



UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE ENGENHARIA ELÉTRICA

**Estudo, modelagem e controle de
manipuladores com enfoque no
Manipulador Inteligente AX-12A**

Zózimo Pereira Guedes da Silva

Campina Grande, Paraíba

Outubro de 2016

UNIVERSIDADE FEDERAL DE CAMPINA GRANDE

**Estudo, modelagem e controle de manipuladores com
ênfase no Manipulador Inteligente AX-12A**

Zózimo Pereira Guedes da Silva

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Engenharia Elétrica.

Orientador: Prof. Dr. George Acioli Júnior

Campina Grande, Paraíba

Outubro de 2016

**Estudo, modelagem e controle de manipuladores com
ênfase no Manipulador Inteligente AX-12A**

ZÓZIMO PEREIRA GUEDES DA SILVA

Trabalho de Conclusão de Curso apresentado
à Coordenação do Curso de Graduação em
Engenharia Elétrica da Universidade Federal
de Campina Grande como parte dos requisi-
tos necessários para a obtenção do grau de
Bacharel em Engenharia Elétrica.

Aprovado em ____ / ____ / _____

Professor Avaliador: Prof. Dr. Péricles Rezende Barros
Universidade Federal de Campina Grande

Orientador: Prof. Dr. George Acioli Júnior
Universidade Federal de Campina Grande

Campina Grande, Paraíba
Outubro de 2016

Pressão faz diamantes!

A UFCG por oferecer um curso de Engenharia Elétrica mais que especial, um verdadeiro show!

Agradecimentos

Agradeço especialmente ao Professor Dr. George Acioli Júnior por toda paciência e apoio nessa etapa final do curso, ao Professor Dr. Péricles Rezende Barros por ter me dado a oportunidade de crescer em conhecimento no curso e apresentar-me o LIEC.

Agradeço a minha família, em especial a minha mãe Hélia Pereira, meu pai Iolando Guedes, a meu amor Márcia Cristina Araújo, a minha tia Ivalmira Guedes (peça fundamental na minha formação), a seu esposo Antônio Carlos e a amiga Adail Silva por apoiarem fortemente a realização desse sonho.

Agradeço também a todos os colegas do Laboratório de Instrumentação Eletrônica e Controle, em especial a Alequine Batista de Lima, Lucas José da Silva Moreira e Simões Toledo. Por fim agradeço a "Turma do Quiosque", em especial a Oeslle Lucena, Raoni Araújo, Ruan Gomes, Suelson Lopes, Marconi Menezes, Edson Américo, Adriel Nascimento, Luiz Fernando, Júlio César, Klênio Antonovisk, entre outros, cujo apoio e ajuda foram de grande importância para a concretização do curso.

Sumário

1	INTRODUÇÃO	17
2	CINEMÁTICA	19
2.1	Pose de um corpo rígido	19
2.2	Ângulos de Euler	23
2.3	Ângulos ZYZ	24
2.4	Ângulos ZYX ou de rolamento (<i>roll</i>), passo (<i>pitch</i>) e gui- nada(<i>yaw</i>)	25
2.5	Quartênios	27
2.5.1	Notações dos quartênios	28
2.5.2	Algumas propriedades dos quartênios	28
2.6	Transformações homogêneas	33
2.7	Cinemática direta	34
2.8	Cinemática inversa	37
2.8.1	Método geométrico	37
2.8.2	Método usando lógica Fuzzy	38
2.8.3	Comparação entre o Método Geométrico e o Fuzzy	39
2.9	Convenção de Denavit-Hartenberg	40
3	CINEMÁTICA DIFERENCIAL E ESTÁTICA	51
3.1	Jacobiano geométrico	51
3.2	Derivada de uma matriz de rotação	52
3.3	Velocidades dos links	54
3.4	Determinação do Jacobiano	56
3.5	Estática	60
4	GERAÇÃO DE TRAJETÓRIA	61
4.1	Considerações gerais sobre a descrição e geração de rotas	61
4.2	Técnicas para geração de trajetórias	61

4.3	Implementação das trajetórias no Manipulador Inteligente	
	AX-12A	63
4.3.1	Implementação via MatLab e RVC Tools	66
4.3.2	Implementação via MatLab e V-REP	72
4.3.3	Implementação via ROS	72
5	MODELAGEM E IDENTIFICAÇÃO NO MANIPULADOR	
	INTELIGENTE AX-12A	75
5.1	Modelagem	75
5.2	Identificação	75
6	CONSIDERAÇÕES FINAIS	79

Lista de ilustrações

Figura 1 – Posição e orientação de um corpo rígido	19
Figura 2 – Rotacionando um vetor em torno do eixo y_0	21
Figura 3 – Representação dos ângulos de Euler ZYZ	24
Figura 4 – Ângulos de rolamento, passo e guinada	26
Figura 5 – Representação de um ponto P em diferentes <i>frames</i>	33
Figura 6 – Tipos de juntas: revolução e prismática	34
Figura 7 – Descrição dos vetores da matriz de transformação homogênea	35
Figura 8 – Braço robótico inteligente AX-12A	36
Figura 9 – Modelo geométrico do braço robótico	38
Figura 10 – Posições alcançáveis pelo braço robótico	39
Figura 11 – Comparação entre o método geométrico e o Fuzzy	40
Figura 12 – Sistema constituído de 1 junta prismática e três de revolução	41
Figura 13 – Posicionamento dos eixos z	42
Figura 14 – Posicionamento dos eixos x	43
Figura 15 – Ilustração contendo os comprimentos a_i	43
Figura 16 – Ilustração contendo os ângulos θ_i	44
Figura 17 – Ilustração contendo a excentricidade d_1	44
Figura 18 – Braço Robótico Inteligente AX-12A	46
Figura 19 – Posicionamento dos eixos z	47
Figura 20 – Posicionamento dos eixos x	47
Figura 21 – Ilustração contendo os comprimentos a_i	48
Figura 22 – Ilustração contendo os ângulos θ_i	48
Figura 23 – Ilustração contendo as excentricidades d_i	49
Figura 24 – Caracterização de um <i>link</i> i genérico	55
Figura 25 – Manipulador planar dotado de três <i>links</i>	57
Figura 26 – Representação dos parâmetros DH do manipulador dotado de três <i>links</i>	57
Figura 27 – Possíveis trajetórias para alcançar a posição final	62
Figura 28 – Aspecto físico e pinagem do AX-12A	65

Figura 29 – Aspecto físico do USB2Dynamixel	65
Figura 30 – Conexão entre o USB2Dynamixel e o AX-12A	66
Figura 31 – Pinagem do USB2Dynamixel	66
Figura 32 – Manipulador criado com RVC Tools	69
Figura 33 – Representação do braço robótico inteligente AX-12A usando RVC Tools	70
Figura 34 – Interface elaborada no GUIDE do MatLab	71
Figura 35 – Implementação do braço robótico inteligente AX-12A com V-REP	72
Figura 36 – Interface elaborada no RVIZ do ROS	74
Figura 37 – Modelagem mecânica da base	75
Figura 38 – Modelagem mecânica do ombro	76
Figura 39 – Modelagem mecânica do cotovelo	76
Figura 40 – Resposta do servo da base.	77
Figura 41 – Resposta do servo pertencente ao ombro	78
Figura 42 – Resposta do servo do cotovelo	78

Lista de tabelas

Tabela 1 – Parâmetros DH do manipulador dotado de três <i>links</i>	58
Tabela 2 – Parâmetros DH do manipulador dotado de três <i>links</i>	59
Tabela 3 – Características físicas e elétricas do dynamixel AX-12A	64

Resumo

O presente trabalho trata do estudo da cinemática estática e diferencial de manipuladores robóticos, assim como a geração da trajetória, modelagem e identificação dos manipuladores. Apesar do desenvolvimento teórico ser generalizado, foi dado enfoque a exemplos e aplicações envolvendo o Manipulador Inteligente AX-12A. Ao longo do trabalho optou pelo uso das ferramentas MatLab, a *toolbox RVC Tools*, o *software V-REP* e por fim o ROS (*Robot Operating System*).

