Robótica e Automação

Raphael Barros Parreira

1 Controle Cinemático de Manipuladores

O exercício tem como objetivo o **controle cinemático** da posição do punho do manipulador, sem se preocupar com a orientação, nesse primeiro momento. Para isso, o manipulador escolhido foi um 7R, com a tabela de Denavit-Hartenberg (tabela 1) e a posição inicial (q_0 representada na figura 1) dadas.

Além disso, o enunciado define 3 trajetórias distintas, define os controles a serem implementados e insere restrições.

Como o objetivo é apenas controlar a posição do manipulador, pode-se ignorar as 3 últimas juntas, já que, pela tabela de DH, elas apenas influenciam na orientação do manipulador. Essa decisão implicará numa série de simplificações feitas no sistema, começando por sempre definir a posição e a velocidade dessas 3 juntas como $zero$.

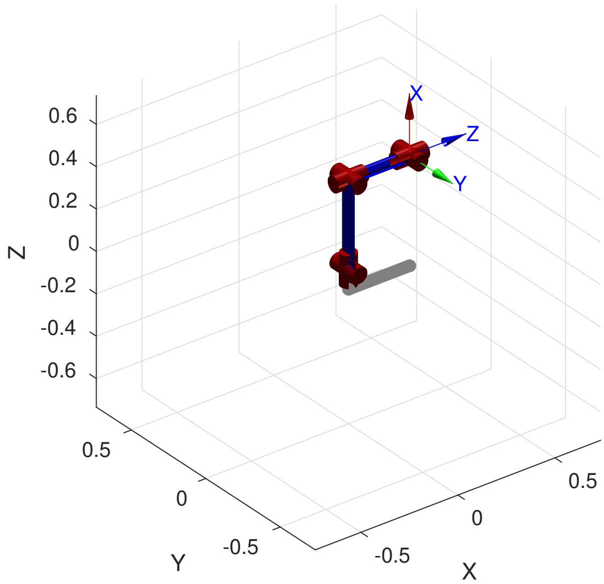


Figure 1: Manipulador Antropomórfico na posição inicial

$$\theta_5 = \theta_6 = \theta_7 = 0$$
$$\dot{\theta}_5 = \dot{\theta}_6 = \dot{\theta}_7 = 0$$
$$q_0 = [0, 0, 0, \pi/2]$$
$$\dot{\theta}_{max} = 3rad/s \Rightarrow \|u_i\| \leq 3 (i = 1, 2, 3)$$

Table 1: Tabela de Denavit-Hartenberg do Manipulador Antropomórfico.

Junta	$\alpha(rad)$	$A(m)$	$\theta(rad)$	$D(m)$	$Offset(rad)$
1	$\pi/2$	0	θ_1	0	0
2	$\pi/2$	0	θ_2	0	π
3	$\pi/2$	0	θ_3	-0.4208	π
4	$\pi/2$	0	θ_4	0	π
5	$\pi/2$	0	θ_5	-0.3143	π
6	$\pi/2$	0	θ_6	0	π
7	π	0	θ_7	0	π

1.1 Controle

No controle cinemático assume-se que a velocidade do manipulador é a variável manipulada e que a **dinâmica** do manipulador pode ser **desprezada**. Além disso, considera-se que a velocidade das juntas responde instantaneamente ao sinal de entrada do motor da mesma, implicando na condição de $u \approx \dot{\theta}$

$$\begin{bmatrix} \vec{v} \\ \vec{\omega} \end{bmatrix} = J(\theta)\dot{\theta} \quad \Rightarrow \quad \begin{bmatrix} \vec{v} \\ \vec{\omega} \end{bmatrix} = J(\theta)u$$

$$u = \dot{\theta} \quad \Rightarrow \quad \dot{x} = J(\theta)u$$

$$x(t) \longrightarrow x_d(t)$$

$$e = x - x_d$$

O objetivo de controle é que o manipulador siga uma trajetória pré-determinada. Portanto, já que se conhece a trajetória, por conseguinte, também se conhece a sua derivada. O que permite adicionar informações no sistema de controle (ação *feed-forward*, tornando possível a **liminação do erro em estado estacionário** ao seguir uma trajetória.

