

Cinemática Inversa Aplicada a Manipulador Robótico a Partir de Coordenadas Capturadas pelo Microsoft Kinect

Flavio Augusto Glapinski Zacca¹, Luma Wanderley de Oliveira²,
Gabriel Macedo de Carvalho¹, Rogério Pereira², Wanderson Rainer Hilário de Araújo¹

¹Pontifícia Universidade Católica de Goiás (PUC-GO)
Caixa Postal 86 – 74.605-010 – Goiânia – GO – Brasil

²Universidade Federal de Goiás (UFG)
Caixa Postal 131 – 74.001-970 – Goiânia – GO – Brasil

{flaviosye,luma.lwo,gabrielmcea,rogertecnic,wandersonrainer}@gmail.com

Abstract. *Some industrial production processes, operators of industrial plants are normally exposed to risks that can compromise health and safety at work. This article presents a proposal for a system to mitigate these risks, using a robotic manipulator associated with the Microsoft Kinect, which is used in order to obtain the operator's arm joints Cartesian coordinates and, remotely, applying inverse kinematics, to act in a robotic manipulator that performs the work of the operator and minimize potential risks. It was observed that the robotic arm presented a satisfactory result in terms of response time and positioning, however improvements are needed in the stability.*

Keywords: *Robotic manipulator, inverse kinematics, servomotors, Microsoft Kinect.*

Resumo. *Em muitos processos produtivos industriais, os operadores das plantas industriais normalmente são expostos a riscos que podem comprometer a saúde e a segurança no trabalho. Este artigo apresenta uma proposta de sistema para mitigação desses riscos, utilizando manipulador robótico associado ao Microsoft Kinect, que é empregado de forma a obter as coordenadas cartesianas das articulações do braço do operador e, remotamente, aplicando cinemática inversa, atuar em um manipulador robótico que realiza o trabalho do operador e, minimizar potenciais riscos. Observou-se que o braço robótico apresentou resultado satisfatório nos quesitos de tempo de resposta e posicionamento, entretanto melhorias são necessárias na estabilidade.*

Palavras-chave: *Manipulador robótico, cinemática inversa, servomotores, Microsoft Kinect.*

1. Introdução

Seja na indústria de processos contínuos ou na automação da manufatura, os operadores das plantas industriais normalmente são expostos a riscos que podem comprometer a saúde e a segurança no trabalho. Observa-se que a indústria é responsável por 37,2% dos acidentes de trabalho registrados com CAT (Comunicado de Acidente de Trabalho) e que, nas regiões mais industrializadas do Brasil, Sudeste e Sul, são responsáveis por 53,3% e 22,7% respectivamente dos acidentes de trabalho totais (AEAT, 2017) [da Fazenda (EAT) 2017].

No Brasil, a legislação vigente garante direitos e deveres aos trabalhadores e aos empregadores, para que se possa estabelecer um ambiente com riscos minimizados. Essa legislação é representada pelas normas regulamentadoras NR-10 [Regulamentadora 2016a] à NR-35 [Regulamentadora 2016b]. Nacionalmente, é recomendado que as áreas de uma instalação sejam classificadas pelo seu grau de risco, gerando um documento chamado mapa de riscos.

Essas regulamentações demonstram preocupações quanto aos riscos pelos quais operadores humanos são expostos, especialmente em ambientes industriais, onde essa exposição é frequente.

Diante desse cenário, os manipuladores robóticos têm participação substancial nas indústrias de alimentos e bebidas, com crescimento acentuado a partir do ano 2000, com o lançamento do robô *Flex Pick* [Iqbal et al. 2017], demonstrando que o uso de manipuladores robóticos além de estarem associados a atividades de embalagem, paletização, coleta e produção, dentre outras funções; também, podem minimizar os riscos que humanos são expostos em trabalho.

Além disso, a associação da robótica com visão computacional, como a utilização de Microsoft Kinect [Amatya and Petchartee 2015] para movimentar um dispositivo à distância, oferece ainda mais segurança no exercício de atividades perigosas. O Kinect é capaz de reconhecer o corpo e os movimentos do operador, transmitindo por meio de um software, esses movimentos com um grau confiável de precisão ao braço robótico. A tradução dos pontos cartesianos, obtidos através do Kinect, é realizada através da cinemática inversa, que será detalhada no decorrer do trabalho.

Desta forma, considerando a crescente participação dos robôs nos processos produtivos industriais e sua flexibilidade de atuação, este artigo utiliza manipuladores robóticos como agente para a realização de tarefas consideradas perigosas para seres humanos e, associa um Kinect para que o operador possa realizar o trabalho remotamente de forma a potencializar a segurança do operador, minimizando os riscos nos ambientes industriais.

2. Fundamentação Teórica

Esta seção irá apresentar os principais conceitos que regem o desenvolvimento deste artigo. Iniciando com o contexto de ambientes insalubres em processos produtivos industriais, o funcionamento de um manipulador robótico articulado, e por fim, apresentando os conceitos de cinemática.