Palavras-chave: Manipulador, AX-12A, V-REP, *RVC Tools*, ROS (*Robot Operating System*).

Abstract

This work deals with the static kinematics study and differential robotic manipulators, as well as the generation of trajectory modeling and identification of handlers. Despite the theoretical development is widespread, focus was given to examples and applications involving intelligent manipulator AX-12A. Throughout the work chose to use the MatLab tools, textit toolbox RVC Tools, the textit software V-REP and finally the ROS (textit Robot Operating System).

Keywords: robot arm, AX-12A, V-REP, RVC Tools , ROS (*Robot Operating System*).

1 Introdução

Embora a ciência da robótica tenha surgido apenas no século 20, a história da invenções automatizadas criadas pelo homem tem um passado muito maior. Na verdade, um antigo engenheiro herói grego de Alexandria, produziu dois textos, Pneumática e Automata, que testemunham a existência de centenas de diferentes tipos de máquinas, maravilhas capazes de movimento automatizado.

Evidentemente a robótica passou por avanços ao longo dos séculos XX e XXI, com a criação de máquinas capazes de reunir outras máquinas e até mesmo robôs que podem ser confundidos com seres humanos.

É importante dizer que a palavra robótica só foi cunhada em 1941 pelo autor de ficção científica Isaac Asimov em seu livro “O Mentiroso”. Passado isso, autores de ficção científica ao longo da história passaram a se interessar pela capacidade de produção de máquinas que imitem ou motivem os seres humanos, como no mito grego de Pigmalião, em Dr. Frankenstein de Mary Shelley, dentre outros.

Sendo assim, as aplicações envolvendo robótica tem se tornado cada vez mais frequentes em nosso dia a dia. No geral, tais aplicações consistem na realização de movimentos específicos prescritos para o robô. A correta execução de tal movimento fica a cargo do sistema de controle que deve fornecer de maneira precisa sinais aos atuadores capazes de gerar o movimento desejado.

De forma que o presente trabalho encontra-se organizado em seis capítulos.

No primeiro encontra-se a introdução do assunto, no segundo capítulo é tratada a questão da cinemática, que envolve a definição de *pose* e apresentação das principais representações.

No terceiro capítulo trabalha-se a cinemática diferencial, conceitos como Jacobiano e derivada de uma matriz de rotação são elucidados. Além disso é dada uma pequena descrição da estática de manipulador, assim como sua principal equação no estudo de manipuladores.

No quarto capítulo o assunto alvo foi a geração de trajetórias, para isso abordou-se duas técnicas e implementou-se de três maneiras distintas tais trajetórias.

Já no capítulo cinco realizou-se a modelagem e a identificação do Manipulador

Inteligente AX-12A, para isso foi utilizado conhecimentos da teoria de controle e o software MatLab.

Por fim, no capítulo seis são apresentadas algumas sugestões de forma a melhorar trabalhos futuros e uma breve conclusão

2 Cinemática

Do ponto de vista estrutural um manipulador nada mais é que um conjunto de corpos rígidos (*links*) interligados entre si por meio de juntas (prismáticas ou de revolução). Na qual o movimento do elemento terminal será dessa forma a composição dos movimentos individuais de cada um dos *links* em relação ao *link* anterior.

Assim tal capítulo será dedicado a derivação das equações cinemáticas diretas através de uma sistemática abordagem baseada na álgebra linear. Isso permite a obtenção da posição e orientação (*pose*) do elemento terminal como uma função das variáveis das juntas.

Por fim, este capítulo termina com a derivação da solução de um problema de cinemática inversa e com a elucidação da convenção de Denavit-Hartenberg.

2.1 Pose de um corpo rígido

Pode-se descrever um corpo rígido no espaço a partir da elucidação de sua *pose*, isto é, sua posição e orientação em relação a um *frame* de referência O-XYZ, como mostra a Figura 1.

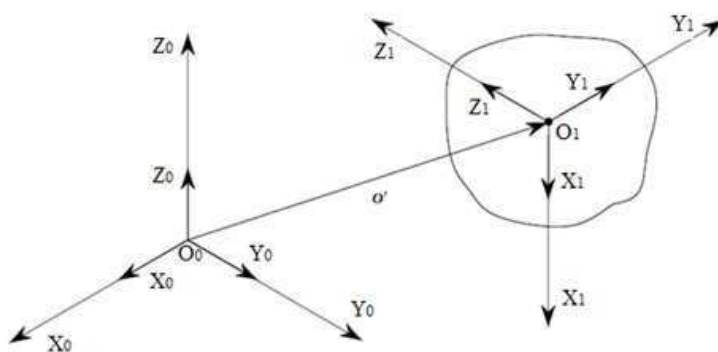


Figura 1 – Posição e orientação de um corpo rígido

A posição de um ponto o' no corpo rígido com respeito ao *frame* $O_0-X_0Y_0Z_0$ é expresso pela relação Equação (2.1).

$$\vec{o}' = o'_x \vec{x}_0 + o'_y \vec{y}_0 + o'_z \vec{z}_0 \quad (2.1)$$

Assim, as coordenadas de o' em relação ao *frame* $O_0-X_0Y_0Z_0$ são $(o'_x, o'_y, o'_z)^T$ uma vez que satisfazem a Equação (2.1). De maneira similar é possível obter as coordenadas do vetor o'_1 associado a um outro *frame* $O_1-X_1Y_1Z_1$ valendo-se da operação produto escalar. Em suma, usa-se o *produto escalar* (\cdot) para determinar a *projeção* de um vetor em uma dada direção. Tal projeção pode ser de natureza escalar ou vetorial.

Assim,

$$\begin{aligned} o'_1 &= \begin{bmatrix} \vec{o}'_0 \cdot \vec{x}_1 \\ \vec{o}'_0 \cdot \vec{y}_1 \\ \vec{o}'_0 \cdot \vec{z}_1 \end{bmatrix} = \begin{bmatrix} (o'_x \vec{x}_0 + o'_y \vec{y}_0 + o'_z \vec{z}_0) \cdot \vec{x}_1 \\ (o'_x \vec{x}_0 + o'_y \vec{y}_0 + o'_z \vec{z}_0) \cdot \vec{y}_1 \\ (o'_x \vec{x}_0 + o'_y \vec{y}_0 + o'_z \vec{z}_0) \cdot \vec{z}_1 \end{bmatrix} \\ o'_1 &= \begin{bmatrix} o'_x \vec{x}_0 \cdot \vec{x}_1 + o'_y \vec{y}_0 \cdot \vec{x}_1 + o'_z \vec{z}_0 \cdot \vec{x}_1 \\ o'_x \vec{x}_0 \cdot \vec{y}_1 + o'_y \vec{y}_0 \cdot \vec{y}_1 + o'_z \vec{z}_0 \cdot \vec{y}_1 \\ o'_x \vec{x}_0 \cdot \vec{z}_1 + o'_y \vec{y}_0 \cdot \vec{z}_1 + o'_z \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} \\ o'_1 &= \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} \cdot \begin{bmatrix} o'_x \\ o'_y \\ o'_z \end{bmatrix} \\ o'_1 &= \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} \cdot o'_0 \quad (2.2) \end{aligned}$$

É importante ressaltar que os conjuntos de vetores $\begin{bmatrix} x \\ y \\ z \end{bmatrix}$ e $\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$ são ortonormais, ou seja, todos os vetores do conjunto tem norma 1 e o produto escalar entre dois vetores do conjunto é zero.

