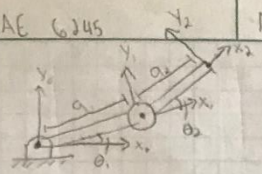
Jonathan Schwartz
MAE 6245

Assignment #5

Written answers and derivations:

MAE 6245 | Assignment #5 | Jonathan Schwartz

1. a)



$$T_1^0 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & a_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & a_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^0 = T_1^0 T_2^1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & a_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & a_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^0 = \begin{bmatrix} \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2 & (-\cos\theta_1\sin\theta_2 - \sin\theta_1\cos\theta_2) & 0 & a_2(\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2) + a_1\cos\theta_1 \\ \sin\theta_1\cos\theta_2 + \cos\theta_1\sin\theta_2 & (-\sin\theta_1\sin\theta_2 + \cos\theta_1\cos\theta_2) & 0 & a_2(\sin\theta_1\cos\theta_2 + \cos\theta_1\sin\theta_2) + a_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\sin(\theta_1+\theta_2) = \sin\theta_1\cos\theta_2 + \cos\theta_1\sin\theta_2$

$\cos(\theta_1-\theta_2) = \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2$

$$T_2^0 = \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) & 0 & a_1\cos\theta_1 + a_2\cos(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & a_1\sin\theta_1 + a_2\sin(\theta_1+\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$x = a_1\cos\theta_1 + a_2\cos(\theta_1+\theta_2)$ $\quad x^2 = a_1^2\cos^2\theta_1 + a_2^2\cos^2(\theta_1+\theta_2) + 2a_1a_2\cos\theta_1\cos(\theta_1+\theta_2)$

$y = a_1\sin\theta_1 + a_2\sin(\theta_1+\theta_2)$ $\quad y^2 = a_1^2\sin^2\theta_1 + a_2^2\sin^2(\theta_1+\theta_2) + 2a_1a_2\sin\theta_1\sin(\theta_1+\theta_2)$

$$x^2+y^2 = a_1^2(\cos^2+\sin^2) + a_2^2(\cos^2+\sin^2) + 2a_1a_2[\cos\theta_1\cos(\theta_1+\theta_2) + \sin\theta_1\sin(\theta_1+\theta_2)]$$

$$x^2+y^2 = a_1^2 + a_2^2 + 2a_1a_2[\cos\theta_1\cos(\theta_1+\theta_2) + \sin\theta_1\sin(\theta_1+\theta_2)]$$

$\cos(A-B) = \cos A\cos B + \sin A\sin B$

$$x^2+y^2 = a_1^2+a_2^2 + 2a_1a_2\cos(\theta_1+\theta_2-\theta_1) = a_1^2+a_2^2+2a_1a_2\cos\theta_2$$

$$\boxed{\cos^{-1}\left(\frac{x^2+y^2-a_1^2-a_2^2}{2a_1a_2}\right) = \theta_2}$$

$X = a_1\cos\theta_1 + a_2\cos(\theta_1+\theta_2)$

$x = a_1\cos\theta_1 + a_2\left[\cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2\right]$ $\cos(A+B) = \cos A\cos B - \sin A\sin B$

$x = \cos\theta_1\left(a_1 + a_2\cos\theta_2\right) - a_2\sin\theta_1\sin\theta_2$

$0 = \cos\theta_1 \underbrace{(a_1+a_2\cos\theta_2)}_{a} - \sin\theta_1 \underbrace{a_2\sin\theta_2}_{b} - \underbrace{x}_{c}$

Since for: $a\cos\phi + b\sin\phi - c = 0$ we know: $\phi = \text{atan2}(b,a) \pm \text{atan2}\left(\sqrt{a^2+b^2-c^2}, c\right)$

$$\boxed{\theta_1 = \text{atan2}\left(-a_2\sin\theta_2,\ a_1+a_2\cos\theta_2\right) \pm \text{atan2}\left(\sqrt{(a_1+a_2\cos\theta_2)^2+(a_2\sin\theta_2)^2},\ x\right)}$$

(1) g)

Inverse Kinematics:	0.003889	sec
Optimization:	0.094386	sec
Brute force:	0.053358	sec

* Without Plotting *

All seem valid for this application. Would be interested to see how their times compare from a system with a large number of joint variables. I'd expect the brute force approach to quickly become slow, as well as the optimization approach to a lesser degree. Inverse Kinematics is clearly the most efficient approach.