2.1. Ambientes Insalubres em Processos Produtivos Industriais

Ambientes industriais em sua maioria oferecem riscos aos operadores, seja em atividades de produção ou manutenção. Com o objetivo de minimizar esses riscos, há trabalhos que desenvolvem soluções robóticas para substituir o ser humano na realização de tarefas [Lee et al. 2015]; outros, apresentam técnicas de projeto para minimizar riscos em ambientes com alta periculosidade [Parikh et al. 2011, Srinivasan et al. 2018, Leon-Rodriguez et al. 2013].

Além das atividades realizadas em ambientes insalubres, outras atividades poderiam ser beneficiadas com a utilização de robôs, atendendo assim diretamente às necessidades impostas pelas normas regulamentadoras:

- Segurança em Instalações e Serviços em Eletricidade - NR10 [Regulamentadora 2016a]:
 - Esta norma estabelece os requisitos e condições mínimas objetivando a implementação de medidas de controle e sistemas preventivos, de forma a garantir a segurança e a saúde dos trabalhadores que, direta ou indiretamente, interajam em instalações elétricas e serviços com eletricidade.
- Atividades e Operações Insalubres - NR15 [Regulamentadora 2007]:
 - Esta norma estabelece que são consideradas atividades insalubres aquelas que se desenvolvem acima dos limites de tolerância previstos na norma.
- Atividades e Operações Perigosas - NR16 [Regulamentadora 2015]:
 - Esta norma estabelece que são consideradas atividades perigosas aquelas que estão presentes nos anexos da norma.
- Atividades em Espaços Confinados - NR33 [Regulamentadora 2012]:
 - Esta norma tem como objetivo estabelecer requisitos mínimos para identificação de espaços confinados, para monitorar e controlar os riscos existentes.

2.2. Manipulador Robótico Articulado

No estudo da robótica, há o interesse na localização de elementos do robô e objetos no espaço tridimensional. Esses elementos são os elos do manipulador e ferramentas com as quais lidam com os objetos em seu ambiente. Os manipuladores consistem em elos rígidos que são conectados por juntas, as quais permitem o movimento relativo dos elos vizinhos. Essas juntas são geralmente equipadas com sensores de posição, os quais permitem que a posição relativa dos elos vizinhos seja medida. No caso das juntas rotacionais (ou de revolução), tais deslocamentos são chamados ângulos de junta. Alguns manipuladores contêm juntas deslizantes (ou prismáticas), nas quais o deslocamento relativo entre os elos é uma translação, às vezes chamada de deslocamento da junta [Mikell 2011]. Neste artigo, os conceitos serão aplicados a um manipulador como o ilustrado na Figura 1.

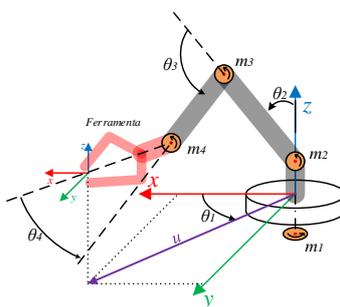


Figura 1. Ilustração do Manipulador Robótico Articulado.

Fonte. Autoria própria.

Os robôs por juntas articuladas, Figura 1, se assemelham muito ao braço humano, uma vez que eles podem possuir várias juntas e graus de liberdade para se movimentar, podendo alcançar locais de difícil acesso para realização de atividades complexas, o que é algo versátil numa indústria.

O manipulador mecânico ilustrado possui quatro elos e quatro juntas. Ao último elo normalmente é conectado um dispositivo denominado efetuador, ilustrado em vermelho como ferramenta na Figura 1. O efetuador é o dispositivo empregado pelo robô para executar determinada tarefa, podendo ser uma máquina para solda, uma garra, uma máquina de pintura, entre outras ferramentas.

2.3. Cinemática Direta e Inversa

Utilizando a cinemática direta, é possível determinar a localização do manipulador robótico em um plano cartesiano, uma vez que as coordenadas articulares do robô são conhecidas. De modo semelhante, utilizando a cinemática inversa, é possível determinar as coordenadas articulares de um braço robótico, uma vez que a localização do manipulador em um plano cartesiano é conhecida. As equações de cinemática inversa em duas dimensões são:

$$X = L_1 \cdot \cos\Theta_1 + L_2 \cdot \cos(\Theta_1 + \Theta_2) + L_3 \cdot \cos(\Theta_1 + \Theta_2 + \Theta_3)$$

$$Y = L_1 \cdot \sen\Theta_1 + L_2 \cdot \sen(\Theta_1 + \Theta_2) + L_3 \cdot \sen(\Theta_1 + \Theta_2 + \Theta_3)$$

$$\phi = (\Theta_1 + \Theta_2 + \Theta_3)$$

Onde as variáveis são:

- Variáveis de entrada:
 - θ_1 : ângulo da junta 1.
 - θ_2 : ângulo da junta 2.
 - θ_3 : ângulo da junta 3.
- Variáveis de saída:
 - X : Deslocamento do ponto no eixo das abcissas.
 - Y : Deslocamento do ponto no eixo das ordenadas.
 - ϕ : Soma dos ângulos dos elos em relação ao eixo das abcissas.
- Constantes utilizadas nas equações descritas na cinemática:
 - L_1 = Comprimento do elo 1.
 - L_2 = Comprimento do elo 2.
 - L_3 = Comprimento do elo 3.