Para que os cálculos sejam feitos, é necessário escolher uma base para expressar os vetores e calcular o Jacobiano. Já que a trajetória dada está expressa na base do sistema inercial, uma boa escolha é usar essa mesma base.

Uma boa ideia para a lei de controle é eliminar as não-linearidades do sistema, incluindo-se a inversa do Jacobiano no sinal de controle.

$$\dot{x} = J\bar{u}$$
$$\bar{u} = J^{-1}[\dot{x_d} + K(x_d - x)]$$
$$\dot{x} = JJ^{-1}[\dot{x_d} + K(x_d - x)]$$
$$\dot{x} - \dot{x_d} = K(x_d - x)$$
$$\dot{e}(t) = Ke(t)$$
$$e \longrightarrow 0, \quad t \longrightarrow \infty$$

Como no caso de um manipulador com 7 juntas, o Jacobiano não é uma matriz quadrada, ele não possui inversa. Sendo assim, se faz necessário o uso da **pseudo-inversa** ($J^{\dagger}$).

$$J^{\dagger} = J^T(JJ^T)^{-1} \quad \Rightarrow \quad JJ^{\dagger} = I$$
$$\bar{u} = J^{\dagger}[\dot{x_d} + K(x_d - x)]$$

1.2 Trajetórias

O enunciado considera 3 trajetórias de referência distintas, com $w_n = 2\pi/10$. Além disso, não se pode usar derivadores puros. Logo, as derivadas das trajetórias foram calculadas como se segue:

a Trajetória a

$$x_d = \begin{bmatrix} 0.075(sin(w_n t) + sin(4w_n t)) + 0.456 \\ 0 \\ 0.075(cos(w_n t) + cos(4w_n t)) + 0.069 \end{bmatrix}$$
$$\dot{x_d} = \begin{bmatrix} 0.075w_n(cos(w_n t) + 4cos(4w_n t)) \\ 0 \\ -0.075w_n(sin(w_n t) + 4sin(4w_n t)) \end{bmatrix}$$

b Trejetória b

$$x_d = \begin{bmatrix} 0.075sin(w_n t) + 0.456 \\ 0 \\ 0.075cos(w_n t) + 0.2 \end{bmatrix}$$
$$\dot{x_d} = \begin{bmatrix} 0.075w_n cos(w_n t) \\ 0 \\ -0.075w_n sin(w_n t) \end{bmatrix}$$

c Trejetória c

$$x_d = \begin{bmatrix} 0.075sin(w_n t) + 0.456 \\ 0.075(sin(w_n t) + cos(w_n t)) \\ 0.075cos(w_n t) + 0.2 \end{bmatrix}$$

$$\dot{x_d} = \begin{bmatrix} 0.075 w_n cos(w_n t) \\ 0.075 w_n (sin(w_n t) - sin(w_n t)) \\ -0.075 w_n sin(w_n t) \end{bmatrix}$$

1.3 Simulação

O sistema representado na topologia da figura 2 está simulando a lei de controle com o feed-forward. Porém, para efeitos de comparação, o exercício pede para desligar essa ação. Para isso, basta apenas excluir o sinal de $xddot$ que é somado depois do ganho.

Cada uma das trajetórias foi simulada com e sem ação do feed-forward.

