Class: EML6281

Assignment: Homework 5

Name: Elias Reyes

In [1]:
```
# load libraries
from math import sqrt, sin, cos
import numpy as np
from numpy import deg2rad, pi, rad2deg
import math
```

Problem 6.2

In [2]:
```
# solve trig function
def solve_trig(A, B, D):

    valid   = 0
    theta_a = 0
    theta_b = 0
    sin_gamma = B / sqrt(A*A + B*B)
    cos_gamma = A / sqrt(A*A + B*B)

    gamma           = math.atan2(sin_gamma,cos_gamma)
    ratio = -D/sqrt(A*A + B*B)

    if ratio >= -1.0 and ratio <= 1.0:
        valid = 1
        cos_minus_gamma = math.acos(ratio)
        theta_a =  cos_minus_gamma + gamma
        theta_b = -cos_minus_gamma + gamma

    return valid, theta_a, theta_b
```

In [3]:
```
# test solve trig function

A = 1.2
B = -5.1
D = 0.8

valid, theta_a, theta_b = solve_trig(A, B, D)

print('Valid:',valid)
print('Theta_a:',rad2deg(theta_a))
print('Theta_b:',rad2deg(theta_b))
```

```
Valid: 1
Theta_a: 22.023535734376182
Theta_b: -175.54249590400178
```

Problem 6.3

In [4]:
```
alpha_12 = deg2rad(75.0)
alpha_23 = deg2rad(110.0)
alpha_34 = deg2rad(60.0)
alpha_41 = deg2rad(80.0)
theta_1 = deg2rad(120.0)
```

In [5]:
```
# 416
```

```
# solve for theta_2 first
# use Z_12 = C_34
X_1 = sin(alpha_41)*sin(theta_1)
Y_1 = -(sin(alpha_12)*cos(alpha_41) + cos(alpha_12)*sin(alpha_41)*cos(theta_1))
Z_1 = cos(alpha_12)*cos(alpha_41)-sin(alpha_12)*sin(alpha_41)*cos(theta_1)
A2  = sin(alpha_23)*Y_1
B2  = sin(alpha_23)*X_1
D2  = cos(alpha_23)*Z_1 - cos(alpha_34)
```

In [6]:
```
valid, theta_2a, theta_2b = solve_trig(A2, B2, D2)
print(theta_2a*180/pi)
```

```
125.02217073092707
```

In [7]:
```
# solve for theta_3

# X_12 = sin_34 * sin_3
# Y_12 = sin_34 * cos_3

X_12a = X_1 * cos(theta_2a) - Y_1 * sin(theta_2a)
Y_12a = cos(alpha_23)*(X_1*sin(theta_2a)+Y_1*cos(theta_2a)) - sin(alpha_23)*Z_1

sin_3a = X_12a / sin(alpha_34)
cos_3a = Y_12a / sin(alpha_34)

theta_3a = math.atan2(sin_3a,cos_3a)

X_12b = X_1 * cos(theta_2b) - Y_1 * sin(theta_2b)
Y_12b = cos(alpha_23)*(X_1*sin(theta_2b)+Y_1*cos(theta_2b)) - sin(alpha_23)*Z_1

sin_3b = X_12b / sin(alpha_34)
cos_3b = Y_12b / sin(alpha_34)

theta_3b = math.atan2(sin_3b,cos_3b)

print(theta_3a*180/pi)
print(theta_3b*180/pi)
```

```
-148.1917767900727
148.19177679007274
```

In [8]:
```
# solve for theta_4

# X_21 = sin_34 * sin_4
# Y_21 = sin_34 * cos_4

X_2a_bar = sin(alpha_23)*sin(theta_2a)
Y_2a_bar = -(sin(alpha_12)*cos(alpha_23) + cos(alpha_12)*sin(alpha_23)*cos(theta_2a))
Z_2a_bar = cos(alpha_12)*cos(alpha_23) - sin(alpha_12)*sin(alpha_23)*cos(theta_2a)
X_21a    = X_2a_bar * cos(theta_1) - Y_2a_bar * sin(theta_1)
Y_21a    = cos(alpha_41)*(X_2a_bar*sin(theta_1)+Y_2a_bar*cos(theta_1))-sin(alpha_41)*Z_2a_bar

sin_4a = X_21a / sin(alpha_34)
cos_4a = Y_21a / sin(alpha_34)

theta_4a = math.atan2(sin_4a,cos_4a)

X_2b_bar = sin(alpha_23)*sin(theta_2b)
Y_2b_bar = -(sin(alpha_12)*cos(alpha_23)+cos(alpha_12)*sin(alpha_23)*cos(theta_2b))
Z_2b_bar = cos(alpha_12)*cos(alpha_23) - sin(alpha_12)*sin(alpha_23)*cos(theta_2b)
X_21b    = X_2b_bar * cos(theta_1) - Y_2b_bar * sin(theta_1)
Y_21b    = cos(alpha_41)*(X_2b_bar*sin(theta_1)+Y_2b_bar*cos(theta_1))-sin(alpha_41)*Z_2b_bar

sin_4b = X_21b / sin(alpha_34)
cos_4b = Y_21b / sin(alpha_34)

theta_4b = math.atan2(sin_4b,cos_4b)
```

```
print(theta_4a*180/pi)
print(theta_4b*180/pi)
```

```
-113.90218976416982
-42.98862455401464
```

In [9]:
```
from tabulate import tabulate

table = [['Theta_1','Theta_2','Theta_3','Theta_4'],
         ['Theta_a',rad2deg(theta_1),rad2deg(theta_2a),rad2deg(theta_3a),rad2deg(theta_4a)],
         ['Theta_b',rad2deg(theta_1),rad2deg(theta_2b),rad2deg(theta_3b),rad2deg(theta_4b)]]
print(tabulate(table,headers='firstrow', tablefmt='fancy_grid'))
```

	Theta_1	Theta_2	Theta_3	Theta_4
Theta_a	120	125.022	-148.192	-113.902
Theta_b	120	60.3869	148.192	-42.9886

In [ ]: