

CINEMÁTICA INVERSA DE UM MANIPULADO ROBÓTICO DE QUATRO GRAUS DE LIBERDADE UTILIZANDO MÉTODO NUMÉRICO ITERATIVO DA JACOBIANA PSEUDO-INVÉRSIA.

TARCISIO CARLOS F. PINHEIRO¹, MAX RICARDO P. DA TRINDADE¹, BRENO R. PANTOJA².

INSTITUTO DE ESTUDOS SUPERIORES DA AMAZÔNIA – IESAM¹ AVENIDA GOVERNADOR JOSÉ MALCHER,
Nº 1148, BAIRRO: NAZARÉ, BELÉM-PA.

UNIVERSIDADE FEDERAL DO PARÁ – UFPA² RUA AUGUSTO CORRÊA, 01, BAIRRO: GUAMÁ, BELÉM-PA.

E-MAILS: TARCISIO.PINHEIRO@OUTLOOK.COM, MAX@PROF.IESAM-PA.EDU.BR,
BRENOPANTOJA@UFPA.BR

Abstract— The kinematic of a manipulator is the study of the movement without taking consideration of physics forces involved. Frequently, difficult while positioning and constructing trajectories of robotics handlers are found, mainly in case where transformations of the desired points at spatial plan, for the angles corresponding of each joint, are needed. The problem is because differential equations may present more of one or even no solution to the stipulated position. To solve this trouble, the implementations of numeric, or analytical, solutions are necessary, each one with its advantage or disadvantage. The work's objective is to demonstrate the utilization of numeric method by using pseudo-inverse Jacobin to solve the inverse cinematic utilizing the Matrix Laboratory (MATLAB). In this article, computational simulations made using the robotic toolbox together with tests into prototype are noticed.

Keywords— Inverse kinematics, MATLAB robotic toolbox, four degrees of freedom manipulator, pseudo-inverse Jacobin.

Resumo— A cinemática de um manipulador é nada mais que estudo do seu movimento sem levar em consideração as forças físicas envolvidas. Dificuldades com o posicionamento e trajetórias de manipuladores robóticos são frequentes, principalmente nos casos onde é necessário fazer a transformação dos pontos desejados no plano espacial para os ângulos correspondentes de cada junta. A problemática ocorre por causa das equações diferenciais terem mais de uma ou até mesmo nenhuma solução para a posição estipulada. Para resolver este problema, as implementações de soluções numéricas ou analíticas são indispensáveis, cada um com suas vantagens ou desvantagens em relação ao outro. Esse trabalho tem como objetivo demonstrar a utilização de um método numérico através da jacobiana pseudo-inversa para solução da cinemática inversa por meio do *Matrix Laboratory (MATLAB)*. As simulações computacionais realizadas no *toolbox robotic* juntamente com testes no protótipo do manipulador são apresentadas no final desse trabalho.

Palavras-chave— Cinemática inversa, MATLAB *toolbox robotic*, manipulador de quatro graus de liberdade, Jacobiano pseudo-inverso.

1 Introdução

A utilização da robótica vem crescendo em diversas áreas. Na indústria isso ocorre devido a uma série exigências do mercado onde se pode destacar o aumento da produção, da qualidade e do custo da mão de obra. Tendo isso vinculado a uma concorrência altamente competitiva a redor do mundo.

Os robôs mais utilizados na indústria, são os manipuladores robóticos capazem de ser definidos como um conjunto de corpo rígido interligados em série por juntas rotativas ou prismáticas. Essas interligações formam uma cadeia cinemática aberta, na qual uma das extremidades do braço robótico, se encontra fixado a uma base cérceo, enquanto a extremidade oposta possui o órgão terminal que pode mover-se livremente no espaço (LOPES, 2001). Essas máquinas são utilizadas em várias atividades tais como: Paletização de materiais, deslocamento de cargas, pintura, soldagem, etc.