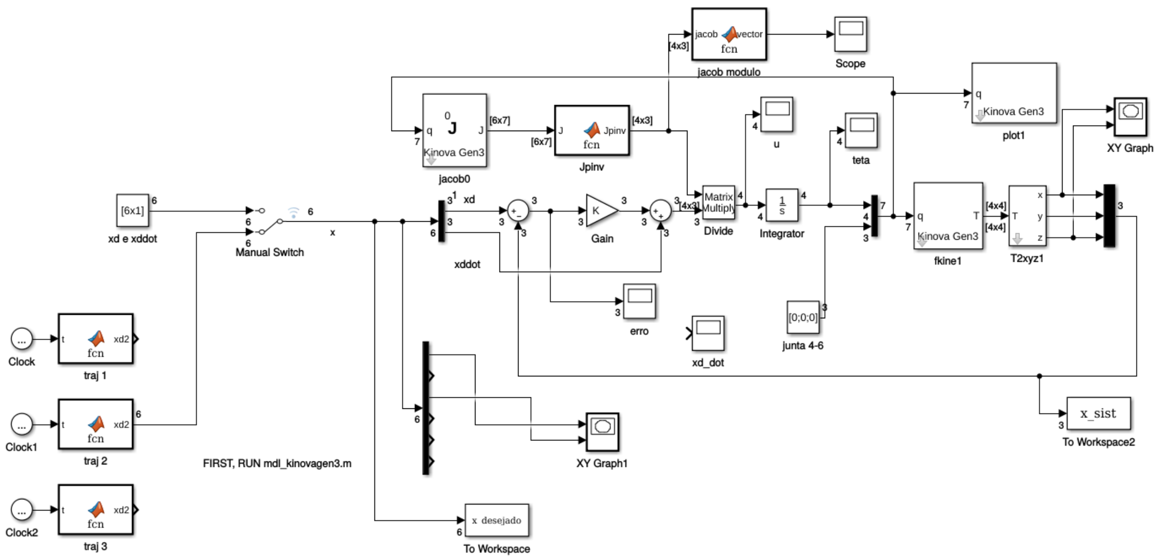


Figure 2: Topologia do Simulink.

1.4 Ganhos

Um ponto importante do trabalho é o ganho K do controlador. Isso se dá porque foi pedido o maior ganho possível atendendo à especificação de $\|u_i\| \leq 3(i = 1, 2, 3)$.

Sendo assim, se faz necessário calcular a ordem de grandeza do ganho e depois se fazer o ajuste fino para cada um dos casos.

$$\|\bar{u}\| \leq 3 \quad \Rightarrow \quad \left\| J^{\dagger}[\dot{x_d} + K(x_d - x)] \right\| \leq 3$$
$$\left\| J^{\dagger} \right\|_{max} [\|\dot{x_d}\|_{max} + K_{base}\|e\|_{max}] = 3$$
$$K_{base} = \frac{\dfrac{3}{\|J^{\dagger}\|_{max}} - \|\dot{x}_d\|_{max}}{\|e\|_{max}}$$
$$\left\| J^{\dagger} \right\|_{max} = 4.277 \qquad \|\dot{x}_d\|_{max} = 0.2356 \qquad \|e\|_{max} = 0.1417$$
$$K_{base} \approx 3.3$$

Os valores $\left\| J^{\dagger} \right\|_{max}$, $\|\dot{x}_d\|_{max}$ e $\|e\|_{max}$ foram calculados graficamente na simulação, considerando o valor no instante inicial ($\theta(0)$), onde se mostrou ser o maior valor em todos os casos.

Após esse passo, foi feita a sintonia fina, chegando o mais próximo possível do valor máximo do sinal de controle em cada um dos casos. Os valores encontrados podem ser vistos na tabela 2.

Table 2: Ganhos para os controles com e sem Feed-forward em todas as 3 trajetórias.

Controles	FB	FF
Trajetória 1	8.91	7.24
Trajetória 2	8.91	8.57
Trajetória 3	8.91	8.57

1.5 Resultados

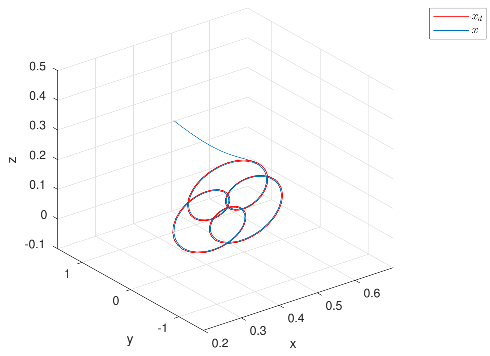
Cada uma das trajetórias tem dois conjuntos de figuras representando os dois controles pedidos (com e sem feed-forward).

