

- Transformação Homogênea
- Caracterização dos elos por meio dos parâmetros de D-H
- Geração de trajetória
- Cinemática direta
- Animação gráfica

Prof. Mário Luiz Tronco

Alexandre Padilha Senni

Instalação

Toolbox de robótica

Disponível gratuitamente em :

<http://www.petercorke.com/RVC>

Livro: Peter Corke. **Robotics, vision and control.**

Procedimento para instalação:

- Descompactar o arquivo 'rvctools';
- Mover o arquivo descompactado para: C:\ProgramFiles\MATLAB\R2011b\toolbox;
- Abrir o MATLAB. Adicionar a pasta em **Set Path, Add Folder**;
- Clicar em **Save e Close**.

Transformação Homogênea

$$u = Hv$$

Transformação Homogênea

$$u = Hv$$

$$\begin{bmatrix} u_x \\ u_y \\ u_z \\ w \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ w \end{bmatrix}$$

Transformação Homogênea

$$u = Hv$$

$$\begin{bmatrix} u_x \\ u_y \\ u_z \\ w \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ w \end{bmatrix}$$

Translação no MATLAB

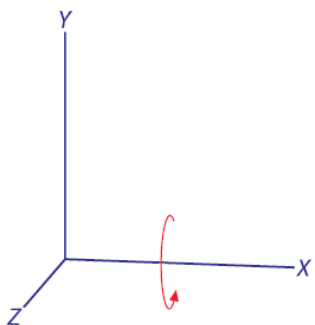
```
>> T = transl(a,b,c)
```

```
>> T =
```

```
    1    0    0    a  
    0    1    0    b  
    0    0    1    c  
    0    0    0    1
```

Transformação Homogênea

Rotação no MATLAB

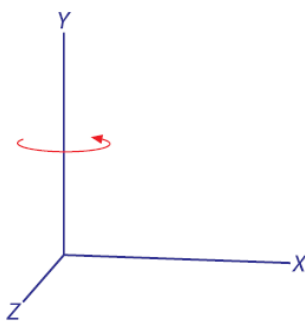


x

```
>> T = trotx(theta)
```

```
>> T =
```

1	0	0	0
0	cos θ	-sin θ	0
0	sin θ	cos θ	0
0	0	0	1

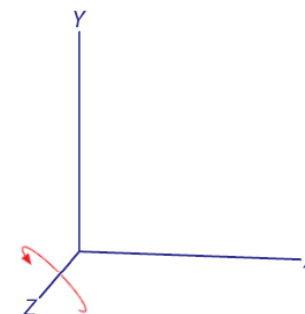


y

```
>> T = troty(theta)
```

```
>> T =
```

cos θ	0	sin θ	0
0	1	0	0
-sin θ	0	cos θ	0
0	0	0	1



z

```
>> T = trotz(theta)
```

```
>> T =
```

cos θ	-sin θ	0	0
sin θ	cos θ	0	0
0	0	1	0
0	0	0	1

Transformação Homogênea

```
>>T = transl(0.5, 0, 0);%Matriz de translação
```

```
>>R1 = troty(pi/2);%Matriz de rotação
```

```
>>R2 = troz(-pi/2);%Matriz de rotação
```

```
>>H = T*R1*R2;%Matriz homogênea concatenada
```

```
>>v = [0.5 0 0 1]';%Vetor v
```

```
>>u = H * v;%Vetor u transformado
```

```
u =
```

```
0.5000
```

```
-0.5000
```

```
-0.0000
```

```
1.0000
```

```
>>RPY = tr2rpy(H)%Roll, Pitch e Yaw
```

```
RPY =
```

```
0    1.5708   -1.5708
```

Caracterização dos elos por meio dos parâmetros de D-H

Criando elos no MATLAB:

$$\text{Elo} = \text{link}([\alpha_i \ a_i \ \theta_i \ d_i \ \sigma_i])$$

α_i : torção entre os eixos das juntas num plano perpendicular a a_i ;

a_i : extensão do elo;

θ_i : ângulo entre os elos;

d_i : distância entre a normal a_i e a_{i-1} dos dois elos (deslocamento de junta);

σ_i : tipo de junta (opcional): 0 – rotacional (default), 1 – prismática.

Caracterização dos elos por meio dos parâmetros de D-H

Criando um robô no MATLAB:

```
r1 = robot({link1 link 2 ...})
```

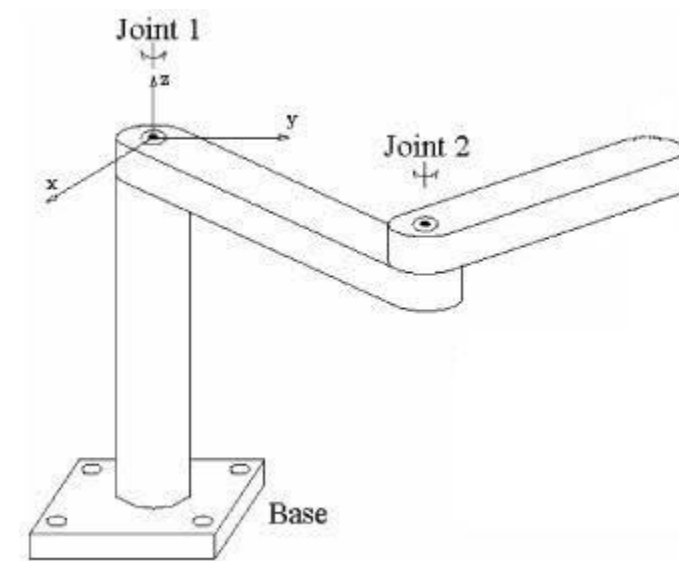
```
l1 = link([0 0.30 0 0 0]);%Criando elo 1
l2 = link([0 0.30 0 0 0]);%Criando elo 2

r1 = robot({l1 l2})%Criando o robô
r1.name = '2R';%Atribuindo nome ao robô

q0 = [0 0];%Posição angular inicial (radianos)

plot(r1, q0);%Gráfico do robô r1 na posição q0

drivebot(r1);%Ajustar manualmente as juntas do robô
```



Caracterização dos elos por meio dos parâmetros de D-H

Criando um robô no MATLAB:

```
r1 = robot({link1 link 2 ...})
```

```
l1 = link([0 0.30 0 0 0]);%Criando elo 1  
l2 = link([0 0.30 0 0 0]);%Criando elo 2  
  
r1 = robot({l1 l2});%Criando o robô  
r1.name = '2R';%Atribuindo nome ao robô  
  
q0 = [0 0];%Posição angular inicial (radianos)  
  
plot(r1, q0);%Gráfico do robô r1 na posição q0  
  
drivebot(r1);%Ajustar manualmente as juntas do robô
```

r1 =

noname (2 axis, RR)

grav = [0.00 0.00 9.81]

standard D&H parameters

alpha	A	theta	D	R/P	
0.000000	0.300000	0.000000	0.000000	R	(std)
0.000000	0.300000	0.000000	0.000000	R	(std)

Caracterização dos elos por meio dos parâmetros de D-H

Criando um robô no MATLAB:

`r1 = robot({link1 link 2 ...})`

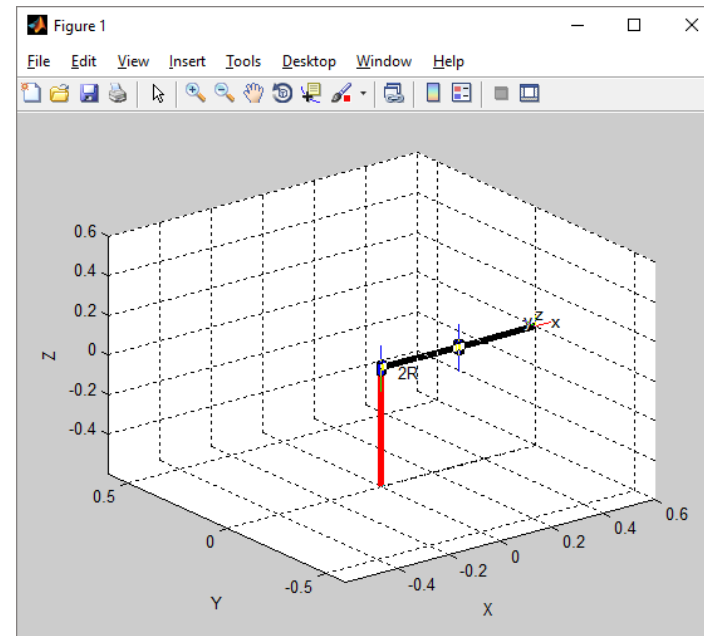
```
l1 = link([0 0.30 0 0 0]);%Criando elo 1
l2 = link([0 0.30 0 0 0]);%Criando elo 2

r1 = robot({l1 l2})%Criando o robô
r1.name = '2R';%Atribuindo nome ao robô

q0 = [0 0];%Posição angular inicial (radianos)

plot(r1, q0);%Gráfico do robô r1 na posição q0

drivebot(r1);%Ajustar manualmente as juntas do robô
```



Caracterização dos elos por meio dos parâmetros de D-H

Criando um robô no MATLAB:

`r1 = robot({link1 link 2 ...})`

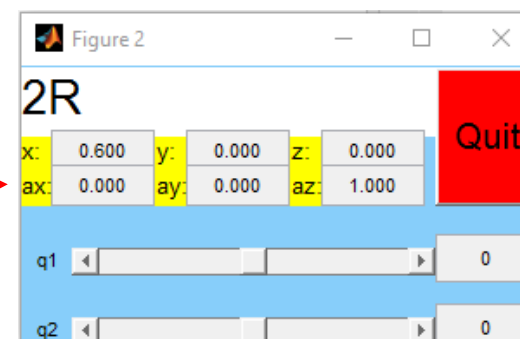
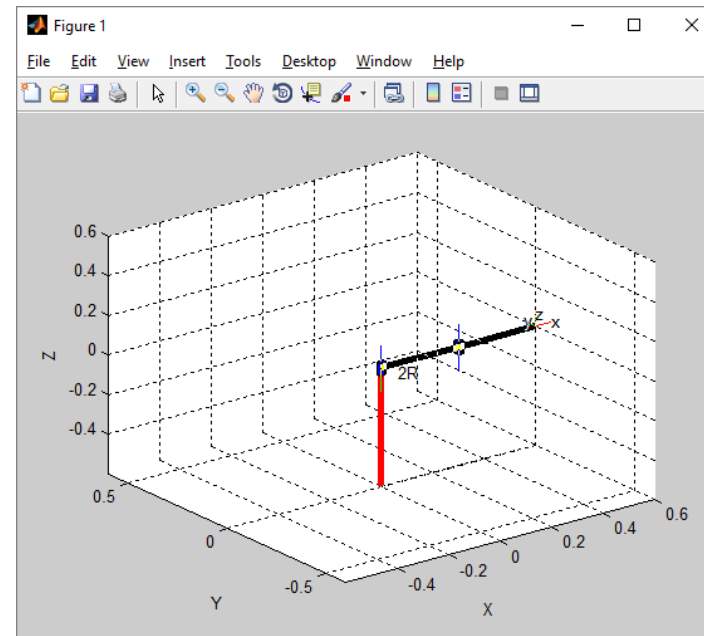
```
l1 = link([0 0.30 0 0 0]);%Criando elo 1
l2 = link([0 0.30 0 0 0]);%Criando elo 2

r1 = robot({l1 l2})%Criando o robô
r1.name = '2R';%Atribuindo nome ao robô

q0 = [0 0];%Posição angular inicial (radianos)

plot(r1, q0);%Gráfico do robô r1 na posição q0

drivebot(r1);%Ajustar manualmente as juntas do robô
```



Cinemática direta

```
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [pi/2 0];%Posição angular inicial (radianos)

>> H = fkine(r1,q0)%Matriz transformação homogênea
H =

    0.0000    -1.0000         0     0.0000
    1.0000     0.0000         0     0.6000
         0         0     1.0000         0
         0         0         0     1.0000

>> RPY = tr2rpy(H)%Roll, Pitch e Yaw
RPY =

         0         0     1.5708
```

Cinemática direta

```
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [0 0];%Posição angular inicial (radianos)
>> q1 = [pi/2 0];%Posição angular final (radianos)

>> t = [0:.05:2];%Vetor de tempo (0 até 2 segundos com incremento de 0,05 segundos)
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

>> figure(1);
>> subplot(2,1,1)
>> plot(t,q(:,1));%Gráfico da posição angular da junta 1 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 1 (rad)');

>> subplot(2,1,2)
>> plot(t,q(:,2));%Gráfico da posição angular da junta 2 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 2 (rad)');
```

Cinemática direta

```
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [0 0];%Posição angular inicial (radianos)
>> q1 = [pi/2 0];%Posição angular final (radianos)

>> t = [0:.05:2];%Vetor de tempo (0 até 2 segundos com incremento de 0,05 segundos)
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

>> figure(1);
>> subplot(2,1,1)
>> plot(t,q(:,1));%Gráfico da posição angular da junta 1 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 1 (rad)');

>> subplot(2,1,2)
>> plot(t,q(:,2));%Gráfico da posição angular da junta 2 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 2 (rad)');
```

Elemento	Tempo (s)	q	
		junta 1	junta 2
1	0,00	0	0
2	0,05	0,0002	0
3	0,10	0,0018	0
4	0,15	0,0059	0
...
41	2,00	$\pi/2$	0

Cinemática direta

```
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

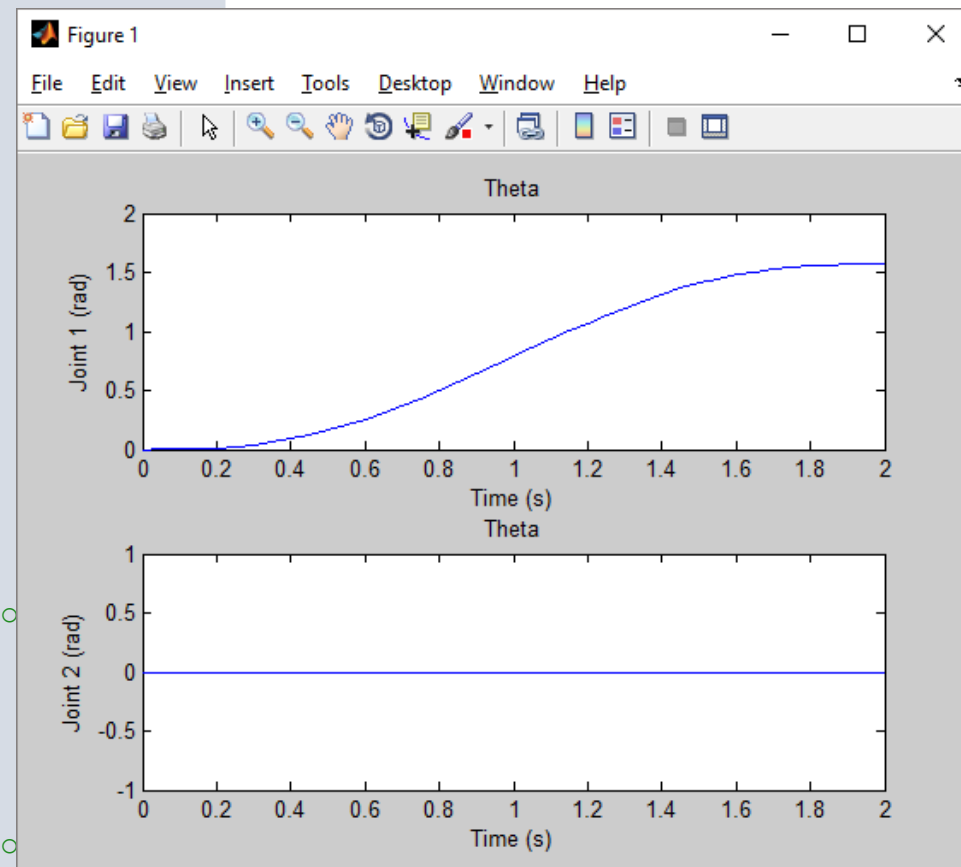
>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [0 0];%Posição angular inicial (radianos)
>> q1 = [pi/2 0];%Posição angular final (radianos)

>> t = [0:.05:2];%Vetor de tempo (0 até 2 segundos com incremento de 0,05)
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

>> figure(1);
>> subplot(2,1,1)
>> plot(t,q(:,1));%Gráfico da posição angular da junta 1 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 1 (rad)');

>> subplot(2,1,2)
>> plot(t,q(:,2));%Gráfico da posição angular da junta 2 variando no tempo
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 2 (rad)');
```



Cinemática direta

```
>> [q, dq, ddq] = jtraj(q0, q1, t);%Posição angular, velocidade angular e aceleração angular

>> figure(1);
>> subplot(3,1,1)
>> plot(t,q(:,1));%Gráfico da posição angular
>> title('Theta');
>> xlabel('Time (s)');
>> ylabel('Joint 1 (rad)');

>> subplot(3,1,2)
>> plot(t,dq(:,1));%Gráfico da velocidade angular
>> title('Velocity')
>> xlabel('Time (s)');
>> ylabel('Joint 1 vel (rad/s)');

>> subplot(3,1,3)
>> plot(t,ddq(:,1));%Gráfico da aceleração angular
>> title('Acceleration')
>> xlabel('Time (s)');
>> ylabel('Joint 1 accel (rad/s2)')
```

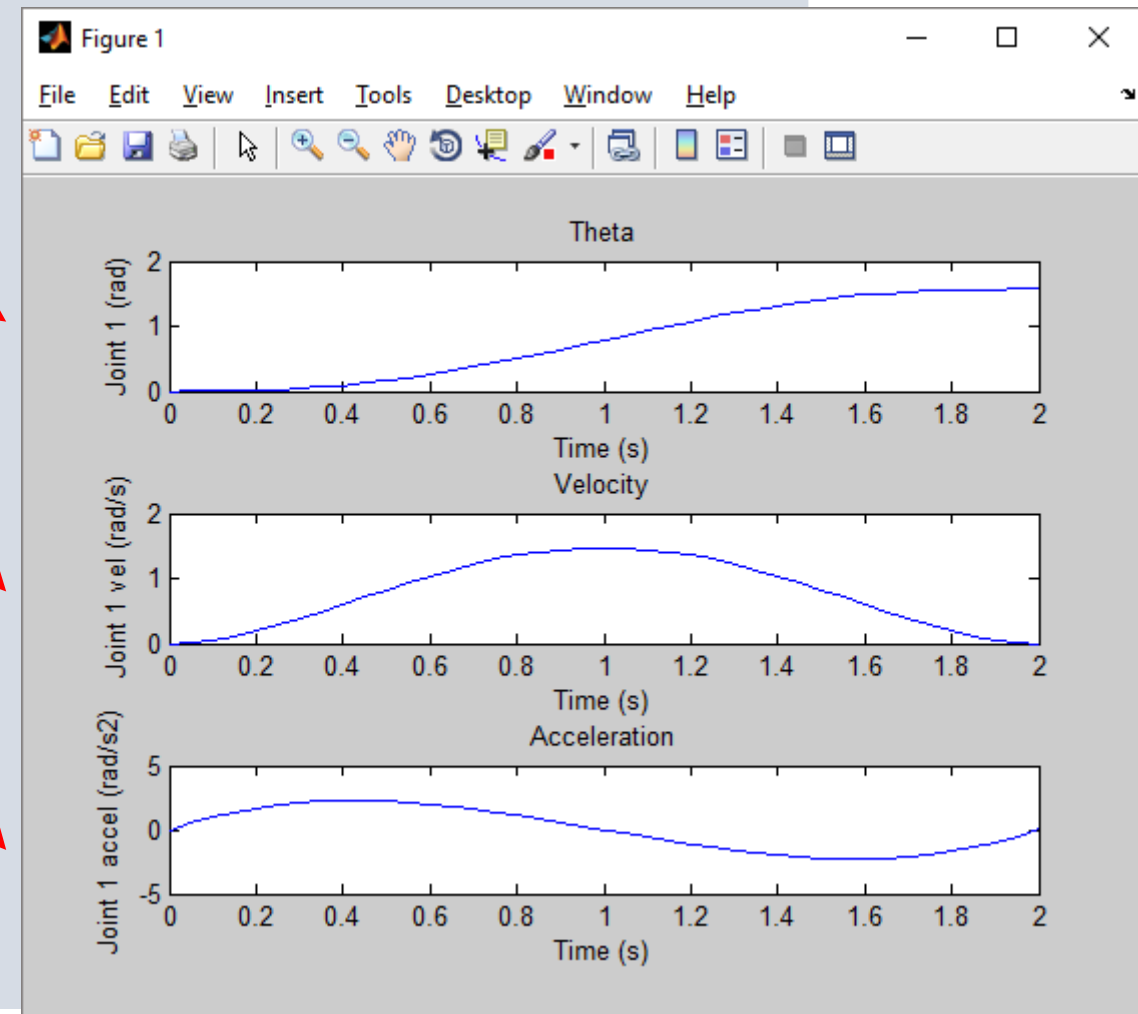
Cinemática direta

```
>> [q, dq, ddq] = jtraj(q0, q1, t); %Posição angular, velocidade angular e aceleração angular
```

```
>> figure(1);  
>> subplot(3,1,1)  
>> plot(t,q(:,1)); %Gráfico da posição angular  
>> title('Theta');  
>> xlabel('Time (s)');  
>> ylabel('Joint 1 (rad)');
```

```
>> subplot(3,1,2)  
>> plot(t,dq(:,1)); %Gráfico da velocidade angular  
>> title('Velocity')  
>> xlabel('Time (s)');  
>> ylabel('Joint 1 vel (rad/s)');
```

```
>> subplot(3,1,3)  
>> plot(t,ddq(:,1)); %Gráfico da aceleração angular  
>> title('Acceleration')  
>> xlabel('Time (s)');  
>> ylabel('Joint 1 accel (rad/s2)');
```



Cinemática direta

```
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas
>> H = fkine(r1,q);%Matriz transformação homogênea
>> figure(1);
>> subplot(3,1,1)
>> plot(t,squeeze(H(1,4,:)));%Gráfico da coordenada x variando no tempo
>> xlabel('Time (s)');
>> ylabel('x (m)');
>> subplot(3,1,2)
>> plot(t,squeeze(H(2,4,:)));%Gráfico da coordenada y variando no tempo
>> xlabel('Time (s)');
>> ylabel('y (m)');
>> subplot(3,1,3)
>> plot(t,squeeze(H(3,4,:)));%Gráfico da coordenada z variando no tempo
>> xlabel('Time (s)');
>> ylabel('z (m)');
>> figure(2);
>> plot(squeeze(H(1,4,:)),squeeze(H(2,4,:)));%Gráfico das coordenadas x e y
>> xlabel('x (m)');
>> ylabel('y (m)');
```

Cinemática direta

```
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas
>> H = fkine(r1,q);%Matriz transformação homogênea
>> figure(1);
>> subplot(3,1,1)
>> plot(t,squeeze(H(1,4,:)));%Gráfico da coordenada x variando no tempo
>> xlabel('Time (s)');
>> ylabel('x (m)');
>> subplot(3,1,2)
>> plot(t,squeeze(H(2,4,:)));%Gráfico da coordenada y variando no tempo
>> xlabel('Time (s)');
>> ylabel('y (m)');
>> subplot(3,1,3)
>> plot(t,squeeze(H(3,4,:)));%Gráfico da coordenada z variando no tempo
>> xlabel('Time (s)');
>> ylabel('z (m)');
>> figure(2);
>> plot(squeeze(H(1,4,:)),squeeze(H(2,4,:)));%Gráfico das coordenadas x e y
>> xlabel('x (m)');
>> ylabel('y (m)');
```

$$H = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

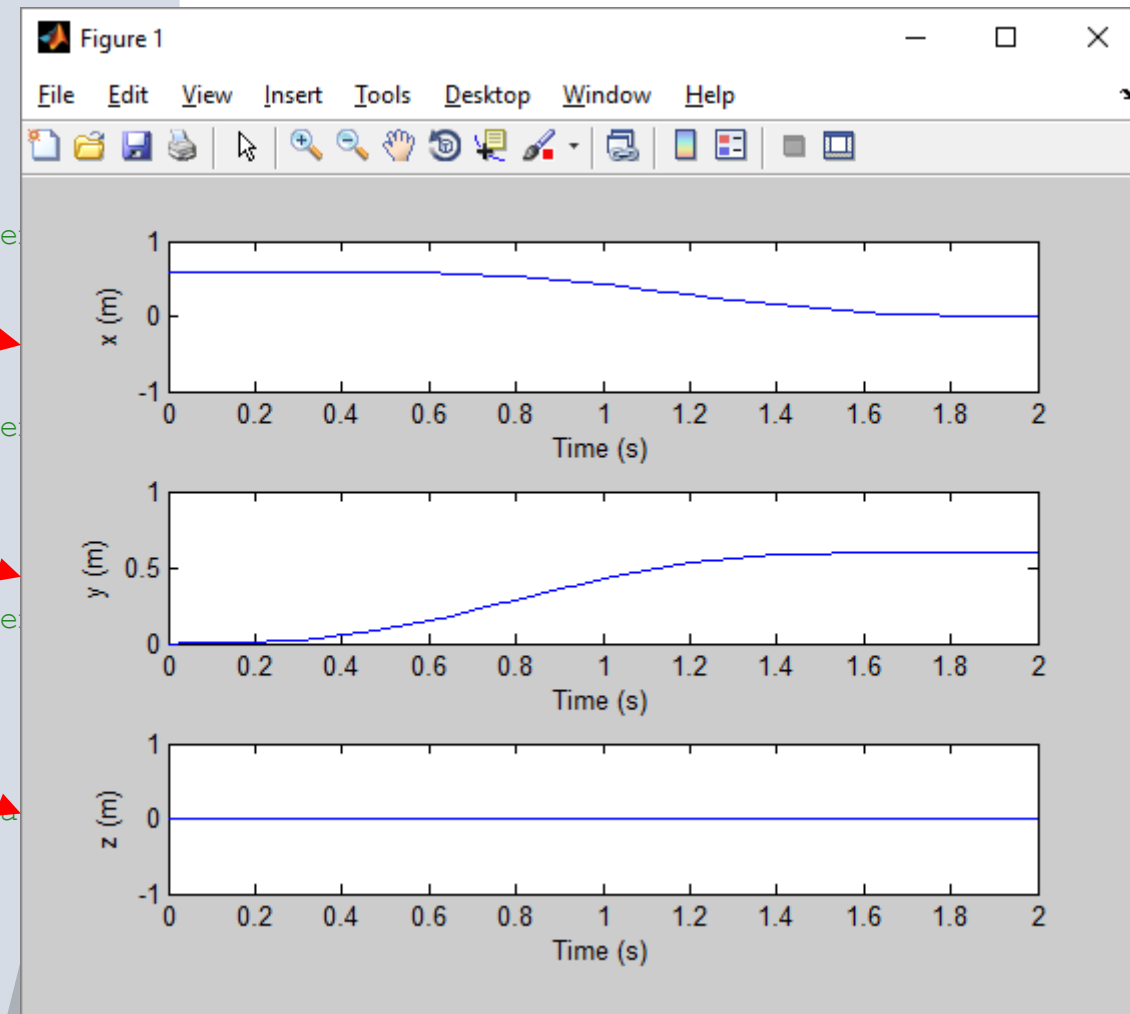
Cinemática direta

```
>> q = jtraj(q0, q1, t); %Trajetória das coordenadas das juntas
>> H = fkine(r1, q); %Matriz transformação homogênea
>> figure(1);
>> subplot(3,1,1)
>> plot(t, squeeze(H(1,4, :))); %Gráfico da coordenada x variando no te
>> xlabel('Time (s)');
>> ylabel('x (m)');

>> subplot(3,1,2)
>> plot(t, squeeze(H(2,4, :))); %Gráfico da coordenada y variando no te
>> xlabel('Time (s)');
>> ylabel('y (m)');

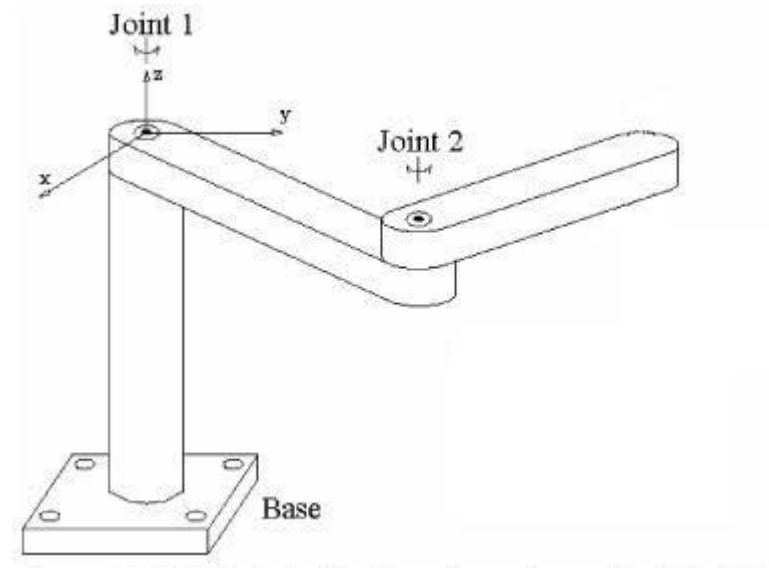
>> subplot(3,1,3)
>> plot(t, squeeze(H(3,4, :))); %Gráfico da coordenada z variando no te
>> xlabel('Time (s)');
>> ylabel('z (m)');

>> figure(2);
>> plot(squeeze(H(1,4, :)), squeeze(H(2,4, :))); %Gráfico das coordenada
>> xlabel('x (m)');
>> ylabel('y (m)');
```



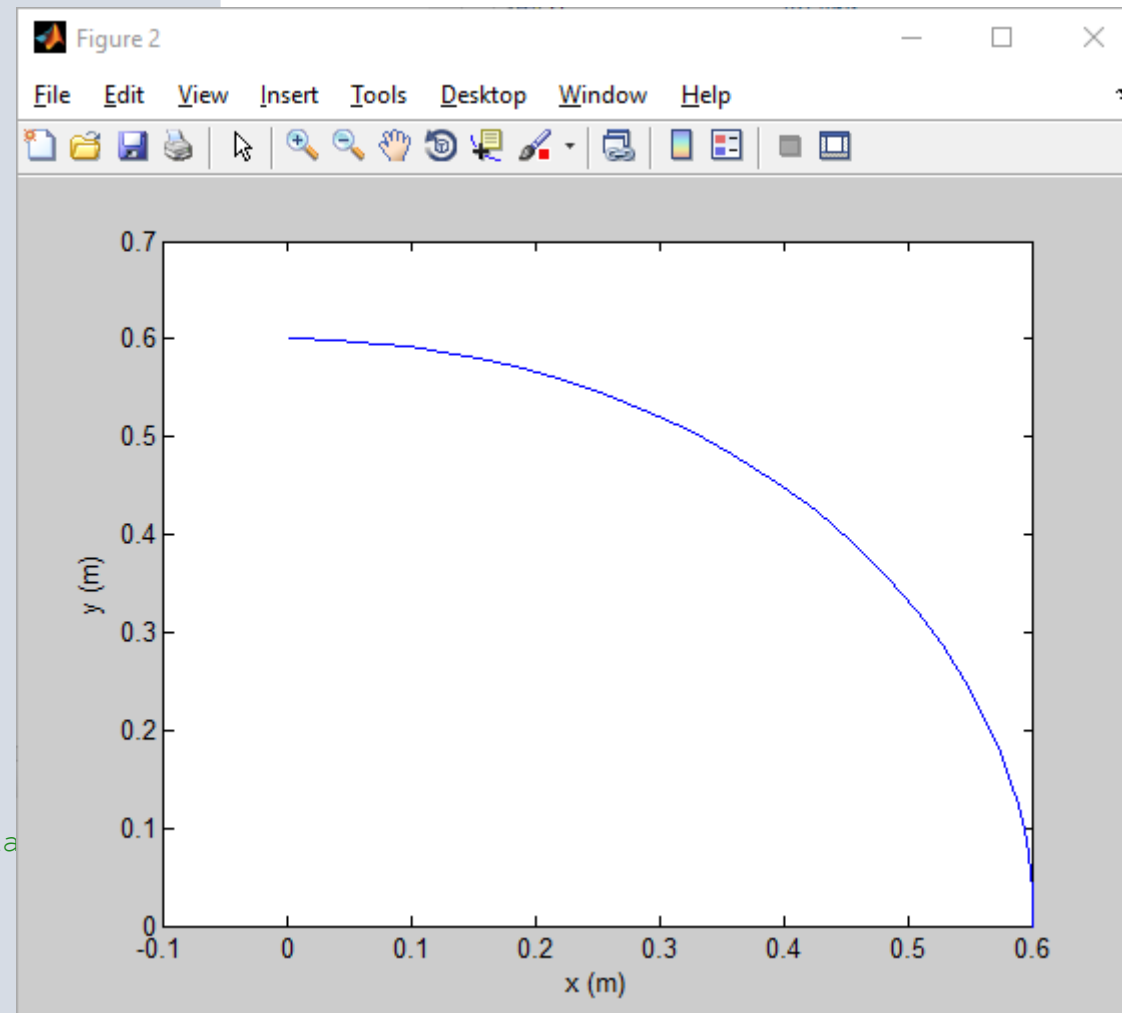
Cinemática direta

```
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas
>> H = fkine(r1,q);%Matriz transformação homogênea
>> figure(1);
>> subplot(3,1,1)
>> plot(t,squeeze(H(1,4,:)));%Gráfico da coordenada x variando no tempo
>> xlabel('Time (s)');
>> ylabel('x (m)');
>> subplot(3,1,2)
>> plot(t,squeeze(H(2,4,:)));%Gráfico da coordenada y variando no tempo
>> xlabel('Time (s)');
>> ylabel('y (m)');
>> subplot(3,1,3)
>> plot(t,squeeze(H(3,4,:)));%Gráfico da coordenada z variando no tempo
>> xlabel('Time (s)');
>> ylabel('z (m)');
>> figure(2);
>> plot(squeeze(H(1,4,:)),squeeze(H(2,4,:)));%Gráfico das coordenadas x e y
>> xlabel('x (m)');
>> ylabel('y (m)');
```



Cinemática direta

```
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas
>> H = fkine(r1,q);%Matriz transformação homogênea
>> figure(1);
>> subplot(3,1,1)
>> plot(t,squeeze(H(1,4,:)));%Gráfico da coordenada x variando no
>> xlabel('Time (s)');
>> ylabel('x (m)');
>> subplot(3,1,2)
>> plot(t,squeeze(H(2,4,:)));%Gráfico da coordenada y variando no
>> xlabel('Time (s)');
>> ylabel('y (m)');
>> subplot(3,1,3)
>> plot(t,squeeze(H(3,4,:)));%Gráfico da coordenada z variando no
>> xlabel('Time (s)');
>> ylabel('z (m)');
>> figure(2);
>> plot(squeeze(H(1,4,:)),squeeze(H(2,4,:)));%Gráfico das coordena
>> xlabel('x (m)');
>> ylabel('y (m)');
```



Animação gráfica

```
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [0 0];%Posição angular inicial (radianos)
>> q1 = [pi/6 pi/2];%Posição angular final (radianos)

>> t = [0:.1:8];%Vetor de tempo
>> q = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

>> plot(r1, q);%Animação do robô r1 na trajetória q
```


Animação gráfica

```
%Movimentos sequenciais
>> l1 = link([0 0.30 0 0 0]);%Criando elo 1
>> l2 = link([0 0.30 0 0 0]);%Criando elo 2

>> r1 = robot({l1 l2})%Criando o robô
>> r1.name = '2R';%Atribuindo nome ao robô

>> q0 = [0 0];%Posição angular inicial (radianos)
>> q1 = [pi/6 0];%Posição angular final (radianos)

>> t = [0:.1:8];%Vetor de tempo
>> qa = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

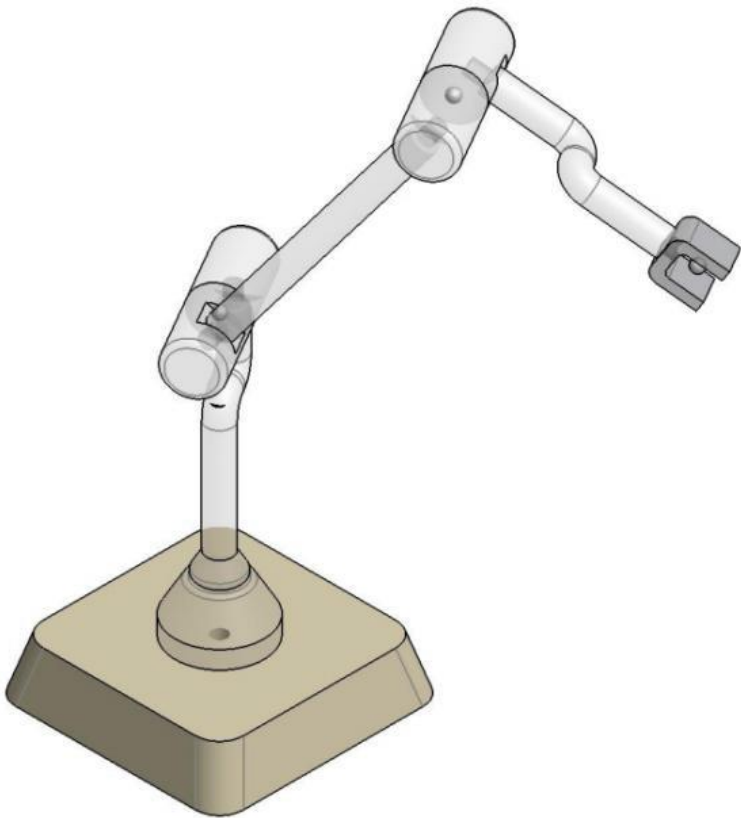
>> plot(r1, qa);%Animação do robô r1 na trajetória qa

>> q0 = [pi/6 0];%Posição angular inicial (radianos)
>> q1 = [pi/6 pi/2];%Posição angular final (radianos)

>> t = [0:.1:8];%Vetor de tempo
>> qb = jtraj(q0, q1, t);%Trajetória das coordenadas das juntas

>> plot(r1, qb);%Animação do robô r1 na trajetória qb
```

Robô manipulador com três juntas rotacionais:



```
>> l1 = link([pi/2 0 0 0.3 0]);%Criando elo 1
>> l2 = link([0 0.3 0 0 0]);%Criando elo 2
>> l3 = link([pi/2 0.3 0 0 0]);%Criando elo 3

>> r2 = robot({l1 l2 l3})%Criando o robô
r2 =

noname (3 axis, RRR)
      grav = [0.00 0.00 9.81]          standard D&H parameters

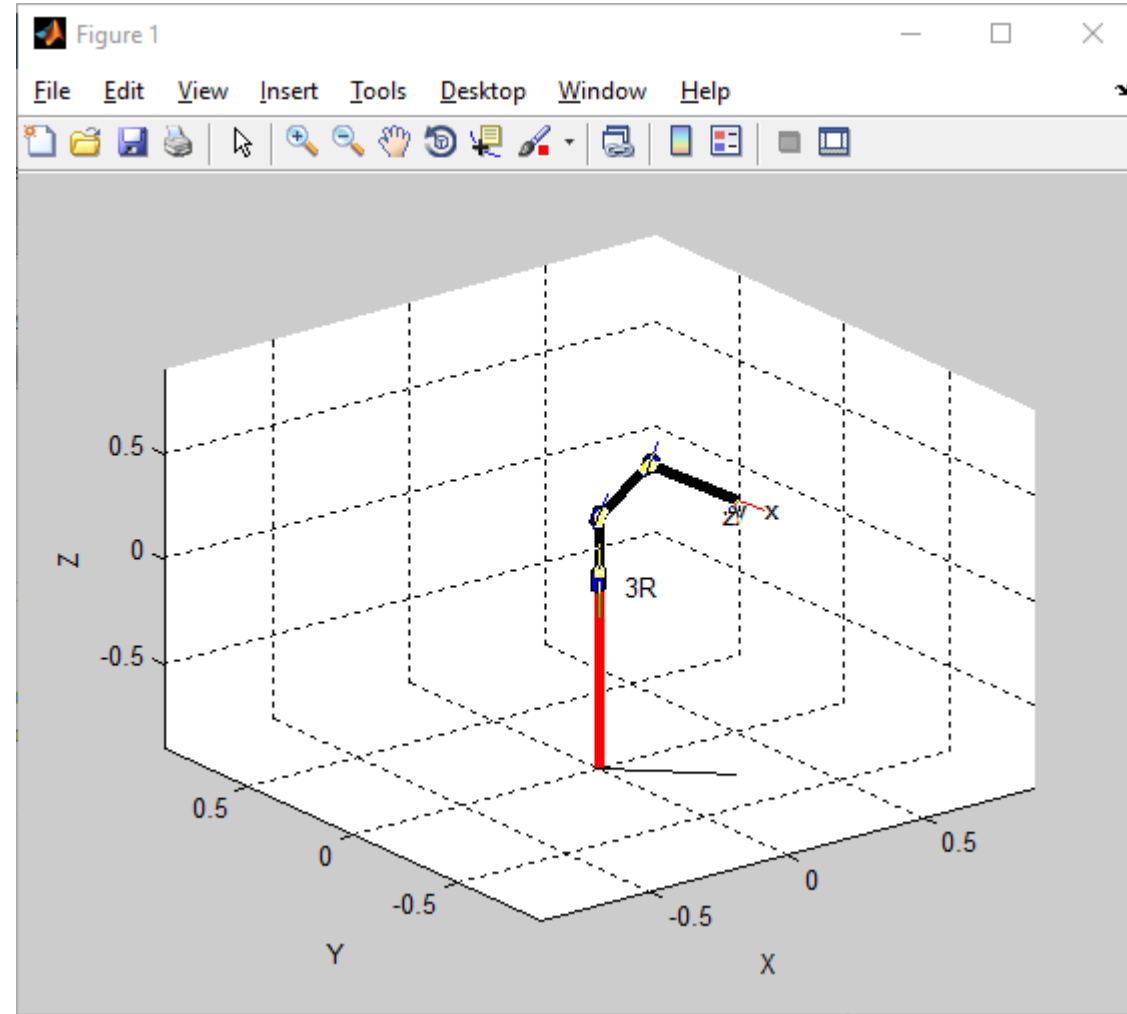
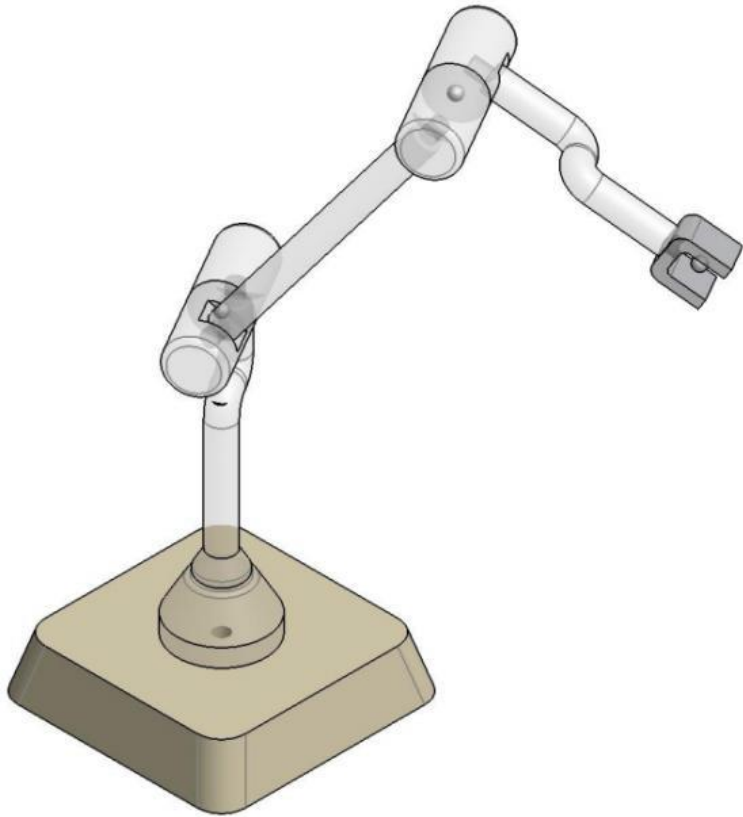
alpha  A      theta  D      R/P
1.570796 0.000000 0.000000 0.300000 R      (std)
0.000000 0.300000 0.000000 0.000000 R      (std)
1.570796 0.300000 0.000000 0.000000 R      (std)

>> r2.name = '3R';%Atribuindo nome ao robô

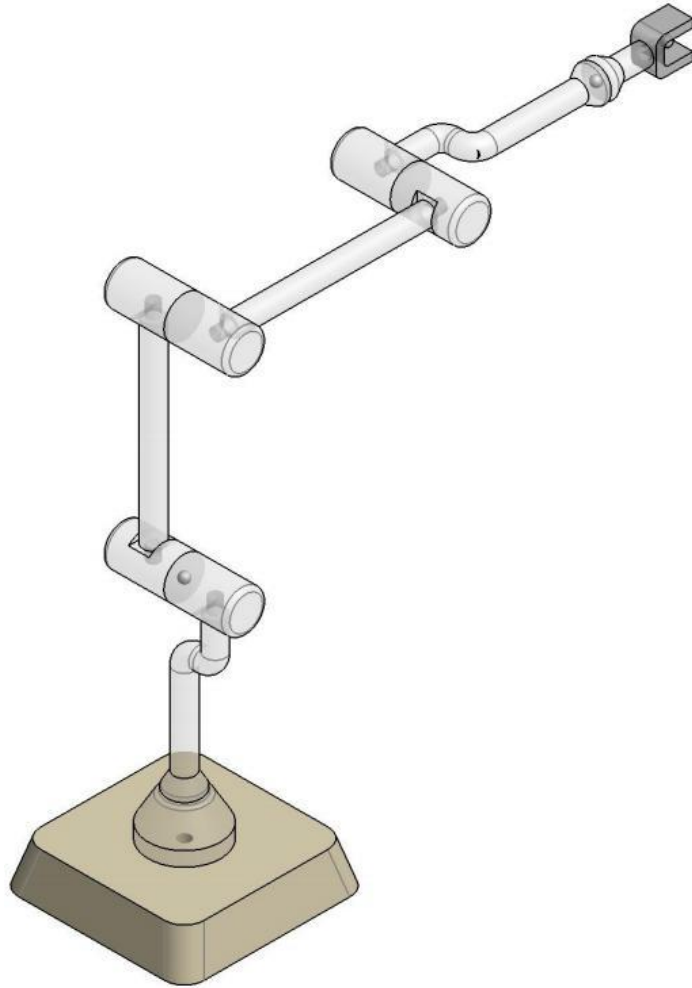
>> q0 = [pi/2 pi/3 pi/4];%Posição angular inicial (radianos)

>> plot(r2, q0);
```

Robô manipulador com três juntas rotacionais:



Robô manipulador com cinco juntas rotacionais:



```
>> l1 = link([pi/2 0 0 0.3 0]);%Criando elo 1
>> l2 = link([0 0.3 0 0 0]);%Criando elo 2
>> l3 = link([0 0.3 0 0 0]);%Criando elo 3
>> l4 = link([pi/2 0 0 0 0]);%Criando elo 4
>> l5 = link([0 0 0 0.3 0]);%Criando elo 5

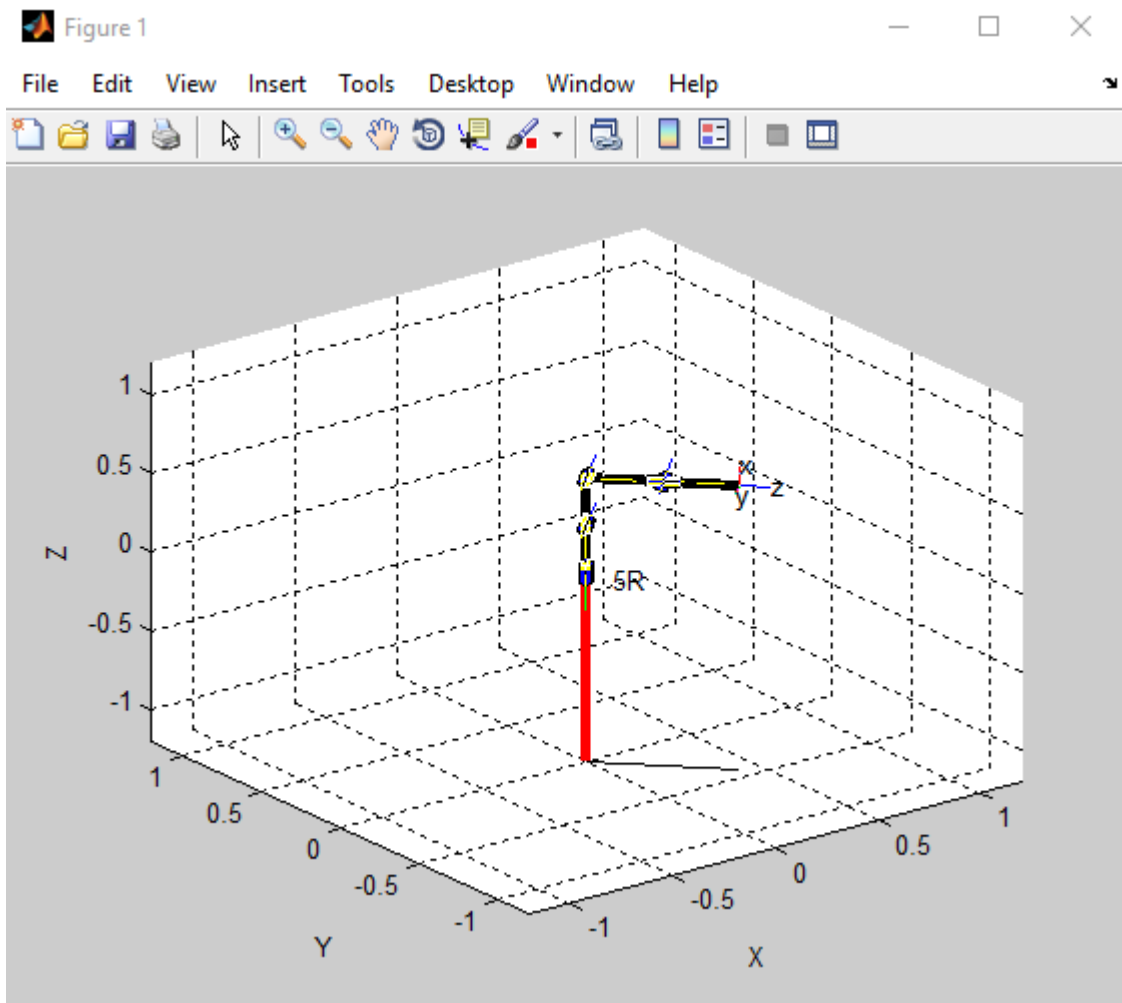
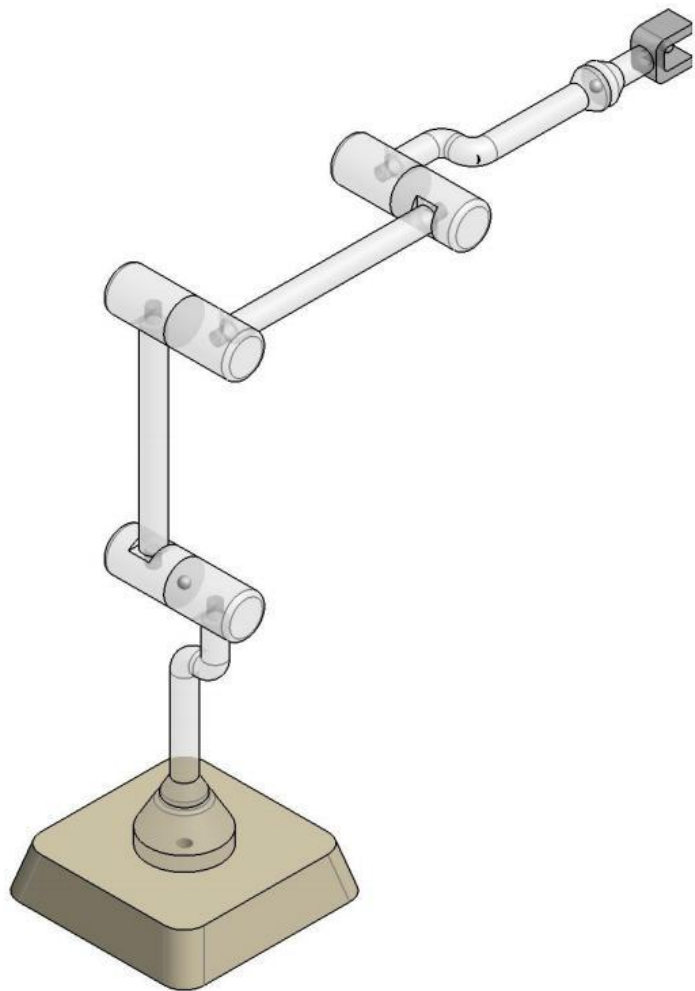
>> r3 = robot({l1 l2 l3 l4 l5})%Criando o robô
r3 =

noname (5 axis, RRRRR)
grav = [0.00 0.00 9.81]          standard D&H parameters

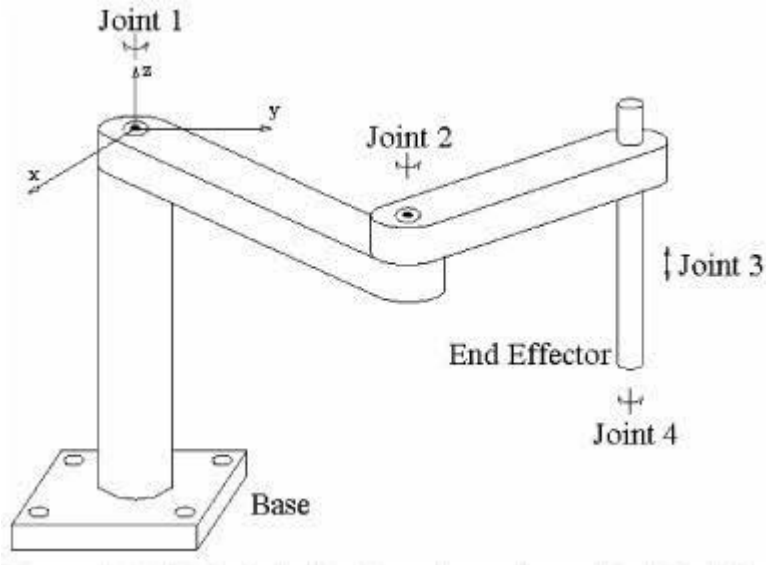
alpha  Athet  a      D      R/P
1.570796 0.000000 0.000000 0.300000 R      (std)
0.000000 0.300000 0.000000 0.000000 R      (std)
0.000000 0.300000 0.000000 0.000000 R      (std)
1.570796 0.000000 0.000000 0.000000 R      (std)
0.000000 0.000000 0.000000 0.300000 R      (std)

>> r3.name = '5R';%Atribuindo nome ao robô
>> q0 = [-pi/4 pi/2 -pi/2 pi/2 0]; %Posição angular inicial
>> plot(r3, q0);
```

Robô manipulador com cinco juntas rotacionais:



Robô SCARA:



```
>> l1 = link([0 0.3 0 0 0]);%Criando elo 1
>> l2 = link([pi 0.3 0 0 0]);%Criando elo 2
>> l3 = link([0 0 0 0 1]);%Criando elo 3
>> l4 = link([0 0 0 0.2 0]);%Criando elo 4
```

```
>> r4 = robot({l1 l2 l3 l4})%Criando o robô
r4 =
```

```
noname (4 axis, RRPR)
```

```
grav = [0.00 0.00 9.81]
```

```
standard D&H parameters
```

alpha	A	theta	D	R/P	
0.000000	0.300000	0.000000	0.000000	R	(std)
3.141593	0.300000	0.000000	0.000000	R	(std)
0.000000	0.000000	0.000000	0.000000	P	(std)
0.000000	0.000000	0.000000	0.200000	R	(std)

```
>> r4.name = 'SCARA';%Atribuindo nome ao robô
```

```
>> q0 = [-pi/4 -pi/6 0 0];%Posição angular inicial
```

```
>> plot(r4, q0);
```

Robô SCARA:

