

MODELAGEM DE TRAJETO EM MANIPULADOR SCARA

ALINE TIMM¹; RÉGIS DE QUADROS²; KRISTOFER KAPPEL³

¹ Universidade Federal de Pelotas – aline.timm@ufpel.edu.br

² Universidade Federal de Pelotas – regis.quadros@ufpel.edu.br

³ Universidade Federal de Pelotas – kskappel@inf.ufpel.edu.br

1. INTRODUÇÃO

A robótica é uma das áreas mais estudadas e que mais atrai olhares acadêmicos na atualidade, seja nas engenharias, na programação ou até mesmo na educação, por sua ampla gama de aplicações e crescente necessidade de otimização.

Neste trabalho veremos aplicações diretas destes métodos no âmbito de modelagem de trajetórias e otimização de movimento, bem como possíveis aplicações na análise e prevenção de desgastes do sistema, utilizando o método de otimização de trajetórias do Gradiente conjugado. Este método pode ser usado para otimizar trajetórias em manipuladores robóticos, minimizando critérios como o tempo de execução, consumo de energia, ou a suavidade da trajetória, neste trabalho será proposto a otimização de trajetórias de um manipulador que deve seguir um trajeto predeterminado utilizando cinemática direta.

Para a construção do algoritmo utilizado para gerar os resultados apresentados utilizou-se a linguagem Python, e seguiu-se a metodologia para os cálculos de movimentação de manipuladores disponível em SPONG; MARK (2020).

Além disso, para a utilização e aplicação dos métodos de otimização, foram aplicados os conceitos apresentados em NOCEDAL; Jorge (2006) e para a interpretação dos resultados, foi fundamental a base teórica disponível em BERTSEKAS, Dimitri P (1999).

2. METODOLOGIA

O estudo dos movimentos realizados por robôs pode ser descrito de maneira simples e prática por meio de ferramentas razoavelmente básicas da álgebra linear, como multiplicação de matrizes e rebatimento de ângulos, por exemplo. Quando estuda-se o movimento de um manipulador, precisa-se levar em consideração algumas definições, dentre elas, saber como ocorre o movimento.

O movimento mais simples que pode ser realizado é o linear, este movimento é comumente encontrada em atuadores lineares ou robôs cartesianos, por exemplo, em contrapartida temos os movimentos angulares ou de revolução, este tipo de movimento ocorre quando temos alguma articulação no robô, essa rotação por sua vez permite que o robô produza movimentos angulares ao em vez de somente lineares.

Os manipuladores robóticos podem ser diferenciados quanto ao tipo de movimento em dois grupos: Os robôs cartesianos e os robôs Scara (*Selective Compliant Articulated Robot for Assembly*).

Segundo SPONG; MARK (2020), a descrição cinemática do manipulador cartesiano é a mais simples de todos os manipuladores. Manipuladores cartesianos são úteis para aplicações de montagem em mesas (superfícies lisas e planas) e, como robôs de pórtico, para transferência de materiais ou carga.

Quando se fala de movimento angular, tem-se um manipulador Scara, e é comum estabelecer um sistema de coordenadas fixo, chamado de referencial do mundo ou base, ao qual todos os objetos, incluindo o manipulador, são referenciados. Neste caso, se estabelece o referencial de coordenadas base $o_0x_0y_0$ na base do robô.

Neste caso, tomando conhecimentos de geometria básica, é possível deduzir as equações necessárias para representar as posições das coordenadas x, y , da seguinte forma:

$$x = x_2 = \alpha_1 \cos \theta_1 + \alpha_2 \cos(\theta_1 + \theta_2) \quad (1)$$

$$y = y_2 = \alpha_1 \sin \theta_1 + \alpha_2 \sin(\theta_1 + \theta_2) \quad (2)$$

onde α_1 e α_2 são respectivamente os comprimentos dos dois elos, ou seja, o comprimento total dos “braços” do manipulador. Além disso, a orientação do referencial em relação ao referencial da base é dada pelos cossenos dos eixos x_2 e y_2 em relação aos eixos x_0 e y_0 , ou seja, tem-se:

$$x_0 \cdot x_2 = \cos(\theta_1 + \theta_2) ; \quad x_2 \cdot y_0 = -\sin(\theta_1 + \theta_2) \quad (3)$$

$$y_2 \cdot x_0 = \sin(\theta_1 + \theta_2) ; \quad y_2 \cdot y_0 = \cos(\theta_1 + \theta_2) \quad (4)$$

De posse destas equações, monta-se as chamadas matrizes de orientação, utilizadas na aplicação do método para, de fato, calcular o posicionamento e a movimentação dos manipuladores robóticos.

As equações 1 e 2 são equações de cinemática direta, ou seja, neste caso o robô realizou um movimento saindo de um ponto a e indo até um ponto b , o procedimento é calcular os ângulos e a trajetória que ele utilizou para realizar este caminho.

Tendo em vista as exigências e as definições apresentadas até aqui, aplicamos no presente estudo o método do Gradiente Conjugado para otimizar a trajetória de um manipulador do tipo Scara, que tenha como objetivo percorrer um trajeto pré-determinado.

O que almeja-se otimizar nesta tarefa é o tempo gasto para realizar um trajeto, levando em consideração as variáveis que irão interferir nesta tarefa, neste caso vamos levar em consideração um coeficiente de folga em um dos componentes do manipulador, esta folga será responsável por causar uma perturbação no sistema, este problema deve ser sanado utilizando o método do gradiente conjugado para corrigir adequadamente o trajeto.