Desses conjuntos, a figura (a) exibe a trajetórias desejada (em vermelho) e a trajetória simulada (em azul). A figura (b), por sua vez, exibe o sinal do erro ($x_d - x$). Já a figura (c) representa o sinal de controle u para cada uma das 4 juntas controladas.

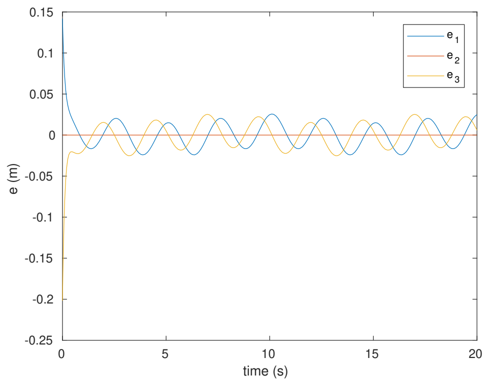
Como esperado, o controle sem feed-forward não é capaz de tirar o erro em regime de estado estacionário, mesmo sendo muito pequeno. Os sinais de controle atendem à restrição pedida, sempre se aproximando o máximo possível do valor máximo.

Pode-se observar na tabela 2 que os ganhos para o controle sem ação feed-forward são iguais para todas as trajetórias, sendo limitado apenas pela restrição do sinal de controle. Porém, mesmo aumentando o valor do ganho não seria possível eliminar o erro em EE.

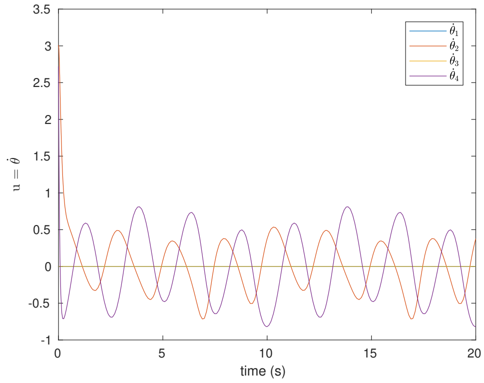
No caso de não ser possível calcular a derivada da trajetória e, dependendo do caso de aplicação do manipulador, o controlador apenas com feed-back de posição pode ser suficiente, já que o erro em EE é pequeno comparado ao tamanho dos elos.



(a) x_d e x

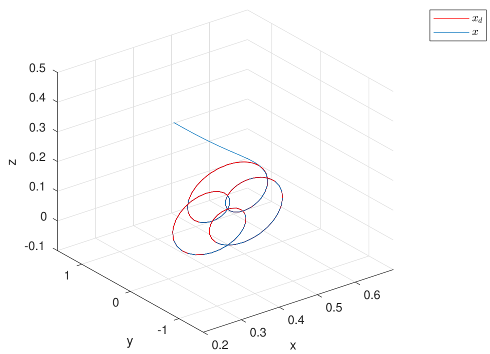


(b) $e = x_d - x$

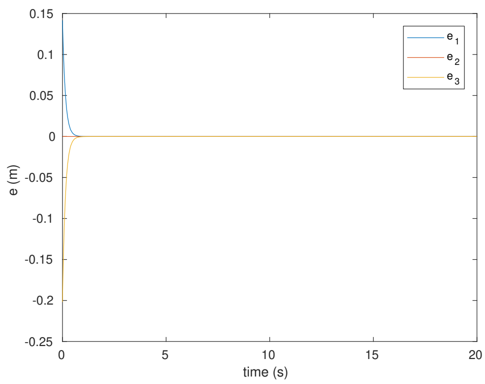


(c) $u = \dot{\theta}$

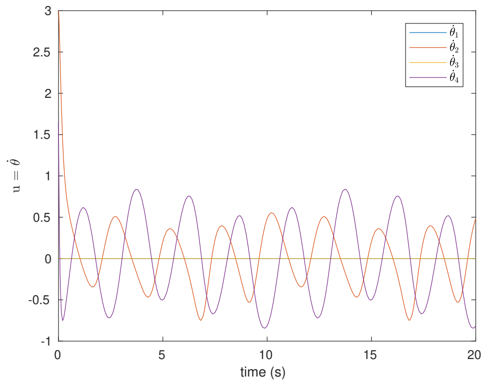
Figure 3: Ex 1: Trajetória a, controle sem FF.



