('y')
drawnow;

% Here, we're just reproducing Figure 1.15 (a) and (b)
```

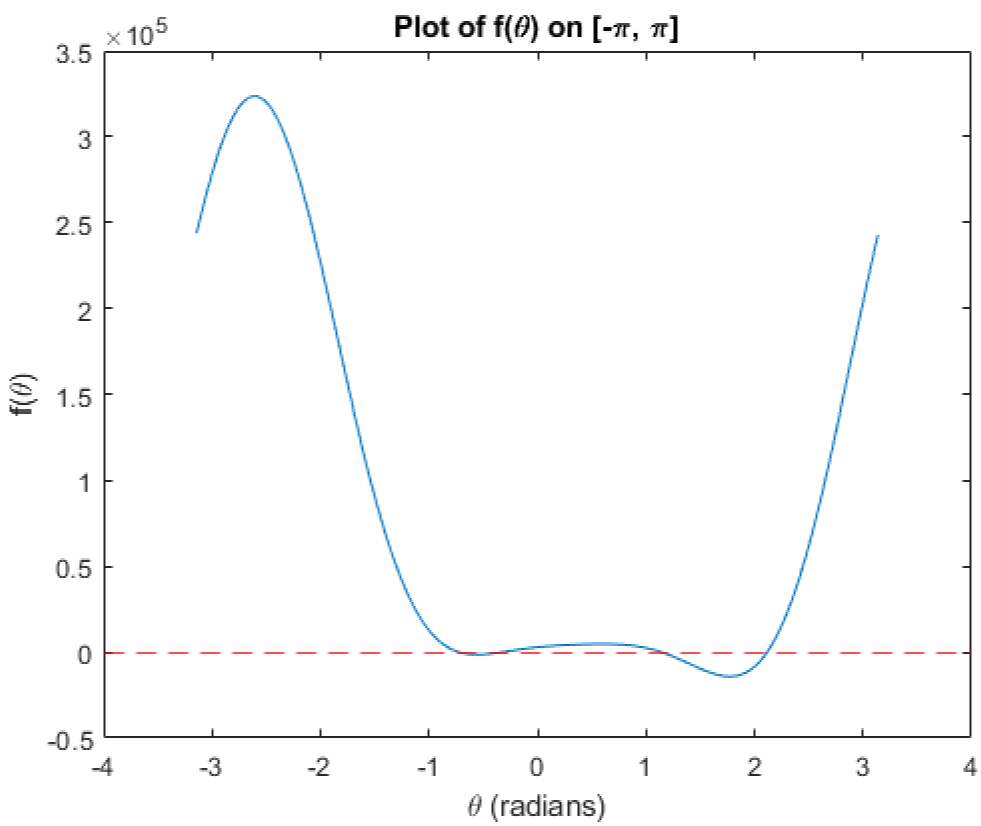


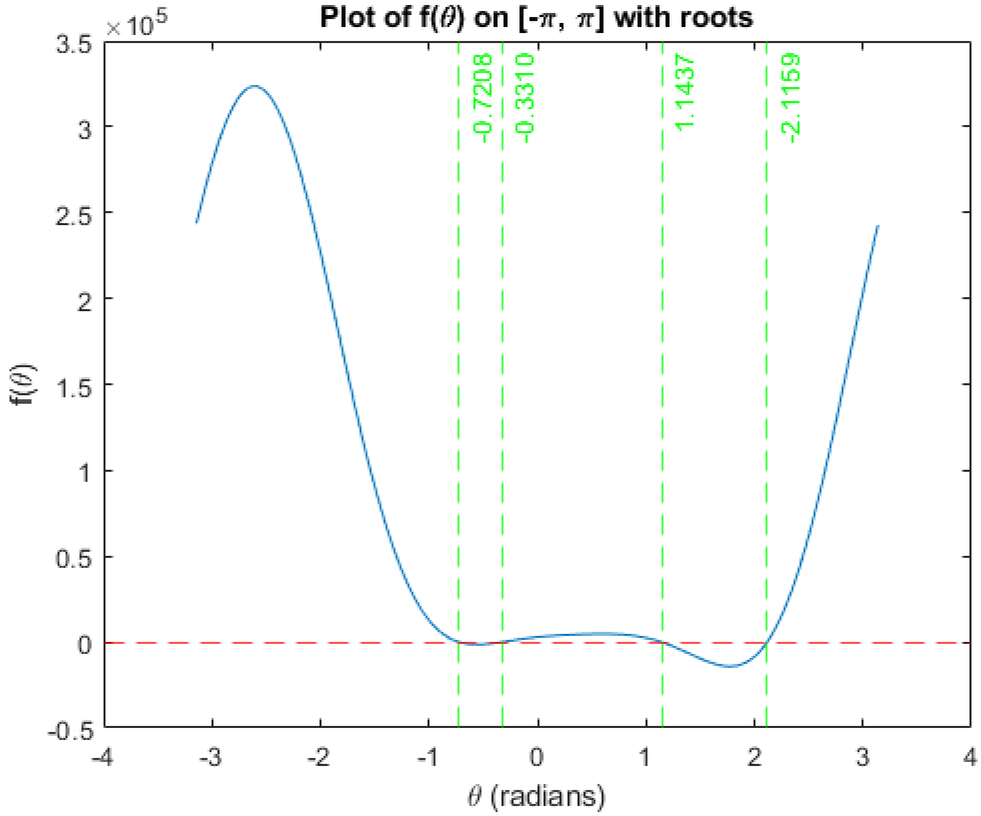


QUESTION 4:

```
% Forward kinematics is when we compute (x, y) and theta for each given p1,
% p2, and p3

% The inverse kinematic problem is when we find p1, p2, p3, given x, y, and
% theta

% The new f(theta) function is in the supporting functions section at the
% bottom (f_4(theta))

% Plotting f_4(theta) on [-pi, pi]
theta_vals = -pi:0.01:pi;

f_vals = f_4(theta_vals);

figure(4)
plot(theta_vals, f_vals)
xlabel('\theta (radians)')
ylabel('f(\theta)')
title('Plot of f(\theta) on [-\pi, \pi]')
yline(0, '--r');
% xline(pi/4, '--g', '\pi/4');
% xline(-pi/4, '--g', '-\pi/4');
drawnow;
```

```
% Finding the four theta values (guesses are from eyeballing the graph)
root1 = fzero(@f_4, -0.72);
root2 = fzero(@f_4, -0.33);
root3 = fzero(@f_4, 1.14);
root4 = fzero(@f_4, 2.11);

% From the above it appears that our roots are at:
% theta = -0.7208, -0.3310, 1.1437, and 2.1159 radians
figure(5)
plot(theta_vals, f_vals)
xlabel('\theta (radians)')
ylabel('f(\theta)')
title('Plot of f(\theta) on [-\pi, \pi] with roots')
yline(0, '--r');
xline(-0.7208, '--g', '-0.7208');
xline(-0.3310, '--g', '-0.3310');
xline(1.1437, '--g', '1.1437');
xline(2.1159, '--g', '-2.1159');
drawnow;
```





QUESTION 5:

QUESTION 6:

QUESTION 7:

QUESTION 8:

ALL FUNCTIONS SUPPORTING THIS CODE

```
% First f(theta) function
function out = f(theta)

    % Platform lengths
    L1 = 2;
    L2 = sqrt(2);
    L3 = sqrt(2);

    % Angle across from L1
    gamma = pi / 2;

    % Strut lengths
```

```
    p1 = sqrt(5);
    p2 = sqrt(5);
    p3 = sqrt(5);

    % Strut base positions
    % Got these from Figure 1.15
    x1 = 4;
    x2 = 0;
    y2 = 4;

    A2 = L3 * cos(theta) - x1;
    B2 = L3 * sin(theta);
    A3 = L2 * (cos(theta) * cos(gamma) - sin(theta) * sin(gamma)) - x2;
    B3 = L2 * (cos(theta) * sin(gamma) + sin(theta) * cos(gamma)) - y2;

    N1 = B3 .* (p2^2 - p1^2- A2.^2 - B2.^2) - B2 .* (p3^2 - p1^2 - A3.^2 -
B3.^2);
    N2 = -A3 .* (p2^2 - p1^2- A2.^2 - B2.^2) + A2 .* (p3^2 - p1^2 - A3.^2 -
B3.^2);

    D = 2 * (A2 .* B3 - B2 .* A3);

    out = N1.^2 + N2.^2 - p1.^2 * D.^2;

end

% f(theta) function for part 4
function out = f_4(theta)

    % Platform lengths
    L1 = 3;
    L2 = 3 * sqrt(2);
    L3 = 3;

    % Angle across from L1
    gamma = pi / 4;

    % Strut lengths
    p1 = 5;
    p2 = 5;
    p3 = 3;

    % Strut base positions
    % Got these from Figure 1.15
    x1 = 5;
    x2 = 0;
    y2 = 6;

    A2 = L3 * cos(theta) - x1;
    B2 = L3 * sin(theta);
    A3 = L2 * (cos(theta) * cos(gamma) - sin(theta) * sin(gamma)) - x2;
    B3 = L2 * (cos(theta) * sin(gamma) + sin(theta) * cos(gamma)) - y2;

    N1 = B3 .* (p2^2 - p1^2- A2.^2 - B2.^2) - B2 .* (p3^2 - p1^2 - A3.^2 -
```

```
B3.^2);
    N2 = -A3 .* (p2^2 - p1^2- A2.^2 - B2.^2) + A2 .* (p3^2 - p1^2 - A3.^2 -
B3.^2);

    D = 2 * (A2 .* B3 - B2 .* A3);

    out = N1.^2 + N2.^2 - p1.^2 * D.^2;

end
```

```
ans =

  -4.5475e-13


ans =

  -4.5475e-13
```

Published with MATLAB® R2024b