

Solução de cinemática híbrida para robô pantógrafo de três graus de liberdade^{*}

Kassio M. Kienitz^{*} Eduardo José Lima II^{**}

^{*} Programa de pós-graduação em Engenharia mecânica, Escola de Engenharia, UFMG - Universidade Federal de Minas Gerais, Minas Gerais Brasil (e-mail: kassio.kienitz@gmail.com).

^{**} Programa de pós-graduação em Engenharia mecânica, Escola de Engenharia, UFMG - Universidade Federal de Minas Gerais, Minas Gerais Brasil (e-mail: ejlima2@gmail.com).

Abstract: This work aims to present a hybrid solution for kinematic models. For this purpose, the closed-form solution and the parametric Denavit-Hartenberg solution are combined. The robot geometry chosen is a four bars pantograph with three DOF. The mathematical model was developed and simulated using the software GNU Octave and tested on an embedded system. The motion command used is based on a real-time online kinematic translator that allows the use of a general Cartesian CNC to command a non-Cartesian robot. The kinematic model was validated by comparing a desired commanded trajectory to the translated, executed trajectory. This proved that the mathematical solution is simple enough to be implemented in a low-cost embedded system.

Resumo: Este trabalho tem como objetivo apresentar uma solução híbrida para modelos cinemáticos baseada na combinação dos métodos de solução analítica geométrica e solução paramétrica de Denavit-Hartenberg. A geometria de robô escolhida é um pantógrafo paralelo planar com três graus de liberdade utilizando a configuração de mecanismo de quatro barras. O modelo matemático foi desenvolvido e simulado na plataforma de programação matemática GNU Octave e posteriormente implementada em um sistema embarcado real. O sistema de comando de movimento utilizado é baseado em um tradutor de cinemática on-line e em tempo real, tornando possível utilizar um comando de CNC cartesiano para comandar um robô não cartesiano. O modelo cinemático foi validado comparando-se uma trajetória comandada no CNC e a trajetória executada pelo modelo do robô após passar pelo tradutor, assim comprovando que a solução matemática é simples o bastante para ser implementada em um sistema embarcado de baixo custo.

Keywords: Inverse Kinematics, Parallel Robotic, Industrial Robotic, Motion Command, Hybrid Robot

Palavras-chaves: Cinemática Inversa, Robótica Paralela, Robótica Industrial, Comando de Movimento, Robôs Híbridos.

1. INTRODUÇÃO

A maioria dos robôs instalados na indústria são de cadeias cinemáticas seriais, como pesquisado por FortuneBusinessInsight (2021), mesmo com as limitações inerentes desta configuração como descrito por Merlet (2006) e Tsai (1999). Robôs paralelos são definidos por Merlet (2006) como manipuladores que tem seu efetuador de trabalho ligado à base do robô por cadeias cinemáticas fechadas e independentes, tendo assim uma maior capacidade de carga, velocidade e aceleração quando comparados a robôs seriais. A multiplicidade de pontos de atuação entre a base e o efetuador induz estes manipuladores a serem mais leves e com menos folgas mecânicas. Porém, as principais desvantagens dos robôs paralelos estão numa menor área útil de trabalho e limitações em seus movimentos para evitar obstáculos. Isso implica em um estudo mais cuidadoso de

sua utilização para um tipo de trabalho alvo para se evitar colisões do robô.

Essa precarização da utilização dos robôs paralelos na indústria em parte se dá por sua cinemática mais complexa, justificando o estudo, simulação e desenvolvimento de soluções que desmistifiquem essa complexidade. Como indicado por de Lima et al. (2019), a utilização de ambientes simulados é uma ferramenta importante para a facilitação e redução de custo do estudo de geometrias robóticas. No ensino da robótica no Brasil, Dâmaso (2019) coloca que existe uma distância entre a teoria nos livros canônicos da área e a praxe do ensino prático, onde os livros abordam o projeto e desenvolvimento de geometrias e sistemas enquanto a prática do ensino foca na operação de robôs consolidados no mercado ou em plataformas didáticas. A modelagem e simulação de soluções cinemáticas de robôs muitas vezes depende de softwares especializados pagos. No presente trabalho foi escolhido um software

^{*} A Far-X Tecnologia deu suporte material a este trabalho.

de modelagem matemática aberto para que a difusão do conhecimento seja facilitada, não limitando por custos a replicação deste estudo. A plataforma de simulação utilizada é baseada em hardware de baixo custo e softwares abertos como: *GRBL CNC*, *Arduino IDE*, *Espressif SOC*, *ATmel 328P*, *Sigrock Pulse View*.

Mesmo geometrias paralelas bem estudadas como a plataforma de Stewart, que tem a vantagem de sua cinemática indireta ser trivial, pode ter sua cinemática direta sem uma solução analítica, dependendo da configuração escolhida e da solução de um sistema de equações não lineares, como explicado por Silva and Giesbrecht (2021), que demonstrou como realizar essa solução utilizando métodos numéricos através de aprendizado de máquina, o que dificulta a utilização de sistemas embarcados de baixo custo devido ao alto custo computacional. Em estudos anteriores, Alvares et al. (2017) aplicaram a solução dos modelos cinemáticos de forma off-line, traduzindo a trajetória no espaço cartesiano para uma trajetória no espaço de juntas, através de um pós processador no planejador de trajetória. Em seguida, Alvares et al. (2018) realizaram a tradução de forma de forma on-line em um software de comando de trajetória modificado especialmente para uma geometria alvo.