Utilizando essas equações é possível encontrar os ângulos que levam o manipulador robótico até o ponto desejado.

A fim de descrever a posição e a orientação de um corpo no espaço, é comum empregar um sistema de coordenadas (ou sistema de referência). Na Figura 1, são apresentadas as coordenadas x , y e z da junta da base e do efetuador. Definidas as coordenadas, é necessário descrever a posição e orientação desse sistema de referência em relação a algum sistema de coordenadas de referência. Qualquer referencial pode servir como sistema de referência de base pelo qual expressa a posição e a orientação do corpo, de forma que quase sempre é necessário transformar ou mudar a descrição desses atributos do corpo, e um sistema para outro.

Na ciência da cinemática estudam-se grandezas como posição, velocidade, aceleração e todas as derivadas de ordem superior das variáveis de posição (com relação ao tempo ou quaisquer outras variáveis). Portanto, o estudo da cinemática dos manipuladores refere-se a todas as propriedades do movimento, tanto geométricas quanto baseadas no tempo.

Para a representação de um sistema mecânico articulado, utiliza-se cadeias cinemáticas, nas quais para cada elo é atribuído um sistema de coordenadas denominado frame, sendo o primeiro frame atribuído a base fixa do sistema, denominado frame 0, pois a base sendo conectada ao primeiro elo através da primeira junta, exerce o papel do elo de número zero, ou seja, quando a i -ésima junta se movimenta, o elo $i+1$ que se move, não o elo i , juntamente com ele o frame $i+1$ se movimenta, a representação de um ponto no espaço com relação a um frame é obtida através de matrizes de transformações homogêneas. O posicionamento dos frames é fundamental para a modelagem da cinemática.

A existência ou a inexistência de uma solução cinemática define o espaço de trabalho do manipulador. A ausência de uma solução significa que o manipulador não poderá chegar à posição, nem à orientação desejadas, porque está fora do espaço de trabalho do manipulador [Craig 2005].

Um problema comum no estudo da manipulação mecânica é a cinemática direta. Trata-se do problema de geometria estática de computar a posição e a orientação do efetuador do manipulador. Em termos específicos, dado um conjunto de ângulos de junta, o problema da cinemática direta é computar a posição e a orientação do sistema de referência da ferramenta, com relação ao sistema da base. Para o manipulador da Figura 1, a cinemática direta é a obtenção das coordenadas x , y e z do efetuador a partir dos valores dos ângulos θ_1 , θ_2 , θ_3 e θ_4 , dos motores m_1 , m_2 , m_3 e m_4 respectivamente [Spong et al. 2004].

Por outro lado, a cinemática inversa tem como objetivo realizar o cálculo inverso a partir das coordenadas x , y e z do efetuador, obtendo os valores dos ângulos das juntas (θ_1 , θ_2 , θ_3 e θ_4) que levem o efetuador para a posição desejada. O que torna a cinemática inversa complexa é o fato de sua solução não ser unívoca, ou seja, vários conjuntos de ângulos que levam o efetuador para a mesma posição e orientação no espaço.

3. Metodologia

Considerando a base teórica exposta na seção anterior, os conceitos apresentados são aplicados a um sistema, no qual deseja-se desenvolver a cinemática inversa de um manipulador robótico, a partir das coordenadas cartesianas do elemento efetuador. Contudo, as coordenadas x , y e z do efetuador devem ser obtidas via Microsoft Kinect. A Figura ?? ilustra os elementos integrantes do sistema desenvolvido.

O operador do sistema tem sua imagem capturada pelo Microsoft Kinect. As coordenadas das mãos do operador são transferidas do Kinect para uma aplicação desenvolvida na plataforma .NET. Em seguida são aplicados processos para condicionamento dos valores obtidos, para acionamento do manipulador robótico conforme a gesticulação do operador, permitindo a operação do robô remotamente. A seguir são apresentados os procedimentos realizados para a integração de todo o sistema.

3.1. Ajustes no Manipulador Robótico

O manipulador empregado no desenvolvimento do sistema, é um braço robótico com cinco juntas e de característica DIY (*do it yourself*), com finalidade de aplicações entusiastas, apresentando características rudimentares se comparado com manipuladores industriais. Os servomotores utilizados na montagem possuem alcance de movimentação limitado (0° a 180°).