Para acompanhar as grandes exigências, principalmente relacionadas à precisão e a velocidade na execução de tarefas, é necessário o desenvolvimento de técnicas de controle capazem de encontrar as trajetórias, com o posicionamento do manipulador. A técnica mais utilizada é a cinemática inversa cuja é

um processo pelo qual os ângulos das juntas de um atuador são calculados a partir da posição em que se deseja colocar o órgão terminal (AZEVEDO & CONCI, 2003). O cálculo desses ângulos possibilita a construção de sistemas de controle tendo no seu desenvolvimento as matrizes jacobianas, em conjunto com as pseudo-inversas (WUNDERLICH, 2004).

O trabalho proposto demonstra a aplicação do método numérico utilizando a Jacobiana pseudo-inversa para as soluções envolvendo a cinemática inversa em um protótipo de manipulador e simulações computacionais por meio do *Matrix Laboratory (MATLAB)*, Software muito utilizado na engenharia.

Na seção 2 são apresentadas as características principais do manipulador utilizado. Na seção 3 é descrito o método utilizado para a solução da cinemática inversa. A simulação computacional é mostrada na seção 4. Análise dos resultados e sugestões são discutidas na seção 5.

2 Características do protótipo Utilizado.

Para se realizar experiências com a cinemática inversa gerada pelo o MATLAB, foi construído um protótipo de manipulador robótico apresentado na Figura 1. Esse braço robótico serial tem as seguintes características: Quatro graus de liberdade, juntas verticais

articuladas, atuadores elétrico do tipo servo motor modelo Tower Pro MG955.

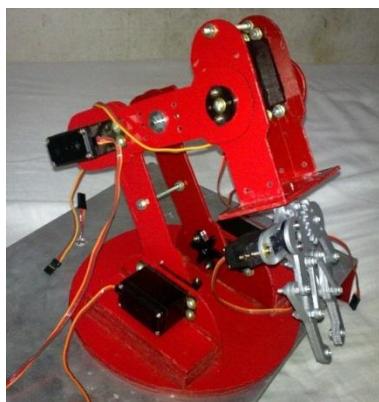


Figura 1. Protótipo utilizado.

A notação de Denavit-Hartenberg (D-H) é uma ferramenta muito utilizada para sistematizar a descrição cinemática de sistemas mecânicos articulados com n graus de liberdade (DENAVIT, 1955). Essa notação é adotada tanto no protótipo como na simulação computacional desse trabalho. Na Figura 2 está o desenho esquemático braço desenvolvido no plano (X, Y, Z) e a Tabela 1 contém os valores dos parâmetros DH, onde os *Links* representam os elos entre juntas, o ângulo de cada articulação. l_i é distância ao longo do eixo X_i , desde a intersecção entre os eixos Z_{i-1} , X_i até a origem do sistema i . d_i é a distância ao longo do eixo Z_{i-1} , desde a origem do sistema de coordenadas $i-1$ até a intersecção entre os eixos Z_{i-1} / X_i . θ_i é o ângulo entre os eixos Z_{i-1} . Por ultimo, o θ_{max} é o ângulo entre um plano perpendicular ao eixo X_i , usando a regra da mão direita. θ_{min} e θ_{max} são os valores angulares máximos, mínimos de cada junta.

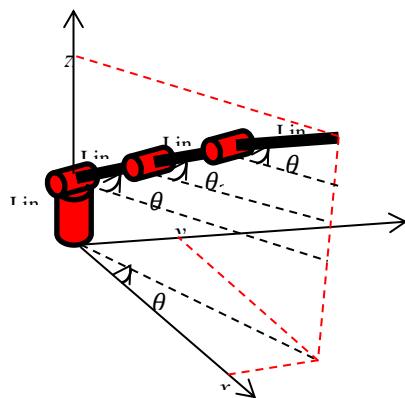


Figura 2. Desenho esquemático do manipulador utilizado

Tabela 1. Parâmetros D-H do manipulador.