No presente trabalho é utilizado um tradutor de cinemáticas on-line e em tempo real, que recebe o comando de movimento cartesiano vindo de um CNC e traduz esse comando para um movimento coordenado das juntas, de acordo com a solução da cinemática inversa e direta implementada. Assim é possível que qualquer controlador de movimento CNC com saídas pulso e direção possa ser utilizado para comandar o movimento do robô, mantendo os recursos nativos destes, como aprendizado ponto a ponto, correção de paralelismo de área de trabalho e peça, correções de movimento on-line entre outros.

O robô escolhido para este estudo é um robô híbrido, composto de cadeias fechadas em paralelo e cadeias abertas em série em sua geometria. Isso para que haja uma extensão da área de trabalho da configuração paralela e uma simplificação da cinemática da orientação de sua ferramenta. Robôs híbridos possivelmente podem melhorar o desempenho de robôs industriais padrão ampliando a gama de trabalhos que possam ser executados por estes, como já indicado por Filho et al. (2010) e Lima II et al. (2020). Como exemplos dessa configuração pode-se citar um robô antropomórfico serial de 6 eixos com uma plataforma de Stewart ligada ao efetuador possibilitando que a ferramenta possa realizar movimentos repetitivos de alta frequência e baixa amplitude com exatidão e sem comprometer a vida útil do conjunto mecânico do robô serial, ou mesmo um robô paralelo com um eixo extra para realizar somente a rotação de orientação da ferramenta, de forma independente da cinemática global. Assim como indicado por Ferrao (2018), utilizando a classificação de Yoshida et al. (1996) para robôs, pode-se colocar estes exemplos como configurações de sistemas robóticos macro-micro, definidos por Sharon et al. (1993) como sistemas robóticos com duas geometrias em série: uma de longo alcance e a segunda acoplada a esta e capaz de realizar tarefas de maior precisão.

Na segunda seção deste trabalho é descrito o robô em estudo, bem como o modelo geométrico e matemático de

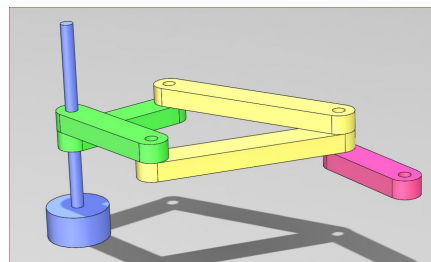


Figura 1. Modelo 3D do robô

suas cinemáticas direta e inversa, assim como as singularidades e similaridades resultantes das soluções destas. Na terceira seção é descrito como foram implementados estes modelos nos algoritmos das simulações e nos experimentos executados. Na quarta seção são apresentados os contornos dos experimentos executados e os resultados obtidos. Na seção final são discutidas as implicações destes resultados e as conclusões implicantes destes.

2. DESCRIÇÃO DO ROBÔ

O robô alvo deste trabalho foi concebido para manipular uma ferramenta não holonômica, como uma faca ou uma tocha de soldagem, com as seguintes características de operação: Posicionar a ferramenta no plano cartesiano XY; Orientar a ferramenta na rotação do eixo Z; compensar a orientação da ferramenta durante os movimentos de translação e rotação. O robô em questão é um pantógrafo de 4 barras com atuação concêntrica na origem, formando uma cadeia cinemática fechada que realiza a translação XY. Ao final desta cadeia é acoplado um terceiro eixo rotativo em série, formando assim um manipulador híbrido, como pode se visto na Figura 1. Essa configuração é um caso especial do robô planar de 5 barras descrito por Briot et al. (2015), porém ao utilizar a atuação concêntrica substitui-se a quinta barra, simplificando a solução da cinemática e possibilitando a utilização do método D-H para solucioná-la. Logo tem-se um robô que pode ter sua cinemática resolvida por: Solução geométrica analítica - Merlet (2006), pelo método Denavit-Hartenberg (D-H) - Hartenberg and Denavit (1955); pelo método Khalil-Kleinfinger (K-K) - Briot et al. (2015) e a possibilidade de usar o método de desacoplamento cinemático - Spong et al. (2006), proporcionando assim que comparações e verificações entre os métodos possam ser realizadas, auxiliando o estudo desses métodos.

2.1 Modelo de cinemática inversa

O modelo de cinemática inversa é responsável por determinar a configuração do robô para que a ferramenta esteja numa posição e orientação alvo. Para isso a posição dos pontos $P1, P2, P3$ e o ângulo θ_4 , mostrados na Figura 2, devem ser calculados de acordo com a solução matemática da cinemática. A solução da cinemática inversa parte da posição alvo da ferramenta, $P4$, considerando um braço (elo) equivalente virtual entre os pontos $P0$ e $P3$, como mostrado na Figura 2. Assim a posição de $P3$ depende da posição de $P4$ e do ângulo de orientação θ_4 , e o robô torna-se um robô planar de dois links. A solução D-H deste robô pode ser observada em Craig (2009), e sua descrição geométrica se traduz em: Assumindo que $P4$ é o centro

do círculo $C3$ e que o $link3$ é o raio deste círculo, temos que a posição de $P3$ pode ser calculada pela projeção do $link3$ nos eixos cartesianos XY , i.e. aplicando diretamente o cosseno de θ_4 como pode ser visualizado na Figura 3. A Tabela 1 é a tabela de parâmetros D-H baseada no robô com link equivalente virtual.