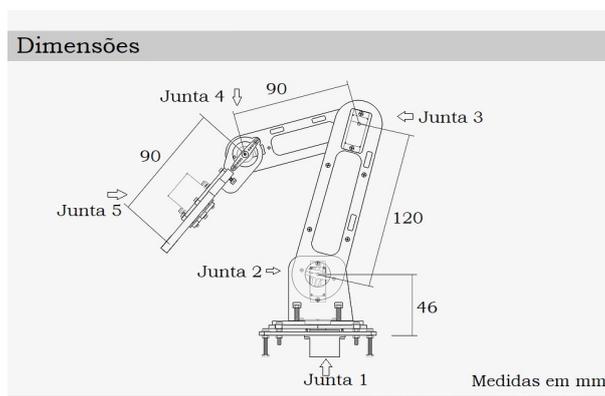


Figura 2. Diagrama do Manipulador Robótico empregado.

Fonte. Autoria própria.

Durante os testes, foi observado que a precisão do ângulo do motor varia em função da posição do servo, para tal foi desenvolvida equação polinomial que pode ser observada em (equação 15). A Figura 3 ilustra o diagrama do manipulador empregado no projeto, destacando as juntas e os elos da estrutura.

Pode-se observar que o manipulador robótico empregado neste estudo possui cinco juntas e quatro elos, sendo a junta 5 responsável pela operação do efetuador e não influenciando no posicionamento. Contudo, foram necessários alguns ajustes para o correto funcionamento do robô.

Para atender os requisitos do projeto, novos componentes foram modelados e, em seguida, foram fabricados por manufatura aditiva, com auxílio de impressora 3D, as medidas foram condizentes ao estipulado em desenho, tendo uma variação inferior a 1 mm.

3.2. Implementação do SDK Kinect

O software de controle foi desenvolvido na plataforma .NET utilizando a IDE (*Integrate Development Environment*) Visual Studio 2019 através da ferramenta *Windows Forms*, programado em linguagem C#, possui integração direta com o SDK do Microsoft Kinect e utilitários para o desenvolvimento da interface gráfica, bem como controle e gerenciamento de comunicação via porta serial, necessária para comunicação com o Arduino. Na Figura 3, pode ser visto o fluxograma do código desenvolvido para operação do Kinect.

3.3. Normalização dos Espaços de Trabalho

Na aplicação desenvolvida, é realizado um processo de normalização entre o espaço de trabalho definido pelo operador e o espaço de trabalho do manipulador robótico. Este processo é necessário devido ao fato de que o espaço de trabalho do manipulador robótico, é menor do que o espaço de trabalho definido pelo operador durante a etapa de sincronismo.

Este espaço de trabalho do manipulador é menor devido as suas dimensões, que serão apresentadas posteriormente. Pelo fato dos eixos coordenados capturados pelo Microsoft Kinect serem diferentes dos eixos do manipulador, é necessário uma transformação do sistema de coordenadas.

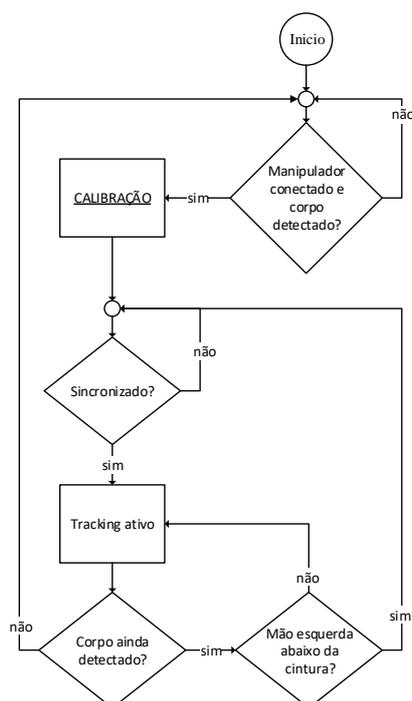


Figura 3. Fluxograma do funcionamento do código da aplicação .NET.

Fonte. Autoria própria.

Assim, dado um ponto no espaço do operador que indica a posição da mão, é feita a rotação dos eixos e logo em seguida a normalização descrita pela equação 1 é aplicada, esta equação é utilizada para cada uma das três coordenadas, assim a transformação de um espaço para outro é realizada. A equação 1 é referente a coordenada x . Para as coordenadas y e z é aplicada a mesma equação, com os limites referentes aos respectivos eixos.

$$x_m = \frac{(x_{1_m} - x_{0_m}) * (x_k - x_{0_k})}{(x_{1_k} - x_{0_k})} + x_{0_m} \quad (1)$$

Onde as variáveis são:

- x_m : coordenada x do manipulador no espaço do manipulador;
- x_k : coordenada x da mão direita do operador no espaço do manipulador;
- x_{1_m} : limite superior do eixo x do espaço do manipulador;
- x_{0_m} : limite inferior do eixo x do espaço do manipulador;
- x_{1_k} : limite superior do eixo x do espaço do operador (Microsoft Kinect);
- x_{0_k} : limite inferior do eixo x do espaço do operador (Microsoft Kinect).

