

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
FACULDADE DE ENGENHARIA
ENGENHARIA ELÉTRICA – HABILITAÇÃO EM ROBÓTICA E AUTOMAÇÃO
INDUSTRIAL

Gabriel Bruno de Oliveira Pereira

SISTEMA SUPERVISÓRIO PARA ATUAÇÃO CONJUNTA DE MANIPULADORES
ROBÓTICOS APLICADOS A LINHAS DE PRODUÇÃO NA INDÚSTRIA 4.0

Juiz de Fora

Julho, 2018

Gabriel Bruno de Oliveira Pereira

**SISTEMA SUPERVISÓRIO PARA ATUAÇÃO CONJUNTA DE MANIPULADORES
ROBÓTICOS APLICADOS A LINHAS DE PRODUÇÃO NA INDÚSTRIA 4.0**

Trabalho de conclusão de curso apresentado à
faculdade de Engenharia da Universidade
Federal de Juiz de Fora, como requisito para
obtenção do grau de Engenheiro Eletricista.

Orientador: Prof. Dr. Leonardo Rocha Olivi.

Co-Orientadora: Prof^ª. Dr^ª. Ana Sophia Cavalcanti Alves Vilas Boas

Juiz de Fora

Julho, 2018

Imprimir na parte inferior, no verso da folha de rosto a ficha disponível em:
<http://www.ufjf.br/biblioteca/servicos/usando-a-ficha-catalografica/>

A FICHA CATALOGRÁFICA SERÁ INSERIDA NA VERSÃO FINAL

Gabriel Bruno de Oliveira Pereira

**SISTEMA SUPERVISÓRIO PARA ATUAÇÃO CONJUNTA DE MANIPULADORES
ROBÓTICOS APLICADOS A LINHAS DE PRODUÇÃO NA INDÚSTRIA 4.0**

Trabalho de conclusão de curso apresentado à
faculdade de Engenharia da Universidade
Federal de Juiz de Fora, como requisito para
obtenção do grau de Engenheiro Eletricista.

BANCA EXAMINADORA

Prof. Dr. Leonardo Rocha Olivi - Orientador
Universidade Federal de Juiz de Fora

Prof^ª. Dr^ª. Ana Sophia Cavalcanti Alves Vilas Boas - Orientadora
Universidade Federal de Juiz de Fora

Prof. Dr. Exuperry Barros Costa
Universidade Federal de Juiz de Fora

Prof. Dr. Luís Henrique Lopes Lima
Universidade Federal de Juiz de Fora

Esp. Ronnie Silveira de Oliveira
Mercedes-Benz - Juiz de Fora

AGRADECIMENTOS

Agradeço primeiramente ao ex-presidente Luiz Inácio Lula da Silva, que em seu governo criou o programa REUNI, programa este que aumentou o número de vagas nas instituições públicas de ensino superior, e foi em uma dessas vagas que pude entrar na Universidade Federal de Juiz de Fora.

Agradeço também aos meus pais que sempre me apoiaram nesta trajetória, sempre me estimularam a estudar, a absorver o máximo de conhecimento possível e me deram todo o suporte para que pudesse terminar minha graduação.

Por fim, gostaria também de agradecer a todos os professores que participaram da minha formação, mas em especial, ao Dr. Leonardo Rocha Olivi e a Dr^a Ana Sophia Cavalcanti Alves Vilas Boas, pois me estenderam a mão quando precisei de um orientador para este trabalho, mas também por me mostrarem que os próximos alunos terão professores de verdade, que se esforçam para se capacitarem, se importam com a didática de suas aulas e ajudam os discentes com o máximo de humildade e respeito.

RESUMO

Este trabalho tem por objetivo desenvolver um sistema supervisorio completo para dois ou mais manipuladores robóticos, de modo a trabalharem em uma linha de montagem industrial de maneira coordenada e simultânea, para atingirem determinadas configurações no espaço geométrico, com posição e orientação, de tal forma que não haja colisão entre eles. Para tal tarefa foram utilizados os conceitos de cinemática direta, cinemática inversa e o método do gradiente descendente. Primeiramente, foi necessário o desenvolvimento da cinemática direta dos manipuladores, utilizando a convenção de Denavit-Hartenberg, para se obter a cadeia cinemática dos braços robóticos. Posteriormente, o problema da cinemática inversa foi resolvido utilizando o método de otimização iterativo do gradiente descendente, o que possibilitou ambos manipuladores chegarem em determinadas configurações objetivo no espaço de trabalho com as posições e orientações desejadas. Esses conceitos matemáticos foram aplicados ao programa MATLAB e com o auxílio do simulador V-REP foi possível simular a tarefa estabelecida. Os resultados obtidos são satisfatórios validando assim os métodos utilizados para solucionar o problema.

Palavras-chave: Sistema Supervisorio, Manipuladores Robóticos, Denavit-Hartenberg, Cinemática Inversa, Método do Gradiente Descendente, MATLAB, V-REP.

ABSTRACT

This work aims to develop a complete supervisory system for two or more robotic manipulators, so as to work in an industrial assembly line in a coordinated and simultaneous way, to reach certain configurations in the geometric space, with position and orientation, in such a way that there is no collision between them. For this task, the concepts of direct kinematics, inverse kinematics and the descending gradient method were used. First, it was necessary to develop the direct kinematics of the manipulators, using the Denavit-Hartenberg convention, to obtain the kinematic chain of the robotic arms. Subsequently, the problem of inverse kinematics was solved using the iterative optimization method of the descending gradient, which enabled both manipulators to arrive in certain objective configurations in the workspace with the desired positions and orientations. These mathematical concepts were applied to the MATLAB program and with the help of the V-REP simulator it was possible to simulate the established task. The results obtained are satisfactory thus validating the methods used to solve the problem.

Keywords: Supervisory System, Robot Manipulators, Denavit-Hartenberg, Inverse Kinematics, Descending Gradient Method, MATLAB, V-REP.

SUMÁRIO

1	INTRODUÇÃO	13
2	REVISÃO BIBLIOGRÁFICA	14
2.1	MANIPULADORES ROBÓTICOS	14
2.1.1	Graus de liberdade	14
2.1.2	Juntas	14
2.1.3	Elos	15
2.1.4	Efetuadores	15
2.2	TRANSFORMAÇÕES HOMOGÊNEAS.....	15
2.2.1	Sistemas de coordenadas	15
2.2.2	Translações	16
2.2.3	 rotações	16
2.2.3.1	<i>Rotação no eixo x</i>	17
2.2.3.2	<i>Rotação no eixo y</i>	18
2.2.3.3	<i>Rotação no eixo z</i>	19
2.3	DENA VIT-HARTENBERG.....	20
2.4	CINEMÁTICA DIRETA.....	20
2.5	CINEMÁTICA INVERSA.....	21
2.6	INDÚSTRIA 4.0.....	23
2.7	V-REP.....	24
3	DESENVOLVIMENTO	26
3.1	V-REP.....	26
3.2	CINEMÁTICA DIRETA.....	28
3.3	CINEMÁTICA INVERSA.....	29
3.3.1	Função objetivo	29
3.3.2	Gradiente	30
3.4	SISTEMA SUPERVISÓRIO.....	32
4	RESULTADOS	33
4.1	RESULTADOS DA TRAJETORIA.....	33
5	CONCLUSÃO	42
	REFERÊNCIAS	43

LISTA DE ILUSTRAÇÕES

Figura 1 – Rotações dos eixos.....	17
Figura 2 – Rotação no eixo x.....	17
Figura 3 – Rotação no eixo y.....	18
Figura 4 – Rotação no eixo z.....	19
Figura 5 – Configuração da matriz transformação homogênea final.....	21
Figura 6 – Pseudo-algoritmo do método do gradiente descendente.....	23
Figura 7 – Manipuladores com as furadeiras.....	26
Figura 8 – <i>Dummy</i>	27
Figura 9 – Cena final, com manipuladores e pontos a serem alcançados.....	27
Figura 10 – Função no <i>script</i> do cubo.....	28
Figura 11 – Primeiro ponto objetivo.....	33
Figura 12 – Segundo ponto objetivo.....	34
Figura 13 – Voltando para o primeiro ponto objetivo.....	35
Figura 14 – Chegando ao terceiro ponto objetivo.....	36
Figura 15 – Quarto ponto objetivo.....	37
Figura 16 – Entre os pontos objetivos 3 e 1.....	38
Figura 17 – Chegando na distância limite de 40 cm de distância.....	39
Figura 18 – Manipulador 1 parado pelo sistema supervisorio.....	40
Figura 19 – Sistema supervisorio libera o manipulador 1.....	40
Figura 20 – Manipuladores na posição final.....	41

LISTA DE TABELAS

Tabela 1 – Tabela de Denavit-Hartenberg para o robô KUKA iiwa 14 R820.....	28
Tabela 2 – Dimensões do robô KUKA iiwa 14 R820.....	29
Tabela 3 – Valor médio do erro.....	41

1 INTRODUÇÃO

Com o passar das décadas, a evolução da tecnologia traz máquinas e equipamentos eletrônicos na participação e desenvolvimento de diversos tipos de atividade, sejam elas domésticas, como os celulares, roteadores, impressoras gráficas, e também em escala industrial como os CLP's, CNC's, sensores e manipuladores robóticos [1].

Juntamente com estes equipamentos, os conceitos de automação e controle ganharam força, visto que aliando-se os dois as indústrias começaram a ser capazes de produzir em larga escala, de forma padronizada, com boa qualidade, e também passam a diminuir as falhas e defeitos de seus produtos, reduzir a quantidade de acidentes de trabalho e isso tudo satisfazendo outros dois requisitos importantes da área industrial que são otimização de tempo e de custos, apesar do alto valor inicial que se tem para a aquisição desses equipamentos [1], [2].

Os efeitos positivos que essas tecnologias trouxeram para o setor industrial as tornam parte imprescindível em qualquer processo de produção. Sendo assim, a compreensão das técnicas de controle, o conhecimento sobre as partes que formam a automação, como os sensores, as linguagens de programação para CLP's, os tipos de protocolos e o conhecimento de como funcionam os atuadores, que podem ser os manipuladores robóticos ou até mesmo carros autônomos que ajudam no transporte de peças, fazem com que os engenheiros que atuam na área de controle e automação sejam profissionais de muita relevância dentro das indústrias [2], [3].

Considerando a importância da tecnologia nos processos industriais, este trabalho tem por objetivo a automação de uma célula de uma linha de produção de montagem de veículos automotores. Propõe-se um sistema supervisor para que mais de um braço robótico, no caso o KUKA LBR iiwa 14 R820 produzido pela KUKA Industrial Robots [4] com uma furadeira acoplada aos seus efetadores (do inglês, *end-effector*), consigam, em conjunto e de maneira coordenada, atingir um número específico de pontos objetivos com precisão e acurácia, e realizando ações nos chassis dos veículos de maneira satisfatória.

Pela impossibilidade de realizar este experimento em um robô real, este trabalho utilizará uma simulação da cena real, desenvolvida por meio do programa MATLAB [5], e o simulador V-REP PRO EDU [6], disponibilizado de forma gratuita em sua versão educacional, o qual fornece os manipuladores em tamanho real, tornando a simulação mais fiel a realidade.

2 REVISÃO BIBLIOGRÁFICA

Para a realização deste trabalho, diversos conceitos e técnicas foram utilizadas em conjunto, possibilitando o sucesso das missões impostas aos robôs. Neste capítulo serão apresentados os conceitos teóricos utilizados para a realização deste trabalho.

2.1 MANIPULADORES ROBÓTICOS

Os manipuladores robóticos, também chamados por braços robóticos, são robôs normalmente com a aparência de um braço humano, e como tal, são projetados para que sejam capazes de realizar tarefas de acordo com a necessidade da sua atividade, sendo necessário o acoplamento de alguma ferramenta ao seu punho, a qual servirá como efetuator (do inglês, *end-effector*) [1].

Assim como o braço humano, que possui partes móveis e rígidas, como o cotovelo e o antebraço, respectivamente, analogamente podemos definir as partes que compõem um manipulador. As partes rígidas são denominadas por elos (do inglês, *links*) e as partes móveis são denominadas por juntas (do inglês, *joints*). A seguir, definem-se alguns dos principais termos relativos à área [1], [7].

2.1.1 Graus de Liberdade

Graus de liberdade (do inglês, *Degrees of Freedom*) é o número de variáveis independentes de posição que precisam ser especificadas para se definir a localização de todas as partes do mecanismo robótico (articulações). Sendo assim, quando o movimento relativo ocorre em apenas um eixo cartesiano, a junta tem apenas um grau de liberdade [1].

2.2.2 Juntas

As juntas (do inglês, *joints*), assim como o ombro e o cotovelo, são as partes capazes de realizar movimento nas partes rígidas, os elos (*links*), acopladas ao seu eixo. Existem juntas de diferentes tipos como as esféricas, as cilíndricas, entretanto as mais comuns nos manipuladores industriais são as juntas prismáticas e as rotacionais [1], [7].

As juntas prismáticas são aquelas responsáveis pelo movimento de translação sobre o eixo, ou seja, elas apenas tem a capacidade de se estender ao longo do eixo e aumentar ou

diminuir seu comprimento. Possui apenas um grau de liberdade. As juntas rotacionais, por sua vez, são aquelas que proporcionam uma rotação no eixo. Assim como a prismática, também possui apenas um grau de liberdade [1], [7].

2.2.3 Elos

Os elos (do inglês, *links*), são as partes rígidas de um manipulador robótico. Essas partes servem como a ligação entre duas juntas e, por serem rígidas, não realizam movimento. A sua mudança de posição, relativa a base do manipulador, depende do tipo de movimento realizado pela junta à qual o elo se liga [1], [8].

2.2.4 Efetuador

O efetuador, avindo do termo em inglês *end-effector*, é a parte final do robô, que está acoplada ao punho do manipulador, assim como a mão se liga ao punho humano. É a parte que, por meio de uma ferramenta acoplada como uma furadeira ou uma ponta de solda, dentre outros é capaz de realizar o trabalho definido [1], [7], [8].

2.2 TRANSFORMAÇÕES HOMOGÊNEAS

Como foi dito anteriormente, os braços robóticos possuem componentes denominados por juntas que são capazes de movimentar as partes do robô. Com isso, é necessário compreender como funciona a dinâmica do robô para que ele possa operar no ambiente conforme o planejado, ou seja, como a rotação e/ou a translação das juntas que compõem o manipulador afetam a posição final do efetuador. Desta maneira, um estudo matemático desses movimentos se faz necessário para compreender e modelar todas as partes do manipulador [7].

2.2.1 Sistemas de coordenadas

Um sistema de coordenadas, também conhecido por seu termo em inglês, *frame*, é um conjunto de coordenadas geométricas utilizadas como referência, no caso da aplicação deste trabalho, para as diversas partes do robô manipulador. Sendo assim, a partir de um sistema de coordenadas inercial é possível descrever qualquer ponto do espaço [8].

Desta forma, é possível representar as juntas e elos de um manipulador por um conjunto de sistemas de coordenadas (*frames*) coerentemente posicionados a partir do *frame* inercial, e através das matrizes de translação e rotação, saber onde o *frame* relacionado ao efetuador está posicionado com relação ao *frame* inercial [8].

2.2.2 Translações

Translações são deslocamentos do sistema de coordenadas (*frames*) pelo espaço geométrico sem alterar a orientação dos eixos, ou seja, sem movimento de rotação [8]. Assim, a translação de um *frame* {0} para o *frame* {1} pode ser descrito matricialmente como mostram as Equações (1) e (2):

$$X_{\{1\}} = T(\Delta x, \Delta y, \Delta z). X_{\{0\}} \quad (1)$$

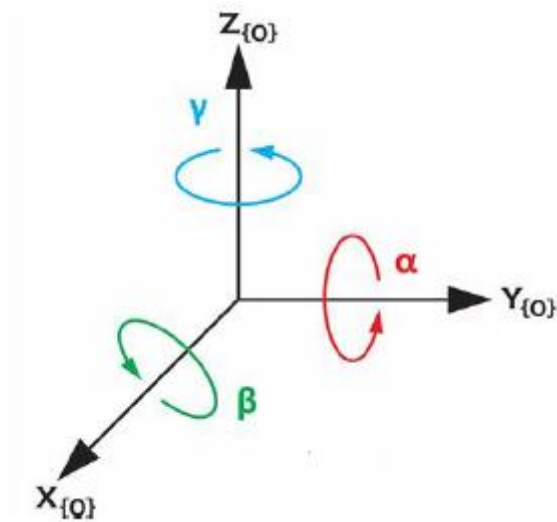
$$\begin{bmatrix} x_{\{1\}} \\ y_{\{1\}} \\ z_{\{1\}} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{\{0\}} \\ y_{\{0\}} \\ z_{\{0\}} \\ 1 \end{bmatrix} \quad (2)$$

A matriz T, na Equação (1), expandida na Equação (2), é denominada Matriz de Translação, pois realiza o movimento descrito e, de maneira cinemática, descreve a atuação de uma junta prismática.

2.2.3 Rotações

Rotações são movimentos que apenas alteram a orientação dos eixos, mantendo a origem dos sistemas de coordenadas inalterada. Para um espaço de três dimensões existem três tipos de rotações possíveis para transformar um *frame* {0} em um *frame* {1}, as rotações sobre o eixo x, conhecida pelo termo em inglês *roll*, sobre o eixo y (do inglês, *pitch*) e sobre o eixo z (do inglês, *yaw*). O sentido de rotação positivo respeita a regra de Fleming, também conhecida como a regra da mão direita, como mostrado na Figura 1 [1], [7], [8], [9].

Figura 1 – Rotações dos eixos.

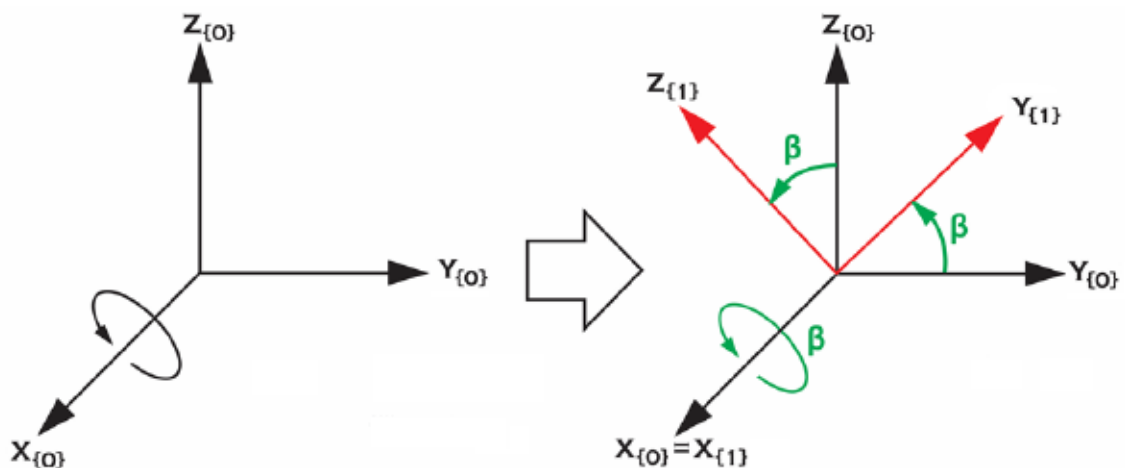


Fonte: Olivi, L.R., 2016 [9].

2.2.3.1 Rotação no eixo x

Uma rotação positiva no eixo x com um ângulo de β graus, por exemplo, como mostra a Figura 2, faz com que a orientação do eixo x permaneça a mesma, entretanto desloca os eixos y e z também em β graus. Assim, a forma matricial de uma rotação no eixo x pode ser descrita como mostram as Equações (3) e (4) [8].

Figura 2 – Rotação no eixo x.



Fonte: Olivi, L.R, 2016 [9].

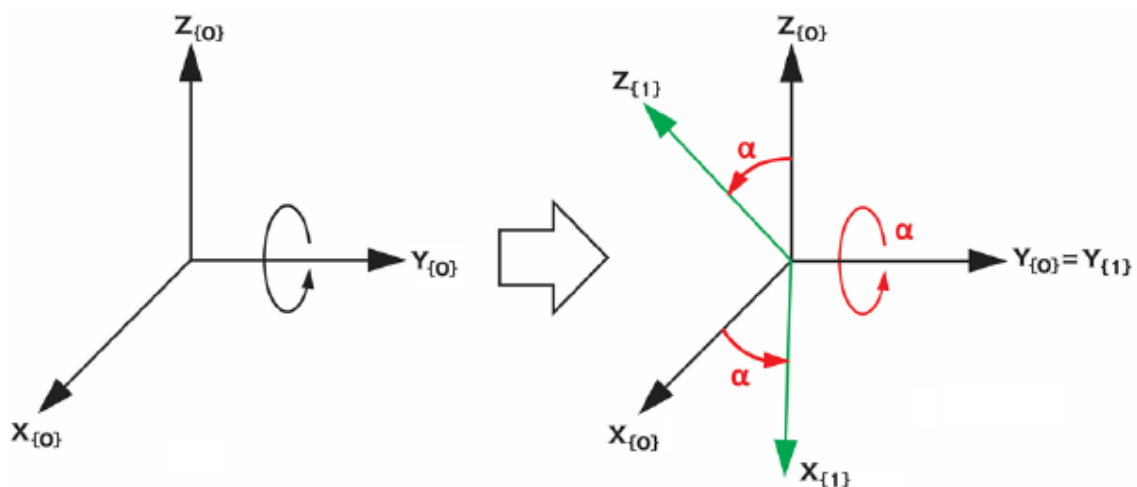
$$\begin{cases} X_{\{1\}} = X_{\{0\}} \\ Y_{\{1\}} = \cos \beta Y_{\{0\}} - \sin \beta Z_{\{0\}} \\ Z_{\{1\}} = \sin \beta Y_{\{0\}} + \cos \beta Z_{\{0\}} \end{cases} \quad (3)$$

$$\begin{bmatrix} x_{\{1\}} \\ y_{\{1\}} \\ z_{\{1\}} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \sin \beta & -\sin \beta & 0 \\ 0 & \sin \beta & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{\{0\}} \\ y_{\{0\}} \\ z_{\{0\}} \\ 1 \end{bmatrix} \quad (4)$$

2.2.3.2 Rotação no eixo y

Uma rotação positiva no eixo y de α graus, por exemplo, como mostra a Figura 3, faz com que a orientação do eixo y permaneça a mesma, entretanto desloca os eixos x e z também em α graus. Assim, a forma matricial de uma rotação no eixo y pode ser descrita como mostra as Equações (5) e (6) [8], [9].

Figura 3 – Rotação no eixo y.



Fonte: Olivi, L.R., 2016 [9].

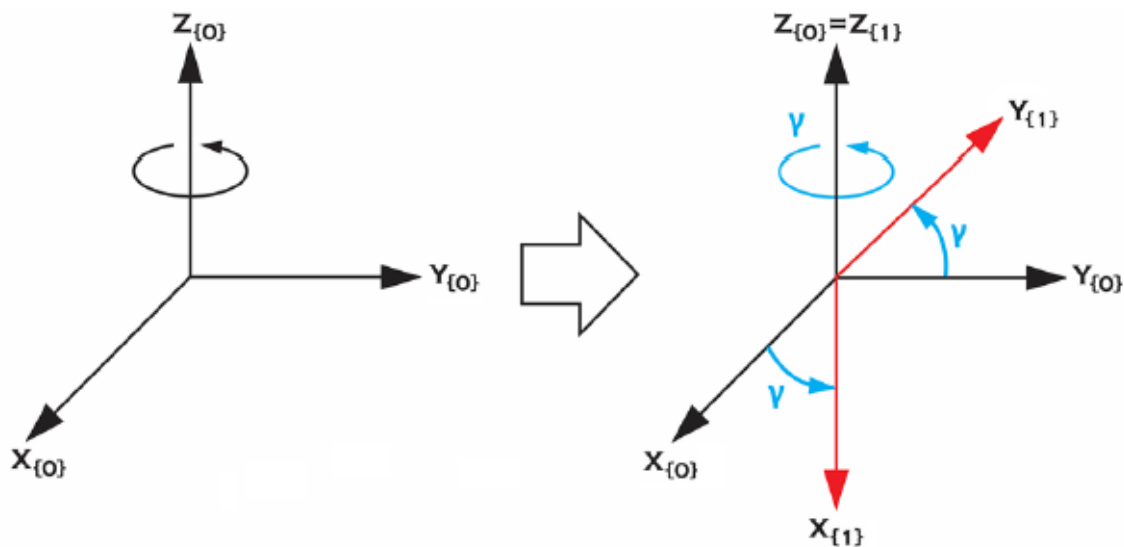
$$\begin{cases} X_{\{1\}} = \cos \alpha X_{\{0\}} + \sin \alpha Z_{\{0\}} \\ Y_{\{1\}} = Y_{\{0\}} \\ Z_{\{1\}} = -\sin \alpha X_{\{0\}} + \cos \alpha Z_{\{0\}} \end{cases} \quad (5)$$

$$\begin{bmatrix} x_{\{1\}} \\ y_{\{1\}} \\ z_{\{1\}} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{\{0\}} \\ y_{\{0\}} \\ z_{\{0\}} \\ 1 \end{bmatrix} \quad (6)$$

2.2.3.3 Rotação no eixo z

Uma rotação positiva no eixo z de γ graus, por exemplo, como mostra a Figura 4, faz com que a orientação do eixo z permaneça a mesma, entretanto desloca os eixos x e y também em γ graus. Assim, a forma matricial de uma rotação no eixo z pode ser descrita como mostra as Equações (7) e (8) [8].

Figura 4 – Rotação no eixo z.



Fonte: Olivi, L.R., 2016 [9].

$$\begin{cases} X_{\{1\}} = \cos \gamma X_{\{0\}} - \sin \gamma Y_{\{0\}} \\ Y_{\{1\}} = \sin \gamma X_{\{0\}} + \cos \gamma Y_{\{0\}} \\ Z_{\{1\}} = Z_{\{0\}} \end{cases} \quad (7)$$

$$\begin{bmatrix} x_{\{1\}} \\ y_{\{1\}} \\ z_{\{1\}} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 & 0 \\ \sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{\{0\}} \\ y_{\{0\}} \\ z_{\{0\}} \\ 1 \end{bmatrix} \quad (8)$$

Tendo em mãos o conhecimento dessas matrizes, com a combinação delas é possível transformar o sistema de coordenadas inercial $\{0\}$ no sistema do efetuador. Essa sequência de combinações é chamada de Transformações Homogêneas, normalmente representadas por T, H, TH ou TR, dependendo do autor [1], [7], [8], [9].

Como os manipuladores são compostos por várias juntas e elos, para se obter a transformação homogênea que descreva todo o robô até o efetuador, são necessárias várias

transformações. Para que a descrição seja correta, utilizam-se convenções estabelecidas mundialmente. Assim, neste trabalho, a transformação do sistema de coordenadas inercial {0} para o do efetuador adota-se a Convenção Denavit-Hartenberg [7].

2.3 DENAVIT-HARTENBERG

O método de modelagem da cadeia cinemática de um manipulador de Denavit-Hartenberg foi desenvolvida pelos pesquisadores Jacques Denavit e por Richard Hartenberg em 1995. Este método segue o conceito das transformações homogêneas no que diz respeito a rotação positiva seguir a regra da mão direita e que o eixo de ação das juntas é o eixo z [7].

O método propõe que a modelagem da cadeia cinemática do braço robótico seja feita apenas utilizando transformações de rotação e translação no eixo x e z, respeitando as características construtivas do robô. Existem abordagens distintas com relação ao método de Denavit-Hartenber no concernente à ordem das transformações, dependendo do autor [8].

Neste trabalho, utiliza-se a abordagem que considera as operações no eixo z precedem às operações no eixo x, e o eixo de ação das juntas sempre se dá no eixo z. Além disso, não é permitido transformar um sistema de coordenadas {0} em um sistema de coordenadas {1} realizando rotações ou translações no eixo y. A transformação entre *frames*, portanto, é dada conforme a Equação (9) a seguir:

$$TH_{\{i-1\}}^{\{i\}} = A_i = T_z(d_i)R_z(\theta_i)T_x(a_i)R_x(\alpha_i) \quad (9)$$

2.4 CINEMÁTICA DIRETA

Os manipuladores robóticos são utilizados para realizarem tarefas, e o princípio básico dessas tarefas, é fazer com que o efetuador do robô atinja determinados objetivos no espaço de trabalho [1].

A cinemática direta tem por objetivo obter a posição e orientação do efetuador em função das variáveis de junta [7]. Desta forma, a cinemática direta consiste em aplicar rotações e translações no braço robótico e obter a configuração final do efetuador, visto que, qualquer matriz de transformação homogênea final possui a seguinte característica, como se destaca na Figura 5.

Figura 5 – Configuração da matriz transformação homogênea final.

$$TR_{\{0\}}^{\{n\}} = \begin{bmatrix}
 \begin{array}{c} \text{Vetor} \\ \text{Unitário} \\ \text{Eixo-X} \end{array} & \begin{array}{c} \text{Vetor} \\ \text{Unitário} \\ \text{Eixo-y} \end{array} & \begin{array}{c} \text{Vetor} \\ \text{Unitário} \\ \text{Eixo-Z} \end{array} & \begin{array}{c} \text{Des-} \\ \text{loca-} \\ \text{mento} \end{array} \\
 \begin{array}{c} P_x^{(x)} \\ P_y^{(x)} \\ P_z^{(x)} \\ \mathbf{0} \end{array} & \begin{array}{c} P_x^{(y)} \\ P_y^{(y)} \\ P_z^{(y)} \\ \mathbf{0} \end{array} & \begin{array}{c} P_x^{(z)} \\ P_y^{(z)} \\ P_z^{(z)} \\ \mathbf{0} \end{array} & \begin{array}{c} \Delta x \\ \Delta y \\ \Delta z \\ \mathbf{1} \end{array}
 \end{bmatrix}$$

Fonte: Olivi, L.R., 2016 [9].

A Figura 5 representa a matriz transformação homogênea de uma manipulador em qualquer parte do sistema robótico que se queira. Tem-se que os elementos destacados nas três primeiras colunas da matriz de transformação representam a configuração dos eixo x, y e z, respectivamente, os quais representam o *frame* do efetuador. Já os elementos da quarta coluna representam a posição espacial cartesiana da origem do sistema de coordenadas do efetuador [1], [9].

2.5 CINEMATICA INVERSA

A cinemática inversa tem como objetivo a determinação das configurações dos parâmetros das juntas de um manipulador que geram a configuração espacial cartesiana desejada para o mesmo [7]. Sendo assim, pode-se interpretar que trata-se do procedimento inverso ao da cinemática direta.

Este tipo de abordagem trata das principais aplicações dadas aos manipuladores robóticos, uma vez que as configurações e trajetórias cartesianas desejadas para o manipulador são conhecidos, ou passíveis de determinação, como as aplicações industriais de montagem de veículos, soldagem industrial, dentre outros [10].

A solução do problema da cinemática inversa possui diversas abordagens. Uma das abordagens trata de métodos analíticos na obtenção das soluções, embora atualmente estas sejam conhecidas para uma pequena e específica quantidade de robôs manipuladores. Para a maior parte dos manipuladores, este tipo de solução não tem descrição na literatura [1].

Outra solução provém dos métodos numéricos, os quais, essencialmente, são métodos de otimização iterativos que convergem para uma solução possível de configuração das juntas, atendendo aos requisitos do sistema. Esses métodos demandam de uma boa capacidade de processamento, o que com os computadores atuais os tornam um meio viável de solução da cinemática inversa [7], [8].

Muitos desses métodos computacionais baseados em gradientes tem como essência a teoria apresentada no método dos mínimos quadrados, desenvolvido por Carl Friedrich Gauss, e dentre eles pode-se citar o Método de Newton-Raphson e o Método do Gradiente Descendente [7], que foi utilizado neste trabalho [9].

O Método do Gradiente Descendente é um método de otimização de funções objetivos. Métodos de otimização podem ser de maximização ou de minimização de resultados [7], como se quer saber quais valores de junta leva o manipulador a um determinado ponto, este método aplicado a este trabalho é um método de minimização.

A função objetivo é uma função matemática, que pode ser de qualquer tipo, em que dentro dela se espera obter a solução para o problema, no caso dos problemas de manipuladores, essa função tem como variável os valores de junta e ela é uma função de erro quadrático médio, pois possui um único ponto de mínimo, também chamado de mínimo global [7].

O procedimento deste método, consiste em calcular, de forma iterativa, o gradiente da função objetivo e verificar se o novo valor de erro satisfaz a condição desejada. Entretanto, o gradiente aplicado a um ponto da função objetivo vai nos direcionar para o máximo crescimento da mesma, o que não é o objetivo, logo utiliza-se o oposto do gradiente, pois assim as iterações caminharão para o ponto de mínimo. Desta forma, serão realizadas quantas iterações necessárias até chegar ao ponto objetivo ou até uma região ζ em que se considera que o objetivo foi alcançado [7], [9]. O pseudo-algoritmo do método do gradiente descendente é mostrado na Figura 6.

Figura 6 – Pseudo-algoritmo do método do gradiente descendente.

Seja $f(x)$, com $x = [x_1, x_2, \dots, x_N]^T$
Defina um valor para η
Inicie x_0 (exemplo: $x_0 = \text{randn}$)
Calcule o primeiro erro: $e_0 = f(x_0)$
enquanto $e_{k+1} > \xi$ faça:
 Calcule o Gradiente em x_k : $\nabla f(x_k)$
 Atualize o ponto: $x_{k+1} = x_k - \eta \nabla f(x_k)$
 Calcule o erro: $e_{k+1} = f(x_{k+1})$
fim do enquanto

Fonte: Olivi, L.R., 2016 [9].

2.6 INDÚSTRIA 4.0

O conceito de indústria 4.0 surgiu pela primeira vez na Feira de Hannover de 2011, na Alemanha, e este conceito propõe que toda a indústria, incluindo todos os setores como os módulos da produção, máquinas e equipamento, o corpo administrativo, produtos e informações, tudo, por meio de um conjunto de tecnologias, esteja conectada entre si, trocando informações e processando estas informações de forma autônoma e descentralizada [10].

Portanto, a indústria, como existe atualmente, passa a não mais ser um conjunto de processos mecânicos e automatizados, a indústria 4.0 passa a ser uma empresa muito além do espaço físico que ela está instalada e que tem a capacidade de interpretar os fatos e tomar decisões, de uma maneira que leve à otimização da sua cadeia produtiva e incorporá-la à Internet das Coisas [10], [11].

Dentre as tecnologias as tecnologias que permitem essa transformação na indústria destacam-se a Inteligência Artificial, a Internet das Coisas (do inglês, *Internet of Things*), *Big Data* e os sistemas Ciber-Físicos [11], [12].

A inteligência artificial é um segmento da computação que busca simular a capacidade humana de raciocinar, tomar decisões, resolver problemas, dotando algoritmos e os robôs de uma capacidade de aprendizado com os próprios erros [11].

A Internet das Coisas é um sistema em que aparelhos, como por exemplo relógios, termômetros, cafeteiras, cortinas, estão conectados à internet, trocando informações para que

seja possível tomadas de decisões corretas de acordo com o estado ou situação a que os dispositivos conectados apresentam [11], [12].

Big Data são bancos de dados de grande capacidade de armazenamento, muitas vezes recorrendo a nuvem (do inglês, *Cloud*), e que armazenam desde informações simples até as mais detalhadas, oferecendo a oportunidade desses dados serem relacionados, analisados e gerenciados de maneira personalizada [13], [14].

Sistemas Ciber-Físicos são a digitalização dos equipamentos e processos físicos. Em outras palavras, é ter um replica digital de uma fábrica ou de uma parte da linha de produção [12].

Segundo o levantamento da ABDI, a estimativa anual de redução de custos industriais no Brasil, a partir da migração da indústria para o conceito 4.0, será de, no mínimo, R\$ 73 bilhões/ano. Essa economia envolve ganhos de eficiência, redução nos custos de manutenção de máquinas e consumo de energia [12].

Em um mercado cada vez mais exigente, muitas empresas já procuram integrar ao produto necessidades e preferências específicas de cada cliente. A customização previa do produto por parte dos consumidores tende a ser uma variável a mais no processo de manufatura, mas as fábricas inteligentes serão capazes de levar a personalização de cada cliente em consideração, se adaptando as preferências [13].

2.7 V-REP

O V-REP é um simulador voltado para a área de robótica que fornece uma grande quantidade de robôs, em tamanho real, e esses robôs podem ser operados internamente através de algoritmos escritos na linguagem de programação padrão do simulador, a LUA, mas também é possível controlar os robôs externamente através de outras linguagem mais conhecidas como MATLAB [5], C/C++, Python ou Java [6].

O ambiente de trabalho do simulador é didático e o simulador dispõe de uma grande quantidade de objetos, equipamentos e robôs já montados e configurados, que juntos podem formar cenários e situações que simulam atividades reais. Além disso, caso a biblioteca não tenha o robô desejado ou uma superfície específica, é possível criá-las no próprio simulador [6].

Através do V-REP, também é possível acoplar diferentes tipos de ferramentas ou garras no efetuador dos manipuladores robóticos, acoplar um manipulador em cima de um robô móvel, como o robô diferencial, e além disso também existem sensores como câmeras e sensores de distância que podem ser aplicados aos robôs [6].

O simulador também fornece a opção de quatro mecanismos que simulam a dinâmica dos objetos presentes na cena, sendo eles o *Bullet Physics Library*, o *Open Dynamics Engine* (ODE), *Vortex Dynamics* e o *Newton Dynamics*, desta forma é possível visualizar como acontece a interação dos robôs com os objetos [6].

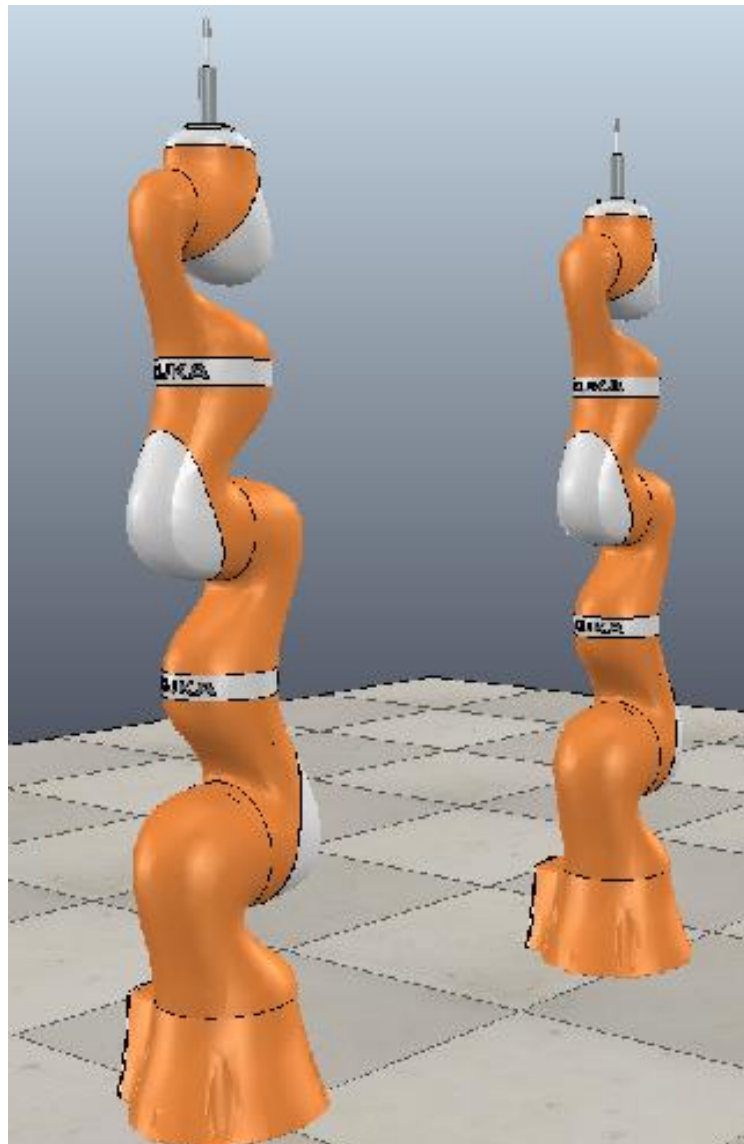
3 DESENVOLVIMENTO

Neste capítulo serão apresentados os métodos utilizados para a solução do problema proposto, aplicados aos manipuladores robóticos.

3.1 V-REP

O simulador V-REP foi utilizado para a construção da cena, onde foram inseridos dois manipuladores KUKA LBR iiwa 14 R820, e foi acoplado uma furadeira aos seus respectivos efetuadores, como mostra a Figura 7.

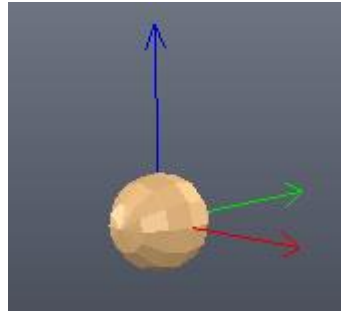
Figura 7 – Manipuladores com as furadeiras.



Fonte: Autor .

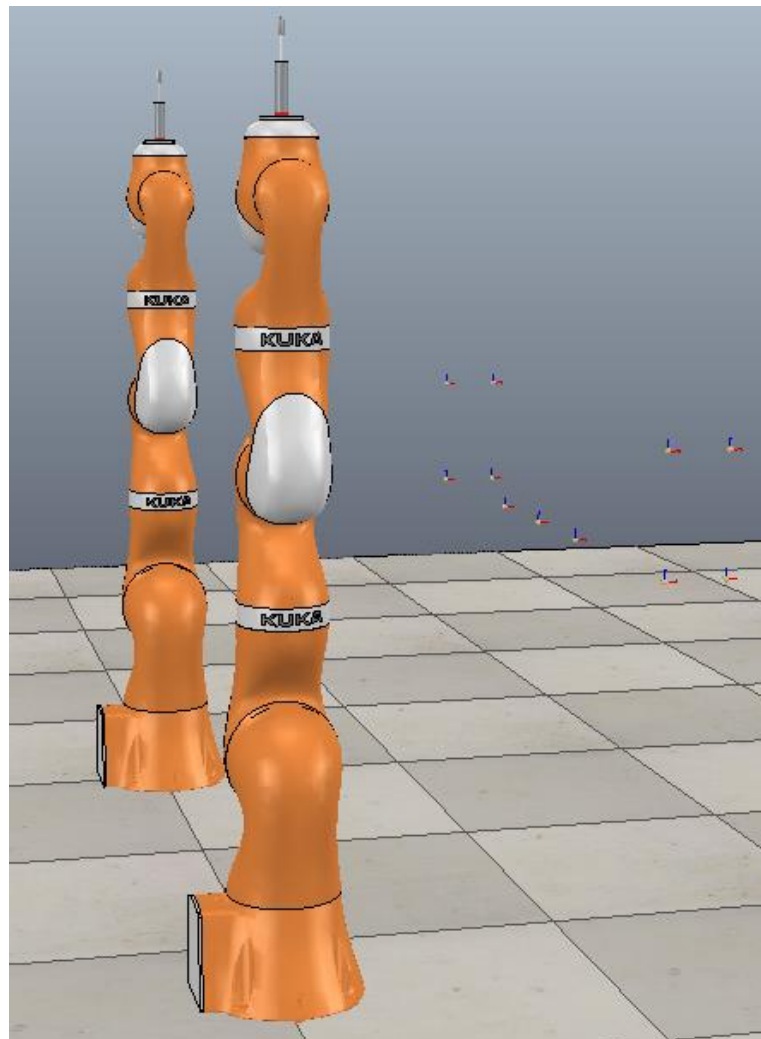
Os pontos geométricos que representam os pontos objetivos a serem atingidos pela ponta das furadeiras são representados por elementos denominados por *Dummy*. A Figura 8 mostra a representação desse elemento, juntamente com o seu próprio sistemas de coordenadas. A representação da cena final é mostrada na Figura 9.

Figura 8 – *Dummy*.



Fonte: Autor.

Figura 9 – Cena final, com manipuladores e pontos a serem alcançados.



Fonte: Autor.

Vale ressaltar que, para a comunicação entre o MATLAB e o V-REP ocorrer, é necessário que algum elemento da cena tenha em seu *script* a seguinte linha comando “simExtRemoteApiStart(19999)” como mostra a Figura 10. Para tal, o elemento escolhido foi um cubo. Outro procedimento que deve ser adotado para que a comunicação ocorra com sucesso, é que os arquivos “remApi.m”, “remoteApi.dll” e “remoteApiProto”, todos presentes no diretório do V-REP, devem estar na mesma pasta do programa principal da simulação.

Figura 10 – Função no *script* do cubo.

```

3 if (sim_call_type==sim_childscriptcall_initialization) then
4   simExtRemoteApiStart(19999)
5   -- Put some initialization code here
6
7   -- Make sure you read the section on "Accessing general-type objects programmatically"
8   -- For instance, if you wish to retrieve the handle of a scene object, use following ins

```

Fonte: Autor.

3.2 CINEMÁTICA DIRETA

O método utilizado para descrever a cadeia cinemática de ambos os robôs foi a convenção de Denavit-Hartenberg [1]. A orientação dos sistemas de coordenadas de cada junta que compõe o manipulador foi obtida diretamente do simulador, esta informação é obtida selecionando a junta que se deseja obter o sistema de coordenada.

As Tabelas 1 e 2 mostram a tabela de Denavit-Hartenberg e as dimensões do braço robótico, respectivamente. As dimensões também foram obtidas diretamente do simulador.

Tabela 1 – Tabela de Denavit-Hartenberg para o robô KUKA LBR iiwa 14 R820

$A_{\{i\}}^{\{i+1\}}$	d_i	θ_i	a_i	α_i
$A_{1\{0\}}^{\{1\}}$	d_1	$\theta_1^* + \pi$	0	+90
$A_{2\{1\}}^{\{2\}}$	0	$\theta_2^* + \pi$	0	-90
$A_{3\{2\}}^{\{3\}}$	d_2	$\theta_3^* + \pi$	0	-90
$A_{4\{3\}}^{\{4\}}$	0	$\theta_4^* + \pi$	0	+90
$A_{5\{4\}}^{\{5\}}$	d_3	$\theta_5^* + \pi$	0	+90
$A_{6\{5\}}^{\{6\}}$	0	$\theta_6^* + \pi$	0	0
$A_{7\{6\}}^{\{7\}}$	d_4	θ_7^*	0	0

Fonte: Autor

Tabela 2 – Dimensões do robô KUKA LBR iiwa 14 R820

d_i	Dimensão [m]
d_1	0,360
d_2	0,420
d_3	0,400
d_4	0,245

Fonte: Autor

3.3 CINEMÁTICA INVERSA

O problema da cinemática inversa, ou seja, obter as variáveis de juntas que correspondem a um determinado ponto geométrico e a orientação com que se deseja chegar ao mesmo foi solucionado utilizando o método do gradiente descendente [8], [9].

3.3.1 Função objetivo

O método do gradiente descendente é um método de otimização de funções objetivos. Desta forma, a função objetivo escolhida foi a de Erro Quadrático Médio (EQM), visto que a mesma possui apenas um ponto de mínimo. Como se deseja resolver um problema de cinemática inversa de posição e orientação, a equação do erro quadrático médio é descrita como mostra a Equação (10):

$$EQM = \frac{(g - y)^2}{2} \quad (10)$$

em que g é uma sequência dos $i = 1, \dots, n$ sistemas de coordenadas que descrevem posição e orientação dos pontos geométricos objetivos e y é o sistema de coordenadas que descreve a posição e a orientação do efetuador, obtido por Denavit-Hartenberg [9]. Cada um dos termos é dado por:

$$g_i = \begin{bmatrix} G_{x,i}^{(x)} & G_{x,i+1}^{(x)} & G_{x,n}^{(x)} \\ G_{y,i}^{(x)} & G_{y,i+1}^{(x)} & G_{y,n}^{(x)} \\ G_{z,i}^{(x)} & G_{z,i+1}^{(x)} & G_{z,n}^{(x)} \\ G_{x,i}^{(y)} & G_{x,i+1}^{(y)} & G_{x,n}^{(y)} \\ G_{y,i}^{(y)} & G_{y,i+1}^{(y)} & G_{y,n}^{(y)} \\ G_{z,i}^{(y)} & G_{z,i+1}^{(y)} & G_{z,n}^{(y)} \\ G_{x,i}^{(z)} & G_{x,i+1}^{(z)} & G_{x,n}^{(z)} \\ G_{y,i}^{(z)} & G_{y,i+1}^{(z)} & G_{y,n}^{(z)} \\ G_{z,i}^{(z)} & G_{z,i+1}^{(z)} & G_{z,n}^{(z)} \\ \Delta g_{x,i} & \Delta g_{x,i+1} & \Delta g_{x,n} \\ \Delta g_{y,i} & \Delta g_{y,i+1} & \Delta g_{y,n} \\ \Delta g_{z,i} & \Delta g_{z,i+1} & \Delta g_{z,n} \end{bmatrix} ; \quad y = \begin{bmatrix} Y_x^{(x)} \\ Y_y^{(x)} \\ Y_z^{(x)} \\ Y_x^{(y)} \\ Y_y^{(y)} \\ Y_z^{(y)} \\ Y_x^{(z)} \\ Y_y^{(z)} \\ Y_z^{(z)} \\ \Delta y_x \\ \Delta y_y \\ \Delta y_z \end{bmatrix} \quad (11)$$

3.3.2 Gradiente

Uma vez que se tem a equação do Erro Quadrático Médio (10), aplica-se o gradiente na mesma para se obter o sentido do máximo crescimento da função, conforme descrito na Equação (12).

$$\nabla EQM_i = -(G_i - f(Q)) = - \begin{bmatrix} G_{x,i}^{(x_i)} - Y_x^{(x)} \\ G_{y,i}^{(x_i)} - Y_y^{(x)} \\ G_{z,i}^{(x_i)} - Y_z^{(x)} \\ \vdots \\ G_{x,i}^{(z_i)} - Y_x^{(z)} \\ G_{y,i}^{(z_i)} - Y_y^{(z)} \\ G_{z,i}^{(z_i)} - Y_z^{(z)} \\ \Delta g_{x,i}^{(i)} - \Delta y_x \\ \Delta g_{y,i}^{(i)} - \Delta y_y \\ \Delta g_{z,i}^{(i)} - \Delta y_z \end{bmatrix} \quad (12)$$

Entretanto, como o gradiente provê o sentido de máximo crescimento de uma função, e, por definição, deseja-se encontrar o ponto de mínimo de uma função de erro como o EQM. Assim, utiliza-se o sentido oposto ao do gradiente para a atualização dos parâmetros no método do Gradiente Descendente, visto que se deseja buscar o sentido do mínimo crescimento da função.

Entretanto, o fato do robô possuir um número diferente de graus de liberdade e coordenadas geométricas torna a matriz do gradiente não quadrada, não sendo possível utilizá-la diretamente na solução de um sistema linear. Para tanto, utiliza-se a solução dada por Carl Friedrich Gauss nos Mínimos Quadrados. No entanto, para a utilização dos Mínimos Quadrados, é necessária a obtenção de um sistema linearizado do robô, pois este é um sistema altamente não-linear. A linearização é dada por meio do Jacobiano do sistema de equações não-lineares $f(Q)$ que descrevem o robô, mostrada na Equação 13.

$$J(f(Q)) = \text{Jacobiano}(f(Q)) = \begin{bmatrix} \frac{\partial f_1}{\partial q_1} & \dots & \frac{\partial f_1}{\partial q_M} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial q_1} & \dots & \frac{\partial f_N}{\partial q_M} \end{bmatrix} \quad (13)$$

em que $f(Q)$ é o sistema de equações não-lineares obtidas por meio de Denavit-Hartenberg que descrevem a cinemática direta do robô, dadas em termos das variáveis de juntas Q .

Assim, a solução da atualização do espaço de juntas Q do robô, de maneira que o mesmo siga para a configuração desejada é dada por meio de Mínimos Quadrados conforme mostra a Equação (14).

$$\underbrace{\nabla Q_k}_{M \times 1} = \underbrace{\left(J(f(Q))^T \cdot J(f(Q)) \right)^{-1}}_{M \times M} \underbrace{J(f(Q))^T}_{M \times N} \underbrace{\nabla EQM_k}_{N \times 1} \quad (14)$$

em que ∇Q é o gradiente dado no espaço de juntas que deverá ser utilizado na atualização das variáveis juntas Q de modo a conduzir o robô ao objetivo, minimizando o EQM.

Desta forma, utilizando o Jacobiano, é possível obter o gradiente das variáveis de junta a partir do gradiente do erro quadrático médio. Portanto, a Equação (15) mostra o quanto se deve atualizar os valores das juntas para atingir um ponto objetivo por meio de Gradiente Descendente, ou seja, atualizando as juntas no sentido oposto ao do gradiente.

$$Q_{k+1} = Q_k - \eta \nabla Q_k \quad (15)$$

em que η é uma taxa de aprendizado, um valor de escala, pequeno, que determina o quanto se utiliza do gradiente na atualização do espaço de juntas Q .

3.4 SISTEMA SUPERVISÓRIO

Como existem dois manipuladores robóticos realizando movimento na mesma área de trabalho, foi necessário a elaboração de um sistema supervisório para verificar a distância entre os dois robôs, e desta forma, evitar uma possível colisão. Portanto, a todo momento o sistema supervisório calcula a posição de cada um dos manipuladores contidos no ambiente e, com isso, a proximidade entre cada uma de suas partes entre si e entre os outros manipuladores. O sistema supervisório elaborado, faz uma verificação em duas etapas.

Primeiramente, o sistema verifica quantas juntas de cada robô estão dentro da zona de perigo, definida pelo autor deste trabalho, como sendo uma proximidade entre os manipuladores de 60 centímetros ou menos. O primeiro manipulador, o manipulador 1, está posicionado em $x = 0$ e $y = 0$, o segundo, o manipulador 2, está posicionado em $x = 0$ e $y = -1,20\text{m}$. Considerando o comprimento total de cada um dos manipuladores, sendo 1,425 metros, foi determinado que a zona de perigo é $y = -0,3\text{m} \leq \text{zona de perigo} \leq y = -0,9\text{m}$.

A partir do momento em que mais de uma junta encontram-se dentro da zona de perigo, o sistema calcula constantemente a distância Euclidiana entre todas as juntas dentro da zona de perigo. Enquanto houver um par de juntas dentro da zona de perigo, a uma distância menor ou igual a 40 centímetros uma da outra, o supervisório mantém o manipulador 2 em seu percurso, para a ação do manipulador 1, e o mesmo só voltará ao seu trajeto, quando nenhuma junta do manipulador 2 estiver a uma distância menor ou igual a 40 centímetros de alguma junta do manipulador 1 dentro da zona de perigo.

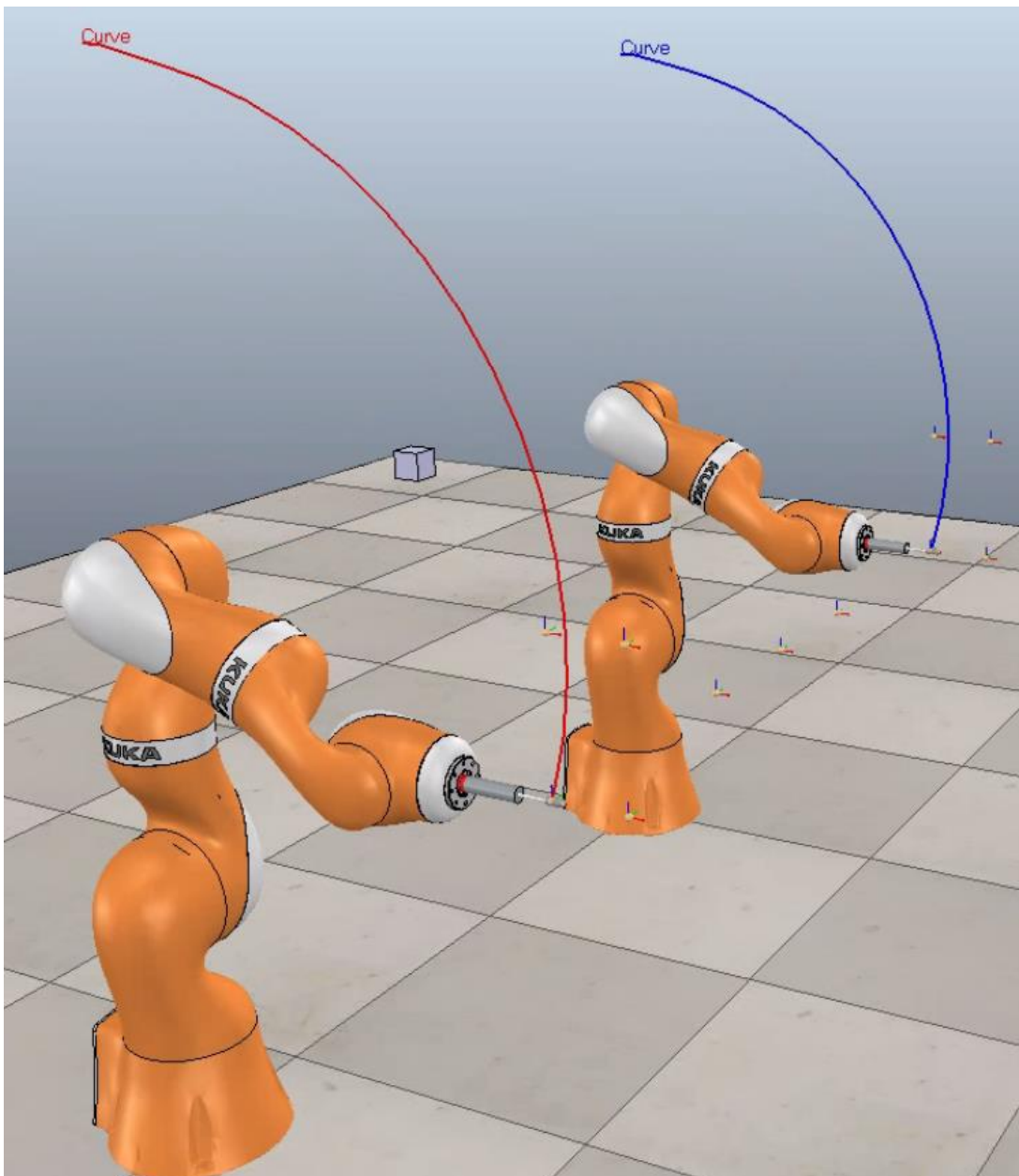
4 RESULTADOS

Este capítulo tem por objetivo mostrar qualitativamente e quantitativamente os resultados obtidos na simulação.

4.1 RESULTADOS DA TRAJETÓRIA

O trajeto percorrido pelos manipuladores robóticos é mostrado a seguir. A Figura 11 mostra o trajeto dos robôs até a primeira configuração desejada.

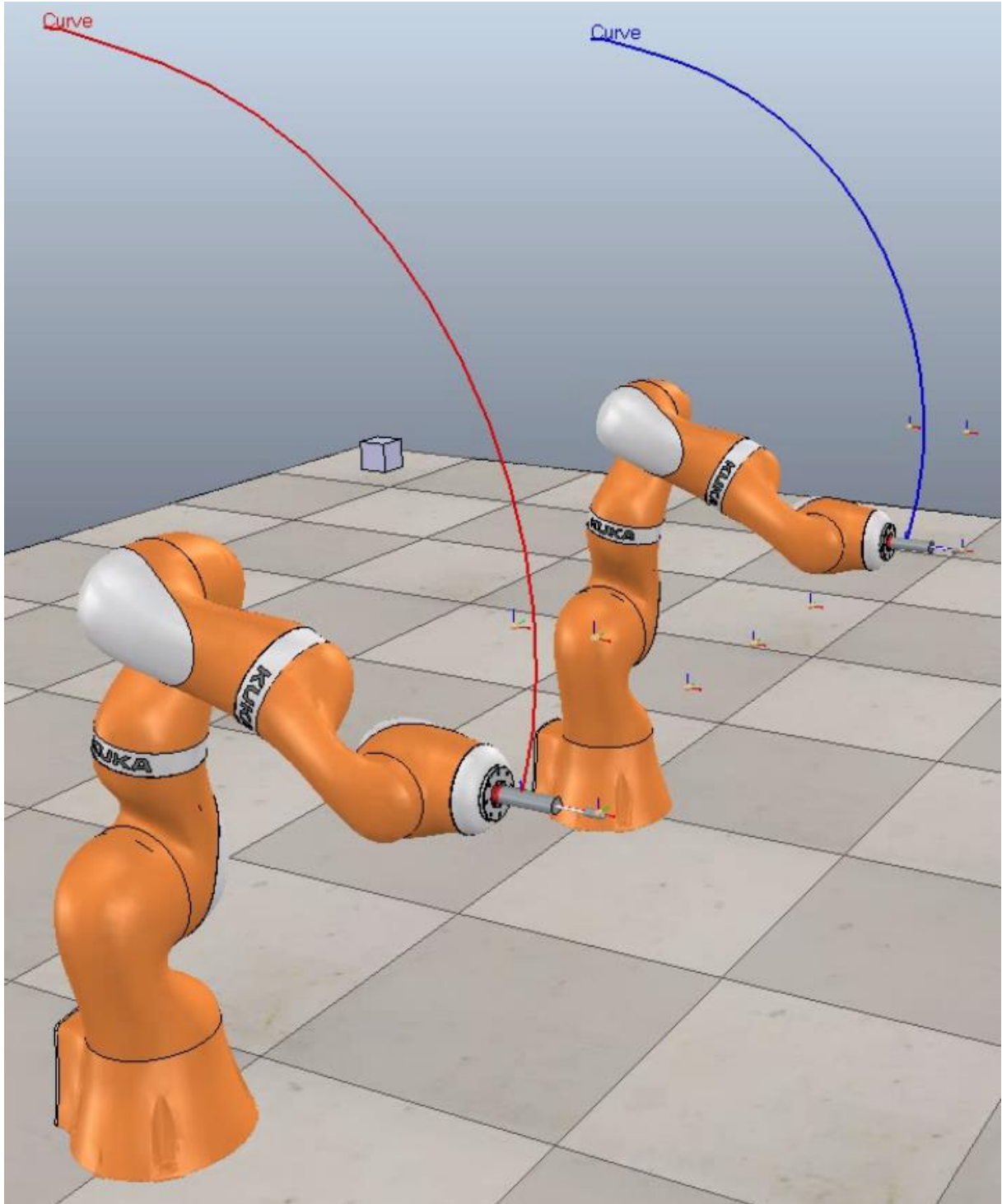
Figura 11 – Primeiro ponto objetivo



Fonte: Autor

A primeira configuração evidencia a aproximação do manipulador ao chassi em que o trabalho será realizado. A Figura 12 evidencia o trajeto do robô até a segunda configuração desejada, a qual indica, então a execução do trabalho de furação do chassi do automóvel.

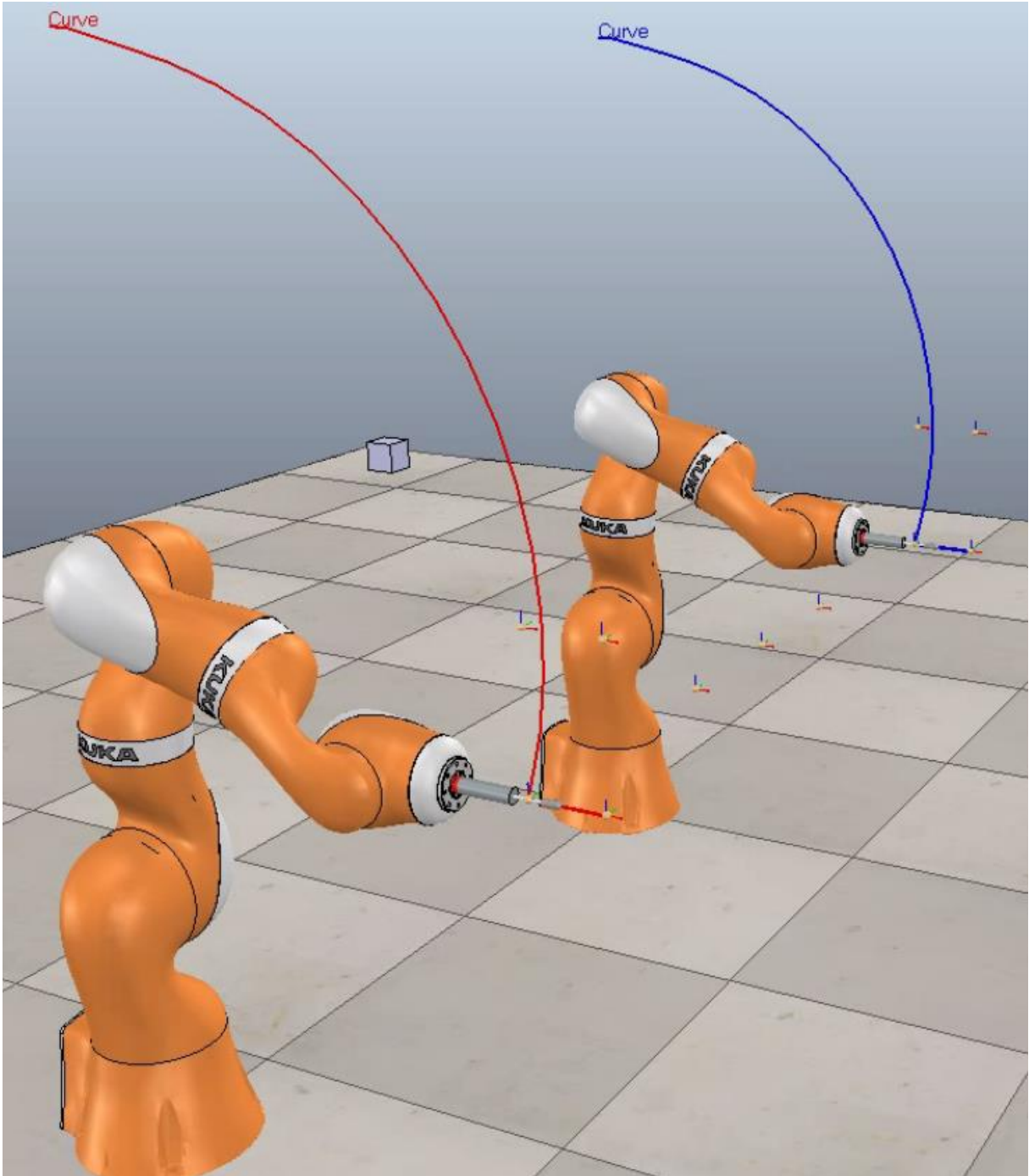
Figura 12 – Segundo ponto objetivo



Fonte: Autor

Terminado o trabalho de furação até a profundidade designada, o robô deve retirar a ferramenta do furo executado, retornando, portanto para a primeira configuração. Tal procedimento é evidenciado na Figura 13.

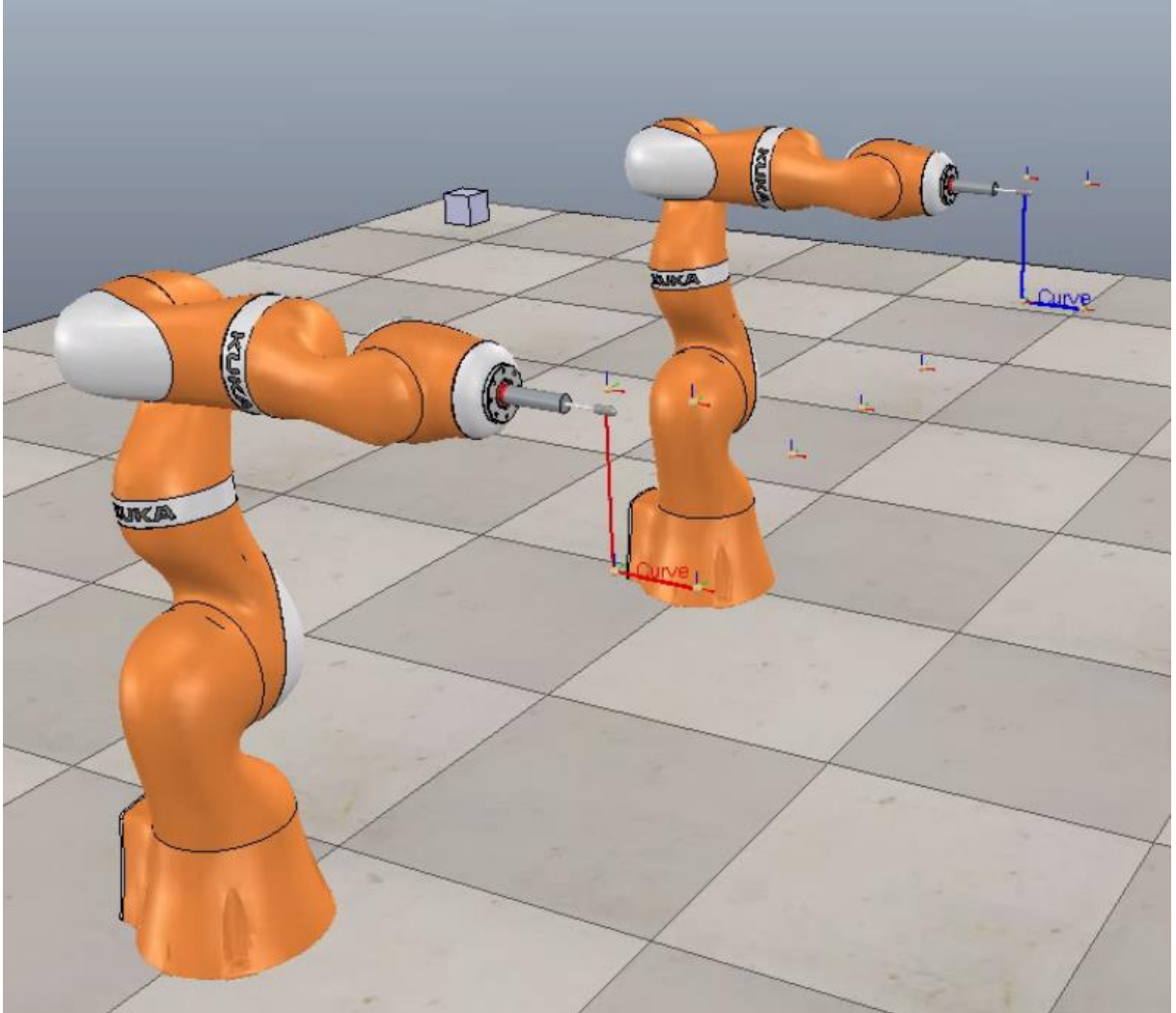
Figura 13 – Voltando para o primeiro ponto objetivo



Fonte: Autor

Em seguida, o robô parte para a terceira configuração programada, que designa a preparação para a execução do segundo furo com a ferramenta. Esta configuração é mostrada na Figura 14.

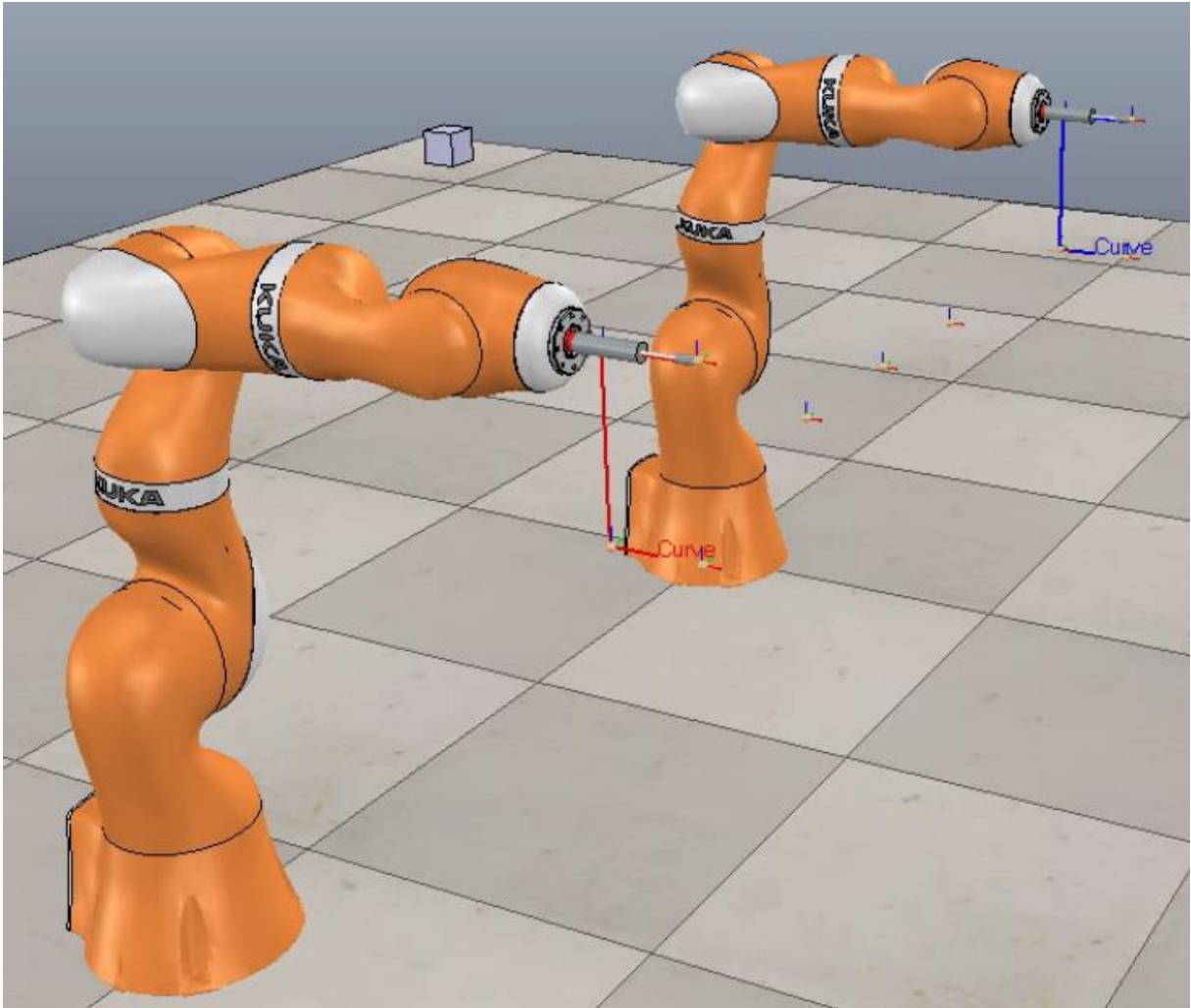
Figura 14 – Chegando ao terceiro ponto objetivo



Fonte: Autor

Uma vez que a terceira configuração é atingida, então o robô está preparado para executar a tarefa. O segundo furo então será executado por meio do robô alcançar a quarta configuração desejada, que levará a ferramenta até a profundidade desejada para o furo no chassi do automóvel. Este objetivo é mostrado na Figura 15.

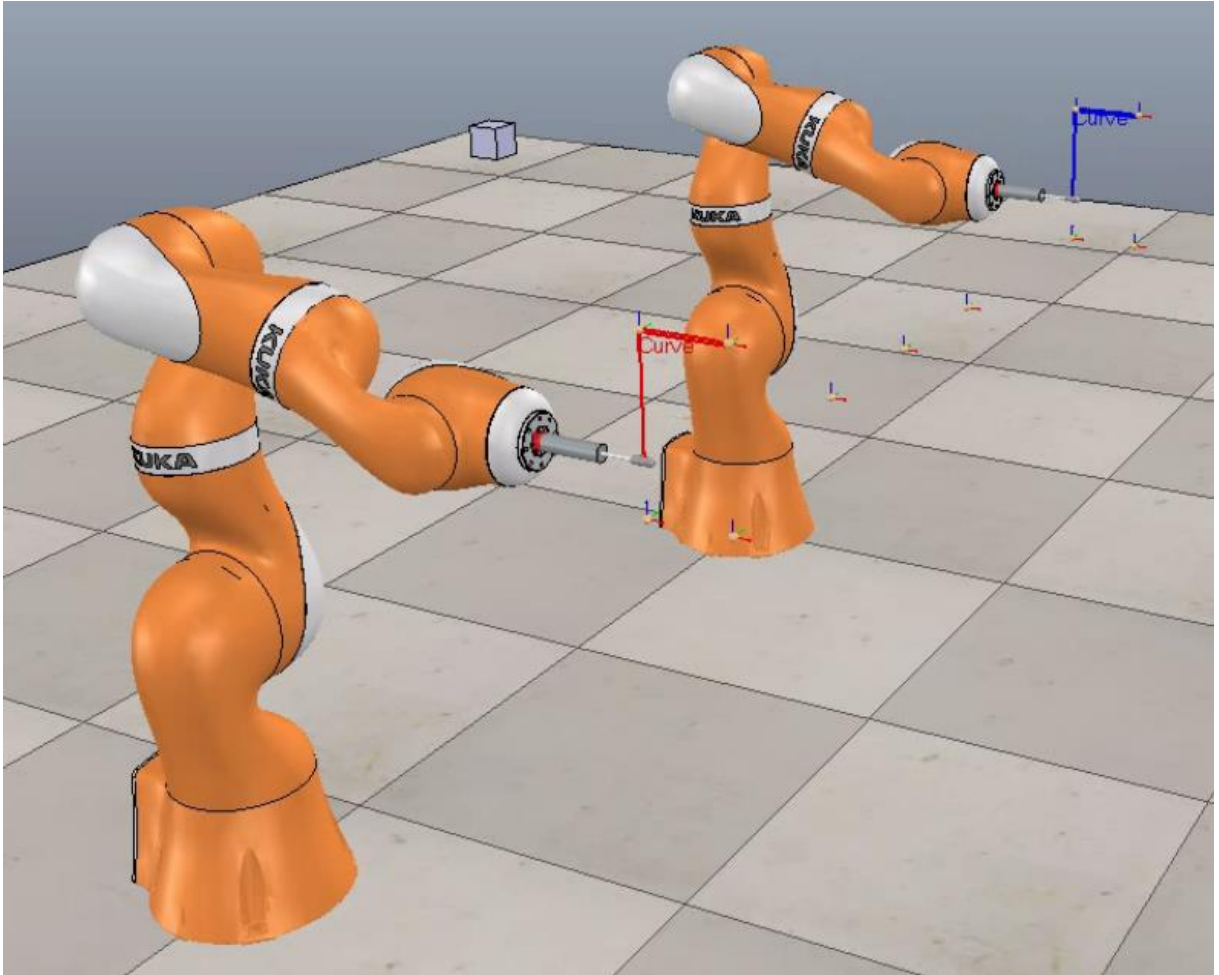
Figura 15 – Quarto ponto objetivo



Fonte: Autor

Finalmente, com o segundo furo executado, o robô pode retirar a ferramenta do furo, retornando-a para a configuração anterior. O procedimento é exibido na Figura 16.

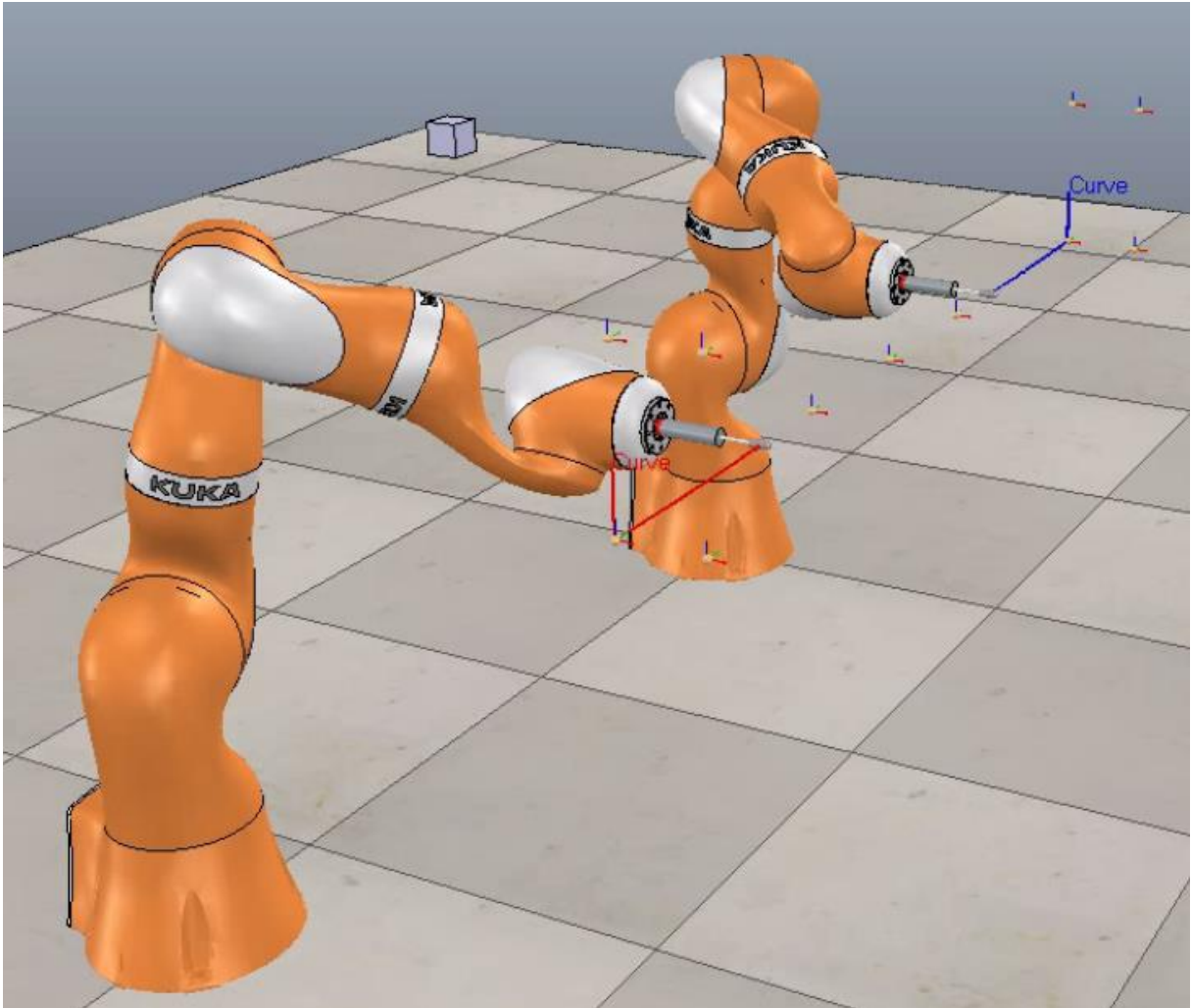
Figura 16 – Entre os pontos objetivos 3 e 1



Fonte: Autor

Assim, o robô pode continuar suas tarefas designadas. A próxima tarefa envolve configurações para os robôs que, ao serem alcançadas, os colocam na zona de perigo de colisão estabelecida e verificada pelo sistema supervisor. O direcionamento dos robôs para esta zona de perigo é mostrada na Figura 17.

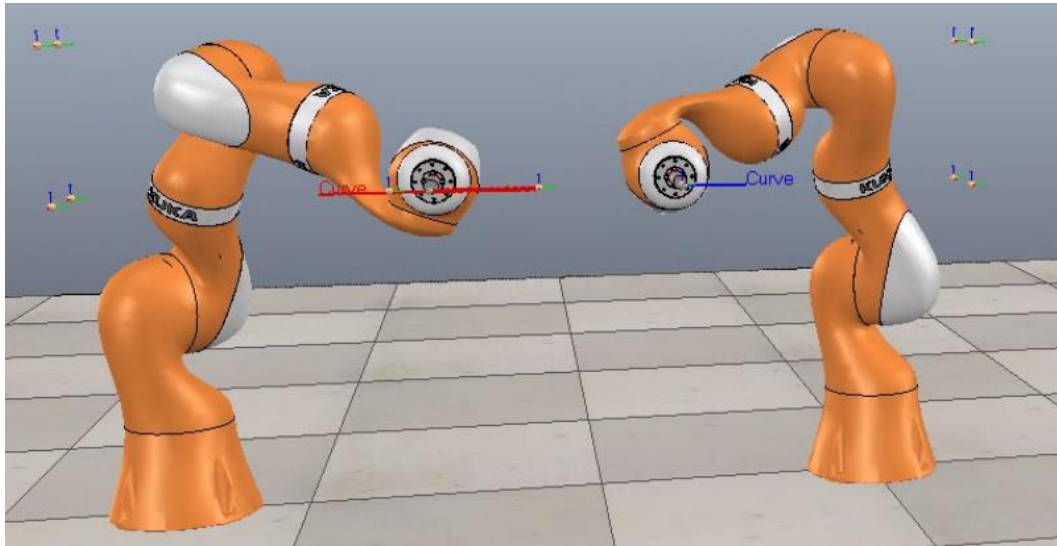
Figura 17 – Chegando na distância limite de 40 cm de distância



Fonte: Autor

O sistema supervisorio verifica continuamente a posição espacial de cada uma das partes dos robôs, de maneira a estabelecer as relações de proximidade entre eles. A política de contenção para situações de perigo implicam na parada de um dos manipuladores, que aguardará o outro desenvolver seu trabalho. Tal situação é mostrada na Figura 18.

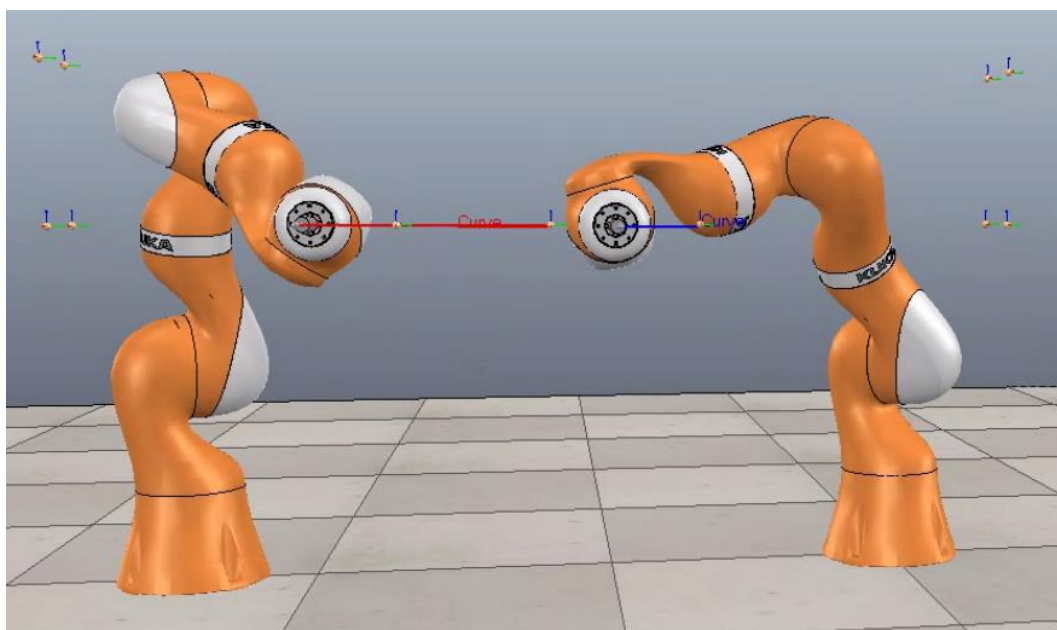
Figura 18 – Manipulador 1 parado pelo sistema supervisorio



Fonte: Autor

Uma vez que o manipulador que foi parado pelo sistema supervisorio se encontra em uma posição segura novamente, retorna às suas atividades, conforme a Figura 19.

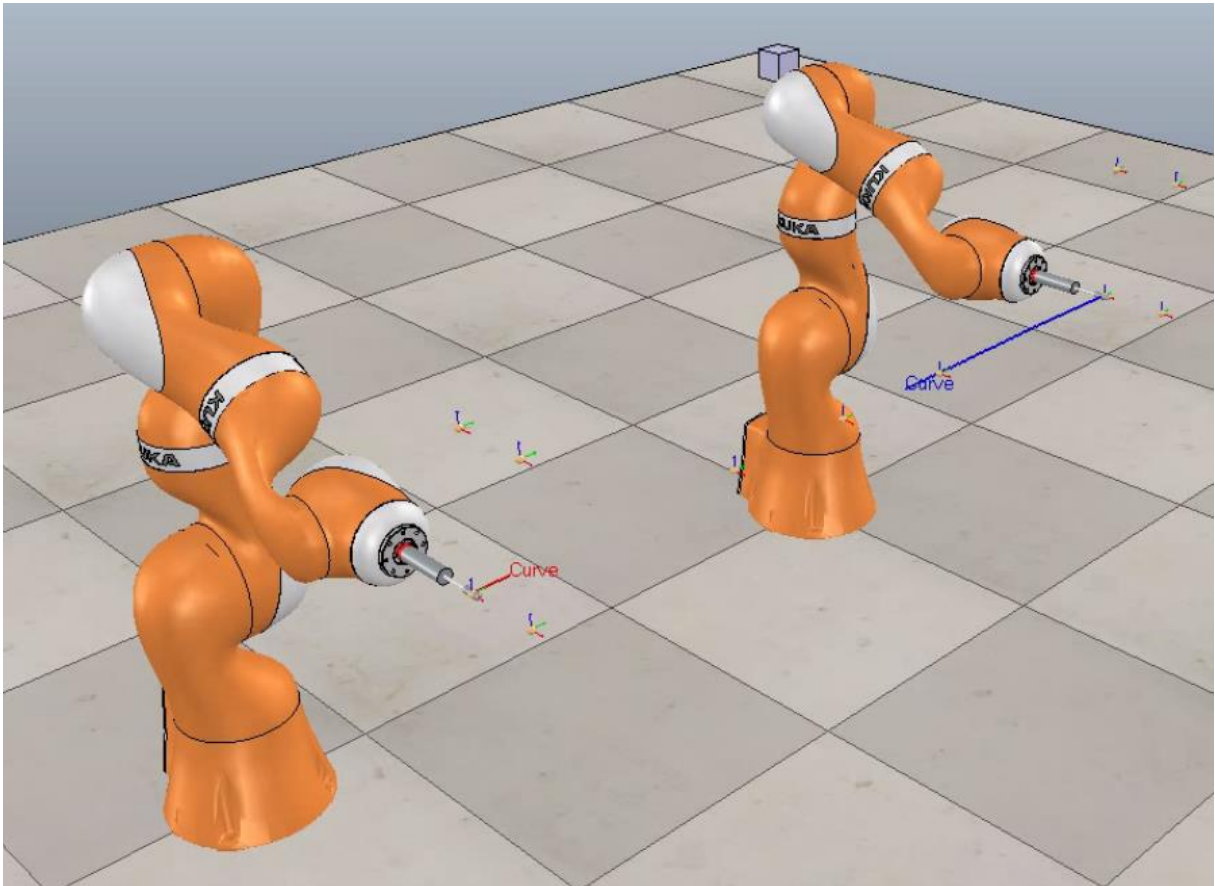
Figura 19 – Sistema supervisorio libera o manipulador 1



Fonte: Autor

Terminado o trabalho de cada um dos manipuladores, todos se dirigem às suas posições finais, encerrando o trabalho designado. A finalização é dada na Figura 20.

Figura 20 – Manipuladores na posição final



Fonte: Autor

A Tabela 3 a seguir, mostra o valor médio do erro em cada um dos três eixos cartesianos e também a acurácia média do robô durante todo o percurso.

Tabela 3 – Valor médio do erro

Erro	Erro médio [milímetros]
Eixo x	1,74 mm
Eixo y	1,40 mm
Eixo z	1,90 mm

Fonte: Autor

Com um erro médio abaixo de 2 milímetros em cada uma das atividades desenvolvidas, tem-se que a qualidade do serviço prestado pelos robôs é satisfatória.

5 CONCLUSÃO

Os resultados obtidos são satisfatórios visto que ao percorrer o trajeto, atingindo em sequência os pontos objetivos, os manipuladores possuem uma alta acurácia, com erro médio menor que 2 mm para cada eixo, e este resultado ainda pode ser melhorado, dependendo da resolução dos sensores tipo *encoder* utilizados para a medição das posições de cada uma das juntas. Assim, basta diminuir o limiar para o qual se considera que o manipulador atingiu o ponto objetivo, visto que o valor desse limiar, dado por ζ , usado para se obter os resultados deste trabalho foi de 2 mm. No entanto, a diminuição do erro resultará em um maior tempo para o método do gradiente descendente convergir.

Como trabalhos futuros, pode ser feito um aprimoramento do supervisório para que além da segurança dada pela zona de perigo, que o supervisório impeça que os robôs se aproximem demais uns dos outros dentro da zona de perigo, podendo o sistema supervisório parar ambos os manipuladores ou então recuando um deles mantendo a distância fixa entre ambos.

Outro ponto interessante, é fazer um planejamento de trajetória, para que os manipuladores tenham alguns perfis de velocidade e aceleração durante o trajeto, e não somente velocidade constante como foi realizado neste trabalho, visto que em alguns casos, o robô precisa variar sua velocidade e aceleração para se aproximar ou se afastar de um ponto objetivo, seja por segurança ou por otimização de tempo. Bem como o controle de torque durante a atividade realizada pela ferramenta, evitando que a mesma seja submetida a uma força e toque maior que pode suportar.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ROSÁRIO, J. M.; “*Princípios de Mecatrônica*”. Pearson, 2004.
- [2] SILVA, E.B.; SCOTON, M.L.R.P.D; DIAS, E.M.; “*Automação & Sociedade: Quarta Revolução Industrial, um olhar para o Brasil*”. Editora BRASPORT, 2018.
- [3] ROSÁRIO, J. M.; “*Automação Industrial*”. Editora Baraúna, 2012.
- [4] KUKA; “*The Power of Automation*”. Disponível em: <https://www.kuka.com/pt-br>. Acesso em: 17 jun 2018.
- [5] MATHWORKS; “*MATLAB*”. Disponível em: <https://www.mathworks.com/products/MATLAB.html>. Acesso em: 17 jun 2018.
- [6] COPPELIA ROBOTICS; “*V-REP: Virtual Experimentation Platform*”. Disponível em: <http://www.coppeliarobotics.com/>. Acesso em: 17 jun 2018.
- [7] SICILIANO, B.; SCIAVICCO, L.; VILLANI, L.; ORIOLO, G.; “*Robotics: Modelling, Planning and Control*”. Springer, 2009.
- [8] SPONG, M. W.; HUTCHINSON. S.; VIDYASAGAR. M.. “*Robot Modeling and Control*”. Editora John Wiley & Sons, 2005.
- [9] OLIVI, L. R.; *Notas de Aula da disciplina de Manipuladores Robóticos*, UFJF, 2016.
- [10] RIZZO, J.; “*Saiba o que é a Indústria 4.0 e descubra as oportunidades que ela gera*”. Disponível em: <http://www.sebrae.com.br/sites/PortalSebrae/artigos/saiba-o-que-e-a-industria-40-e-descubra-as-oportunidades-que-ela-gera,11e01bc9c86f8510VgnVCM1000004c00210aRCRD>. Acesso em: 18 jun 2018
- [11] GILCHRIST, A; “*Industry 4.0: The Industrial Internet of Things*”. Editora Apress, 2016.
- [12] ABDI; “*A Agenda Brasileira para a Indústria 4.0*”. Disponível em: <http://www.industria40.gov.br/>. Acesso em: 17 jun 2018.
- [13] SILVEIRA, C. B. “*O que é a Indústria 4.0 e Como Ela Vai Impactar o Mundo*”. Disponível em: <https://www.citisystems.com.br/industria-4-0/>. Acesso em: 17 jun 2018.
- [14] YÁÑEZ, F.; “*The Goal is Industry 4.0: Technologies and Trends of the Fourth Industrial Revolution*”. Amazon Digital Services LLC, 2017.