(a) x_d e x

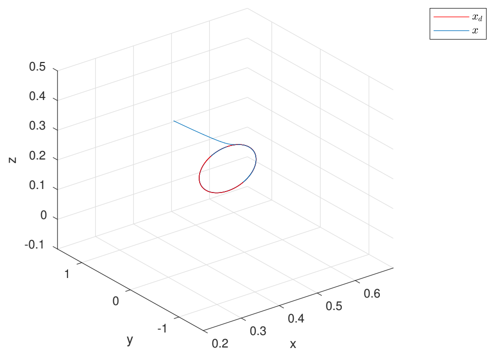


(b) $e = x_d - x$

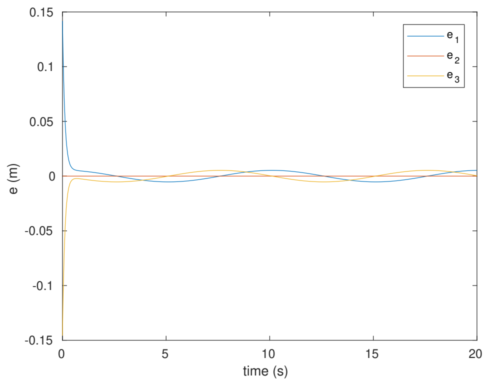


(c) $u = \dot{\theta}$

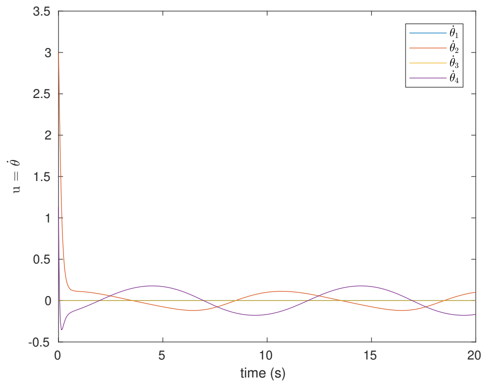
Figure 4: Ex 1: Trajetória a, controle com FF.