Link	l	d	θ	v	θ_{min}	θ_{max}
1	0	0.03	+90°	0°	+180°	
2	0.1	0	0°	0°	+180°	
3	0.1	0	0°	-180°	0°	
4	0.18	0	0°	-90°	+90°	

3 Método para solução da cinemática inversa.

A cinemática de um robô manipulador é definida como o estudo da posição e da velocidade do seu efetuador (órgão terminal), incluindo dos seus *links*. Na cinemática inversa são fornecidas a posição, em conjunto com a velocidade do órgão terminal para obter as posições, incluindo velocidades correspondentes das articulações (Cabral, 2009). Já a cinemática direta são fornecidos os ângulos de cada junta para obter a posição do órgão terminal.

Existem diversas formas de se calcular a cinemática inversa de um manipulador. As soluções mais utilizadas são os métodos analíticos e numéricos iterativos. A escolha de um método depende de sua aplicação do tipo de junta; Da estrutura utilizada em relação da quantidade de graus de liberdade. Na maioria dos casos o método numérico é preferível quando se tem acesso ao um processamento computacional adequado. Existem diversos métodos numéricos iterativos, por exemplo: O Newton-Raphson, Algoritmo de Levenberg–Marquardt, o Jacobiano Pseudo-Inverso, entre outros.

Neste trabalho é usado o método numérico iterativo da Jacobiana pseudo-inversa, o mesmo foi escolhido pela facilidade que a matriz jacobiana pseudo-inversa proporciona na solução de sistemas não lineares para manipuladores redundantes e por sua simples execução em comparação aos outros métodos numéricos (Ferreira, 2008). A Figura 3 apresenta o esquema do problema de transformação inversa de coordenadas para um robô com N graus de liberdade. Nela, os ângulos das juntas são obtidos a partir das informações dos parâmetros do braço/da posição desejada para o órgão terminar.

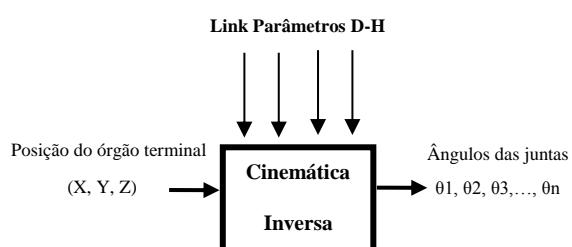


Figura 3. Problema da transformação inversa.

3.1 Método numérico iterativo da Jacobiana Pseudo-Inversa.

Este método numérico consiste em calcular a matriz jacobiana pseudo-inverso (J^\dagger). Essa matriz possibilita relacionar a velocidade do órgão terminal com a velocidade angular das juntas para cada iteração. A quantidade de iterações (k) está relacionada diretamente com o passo (Δt) utilizado. A escolha do Δt , devido a não linearidade, deve ser infinitesimal (ou muito pequeno) para obter bons resultados (Lopes, 2002). Este método foi utilizado para calcular apenas os valores de θ_2 , θ_3 e θ_4 , já que θ_1 pode ser calcula-

do diretamente. Um fluxograma do método numérico iterativo é mostrado na Figura 4, onde os passos principais são descritos nos sub tópicos:

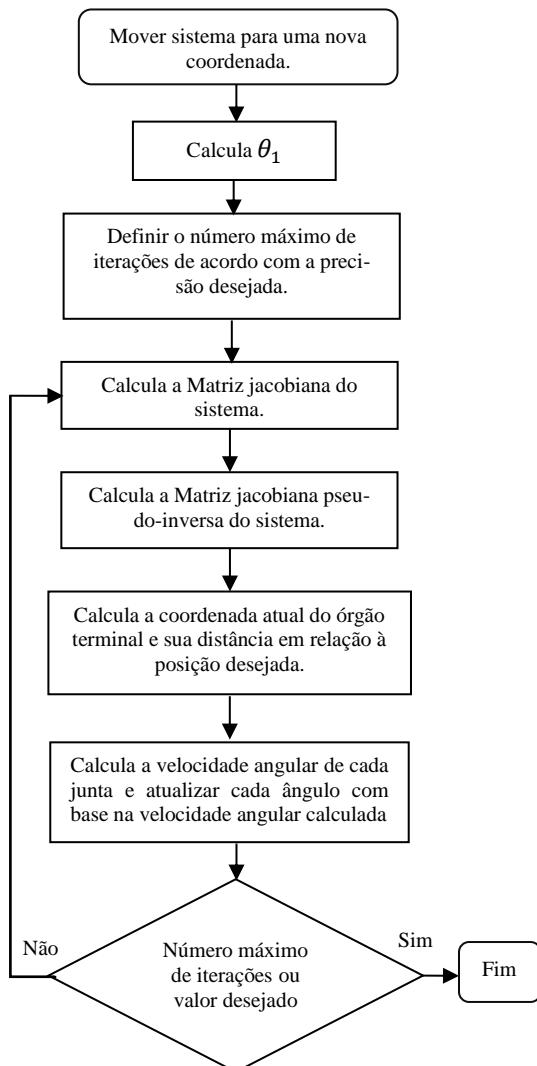


Figura 4. Fluxograma da heurística utilizada.

3.1.1 Determinação da matriz Jacobiana

A matriz jacobiana é a derivada parcial de um vetor pelo outro que pode ser definido através da Equação 1. É construída a partir das relações das equações geométricas (f_n) nas quais descrevem a arquitetura do manipulador, note a seguir.

$$J(f, \theta) = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} & \dots & \frac{\partial f_1}{\partial \theta_n} \\ \frac{\partial f_2}{\partial \theta_1} & \frac{\partial f_2}{\partial \theta_2} & \dots & \frac{\partial f_2}{\partial \theta_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial \theta_1} & \dots & \dots & \frac{\partial f_m}{\partial \theta_n} \end{bmatrix} \quad (1)$$

Segundo (SANTOS, 2004), a obtenção analítica por diferenciação da cinemática direta é o processo mais natural para obter a matriz jacobiana. Para determinar as equações geométrica leva-se em consideração que espaço operacional do manipulador se tornou 2D, desse modo gera-se duas análises. A primeira em relação ao plano (X' , Z) visto na Figura 6, responsável em gerar a translação, onde Z é a coordenada do espaço (altura) e X' é a projeção do manipulador sobre o plano (X , Y). A segunda análise é no plano (X , Y) visto na Figura 5 determina a rotação em torno do eixo Z . O valor de θ_1 basta utilizar a Equação 2 obtida a partir da mesma ilustração.

$$\theta_1 = \arctan\left(\frac{y}{x}\right) \quad (2)$$

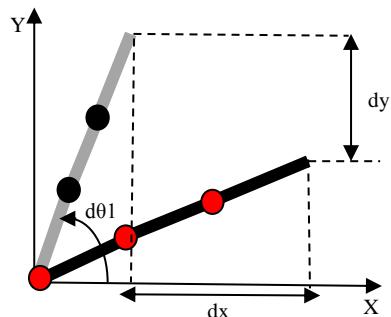


Figura 5. Manipulador 2D em relação (X , Y).

Para o manipulador em estudo, a matriz jacobiana na Equação 4 é calculada pela derivação parcial da Equação 3 que é obtida por meio da análise geométrica da cinemática direta da Figura 6. Para simplificar a escrita das Equações 3, 4 foi adotado a seguinte nomenclatura: $\sin(\theta_1 + \theta_2 + \dots + \theta_n) = S1 \dots n$ e $\cos(\theta_1 + \theta_2 + \dots + \theta_n) = C1 \dots n$.

