SIMULAÇÃO DE TRAJETÓRIAS DE UM ROBÔ PLANAR COM DOIS ELOS CONSIDERANDO CONTROLE TEMPORAL

Lidia Martinelli de Oliveira[1]; Alexandre Molter[2]

[1]Universidade Federal de Pelotas – lidia.martinelli@ufpel.edu.br
[2]Universidade Federal de Pelotas – alexandre.molter@ufpel.edu.br

1. INTRODUÇÃO

Na robótica, o planejamento das trajetórias de um robô é essencial para sua operacionalidade. As trajetórias devem ser suaves e precisas nas aplicações em robótica. Este estudo aborda a utilização de robôs planares de dois elos, e dois graus de liberdade rotacionais. É considerado um método polinomial para planejar as trajetórias do robô e calcular os ângulos das juntas necessários para seguir essas trajetórias através da cinemática inversa.

Para o planejamento de trajetórias, utilizamos um polinômio interpolador cúbico para os ângulos das juntas ao longo do tempo, garantindo uma transição suave entre os pontos inicial e final da trajetória. Os cálculos são fundamentados nas técnicas de cinemática direta e inversa para o cálculo das posições do efetuador e dos ângulos das juntas (CRAIG, 2005; SPONG, HUTCHINSON, & VIDYASAGAR, 2006). Este trabalho é uma sequência dos trabalhos DIEHL, OLIVEIRA e MOLTER (2023, 2024).

Simulações computacionais foram realizadas em MATLAB, e apresentados os resultados das trajetórias temporais e a trajetória do efetuador final no plano cartesiano.

2. METODOLOGIA

Neste trabalho, foi desenvolvido um código MATLAB para o planejamento e simulação da trajetória de um robô planar com dois elos, levando em consideração o controle temporal. O processo de planejamento inicia-se com a definição dos parâmetros físicos do robô, incluindo os comprimentos dos elos L_1 e L_2, seguido pela especificação da trajetória desejada. A trajetória considerada é uma circunferência com centro em $(0, 1)$ e raio de 2 unidades, garantindo uma resolução espacial adequada através da definição de um número suficiente de pontos ao longo da trajetória.

Para assegurar transições suaves entre os estados iniciais e finais da trajetória, foi adotado um polinômio interpolador cúbico dos ângulos das juntas. O polinômio cúbico utilizado é dado por:

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3,$$

onde a_0, a_1, a_2 e a_3 são coeficientes determinados com base nas condições iniciais e finais da trajetória. As expressões para a velocidade angular e aceleração angular são obtidas respectivamente por:

$$\frac{d\theta(t)}{dt} = a_1 + 2a_2 t + 3a_3 t^2,$$

$$\frac{d^2\theta(t)}{d^2t} = 2a_2 + 6a_3 t.$$

Os coeficientes são calculados para uma trajetória circular com ângulo inicial $\theta_0 = 0$ e ângulo final $\theta_f = 2\pi$, ao longo de um tempo total de 10 segundos. Estes coeficientes são dados por:

$$a_0 = \theta_0,$$
$$a_1 = 0,$$
$$a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0),$$
$$a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0).$$

As coordenadas $x(t)$ e $y(t)$ da trajetória circular são determinadas para cada ponto no tempo utilizando as seguintes equações paramétricas:

$$x(t) = x_0 + r\,cos(\theta(t)),$$
$$y(t) = y_0 + r\,sen(\theta(t)).$$

Essas equações descrevem as coordenadas $x(t)$ e $y(t)$ da trajetória desejada que o efetuador do robô deve seguir ao longo do tempo t. Nesta simulação específica, a trajetória é uma circunferência com centro em (x_0, y_0), e raio r. Elas são independentes da configuração física do robô (comprimentos dos elos) e apenas definem a trajetória no espaço cartesiano que o efetuador deve seguir.

Por outro lado, para calcular as coordenadas reais x e y do efetuador em função dos ângulos das juntas do robô, utilizam-se as equações da cinemática direta:

$$x = L_1 cos(\theta_1) + L_2 cos(\theta_1 + \theta_2),$$
$$y = L_1 sen(\theta_1) + L_2 sen(\theta_1 + \theta_2).$$

Essas equações são fundamentais para determinar a posição do efetuador no espaço cartesiano dado um conjunto específico de ângulos das juntas θ_1 e θ_2. Elas levam em consideração a estrutura física do robô, vinculando os parâmetros físicos do robô (comprimentos dos elos e ângulos das juntas) à posição do efetuador.