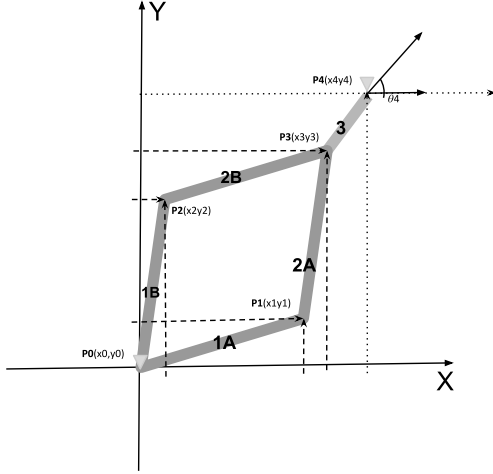


Figura 2. Descritivo das juntas e braços do robô

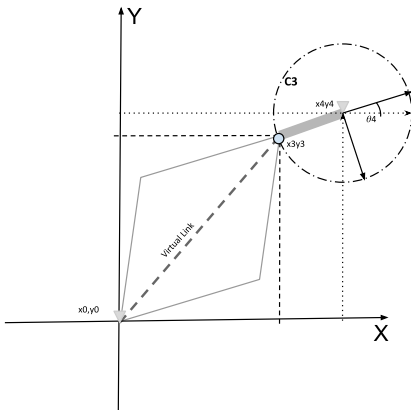


Figura 3. Cinemática inversa do $link3$ utilizando o link virtual equivalente

Tabela 1. Tabela de parâmetros D-H para um robô RR de dois links

Link	a_i	α_i	d_i	θ_i
3	a_1	0	0	θ_1^*
Virtual	a_2	0	0	θ_2^*

E a matriz transformação homogênea é dada por:

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & a_1 \cdot c_1 \\ s_1 & c_1 & 0 & a_1 \cdot s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 \cdot c_2 \\ s_2 & c_2 & 0 & a_2 \cdot s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_1^0 = A_1$$

(1)

(2)

(3)

$$T_2^1 = A_2 \quad (4)$$

$$T_2^0 = A_1 \times A_2 \quad (5)$$

$$T_2^0 = \begin{bmatrix} c_{12} & -s_{12} & 0 & a_1 \cdot c_1 + a_2 \cdot c_{12} \\ s_{12} & c_{12} & 0 & a_1 \cdot s_1 + a_2 \cdot s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Sendo $s_1 = \sin(\theta_1)$, para c_1, s_2, c_2 e $s_{12} = \sin(\theta_1 + \theta_2)$, e o mesmo para c_{12} .

Resultando em:

$$x_3 = a_1 \cdot c_1 + a_2 \cdot c_{12} \quad (7)$$

$$y_3 = a_1 \cdot s_1 + a_2 \cdot s_{12} \quad (8)$$

Já os pontos $P1$ e $P2$ são calculados pelo método K-K, também chamado de D-H Modificado, como demonstrado por Briot et al. (2015). O resultado pode ser representado geometricamente como os pontos de interseção entre dois círculos, $C1$ e $C2$ da Figura 4, onde $C1$ tem seu centro na origem e seu raio os braços $link1_A$ e $link1_B$, e $C2$ com centro em $P3$ e raio nos braços $link2_A$ e $link2_B$. A solução analítica é encontrada solucionando-se o sistema de equações (9) e (10) de segundo grau com duas soluções possíveis, sendo uma $P1$ e a outra $P2$.

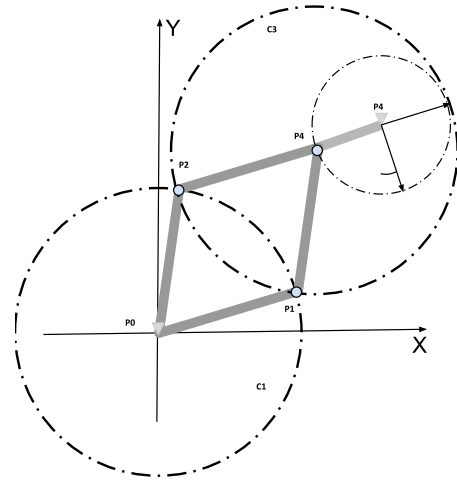


Figura 4. Descrição Geométrica da cinemática inversa

$$X^2 - 2x_3X + x_3^2 + y^2 - 2y_3Y + y_3^2 = \sqrt{a_1} \quad (9)$$

$$X^2 - 2x_0X + x_0^2 + y^2 - 2y_0Y + y_0^2 = \sqrt{a_2} \quad (10)$$

Para resolver o sistema de equações (9) e (10), foi utilizado um método de separação em etapas e parâmetros possibilitando sua implementação de forma facilitada em plataformas computacionais embarcadas, evitando que a solução se torne uma função grande indivisível com um elevado custo computacional. Assim os parâmetros representam os índices das equações do sistema:

$$\begin{aligned} p_{11} &= -2x_3 \\ p_{12} &= x_3^2 \\ p_{13} &= -2y_3 \\ p_{14} &= y_3^2 \\ p_{15} &= \sqrt{a_1} \\ p_{21} &= -2x_0 \\ p_{22} &= x_0^2 \\ p_{23} &= -2y_0 \\ p_{24} &= y_0^2 \\ p_{25} &= \sqrt{a_2} \end{aligned}$$