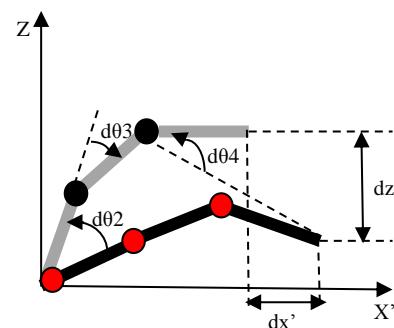


Figura 6. Manipulador 2D em relação (X' , Z).

$$\begin{cases} x' = l_2C_2 + l_3C_{23} + l_4C_{234} \\ z = l_2S_2 + l_3S_{23} + l_4S_{234} \end{cases} \quad (3)$$

$$\left| \frac{dx'}{dz} \right| = \begin{bmatrix} -l_2S_2 - l_3S_{23} - l_4S_{234} & -l_3S_{23} - l_4S_{234} & -l_4S_{234} \\ l_2C_2 + l_3C_{23} + l_4C_{234} & l_3C_{23} + l_4C_{234} & l_4C_{234} \end{bmatrix} \begin{bmatrix} d\theta_2 \\ d\theta_3 \\ d\theta_4 \end{bmatrix} \quad (4)$$

3.1.2 Recurso da Jacobiana pseudo-inversa.

Para resolver a cinemática diferencial de manipuladores redundantes é utilizada a pseudo-inversa da matriz jacobiana (NENCHEV, 1989). Através dessa matriz é possível encontrar os valores para de θ_2 , θ_3 e θ_4 . Este recurso é necessário quando a matriz jacobiana não é quadrada que é o caso da matriz obtida na Equação 4. Por meio da Equação 5, é calculada a matriz pseudo-jacobiana na qual tem origem no método dos multiplicadores de Lagrange (SCIAVICCO; SICILIANO, 1996), onde W é a matriz identidade de ordem $In \in R^{n \times n}$ (Corke, 2011).

$$J^\dagger = W^{-1} (J^T (JW^{-1}J^T)^{-1}) \quad (5)$$

3.1.3 Determinação da velocidade Angular.

A matriz jacobiana de um manipulador mostra-se como um elemento fundamental para relacionar as velocidades no espaço operacional em função da velocidade das juntas (SANTOS, 2004). Essa velocidade (\dot{q}) está relacionada com a velocidade do órgão terminal (V) através do jacobiano, de acordo com a Equação 6. Incluindo com o ângulo atual da junta ($q(k+1)$) é obtido por meio da integração de Euler na Equação 7, são notadas abaixo:

$$\dot{q} = J^\dagger V \quad (6)$$

$$q(k+1) = q(k) + \dot{q}(k)\Delta t \quad (7)$$

3.1.4 Critério de Parada.

O critério de parada de um método numérico iterativo, é um fator muito importante a se observar. Existem várias formas de saber, tempo certo para o fim do *loop* do sistema. A parada pode ser definida, quando é encontrada uma solução desejada (dentro de certa tolerância de erro) ou quando chega ao número máximo de iterações. Esses critérios foram utilizados neste trabalho.

4 Simulação computacional e experiências com o protótipo

Para testes computacionais com o método numérico foi utilizado o MATLAB com suas ferramentas, onde se pode destacar a *Graphical User Interface Development Environment* (GUIDE) e o *toolbox robotic*. O GUIDE é uma área de trabalhos para o desenvolvimento de interfaces gráficas orientada a objeto. O *toolbox robotic* desenvolvido por Peter Corke é uma

ferramenta capaz de fazer simulações virtuais com manipuladores de diferentes modelos, além de auxiliar nos cálculos voltados para cinemática/dinâmica de robôs (Corke, 2011).

4.1 Resultados da simulação computacional.

Sabendo que, os erros na posição do órgão terminal do manipulador estão diretamente ligados ao passo utilizado. Os testes a cerca de três valores de passo ($\Delta t=0.01$, $\Delta t=0.001$ e $\Delta t=0.0001$) em um intervalo de tempo 0 a 2. Na Tabela 2 é possível verificar o passo $\Delta t=0.0001$, obteve menor erro relativo. Portanto o Δt , é diretamente proporcional ao erro.

Tabela 2. Valores obtidos no teste.