Por fim defini-se a matriz $\begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix}$ como a matriz de transformação R_0^1 . E além disso,

$$o'_1 = \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix}^{-1} \cdot o'_1 \quad (2.3)$$

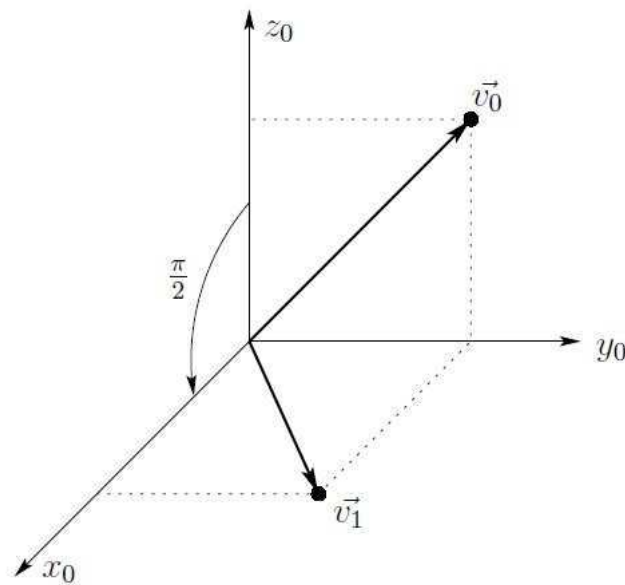


Figura 2 – Rotacionando um vetor em torno do eixo y_0

Exemplo 2.1:

O vetor v com coordenadas $v_0 = (0, 1, 1)^T$ é rotacionado em torno do eixo y_0 de $\frac{\pi}{2}$ conforme ilustra a Figura (2). Dito isso, calcule v_1 em relação ao novo *frame*.

Observação: De agora em diante, quando conveniente, será utilizado as notações $s(\theta)$ e $c(\theta)$ para $\sin(\theta)$ e $\cos(\theta)$, respectivamente.

Resolução

- Passo 01: Determina-se a matriz de transformação R_0^1 ;

$$R_0^1 = \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} = \begin{bmatrix} 1.1.c(\frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) & 1.1.c(\pi) \\ 1.1.c(\frac{\pi}{2}) & 1.1.c(0) & 1.1.c(\frac{\pi}{2}) \\ 1.1.c(0) & 1.1.c(\frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

- Passo 02: Encontra-se v_1

$$v_1 = R_0^1 \cdot v_0 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}$$

Exemplo 2.2:

Baseado na Figura 2 encontre a matriz de transformação para os casos em que aja uma rotação de φ , ϑ e ψ rad em torno do eixos x , y e z no sentido anti-horário respectivamente.

Resolução

- Rotação em torno do eixo x : Determina-se a matriz de transformação R_0^1 .

$$R_x(\varphi) = R_0^1 = \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} = \begin{bmatrix} 1.1.c(0) & 1.1.c(\frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) \\ 1.1.c(\frac{\pi}{2}) & 1.1.c(\varphi) & 1.1.c(\varphi + \frac{\pi}{2}) \\ 1.1.c(\frac{\pi}{2}) & 1.1.c(\varphi - \frac{\pi}{2}) & 1.1.c(\varphi) \end{bmatrix}$$

$$R_x(\varphi) = R_0^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\varphi) & -s(\varphi) \\ 0 & s(\varphi) & c(\varphi) \end{bmatrix}$$

- Rotação em torno do eixo y : Determina-se a matriz de transformação R_0^1 .

$$R_y(\vartheta) = R_0^1 = \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} = \begin{bmatrix} 1.1.c(\vartheta) & 1.1.c(\frac{\pi}{2}) & 1.1.c(\vartheta - \frac{\pi}{2}) \\ 1.1.c(\frac{\pi}{2}) & 1.1.c(0) & 1.1.c(\frac{\pi}{2}) \\ 1.1.c(\vartheta + \frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) & 1.1.c(\vartheta) \end{bmatrix}$$

$$R_y(\vartheta) = R_0^1 = \begin{bmatrix} c(\vartheta) & 0 & s(\vartheta) \\ 0 & 1 & 0 \\ -s(\vartheta) & 0 & c(\vartheta) \end{bmatrix}$$

- Rotação em torno do eixo z : Determina-se a matriz de transformação R_0^1 .

$$R_z(\psi) = R_0^1 = \begin{bmatrix} \vec{x}_0 \cdot \vec{x}_1 & \vec{y}_0 \cdot \vec{x}_1 & \vec{z}_0 \cdot \vec{x}_1 \\ \vec{x}_0 \cdot \vec{y}_1 & \vec{y}_0 \cdot \vec{y}_1 & \vec{z}_0 \cdot \vec{y}_1 \\ \vec{x}_0 \cdot \vec{z}_1 & \vec{y}_0 \cdot \vec{z}_1 & \vec{z}_0 \cdot \vec{z}_1 \end{bmatrix} = \begin{bmatrix} 1.1.c(\psi) & 1.1.c(\psi + \frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) \\ 1.1.c(\psi - \frac{\pi}{2}) & 1.1.c(\psi) & 1.1.c(\frac{\pi}{2}) \\ 1.1.c(\frac{\pi}{2}) & 1.1.c(\frac{\pi}{2}) & 1.1.c(0) \end{bmatrix}$$

$$R_z(\psi) = R_0^1 = \begin{bmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Dessa forma, foram encontradas as três matrizes de transformações que levam vetores de um *frame* originário de uma rotação ao *frame* de referência. Para sucessivas rotações encontramos a matriz de transformação resultante multiplicando cada uma das matrizes de transformação referentes a cada uma das rotações, ou seja, para n rotações temos:

$$R_0^n = R_{n-1}^n \cdot R_{n-2}^{n-1} \dots R_1^2 \cdot R_0^1 \quad (2.4)$$

2.2 Ângulos de Euler

É possível demonstrar que qualquer rotação pode ser obtida a partir de três rotações elementares contanto que não aja duas rotações sucessivas sobre eixos paralelos. Sendo assim um conjunto de três ângulos $\phi = [\varphi \ \vartheta \ \psi]^T$ para descrever a orientação de um corpo. Por conseguinte, haverão 12 conjuntos distintos de ângulos dentre as 27 combinações possíveis, cada um representando uma tripla de ângulos de Euler. Neste trabalho, serão analisados dois conjuntos de ângulos, o ZYZ e o ZYX .

2.3 Ângulos ZYZ

A rotação descrita pelos ângulos ZYZ é obtida pela composição das seguintes rotações elementares Figura 3.