Para resolver os ângulos θ_1 e θ_2 a partir das coordenadas $x(t)$ e $y(t)$, utilizamos as equações da cinemática inversa:

$$D = \frac{x^2 + y^2 - L_1^2 - L_2^2}{2 \cdot L_1 \cdot L_2}, \quad \theta_2 = acos(D),$$
$$\theta_1 = atan2(y, x) - atan2(L_2 sin(\theta_2), L_1 + L_2 cos(\theta_2)).$$

A simulação inclui uma visualização em tempo real da trajetória do efetuador e do movimento dos elos do robô. A posição, velocidade e aceleração das juntas são monitoradas ao longo do tempo. Adicionalmente, a área de trabalho do robô é verificada continuamente para garantir que o efetuador permaneça dentro dos limites operacionais permitidos.

3. RESULTADOS E DISCUSSÃO

A simulação mostra que o robô pode seguir a trajetória circular definida de maneira eficaz. As figuras geradas mostram a posição, velocidade e aceleração dos ângulos das juntas ao longo do tempo, fornecendo uma visão detalhada do desempenho do planejamento de trajetória. No entanto, o controle temporal é crucial para a precisão da trajetória. Por exemplo, tanto a trajetória circular (Figura 1) quanto para a trajetória de limaçon (Figura 2) foram simuladas em um intervalo de 10 segundos. Embora a trajetória de limaçon tenha um número adicional de voltas, o controle temporal assegurou que a simulação de ambas as trajetórias fosse realizada dentro do intervalo estabelecido.

Se o efetuador sair da área de trabalho, a simulação é interrompida, garantindo que o robô opere dentro de suas limitações físicas. O código MATLAB permite ajustar os parâmetros e verificar a precisão e suavidade da trajetória planejada.

Além disso, é possível alterar o desenho do robô modificando as equações que controlam a trajetória. Por exemplo, utilizando diferentes parâmetros no código MATLAB, o robô pode desenhar várias trajetórias, como círculos, limaçons e cardioides.

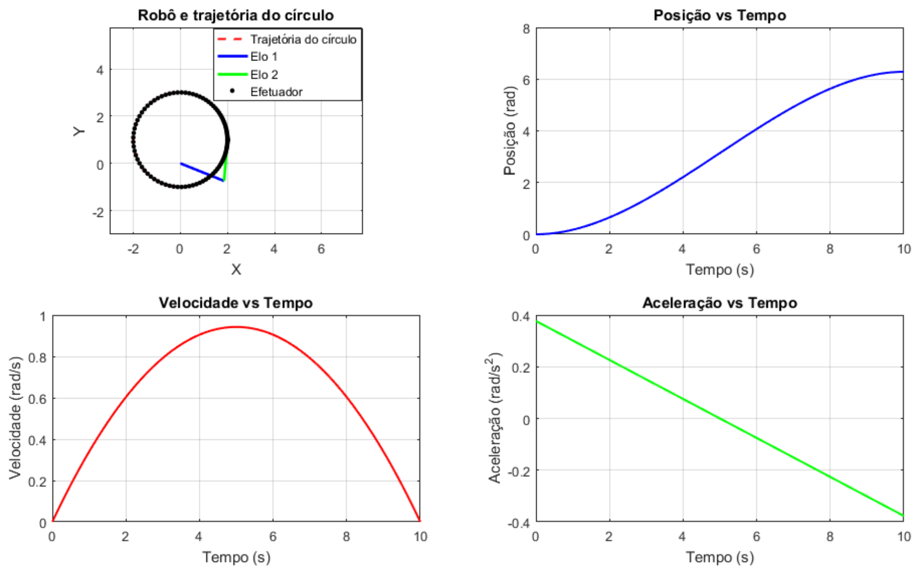

Figura 1: trajetória de um círculo no plano cartesiano e as correspondentes posição, velocidade e aceleração do efetuador final.

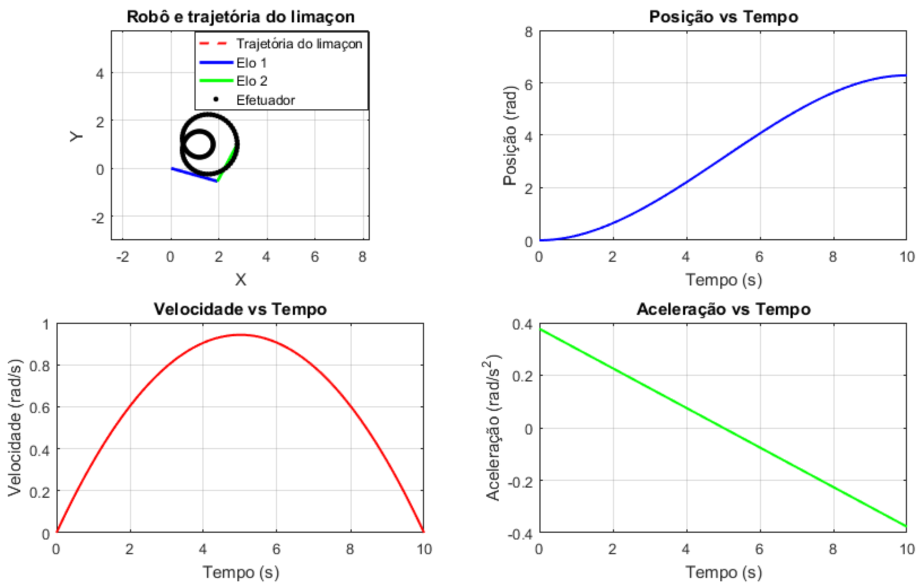

Figura 2: trajetória de um limaçon no plano cartesiano e as correspondentes posição, velocidade e aceleração do efetuador final.

4. CONCLUSÕES

Neste estudo foram implementados códigos computacionais para o controle temporal da trajetória de um robô planar de dois elos. A implementação mostrou a eficiência do uso de polinômios interpoladores cúbicos para criar trajetórias suaves nestes robôs.

A aplicação das técnicas de cinemática direta e inversa permitiu estabelecer o espaço de trabalho do robô e garantiu que o efetuador se mantivesse dentro da área de trabalho. As simulações em MATLAB mostraram que as trajetórias de círculo e de limaçon foram realizadas com precisão dentro do intervalo de 10 segundos. O código MATLAB desenvolvido é adaptável para diferentes trajetórias e configurações de robôs, oferecendo uma base sólida para futuras pesquisas e aplicações em robótica.

5. REFERÊNCIAS BIBLIOGRÁFICAS

SPONG, Mark W.; HUTCHINSON, Seth; VIDYASAGAR, Mathukumalli. **Robot modeling and control**. 2ª ed., New York: John Wiley & Sons, 2020.

CRAIG, John J. **Introduction to robotics**. Upper Saddle River: Pearson Education, 2006.

DIEHL, Pedro Henrique; OLIVEIRA, Lídia Martinelli de; MOLTER, Alexandre. PROGRAMACÃO ORIENTADA A OBJETOS APLICADA À MODELAGEM MATEMÁTICA DE UM ROBÔ. In: **Anais do Encontro Regional de Matemática Aplicada e Computacional do Rio Grande do Sul**. Pelotas(RS) Campus Capão do Leão - UFPel, 2023. Disponível em: https//www.even3.com.br/anais/xiermacrs/628902-PROGRAMACAO-ORIENTADA-A-OBJETOS-APLICADA-A-MODELAGEM-MATEMATICA-DE-UM-ROBO. Acesso em: 03/09/2024

DIEHL, Pedro Henrique; OLIVEIRA, Lídia Martinelli de; MOLTER, Alexandre. Modelagem Matemática de um Robô com Simulações Computacionais usando Programação Orientada a Objetos. **Revista Interdisciplinar de Pesquisa em Engenharia**, v. 9, n. 2, p. 46–56, 2024. Disponível em: https://periodicos.unb.br/index.php/ripe/article/view/52313.