

Rogério Pereira Batista

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

Brasil

2019

---

**TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR  
VERSÕES ELETRÔNICAS DE TRABALHO DE CONCLUSÃO DE CURSO DE  
GRADUAÇÃO NO REPOSITÓRIO INSTITUCIONAL DA UFG**

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio do Repositório Institucional (RI/UFG), regulamentado pela Resolução CEPEC nº 1204/2014, sem ressarcimento dos direitos autorais, de acordo com a Lei nº 9610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou *download*, a título de divulgação da produção científica brasileira, a partir desta data.

**1. Identificação do Trabalho de Conclusão de Curso de Graduação (TCCG):**

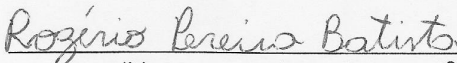
Nome completo do autor: Rogério Pereira Batista

Título do trabalho: Cinemática Inversa Aplicada a Manipulador Robótico a Partir de Coordenadas Capturadas pelo Microsoft Kinect


**2. Informações de acesso ao documento:**

Concorda com a liberação total do documento [ X ] SIM [ ] NÃO<sup>1</sup>

Havendo concordância com a disponibilização eletrônica, torna-se imprescindível o envio do(s) arquivo(s) em formato digital PDF do TCCG.

  
(Nome completo do autor)<sup>2</sup>

Ciente e de acordo:

  
(Nome completo do orientador)<sup>2</sup>

Data: 30/12/2019

---

<sup>1</sup> Neste caso o documento será embargado por até um ano a partir da data de defesa. A extensão deste prazo suscita justificativa junto à coordenação do curso. Os dados do documento não serão disponibilizados durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro;
- Publicação da dissertação/tese em livro.

<sup>2</sup> As assinaturas devem ser originais sendo assinadas no próprio documento, imagens coladas não serão aceitas.

Rogério Pereira Batista

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

Trabalho de Conclusão de Curso de Graduação, apresentado à disciplina TCC, do curso de Engenharia Elétrica da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Universidade Federal de Goiás - UFG  
Escola de Engenharia Elétrica, Mecânica e de Computação  
Trabalho de Conclusão de Curso

Orientador: Wesley Pacheco Calixto  
Coorientador: Wanderson Rainer Hilario de Araujo

Brasil  
2019

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Batista, Rogério Pereira

Cinemática Inversa Aplicada a Manipulador Robótico a Partir de Coordenadas Capturadas pelo Microsoft Kinect [manuscrito] / Rogério Pereira Batista. - 2019.

15 f.: il.

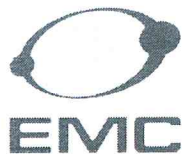
Orientador: Prof. Dr. Wesley Pacheco Calixto; co-orientador Wanderson Rainer Hilario Araujo.

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Goiás, Escola de Engenharia Elétrica, Mecânica e de Computação (EMC), Engenharia Elétrica, Goiânia, 2019.

Inclui gráfico, tabelas.

1. Braço robótico. 2. manipulador mecânico. 3. Microsoft Kinect. 4. cinemática inversa. I. Calixto, Wesley Pacheco, orient. II. Título.

CDU 621.3



### ATA DE AVALIAÇÃO DE PROJETO FINAL

CURSO		
(X) Eng. Elétrica	( ) Eng. Mecânica	( ) Eng. de Computação
		( ) Projeto Final 1
		( ) Projeto Final II

AVALIAÇÃO DE PROJETO FINAL	
Título do projeto:	Cinemática inversa aplicada a manipulador robótico a partir de coordenadas capturadas pelo Microsoft Kinect

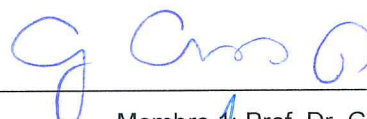
BANCA AVALIADORA	
Membro 1:	Gelson da Cruz Júnior
Membro 2:	Márcio Rodrigues da Cunha Reis
Membro 3:	Wanderson Rainer Hilário de Araújo

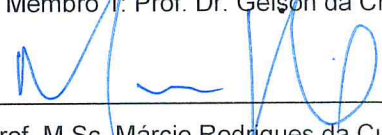
ESTUDANTES	
Matrícula	Nome
201408300	Rogério Pereira Batista

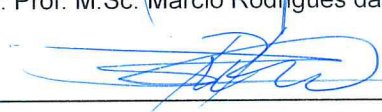
NOTAS													
Matrícula	Membro 1				Membro 2				Membro 3				Média
	NPT	NTE	NAA	NF	NPT	NTE	NAA	NF	NPT	NTE	NAA	NF	
201408300	9,0	9,0	9,0	9,0	9,0	9,0	9,5	9,22	9,0	8,0	9,5	8,77	8,99

NPT – Nota plano de trabalho; NTE – Nota do trabalho escrito; NAA – Nota de apresentação e arguição  
Para Eng. Elétrica, Mecânica e PFC2 da Eng. Da Computação:  $NF = 0,1 \times NPT + 0,45 \times NTE + 0,45 \times NAA$   
Para PFC1 da Eng. Da Computação:  $NF = 0,3 \times NPT + 0,7 \times NAA$

Goiânia, 11 de dezembro de 2019.

  
Membro 1: Prof. Dr. Gelson da Cruz Júnior

  
Membro 2: Prof. M.Sc. Márcio Rodrigues da Cunha Reis

  
Membro 3: Prof. M.Sc. Wanderson Rainer Hilário de Araújo

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

Rogério Pereira Batista\*, Wesley Pacheco Calixto<sup>†</sup> e Wanderson Rainer Hilario de Araujo<sup>§</sup>  
Graduando em Engenharia Elétrica\*, Professor Dr. pela Universidade Federal de Goiás<sup>†</sup>  
e Professor Me. pela Pontifícia Universidade Católica de Goiás<sup>§</sup>  
roger\_tecnic@hotmail.com\*, wpcalixto@gmail.com<sup>†</sup>, rainer@pucgoias.edu.br<sup>§</sup>

**Resumo**—Este artigo apresenta o desenvolvimento da cinemática inversa aplicada a um manipulador robótico empregando o Microsoft Kinect como dispositivo de entrada de coordenadas. O objetivo desta pesquisa é contribuir positivamente com processos produtivos industriais que requerem a manipulação de objetos por parte de operadores que são submetidos a exposições de riscos. Em tais ocasiões, o sistema desenvolvido pode minimizar os riscos de acidentes aos operadores e promover ambientes de trabalho mais seguros, uma vez que o operador pode operar remotamente um manipulador robótico sem ter contato direto com os riscos oferecidos. Neste artigo é apresentado o manipulador robótico empregado para testes, bem como a cinemática inversa desenvolvida para o mesmo. Resultados experimentais são discutidos bem como os obstáculos provenientes da integração entre o manipulador e o Microsoft Kinect.

**Palavras-chave**—Braço robótico, manipulador mecânico, Microsoft Kinect, cinemática inversa.

**Abstract**—This article presents the development of inverse kinematics applied to a robotic arm manipulator using Microsoft Kinect as a coordinate input device. The objective of this research is to contribute positively to industrial processes that use object manipulation by operators who are exposed to risks. At such times, the developed system can reduce the risk of accidents to operator, promoting safer working environments as the operator can remotely operate a robotic arm manipulator without direct contact with the hazards used. In this article is presented a robotic manipulator employed for testing, as well as an inverse kinematics developed for it. Experimental results are discussed as well as obstacles Handling integration between handler and Microsoft Kinect.

**Keywords**—Robotic arm, mechanical manipulator, Microsoft Kinect, reverse kinematics.

## I. INTRODUÇÃO

O aumento da competitividade mundial impulsionado pela globalização faz com que as organizações direcionem esforços com a implantação de sistemas de gestão específicos, tais como qualidade, meio ambiente, segurança e saúde do trabalho e responsabilidade social. Neste contexto, o bom desempenho em segurança e saúde no trabalho é decisivo para as empresas, uma vez que este sistema reduz os riscos de acidentes, promove a saúde e a satisfação dos trabalhadores, melhora os resultados operacionais e a imagem da organização, criando novas oportunidades de crescimento [1].

No Brasil, a legislação vigente garante direitos e deveres a trabalhadores e empreendedores com vistas a estabelecer ambientes com menores riscos de acidentes. Os principais

aspectos desta legislação são apresentados pelas normas regulamentadoras [12] a [16].

Na comunidade científica, estudos são desenvolvidos de forma a contribuir positivamente com ambientes de trabalho seguros. Em 1, é proposto um sistema de gestão da segurança e saúde no trabalho em empresas produtoras de baterias automotivas. Na ocasião, dois estudos de caso foram estabelecidos para se chegar em um conjunto de boas práticas para a gestão de sistemas de segurança e saúde do trabalho. Na área industrial, os riscos referentes à exposição em áreas energizadas é fortemente considerado. As normas e precauções são estabelecidas na norma regulamentadora NR10 [12]. Em 2, são feitas considerações sobre a instalação, operação e manutenção de equipamentos elétricos em locais perigosos. Neste caso, é apresentado que as rotinas de manutenção de equipamentos elétricos devem ser devidamente alteradas quando são executadas em locais perigosos. No Brasil é recomendado que as áreas de uma instalação sejam classificadas pelo grau de risco, o que gera um documento denominado mapa de risco. Em outros países processos semelhantes são implementados, pois verifica-se em 3 um estudo de caso utilizando desenhos de classificação de área de risco elétrico como base para avaliar o grau de extensão de riscos de explosão em instalações industriais e assim auxiliar em operações mais seguras. Por outro lado, 4 apresenta condições para manipulação de materiais químicos perigosos. Neste caso, um sistema com vários níveis de sensoriamento promove maior segurança referente a incêndios.

Logo, percebe-se que globalmente há preocupações quanto aos riscos que os operadores são expostos, especialmente em ambientes industriais. Em muitos processos produtivos, seja na indústria de processos contínuos ou na automação da manufatura, os operadores das plantas industriais normalmente são expostos a riscos [13] [14] que podem comprometer negativamente a saúde e a segurança no trabalho. Tais riscos são extrapolados para outros setores industriais como o da construção civil e da manutenção. Neste contexto, a robótica apresenta papel de destaque na contribuição para minimização de riscos

No cenário nacional, 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* [5].

