Table of Contents

QUESTION 6:

```
% Now we need to find a strut length p2, for which there are only two
% poses. It is found that when p2=4,  there are only two poses

theta_vals = -pi:0.01:pi;
p2_range = 1:7;  % Test p2 values between 1 and 7
target_num_roots = 2;
found = false;

for p2 = p2_range
    f_vals = f_variable_p2(theta_vals, p2);
    sign_changes = sum(abs(diff(sign(f_vals))) == 2);  % # of times f(theta)
crosses 0
    if sign_changes == target_num_roots
        fprintf("Found p2 = %.2f with exactly %d poses\n", p2,
target_num_roots);
        found = true;
        break
    end
end

if ~found
    fprintf("No p2 in range [%0.2f, %0.2f] gives exactly %d poses\n", ...
        p2_range(1), p2_range(end), target_num_roots);
end

% Here we are changing p2 to 4

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

f_vals = f_variable_p2(theta_vals, 4); % p2=4

figure(18)
plot(theta_vals, f_vals)
xlabel('\theta (radians)')
ylabel('f(\theta)')
title('Plot of f(\theta) on [-\pi, \pi] for Question #6')
yline(0, '--r');
drawnow;

% Finding the six theta values (guesses are from eyeballing the graph)
p2 = 4;
f_p2 = @(theta) f_variable_p2(theta, p2);
```

```
theta1 = fzero(f_p2, 1.32);
theta2 = fzero(f_p2, 1.77);

thetas = [theta1 theta2];

% theta vals are 1.3316 and 1.7775 rad

% Since we're asked to solve the forward kinematics problem, we need to
% solve for x and y now (we just solved for theta)

% Finding the x and y coordinates for the four poses
[x_1 y_1] = forward_kinematics_variable_p2(theta1, p2);
[x_2 y_2] = forward_kinematics_variable_p2(theta2, p2);

xs = [x_1 x_2];
ys = [y_1 y_2];

% It was found that
% (x_1, y_1) = (4.8907, 1.0399)
% (x_2, y_2) = (4.8992, 0.9992)

% Now we need to plot the four poses
% Helper function is in the supporting functions section

for i = 1:2
    draw_pose(18+i, xs(i), ys(i), thetas(i), 6, i);
    drawnow;
end
```