1. Rotaciona-se o *frame* de referência de um ângulo φ em torno do eixo z ; tal rotação é descrita pela matriz de transformação $R_z(\varphi)$;
2. Rotaciona-se o *frame* de referência de um ângulo ϑ em torno do eixo y ; tal rotação é descrita pela matriz de transformação $R_y(\vartheta)$;
3. Rotaciona-se o *frame* de referência de um ângulo ψ em torno do eixo z ; tal rotação é descrita pela matriz de transformação $R_z(\psi)$.

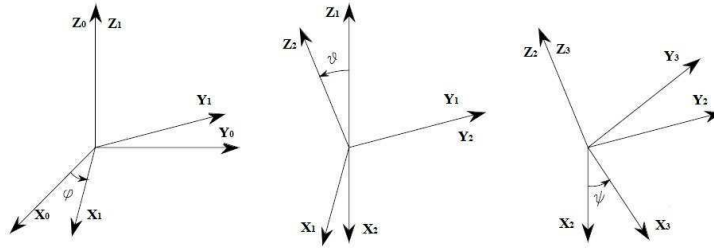


Figura 3 – Representação dos ângulos de Euler ZYZ

Analiticamente, a sequência ZYZ pode ser expressa a partir da Equação (2.5) obtida a seguir

$$R_{ZYZ}(\varphi, \vartheta, \psi) = R_z(\varphi) \cdot R_y(\vartheta) \cdot R_z(\psi)$$

$$R_{ZYZ}(\varphi, \vartheta, \psi) = \begin{bmatrix} c(\varphi) & -s(\varphi) & 0 \\ s(\varphi) & c(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} c(\vartheta) & 0 & s(\vartheta) \\ 0 & 1 & 0 \\ -s(\vartheta) & 0 & c(\vartheta) \end{bmatrix} \cdot \begin{bmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_{ZYZ}(\varphi, \vartheta, \psi) = \begin{bmatrix} c(\varphi)c(\vartheta)c(\psi) - s(\varphi)s(\psi) & -c(\varphi)c(\vartheta)s(\psi) - s(\varphi)c(\psi) & c(\varphi)s(\vartheta) \\ s(\varphi)c(\vartheta)c(\psi) - c(\varphi)s(\psi) & -s(\varphi)c(\vartheta)s(\psi) + c(\varphi)c(\psi) & s(\varphi)s(\vartheta) \\ -s(\vartheta)c(\psi) & -s(\vartheta)s(\psi) & c(\vartheta) \end{bmatrix} \quad (2.5)$$

A matriz $R_{ZYZ}(\varphi, \vartheta, \psi)$ é denominada **Transformação em Ângulos de Euler – ZYZ**. Todavia é importante ressaltar que o mais importante e difícil não consiste na obtenção de tal matriz mas sim na resolução do seguinte problema: Dada uma matriz R tal que,

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

determinar um conjunto de ângulos de Euler φ , ϑ e ψ tal que

$$R = R_{ZYZ}(\varphi, \vartheta, \psi)$$

2.4 Ângulos ZYX ou de rolamento (*roll*), passo (*pitch*) e guinada(*yaw*)

Uma matriz R pode ser descrita como o produto de sucessivas rotações em torno dos eixos coordenados x_0, y_0 e z_0 realizadas em uma ordem específica. Essa rotações definem os ângulos de rolamento, passo e guinada denotados também por φ , ϑ e ψ . A rotação descrita pelos ângulos ZYX é obtida pela composição das seguintes rotações elementares Figura 4.

1. Rotaciona-se o *frame* de referência de um ângulo φ em torno do eixo z (guinada); tal rotação é descrita pela matriz de transformação $R_z(\varphi)$;
2. Rotaciona-se o *frame* de referência de um ângulo ϑ em torno do eixo y (passo); tal rotação é descrita pela matriz de transformação $R_y(\vartheta)$;
3. Rotaciona-se o *frame* de referência de um ângulo ψ em torno do eixo x (guinada); tal rotação é descrita pela matriz de transformação $R_x(\psi)$.

Analiticamente, a sequência ZYX pode ser expressa a partir da Equação (2.6)

$$R_{ZYX}(\varphi, \vartheta, \psi) = R_z(\varphi) \cdot R_y(\vartheta) \cdot R_x(\psi)$$

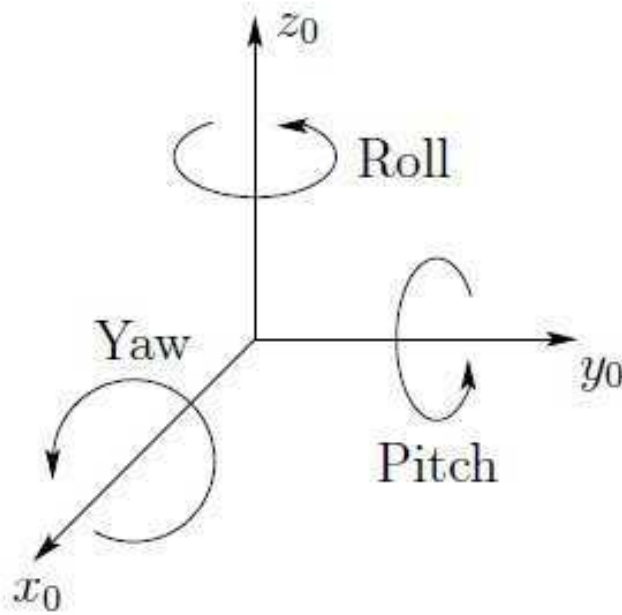


Figura 4 – Ângulos de rolamento, passo e guinada

$$R_{ZYX}(\varphi, \vartheta, \psi) = \begin{bmatrix} c(\varphi) & -s(\varphi) & 0 \\ s(\varphi) & c(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} c(\vartheta) & 0 & s(\vartheta) \\ 0 & 1 & 0 \\ -s(\vartheta) & 0 & c(\vartheta) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\psi) & -s(\psi) \\ 0 & s(\psi) & c(\psi) \end{bmatrix}$$

$$R_{ZYX}(\varphi, \vartheta, \psi) = \begin{bmatrix} c(\varphi)c(\vartheta) & c(\varphi)c(\vartheta)s(\psi) - s(\varphi)c(\psi) & c(\varphi)s(\vartheta)c(\psi) + s(\varphi)s(\psi) \\ s(\varphi)c(\vartheta) & -s(\varphi)s(\vartheta)s(\psi) + c(\varphi)c(\psi) & s(\varphi)s(\vartheta)c(\psi) - c(\varphi)s(\psi) \\ -s(\vartheta) & c(\vartheta)s(\psi) & c(\vartheta)c(\psi) \end{bmatrix} \quad (2.6)$$

Assim como para o caso dos ângulos ZYZ , a solução inversa para uma dada matriz

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

pode ser obtida igualando R a matriz $R_{ZYZ}(\varphi, \vartheta, \psi)$ (Equação (2.6)).

2.5 Quartênios

A partir do momento que os números da forma $a+jb$, isto é, números complexos de posto 2 ganharam grande aceitação, matemáticos buscaram outros sistemas matemáticos sobre o campo dos números hiper-complexos de postos 3, 4, ..., n .