Não só nestes mas como em outros ambientes industriais, os manipuladores robóticos estão associados a atividades de embalagem, paletização, coleta e produção. Na indústria automotiva, os manipuladores robóticos contam com ferramentas específicas em seus efetuadores para atividades de soldagem e pintura. Porém os manipuladores robóticos possuem uma programação de movimentos no espaço bem definida, tornando o espaço de trabalho onde operam um lugar com alto risco de colisão com algum objeto ou ser humano que adentrar em sua área durante a sua operação, sendo assim estes manipuladores possuem seu espaço de trabalho confinado, onde a entrada de objetos e pessoas é controlada.

Desta forma, considerando: i) a crescente participação dos robôs nos processos produtivos industriais, como em [5], ii) a versatilidade de tais manipuladores proporcionada pelo desenvolvimento tecnológico e iii) a flexibilidade oferecida por estes manipuladores, o objetivo deste trabalho é a utilização de manipulador robótico para a realização de tarefas consideradas perigosas para seres humanos.

Assim este estudo parte da hipótese de que o manipulador robótico pode ser operado remotamente pelo operador para realizar atividades que antes necessitariam da intervenção direta do operador, evitando assim expô-lo a um risco, contribuindo para a sua segurança, minimizando os riscos nos ambientes industriais.

## II. FUNDAMENTAÇÃO TEÓRICA

Dispositivos robóticos industriais podem ser classificados de diversas formas. Máquinas de comando numérico computadorizado (CNC) são classificadas como dispositivos robóticos por alguns autores. Neste trabalho, os aspectos funcionais são concentrados na mecânica e no controle do tipo mais comum de robô industrial, que é o manipulador mecânico [6].

### A. Normas Regulamentadoras Nacionais

As Normas Regulamentadoras Nacionais para segurança em locais de trabalho são:

- Segurança em Instalações e Serviços em Eletricidade - NR10 [12]:
  - 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 [13]:
  - 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 [14]:
  - Esta norma estabelece que são consideradas atividades perigosas aquelas que estão presentes nos anexos da norma.
- Atividades em Espaços Confinados - NR33 [15]:

- Esta norma tem como objetivo estabelecer requisitos mínimos para identificação de espaços confinados, para monitorar e controlar os riscos existentes.
- Atividades que Envolvem Trabalho em Altura - NR35 [16]:
  - Esta norma estabelece requisitos mínimos e medidas de proteção para trabalhos em altura.

### B. Manipulador Robótico Articulado

No estudo da robótica, há o interesse na localização de objetos no espaço tridimensional. Esses objetos são os elos do manipulador, as peças e ferramentas com as quais lida e outros objetos em seu ambiente. Os manipuladores consistem em elos quase 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. Existem manipuladores onde as juntas são conectadas e mais de 2 elos, são conhecidos como manipuladores paralelos e sua característica é a velocidade, porém o espaço de trabalho é limitado. Neste trabalho optou-se pela utilização de um manipulador robótico série, onde cada junta conecta apenas 2 elos, pois seu espaço de trabalho é maior comparado aos manipuladores paralelos. Os conceitos serão aplicados a um manipulador como o ilustrado na Fig. 1.

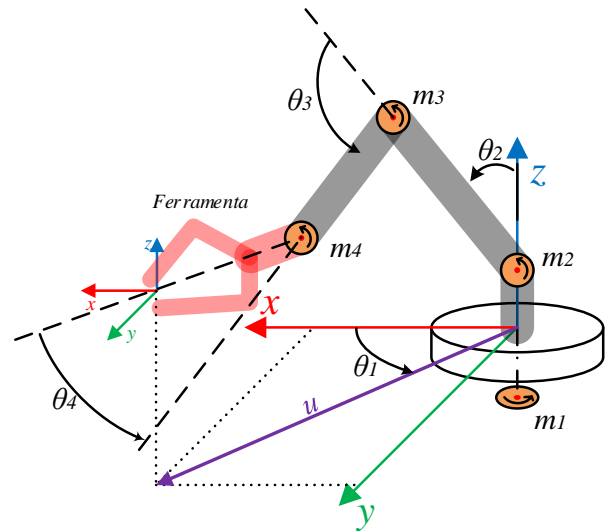


Figura 1. Ilustração do manipulador mecânico.

Onde:

- $m_1$  à  $m_4$ : denominação da localização dos servomotores 1 ao 4 nas juntas 1 à 4;

- $\theta_1$  à  $\theta_4$ : posição angular das juntas 1 à 4;
- $u$ : vetor composto pelas coordenadas  $x$  e  $y$  do manipulador robótico.

Na Fig. 1, o manipulador mecânico possui 4 elos e 4 juntas. Ao último elo normalmente é conectado um dispositivo denominado efetuador, ilustrado como ferramenta na Fig. 1. O efetuador é o dispositivo empregado pelo robô para executar determinada tarefa, podendo ser um eletrodo para solda, uma garra, um jato de pintura, entre outros.

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 Fig. 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, de um sistema para outro.

A técnica que trata do movimento sem considerar as forças que o causam é denominada cinemática. Na ciência da cinemática estudam-se 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, onde 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*  $i+1$  e os demais *frames* subsequente se movimentam, 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 [7].

O espaço de trabalho do manipulador é o volume total percorrido pelo efetuador (órgão terminal ou ferramenta) dadas todas as possibilidades de execução de movimento do robô. O espaço de trabalho é limitado pela geometria do manipulador, bem como pelas restrições físicas das juntas (limites mecânicos), como é mostrado na Fig. 2.

A existência ou a inexistência de uma solução cinemática define o espaço de trabalho dado manipulador. A ausência de uma solução significa que o manipulador não poderá chegar à posição e à orientação desejadas porque está fora do espaço de trabalho do manipulador [6].

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

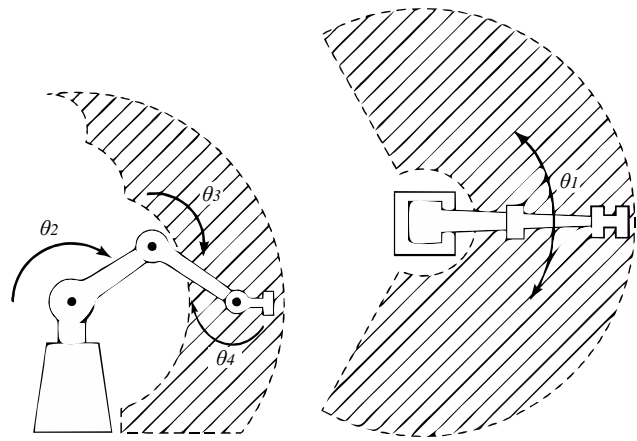


Figura 2. Ilustração do espaço de trabalho do manipulador mecânico.

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 Fig. 1, a cinemática direta é a obtenção das coordenadas  $x$ ,  $y$  e  $z$  do efetuador a partir dos valores dos ângulos  $\theta_1$ ,  $\theta_2$ ,  $\theta_3$  e  $\theta_4$ , dos servomotores  $m_1$ ,  $m_2$ ,  $m_3$  e  $m_4$  respectivamente [7].

Por outro lado, a cinemática inversa tem como objetivo realizar o cálculo inverso: a partir das coordenadas  $x$ ,  $y$  e  $z$  do efetuador, obter os valores dos ângulos das juntas ( $\theta_1$ ,  $\theta_2$ ,  $\theta_3$  e  $\theta_4$ ) que levem o efetuador para a posição desejada [9]. O que torna a cinemática inversa complexa é o fato de sua solução não ser unívoca, ou seja, podem existir várias soluções para um mesmo ponto no espaço, ou ainda, vários conjuntos de ângulos que levam o efetuador para a mesma posição e orientação no espaço. Outro problema encontrado na modelagem da cinemática inversa é a existência de posições denominadas *gimbal lock* [8], estas posições acontecem quando duas ou mais juntas de um sistema mecânico se alinham de forma que seus eixos se sobrepõem, assim o acionamento de qualquer uma destas juntas causam a mesma movimentação da estrutura, ou seja, o sistema mecânico perde um ou mais graus de liberdade. Este cálculo é igualmente comum para manipuladores industriais e é objeto de estudo deste trabalho [7].

Para o cálculo da cinemática inversa existem métodos, onde um algoritmo de forma recursiva calcula os ângulos e métodos algébricos [6], o método escolhido foi o algébrico, utilizando relações trigonométricas para os cálculos.

### III. METODOLOGIA

O Microsoft Kinect apresenta-se como uma alternativa adequada à captura das coordenadas das mãos do operador do sistema, pois a imagem é processada no próprio aparelho e a sua integração com a plataforma de programação permite que a posição dos pontos desejados da figura sejam diretamente coletados, diferente do uso de câmeras normais, onde o processamento e tratamento da imagem tem que ser feita totalmente dentro do software. Assim as variáveis de entrada

do algoritmo de cinemática inversa, ou seja, os pontos no espaço, serão fornecidos pelo Microsoft Kinect, Fig. 3.



Figura 3. Microsoft Kinect 2.0.

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 efetuator. Contudo, as coordenadas  $x$ ,  $y$  e  $z$  do efetuator devem ser obtidas via captura de imagens empregando o Microsoft Kinect. A Fig. 4 ilustra os elementos integrantes do sistema a ser desenvolvido.