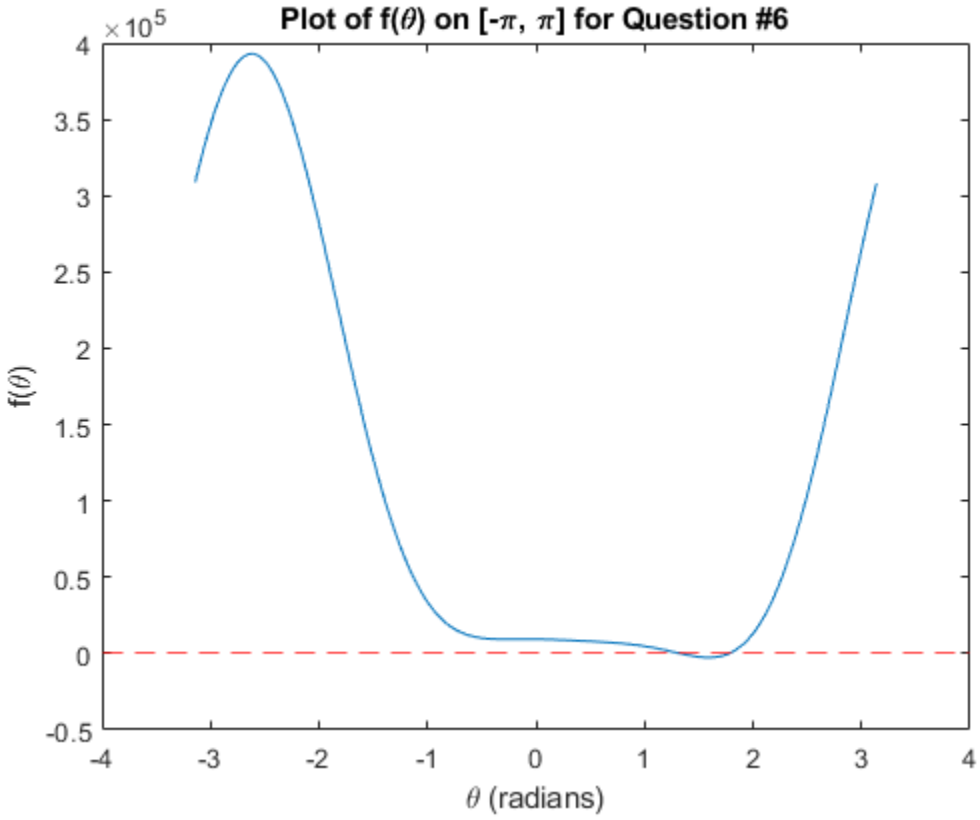
ALL FUNCTIONS SUPPORTING THIS CODE

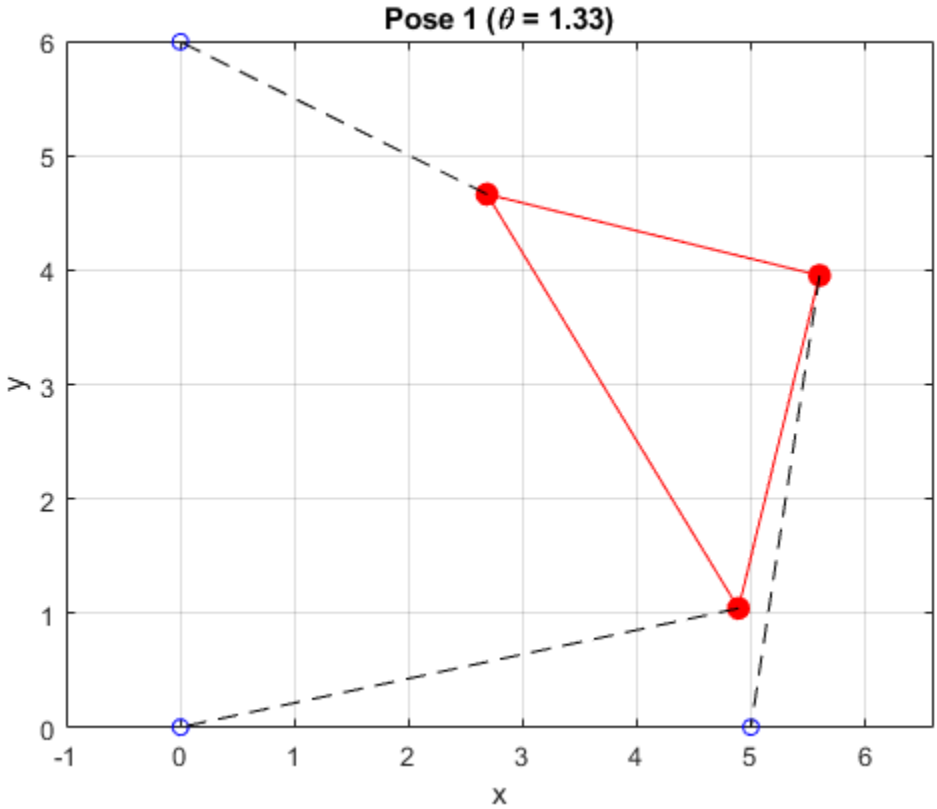
```
% f(theta) function with ability to change p2
function out = f_variable_p2(theta, p2)
    L1 = 3; L2 = 3 * sqrt(2); L3 = 3;
    gamma = pi / 4;
    p1 = 5; p3 = 3;
    x1 = 5; x2 = 0; y2 = 6;

    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

% Forward kinematics problem solver with variable p2
```