Depois de anos de luta para criar tal sistema, um súbito golpe de intuição matemática veio sobre William Rowan Hamilton em 1843. Hamilton contava a história de que a ideia de abandonar a lei comutativa da multiplicação ocorreu enquanto caminhava com a esposa ao longo do Royal Canal perto de Dublin, pouco antes de escurecer. Ele pegou seu canivete e com ele gravou a parte fundamental da tábua de multiplicação dos quatérnios numa das pedras da Ponte Brougham (ainda hoje existe a pedra e uma placa contando a história citada).

$$i^2 = j^2 = k^2 = ijk = -1 \quad (2.7)$$

da Equação (2.7) pode-se chegar a

$$ij = i x j = k = -j x i = -ji \quad (2.8)$$

$$jk = j x k = i = -k x j = -kj$$

Todos as operações algébricas entre quartênios procedem de (2.7) e (2.9), isto é, o produto de dois quartênios \mathbf{p} e \mathbf{q} em que

$$p = p_0 + \mathbf{p} = p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3 \quad (2.9)$$

$$q = q_0 + \mathbf{q} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 \quad (2.10)$$

cujo produto é

$$pq = p_0q_0 - \mathbf{p}\mathbf{q} + p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p}x\mathbf{q} \quad (2.11)$$

Tudo de acordo com as Equações (2.7) e (2.9).

2.5.1 Notações dos quartênios

As letras em negrito denotam vetores no espaço tridimensional, \mathbb{R}^3 , em particular utilizar-se-á i, j e k para denotar a base ortonormal padrão do \mathbb{R}^3 . Assim os vetores tridimensionais são escritos com a base

$$i = (1, 0, 0) \quad j = (0, 1, 0) \quad k = (0, 0, 1) \quad (2.12)$$

Um quartênio, como o nome sugere, é uma 4 – *upla* que define um elemento em \mathbb{R}^4 . Assim para o quartênio, escrevemos

$$q = (q_0, q_1, q_2, q_3) \quad (2.13)$$

em que q_0, q_1, q_2 e q_3 são simplesmente número reais ou escalares. Como uma representação alternativa para o quartênio, define-se uma parte escalar – algum número real q_0 – e associado com ele uma parte vetorial \mathbf{q} pertencente ao \mathbb{R}^3 .

Assim define-se o quartênio como a soma

$$q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 \quad (2.14)$$

no geral denomina-se q_0 como a parte escalar do quartênio enquanto \mathbf{q} é chamada de parte vetorial do quartênio. Os escalares q_0, q_1, q_2 e q_3 são chamadas componentes do quartênio. Além disso, defini-se como quartênio puro aquele cuja parte escalar é zero.

2.5.2 Algumas propriedades dos quartênios

1. Conjugado complexo

O conjugado complexo do quartênio

$$q = q_0 + \mathbf{q} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 \quad (2.15)$$

é denotado por

$$q^* = q_0 - \mathbf{q} = q_0 - \mathbf{i}q_1 - \mathbf{j}q_2 - \mathbf{k}q_3 \quad (2.16)$$

Segue-se que

$$(pq)^* = p^*q^* \quad e \quad (p^*q) = p^*q \quad (2.17)$$

2. Norma do quartênio

A norma do quartênio q é denotada pelo escalar $N(q)$ em que

$$N(q) = \sqrt{q^*.q} \quad ou \quad N^2(q) = q^*.q \quad (2.18)$$

3. Quartênio unitário

Um quartênio unitário, q , tem uma norma igual a um, isto é,

$$|q| = |q^*| = 1 \quad e \quad N^2(q) = q^*.q = 1 \quad (2.19)$$

Assim, o produto de quartênios unitários também é um quartênio unitário.

4. Quartênio inverso

Por definição de inversa tem-se $q^{-1}q = qq^{-1} = 1$. Então multiplicando o segundo e terceiro termo da expressão por q^* temos

$$q^*qq^{-1} = N^2(q)q^{-1} = q^* \quad (2.20)$$

da qual de segue

$$q^{-1} = \frac{q^*}{N^2} = \frac{q^*}{|q|^2} \quad (2.21)$$

se q é um quartênio unitário, então

$$q^{-1} = q^* \quad (2.22)$$

5. O operador Rotação

Antes de tudo é importante notar que qualquer quartênio unitário pode ser escrito como

$$q = q_o + \mathbf{q} = \cos(\theta) + \mathbf{u} \operatorname{sen}(\theta) \quad (2.23)$$

em que

$$\mathbf{u} = \frac{\mathbf{q}}{|\mathbf{q}|} \quad e \quad \operatorname{tg}(\theta) = \frac{|\mathbf{q}|}{q_o} \quad (2.24)$$

Teorema 1. Para qualquer quartênio unitário

$$q = q_o + \mathbf{q} = \cos(\theta) + \mathbf{u} \operatorname{sen}(\theta) \quad (2.25)$$

e para qualquer vetor $v \in \mathbb{R}^3$ a ação do operador

$$L_q(\mathbf{v}) = q\mathbf{v}q^* \quad (2.26)$$

pode ser interpretada como uma rotação do vetor v de um ângulo 2θ tendo \mathbf{q} como eixo de rotação.

Teorema 2. Para qualquer quartênio unitário

$$q = q_o + \mathbf{q} = \cos(\theta) + \mathbf{u} \operatorname{sen}(\theta) \quad (2.27)$$

e para qualquer vetor $v \in \mathbb{R}^3$ a ação do operador

$$L_{q^*}(\mathbf{v}) = q^*\mathbf{v}q \quad (2.28)$$

pode ser interpretada geometricamente como

- Uma rotação do *frame* de coordenadas com respeito ao vetor v de um ângulo 2θ tendo \mathbf{q} como eixo de rotação, ou
- Uma rotação oposta do vetor v com respeito ao *frame* de um ângulo 2θ tendo \mathbf{q} como eixo de rotação.

Exemplo 2.3:

O vetor v com coordenadas $v_0 = (0, 1, 1)^T$ é rotacionado em torno do eixo y_0 de $\frac{\pi}{2}$ conforme ilustra a Figura 2. Dito isso, calcule v_1 em relação ao novo *frame* usando quartênios.

Resolução

- Passo 01: Valendo-se das Equações (2.25), (2.27), (2.28) e tomando $\mathbf{v} = \mathbf{i}$ temos:

$$q_{y_0, \frac{\pi}{2}} = \cos\left(\frac{\pi}{4}\right) + \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right) q_{y_0, \frac{\pi}{2}}^* = \cos\left(\frac{\pi}{4}\right) - \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right)$$

então

$$q_{y_0, \frac{\pi}{2}}^* \mathbf{v} q_{y_0, \frac{\pi}{2}} = \left(\frac{1}{\sqrt{2}} - \mathbf{j} \frac{1}{\sqrt{2}}\right) \mathbf{i} \left(\frac{1}{\sqrt{2}} + \mathbf{j} \frac{1}{\sqrt{2}}\right) = \mathbf{i} \frac{1}{2} + \mathbf{k} \frac{1}{2} + \mathbf{k} \frac{1}{2} - \mathbf{i} \frac{1}{2} = \mathbf{k} \mathbf{1} \quad \text{ou} \quad \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

que coincide com a primeira coluna da matriz de transformação obtida no Exemplo 2.1.