E a solução do sistema por subtração das equações se torna:

$$\begin{aligned} p_{31} &= p_{21} - p_{11} \\ p_{32} &= p_{22} - p_{12} \\ p_{33} &= p_{23} - p_{13} \\ p_{34} &= p_{24} - p_{14} \\ p_{35} &= p_{25} - p_{15} \end{aligned}$$

$$\begin{aligned} p_{31}X + p_{32} + p_{33}y + p_{34} &= p_{35} \\ X &= \frac{p_{35} - (p_{32} + p_{34}) - p_{33}Y}{p_{31}} \\ p_{41}X &= \frac{p_{35} - p_{32}p_{34}}{p_{31}} \\ p_{42}X &= \frac{-p_{33}Y}{p_{31}} \\ X &= p_{41} + p_{42}Y \end{aligned}$$

Por fim realiza-se a substituição das variáveis novamente, possibilitando encontrar os valores de x_1 e x_2 :

$$\begin{aligned} ((p_{42}^2) + 1)X^2 + ((2p_{41}p_{42}) + (p_{11}p_{42}) + p_{13})X \\ + ((p_{41}^2) + (p_{11}p_{41}) + p_{14} + p_{12} - \sqrt{a_1}) = 0 \end{aligned} \quad (11)$$

E em sequência os valores de y_1 e y_2 :

$$Y = \frac{X - p_{41}}{p_{42}} \quad (12)$$

2.2 Modelo de cinemática direta

Uma vez que as posições dos pontos $P1$ e $P2$ são conhecidas, pode-se resolver o problema da cinemática direta, i.e, partindo da base do robô quais são os ângulos e posições entre os braços e os atuadores para que estes formem a configuração alvo determinada pela cinemática indireta. Para o robô em questão a solução representa quais os ângulos dos braços $link1_A$ e $link1_B$ em relação ao eixo X para que estes coloquem os pontos $P1$ e $P2$ nas posições x_1, y_1 e x_2, y_2 determinadas nas equações (11) e (12). Para isso basta utilizar as funções arco-seno e arco-cosseno destes pontos. Para a posição do ângulo θ_4 é necessário retornar ao robô com o link equivalente e realizar a transformação homogênea novamente utilizando a Tabela 2.

Tabela 2. Tabela de Parâmetros D-H

Link	a_i	α_i	d_i	θ_i
Virtual	e	0	0	θ_1
3	c	0	0	θ_2

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & eC_1 \\ S_1 & C_1 & 0 & eS_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

$$A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & cC_2 \\ S_2 & C_2 & 0 & cS_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

$$T_3^0 = A_1 \quad (15)$$

$$T_4^0 = A_1 A_2 = \begin{bmatrix} C_{12} & -S_{12} & 0 & eC_1 + cC_{12} \\ S_{12} & C_{12} & 0 & eS_1 + cC_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (16)$$

$$x = eC_1 + cC_{12} \quad (17)$$

$$y = eS_1 + cS_{12} \quad (18)$$

2.3 Limites de Área de trabalho, Singularidade e Similaridades

O robô em questão resolve a multiplicidade de soluções, similaridades, encontradas no robô planar de dois links discutido por Craig (2009). Cada uma das soluções na verdade resulta na posição de uma das juntas da cadeia fechada. Porém, este ainda tem singularidades e similaridades possíveis. Segundo a classificação de de Briot et al. (2015) podemos citar as seguintes situações:

- Quando os $link1_x$ e os $link2_x$ tendem a ficar colineares, se configura uma singularidade de tipo I, quando uma das variáveis das soluções cinemáticas passa a ser linearmente dependente de outra.
- Quando o ângulo entre os $link1_A$ e $link1_B$ tende a ser nulo, aparece uma singularidade do tipo II, ou uma similaridade, onde o movimento não pode ser previsto, os $link2_x$ ficam sem controle e os modelos de cinemáticas têm infinitas soluções.
- Quando os braços $link2_x$ se tornam colineares, configura-se uma singularidade tipo III. Neste caso o ângulo entre os braços $link1_x$ ficará travado e o movimento do ponto $P3$ ficará imprevisível, e mesmo que se introduza uma diretiva especial para que a ferramenta passe por esse ponto, ele também configura uma singularidade dinâmica onde o torque nos motores tende ao infinito.

A área de trabalho do robô em questão é representada graficamente por dois círculos concêntricos onde existe uma área de exclusão próxima à origem, onde se encontram os atuadores dos braços $link1_A$ e $link1_B$, para evitar colisões destes links com seus atuadores e a formação de uma singularidade tipo II, que ocorre quando $P3$ é posicionada na origem.

3. SIMULAÇÃO E IMPLEMENTAÇÃO

O sistema de comando de movimento do robô foi completamente modelado e simulado baseando-se no diagrama

proposto por Lewis et al. (2003), porém, alguns passos foram acrescentados como mostrado no diagrama da Figura 5.