Posição inicial	Posição Desejada (metros)	Resultado obtido da posição final do órgão terminal (metros) e posição final das Juntas (graus)					
		$\Delta t = 0.01$		$\Delta t = 0.001$		$\Delta t = 0.0001$	
		Valores	Valores	Valores	Erro relativo	Valores	Erro relativo
θ_1	90	-	26.5650	-	26.5650	-	26.5650
θ_2	90	-	113.201	-	111.968	-	111.453
θ_3	-90	-	-95.285	-	-88.773	-	-87.842
θ_4	-36	-	-33.141	-	-43.447	-	-44.107
x	0	0.2	0.20521	2.605%	0.19979	0.105%	0.20006
y	0.2	0.1	0.10260	2.6%	0.09989	0.11%	0.10003
z	0.02	0.1	0.10540	5.4%	0.09981	0.19%	0.10001

Na Figura 7 observa-se uma grande variação nos valores dos ângulos das juntas, ou seja, dificuldade em convergência do Algoritmo, para valores de passo Δt elevados. Desta forma, quando o passo é menor, a convergência é mais rápida e menor é o erro, ilustrados nas Figuras 8 e 9:

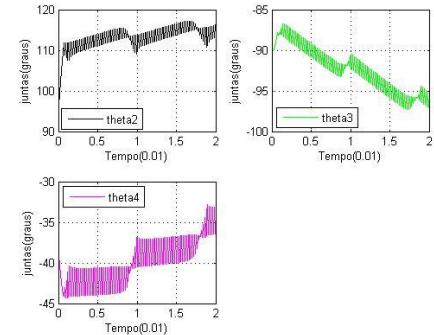


Figura 7. Convergência dos ângulos para o valor desejado para $\Delta t=0.01$.

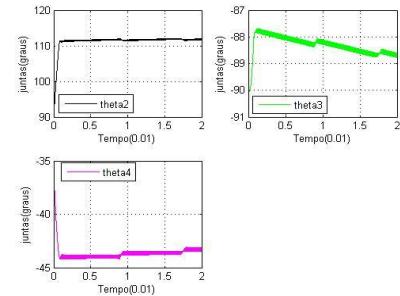


Figura 8. Convergência dos ângulos para o valor desejado para $\Delta t=0.001$.

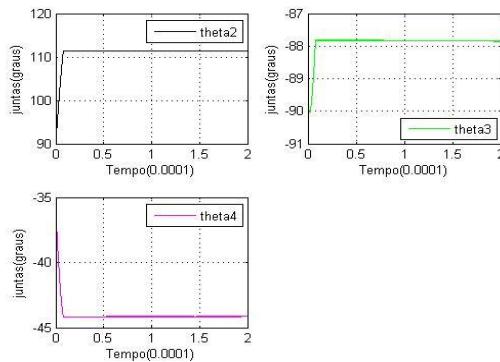


Figura 9. Convergência dos ângulos para o valor desejado para $\Delta t=0.0001$.

Para gerar a trajetória de modo suave entre a posição angular inicial e a posição final calculada foi utilizada a função *jtraj* do *toolbox robotic*. Esta função obtém os valores angulares intermediários da trajetória por interpolação para um intervalo de tempo definido (Cork, 2011).

Na Figura 10 abaixo, é demonstrado 4 gráficos obtidos através função *jtraj* para um intervalo de tempo de 0 a 4 segundos. Nesses gráficos é possível analisar os valores das juntas e do órgão terminal para cada instante de tempo. No gráfico 1, 2, 3 é possível analisar os valores das juntas em relação à posição, velocidade, juntamente com aceleração, respectivamente. No gráfico 4 é verificado a trajetória do órgão terminal (Cork, 2011).

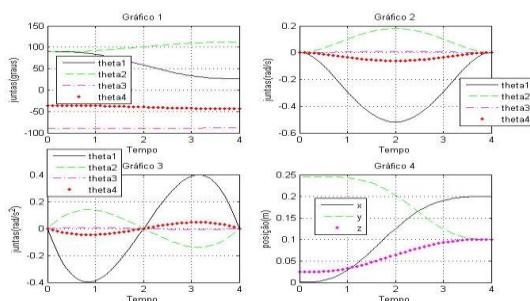


Figura 10. Trajetória em relação ao tempo.