- Passo 02: Valendo-se das Equações (2.25), (2.27), (2.28) e tomando $\mathbf{v} = \mathbf{j}$ temos:

$$q_{y_0, \frac{\pi}{2}} = \cos\left(\frac{\pi}{4}\right) + \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right) q_{y_0, \frac{\pi}{2}}^* = \cos\left(\frac{\pi}{4}\right) - \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right)$$

então

$$q_{y_0, \frac{\pi}{2}}^* \mathbf{v} q_{y_0, \frac{\pi}{2}} = \left(\frac{1}{\sqrt{2}} - \mathbf{j} \frac{1}{\sqrt{2}}\right) \mathbf{j} \left(\frac{1}{\sqrt{2}} + \mathbf{j} \frac{1}{\sqrt{2}}\right) = \mathbf{j} \frac{1}{2} - \frac{1}{2} + \frac{1}{2} + \mathbf{j} \frac{1}{2} = \mathbf{j} \mathbf{1} \quad \text{ou} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

que coincide com a segunda coluna da matriz de transformação obtida no Exemplo 2.1.

- Passo 03: Valendo-se das Equações (2.25), (2.27), (2.28) e tomando $\mathbf{v} = \mathbf{k}$ temos:

$$q_{y_0, \frac{\pi}{2}} = \cos\left(\frac{\pi}{4}\right) + \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right) q_{y_0, \frac{\pi}{2}}^* = \cos\left(\frac{\pi}{4}\right) - \mathbf{j} \operatorname{sen}\left(\frac{\pi}{4}\right)$$

então

$$q_{y_0, \frac{\pi}{2}}^* \mathbf{v} q_{y_0, \frac{\pi}{2}} = \left(\frac{1}{\sqrt{2}} - \mathbf{j} \frac{1}{\sqrt{2}}\right) \mathbf{k} \left(\frac{1}{\sqrt{2}} + \mathbf{j} \frac{1}{\sqrt{2}}\right) = \mathbf{k} \frac{1}{2} - \mathbf{i} \frac{1}{2} - \mathbf{i} \frac{1}{2} - \mathbf{k} \frac{1}{2} = -\mathbf{i} \mathbf{1} \quad \text{ou} \quad \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$$

que coincide com a terceira coluna da matriz de transformação obtida no exemplo 2.1.

Exemplo 2.4:

Encontre a matriz de transformação para os casos em que aja duas rotações sucessivas de α e β rad em torno do eixos z e y no sentido anti-horário respectivamente.

Resolução

- Passo 01: Valendo-se das Equações (2.25), (2.27), (2.28) e tomando $\mathbf{v} = \mathbf{i}$ temos:

$$q_{z, \alpha} = \cos\left(\frac{\alpha}{2}\right) + \mathbf{k} \operatorname{sen}\left(\frac{\alpha}{2}\right) q_{y, \beta} = \cos\left(\frac{\beta}{2}\right) + \mathbf{j} \operatorname{sen}\left(\frac{\beta}{2}\right)$$

então

$$q_{y, \beta}^* q_{z, \alpha}^* \mathbf{v} q_{z, \alpha} q_{y, \beta} = \left(\cos\left(\frac{\beta}{2}\right) - \mathbf{j} \operatorname{sen}\left(\frac{\beta}{2}\right)\right) \mathbf{i} \left(\cos\left(\frac{\alpha}{2}\right) + \mathbf{k} \operatorname{sen}\left(\frac{\alpha}{2}\right)\right) \left(\cos\left(\frac{\beta}{2}\right) + \mathbf{j} \operatorname{sen}\left(\frac{\beta}{2}\right)\right)$$

$$q_{y, \beta}^* q_{z, \alpha}^* \mathbf{v} q_{z, \alpha} q_{y, \beta} = \mathbf{i} c(\alpha) c(\beta) - \mathbf{j} s(\alpha) + \mathbf{k} c(\alpha) s(\beta) \quad \text{ou} \quad \begin{bmatrix} c(\alpha) c(\beta) \\ -s(\alpha) \\ c(\alpha) s(\beta) \end{bmatrix}$$

De maneira análoga é possível determinar as duas colunas restantes da matriz de transformação tomando $\mathbf{v} = \mathbf{j}$ e $\mathbf{v} = \mathbf{k}$ respectivamente.

2.6 Transformações homogêneas

Considere um ponto arbitrário P no espaço (Figura 5). Seja p^0 o vetor de coordenadas de P com respeito ao *frame* de referência $O_0 - x_0y_0z_0$. Considere outro *frame* no espaço $O_1 - x_1y_1z_1$. Seja \mathbf{o}_1^0 o vetor ligando a origem do *frame* 1 ao *frame* 0 e R_1^0 a matriz de transformação do *frame* 1 com relação ao *frame* 0. Seja também p^1 o vetor de coordenadas de P com respeito ao *frame* de referência $O_1 - x_1y_1z_1$. De uma simples análise geométrica temos que a posição do ponto P com relação ao *frame* de referência pode ser expresso como

$$\mathbf{p}^0 = \mathbf{o}_1^0 + R_1^0 \mathbf{p}^1 \quad (2.29)$$

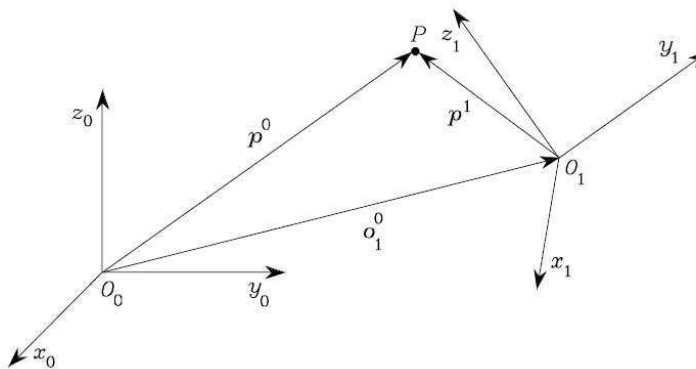


Figura 5 – Representação de um ponto P em diferentes *frames*

Assim, a Equação (2.29) representa a transformação de coordenadas (translação + rotação) entre dois *frames*. A transformação inversa pode ser obtida pré-multiplicando ambos os lados da Equação (2.29) por R_1^{0T} , ou seja,

$$\mathbf{p}^1 = -R_1^{0T} \mathbf{o}_1^0 + R_1^{0T} \mathbf{p}^0 = -R_0^1 \mathbf{o}_1^0 + R_0^1 \mathbf{p}^0 \quad (2.30)$$

Buscando uma notação compacta acrescenta-se ao vetor \mathbf{p} uma quarta componente unitária formando o vetor $\tilde{\mathbf{p}}$

$$\tilde{\mathbf{p}} = \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} \quad (2.31)$$

Assim definindo,

$$A_1^0 = \begin{bmatrix} R_1^0 & \mathbf{o}_1^0 \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (2.32)$$

$$\tilde{\mathbf{p}}^0 = A_1^0 \tilde{\mathbf{p}}^1 \quad (2.33)$$

A matriz A_1^0 recebe o nome de **Transformação Homogênea**.

2.7 Cinemática direta

Um manipulador consiste de uma série de corpos rígidos (*links*) interligados por **juntas** (joints). Tais juntas podem ser de dois tipos: de revolução ou prismáticas (Figura 6). A estrutura completa é denominada **cadeia cinemática**. Toda cadeia cinemática projetada possui um **elemento terminal** (garra ou ferramenta) e uma **base**.