Sendo assim, de posse das coordenadas já normalizadas para o espaço de trabalho do manipulador, o algoritmo de cinemática inversa que será apresentada posteriormente é aplicado, se o ponto indicado pelas coordenadas estiver dentro do alcance do manipulador, ou seja, dentro do espaço de trabalho, a cinemática inversa retorna os ângulos das juntas $\theta_1, \theta_2, \theta_3$ e θ_4 responsáveis por fazer o efetuador do manipulador atingir o ponto desejado.

Se o ponto desejado não estiver no espaço de trabalho, a cinemática inversa retorna todos os quatro ângulos iguais a 0° , que correspondem a posição neutra, ou de referência, que será ilustrada posteriormente.

Logo após aos ângulos das juntas são adicionados os *offsets*, denominados como θ_i , para a obtenção dos ângulos dos servomotores θ_f . Contudo, os servomotores apresentam um erro de posicionamento. Para a correção deste erro, é aplicada uma equação resultada da regressão polinomial de um conjunto de posições coletados para cada servomotor, os ângulos corrigidos são denominados θ_{fm} . Estes ângulos por sua vez, são enviados para o Arduino através de comunicação serial. O Arduino ao receber os ângulos, realiza o envio para seus respectivos servomotores por sinais PWM *Pulse Width Modulation*. Nesta etapa, a velocidade de movimento dos servomotores é controlada através do envio de posições intermediárias, entre a posição atual do servomotor e a próxima posição desejada, melhorando a estabilidade e suavidade do movimento do manipulador como um todo.

3.4. Cinemática Inversa

O manipulador possuindo quatro graus de liberdade necessita, de no mínimo, quatro variáveis para descrever sua posição no espaço. Foram escolhidas as variáveis de posição linear (x, y, z) e a inclinação da garra em relação ao solo (denominada pela letra g), como pode ser visto na Figura 4. Para que a cinemática inversa possa apresentar apenas uma solução, foi considerado que o ângulo da junta três nunca será negativo, isso implica que o espaço de manipulação do manipulador será reduzido, não permitindo que o mesmo opere objetos atrás de sua base.

As variáveis envolvidas nos cálculos da cinemática direta são:

- Variáveis de entrada:
 - x : coordenada da garra em relação ao eixo x do frame base.
 - y : coordenada da garra em relação ao eixo y do frame base.
 - z : coordenada da garra em relação ao eixo z do frame base.
 - g : ângulo da garra com relação ao solo (*Figura 4*).
- Variáveis de saída:
 - θ_1 : ângulo da junta 1.
 - θ_2 : ângulo da junta 2.
 - θ_3 : ângulo da junta 3.
 - θ_4 : ângulo da junta 4.
- Constantes utilizadas nas equações descritas na cinemática:
 - $L_1 = 46$.
 - $L_2 = 120$.
 - $L_3 = 90$.
 - $L_4 = 90$.

O ângulo θ_1 , que pode ser observado na Figura 4, é dado pela equação 2:

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

Para os demais cálculos, o manipulador é considerado planar e toda a sua estrutura no mesmo plano conforme Figura 4, logo a posição do manipulador pode ser descrita

apenas por duas coordenadas neste plano, estas coordenadas são u e z , o ângulo g também não sofre alteração, visto que este plano está perpendicular ao plano da base. O plano no qual o manipulador está localizado é denominado uz e as variáveis de posição são dadas pela coordenada sobre o eixo u , calculada de acordo com a equação 3, pela coordenada z e pela inclinação da garra g .

$$u = \sqrt{x^2 + y^2} \quad (3)$$

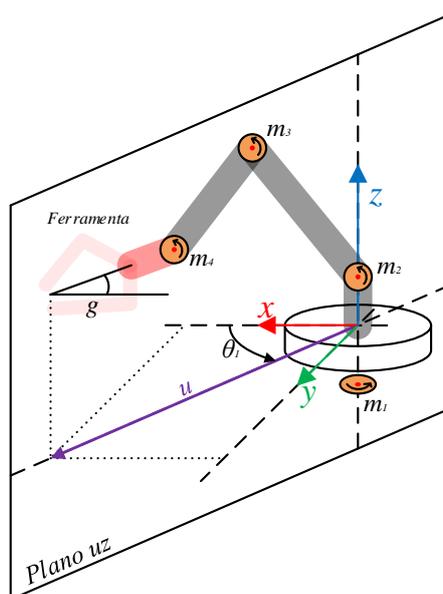


Figura 4. Detalhes de θ_1 , g e do plano uz .
Fonte. Autoria própria.