O *toolbox robotic* faz as simulações virtuais do manipulador em conjunto com o robô. Essas simulações são de extrema vantagem para analisar seus possíveis movimentos além de possibilitar a verificação da posição exata do órgão terminal no espaço de trabalho. Na Figura 11, é exibido o resultado de uma simulação utilizando $\Delta t=0.0001$, onde coordenada desejada do órgão terminal é $(0.2, 0.1, 0.1)$.

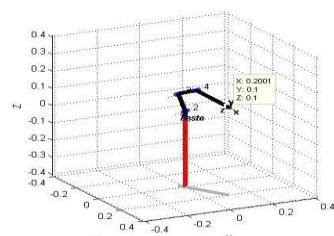


Figura 11. Resultados obtidos na simulação virtual no *toolbox robotic*.

4.2 Resultados com o protótipo.

Os testes realizados com o protótipo a fim de verificar a eficácia da cinemática inversa desenvolvida no MATLAB. A criação de uma interface gráfica para facilitar o controle do protótipo e a visualização dos resultados mostrado na Figura 12. No mesmo é possível configurar a comunicação *serial Universal Asynchronous Receiver/Transmitter* (UART), fazendo o controle dos ângulos das juntas, entrando com os valores da coordenada desejada do órgão terminal. Nessa interface foi inserido o gráfico com a simulação virtual do manipulador, a imagem do protótipo pode ser capturada através de uma câmera.

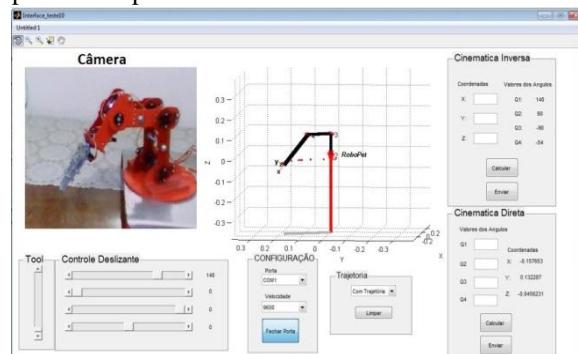


Figura 12. Interface gráfica no GUIDE para controle do protótipo.

Na Figura 13, é visto o esquema físico da comunicação entre o computador e o manipulador. O controle do manipulador é realizado diretamente por um Microcontrolador 16F877A. Sendo responsável por receber os valores dos ângulos calculados pelo MATLAB, logo processa todas as informações.

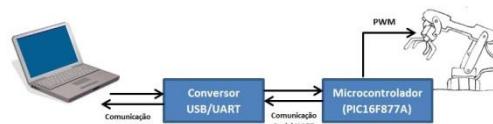


Figura 13. Comunicação Física entre o computador e o manipulador.

No teste com o protótipo, é definido uma trajetória composta de vinte pontos no espaço. Para facilitar a medição leva-se em consideração apenas o plano (X, Y), ou seja, os valores do eixo Z são iguais à zero. A interface gráfica são inseridos os valores desses pontos e depois é feito, visualmente, a medição da posição do órgão terminal com o auxílio do papel milimetrado, desta maneira, uma caneta é fixada na extremidade do manipulador. Na Figura 14 mostra a trajetória executada pelo manipulador, em seguida na Figura 15 ilustra os resultados obtidos:



Figura 14. Experiência com o protótipo.

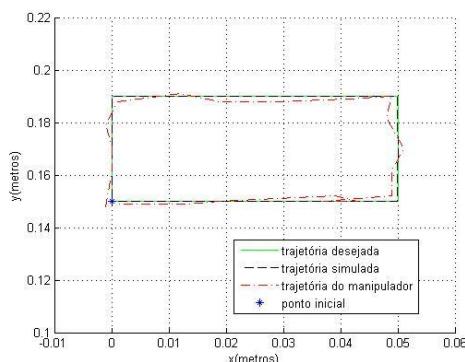


Figura 15. Medição do dos pontos desejados.