(2) The functions seem to operate similarlly, with fmincon()'s ability to add constraints on the answer allowing for more pointed solutions. I noticed this most when I had entered other desired positions, particularly when x and/or Y were negative

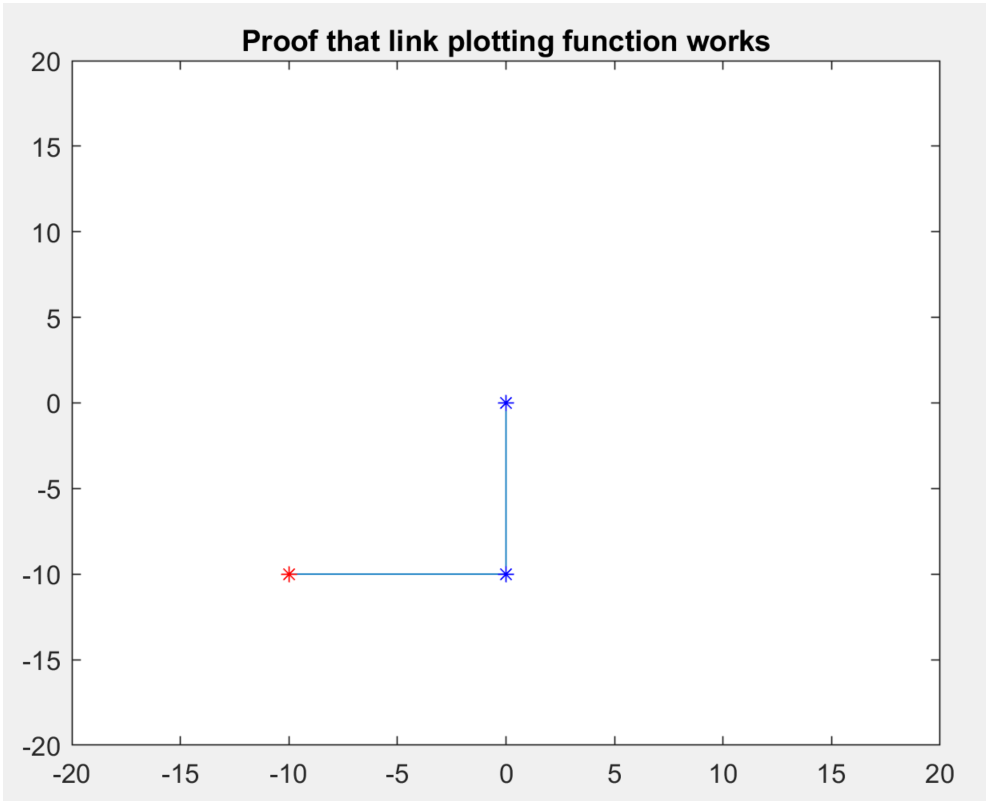
MATLAB Main Script

```
clear all;
close all;

% NOTE: for parts 1f and 1g, it is easier to see the output of the
%   plotLinks() function by commenting it out in 2 of the 3 loops at a time

theta1_init = 0.0;
theta2_init = 0.0;

xDes = -10;
yDes = -10;

% 1a) Inverse Kinematics Approach
[invKinTheta1, invKinTheta2] = findAnglesInverseKin(xDes, yDes)

% 1b) Optimization Approach (the global variables are for part 1f, where
% several desired positions are passed to the optimization function)
global xDesired;
global yDesired;
xDesired = xDes;
yDesired = yDes;
opt_answer = fminsearch(@findAnglesOptimization,[theta1_init,theta2_init])

% 1c) Brute Force Approch
[bForceTheta1, bForceTheta2] = findAnglesBruteForce(xDes,yDes)


% 1d) Circle of Points
xVals = [];
yVals = [];
radius = 15;
for i = 0:pi/8:2*pi
    xVals = [xVals radius*cos(i)];
    yVals = [yVals radius*sin(i)];
end

% 1e) Plotting Links Function
plotLinks(-pi/2, -pi/2);
title("Proof that link plotting function works");
```

```
% 1f) Plotting links for each point on the circle using inverse kinematics
tic
for i = 1:length(xVals)
    [angle1, angle2] = findAnglesInverseKin(xVals(i), yVals(i));
    plotLinks(angle1, angle2);
end
toc

% 1g) Plotting links for each point on the circle using optimization
tic
for i = 1:length(xVals)
    xDesired = xVals(i);
    yDesired = yVals(i);
    opt_answer = fminsearch(@findAnglesOptimization,[theta1_init,theta2_init]);
    plotLinks(opt_answer(1), opt_answer(2));
end
toc

% 1g) Plotting links for each point on the circle using brute force (guess
% and check) approach
%   NOTE: Joint angles limited to [0,90 deg] and [-90,90 deg] for theta1
%   and theta2, respectively.
tic
for i = 1:length(xVals)
    [bForceTheta1, bForceTheta2] = findAnglesBruteForce(xVals(i), yVals(i));
    plotLinks(bForceTheta1, bForceTheta2);
end
toc

% 2) Using fmincon function to solve inverse kinematic equation
fun2 = @inverseEqn2;
theta1_init = 0.6;
z = fminunc(fun2,theta1_init)
z = fmincon(fun2,theta1_init,[1;-1],[pi;pi])
```