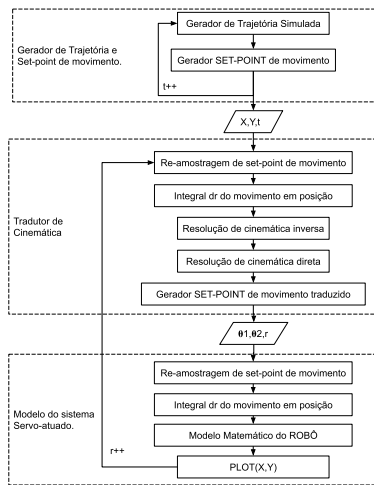


Figura 5. Modelo simulado do sistema

Este modelo computacional simulado foi concebido para ser prova de conceito para:

- Re-amostragem dos set-points de movimento;
- Tradução da cinemática cartesiana para a cinemática específica do robô, utilizando a solução de cinemática proposta;
- Re-geração dos set-points de movimento traduzidos;
- Geometria do robô.

O modelo computacional simulado pode ser separado em três partes principais:

- Comando de trajetória: um modelo de interpolação simples de três eixos.
- Tradutor de cinemática: captura o comando de movimento cartesiano e utiliza a cinemática inversa e direta do robô para calcular o movimento de juntas específico para que a geometria do robô execute o movimento desejado.
- Simulação do robô: utilizando a cinemática direta e de juntas para simular e desenhar a configuração do robô durante o movimento.

4. RESULTADOS

Os resultados foram obtidos comparando a trajetória comandada inicialmente no plano cartesiano com a trajetória executada no modelo do robô, passando pela tradução de cinemáticas baseada na solução cinemática proposta, como indicado na Figura 6. A função erro é a distância entre as duas trajetórias, demonstrando a eficácia do sistema.

Para se evitar pontos de imprevisibilidade das soluções cinemáticas algumas premissas e limitações da construção do robô foram utilizadas:

- A área de trabalho do robô em questão é formada por dois círculos concêntricos de forma que a ferramenta não consiga atingir a origem em qualquer orientação, evitando-se assim a colinearidade dos $link1_x$ neste ponto.

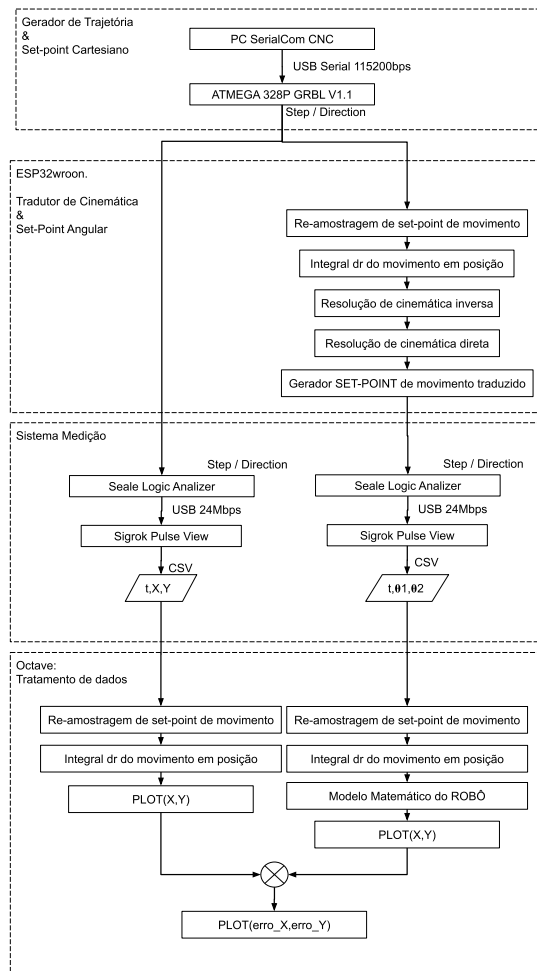


Figura 6. Modelo computacional implementado

- O círculo externo da área de trabalho é menor que o raio máximo quando os $link1_x$ e $link2_x$ se tornam colineares.
- Os $link2_x$ tem sempre comprimento maior que os $link1_x$ para se evitar a singularidade do tipo III.

Foram utilizadas três configurações de teste:

- Na primeira o sistema foi completamente simulado no Software computacional matemático *GNU Octave* em um PC para avaliar todas as etapas do processo em um ambiente sem as variações da aleatórias de um sistema real;
- Na segunda foi utilizado um comando de CNC de código e hardware abertos GRBL para comandar uma trajetória, gerando os set-points de movimento enviados diretamente ao hardware tradutor de cinemática on-line desenvolvido e implementado no SOC *Espressif ESP32*. Então os sinais de saída deste são adquiridos por um módulo de aquisição *Saleae Sigrok Pulse View* e novamente inseridos no modelo cinemático no *GNU Octave*;
- Na terceira configuração somente a geometria é testada em um modelo real, impresso em 3D.

4.1 Sistema completamente simulado

Na Figura 7 estão plotados os pontos de interesse da cinemática inversa. Estes pontos representam uma configuração solucionada para um ponto $P4$ e um ângulo de orientação da ferramenta. Os círculos $C1$, $C2$ e $C3$ são a representação geométrica do modelo cinemático inverso.

