

ECE 4560 - Homework 7

Brett Erickson

1.

a)

```
L1 = 2;
L2 = 4;

x = 7;
y = 0;
alpha = acos((x^2+y^2+L1^2-L2^2)/(2*L1*sqrt(x^2 + y^2)));
beta = acos((L1^2+L2^2-x^2-y^2)/(2*L1*L2));
gamma = atan2(0,7);
theta_1 = gamma - alpha;
theta_2 = pi - beta;

fprintf("Righty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

```
Righty joint angles - Theta 1: 0.000000 radians
Theta 2: 0.000000 radians
```

```
theta_1 = gamma + alpha;
theta_2 = beta - pi;

fprintf("Lefty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

```
Lefty joint angles - Theta 1: 0.000000 radians
Theta 2: 0.000000 radians
```

Given that the point (7,0) is not within the arm workspace, the alpha and beta functions yield imaginary numbers and throw off the rest of the calculation.

b)

```
x = 3;
y = 3;
alpha = acos((x^2+y^2+L1^2-L2^2)/(2*L1*sqrt(x^2 + y^2)));
beta = acos((L1^2+L2^2-x^2-y^2)/(2*L1*L2));
gamma = atan2(0,7);
theta_1 = gamma - alpha;
theta_2 = pi - beta;

fprintf("Righty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

Righty joint angles - Theta 1: -1.209429 radians
Theta 2: 1.696124 radians

```
theta_1 = gamma + alpha;
theta_2 = beta - pi;

fprintf("Lefty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

Lefty joint angles - Theta 1: 1.209429 radians
Theta 2: -1.696124 radians

```
L1 = 2;
L2 = 4;
```

c)

```
x = -2;
y = 2;
alpha = acos((x^2+y^2+L1^2-L2^2)/(2*L1*sqrt(x^2 + y^2)));
beta = acos((L1^2+L2^2-x^2-y^2)/(2*L1*L2));
gamma = atan2(0,7);
theta_1 = gamma - alpha;
theta_2 = pi - beta;

fprintf("Righty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

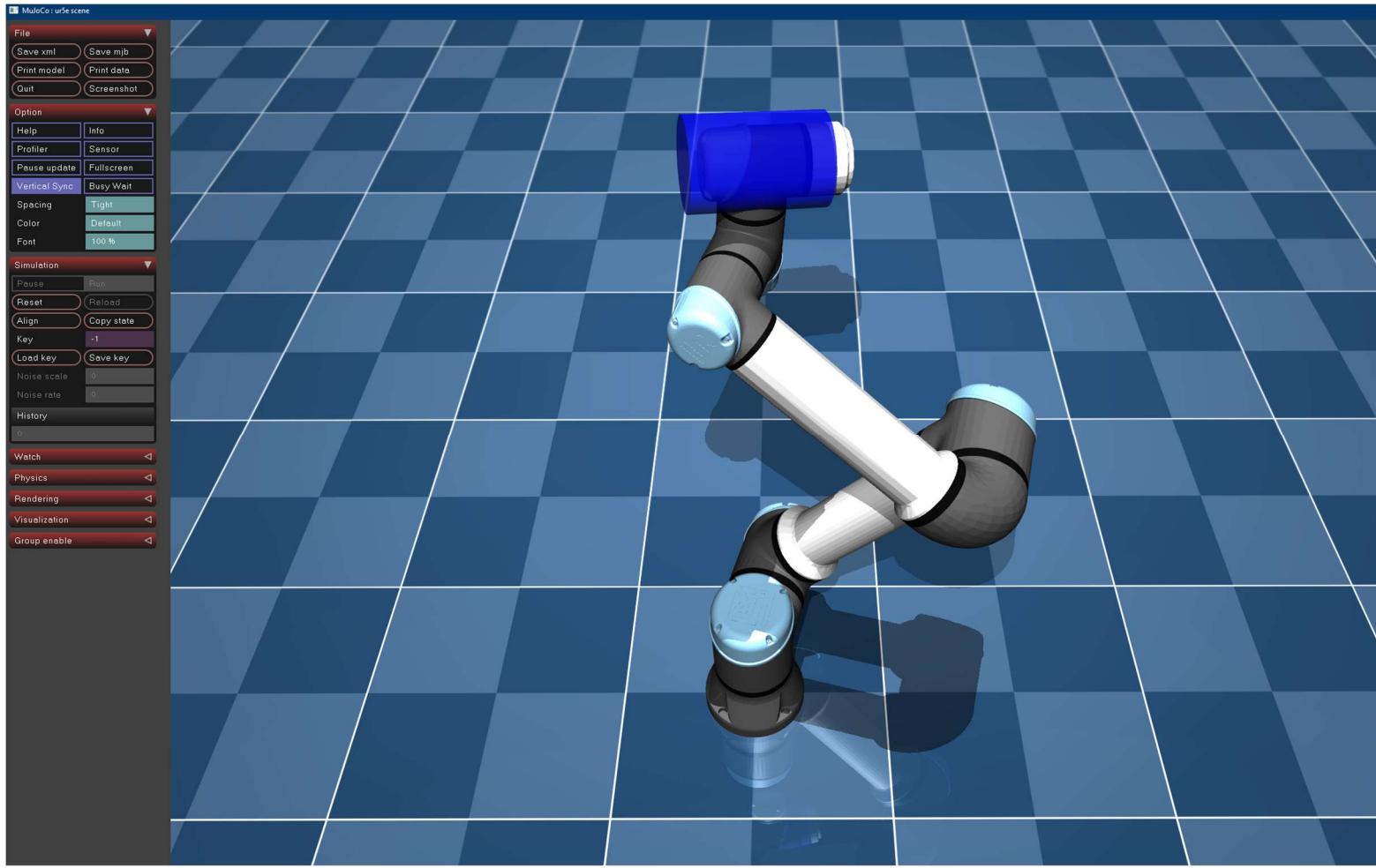
Righty joint angles - Theta 1: -1.932163 radians
Theta 2: 2.418858 radians

```
theta_1 = gamma + alpha;
theta_2 = beta - pi;

fprintf("Lefty joint angles - Theta 1: %f radians \n Theta 2: %f radians",theta_1, theta_2);
```

Lefty joint angles - Theta 1: 1.932163 radians
Theta 2: -2.418858 radians

2.



$$\omega_1 = [0, 0, 1], q_1 = [0, 0, 0]$$

$$\omega_2 = [0, 1, 0], q_2 = [0, 0, 0.163]$$

$$\omega_3 = [0, 1, 0], q_3 = [0.425, 0.007, 0.163]$$

$$\omega_4 = [0, 1, 0], q_4 = [0.817, 0.007, 0.163]$$

$$\omega_5 = [0, 0, -1], q_5 = [0.817, 0.134, 0.163]$$

$$\omega_6 = [0, 1, 0], q_6 = [0.817, 0.134, 0.063]$$

$$g_0 = \begin{bmatrix} 1 & 0 & 0 & 0.817 \\ 0 & 1 & 0 & 0.134 \\ 0 & 0 & 1 & 0.063 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.

Success: 1

Crushing failure: 4

b)

$\text{argmin}(|\text{actualPosition} - \text{intendedPosition}|)$

where the method to minimize is the Broyden–Fletcher–Goldfarb–Shanno algorithm.