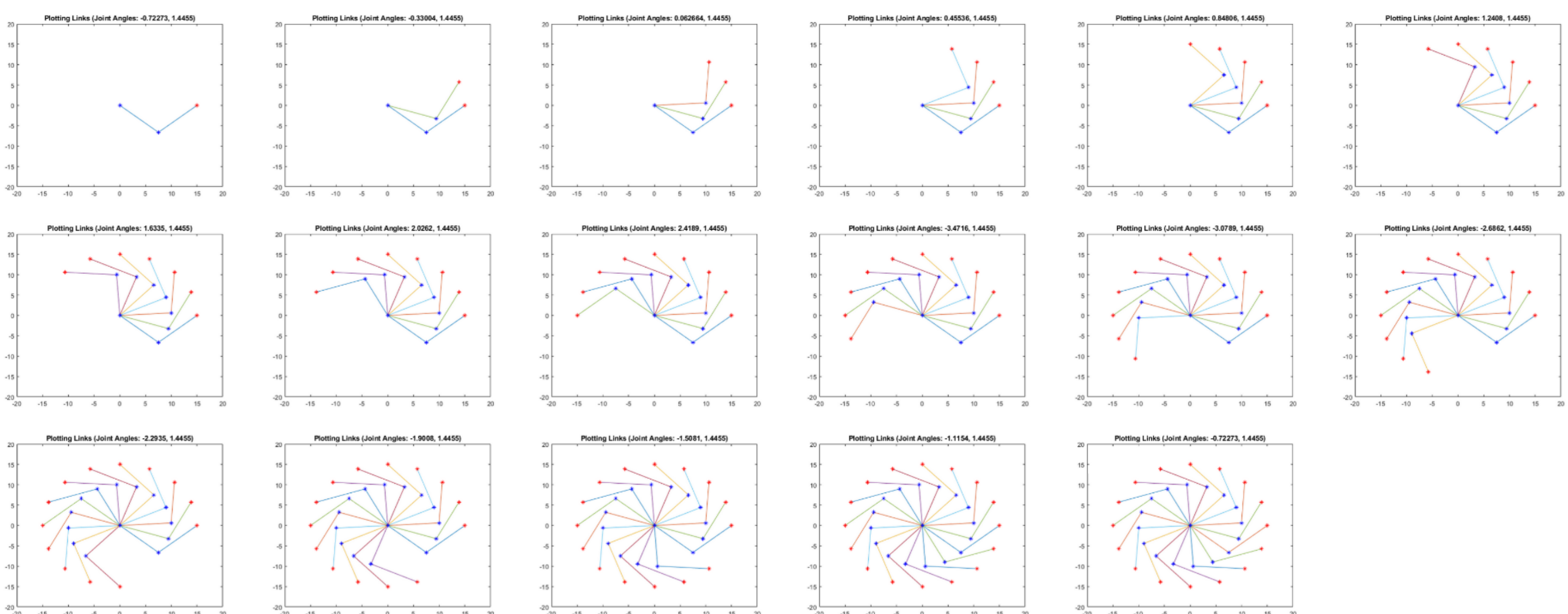
Script Output:

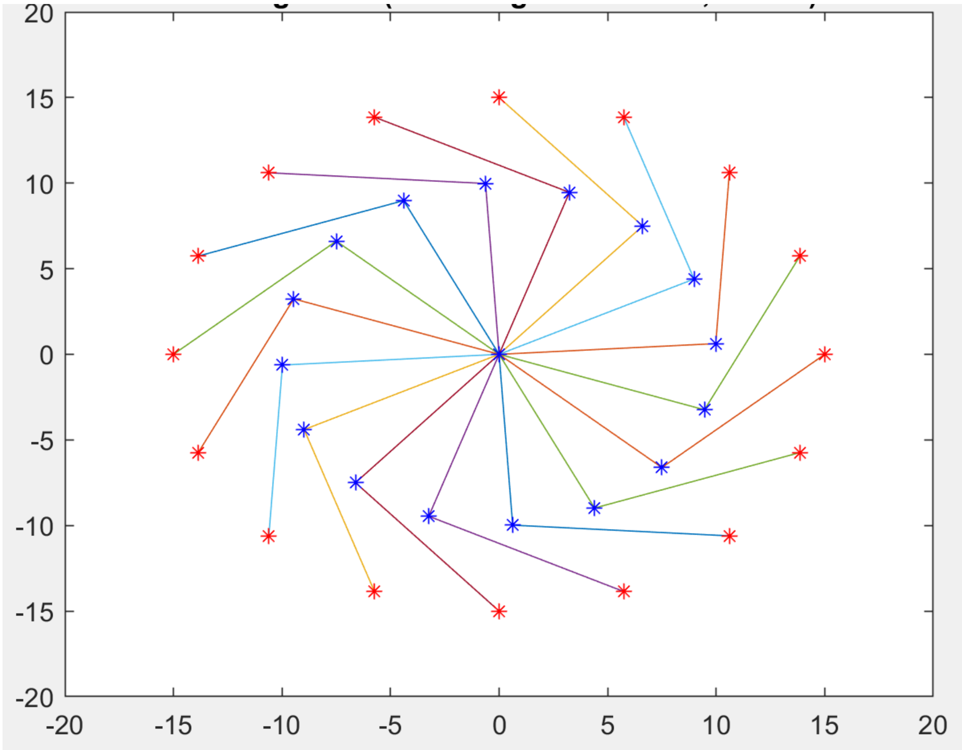
```
invKinTheta1 =

   -3.1416


invKinTheta2 =

    1.5708


opt_answer =

   -3.1416    1.5708


bForceTheta1 =

     0


bForceTheta2 =

   -1.5708
```