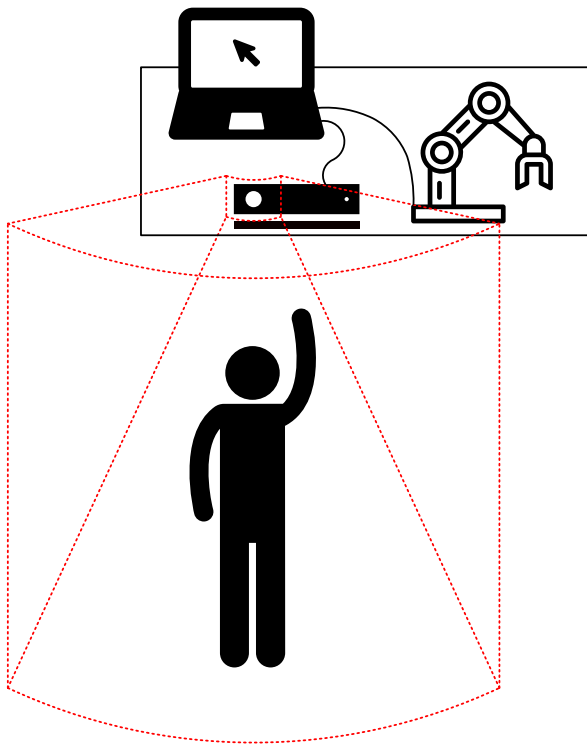


Figura 4. Ilustração do sistema desenvolvido.

Na Fig. 4 estão presentes o manipulador robótico, um computador e o Microsoft Kinect, responsável por efetuar a leitura das posições dos membros do operador, conforme ilustra a própria figura. Percebe-se então que neste sistema

está envolvida a cinemática inversa, pois o objetivo é fazer com que o operador da Fig. 4 seja capaz de movimentar o manipulador robótico remotamente. Logo, as coordenadas das mãos do operador, devidamente capturadas pelo Microsoft Kinect, devem ser recebidas pelo computador que, por sua vez, faz o cálculo da cinemática inversa para obter os ângulos das juntas do manipulador para o devido posicionamento do efetuator.

A Fig. 5 ilustra o diagrama de blocos geral do sistema.

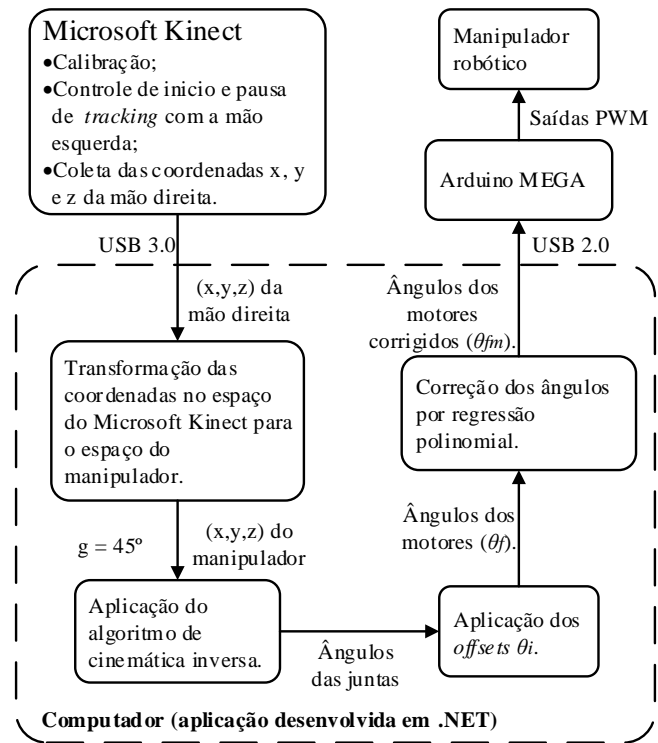


Figura 5. Diagrama de blocos ilustrando uma visão geral do fluxo de dados.

Conforme pode ser observado na Fig. 5, inicialmente é realizada a calibração do Microsoft Kinect. Este processo é necessário para que a captura das coordenadas do operador do sistema possa ser realizada independente do ambiente no qual o dispositivo será instalado, bem como possibilitar a operação por pessoas de diferentes tamanhos e em diferentes posicionamentos em relação aparelho. Desta forma, são definidos os limites inferior e superior de cada um dos eixos coordenados, ou seja, nesta etapa é definido o espaço de trabalho do operador. Por fim, as coordenadas  $x$ ,  $y$  e  $z$  são capturadas pelo Microsoft Kinect e transferidas para uma aplicação desenvolvida na plataforma .NET via porta USB 3.0 (requisito de hardware do Microsoft Kinect), a coordenada  $g$  não é capturada pelo Microsoft Kinect, porém pode-se utilizar um sensor que o operador controle para a obtenção desta inclinação de maneira a controlá-lo melhor ou apenas fixá-

lo em um valor. De forma a estabelecer um protocolo de comunicação entre o operador e a captura de coordenadas, é estabelecida uma etapa de sincronismo, na qual o operador indica ao sistema, via mão esquerda, o início e o fim da captura de coordenadas definidas pela mão direita.

Na aplicação desenvolvida, é realizado um processo de normalização entre o espaço de trabalho definido pelo operador (descrito anteriormente) e o espaço de trabalho do manipulador robótico. Este processo é necessário devido ao fato de que o espaço de trabalho do manipulador robótico, ilustrado na Fig. 2, é menor do que o espaço de trabalho definido pelo operador durante a etapa de sincronismo. Este espaço de trabalho do manipulador é menor devida a suas dimensões, que serão apresentadas posteriormente. Devido ao fato dos eixos coordenados capturados pelo Microsoft Kinect serem diferentes dos eixos do manipulador, é necessária uma transformação do sistema de coordenadas. O manipulador possui um sistema de coordenadas como o ilustrado na Fig. 1, porém, o Microsoft Kinect possui sistema de coordenadas rotacionado em relação ao do manipulador, na Fig. 6 pode ser observado ao lado esquerdo o sistema de coordenadas do manipulador em comparação com o sistema de coordenadas do Microsoft Kinect a direita. Para a obtenção de um ponto no espaço referenciado no sistema de coordenadas do manipulador é necessário girar  $-90^\circ$  no eixo  $y$ , depois girar  $90^\circ$  no eixo  $x$ , transformando assim o referencial de um ponto do sistema do Microsoft Kinect no sistema do manipulador. como as rotações são múltiplos de  $90^\circ$  nos eixos a rotação é simplificada apenas fazendo as operações descritas na tabela I.

Tabela I  
ROTAÇÃO DO SISTEMA DE COORDENADAS DO MICROSOFT KINECT PARA O SISTEMA DE COORDENADAS DO MANIPULADOR.

Microsoft Kinect	Manipulador
$x$	$-y$
$y$	$z$
$z$	$-x$

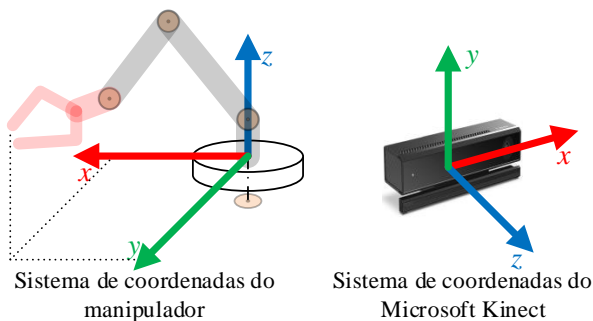


Figura 6. Comparação entre os sistemas de coordenadas do manipulador e do Microsoft Kinect.

Além da rotação do sistema de coordenadas também é necessário fazer uma mudança de base nas dimensões dos

eixos para que a transformação seja completa, pois uma certa distância movimentada pela mão do operador é relacionada com a distância que o manipulador irá movimentar através da relação entre as dimensões do espaço de trabalho do operador definidas na calibração e as dimensões do espaço de trabalho do manipulador. As dimensões do espaço de trabalho do manipulador são definidas utilizando-se os limites máximo e mínimo alcançado pelo manipulador em cada eixo coordenado manipulador, similar ao que é feito durante a calibração para os limites do operador, porém estes limites são fixos e apresentados na Tab. II.

Tabela II  
LIMITES SUPERIOR E INFERIOR EM MILÍMETROS DOS EIXOS COORDENADOS DO MANIPULADOR.

	Valor mínimo	Valor máximo
$x$	160	290
$y$	100	100
$z$	0	250

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 3 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.

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

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).

A equação 1 foi desenvolvida utilizando um método de transformação de escalas, ilustrado na Fig. 7, pois a escala dos eixos coordenados do espaço de trabalho do operador, cujos limites são adquiridos no processo de calibragem, é diferente da escala dos eixos coordenados do espaço de trabalho do manipulador, que são valores fixos determinados pela geometria do manipulador, representados na Tab. II.

Sendo assim, de posse das coordenadas já normalizadas para o espaço de trabalho do manipulador o algoritmo de cinemática inversa 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  $\theta_4$  responsáveis por fazer o efetuator do manipulador atingir o ponto desejado. Se o ponto

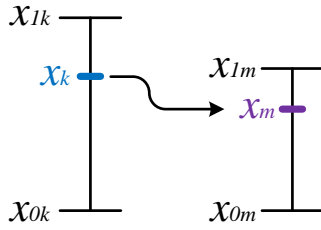


Figura 7. Ilustração da transformação entre escalas do eixo  $x$  do operador para o eixo  $x$  do manipulador.

desejado não estiver no espaço de trabalho a cinemática inversa retorna todos os 4 ângulos iguais a  $0^\circ$ , que correspondem a posição neutra, ou de referência.

Logo após, aos ângulos das juntas são adicionados os *offsets* denominados como  $\theta_i$  para a obtenção dos ângulos dos servomotores  $\theta_f$ . Em seguida os ângulos são passados para os servomotores pelo Arduino MEGA. A importância do Arduino MEGA se dá no fato de possuir uma interface de programação amigável, fácil utilização das suas funções e ser uma placa de prototipagem embarcada de baixo custo.