(a) x_d e x

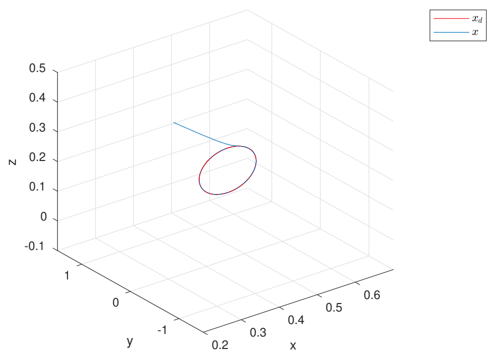


(b) $e = x_d - x$

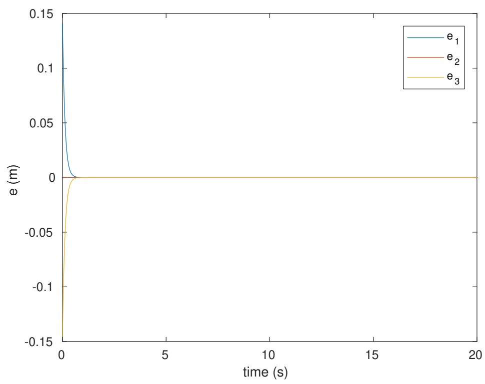


(c) $u = \dot{\theta}$

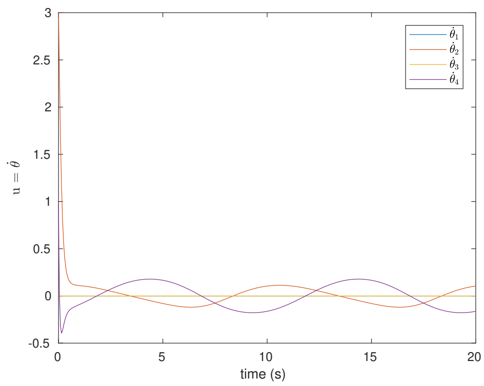
Figure 5: Ex 1: Trajetória b, controle sem FF.



(a) x_d e x

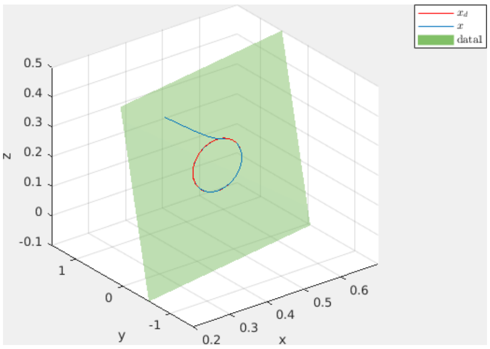


(b) $e = x_d - x$

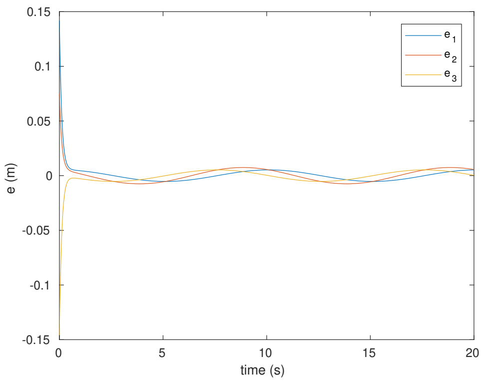


(c) $u = \dot{\theta}$

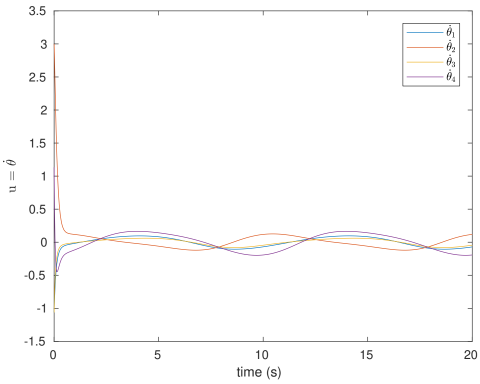
Figure 6: Ex 1: Trajetória b, controle com FF.



(a) x_d e x

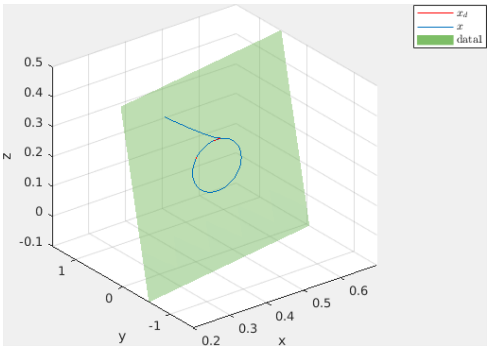


(b) $e = x_d - x$

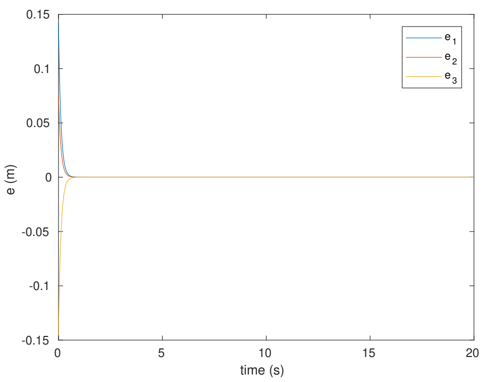


(c) $u = \dot{\theta}$

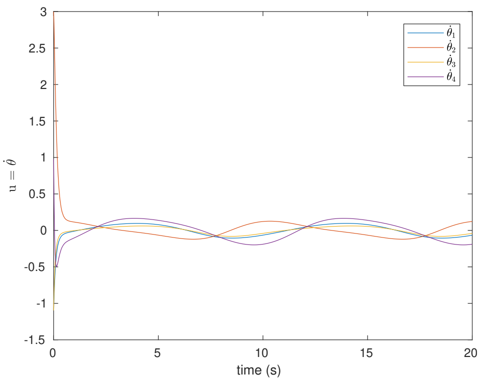
Figure 7: Ex 1: Trajetória c, controle sem FF.



(a) x_d e x



(b) $e = x_d - x$



(c) $u = \dot{\theta}$

Figure 8: Ex 1: Trajetória c, controle com FF.

2 Controle Cinemático de Manipuladores Redundantes

Diferente do manipulador anterior, o objetivo agora é controlar um manipulador redundante. Onde o Jacobiano perde posto por ter mais juntas do que graus de liberdade para a tarefa.

Nesse caso aparece um comportamento muito interessante, o espaço nulo do Jacobiano, que permite otimizar um grau de liberdade adicional.

A função objetivo a ser otimizada pode ser de diversos tipos. Nesse trabalho, as escolhidas serão a **orientação**, a **manipulabilidade** e o **limite das juntas**.

O manipulador proposto para estudar esse comportamento é o manipulador planar 3R, com comprimento dos 3 elos de $0.5m$.

Assim como foi feito para o manipulador anterior, é necessário utilizar o método de Denavit-Hartenberg, cujos parâmetros estão na tabela 3.

A posição inicial escolhida foi $q_0 = [\pi, -\pi/2, -\pi/2]^T$, representada na figura 9.

Neste caso, a trajetória desejada é um círculo de raio $0.25m$ com centro no ponto $[0.250.5]$. Parametrizada por x_d.

$$x_d = \begin{bmatrix} 0.25(1 - cos(\pi t)) \\ 0.25(2 + sin(\pi t)) \end{bmatrix} \qquad t \in [0, 4]$$

Table 3: Tabela de Denavit-Hartenberg do Manipulador Planar 3R.

Junta	$\alpha(rad)$	$A(m)$	$\theta(rad)$	$D(m)$	$Offset(rad)$
1	0	0.5	θ_1	0	0
2	0	0.5	θ_2	0	0
3	0	0.5	θ_3	0	0

2.1 Controle

Este caso é caracterizado por o numero de juntas n ser maior do que o número de graus de liberdade da tarefa m. O que traria a necessidade de usar novamente a pseudo-inversa.

Porém, diferente do caso anterior, esse tem outra particularidade. Como o manipulador é planar com todas as juntas de rotação no eixo z, duas linhas do Jacobiano serão nulas. Além disso, por todas as juntas estarem em todo o tempo no plano XY, a terceira linha do Jacobiano, referente a posição no eixo z, também será zero.

Com isso, pode-se trabalhar com um Jacobiano reduzido chamado de Jacobiano Analítico ($J_a \in R^{3x3}$). Por J_a ser quadrado e não singular, ele é inversível.

Ao reduzir o Jacobiano, também se faz necessário reduzir o sinal de controle ($u \in R^6$), passando a ter dimensão 3.