Através do ângulo g e do tamanho dos elos, as posições das juntas 2 e 4 são calculadas, sendo os vetores \vec{P}_2 e \vec{P}_4 , vistos na Figura 5, conforme as equações 4 e 5.

$$\vec{P}_2 = (0, 46) \quad (4)$$

$$\vec{P}_4 = (u - L_4 * \cos(g), z + L_4 * \sin(g)) \quad (5)$$

Os vetores \vec{P}_2 e \vec{P}_4 são utilizados para calcular o vetor \vec{h} , conforme equação 6.

$$\vec{h} = \vec{P}_2 - \vec{P}_4 \quad (6)$$

O vetor \vec{h} , juntamente com os elos L_2 e L_3 , formam um triângulo observado na Figura 5, com cor roxa, os ângulos internos deste triângulo são denominados θ_{hL_2} , $\theta_{L_2L_3}$ e θ_{hL_3} como pode ser visto na Figura 5. Estes ângulos são calculados utilizando a lei dos cossenos, conforme as equações 7, 8 e 9.

$$\theta_{hL_2} = \arccos \left(\frac{\|\vec{h}\|^2 + L_2^2 - L_3^2}{2 * \|\vec{h}\| * L_2} \right) \quad (7)$$

$$\theta_{L_2L_3} = \arccos \left(\frac{L_2^2 + L_3^2 - \|\vec{h}\|^2}{2 * L_2 * L_3} \right) \quad (8)$$

$$\theta_{hL_3} = \arccos \left(\frac{\|\vec{h}\|^2 + L_3^2 - L_2^2}{2 * \|\vec{h}\| * L_3} \right) \quad (9)$$

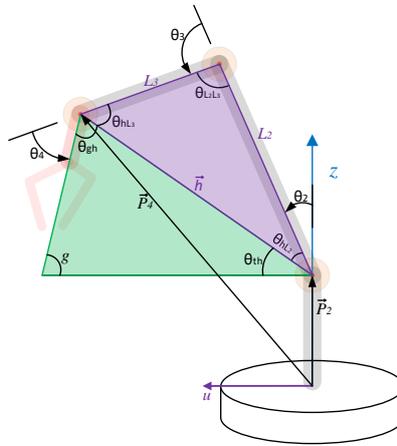


Figura 5. Vetores \vec{P}_2 , \vec{P}_4 e triângulo formado pelas juntas 2, 3 e 4.
Fonte. Autoria própria.

Também é calculado o ângulo θ_{th} que é o ângulo entre o vetor \vec{h} e o plano da base através da equação 10, utilizando a coordenada u do vetor \vec{h} h_u .

$$\theta_{th} = \arccos \left(\frac{h_u}{\|\vec{h}\|} \right) \quad (10)$$

É importante, no cálculo dos ângulos das juntas, observar que os parâmetros angulares calculados até o momento não possuem sentido, logo todos os valores são positivos. Já os ângulos das juntas possuem sentido de giro, portanto podem ser negativos, apenas o ângulo da junta 3 será sempre positivo.

Os ângulos θ_2 e θ_3 são calculados utilizando as equações 11 e 12.

$$\theta_2 = 90 - (\theta_{th} + \theta_{hL_2}) \quad (11)$$

$$\theta_3 = 180 - \theta_{L_2L_3} \quad (12)$$

Na Figura 5, desenhado em verde, está um triângulo formado pelo vetor \vec{h} e que dois de três ângulos internos são conhecidos, sendo g e θ_{th} . O terceiro ângulo denominado

θ_{gh} é facilmente obtido, sabendo-se que a soma dos ângulos internos de um triângulo é igual a 180° , conforme equação 13. Logo, o ângulo θ_4 pode ser obtido através da equação 14.

$$\theta_{gh} = 180 - \theta_{th} - g \quad (13)$$

$$\theta_4 = 180 - \theta_{gh} - \theta_{hL_3} = \theta_{th} + g - \theta_{hL_3} \quad (14)$$

Se a posição desejada da garra se encontrar fora do espaço de trabalho do manipulador, não é possível o cálculo das funções arccos. A verificação é feita através do módulo de \vec{h} , se este valor for maior que $L_2 + L_3$, então não é possível fechar o triângulo roxo, logo a posição é inválida e o algoritmo da cinemática inversa é interrompido.

4. Resultados

Nesta seção serão apresentados os resultados dos procedimentos realizados. Será apresentado o que foi obtido com a integração do Microsoft Kinect e o manipulador robótico quando submetido à cinemática inversa.

4.1. Ajustes no Manipulador Robótico

Os servomotores utilizados nas juntas, possuem um alcance de movimentação limitado (0° a 180°), porém cada junta possui um alcance específico, delimitado por impedimentos físicos, causando assim redução de amplitude de movimento. Essa limitação é contornada via código, feita através da verificação dos ângulos antes do envio deles ao Arduino.