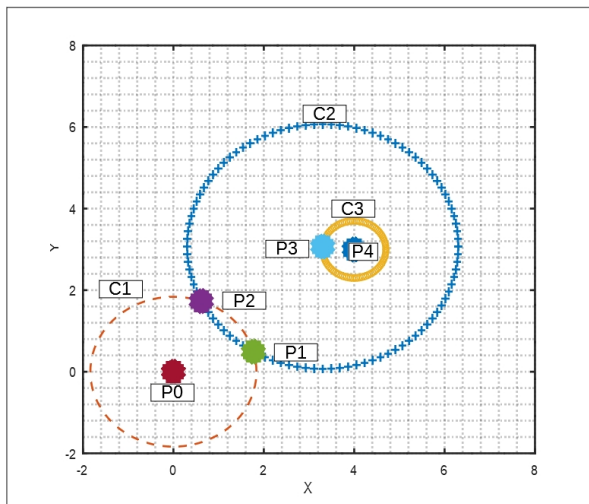


Figura 7. $P0, P1, P2, P3, P4, C1, C2$ e $C3$ calculados para uma posição alvo

Já na Figura 8 foram sobrepostos vetores utilizando os mesmos pontos de interesse para construir uma representação da posição dos links do robô nesta configuração.

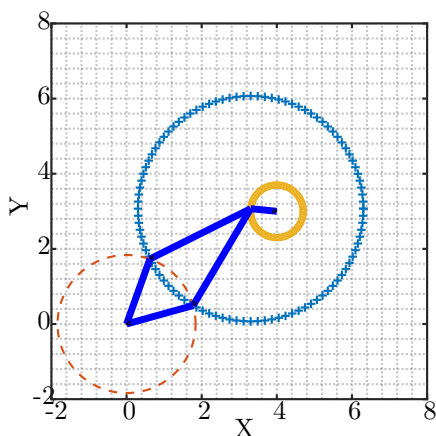


Figura 8. Configuração do robô, pela solução da cinemática inversa

De posse da configuração utiliza-se o modelo cinemático direto para o cálculo dos ângulos $arc1$, $arc2$ e $arc3$ que representam a posição angular dos braços para que a configuração seja atingida. Pode-se observar estes ângulos na Figura 9.

Após calcular os ângulos essa informação é discretizada, transformada em um trem de pulsos, re-amostrada e reintegrada de forma a se simular o sistema de *setpoint* de movimento entre o tradutor de cinemáticas e o *driver* de controle de movimento de cada eixo. Com a informação

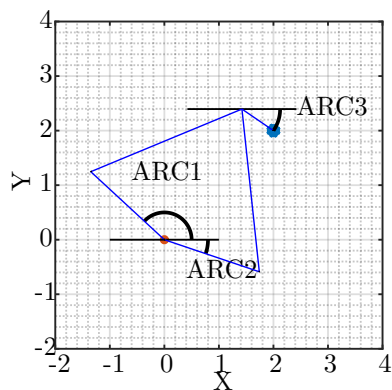


Figura 9. Posição a ser executada pelas juntas de acordo com a cinemática direta

dos ângulos reconstituída o modelo cinemático direto é novamente utilizado, porém agora para calcular a posição dos vetores que representam geometricamente o robô, sendo o resultado desta representação a plotagem da Figura 10.

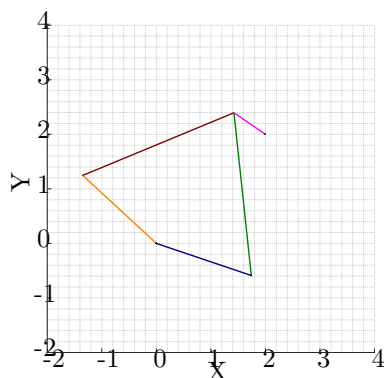


Figura 10. Representação gráfica dos links e juntas da configuração do robô após a tradução cinemática

Após a simulação para um ponto, foram realizadas simulações utilizando uma sequência de posições encadeadas, simulando assim a trajetória interpretada por um controlador de movimento cartesiano. O resultado para uma trajetória descrevendo uma linha com a orientação da ferramenta sendo alterada em $1rad$ pode ser observada na Figura 11.

Um segundo exemplo de trajetória pode ser observado na Figura 12, onde a orientação da ferramenta é alterada em um giro completo sem alteração da posição da mesma. Realizando assim um giro com TCP compensado.

4.2 Experimento com o tradutor de trajetória Embarcado

Na Figura 13 é demonstrado o resultado da comparação no teste utilizando o sistema embarcado real, onde está em vermelho a representação da trajetória comandada e os vetores em preto representam a trajetória executada pela tradução de cinemática e desenhada pelo modelo do robô no PC.