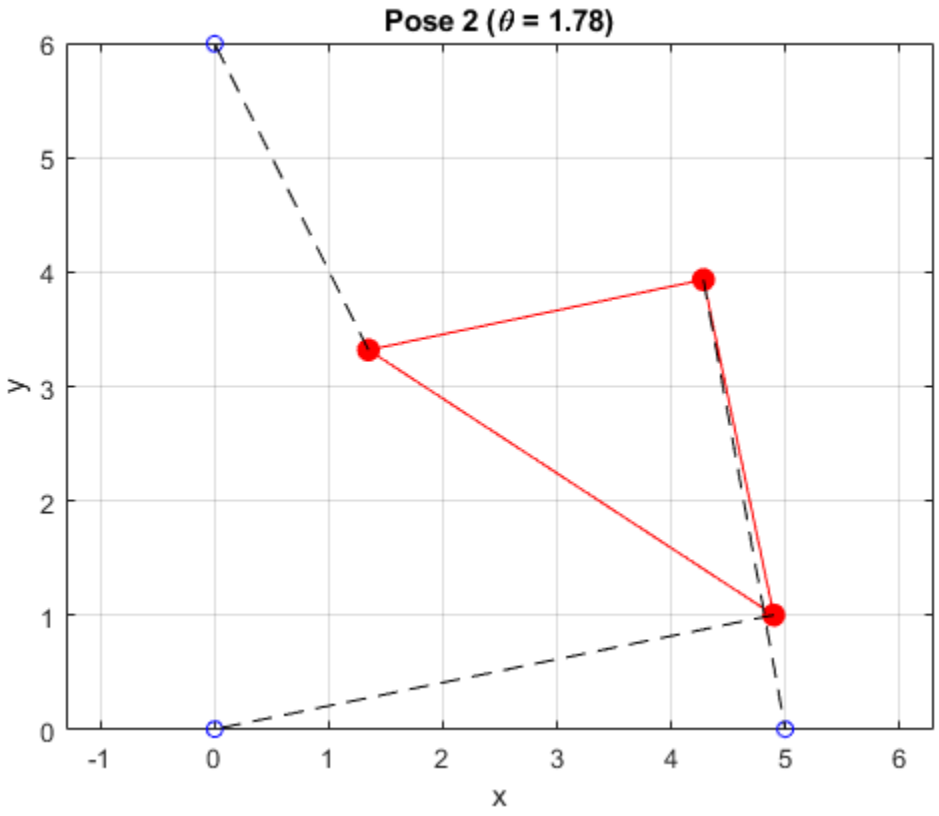
```
function [x, y] = forward_kinematics_variable_p2(theta, p2)

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

    % Angle across from L1
    gamma = pi / 4;

    % Strut lengths
    p1 = 5;
    p3 = 3;

    % Strut base positions
    x1 = 5;
    x2 = 0;
    y2 = 6;

    % Compute intermediate terms
    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;

    % Numerators and denominator
    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);

    % Solve for x and y
    x = N1 / D;
    y = N2 / D;

end


function draw_pose(fig_num, x, y, theta, question_number, pose_index)
    % Constants
    L2 = 3 * sqrt(2);
    L3 = 3;
    gamma = pi/4;
    x1 = 5; x2 = 0; y2 = 6;

    % Triangle corner positions
    u1 = x;
    v1 = y;
    u2 = x + L3 * cos(theta);
    v2 = y + L3 * sin(theta);
    u3 = x + L2 * cos(theta + gamma);
    v3 = y + L2 * sin(theta + gamma);
```

```
    % Compute strut lengths
    p1 = norm([u1, v1] - [0, 0]);
    p2 = norm([u2, v2] - [x1, 0]);
    p3 = norm([u3, v3] - [x2, y2]);

    % Plot
    figure(fig_num)
    plot([u1 u2 u3 u1], [v1 v2 v3 v1], 'r'); hold on
    plot([0 x1 x2], [0 0 y2], 'bo')
    plot([u1 u2 u3], [v1 v2 v3], 'ro', 'MarkerSize', 8, 'MarkerFaceColor',
'r')
    plot([u1 0], [v1 0], 'k--')    % p1
    plot([u2 x1], [v2 0], 'k--')   % p2
    plot([u3 x2], [v3 y2], 'k--')  % p3

    % Pose label
    title_str = sprintf('Pose %d (\\theta = %.2f)', pose_index, theta);


    title(title_str)
    xlabel('x')
    ylabel('y')
    axis equal
    grid on

    % Print strut lengths
    fprintf("Pose %d: p1 = %.4f, p2 = %.4f, p3 = %.4f\n", pose_index, p1,
p2, p3);
end
```

```
Found p2 = 4.00 with exactly 2 poses
Pose 1: p1 = 5.0000, p2 = 4.0000, p3 = 3.0000
Pose 2: p1 = 5.0000, p2 = 4.0000, p3 = 3.0000
```


Plot of f(θ) on [-π, π] for Question #6
×10^5
f(θ)
θ (radians)


Pose 1 (θ = 1.33)
x
y



Published with MATLAB® R2024b