Para a programação do Arduino Mega foi utilizada a IDE *Integrated Development Environment* do próprio Arduino, juntamente com a funcionalidade de comunicação serial e uma biblioteca de controle de servomotores externa, denominada *VarSpeedServo.h*, que permite além do controle de posição dos servos o ajuste da velocidade dos mesmos. O código do Arduino Mega foi desenvolvido de maneira a minimizar as tarefas a serem desempenhadas pelo Arduino Mega, devido a sua velocidade limitada comparada ao computador, onde é executado o software de controle desenvolvido em .NET. Portanto o o Arduino Mega aguarda o recebimento de uma mensagem através da comunicação serial, esta mensagem é uma *string* que contém os valores dos ângulos a serem passados para cada servomotor seguido de um caractere de identificação para qual servomotor o valor anterior se refere, ao todo são 5 valores numéricos de 1 a 3 caracteres cada e 5 caracteres de identificação. Através dos caracteres de identificação os ângulos são devidamente lidos da *string* e então escritos através de uma porta digital com sinal PWM *Pulse Width Modulation*. Na Fig. 8 pode ser observado o funcionamento do código gravado no Arduino Mega.

Foram feitos testes de precisão e posicionamento. todos os servomotores foram acionados de forma que o robô se posicionasse verticalmente em linha reta, conforme Fig. 9.

Estando nesta posição, foi observado que para cada servomotor há um *offset* inicial de ângulo ( $\theta_i$ ) que deve ser levado em consideração quando o robô é manipulado. Além disto, dadas as características rudimentares do manipulador como estrutura em plástico e mdf e servomotores sem *feedback* de posição, foi constatado que durante o acionamento de qualquer um dos servomotores para uma posição final  $\theta_f$ , sempre ocorria um desvio, resultando em um ângulo  $\theta_f + \Delta\theta$ , esse desvio foi observado utilizando-se transferidor e esquadros.

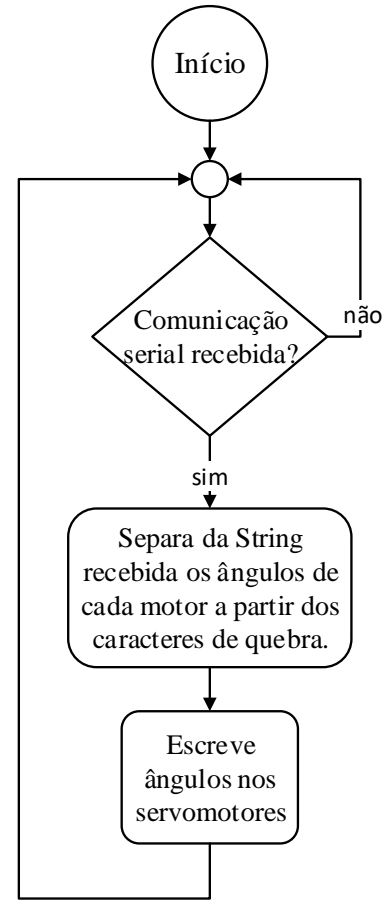


Figura 8. Fluxograma do funcionamento do código do Arduino Mega.

Isto impacta no valor a ser passado para os servomotores das juntas. Ou seja, para que um determinado servomotor alcance uma posição final  $\theta_f$ , é necessário efetuar uma escrita na posição do servomotor ( $\theta_{fm}$ ) um valor devidamente corrigido, de tal forma que:

$$\theta_{fm}(\theta_f) \quad (2)$$

onde  $\theta_f$  é a posição final desejada para o servomotor e  $\theta_{fm}$  é o valor a ser escrito no servomotor para alcançar a posição  $\theta_f$  desejada, é importante observar que esta variação  $\Delta\theta$  depende do ângulo desejado  $\theta_f$ , logo o ângulo  $\theta_{fm}$  é uma função do ângulo  $\theta_f$  não linear. Foi utilizado o método da interpolação polinomial como será mostrado mais adiante, na equação 17 pode ser observado a função para  $\theta_{fm2}(\theta_{f2})$ . Além disto:

$$\theta_f = \theta_c + \theta_i \quad (3)$$

onde  $\theta_c$  é o valor obtido na cinemática e  $\theta_i$  é o *offset* inicial obtido quando o manipulador é posicionado verticalmente, conforme a Fig. 9.

comportamento do software.

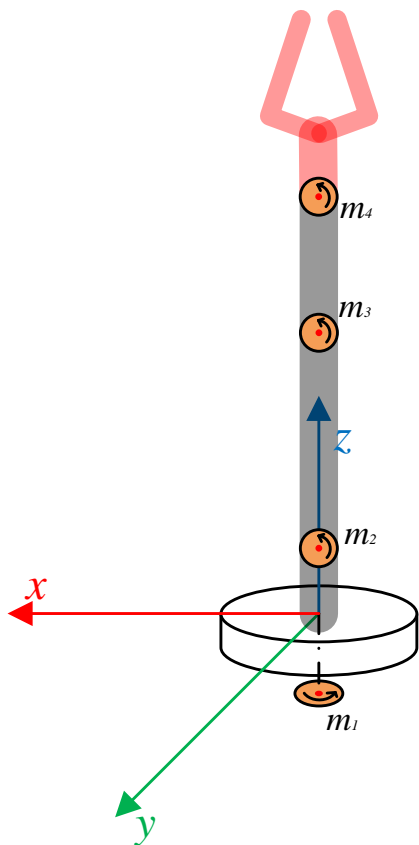


Figura 9. Manipulador na posição de referência.

### A. Desenvolvimento da aplicação .NET

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 Mega. Na Fig. 10 pode ser visto o fluxograma do código desenvolvido. O SDK do Microsoft Kinect é utilizado através de um sistema de diversas funções que são executadas quando certos gatilhos são disparados, o gatilho utilizado aqui é a detecção de um corpo por parte do Microsoft Kinect em frente a sua câmera. O Kinect ao detectar que um corpo entrou em seu campo de visão executa indefinidamente uma função que contém toda a lógica de controle do programa. Quando não há mais nenhum corpo no campo de visão do Microsoft Kinect a execução da função é interrompida, até que outro corpo seja detectado. Além disso a função é executada para cada corpo, então se houver dois ou mais corpos diante do Microsoft Kinect, ocorrem comportamentos imprevisíveis do

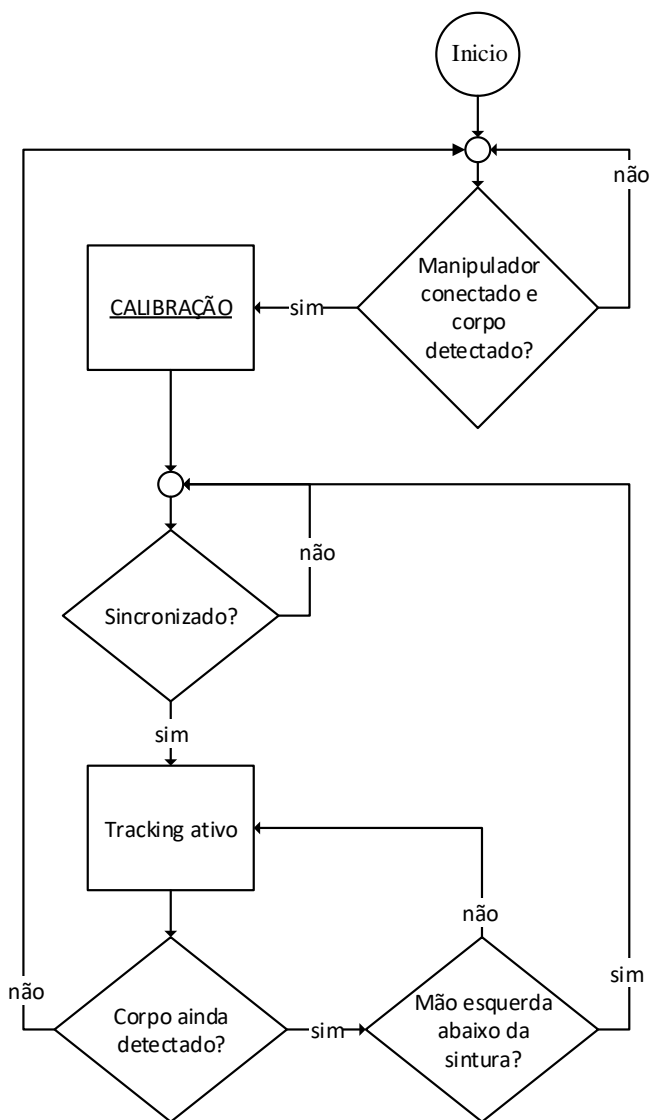


Figura 10. Fluxograma do funcionamento do código da aplicação .NET.

Como observado na Fig. 10, o programa aguarda um corpo ser detectado pelo Microsoft Kinect, ao ser detectado um corpo (o operador) o gatilho dispara as outras funções. Durante as diversas etapas da lógica o software indica ao operador instruções de como proceder. Primeiramente a etapa de calibração é executada. A calibração é feita apenas na primeira vez que o corpo do operador é detectado até que o operador se ausente do campo de visão do Kinect, só nesta situação a calibração é executada novamente. Durante o processo de calibração o operador deve seguir os passos indicados na interface gráfica, lembrando que a mão esquerda é utilizada para controlar a calibração e o sincronismo e a mão direita é utilizada para a coleta do ponto no espaço que será passado

para o manipulador. O operador deve sempre ter em mente que todos os gestos em frente ao Microsoft Kinect devem ser feitos de forma a sempre manter as palmas das mãos de frente para a câmera do Microsoft Kinect e deve-se evitar a sobreposição das mãos entre si e no corpo, pois quando isto ocorre acontece o sombreamento de pontos utilizados pelo Microsoft Kinect para a *tracking*, isto gera instabilidade nas leituras.