Conforme exposto anteriormente, o manipulador empregado neste estudo possui ângulo de *offset* a serem considerados durante a aplicação da cinemática. Os ângulos iniciais θ_i foram observados para cada uma das juntas m_1 , m_2 e m_3 e m_4 . Desta forma, após a obtenção dos ângulos para alcançar as coordenadas desejadas através da cinemática, deve-se considerar os ângulos θ_i de cada servomotor das juntas.

Além disso, o ângulo final θ_f possui relação com o ângulo θ_{fm} a ser aplicado ao servomotor da respectiva junta. Neste caso, foram levantados pontos experimentais, de forma a identificar os valores a serem escritos em cada servomotor em função da posição final desejada. A Figura 6 ilustra os pontos levantados experimentalmente e a interpolação polinomial de terceiro grau para obtenção do valor de θ_{fm} .

O polinômio encontrado, para relacionar os pontos obtidos experimentalmente, retornam o valor corrigido a ser enviado ao servomotor da junta m_2 é:

$$\theta_{fm_2}(\theta_f) = 8,13^{-6} \cdot \theta_f^3 + 0,0018 \cdot \theta_f^2 + 0,9904 \cdot \theta_f - 7,9932 \quad (15)$$

O mesmo procedimento foi realizado para as juntas m_3 e m_4 , resultando nos gráficos das Figuras 7 e 8.

A fim de suavizar a movimentação do manipulador, a implementação do código no Arduino faz uso de uma biblioteca para controle de servomotores (VarSpeedServo.h) que, além de controlar as posições dos servomotores individualmente, também controla a

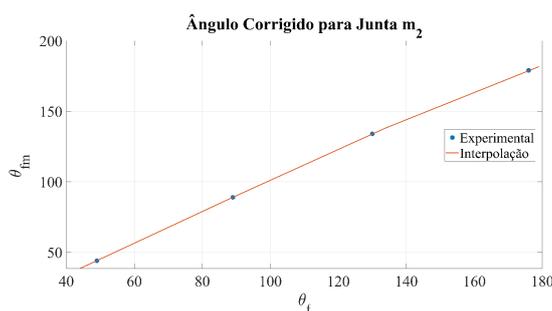


Figura 6. Relação entre o ângulo do servomotor θ_{fm} e a posição final θ_f para o servomotor da junta m_2 .

Fonte. Autoria própria.

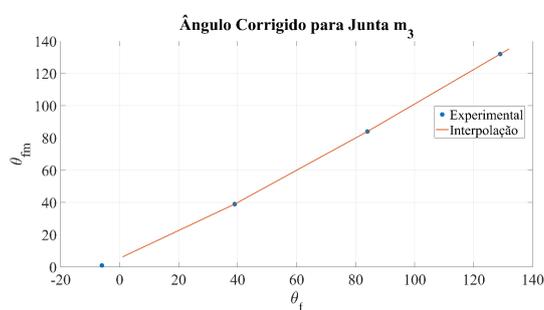


Figura 7. Relação entre o ângulo do servomotor θ_{fm} e a posição final θ_f para o servomotor da junta m_3 .

Fonte. Autoria própria.

velocidade. Desta forma, é possível determinar uma velocidade de movimentação diferente para cada servomotor e diferente da velocidade máxima, que é a única velocidade com que o servomotor se movimenta ao ser escrito nele o valor de posição, já que os servomotores em si não possuem controle de velocidade.

Quando a função de escrita no servomotor é chamada, o argumento a ser passado para ela é a posição final desejada de 0° a 180° e a velocidade de 0 a 255, que equivale de 0 a 100% da velocidade máxima do servo.

4.2. Implementação do SDK e do Microsoft Kinect

A aplicação desenvolvida em .NET utilizando a IDE Visual Studio como descrita anteriormente, foi desenhada e programada ao longo do desenvolvimento de todo o trabalho, as funcionalidades colocadas nela tiveram papel fundamental em todas as etapas do trabalho, principalmente nos testes de posicionamento individual das juntas.

A tela foi dividida em regiões, nas quais os controles e indicadores diversos estão organizados, as regiões, bem como seus controles e indicadores são descritos a seguir:

- Região "*mão direita*": indicadores das coordenadas x , y e z que o Microsoft Kinect coleta da mão direita.
- Região "*calibração*": campos que indicam os limites resultantes da calibração,

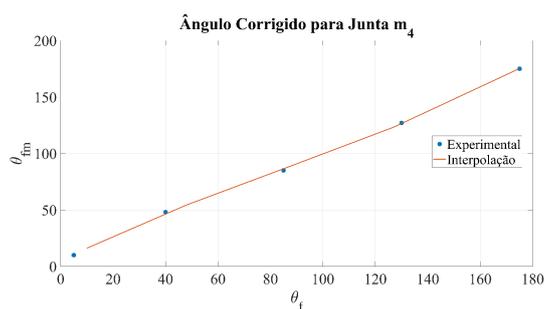


Figura 8. Relação entre o ângulo do servomotor θ_{fm} e a posição final θ_f para o servomotor da junta m_4 .