No teste foi obtido um erro máximo de 0.05mm e um erro médio de 0.02mm. Esses erros podem ser associados à

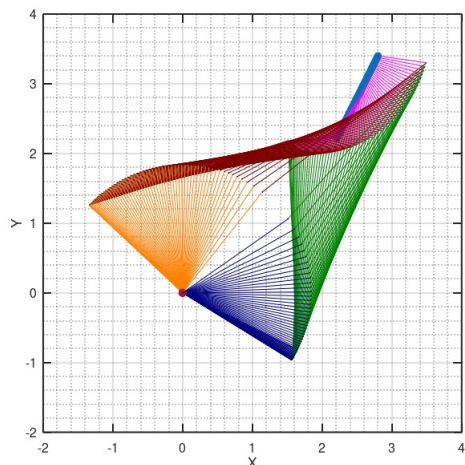


Figura 11. Exemplo de trajetória interpolada

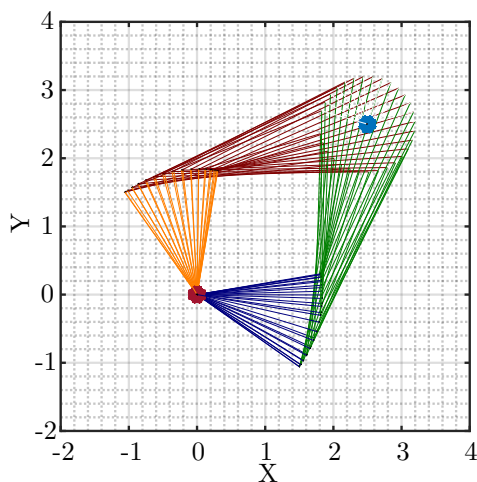


Figura 12. Exemplo de rotação com TCP compensado

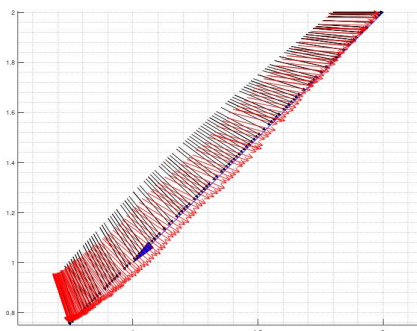


Figura 13. Comparação entre trajetória planejada e executada

resolução do sistema de tradução de cinemática. Na Figura 14 pode-se observar a montagem do sistema embarcado utilizado como comando de movimento, tradutor de trajetória e aquisição de sinais.

4.3 Modelo real impresso 3D:

Na Figura 15 está a montagem do robô real para validar a geometria.

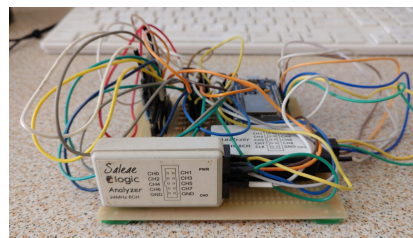


Figura 14. Sistema embarcado real implementado

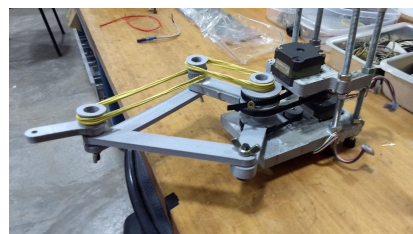


Figura 15. Implementação do robô real impresso em 3D

5. DISCUSSÃO E CONCLUSÃO

A geometria do robô escolhido é baseada numa plataforma de estudo sólida, com uma literatura abrangente e muitos estudos de casos validados. Os robôs planares com duas juntas rotacionais têm a condição desfavorável de possuírem duas configurações possíveis para cada ponto alvo de posição da ferramenta, como demonstrado na Figura 16. A utilização da geometria pantográfica de quatro barras remove essa ambiguidade de soluções. Dessa forma, uma solução resolve a posição de um par de braços e a segunda solução resolve a posição do segundo par, com a conveniência de se ter uma atuação concêntrica na base do robô, evitando a necessidade de se colocar um motor na junta do segundo braço ou de se utilizar um sistema de transmissão ao longo dos braços, reduzindo assim a inércia do robô.

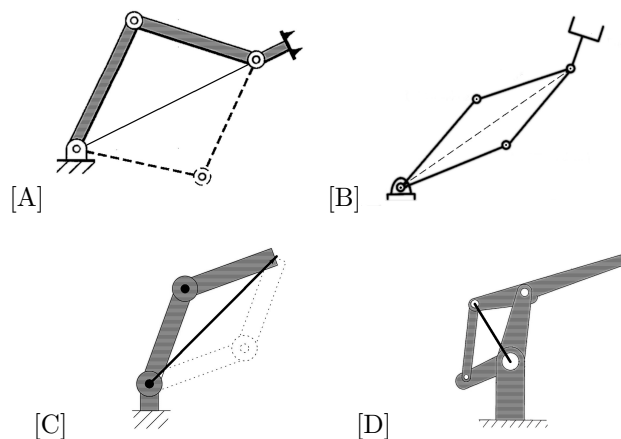


Figura 16. (A) Robô RR serial de Craig (2009) (B) Robô serial de dois braços de Tsai (1999) (C) Robô serial RR de Spong et al. (2006) (D) Robô de de dois links de cadeia fechada de Lewis et al. (2003)

O modelo cinemático do robô pode ser solucionado utilizando vários métodos, inclusive a junção destes. Isto proporciona que no estudo de cinemáticas robóticas possa-se traçar paralelos entre os métodos para compará-los,

bem como verificar se a solução atingida está correta, facilitando a modelagem e validação de novas geometrias de robôs.

As plataformas computacionais demonstraram serem capazes de suportar a implementação do trabalho. A utilização de plataformas de código aberto e hardware aberto como *Arduino*, *Sigrock* e *Octave*, acaba por facilitar a solução do problema proposto, já que essas possuem uma grande comunidade de suporte, provendo os recursos necessários para implementação de modelos cinemáticos computacionais. Assim o desenvolvedor pode manter o foco na resolução do problema proposto, e não precisar resolver questões secundárias como deficiências de bibliotecas, falta de documentação e custeio de aquisição de licenças privadas.

