

UNIVERSIDADE FEDERAL DE SANTA CATARINA - UFSC CAMPUS BLUMENAU



Departamento de Engenharia

Professor: Leonardo Mejia Rincon

Trabalho 1 de Introdução à Robótica Industrial

Carlos Eduardo dos Santos Junior
Marco Antônio Priotto Tonon

Blumenau, Novembro de 2019

Sumário

1. Apresentação do Tema.....	2
2. Cálculo da cinemática direta e inversa.....	4
2.1 Cinemática direta utilizando os parâmetros de DH.....	6
2.2 Cinemática Inversa do manipulador.....	16
2.2 Cinemática Inversa utilizando um algoritmo de otimização.....	21
3. Cálculo da Trajetória.....	23
3.1 Introdução.....	23
3.2 Ângulos de Euler.....	23
3.3 Quatérnions.....	26
3.4 Movimentação da peça com os manipuladores.....	28
4. Critérios Escolhidos para distância L.....	30
5. Conclusão.....	32
6. Referências.....	33

Lista de Figuras

Figura 1 - UFSC-BOT.....	4
Figura 2 - YRB 14000 (Yumi).....	5
Figura 3 - Apresentação dos Braços.....	6
Figura 4 - Coordenadas Para o Cálculo de DH.....	7
Figura 5 - Parâmetros de D-H Braço Esquerdo.....	8
Figura 6 - Parâmetros de D-H Braço Direito.....	8
Figura 7 - Convenção de Denavit Hartenberg.....	8
Figura 8 - Matrizes do braço esquerdo I.....	12
Figura 9 - Matrizes do braço esquerdo II.....	13
Figura 10 - Matrizes do braço direito I.....	14
Figura 11 - Matrizes do braço direito II.....	15
Figura 12 - Matriz final braço esquerdo.....	16
Figura 13 - Matriz final braço direito.....	16
Figura 14 - Modelo de Articulação.....	18
Figura 15 - Modelo da articulação para encontrar θ_2 e θ_4	19
Figura 16 - Modelo da articulação para encontrar θ_6	20
Figura 17 - Modelo de conclusão devido aos ângulos achados.....	20
Figura 18 - Resposta no formato do MatLAB.....	22
Figura 19 - Ângulos de Euler braço esquerdo.....	22
Figura 20 - Ângulos de Euler braço direito.....	22
Figura 21 - Matriz de Transformação de Coordenadas.....	24
Figura 22 - Matriz de Euler.....	24
Figura 23 - Trajetória de Euler.....	25
Figura 24 - Trajetória de Euler II.....	25
Figura 25 - Representação de Quatérnions.....	26

Figura 26 - Matriz de Quatérnion.....	26
Figura 27 - Variáveis utilizadas para rotação dos eixos.....	26
Figura 28 - Trajetória Por Quaternions, Vista superior.....	27
Figura 29 - Trajetória em vista tridimensional com Quatérnions.....	27
Figura 30 - Manipuladores movimentando a caixa em torno do eixo X.....	28
Figura 31 - Manipuladores movimentando a caixa em torno do eixo Z.....	29
Figura 32 - Dimensões do Braço Yumi.....	30

1. Apresentação do Tema

Neste trabalho iremos abordar questões de Cinemática inversa, direta, ângulos de Euler, Quatérnions e etc, o trabalho aqui exposto contém algoritmos e cálculos que não serão totalmente explicados e detalhados, pois os mesmos possuem resoluções extensas não se inserindo no escopo que foi pedido.

Abaixo na Figura 1, está o esquemático dos braços propostos para análise e cálculos que iremos expor neste trabalho. Utilizaremos o Software MatLAB para aplicações, tais como multiplicação de matrizes, visualização de resultados e constatação de definições e teorias que aprendemos em sala de aula, caso haja interesse na execução dos arquivos e visualização de vídeos feitos para este trabalho, basta acessar o meu repositório da disciplina de robótica industrial no github, segue o link <<https://github.com/juninhocb/RoboticalIndustrial>>.

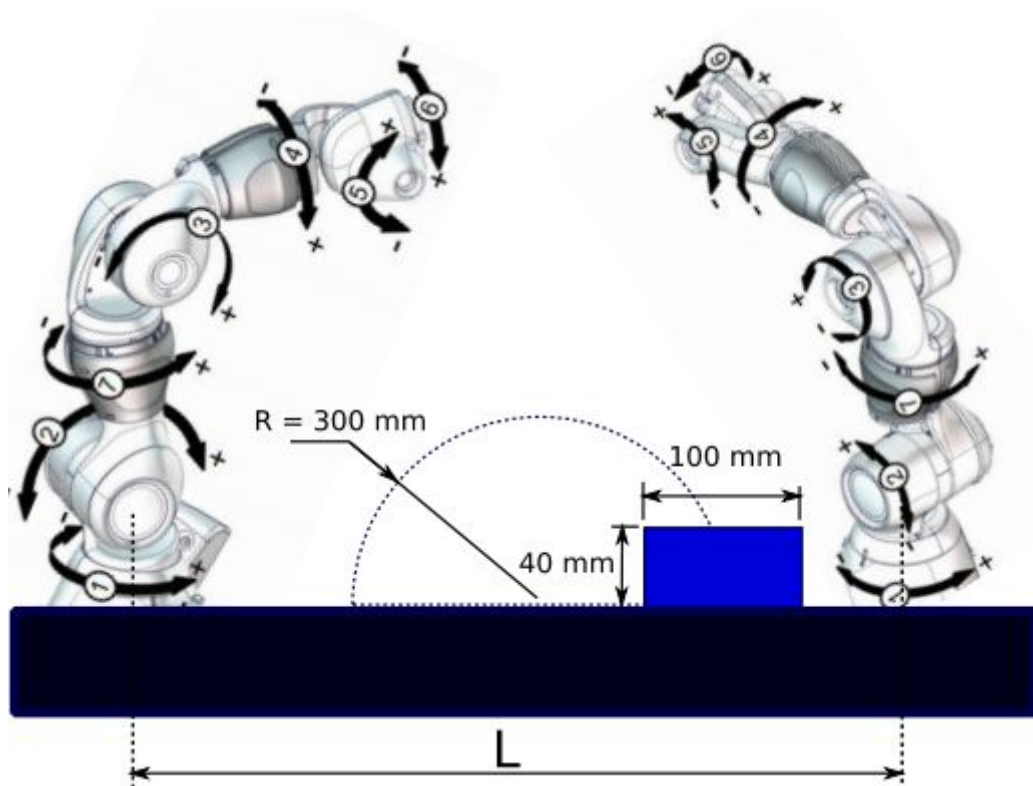


Figura 1: UFSC-BOT

Iremos então simular e realizar cálculos com base no robô comercial IRB 14000(Yumi), assim como utilizar seus elementos pré-fabricados respeitando suas dimensões originais, o Yumi será mostrado na figura abaixo.



Figura 2: YRB 14000 (Yumi).

O presente trabalho consiste em realizar cálculos e utilizar algoritmos com a finalidade de responder questões pedidas, no início do mesmo, iremos trabalhar e fazer cálculos utilizando os parâmetros de Denavit Hartenberg e de matrizes utilizando os conceitos de cinemática inversa e direta em nossos manipuladores a fim de representar as características construtivas, direção, orientação de nossos braços expostos nas Figura 1 e 2. Após a realização do excerto explicado acima, iremos utilizar a teoria de Ângulos de Euler e Quaternions para representar no espaço a trajetória de pontos, em uma abordagem que serve tanto para constatar suas veracidades como para compararmos ambas as soluções, então, segue-se com o orientações e avaliações a respeito de critérios utilizados em robótica para escolha da distância L como será explicada no tópico 4, por fim faremos uma conclusão do trabalho como um todo, tendo em vista algumas aplicações que podem ser utilizadas no campo da robótica.

2. Cálculo da cinemática direta e inversa

2.1 Cinemática Direta utilizando os parâmetros de DH

Primeiramente, foi-se definindo um eixo de referências para cada braço, utilizaremos neste presente trabalho as seguintes nomenclaturas: Braço esquerdo para o braço que se encontra a esquerda da imagem e Braço direito ao braço que se encontra a direita da imagem.

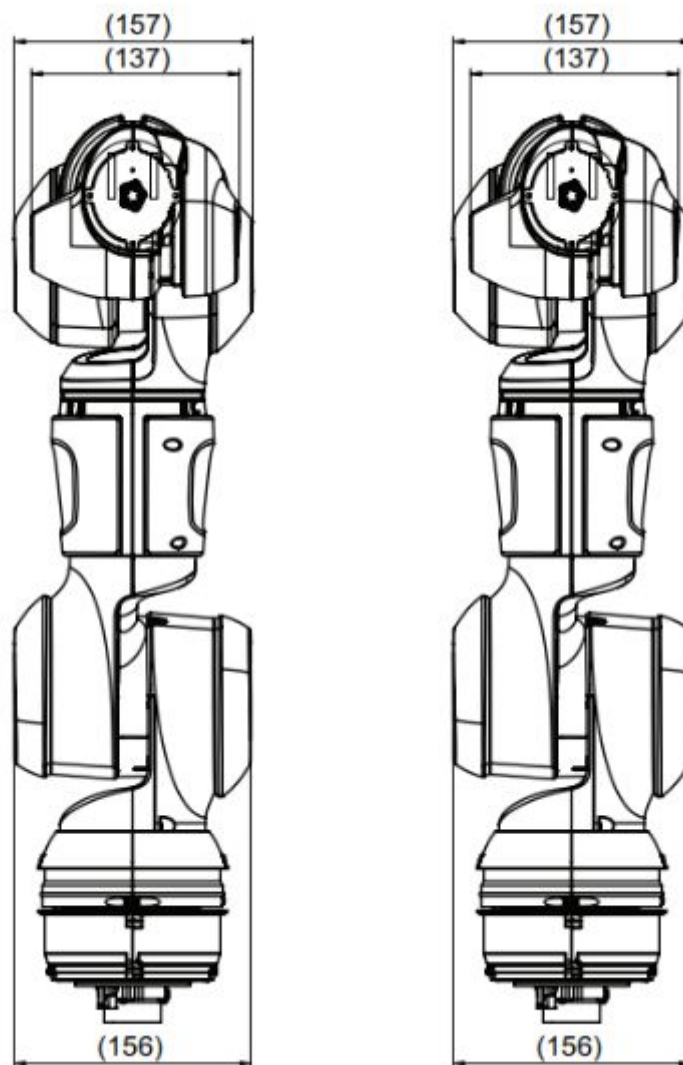


Figura 3: Apresentação dos braços, para identificação de nomenclatura

A partir das determinações das coordenadas X e Z utilizando a figura abaixo, onde cada eixo de coordenada deve ser perpendicular ao anterior e o último é uma "cópia" do penúltimo e assumindo, X0 até X5 e Z0 até Z5 de baixo para cima para ambos os braços, podemos enfim calcular os parâmetros de D-H.

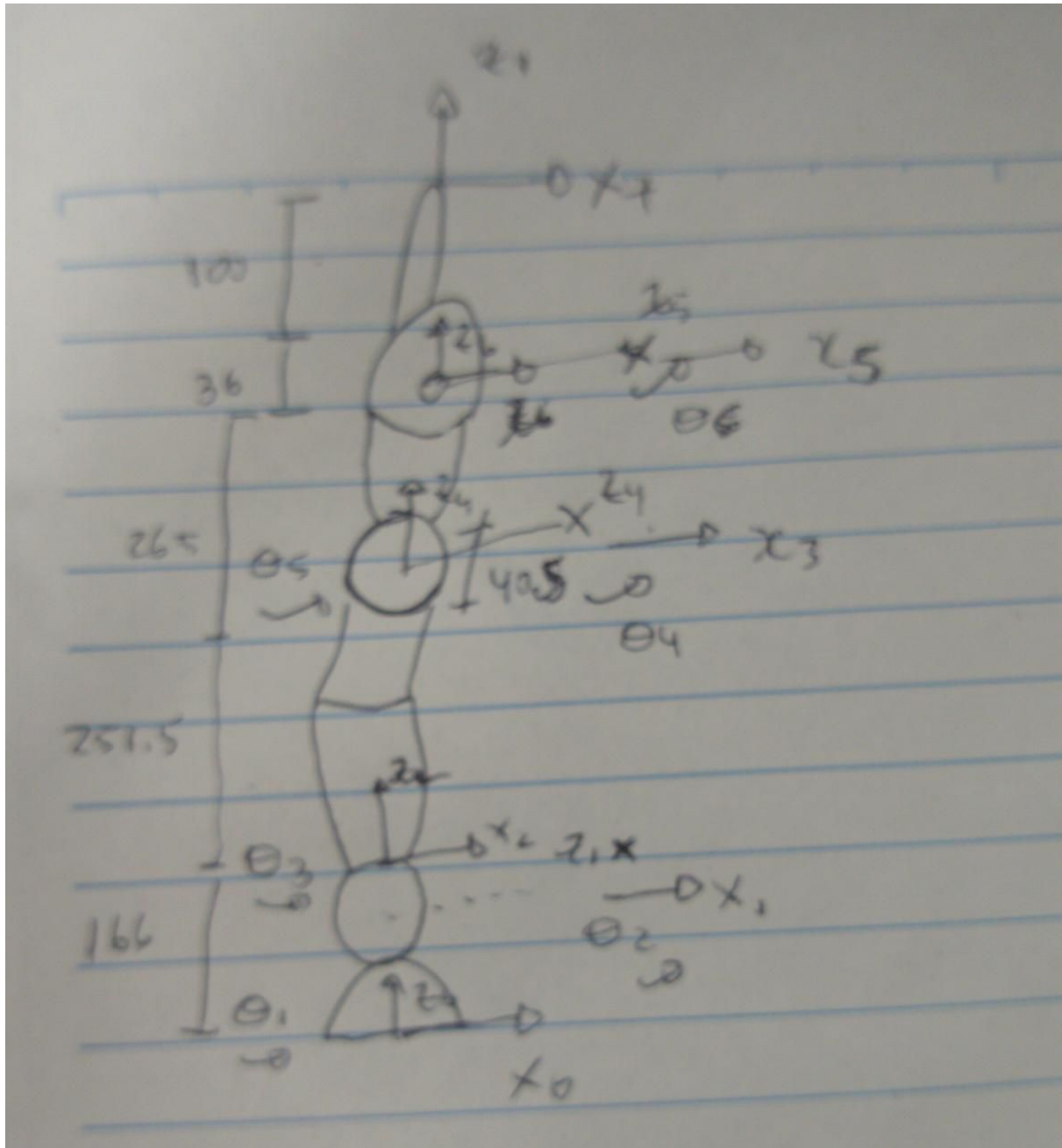


Figura 4: Coordenadas para o cálculo de DH

As referências são as mesmas tanto para o braço esquerdo como para o direito e os parâmetros de DH serão mostrados logo abaixo.

	Parâmetros de DH do Braço Esquerdo			
	θ	d	a	α
OT0	0	0	-325	0
OT1	θ_1	166	-30	90°
1T2	θ_2	0	30	-90°
2T3	θ_3	251.5	40.5	90°
3T4	θ_4	0	-40.5	-90°
4T5	θ_5	265	27	90°
5T6	θ_6	0	-27	-90°
6T7	θ_7	136	0	0

Figura 5: Parâmetros de DH Braço Esquerdo.

	Parâmetros de DH do Braço Direito			
	θ	d	a	α
OT0	0	0	325	0
OT1	θ_1	166	30	90°
1T2	θ_2	0	-30	-90°
2T3	θ_3	251.5	-40.5	90°
3T4	θ_4	0	40.5	-90°
4T5	θ_5	265	-27	90°
5T6	θ_6	0	27	-90°
6T7	θ_7	136	0	0

Figura 6: Parâmetros de DH Braço Direito.

Após acharmos os parâmetros, conseguimos descrever os 2 braços com as seguintes matrizes, de acordo com a convenção de Denavit Hartenberg mostrada na figura abaixo.

$$\begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 7: Convenção de Denavit Hartenberg.

Utilizando-se da matriz acima e fazendo as devidas substituições dos parâmetros que obtivemos, demonstrados nas Figuras 4 e 5 , obtivemos as matrizes de cada posição do nosso manipulador, demonstradas abaixo.

Matrizes Braço Esquerdo:

$$T_0 = \begin{vmatrix} 1 & 0 & 0 & -L/2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T_1 = \begin{vmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & -30\cos(\theta_1) \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & -30\sin(\theta_1) \\ 0 & 1 & 0 & 166 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T_2 = \begin{vmatrix} \cos(\theta_2) & 0 & -\sin(\theta_2) & 30\cos(\theta_2) \\ \sin(\theta_2) & 0 & \cos(\theta_2) & 30\sin(\theta_2) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T_3 = \begin{vmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) & 40.5\cos(\theta_3) \\ \sin(\theta_3) & 0 & -\cos(\theta_3) & 40.5\sin(\theta_3) \\ 0 & 1 & 0 & 251.5 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T_4 = \begin{vmatrix} \cos(\theta_4) & 0 & -\sin(\theta_4) & -40.5\cos(\theta_4) \\ \sin(\theta_4) & 0 & \cos(\theta_4) & -40.5\sin(\theta_4) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T5 = \begin{vmatrix} \cos(\theta_5) & 0 & \sin(\theta_5) & 27\cos(\theta_5) \\ \sin(\theta_5) & 0 & -\cos(\theta_5) & 27\sin(\theta_5) \\ 0 & 1 & 0 & 265 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T6 = \begin{vmatrix} \cos(\theta_6) & 0 & -\sin(\theta_6) & -27\cos(\theta_6) \\ \sin(\theta_6) & 0 & \cos(\theta_6) & -27\sin(\theta_6) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T7 = \begin{vmatrix} \cos(\theta_7) & -\sin(\theta_7) & 0 & 0 \\ \sin(\theta_7) & \cos(\theta_7) & 0 & 0 \\ 0 & 0 & 1 & 136 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Matrizes Braço Direito:

$$T0 = \begin{vmatrix} 1 & 0 & 0 & L/2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T1 = \begin{vmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & 30\cos(\theta_1) \\ \sin(\theta_1) & 0 & -\cos(\theta_1) & 30\sin(\theta_1) \\ 0 & 1 & 0 & 166 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T2 = \begin{vmatrix} \cos(\theta_2) & 0 & -\sin(\theta_2) & -30\cos(\theta_2) \\ \sin(\theta_2) & 0 & \cos(\theta_2) & -30\sin(\theta_2) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T3 = \begin{vmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) & -40.5\cos(\theta_3) \\ \sin(\theta_3) & 0 & -\cos(\theta_3) & -40.5\sin(\theta_3) \\ 0 & 1 & 0 & 251.5 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T4 = \begin{vmatrix} \cos(\theta_4) & 0 & -\sin(\theta_4) & 40.5\cos(\theta_4) \\ \sin(\theta_4) & 0 & \cos(\theta_4) & 40.5\sin(\theta_4) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T5 = \begin{vmatrix} \cos(\theta_5) & 0 & \sin(\theta_5) & -27\cos(\theta_5) \\ \sin(\theta_5) & 0 & -\cos(\theta_5) & -27\sin(\theta_5) \\ 0 & 1 & 0 & 265 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T6 = \begin{vmatrix} \cos(\theta_6) & 0 & -\sin(\theta_6) & 27\cos(\theta_6) \\ \sin(\theta_6) & 0 & \cos(\theta_6) & 27\sin(\theta_6) \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$T7 = \begin{vmatrix} \cos(\theta_7) & -\sin(\theta_7) & 0 & 0 \\ \sin(\theta_7) & \cos(\theta_7) & 0 & 0 \\ 0 & 0 & 1 & 136 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Substituindo os valores dos thetas(zeramos todos) e já utilizando o software do MatLAB, obteremos os seguintes resultados.

```
>> T0

T0 =

     1     0     0   -325
     0     1     0     0
     0     0     1     0
     0     0     0     1

>> T1

T1 =

     1     0     0   -30
     0     0    -1     0
     0     1     0   166
     0     0     0     1

>> T2

T2 =

     1     0     0    30
     0     0     1     0
     0    -1     0     0
     0     0     0     1

>> T3

T3 =

    1.0000         0         0   40.5000
         0         0   -1.0000         0
         0    1.0000         0   251.5000
         0         0         0    1.0000

>> T4

T4 =

    1.0000         0         0  -40.5000
         0         0    1.0000         0
         0   -1.0000         0         0
         0         0         0    1.0000
```

Figura 8: Matrizes Braço esquerdo I.

```
>> T5
```

```
T5 =
```

1	0	0	27
0	0	-1	0
0	1	0	265
0	0	0	1

```
>> T6
```

```
T6 =
```

1	0	0	-27
0	0	1	0
0	-1	0	0
0	0	0	1

```
>> T7
```

```
T7 =
```

1	0	0	0
0	1	0	0
0	0	1	136
0	0	0	1

Figura 9: Matrizes braço esquerdo II.

```

>> T0

T0 =

    1    0    0   325
    0    1    0    0
    0    0    1    0
    0    0    0    1

>> T1

T1 =

    1    0    0    30
    0    0   -1    0
    0    1    0   166
    0    0    0    1

>> T2

T2 =

    1    0    0   -30
    0    0    1    0
    0   -1    0    0
    0    0    0    1

>> T3

T3 =

    1.0000    0    0   -40.5000
         0    0   -1.0000         0
         0    1.0000    0   251.5000
         0    0    0     1.0000

```

Figura 10: Matrizes braço direito I.


```

>> T4

T4 =

    1.0000         0         0    40.5000
         0         0    1.0000         0
         0   -1.0000         0         0
         0         0         0    1.0000

>> T5

T5 =

     1     0     0   -27
     0     0    -1     0
     0     1     0   265
     0     0     0     1

>> T6

T6 =

     1     0     0    27
     0     0     1     0
     0    -1     0     0
     0     0     0     1

>> T7

T7 =

     1     0     0     0
     0     1     0     0
     0     0     1   136
     0     0     0     1

```

Figura 11: Matrizes braço direito II.

Após acharmos as matrizes de cada parte do braço, utilizamos o software MatLAB para multiplicá-las, tendo assim, um resultado final dos nossos manipuladores.

$${}^0T_7 = T_0 * T_1 * T_2 * T_3 * T_4 * T_5 * T_6 * T_7$$

Onde T é a nossa Transformação do 'ponto' anterior ao próximo, fazendo a multiplicação de matrizes iremos obter:

```
>> ParametrosDH_ESQ

T =

    1.0000    0    0 -325.0000
    0    1.0000    0    0
    0    0    1.0000  818.5000
    0    0    0    1.0000
```

Figura 12: Matriz final braço esquerdo.

```
>> ParametrosDH_DIR

T =

    1.0000    0    0  325.0000
    0    1.0000    0    0
    0    0    1.0000  818.5000
    0    0    0    1.0000
```

Figura 13: Matriz final braço direito.

Nosso resultado obtido, mostra-nos um resultado já esperado para o eixo Z, pois se somarmos as distâncias(L) que influenciam em nosso movimento em Z quando todos os thetas serão zerados, obteremos exatamente o valor de 818.5mm.

2.2 Cinemática Inversa do manipulador

Dado um manipulador robótico, podemos solucioná-lo de várias maneiras, como exposto no excerto acima, utilizamos os parâmetros de DH e a cinemática direta para conseguir obter uma fiel representação do mesmo em forma matricial no qual podemos simular e constatar sua veracidade e coerência utilizando o software MatLAB, a cinemática direta nos fornece a orientação e posição do efetuador, uma vez que conhecemos os valores de juntas, já a cinemática inversa, como o nome já nos informa, obtém os valores das juntas a partir de orientação e posição do efetuador, o que torna o problema significativamente mais difícil de resolver, pelas seguintes razões, segundo (SCIAVICCO; SICILIANO, 2004):

- As equações são geralmente não-lineares e muitas vezes não é possível encontrar uma solução fechada;
- Podem existir múltiplas soluções;

- Podem existir infinitas soluções (no caso de manipuladores redundantes);
- É possível que não haja solução em virtude da própria estrutura cinemática do manipulador.

Apesar de admitir soluções simples para cadeias cinemáticas simples, todavia usualmente as soluções são complexas e um meio que pode ser mais fácil de solucioná-las é utilizando a integração das equações de cinemática diferencial, tópico abordado para a segunda parte da disciplina ou por meio de técnica de programação não linear.

Para este trabalho utilizaremos o conceito de cinemática inversa em nossos manipuladores IBR 14000, e dispondo-se do fato do sistema ter 6 graus de liberdade e 7 juntas, é sabido que há redundâncias. com o intuito de facilitar nosso trabalho, travamos duas juntas que serão redundantes, são elas: θ_3 e θ_5 , fazendo com que todos os graus de liberdade sejam oriundas das juntas que são atuadas.

Primeiramente, elaboramos um modelo do sistema com base em nossa articulação, tomando como premissa que a articulação possui pontos alcançáveis limitados e espaço hábil no seu espaço de trabalho.

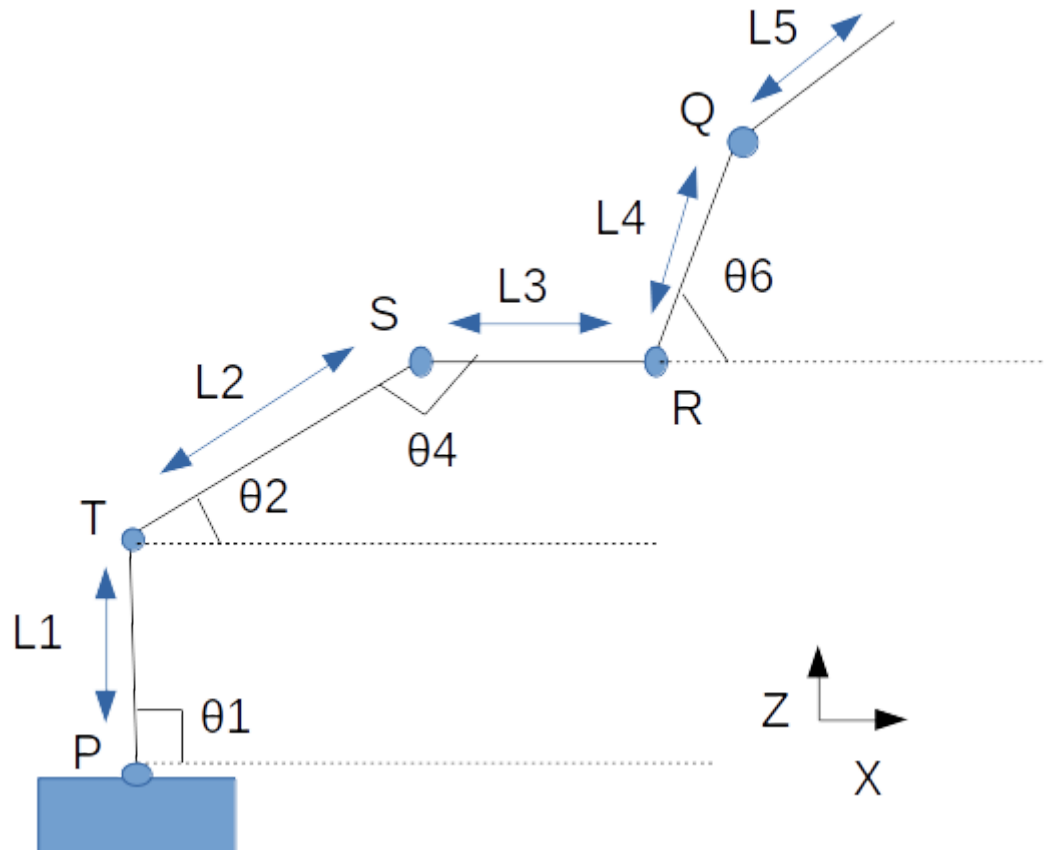


Figura 14: Modelo da articulação.

Para facilitar nossos cálculos e fazer com que a solução se tornasse possível, fizemos mais algumas considerações e constatações, seguem elas:

- Os ângulos θ_3 e θ_5 foram travados, como dito e explicado anteriormente;
- Visto que não trabalhamos com nenhum exemplo de cinemática inversa com manipuladores 6DoF onde seu modelo fica no espaço, consideramos $\theta_0 = 90^\circ$, pois assim, anulando seu efeito, nosso sistema fica restrito ao plano Oxz;
- O ângulo θ_7 não fará interferência no sistema, visto que sua rotação está no plano Oxy;
- Arbitramos um valor de ponto final Q(450, 550), como já explicado anteriormente que para cálculos de cinemática inversa, precisamos de um ponto final, o ponto foi escolhido com base nas simulações e na execução do problema ilustrado em 2.1, onde obtivemos um valor máximo alcançável;

- Além do ponto final, arbitramos ao valor de 0 a orientação de nosso manipulador;
- Atribuímos o ponto P(0,0) ao ponto P demonstrado na Figura 14;
- Ressalta-se que nosso modelo não representa escalas reais dos valores de L1,L2 e etc;
- Está representado aqui, apenas o modelo da articulação do braço esquerdo, visto que a orientação e ângulos do braço direito, seriam iguais porém opostas.

Como nosso ponto Q é (450,550), temos que EX = 450 e EZ é 550, logo encontramos nosso ponto R, utilizando a seguinte expressão:

$$R_x = E_x - (L_4 + L_5) \cdot \cos(o) \quad ; \text{onde 'o' é a orientação final}$$

$$R_z = E_z - (L_4 + L_5) \cdot \sin(o)$$

Então temos R(314,550), feito isso, notamos que nosso problema para θ_2 e θ_4 , pode-se resumir na figura mostrada abaixo:

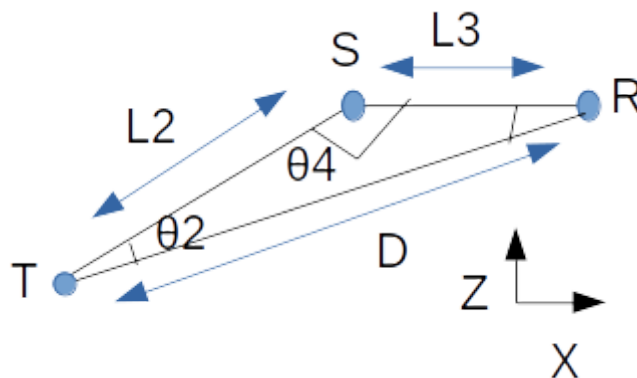


Figura 15: Modelo da articulação para encontrar θ_2 e θ_4 .

Utilizando-se do conceito de Álgebra linear de distância entre dois pontos onde o ponto T se encontra na posição (0, 166), podemos calcular a distância D utilizando-se da seguinte expressão:

$$D = \sqrt{(R_x - T_x)^2 + (R_z - T_z)^2} = 496mm$$

Agora que temos D, podemos aplicar a lei dos cossenos para encontrarmos o valor de θ_4 que segue abaixo:

$$D^2 = L_2^2 + L_3^2 - 2 * \cos(\theta_4) * L_2 * L_3$$

Isolando nossa incógnita temos $\theta_4 = 147.59^\circ$

De forma análoga a partir da mesma lei

$L3^2 = L2^2 + D^2 - 2 * \cos(\theta_2) * D * L2$, conseguimos obter $\theta_2 = 16.64^\circ$

De posse dos pontos Q e R, nosso problema para achar θ_6 é resumido na figura abaixo.

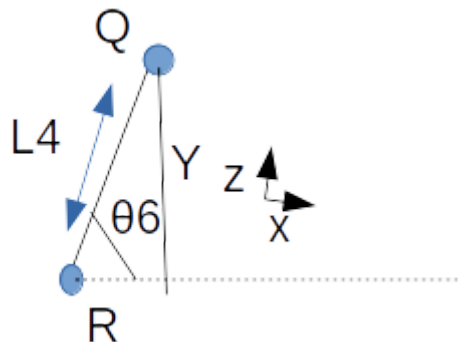


Figura 16: Modelo da articulação para encontrar θ_6 .

Sabemos que Y é $(Q_z - R_z)$, logo $Y = 0$, então $\theta_6 = 0$, o que nos leva a conclusão de um manipulador do seguinte tipo:

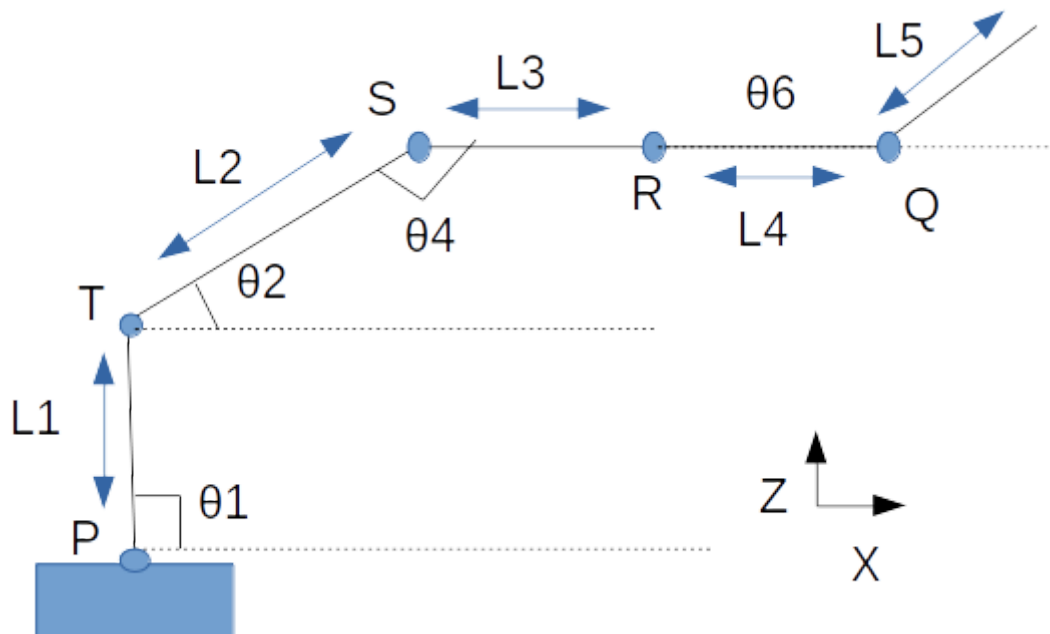


Figura 17: Modelo de conclusão devido aos ângulos achados.

Por fim, nossos ângulos ficaram com os seguintes valores:

$$\theta_1 = 90^\circ$$

$$\theta_2 = 16.64^\circ$$

$$\theta_3 = 0^\circ \text{ (travamento)}$$

$$\theta_4 = 147.59^\circ$$

$$\theta_5 = 0^\circ \text{ (travamento)}$$

$\theta_6 = 0^\circ$ (devido a orientação)

θ_7 = não interfere

2.3 Cinemática Inversa utilizando um algoritmo de Otimização

É possível determinar os valores dos ângulos do manipulador, através de técnicas de programação e utilizando-se de algoritmos disponibilizados pelo professor e outros que buscamos para essa realização, aqui utilizamos um algoritmo encontrado em: <<http://www.icsi.berkeley.edu/~storn/code.html>>, para auxílio do mesmo.

Utilizando do mesmo procedimento feito em 2.2 e travando o ângulo θ_3 porque o mesmo se torna redundante juntamente com θ_5 (como já explicado), sendo assim, bloqueando a dinâmica dele, pode-se dizer que o mesmo não fará mais tanta interferência com o manipulador.

Então, criamos um código com auxílio do código já citado e das funções objetivos onde uma pertence ao braço esquerdo e outra ao braço direito, a função `devec3`, nos retorna um vetor com 3 parâmetros, onde o primeiro dele é o melhor ângulo para cada iteração, no código criado, realizamos 2000 iterações para tal.

Assim, utilizando o código e considerando o ponto P(300, 0, 20), dado que assim estaremos colocando o bloco no centro do domo. considerando que o raio é de 300mm e a largura do bloco é de 40mm, o algoritmo resultou o seguinte para cada junta:

t1	180.0240
t2	100.7321
t3	0
t4	18.5268
t5	180.2577
t6	27.8526
t7	89.8535
VTR	1.0000e-10
w1	169.4889
w2	55.0687
w3	0
w4	162.1999
w5	171.1322
w6	309.2200
w7	276.1806

Figura 18: Resposta no formato do MatLAB.

Para o braço esquerdo:

Nomenclaturas	Ângulo(Graus)
θ_1	180.024°
θ_2	100.732°
θ_3	0°
θ_4	18.527°
θ_5	180.258°
θ_6	27.853°
θ_7	89.853°

Figura 19: Ângulos de Euler Braço Esquerdo

Para o braço direito:

Nomenclaturas	Ângulo(Graus)
θ_1	169.489°
θ_2	55.069°
θ_3	0°
θ_4	162.2°
θ_5	171.132°
θ_6	309.220°
θ_7	276.18°

Figura 20: Ângulos de Euler Braço Direito

3.Cálculo da Trajetória

3.1 Introdução

Para o cálculo da trajetória, iremos abordar dois métodos distintos para tal, cada método apresenta características, vantagens e desvantagens que os tornam melhor ou pior para ser utilizado, no ponto de vista da aplicação desejada,

Neste trabalho iremos fazer o cálculo se utilizando dos seguintes métodos: Ângulos de Euler e Quaternions, onde os braços deverão conseguir manipular uma peça de dimensões (100x40x40 mm), abaixo também iremos demonstrar, uma imagem final desse processo, mostrando toda a trajetória dos nosso manipuladores que irão “carregar” o cubo, fazendo uma rotação em torno dos eixos.

3.2 Ângulos de Euler

Os Ângulos de Euler foram formulados por Leonard Euler para descreverem a orientação de um corpo rígido girando em um espaço euclidiano tridimensional. Neste caso, é útil fazer-se uso de dois sistemas de coordenadas: um sistema inercial fixo e outro que gira junto ao corpo em rotação.

Se baseiam a partir da posição inicial e de uma sequência de ângulos, por meio deste método podemos descobrir a posição e incrementando ou decrementando os ângulos podemos definir a trajetória.

O método de Euler se utiliza da matriz de transformação de coordenadas para obtenção de resultados que será mostrada na figura abaixo.

$$\mathbf{T}_{CI} \equiv \mathbf{T}_{CB} \mathbf{T}_{BA} \mathbf{T}_{AI} = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & \sin \beta \\ 0 & -\sin \beta & \cos \beta \end{bmatrix} \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Figura 21: Matriz de Transformação de Coordenadas.

Para resolução de nosso problema, explanamos abaixo a matriz de ângulos de Euler abaixo, com base na Matriz de Coordenadas e em variáveis que utilizaremos para resolução do problema. onde podemos rotacionar em torno do eixo Z, incrementando ou decrementando “phi”, rotacionar em torno do eixo Y fazendo o mesmo com “psi” e rotacionar em X aplicando o procedimento em “theta”.

$$\begin{bmatrix} \cos(\phi) * \cos(\theta) & \cos(\phi) * \sin(\theta) * \sin(\psi) - \sin(\phi) * \cos(\psi) & \cos(\phi) * \sin(\theta) * \cos(\psi) + \sin(\phi) * \sin(\psi) \\ \sin(\phi) * \cos(\theta) & \sin(\phi) * \sin(\theta) * \sin(\psi) + \cos(\phi) * \cos(\psi) & \sin(\theta) * \sin(\phi) * \cos(\psi) - \cos(\phi) * \sin(\psi) \\ -\sin(\theta) & \cos(\theta) * \sin(\psi) & \cos(\theta) * \cos(\psi) \end{bmatrix}$$

Figura 22: Matriz de Euler.

Para melhor visualização e entendimento do leitor, colocamos cada rotação com uma cor diferente, sendo vermelho para Theta, verde para Phi e azul para Psi então rotacionando nossos eixo espaciais entre 0° a 180°, com passo de -20, -10 e 10 para Phi, Theta e Psi respectivamente, obtivemos os seguintes gráficos:

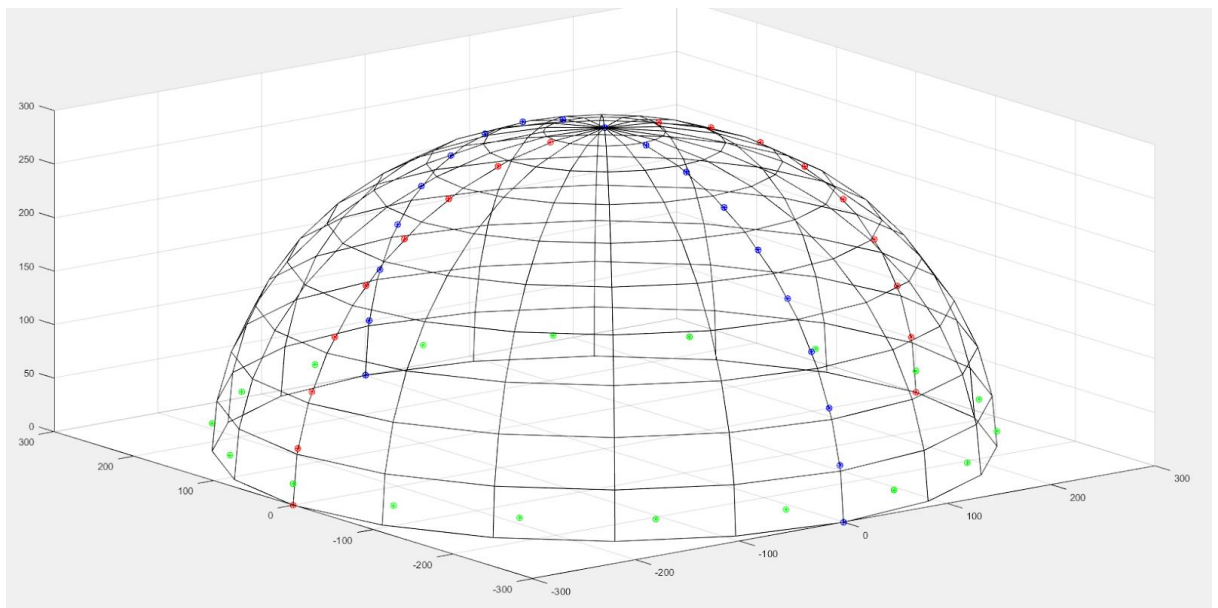


Figura 23: Trajetória de Euler.

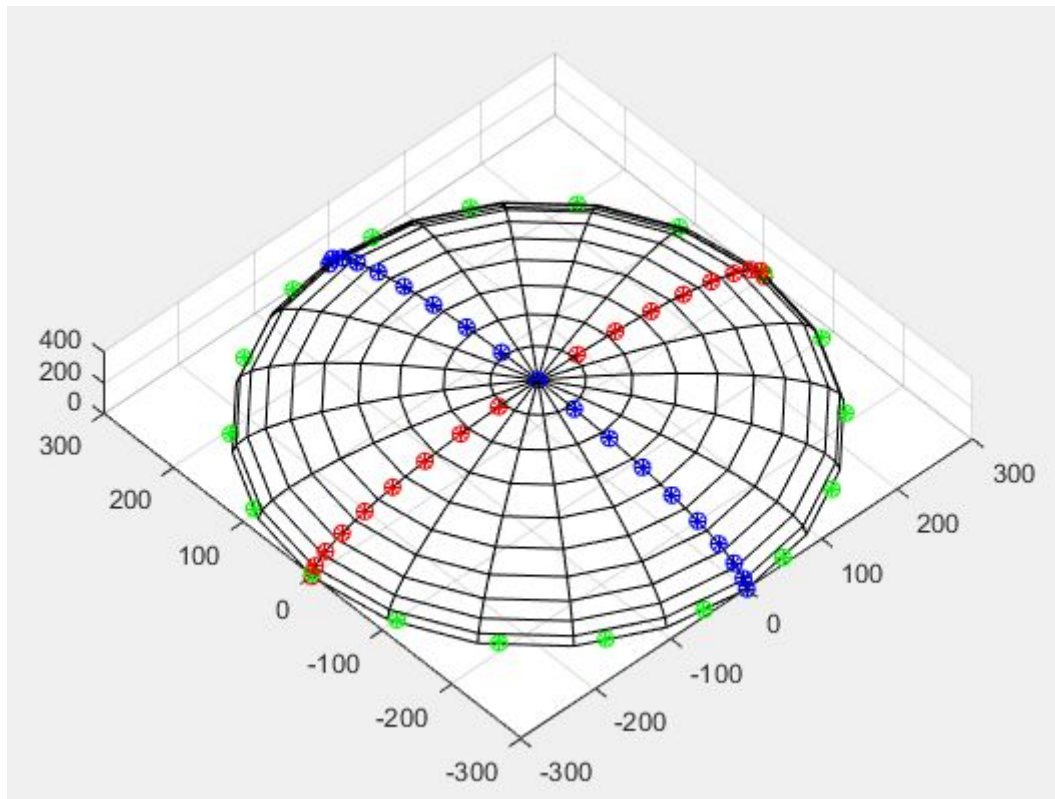


Figura 24: Trajetória de Euler II.

3.3 Quatérnions

Quatérnions são um sistema de numeração que se estende aos número complexos. Eles foram primeiramente descrito pelo matemático irlandês William Rowan Hamilton em 1843 e aplicado a mecânica no espaço tridimensional. Os Quatérnions são geralmente representados sob a forma demonstrada abaixo:

$$a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$$

Figura 25: Representação de Quaternions.

Onde a,b,c e d são número reais e i,j e k são os fundamentais unidades quatérnio, utilizaremos então desses recursos para enfim fazer nosso cálculo de trajetória.

A fim de obter uma comparação entre Quaternions com o método de Euler, foram aplicadas as mesmas rotações citadas anteriormente foi-se também utilizada a seguinte matriz de Quaternions mostrada abaixo para esse procedimento.

$$\begin{bmatrix} 2 * ((q0^2) + (q1^2)) - 1 & 2 * (q1 * q2 + q0 * q3) & 2 * (q1 * q3 - q0 * q2) & 0 \\ 2 * (q1 * q2 - q0 * q3) & 2 * ((q0^2) + (q2^2)) - 1 & 2 * (q2 * q3 + q0 * q2) & 0 \\ 2 * (q1 * q3 - q0 * q2) & 2 * (q2 * q3 - q0 * q1) & 2 * ((q0^2) + (q3^2)) - 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 26: Matriz de Quaternions.

Com auxílio do MatLab foi realizada rotações nos 3 eixos, com as seguintes construções demonstradas abaixo:

Eixo de Rotação	Graus de Rotação	q0	q1	q2	q3
X	180°	cos(θ/2)	sen(θ/2)	0	0
Y	180°	cos(θ/2)	0	sin(θ/2)	0
Z	360°	cos(θ/2)	0	0	sin(θ/2)

Figura 27: Variáveis utilizadas para rotação dos eixos.

Rodando o código no MatLAB para todas as rotações acima e plotando-as, foi obtido o seguinte resultado:

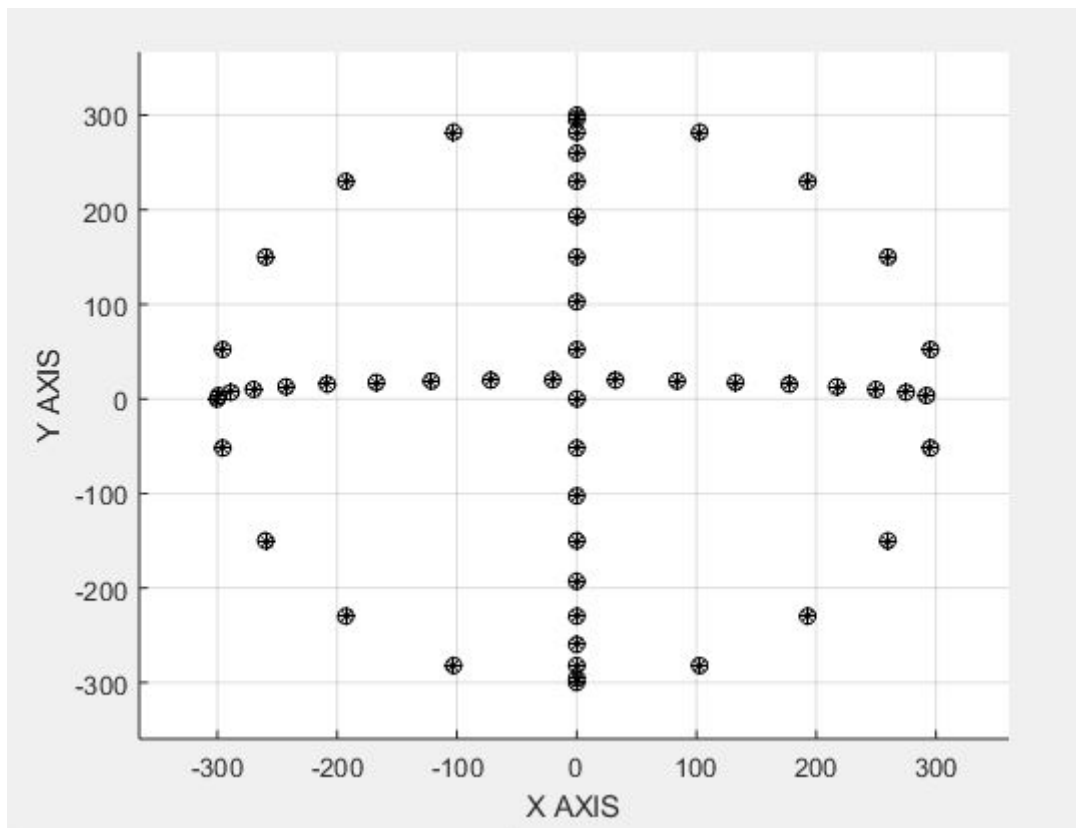


Figura 28: Trajetória Por Quaternions, Vista superior.

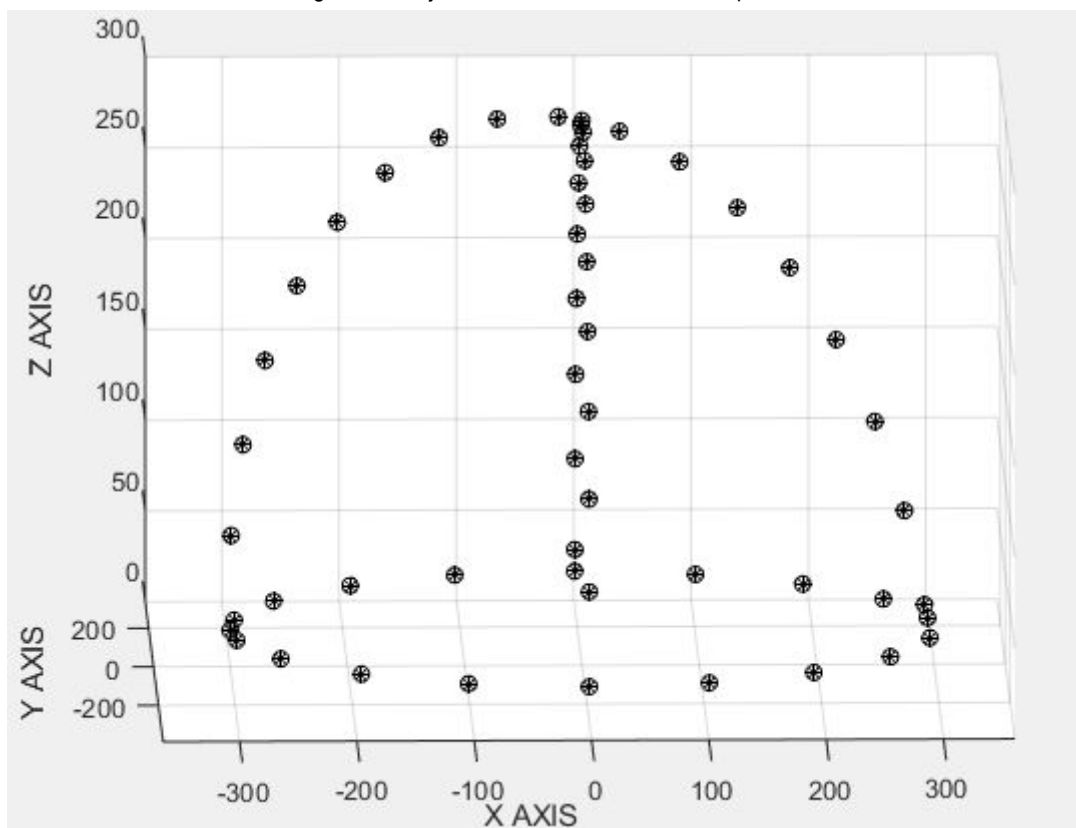


Figura 29: Trajetória em vista tridimensional com Quaternions.

3.4 Movimentação da peça com os manipuladores

Após obtermos os pontos, podemos finalmente em conjunto do software MatLAB, realizar a simulação dos nossos manipuladores, movimentando a peça ao longo da trajetória, para isso utilizamos um algoritmo com laço de repetição “for” para essa aplicação e rotacionarmos a caixa ao longo do eixos X e Z, pois Y está estático na posição em nossa simulação.

Demonstraremos abaixo como ficaram os resultados de nossa implementação(simulação), onde as dimensões da caixa é de (100x40x40 MM) e será denotada pela cor amarela, nossos manipuladores serão representados pela cor preta.

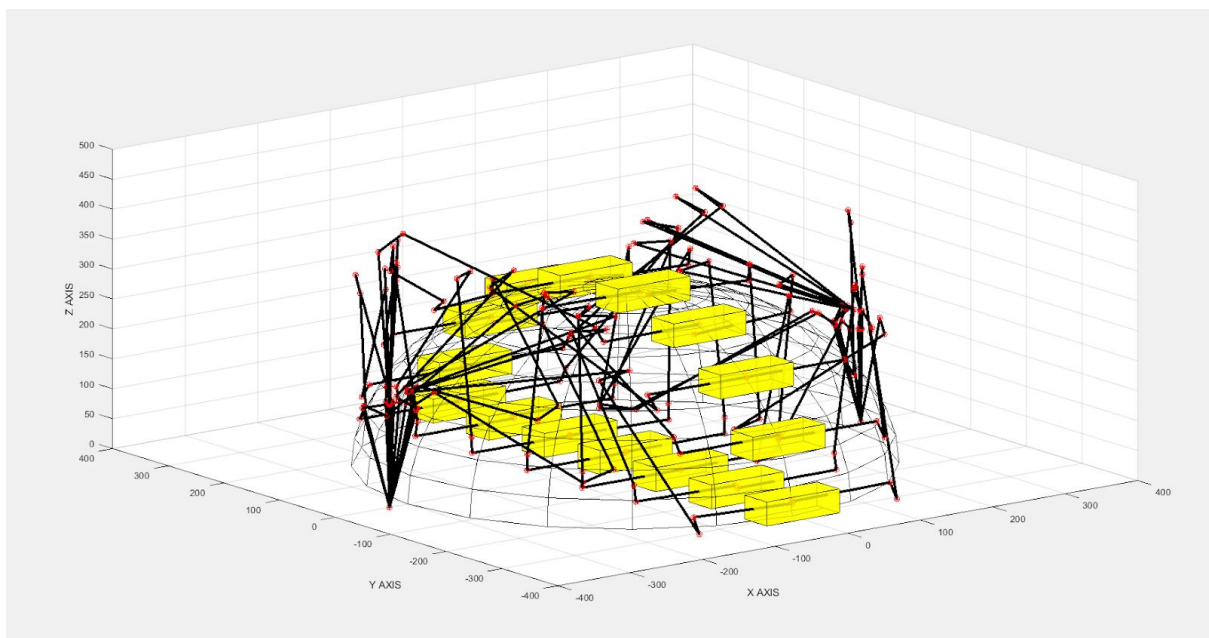


Figura 30: Manipuladores movimentando a caixa em torno do eixo X.

E para o eixo Z:

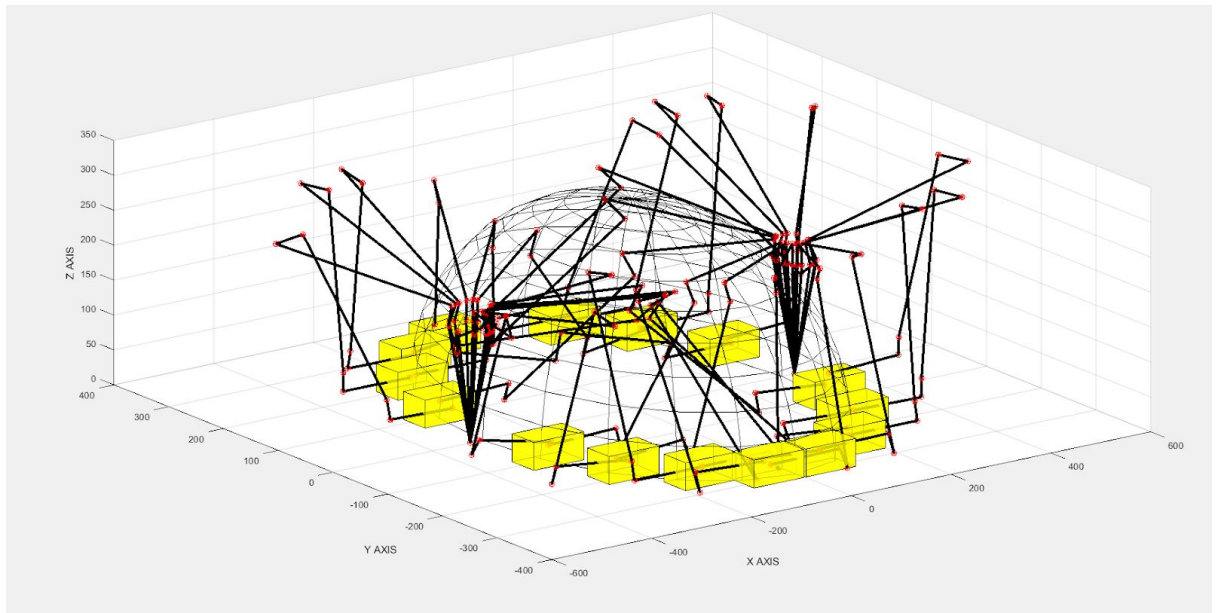


Figura 31: Manipuladores movimentando a caixa em torno do eixo Z.

4. Critérios Escolhidos para distância L

Os critérios utilizados por este tópico foram definidos de acordo com as dimensões do braço Yumi, que como citado é um braço robótico comercial, segue na figura abaixo suas dimensões.

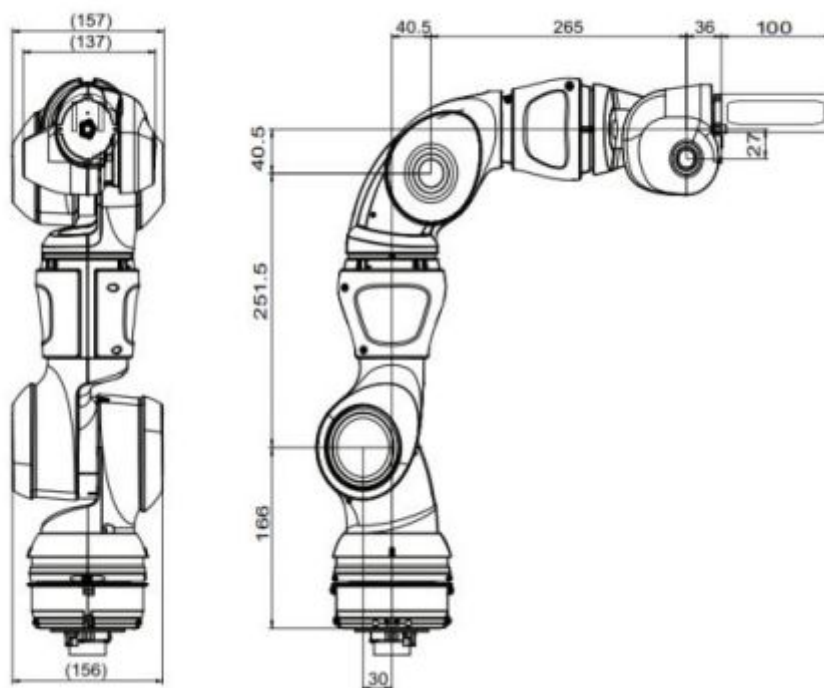


Figura 32: Dimensões do Braço Yumi.

Para achar um meio de calcular a distância denominada “L” para nosso sistema, foram definidos alguns fatores que serão cruciais para tal, enumeramos os mesmos em tópicos exposto abaixo.

- Dimensões da caixa: 100 mm de largura, 40 mm de altura e 40 mm de profundidade. Definimos também que seu centro se encontrará no centro do domo;
- Tem-se 78 mm de carenagem(espacamento ao longo do braço), o que eventualmente pode-se tornar um problema no deslocamentos dos braços, sendo assim um critério a ser considerado;

- Como as articulações se encontram no centro do domo, ponto (300, 0, 0), logo, podemos constatar que o alcance máximo dos braços ao longo do eixo X, é de 0 a 600mm, aliado ao critério acima, podemos definir que não podemos movimentar os dois braços desta maneira;
- A distância máxima que o braço poderá alcançar que é de 687,5 mm no eixo X, essa será a distância ao eixo;
- Então concluímos que as distâncias entre os braços deve estar em uma faixa de 600 ~ 687.5 mm, levando em conta o alcance máximo do domo e dos braços robóticos;
- A partir disto, arbitramos um valor L de 650mm, esta distância nos garantirá uma folga de aproximadamente 25mm;
- Deslocamos o braço esquerdo para o ponto (-325, 0, 0) o que fez o braço direito ser fixado no ponto (325, 0, 0), esta é 'meio' que uma constatação trivial ao olhar as imagens dos manipuladores e suas dimensões.

5. Conclusão

Foi-se aplicado durante o trabalho, vários conceitos aprendidos em sala de aula, como cinemática inversa e direta, ângulos de Euler, Quaternions, parâmetros de D-H entre outros, sendo assim, ficando mais claro suas aplicações e utilizações em um cenário de simulação, coisas difíceis de serem observadas apenas em sala de aula, o trabalho nos mostrou que podemos representar fielmente um manipulador de forma matemática com representação matricial utilizando os conceitos de cinemática, tal abordagem foi constatada, quando utilizamos o software MatLAB e ilustramos seus comportamentos, vale salientar também que os parâmetros de D-H obtidos, correspondem a algo real e comercial que já existe, tendo assim uma noção maior do que está por vir em futuras ocasiões.

Um ponto importante no desenvolvimento deste trabalho foram as práticas aprendidas para passar ao leitor uma melhor visão e explanação de tudo que estava sendo feito, o que nos fez aprimorar em utilização de ferramentas de design e de editores de texto, assim como descobrir novas ferramentas para esses processos.

Tivemos também a experiência de conviver com vários problemas de código e isso nos obrigou a melhorar nossa pesquisa de documentação do software MatLAB que é um software profissional e que certamente estará presente em nossa carreira como engenheiro, além do conhecimento da própria disciplina a prática de pesquisa e dinâmica que adquirimos durante a execução deste trabalho foi de grande valia e que mostra que não precisamos ficar “reinventando a roda” sempre, enfim, vários problemas e dúvidas foram surgindo durante o trabalho, mas procuramos solucioná-las da melhor maneira possível, onde todo o esforço se tornou gratificante ao ver tudo pronto.

6.Referências

[1] *ESPECIFICAÇÕES DO PRODUTO. IRB 14000 (Yumi)*, Disponível em: <<https://www.manualslib.com/manual/1597471/Abb-Irb-14000.html>>. Acesso em: 25 de out. de 2019.

[2] ALGORITHMS. Devec3, Disponível em: <<http://www.icsi.berkeley.edu/~storn/code.html>>. Acesso em: 27 de out. de 2019.

[3] Corke, Peter. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer Verlag NY, 1, 2011. ISBN: 3642201431

[4]Ângulos de Euler.In: Leonard Euler. Disponível em: <https://pt.wikipedia.org/wiki/%C3%82ngulos_de_Euler>. Acesso em: 27 de out. de 2019.

[5]Quaternions.In: Quaterniões. Disponível em: <<https://pt.wikipedia.org/wiki/Quaterni%C3%B5es>>. Acesso em: 27 de out. de 2019.

[6] Prof. André Marcato, *Convenção de Denavit Hartenberg*, Universidade Federal de Juiz de Fora, (2019)

[7] Daniel Fontan Maia da Cruz, *Implementação da Cinemática Inversa de robôs redundantes operando em ambientes confinados no projeto Roboturb, Cinemática diferencial e inversa*, Dissertação em Engenharia Mecânica - Universidade Federal de Santa Catarina. Florianópolis(2007).

[8] *MATLAB DOCUMENTATION*. MatLAB: help, c2019. Disponível em: <<https://www.mathworks.com/help/matlab/>>. Último Acesso em: 01 de nov. de 2019.

