Guillermo Fidel Navarro Vega A01274191

HW3

2DOF Links:

```
PL1 = Link('d', 0, 'a', 0.2, 'alpha', pi);
PL2 = Link('d', 0, 'a', 0.25, 'alpha', pi);
```

Creation of 2DOF robot:

```
TDOF = SerialLink([PL1 PL2 ], 'name', '2DOF')
```

```
TDOF =

2DOF:: 2 axis, RR, stdDH, slowRNE
+---+-----------+-----------+-----------+-----------+-----------+
| j |     theta |         d |         a |     alpha |    offset |
+---+-----------+-----------+-----------+-----------+-----------+
|  1|         q1|          0|        0.2|    3.14159|          0|
|  2|         q2|          0|       0.25|    3.14159|          0|
+---+-----------+-----------+-----------+-----------+-----------+
```

```
qz = [0,0];
TDOF.plot(qz)
inv1 =TDOF.fkine(qz)
```

```
inv1 =
        1        0        0     0.45
        0        1        0        0
        0        0        1        0
        0        0        0        1
```

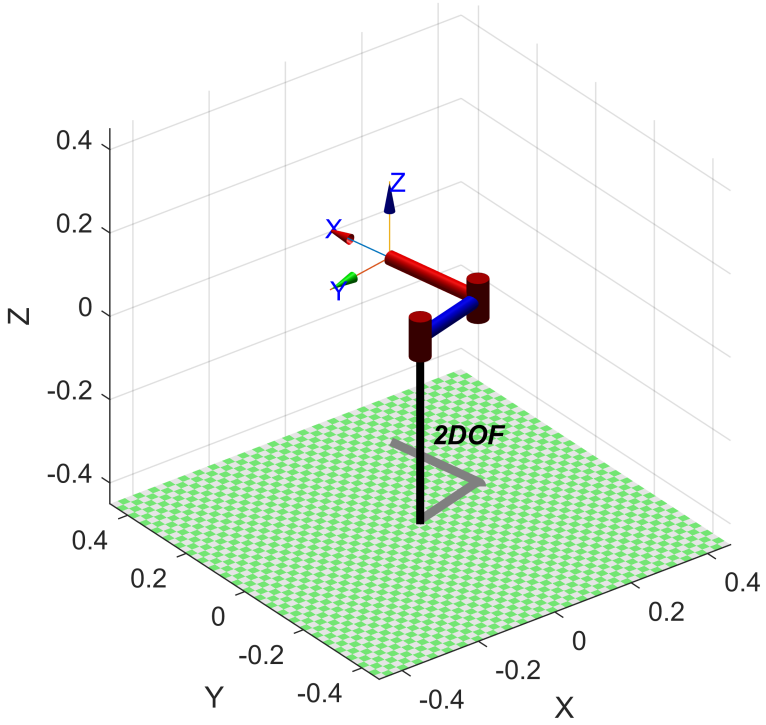
```
inv2 = [1 0 0 0.15; 0 1 0 0.3; 0 0 1 0; 0 0 0 1]
```

```
inv2 = 4×4
    1.0000         0         0    0.1500
         0    1.0000         0    0.3000
         0         0    1.0000         0
         0         0         0    1.0000
```

```
inverseRes = TDOF.ikine(inv2,[0,0],'mask',[1 1 0 0 0 0],'lu')
```

```
inverseRes = 1×2
    0.2717   -1.4706
```

```
TDOF.plot(inverseRes);
```



Jacobian

```
syms a1 a2
l1 = 0.2;
l2 = 0.25;
J = [-l1*sin(a1)-l2*sin(a1+a2) -l2*sin(a1+a2); l1*cos(a1)+l2*cos(a1+a2) l2*cos(a1+a2)]
```

```
J =
```

$$\begin{pmatrix} -\frac{\sin(a_1+a_2)}{4}-\frac{\sin(a_1)}{5} & -\frac{\sin(a_1+a_2)}{4} \\ \frac{\cos(a_1+a_2)}{4}+\frac{\cos(a_1)}{5} & \frac{\cos(a_1+a_2)}{4} \end{pmatrix}$$