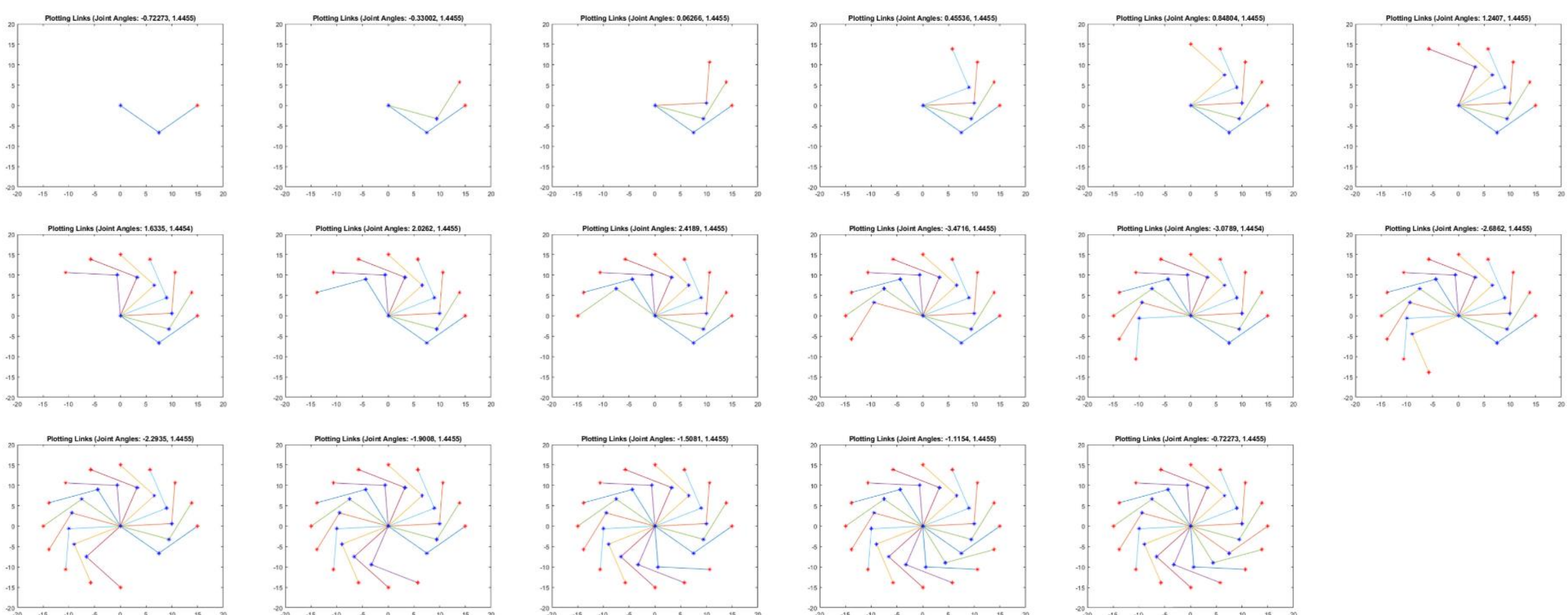
Note: *Joint angles for this test were* $(\frac{-\pi}{2}, \frac{-\pi}{2})$