Para a calibração o operador deve posicionar sua mão esquerda aberta acima da cintura, então posicionar a sua mão direita nos limites dos eixos coordenados do seu espaço de trabalho, ou seja, onde a sua mão direita alcança, então com a mão direita posicionada, a mão esquerda deve ser fechada por um breve instante para que o ponto da mão direita seja coletado, este procedimento é realizado para um total de 6 pontos, sendo os limites máximo e mínimo nos eixos  $x$ ,  $y$  e  $z$  do seu espaço de trabalho, é importante notar que, devido a rotação dos eixos coordenados do Microsoft Kinect para os eixos coordenados do manipulador, quando o operador se encontra em frente ao Microsoft Kinect na posição de operação, como na Fig. 4, o eixo  $x$  é tal que seus valores crescem de traz do operador para frente do operador, o eixo  $y$  é tal que seus valores crescem da direita do operador para a esquerda do operador e o eixo  $z$  é tal que seus valores crescem de baixo para cima, estes eixos seguem a mesma orientação dos eixos indicados na Fig. 1. Após a coleta dos 6 pontos de calibração o operador é solicitado a abaixar sua mão esquerda para se posicionar para a etapa de sincronismo de tracking. A Fig. 11 ilustra a etapa de calibração.

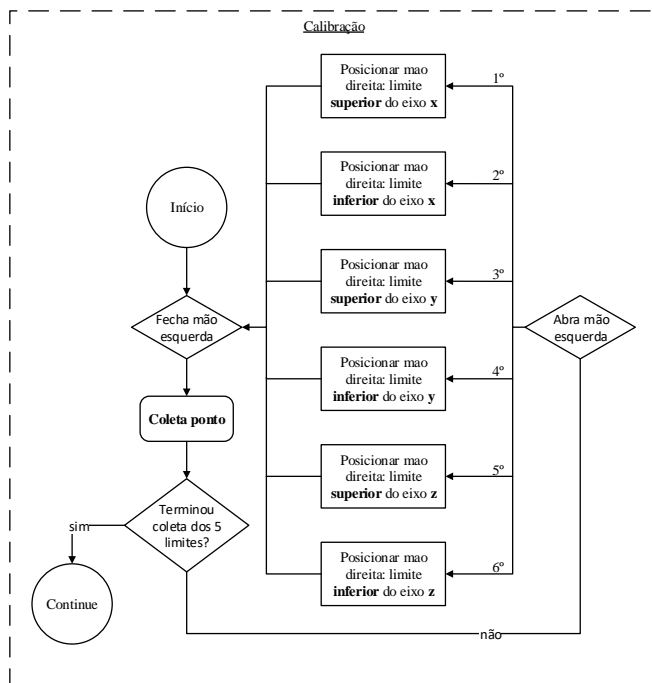


Figura 11. Fluxograma resumido da etapa de calibração.

Após a etapa de calibração o operador entra na etapa de sin-

cronismo, para iniciar o *tracking* da mão direita e controle do movimento do manipulador. A etapa de sincronismo consiste em posicionar a mão direita no centro do espaço de trabalho que foi previamente definido os limites durante a calibração, então levantar a mão esquerda acima da cintura aberta, nesta posição o operador ao fechar a mão esquerda sincroniza a movimentação e todo o movimento feito com a mão direita é transferido para o manipulador, que imita o movimento, neste momento o efetuador em forma de garra é operado (aberto ou fechado) através do fechamento da mão direita, se o operador fechar a mão direita a garra fecha, se abrir a garra abre.

Se o operador abaixar a mão esquerda o *tracking* do movimento é pausado, o manipulador para na posição atual e o operador poderá movimentar a mão direita sem a movimentação equivalente do manipulador. Neste momento o software retorna para a etapa de sincronismo, bastando o operador posicionar sua mão direita no centro do espaço de trabalho e a mão esquerda acima da cintura e a fechar para retornar o *tracking*. Se o operador sair do campo de visão do Microsoft Kinect o código retorna ao estado inicial, excluindo toda a informação de calibração e aguarda novamente um corpo ser detectado.

## B. Cinemática inversa

A cinemática inversa foi desenvolvida utilizando conceitos de geometria analítica e álgebra linear. Não foi necessário o tratamento de possíveis pontos de *gimbal lock* visto que o corpo do manipulador não torna possível a sobreposição de eixos de juntas. O manipulador possuindo 4 graus de liberdade necessita de no mínimo 4 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 Fig. 12. Para que a cinemática inversa possua apenas uma solução foi considerada a premissa de que o ângulo da junta 3 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 ver Fig. 12.
- Variáveis de saída:
  - $\theta_1$ : ângulo da junta 1.
  - $\theta_2$ : ângulo da junta 2.
  - $\theta_3$ : ângulo da junta 3.
  - $\theta_4$ : 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  $\theta_1$  pode ser observado na Fig. 12, é dado pela equação 4:

$$\theta_1 = \arctan\left(\frac{y}{x}\right) \quad (4)$$

Para os demais cálculos o manipulador é considerado planar e toda a sua estrutura no mesmo plano conforme Fig. 12, logo a posição do manipulador pode ser descrita apenas por 2 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 onde 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 5, pela coordenada  $z$  e pela inclinação da garra  $g$ .

$$u = \sqrt{x^2 + y^2} \quad (5)$$

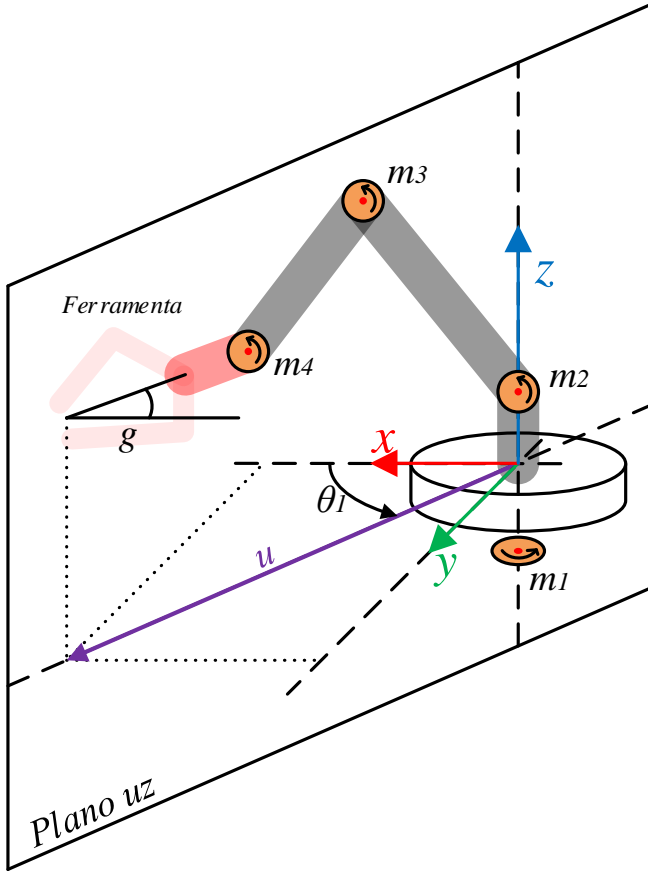


Figura 12. Detalhes de  $\theta_1$ ,  $g$  e plano  $uz$ .

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  $\vec{P}_4$  conforme as equações 6 e 7 e ilustrados na Fig. 13.

$$\vec{P}_2 = (0, 46) \quad (6)$$

$$\vec{P}_4 = (u - L_4 \cdot \cos(g), z + L_4 \cdot \sin(g)) \quad (7)$$

Os vetores  $\vec{P}_2$  e  $\vec{P}_4$  são utilizados para calcular o vetor  $\vec{h}$  conforme equação 8.

$$\vec{h} = \vec{P}_2 - \vec{P}_4 \quad (8)$$

O vetor  $\vec{h}$  juntamente com os elos  $L_2$  e  $L_3$  formam um triângulo observado na Fig. 13 com cor roxa, os ângulos internos deste triângulo são denominados  $\theta_{hL_2}$ ,  $\theta_{L_2L_3}$  e  $\theta_{hL_3}$  como pode ser visto na Fig. 13. Estes ângulos são calculados utilizando a lei dos cossenos, conforme as equações 9, 10 e 11.

$$\theta_{hL_2} = \arccos\left(\frac{\|\vec{h}\|^2 + L_2^2 - L_3^2}{2 \cdot \|\vec{h}\| \cdot L_2}\right) \quad (9)$$

$$\theta_{L_2L_3} = \arccos\left(\frac{L_2^2 + L_3^2 - \|\vec{h}\|^2}{2 \cdot L_2 \cdot L_3}\right) \quad (10)$$

$$\theta_{hL_3} = \arccos\left(\frac{\|\vec{h}\|^2 + L_3^2 - L_2^2}{2 \cdot \|\vec{h}\| \cdot L_3}\right) \quad (11)$$

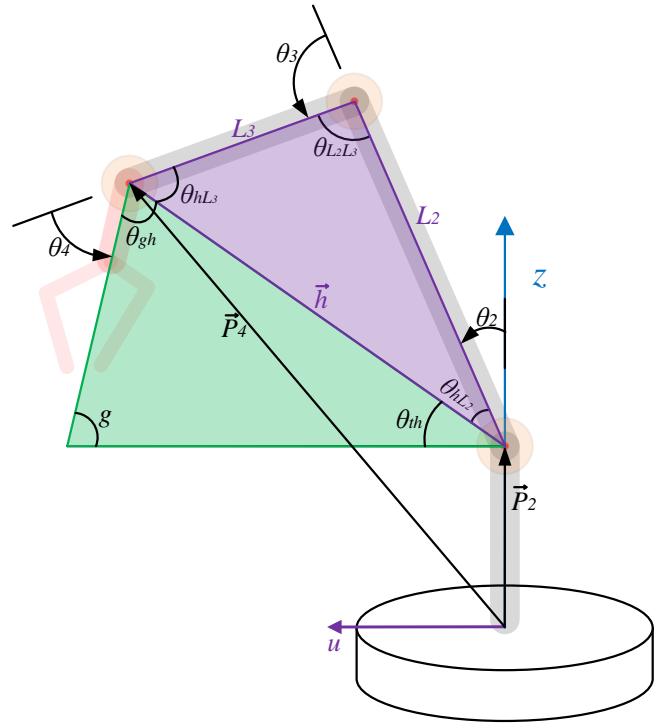


Figura 13. Vetores  $\vec{P}_2$ ,  $\vec{P}_4$  e triângulo formado pelas juntas 2, 3 e 4.