Inverse Jacobian

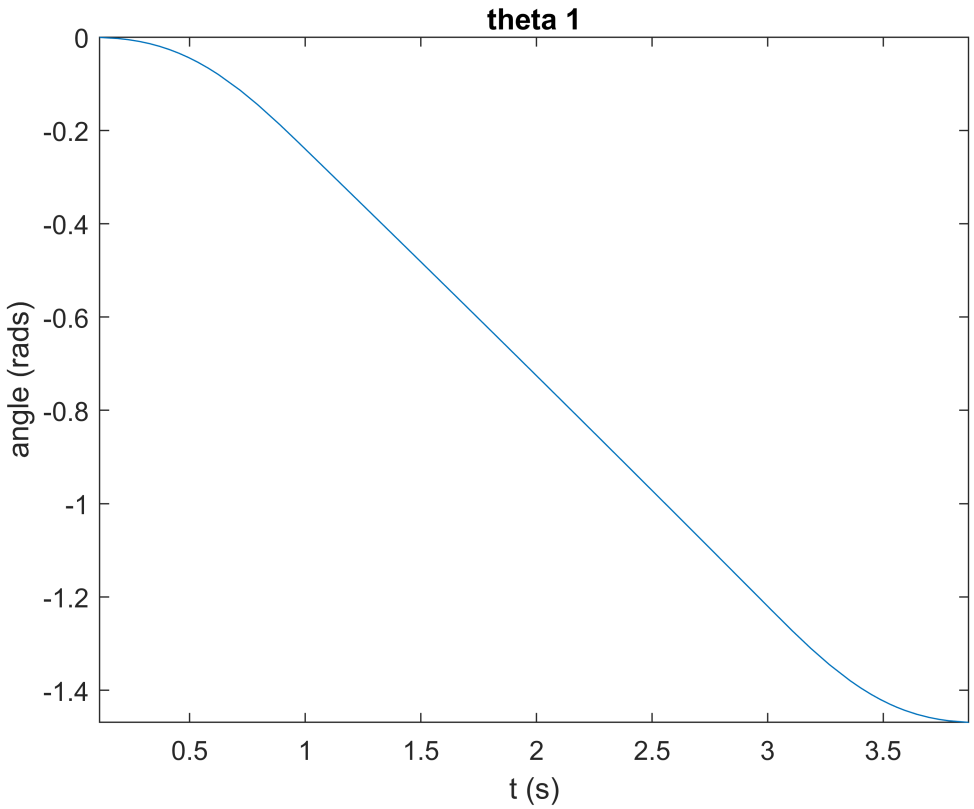
```
invJ = simplify(J^-1)
```

```
invJ =
```

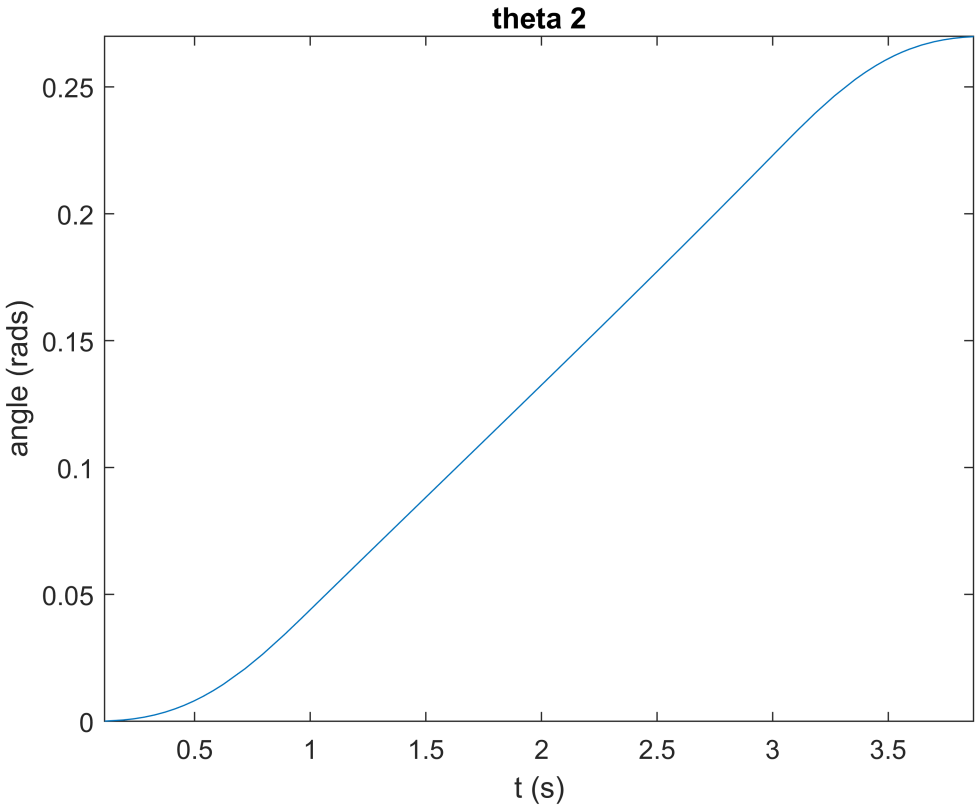
$$\begin{pmatrix} \frac{5\cos(a_1+a_2)}{\sin(a_2)} & \frac{5\sin(a_1+a_2)}{\sin(a_2)} \\ -\frac{5\cos(a_1+a_2)+4\cos(a_1)}{\sin(a_2)} & -\frac{5\sin(a_1+a_2)+4\sin(a_1)}{\sin(a_2)} \end{pmatrix}$$