Vale ressaltar que o Gradiente conjugado é excelente em otimização de trajetórias, todavia ele pode não ser o mais adequado para todas as situações,

especialmente para problemas com trajetórias altamente não lineares e complexas.

3. RESULTADOS E DISCUSSÃO

Os resultados obtidos podem ser verificados primeiramente na Figura 1, onde vê-se o percurso do manipulador para desenhar um círculo e um quadrado, figuras simples e que demandam de pouca otimização no trajeto, mas assim exemplificam de maneira concisa os conceitos apresentados.

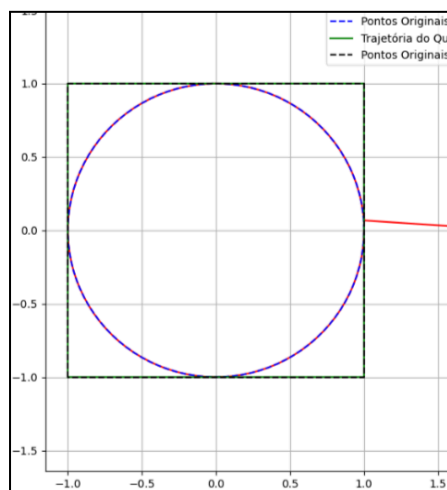


Figura 1 - Trajetória simulada com o método do gradiente conjugado. Fonte: A autora.

Agora, pode-se observar o que acontece ao tentar percorrer uma trajetória não linear e mais complexa, com mudanças mais bruscas de direção, como por exemplo escrever uma palavra completa. À medida que o manipulador faz o trajeto ele precisa minimizar o tempo, utilizando o método do gradiente conjugado para isso.

A seguir, na Figura 2, estão disponíveis os resultados obtidos aplicando 100 e 10000 iterações no algoritmo, respectivamente da esquerda para direita.

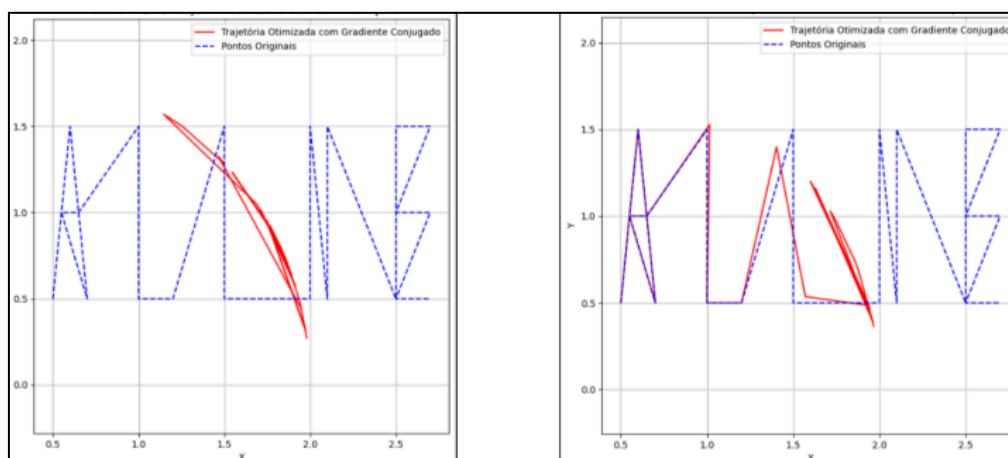


Figura 2 - Trajetória simulada com 100 e 10000 iterações. Fonte: A autora.

Os problemas relacionados a convergência do método e que observados nas Figuras 2 e 3, podem ser explicados segundo BERTSEKAS; DIMITRI P, (1999):

Considerando que a função que deseja-se minimizar ou otimizar é chamada de f , e idealmente se almeja que a sequência gerada x^k convirja para um mínimo global, porém de fato, um método de gradiente é guiado para baixo pela forma de f próximo à iteração atual, enquanto ignora a estrutura global de f , e assim, pode facilmente ser atraído para qualquer tipo de mínimo, global ou não.

4. CONCLUSÕES

O método de gradiente conjugado, usado para otimizar trajetórias em espaços complexos, como no planejamento de caminhos, otimização de tempo de percurso e desvio de obstáculos, entrega resultados satisfatórios quando aplicado a trajetórias simplificadas, ou quando as variáveis que influenciam no trajeto são previsíveis e lineares.

Quando aplicado a uma trajetória não linear e sujeita a mais variáveis de influência, o método demora mais para convergir mostrando a necessidade de milhares de interações para corrigir adequadamente o trajeto.

Como perspectivas futuras, considera-se a melhora do método com a combinação de ferramentas de aprendizado de máquina que auxiliem no processo de mapeamento das variáveis, bem como correção e minimização dos erros de trajeto, desta objetivando é diminuir as interações necessárias para a correção do trajeto.

5. REFERÊNCIAS BIBLIOGRÁFICAS

SPONG, Mark W, et al. **Robot Modeling and Control**. Hoboken, Nj, John Wiley & Sons, Inc, 2020.

NOCEDAL, Jorge, and Stephen J Wright. **Numerical Optimization**. New York, Springer, 2006.

BERTSEKAS, Dimitri P. **Nonlinear Programming**. Goodman Publishers, 1999.

SCHEFFE, Patrick, et al. **Sequential Convex Programming Methods for Real-Time Optimal Trajectory Planning in Autonomous Vehicle Racing**. IEEE Transactions on Intelligent Vehicles, 2022, pp. 1–1, <https://doi.org/10.1109/tiv.2022.3168130>. Accessed 16 May 2022.