Também é calculado o ângulo  $\theta_{th}$  que é o ângulo entre o vetor  $\vec{h}$  e o plano da base através da equação 12, utilizando a coordenada  $u$  do vetor  $\vec{h}$   $h_u$ .

$$\theta_{th} = \arccos\left(\frac{h_u}{\|\vec{h}\|}\right) \quad (12)$$

É importante no calculo 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  $\theta_2$  e  $\theta_3$  são calculados utilizando as equações 13 e 14.

$$\theta_2 = 90^\circ - (\theta_{th} + \theta_{hL_2}) \quad (13)$$

$$\theta_3 = 180^\circ - \theta_{L_2L_3} \quad (14)$$

Na Fig. 13 desenhado em verde está um triângulo formado pelo vetor  $\vec{h}$  e que dois de seus três ângulos internos são conhecidos, sendo  $g$  e  $\theta_{th}$ . O terceiro ângulo denominado  $\theta_{gh}$  é facilmente obtido, sabendo-se que a soma dos ângulos internos de um triângulo é igual a  $180^\circ$ , conforme equação 15. Logo o ângulo  $\theta_4$  pode ser obtido através da equação 16.

$$\theta_{gh} = 180^\circ - \theta_{th} - g \quad (15)$$

$$\theta_4 = 180^\circ - \theta_{gh} - \theta_{hL_3} = \theta_{th} + g - \theta_{hL_3} \quad (16)$$

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

#### IV. RESULTADOS

Nesta seção serão apresentados os resultados obtidos nos procedimentos descritos na seção anterior. O objetivo desta seção é apresentar o que foi possível ser obtido com a integração do Microsoft Kinect e o manipulador robótico quando submetido à cinemática inversa.

O Microsoft Kinect é um sensor de movimento que integra uma câmera de resolução  $640 \times 480$  pixels, um sensor de profundidade infravermelho com uma câmera de  $320 \times 240$  pixels e um microfone, além de processamento de imagem e profundidade embarcados. Possui um *frame rate* de 30 Hz, uma distância prática de detecção de 1,2 a 3,5 metros e aberturas de 57 graus na horizontal e 43 graus na vertical. Para a utilização do sensor em aplicações de desenvolvimento a Microsoft disponibiliza um SDK *Software Development Kit* que possui versões para diversos sistemas operacionais e suporte para diversas linguagens de programação. Para a conexão do Microsoft Kinect com o computador é necessário suporte a USB 3.0.

O sensor utilizado para o desenvolvimento deste trabalho foi o Microsoft Kinect 2.0, ilustrado na Fig. 3, inicialmente desenvolvido para aplicações de entretenimento, foi ganhando espaço no desenvolvimento de aplicações que envolvem mapeamento 3D e rastreamento de objetos, por se tratar de um sensor que possibilita aplicações em diversas áreas da ciência e da tecnologia. Este dispositivo, além de ser integrado por uma câmera, como descrito anteriormente, permite a obtenção das coordenadas das juntas do corpo humano, como cabeça, tronco, quadril e mãos [10].

Conforme exposto por 11, o uso do Microsoft Kinect para movimentar um dispositivo à distância oferece mais segurança para as pessoas no exercício de atividades perigosas. Ele reconhece o corpo e os movimentos do operador oferecendo ao manipulador robótico precisão humana em tempo real.

#### A. Ajustes no Manipulador Robótico

O manipulador empregado no desenvolvimento do sistema é um braço robótico com 5 juntas e de característica DIY (*do it yourself*) com finalidade de aplicações hobbystas, apresentando características rudimentares se comparado com manipuladores industriais. Os servomotores utilizados na montagem foram os utilizados em aeromodelismo, cujo alcance de movimentação é limitado ( $0^\circ$  a  $180^\circ$ ). Controlado por sinal pwm, no *datasheet* sua resolução nominal é de aproximadamente  $1^\circ$ , porém esta resolução observada na prática varia em função da posição do servo. A Fig. 14 ilustra o diagrama do manipulador empregado no projeto destacando as juntas e os elos da estrutura.

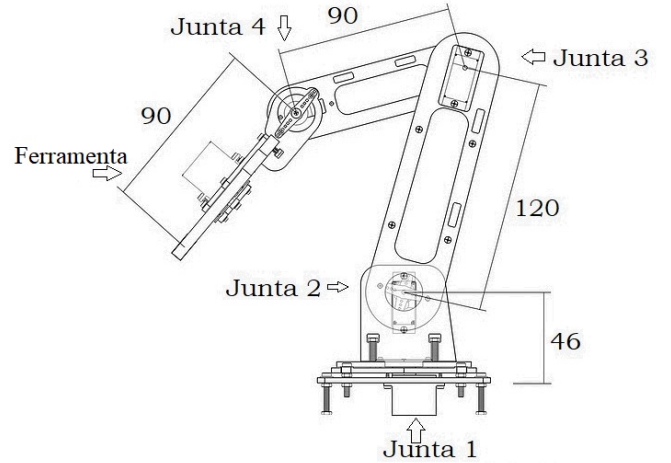


Figura 14. Diagrama do manipulador empregado no projeto (dimensões em mm).

Na Fig. 14, as dimensões dos elos estão apresentadas em milímetros. Desta forma, pode-se observar que o manipulador robótico empregado neste estudo possui 5 juntas e 4 elos, sendo a junta 5 responsável pela operação do efetuator e não influenciando no posicionamento. Cada uma das juntas é composta por um servomotor de modelo 9g SG90 da

TowerPro (Fig. 15), assim cada servomotor foi montado em cada junta de forma que todo o alcance de movimento entre os elos conectados coincidissem com o alcance de movimento do servomotor.



Figura 15. Servomotor TowerPro 9g sg90 utilizado inicialmente.

Contudo, foram necessários alguns ajustes para o correto funcionamento do robô. Inicialmente foram realizados testes de mobilidade e alcance do manipulador mecânico, os servomotores 9g SG90 (Fig. 15) apresentaram funcionamento dentro do esperado, porém o servomotor referente ao atuador do tipo garra necessitou ser substituído, sendo possível então pleno funcionamento. Testes de controle e estabilidade foram realizados, onde foi constatado que o manipulador tinha dificuldades na movimentação quando submetido a alguma carga mecânica. Com tal resultado, houve a necessidade de substituição dos servomotores das juntas de número 2 e 3. Na junta 3 foi instalado um servomotor de modelo DGS-S05NF (Fig. 16), já na junta 2 foi instalado um servomotor de modelo SG-5010 da TowerPro (Fig. 17). Os servomotores foram selecionados para as respectivas juntas de acordo com suas dimensões, pesos e capacidade de conjugado, visto que a junta 2 necessita de um maior conjugado que a junta 3.



Figura 16. Novo servomotor instalado na junta 3. Modelo DGS-S05NF

Para a montagem dos servomotores nas juntas do manipulador foi necessária a adequação da estrutura do manipulador robótico, os elos conectados a junta 2 não atendiam às dimensões dos novos servomotores, logo novas peças foram modeladas em software CAD (*computer aided design*). Na Fig. 18 pode ser observado na parte inferior a peça original e



Figura 17. Novo servomotor instalado na junta 2. Modelo SG-5010 da TowerPro.

acima a nova peça impressa em impressora 3D, foi alterado o encaixe do servomotor. Já na Fig. 19 pode ser vista a peça do elo 1 impressa em 3D para o encaixe do novo servomotor da junta 2.



Figura 18. Peça do elo 2 que conecta as juntas 2 e 3, abaixo a peça original, acima a nova peça impressa em impressora 3D.

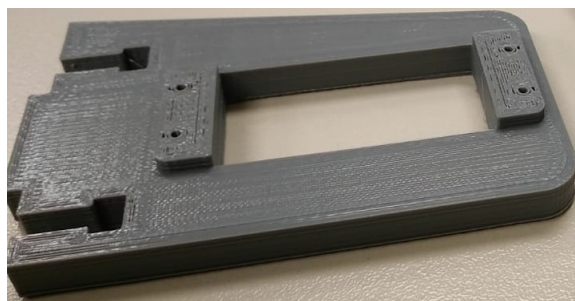


Figura 19. Peça do elo 1 que conecta a junta 1 na base a junta 2, impressa em impressora 3D.

Para o acionamento elétrico dos servomotores foi utilizada uma placa Arduino Mega, para as conexões elétricas optou-se por utilizar uma *proto-board* facilitando assim substituições rápidas nas ligações elétricas, bem como a ligação da alimentação dos vários servomotores em paralelo. Para a alimentação foi utilizada uma fonte chaveada de corrente contínua de 12

volts e 5 amperes e um regulador de tensão DC-DC LM2560 para alimentação dos servomotores, que são todos alimentados em 7 volts, tensão máxima de acordo com os *datasheets*. Todo o circuito elétrico foi fixado na parte interna de uma caixa feita em mdf, sobre a qual o manipulador robótico foi fixado, servindo como base. Na Fig. 20, vista inferior da caixa na qual é a base do manipulador robótico, pode ser observado todo o circuito responsável pelo controle dos servomotores, onde os componentes numerados são: 1) fonte CC 12 volts; 2) regulador *bulk* CC/CC; 3) Arduino MEGA. Na Fig. 21 pode ser visto o manipulador robótico utilizado neste trabalho após as melhorias estruturais descritas, com sua montagem finalizada.

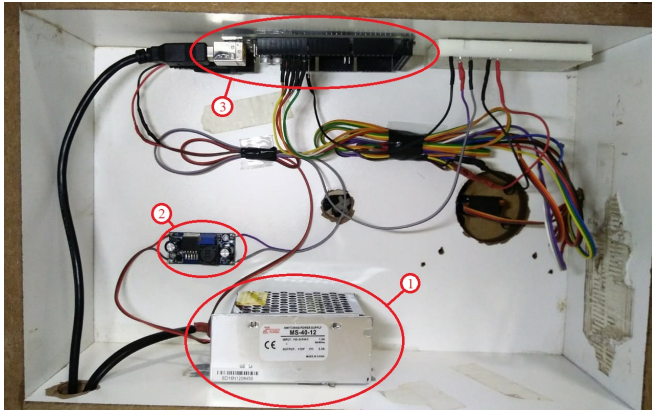


Figura 20. Vista inferior da base do manipulador com a instalação elétrica.

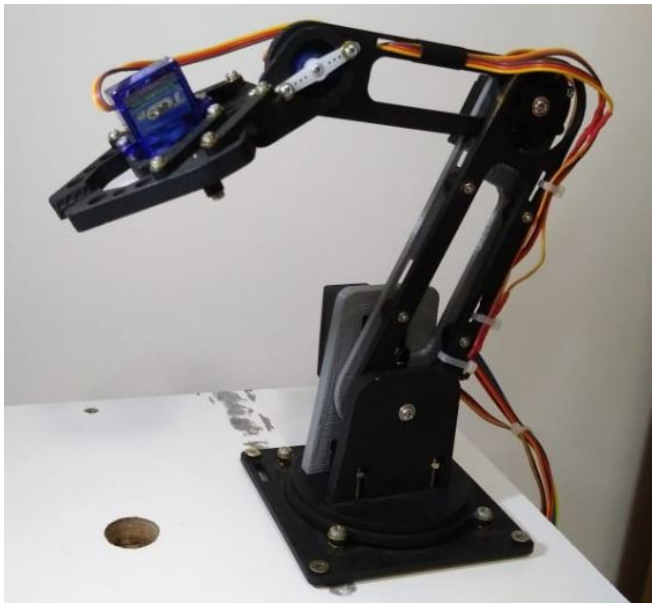


Figura 21. Manipulador robótico finalizado utilizado neste trabalho.

Os servomotores utilizados nas juntas possuem um alcance de movimentação limitado ( $0^\circ$  a  $180^\circ$ ), porém cada junta possui um alcance de movimentação diferente, menor que o alcance do servomotor, este alcance é definido pela geometria da estrutura do manipulador, os ângulos de posição a serem

escritos nos servomotores, mesmo que dentro do alcance do servomotor, não podem estar fora do alcance das juntas a qual o respectivo servomotor é acoplado, pois a própria estrutura impedirá o servomotor de alcançar aquela posição, bloqueando seu eixo e causando sobrecarga, danificando o mesmo. Esta limitação é imposta via código, feita através da verificação dos ângulos antes do envio deles ao Arduino Mega. Estes valores estão relacionados na Tab. III.

Tabela III  
VALORES DOS LIMITES ANGULARES IMPOSTOS AOS SERVOMOTORES DE CADA JUNTA.

	ângulo mínimo	ângulo máximo
Motor 1	$0^\circ$	$180^\circ$
Motor 2	$30^\circ$	$180^\circ$
Motor 3	$0^\circ$	$170^\circ$
Motor 4	$0^\circ$	$180^\circ$
Ferramenta	$48^\circ$	$155^\circ$

Conforme exposto anteriormente, o manipulador empregado neste estudo possui ângulo de *offset* a serem considerados durante a aplicação da cinemática. Os ângulos iniciais  $\theta_i$  foram observados para cada uma das juntas  $m_1$ ,  $m_2$  e  $m_3$  e  $m_4$ . Conforme exposto anteriormente, após o posicionamento do manipulador em posição vertical, foi observado que os ângulos de cada um dos servomotores das juntas possuíam valores iniciais  $\theta_i$ , apresentados na Tab. IV.

Tabela IV  
VALORES DOS ÂNGULOS INICIAIS  $\theta_i$  DE CADA UMA DAS JUNTAS.

	$m_1$	$m_2$	$m_3$	$m_4$
$\theta_i$	$87^\circ$	$89^\circ$	$39^\circ$	$85^\circ$

Desta forma, após a obtenção dos ângulos para alcançar as coordenadas desejadas através da cinemática, deve-se considerar os ângulos  $\theta_i$  de cada servomotor das juntas. Além disso cada servomotor possui um erro de posicionamento que varia em função da sua posição final. 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  $\theta_{fm}$ . Estes ângulos por sua vez são enviados para o Arduino Mega através de comunicação serial. O Arduino Mega ao receber os ângulos os envia 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.

O ângulo final  $\theta_f$  possui relação com o ângulo  $\theta_{fm}$  a ser aplicado ao servomotor da respectiva junta. Neste caso, foram levantados pontos experimentalmente de forma a identificar os valores a serem escritos em cada servomotor em função da posição final desejada. A Fig. 22 ilustra os pontos levantados experimentalmente e a interpolação polinomial de terceiro grau para obtenção do valor de  $\theta_{fm}$ .

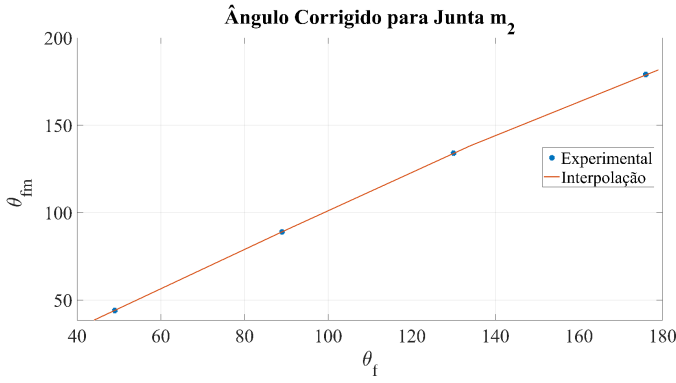


Figura 22. Relação entre o ângulo do servomotor  $\theta_{fm}$  e a posição final  $\theta_f$  para o servomotor da junta  $m_2$ .

O polinômio encontrado para relacionar os pontos obtidos experimentalmente o devido valor corrigido a ser escrito no servomotor da junta  $m_2$  é:

$$\theta_{fm2}(\theta_{f2}) = 8,13 \cdot 10^{-6} \cdot \theta_{f2}^3 + 1,8 \cdot 10^{-3} \cdot \theta_{f2}^2 + 990,4 \cdot 10^{-3} \cdot \theta_{f2} - 7,9932 \quad (17)$$

O mesmo procedimento foi realizado para as juntas  $m_3$  e  $m_4$ , resultando nos gráficos das figuras 23 e 24.

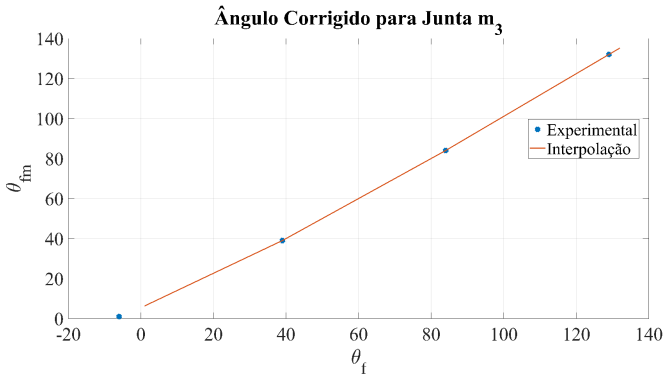


Figura 23. Relação entre o ângulo do servomotor  $\theta_{fm}$  e a posição final  $\theta_f$  para o servomotor da junta  $m_3$ .

As equações para os ângulos das juntas  $m_3$  e  $m_4$  são:

$$\theta_{fm3}(\theta_{f3}) = -7,31 \cdot 10^{-6} \cdot \theta_{f3}^3 + 2,58 \cdot 10^{-3} \cdot \theta_{f3}^2 + 768,83 \cdot 10^{-3} \cdot \theta_{f3} + 5,51 \quad (18)$$

$$\theta_{fm4}(\theta_{f4}) = -202,33 \cdot 10^{-9} \cdot \theta_{f4}^4 + 88,83 \cdot 10^{-6} \cdot \theta_{f4}^3 + -12,23 \cdot 10^{-3} \cdot \theta_{f4}^2 + 1,48 \cdot \theta_{f4} + 2,84 \quad (19)$$

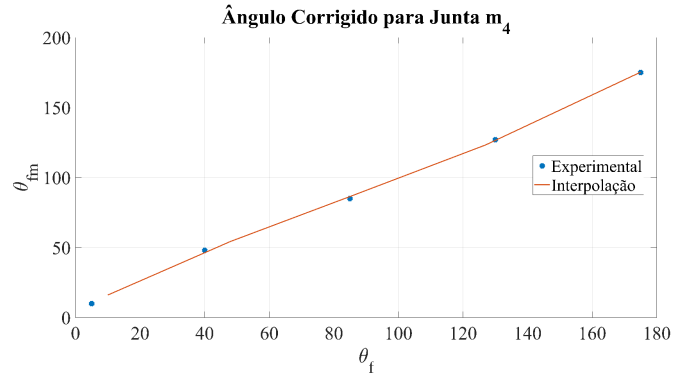


Figura 24. Relação entre o ângulo do servomotor  $\theta_{fm}$  e a posição final  $\theta_f$  para o servomotor da junta  $m_4$ .

A fim de suavizar a movimentação do manipulador a implementação do código no Arduino Mega faz uso de uma biblioteca para controle de servomotores que, além de controlar as posições dos servomotores individualmente, também controla a 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  $0^\circ$  a  $180^\circ$  e a velocidade de 0 a 255, que equivale de 0 a 100% da velocidade máxima do servo. Internamente a biblioteca consegue este ajuste escrevendo no servomotor em intervalos regulares de tempo posições intermediárias incrementadas de grau em grau entre a posição atual do servomotor e a posição final desejada, onde a velocidade determina o intervalo de tempo entre cada incremento.

### B. 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 da interface gráfica pode ser vista na Fig. 25.

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

- Região "mão direita"

Aqui são encontrados os indicadores das coordenadas  $x$ ,  $y$  e  $z$  que o Microsoft Kinect coleta da mão direita, as coordenadas apresentadas estão rotacionadas, ou seja, são coordenadas referentes ao sistema de coordenadas do manipulador robótico, porém elas ainda não foram transformadas completamente para o espaço do manipulador. Estes campos são atualizados em tempo real a todo momento e não são modificados ou influenciados

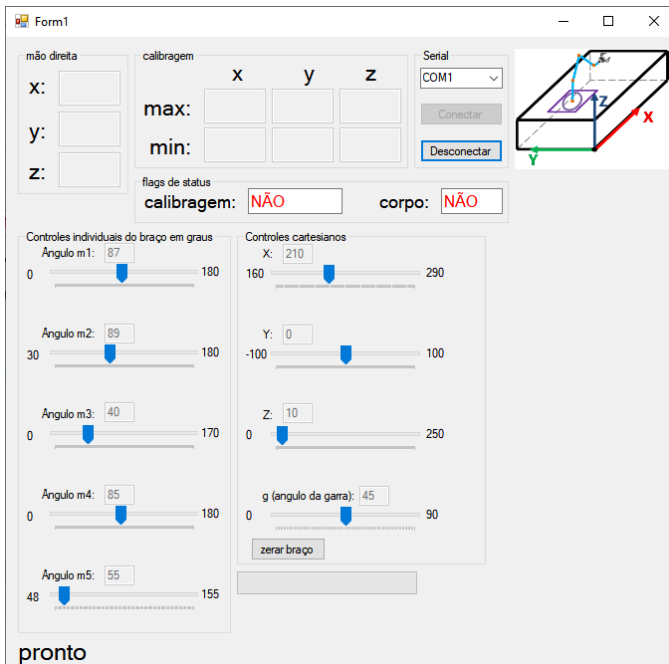


Figura 25. Interface gráfica da aplicação desenvolvida em .NET.

pele funcionamento do programa, ou seja, enquanto o corpo do operador estiver em frente ao campo de visão do Microsoft Kinect estes campos estão sendo atualizados.

- Região "**calibração**"

Aqui são encontrados campos que indicam os limites resultantes da calibração, estes campos são alterados apenas durante a calibração e mantém os valores durante toda a operação, apenas são apagados quando o corpo do operador sai do campo de visão do Microsoft Kinect. Estes valores são utilizados nas equações de transformação do espaço do operador capturado pelo Microsoft Kinect para o espaço do manipulador.

- Região "**Serial**"

Os controles referentes ao gerenciamento da conexão serial com o Arduino Uno. As portas COM são atualizadas automaticamente ao se clicar na lista para selecionar a porta a ser conectada.

- Região "**flags de status**"

As flags de status se modificam de acordo com a execução do controle, como ilustrado na Fig. 10. A flag "corpo" indica quando um ou mais corpos são detectados pelo Microsoft Kinect, esta flag é importante na indicação de falsa detecção de corpos pelo Microsoft Kinect, que ocorreram algumas vezes durante testes executados em ambientes pequenos com muitos móveis e detalhes no campo de visão do Microsoft Kinect. Já a flag de calibração indica quando a calibração já foi executada ou está incompleta, isto ocorre quando, durante o processo de calibração, o operador abaixar a mão esquerda. São

colocadas em seu estado inicial quando o Microsoft Kinect não está detectando um corpo.

- Região "**Controles individuais do braço em graus**"

São controles deslizantes que o ajuste manual da posição angular de cada servomotor individualmente. Durante os testes de posicionamento e precisão do manipulador foi necessário o controle direto de cada servomotor individualmente. Através destes controles os alcances das juntas foram definidos. Inicialmente os limites inferiores e superiores de cada controle deslizante foi colocado em 0° e 180°, respectivamente. Logo depois o manipulador foi conectado e cada controle foi ajustado individualmente, de maneira a verificar os limites angulares de cada junta de forma que o eixo do servomotor não fosse bloqueado pela própria estrutura dos elos e junta. Estes valores descritos na Tab. III foram posteriormente colocados como os limites dos controles deslizantes individuais, de maneira que estes ultrapassem os limites das juntas.

- Região "**Controles cartesianos**"

Estes 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. Estes controles são utilizados para testes do algoritmo de cinemática inversa e testes dos ajustes dos ângulos a serem escritos nos motores, também foram utilizados para determinação dos limites superior e inferior dos eixos coordenados do espaço de trabalho do manipulador. Durante a operação normal através das coordenadas dada pelo Microsoft Kinect a inclinação da garra, que é o quarto parâmetro da cinemática, não é obtido pelo *tracking* do Microsoft Kinect, logo este valor foi fixado em 45°, pois durante os teste este foi o valor observado que maximiza o alcance do manipulador dentro do seu espaço de trabalho. Também é observado um botão com o nome "zerar braço", este botão tem a função de escrever em todos os motores os ângulos referentes a posição inicial, ou posição de referência, ou seja, este botão escreve nos servomotores os ângulos  $\theta_i$ , fazendo o manipulador se movimentar para a posição descrita na Fig. 9.

- Demais elementos gráficos

Na parte inferior da tela onde é observado a palavra "pronto" é mostrada diversas informações referentes aos passos que o operador deve seguir durante as etapas de calibração, sincronização e operação do manipulador conforme são ilustradas na Fig. 10. Já logo abaixo da região dos Controles cartesianos é observada uma barra que tem a função de indicar a contagem de um tempo de 3 segundos ao ser preenchida, este tempo é necessário para o correto posicionamento do operador, pois ao entrar no campo de visão do Microsoft Kinect e ser detectado ele não se encontra totalmente posicionado de frente para o aparelho e pronto para iniciar os procedimentos de

operação. Já no canto superior esquerdo há uma ilustração do manipulador e a orientação do seu sistema de coordenadas para que o operador se situe e compreenda a orientação do seu corpo em relação ao manipulador.

## V. CONCLUSÃO

O objetivo desse estudo foi alcançado, a operação do manipulador robótico por um operador remotamente é possível, podendo até ser realizada pelo operador enquanto o mesmo observa o local de trabalho e o manipulador robótico por imagem de uma câmera instalada no local de trabalho, possibilitando assim a operação em um local totalmente isolado onde o manipulador robótico estaria contido.

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, tronco, etc) se sobrepõem em frente ao campo de visão do Microsoft Kinect. Também foi observado que o software executa as funções para todos os corpos presentes no campo de visão do Microsoft Kinect, sendo assim ao passar uma pessoa por trás do operador o funcionamento é prejudicado, sendo necessário reiniciar o sistema, este problema encontrado pode ser solucionado com o devido tratamento no código fonte da aplicação .NET.

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 remodelar as peças dos elos e juntas até a substituição dos servomotores e a estrutura completamente.

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 trabalho no meio acadêmico, porém para aplicações onde 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.

## REFERÊNCIAS

- [1] Oliveira, Otávio; Oliveira, Alessandra; Almeida, Renan. Gestão da segurança e saúde no trabalho em empresas produtoras de baterias automotivas: um estudo para identificar boas práticas. V. 20, n. 3, p. 481-490, São Paulo, Sept. 2010.
- [2] M. Cole, T. Driscoll and K. Martin, "Installation, operation and maintenance considerations for electrical equipment in hazardous location,"2010 IEEE IAS Electrical Safety Workshop, pp. 68-72. doi:10.1109/ESW.2010.6164452. Memphis, TN, 2010.
- [3] A. Bozek, C. Gordon and N. Phillips, "Electrical Hazardous Area Classification design as a basis for safer operations"2014 IEEE Petroleum and Chemical Industry Technical Conference (PCIC), pp. 459-464. doi:10.1109/PCICon.2014.6961911. San Francisco, CA, 2014,
- [4] J. Ehlenbröker, U. Mönks, D. Wesemann and V. Lohweg, "Condition monitoring for hazardous material storage,"Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), pp. 1-4. doi:10.1109/ETFA.2014.7005264. Barcelona, 2014.
- [5] Iqbal, Jamshed; Khan, Zeashan Hameed; Khalid, Azfar. Prospects of robotics in food industry. Food Sci. Technol (Campinas). V. 37, n. 2, p. 159-165, Campinas, Apr. 2017.
- [6] Craig, J.J. Introduction to Robotics: Mechanics and Control. ISBN 9780133489798. Pearson. 2018.

- [7] Spong, Mark W. Hutchinson, Seth. Vidyasagar M. Robot Dynamics and Control, 2<sup>o</sup> edition. John Wiley & Sons. ISBN 9783319544120. 2004.
- [8] E. Amininan, F. H. Sheikha, F. Baghyari, S. Hosseini, M. Najmabadi and A. Akbarzadeh, "Explicit Inverse Kinematic Solution for the Industrial FUM Articulated Arm using Dual Quaternion Approach,"2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM), pp. 602-607. Tehran, 2017.
- [9] K. Wang and Y. Li, "Comparison of control methods: Learning robotics manipulation with contact dynamics", 2018 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN), pp. 1-7. Brisbane, QLD, 2018.
- [10] R. K. Megalingam, N. Saboo, N. Ajithkumar, S. Unny and D. Menon, "Kinect based gesture controlled Robotic arm: A research work at HuT Labs,"2013 IEEE International Conference in MOOC, Innovation and Technology in Education (MITE), pp. 294-299. Jaipur, 2013.
- [11] S. Amatya and S. Petchartee, "Real time kinect based robotic arm manipulation with five degree of freedom,"2015 Asian Conference on Defence Technology (ACDT), pp. 1-6. Hua Hin, 2015.
- [12] NR, Norma Regulamentadora Ministério do Trabalho e Emprego. NR-10 - Segurança em Instalações e Serviços em Eletricidade. Brasil. 2016.
- [13] NR, Norma Regulamentadora Ministério do Trabalho e Emprego. NR-15 - Atividades e Operações Insalubres. Brasil. 2018.
- [14] NR, Norma Regulamentadora Ministério do Trabalho e Emprego. NR-16 - Atividades e Operações Perigosas. Brasil. 2015.
- [15] NR, Norma Regulamentadora Ministério do Trabalho e Emprego. NR-33 - Segurança e Saúde nos Trabalhos em Espaços Confinados. Brasil. 2012.
- [16] NR, Norma Regulamentadora Ministério do Trabalho e Emprego. NR-35 - Trabalho em Altura. Brasil. 2016.