Para analisar os resultados na Figura 15 em relação à trajetória do manipulador, devem-se observar os erros ocasionados pelo posicionamento do motor e de eventuais problemas com a estrutura, como folgas, desalinhamento, entre outras.

5 Conclusão

Após a análise dos resultados pode ser verificado a eficácia do método dentro de suas limitações. Para o controle de manipuladores, muitas vezes, é necessário calcular em tempo real a solução do problema da cinemática inversa. Assim, o peso computacional e o tempo de resposta devem ser levados em consideração principalmente quando é exigido fazer uma trajetória ponto a ponto. Para a otimização do erro, abrangendo o tempo de resposta das soluções utilizando Inteligência Artificial (AI) tais como: Algoritmo Genético, Rede Neural, Cadeias de Markovs são indicados para trabalhos futuros.

Para melhorias do erro no protótipo pode ser citada a aplicação de um controle *Proportional-Integral-Derivative* (PID) para motor, atualmente o mesmo é controlado por um simples controlador proporcional. Utilização de encode óptico e de potenciômetros para fazer a realimentação do sistema, com melhorias na estrutura do manipulador em relação às folgas nos acoplamentos dos eixos do motor são outras soluções que podem ser abordadas.

Referências Bibliográficas

- Azevedo, E. and Conci, A (2003). Computação Gráfica - Teoria e Prática, Vol.1, Elsevier, Rio de Janeiro.
- Cabral, E.L.L. "Análise de Robôs", ch. 5 ,www.poli.usp.br/p/eduardo.cabral/Cinemática Direta.pdf acessado em 09/03/2013.
- Coiffet, P.(1982), "Les Robôs, Tome 1: Modélisation et commande", Vol.2, Hermes, Paris.
- Corke, P. (2011). Robotics, Vision and Control, Vol.1, Springer-Verlag Berlin Heidelberg, Australia pp. 141- 193. pp. 141-200. DOI: [10.1007/978-3-642-20144-8](https://doi.org/10.1007/978-3-642-20144-8)
- Craig, J. J. (2005). Introduction to Robotics: Mechanics and Control, Vol.3, Pearson Prentice Hall, New York. pp. 62-164.
- Denavit, J.(1955)Description and displacement analysis of mechanics based on the 2x2 dual matrices, 1 edn, University of Northwestern, Evanston.
- Lewis, F. L; Dawson D. M. and Abdallah, C.T. (2004). Robot Manipulator Control Theory and Practice, Vol.2, Marcel Dekker, New York. pp. 107-126.
- Lopes, A. M.(2002) Robótica Industrial, Dissertação (Mestrado em Automação, Instrumentação e Controle), Universidade do Porto, Porto.
- Nenchev,D.N.(1989).Redundancy resolution through local optimization: A review. J. Robot. Syst., Vol.6, pp. 769–798. DOI: [10.1002/rob.4620060607](https://doi.org/10.1002/rob.4620060607)
- Santos, V. M. F. (2004). Robótica Industrial, Vol.1, Universidade de Aveiro, Aveiro. pp. 68-90.
- Sciavicco, L.;Siciliano, B. (1996). Modeling and control of robot manipulator. McGraw-Hill Companies.
- Simas, H. (2008). Planejamento de Trajetórias e Evitamento de colisão em Tarefas de Manipuladores Redundantes Operando em Ambientes Confinados. Tese de Doutorado, Universidade Federal de Santa Catarina, Florianópolis-SC.
- Wunderlich, J. T. (2004). Simulating a robotic arm in a box: redundant kinematics, path planning, and rapid prototyping for enclosed spaces, SIMULATION, Vol. 80, Issue 6, Elizabethtown.