Trajectory 4-3-4

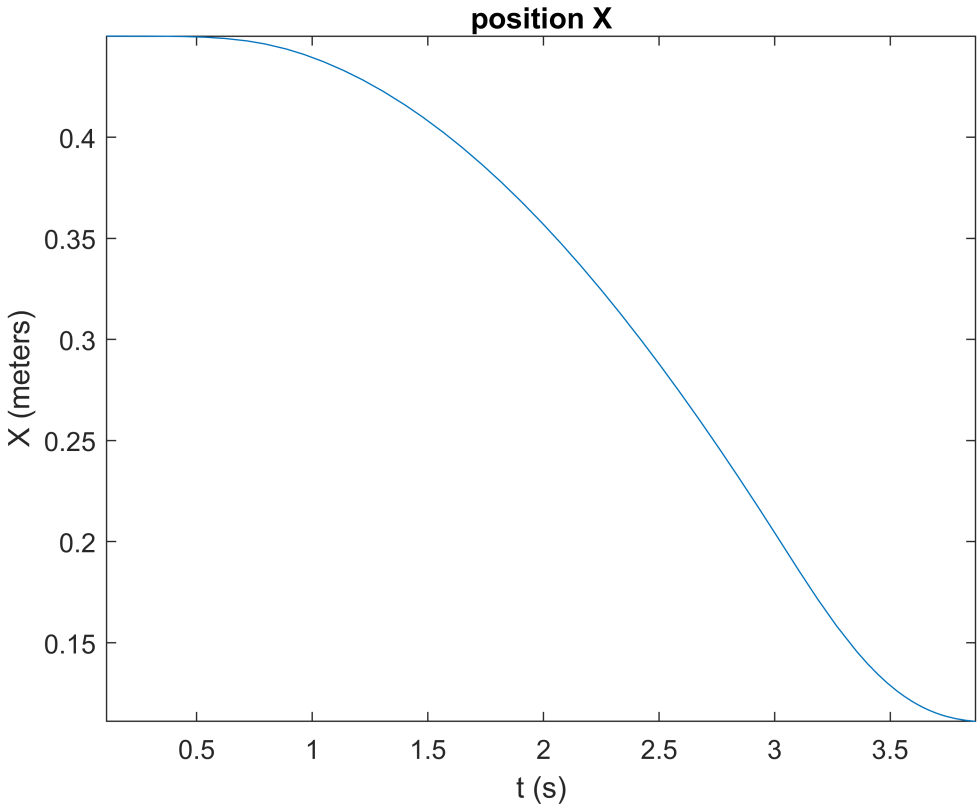
```
f1 = TrajectoryPlan(0, 0, 1, -0.24, 3, -1.22, 4, -1.47);
fplot(f1);
ylabel("angle (rads)");
xlabel("t (s)");
title('theta 1')
```



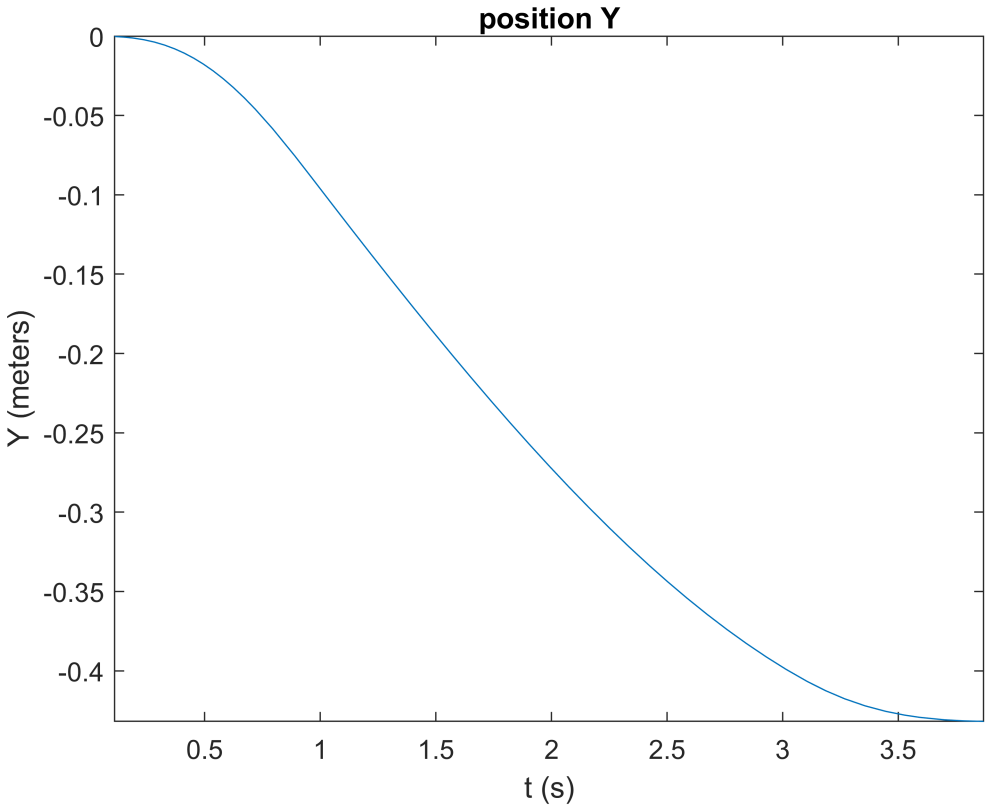

```
f2 = TrajectoryPlan(0, 0, 1, 0.044, 3, 0.223, 4, 0.27);
fplot(f2);
ylabel("angle (rads)");
xlabel("t (s)");
title('theta 2')
```




```
Xpos = l1*cos(f1) + l2*cos(f1+f2);
fplot(Xpos);
ylabel("X (meters)");
xlabel("t (s)");
title('position X')
```




```
Ypos = l1*sin(f1) + l2*sin(f1+f2);
fplot(Ypos);
ylabel("Y (meters)");
xlabel("t (s)");
title('position Y');
```




```
function [f] = TrajectoryPlan(t0, Ang0, t1, Ang1, t2, Ang2, tf, Af)
syms a10 a11 a12 a13 a14 a20 a21 a22 a23 a30 a31 a32 a33 a34
syms h1(t) h2(t) h3(t) h1p(t) h2p(t) h3p(t) h1s(t) h2s(t) h3s(t)
h1(t) = a14*t^4 + a13*t^3 + a12*t^2 + a11*t + a10;
h2(t) = a23*t^3 + a22*t^2 + a21*t + a20;
h3(t) = a34*t^4 + a33*t^3 + a32*t^2 + a31*t + a30;
h1p(t) = diff(h1);
h2p(t) = diff(h2);
h3p(t) = diff(h3);
h1s(t) = diff(h1p);
h2s(t) = diff(h2p);
h3s(t) = diff(h3p);

eq1 = h1(t0) == Ang0;
eq2 = h1p(t0) == 0;
eq3 = h1s(t0) == 0;
eq4 = h1(t1) == Ang1;
eq5 = h1(t1) == h2(t1);
eq6 = h1p(t1) == h2p(t1);
eq7 = h1s(t1) == h2s(t2);
eq8 = h2(t2) == Ang2;
eq9 = h2(t2) == h3(t2);
eq10 = h2p(t2) == h3p(t2);
eq11 = h2s(t2) == h3s(t2);
eq12 = h3(tf) == Af;
```

```
eq13 = h3p(tf) == 0;
eq14 = h3s(tf) == 0;


sol = solve([eq1, eq2, eq3, eq4, eq5, eq6, eq7, eq8, eq9, eq10, eq11, eq12, ...
    eq13, eq14] , [a14, a13, a12, a11, a10, a23, a22, a21, a20, a34, a33, ...
    a32, a31, a30]);
h1(t) = subs(h1(t), [a14, a13, a12, a11, a10], [sol.a14, sol.a13, sol.a12, sol.a11, sol.a10]);
h2(t) = subs(h2(t), [a23, a22, a21, a20], [sol.a23, sol.a22, sol.a21, sol.a20]);
h3(t) = subs(h3(t), [a34, a33, a32, a31, a30], [sol.a34, sol.a33, sol.a32, sol.a31, sol.a30]);
f = piecewise((t0 <= t) <= t1 , h1, (t1 <= t) <= t2, h2, (t2 <= t) <= tf, h3);

end
```