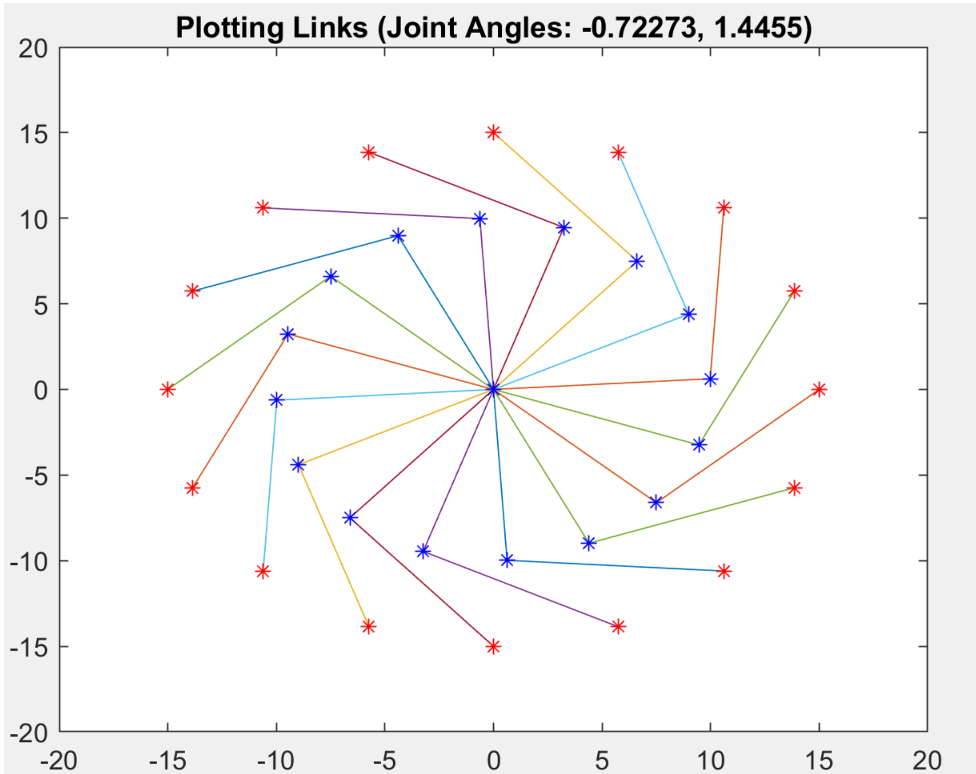
Inverse Kinematic Approach *(Circle with Radius of 15)*



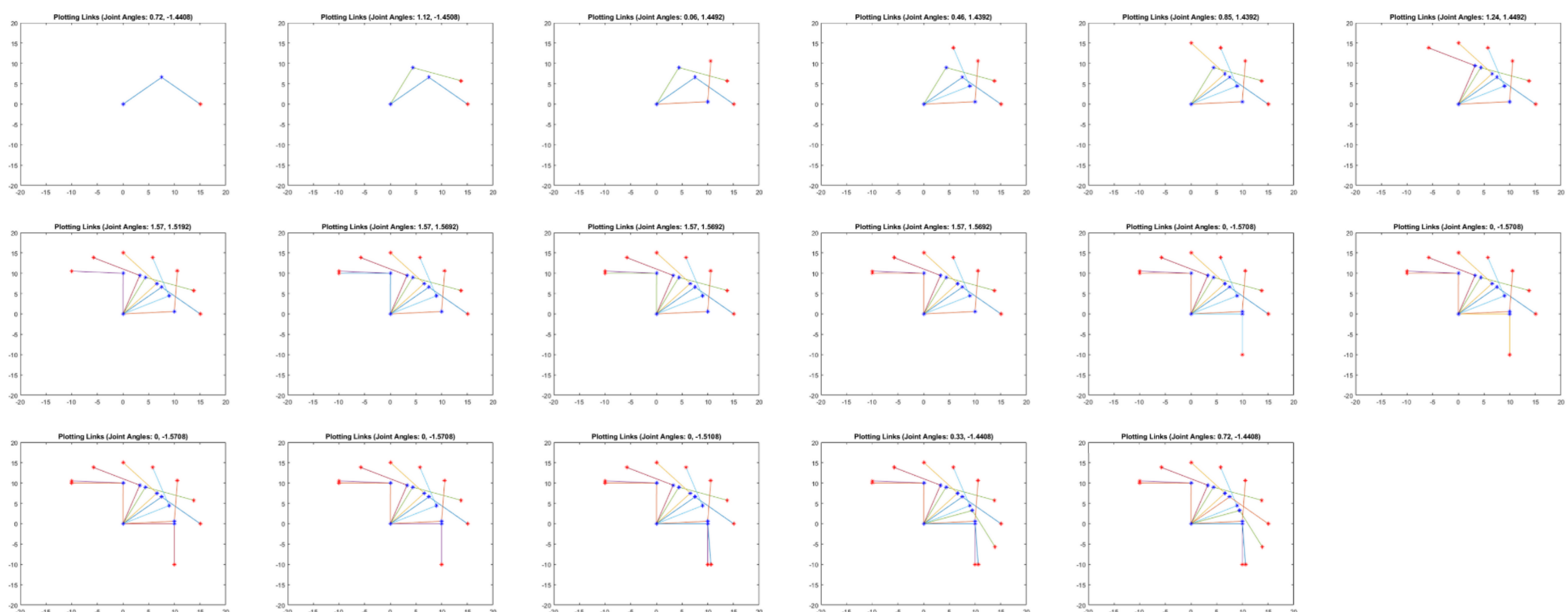


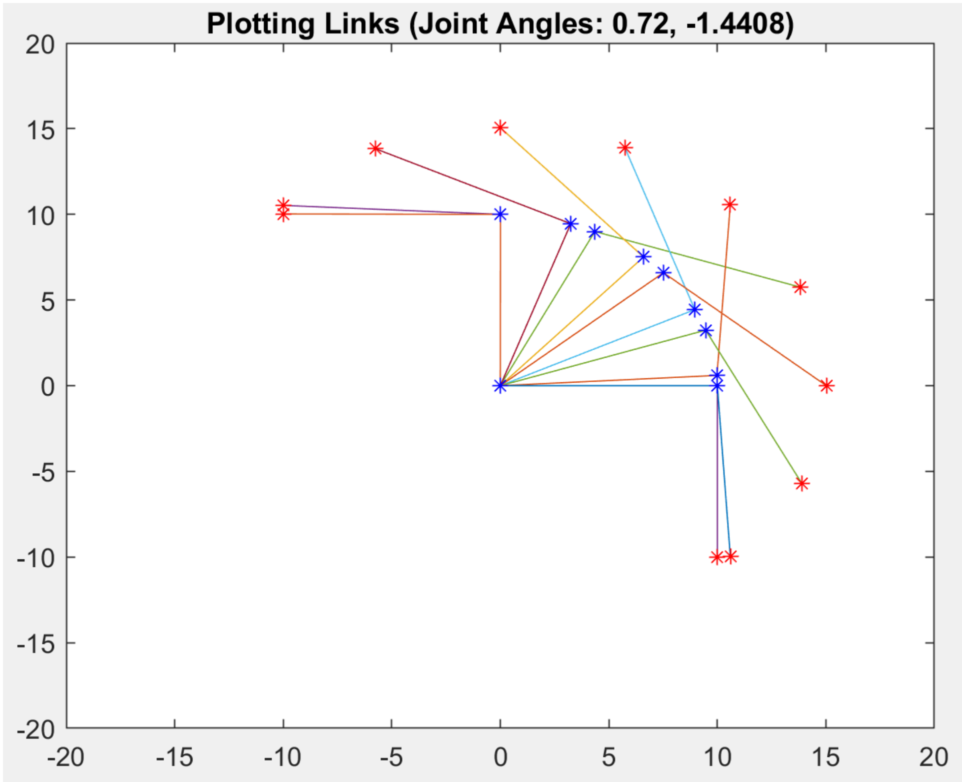
Optimization Approach *(Circle with Radius of 15)*





Brute Force Approach *(Radius of 15, joint 1 limited to [0,90º], joint 2 limited to [-90,90º]*






```
Elapsed time is 0.003633 seconds.
Elapsed time is 0.091100 seconds.
Elapsed time is 0.039136 seconds.

Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.


z =

    0.7854


Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.


z =

    0.7854
```