A complexidade matemática envolvida na solução dos modelos cinemáticos propostos é simples o bastante para que essas possam ser implementadas em sistemas de controle embarcados atuais, o que evita a utilização de soluções numéricas limitadas. Logo, comparando com os trabalhos anteriores, não foi necessário o devolvimento de um pós processador específico para converter o Gcode da trajetória desejada para um código que realiza o movimento necessário pela cinemática direta, o que impede a programação online do sistema robótico, como proposto por Bomfim et al. (2012). Assim o Gcode da trajetória cartesiana planejada pode ser utilizado diretamente no controlador de movimento, ou seja, um mesmo programa utilizado por um robô cartesiano poderia ser utilizado para controlar o robô híbrido em questão.

Outra observação importante é que as singularidades e similaridades encontradas neste robô são facilmente evidenciadas, facilitando o estudo da sua significância matemática e mecânica durante o comando e movimento do robô.

Em comparação com a área de trabalho de um robô cartesiano, temos que para um mesmo comprimento de links, o robô pantógrafo de 4 barras com atuação concêntrica, quando tem sua capacidade de atuar nos 4 quadrantes utilizada, tem uma área de trabalho bem superior. Isso porque mesmo com a limitação de alcance próxima a origem, ele tem a capacidade de executar um giro completo em torno da origem, enquanto um robô cartesiano fica limitado a somente um quadrante do espaço cartesiano.

AGRADECIMENTOS

À Far-x Tecnologia que supriu espaço, materiais e equipamentos para este desenvolvimento. Email: contato@far-x.com.br

REFERÊNCIAS

- Alvares, A.J., Toquica, J.S., Lima II, E.J., and Bomfim, M.H.S. (2017). Retrofitting of ASEA IRB2-S6 industrial robot using numeric control technologies based on LinuxCNC and MACH3-MatLab. In *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE. doi:10.1109/robio.2017.8324737.
- Alvares, A.J., Toquica, J.S., Lima II, E.J., and Bomfim, M.H.S. (2018). Retrofitting of the IRB6-S2 robotic manipulator using computer numerical control-based controllers. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 40(3). doi:10.1007/s40430-018-1073-0.
- Bomfim, M.H.S., Gontijo, R., Bracarense, A.Q., and Lima II, E.J. (2012). Overhauling of a ase robot ir6 with open architecture. In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, 482–489. IEEE.
- Briot, S., Khalil, W., et al. (2015). *Dynamics of parallel robots*. Springer.
- Craig, J.J. (2009). *Introduction to robotics: mechanics and control, 3/E*. Pearson Education India.
- Dâmaso, R.S. (2019). Validação experimental da modelagem cinemática de um robô manipulador baseado em kit didático. In *Congresso Brasileiro de Automática-CBA*, volume 1.
- de Lima, A.B., Barros, P.R., and Júnior, G.A. (2019). Plataforma robótica didática para ensino de identificação e sintonia de controladores pi. In *Congresso Brasileiro de Automática-CBA*, volume 1.
- Ferrao, E.F. (2018). *Avaliação da dinâmica de manipuladores robóticos sobre bases flexíveis*. Ph.D. thesis, Universidade Federal do Rio de Janeiro.
- Filho, F.A.R., Bracarense, A.Q., Lima II, E.J., Fernandes, J.C., and Ribeiro, E.B. (2010). Development of parallel manipulators for welding robots. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, volume 2, 1212–1217. VDE, Munich, Germany.
- FortuneBusinessInsight (2021). Industrial robots market. <https://www.fortunebusinessinsights.com/industry-reports/industrial-robots-market-100360>.
- Hartenberg, R.S. and Denavit, J. (1955). A kinematic notation for lower pair mechanisms based on matrices. *Journal of applied mechanics*, 77(2), 215–221.
- Lewis, F.L., Dawson, D.M., and Abdallah, C.T. (2003). *Robot manipulator control: theory and practice*. CRC Press.
- Lima II, E.J., Souza, H.A.d.M., Kienitz, K.M., and Ramalho Filho, F.A. (2020). Closed-form solution for direct and inverse kinematic models of a parallel manipulator for robotic orbital welding. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 42(1). doi:10.1007/s40430-019-2148-2.
- Merlet, J.P. (2006). *Parallel robots*, volume 128. Springer Science & Business Media.
- Sharon, A., Hogan, N., and Hardt, D.E. (1993). The macro/micro manipulator: An improved architecture for robot control. *Robotics and computer-integrated manufacturing*, 10(3), 209–222.
- Silva, G.G. and Giesbrecht, M. (2021). Solução de cinemática de plataforma de stewart baseada em aprendizado de máquina. In *Simpósio Brasileiro de Automação Inteligente-SBAI*, volume 1.
- Spong, M.W., Hutchinson, S., Vidyasagar, M., et al. (2006). *Robot modeling and control*.
- Tsai, L.W. (1999). *Robot analysis: the mechanics of serial and parallel manipulators*. John Wiley & Sons.
- Yoshida, K., Nenchev, D.N., and Uchiyama, M. (1996). Moving base robotics and reaction management control. In *Robotics Research*, 100–109. Springer.