Fonte. Autoria própria.

estes campos são alterados apenas durante a calibração e mantêm os valores durante toda a operação.

- Região "*Serial*": controles referentes ao gerenciamento da conexão serial com o Arduino.
- Região "*flags de status*": se modificam de acordo com a execução do controle, como ilustrado na Figura 3.
- Região "*Controles individuais do braço em graus*": controles deslizantes que permitem o ajuste manual da posição angular de cada servomotor individualmente.
- Região "*Controles cartesianos*": controles deslizantes são similares aos controles descritos no item anterior, eles controlam diretamente a posição x , y e z do manipulador no espaço de trabalho e a inclinação da garra com relação ao plano da base.

4.3. Validação

Foram realizados testes de validação, para análise de desempenho do braço robótico em relação às coordenadas obtidas através do Kinect, referentes a:

- Tempo de resposta: o atraso entre o movimento do operador e a resposta do braço robótico foi inferior a 200ms, satisfazendo o requisito.
- Posicionamento: foi possível manipular objetos dentro do espaço de trabalho através do braço robótico, podendo alcançar, agarrar, suspender e levar até novo local, atendendo a expectativa.
- Estabilidade: o braço robótico apresentava vibração, devido a baixa precisão dos elos e a baixa estabilidade do robô.

5. Conclusões

O Microsoft Kinect pode ser utilizado para a operação de manipuladores robóticos com algumas restrições, devido a imprecisão de leitura quando os pontos de referência do corpo (mãos, cotovelos, ombros, cabeça, quadril e tronco) que se sobrepõem em frente ao campo de visão do Microsoft Kinect.

A estrutura do manipulador robótico, bem como os servomotores, são a maior fonte de imprecisão no posicionamento e movimentação, logo ela apresenta possibilidades de melhoria, desde remodelagem dos elos e juntas, até a substituição dos servomotores.

O método utilizado para a determinação dos limites dos eixos coordenados do espaço de trabalho do manipulador foi empírico, a sua precisão é satisfatória para a aplicação deste artigo no meio acadêmico, porém para aplicações nas quais o manipulador utilizado é de alta precisão, este método pode ser elaborado de tal forma a obter os limites cartesianos, utilizando o algoritmo da cinemática inversa e o alcance angular das juntas, através de métodos de maximização.

Como trabalho futuro, este projeto será implementado em braço robótico industrial, assim contornando as limitações mecânicas apresentadas durante o projeto.

Referências

- Amatya, S. and Petchartee, S. (2015). Real time kinect based robotic arm manipulation with five degree of freedom. In *2015 Asian Conference on Defence Technology (ACDT)*, pages 1–6. IEEE.
- Craig, J. J. (2005). Introduction to robotics.
- da Fazenda (EAT), M. (2017). Anuário estatístico de acidentes.
- Iqbal, J., Khan, Z. H., and Khalid, A. (2017). Prospects of robotics in food industry. *Food Science and Technology*, 37(2):159–165.
- Lee, W. C., Salam, A. S. A., Ibrahim, M. F., Rahni, A. A. A., and Mohamed, A. Z. (2015). Autonomous industrial tank floor inspection robot. In *2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 473–475. IEEE.
- Leon-Rodriguez, H., Sattar, T., and Park, J.-O. (2013). Wireless climbing robots for industrial inspection. In *IEEE ISR 2013*, pages 1–4. IEEE.
- Mikell, P. (2011). *Automação industrial e sistemas de manufatura*. Pearson Brasil.
- Parikh, S., Sustarich, M. J., Hulett, R. H., and Reitler, E. (2011). Considering process temperatures when designing electrical heat tracing for hazardous areas. In *2011 Record of Conference Papers Industry Applications Society 58th Annual IEEE Petroleum and Chemical Industry Conference (PCIC)*, pages 1–5. IEEE.
- Regulamentadora, N. (2007). Nr-15: Atividades e operações insalubres. *Brasília: MTE*.
- Regulamentadora, N. (2012). Nr-33: Segurança e saúde nos trabalhos em espaços confinados. *Brasília: MTE*.
- Regulamentadora, N. (2015). Nr-16: Atividades e operações perigosas. *Brasília: MTE*.
- Regulamentadora, N. (2016a). Nr-10: Segurança em instalações e serviços em eletricidade. *Brasília: MTE*.
- Regulamentadora, N. (2016b). Nr-35: Trabalho em altura. *Brasília: MTE*.
- Spong, M. W., Hutchinson, S., and Vidyasagar, M. (2004). Robot dynamics and control second edition.

Srinivasan, K., Harikrishnan, V., Venkatesh, R. V., and Gowtham, S. (2018). Smart monitoring and auto-neutralization of ph of alkaline liquids in process industries. In *2018 2nd International Conference on Smart Sensors and Application (ICSSA)*, pages 99–104. IEEE.