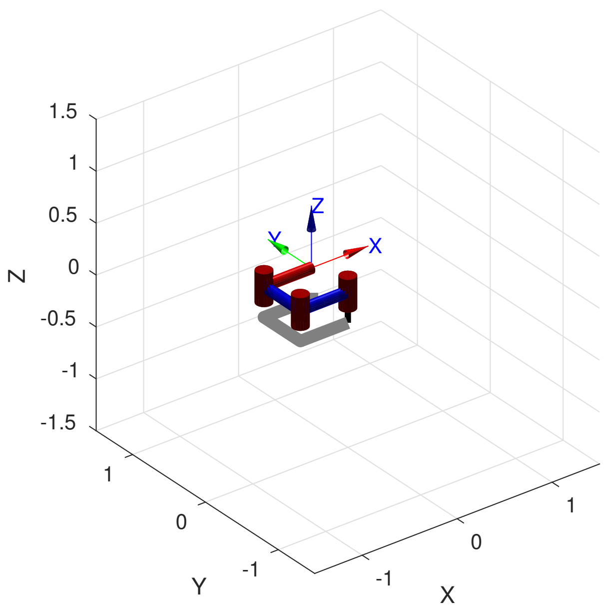


Figure 9: Manipulador Planar 3R na posição inicial

2.1.1 Orientação

No primeiro caso de controle, pede-se para que a orientação também siga uma trajetória desejada (ϕ_d).

$$u = \dot{\theta} \qquad J_a \in R^{3x3}$$

$$\dot{x} = J_a(\theta)\dot{\theta} = J_a(\theta)u \quad \Rightarrow \quad u = J_a(\theta)^{-1}\dot{x}$$

$$u = J_a(\theta)^{-1}\bar{u}$$

$$\bar{u} = \dot{x_d} + K(x_d - x)$$

$$x_d = \begin{bmatrix} p_d \\ \phi_d \end{bmatrix}$$

$$p_d = \begin{bmatrix} 0.25(1 - cos(\pi t)) \\ 0.25(2 + sin(\pi t)) \\ 0 \end{bmatrix}, \qquad \phi_d = \begin{bmatrix} 0 \\ 0 \\ sin(\pi t/24) \end{bmatrix} \quad \Rightarrow \quad \dot{\phi_d} = \begin{bmatrix} 0 \\ 0 \\ \frac{\pi cos(\pi t/24)}{24} \end{bmatrix} \qquad t \in [0, 4]$$

$$x_d = \begin{bmatrix} x \\ y \\ z \\ \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}, \quad x_d = \begin{bmatrix} x \\ y \\ 0 \\ 0 \\ 0 \\ \phi_3 \end{bmatrix}$$

Para se controlar o sistema com 3 graus de liberdade, adiciona-se uma nova parcela no sintal de controle, o espaço nulo do Jacobiano, que é alcançado usando-se a pseudo-inversa dele. A variável μ, o grau de liberdade adicional, é calculada multiplicando-se um ganho K_0 pela derivada da função $\omega(\theta)$ a ser otimizada.

Isso não altera a equação do erro.

$$\bar{u} = J^{\dagger}[\dot{x_d} + K(x_d - x)] + (I - J^{\dagger}J)\mu$$
$$\dot{e} + Ke = 0$$
$$\mu = K_0 \left(\frac{\partial\omega(\theta)}{\partial\theta}\right)^T$$
$$\omega(\theta) = \begin{bmatrix} 0 \\ 0 \\ sin(\pi t/24) \end{bmatrix}$$

$$\phi_d = \theta_1 + \theta_2 + \theta_3 = sin(\pi t/24)$$
$$\omega(\theta) = \begin{bmatrix} 0 \\ 0 \\ sin(\pi t/24) \end{bmatrix} \quad \Rightarrow \quad \dot{\omega}(\theta) = \begin{bmatrix} 0 \\ 0 \\ \frac{\pi cos(\pi t/24)}{24} \end{bmatrix}$$

2.2 Funções Objetivo

2.2.1 Orientação

Como observado anteriormente, uma das funções a ser otimizada é a orientação, que deve seguir uma trajetória parametrizada por ϕ_d.

$$\phi_d = \theta_1 + \theta_2 + \theta_3 = sin(\pi t/24)$$
$$\omega(\theta) = \begin{bmatrix} 0 \\ 0 \\ sin(\pi t/24) \end{bmatrix} \quad \Rightarrow \quad \dot{\omega}(\theta) = \begin{bmatrix} 0 \\ 0 \\ \frac{\pi cos(\pi t/24)}{24} \end{bmatrix}$$