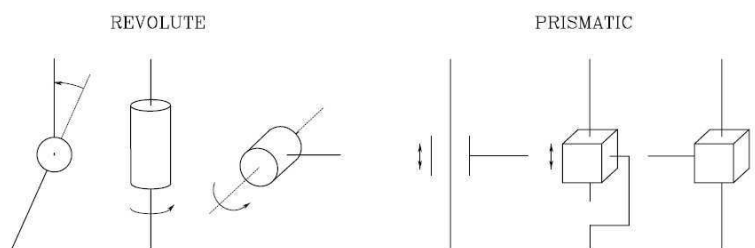


Figura 6 – Tipos de juntas: revolução e prismática

Do ponto de vista topológico uma cadeia cinemática é dita aberta quando os *links* não constituem um **loop** ao passo que as cadeias fechadas tem seus *links* como elementos constitutivos de um **loop**. Os Manipuladores são caracterizados pela quantidade de graus de liberdade (DOF – **Degrees of freedom**). Tal característica é determinada em função do número de juntas, assim um manipulador com três juntas possui 3 graus de liberdade, 4 juntas possui 4 graus de liberdade e assim por diante. Dito isso, definimos como objetivo da cinemática direta como o cálculo da pose do elemento terminal com base nos ângulos (revolução) / posições (prismáticas) das juntas. Assim, com relação a um *frame* de referência

$O_0 - x_0y_0z_0$, a função de cinemática direta é expressa pela matriz de transformação homogênea 2.34.

$$\mathbf{T}_e^0(\mathbf{q}) = \begin{bmatrix} \mathbf{n}_e^0(\mathbf{q}) & \mathbf{s}_e^0(\mathbf{q}) & \mathbf{a}_e^0(\mathbf{q}) & \mathbf{p}_e^0(\mathbf{q}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.34)$$

em que \mathbf{q} é o vetor ($n \times 1$) de valores das juntas, n_e^0 , s_e^0 , a_e^0 são vetores unitários ligados ao *frame* do elemento terminal e p_e^0 é o vetor posição da origem desse *frame* em relação a origem do *frame* base $O_0 - x_0y_0z_0$.

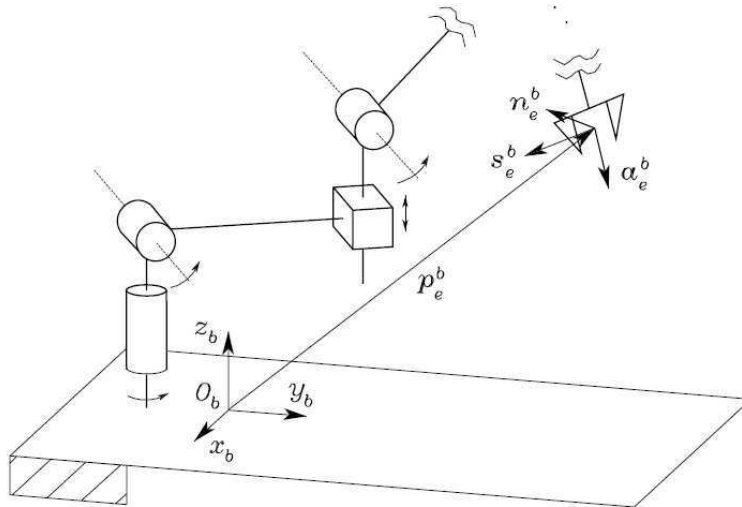


Figura 7 – Descrição dos vetores da matriz de transformação homogênea

Se o elemento terminal é uma garra então o vetor unitário a_e^0 é escolhido na direção de aproximação com o objeto, s_e^0 na direção de deslizamento das pinças da garra e n_e^0 é escolhido normal a a_e^0 e s_e^0 .

Exemplo 2.4:

Considere o braço robótico inteligente AX-12A (Figura 8). Baseado em trigonometria encontre a matriz de transformação homogênea.

Resolução

De forma a simplificar a obtenção da matriz de transformação homogênea considere-se $\beta = 0$ e denotando o ângulo de giro da base como ϕ (na Figura 8 tem-se $\phi = 0$).

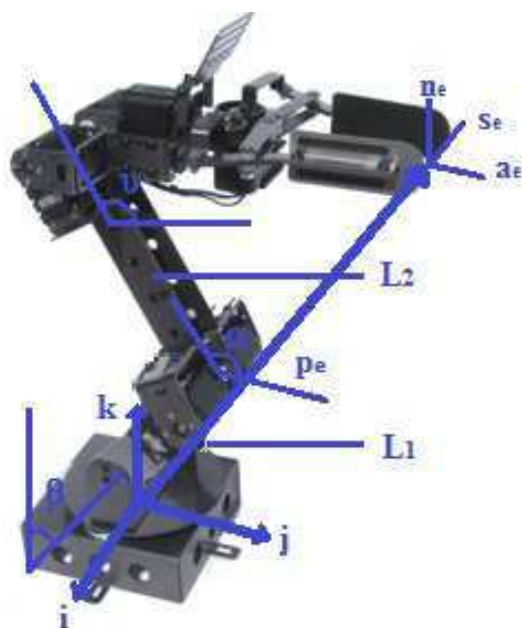


Figura 8 – Braço robótico inteligente AX-12A

Seja então,

$$\vec{L}_1 = \mathbf{i}L_1 \sin(\varphi) \cos(\alpha) + \mathbf{j}L_1 \cos(\varphi) \cos(\alpha) + \mathbf{k}L_1 \sin(\alpha)$$

Podemos escrever a haste \$L_2\$ como sendo:

$$\vec{L}_2 = \mathbf{i}L_2 \sin(\varphi) \cos(\alpha + \nu) + \mathbf{j}L_2 \cos(\varphi) \cos(\alpha + \nu) + \mathbf{k}L_2 \sin(\alpha + \nu)$$

É evidente que, a posição do ponto \$P\$, será dada pelo vetor \$\vec{P} = \vec{L}_1 + \vec{L}_2\$, uma vez que:

$$\vec{P} = \mathbf{i}X_p + \mathbf{j}Y_p + \mathbf{k}Z_p$$

Nota-se que:

$$X_p = \sin(\varphi)(L_1 \cos(\alpha) + L_2 \cos(\alpha + \nu))$$

$$Y_p = \cos(\varphi)(L_1 \cos(\alpha) + L_2 \cos(\alpha + \nu))$$

$$Z_p = L_1 \text{sen}(\alpha) + L_2 \text{sen}(\alpha + \nu)$$

Assim,

$$\mathbf{T}_e^0(\mathbf{q}) = \begin{bmatrix} \mathbf{n}_e^0(\mathbf{q}) & \mathbf{s}_e^0(\mathbf{q}) & \mathbf{a}_e^0(\mathbf{q}) & \mathbf{p}_e^0(\mathbf{q}) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_e^0(\mathbf{q}) = \begin{bmatrix} 0 & -1 & 0 & s(\varphi)(L_1 c(\alpha) + L_2 c(\alpha + \nu)) \\ 0 & 0 & 1 & c(\varphi)(L_1 c(\alpha) + L_2 c(\alpha + \nu)) \\ 1 & 0 & 0 & L_1 s(\alpha) + L_2 s(\alpha + \nu) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.8 Cinemática inversa

Resolver um problema de cinemática inversa consiste em obter os valores das variáveis das juntas a partir das coordenadas cartesianas do elemento terminal, assim sendo mostrar-se-á duas formas de resolvê-lo, a primeira consiste em utilizar a geometria euclidiana ao passo que a segunda usa conceitos da Lógica Fuzzy. É importante ter em mente que o primeiro ratifica a eficácia do segundo método. A título de esclarecimento as análises permanecerão sendo feitas sobre o manipulador Braço Robótico Inteligente AX-12A.

2.8.1 Método geométrico

Adotando a articulação dotada de dois atuadores inferior como referência podemos decompor o braço robótico em apenas dois *links* cujos ângulos denotaremos por θ_1 (ângulo de rotação da base), θ_2 e θ_3 , além disso denotaremos o comprimento de cada um dos *links* como l_1 e l_2 . Conforme ilustra a Figura 9.

Pode-se chegar facilmente a partir da aplicação da Lei dos Cossenos as seguintes expressões:

$$\theta_1 = \arctan\left(\frac{y_{pose}}{x_{pose}}\right) \quad (2.35)$$

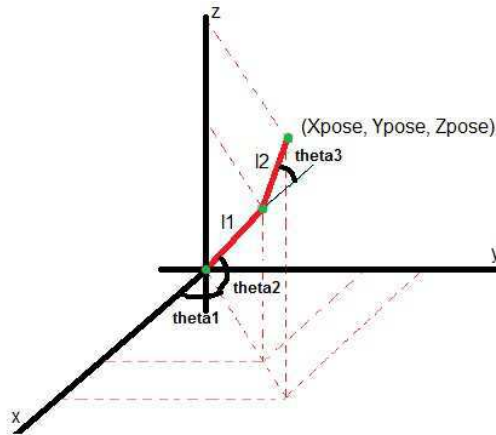


Figura 9 – Modelo geométrico do braço robótico

$$\theta_2 = \arctan\left(\frac{z_{pose}}{\sqrt{x_{pose}^2 + y_{pose}^2}}\right) - \arccos\left(\frac{x_{pose}^2 + y_{pose}^2 + z_{pose}^2 + l_1^2 - l_2^2}{2l_1\sqrt{x_{pose}^2 + y_{pose}^2 + z_{pose}^2}}\right) \quad (2.36)$$

$$\theta_3 = \arccos\left(\frac{x_{pose}^2 + y_{pose}^2 + z_{pose}^2 + l_1^2 - l_2^2}{2l_1l_2}\right) \quad (2.37)$$

2.8.2 Método usando lógica Fuzzy

Usando lógica Fuzzy foi construído um sistema Fuzzy de inferência que deduz a cinemática inversa a partir da cinemática direta. Além disso, a solução Fuzzy é facilmente entendível e não requer elevado conhecimento e compreensão da área Fuzzy. As coordenadas (x, y, z) são mapeadas para os ângulos $(\theta_1, \theta_2, \theta_3)$. Esse treinamento, denominado rede ANFIS (Adaptive Neuro-Fuzzy Inference System) é então usado como parte de um sistema de controle maior de forma a controlar o braço. Assim, conhecendo a localização desejada do braço robótica, o sistema de controle usa a rede treinada ANFIS para deduzir as posições angulares das juntas de forma a mover o braço para a localização desejada. A geração dos dados aplicou-se a posição angular das juntas variando de pequenos incrementos, respeitando as limitações impostas pelos atuadores. , nas equações da cinemática direta dadas a

seguir:

$$x_{pose} = (l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)) \sin(\theta_3) \quad (2.38)$$

$$y_{pose} = (l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)) \cos(\theta_3) \quad (2.39)$$

$$z_{pose} = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) \quad (2.40)$$

Cuja visualização dos pontos alcançáveis pelo braço robótico encontra-se na Figura 10.

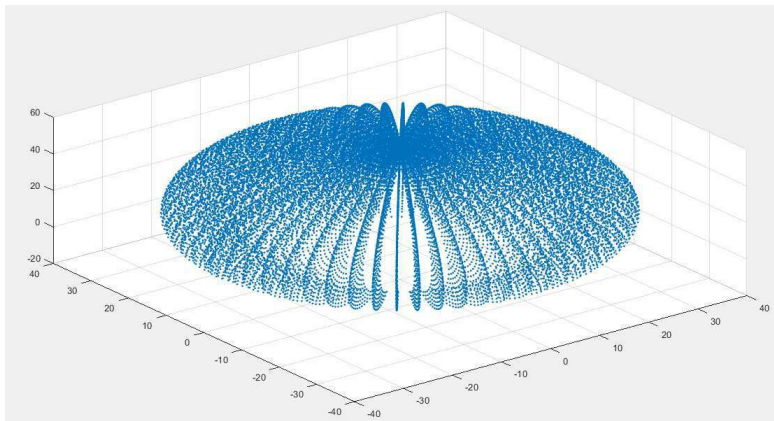


Figura 10 – Posições alcançáveis pelo braço robótico

2.8.3 Comparação entre o Método Geométrico e o Fuzzy

No intuito de validar o método obtido pela técnica Fuzzy elaborou-se um pequeno script no MatLab de forma que uma vez selecionados 150 amostras calculou-se o erro obtido da diferença entre o valor obtido geometricamente e o valor obtido via lógica Fuzzy.

Como é possível perceber para algumas amostras o erro foi considerável (cerca de $\frac{\pi}{12} rad$) enquanto para outras foi pequeno (cerca de $\frac{\pi}{72} rad$). Tal resultado leva a crer que o método Fuzzy pode ser utilizado apenas nos casos em que a solução geométrica é inviável tendo em vista sua relativa precisão como uma solução provisória.

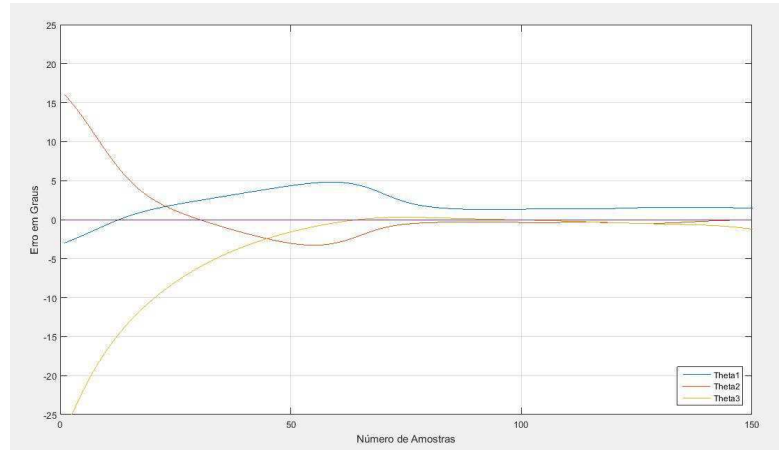


Figura 11 – Comparação entre o método geométrico e o Fuzzy

2.9 Convenção de Denavit-Hartenberg

Uma convenção comumente utilizada em aplicações de robótica é a **Denavit-Hartenberg** ou simplesmente **convenção DH**. Nessa convenção cada matriz de transformação homogênea A_i^{i-1} é representada como um produto de quatro transformações básicas.

$$A_i^{i-1} = Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,\alpha_i} \quad (2.41)$$

$$A_i^{i-1} = \begin{bmatrix} c(\theta_i) & -s(\theta_i) & 0 & 0 \\ s(\theta_i) & c(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & d_i \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c(\alpha_i) & -s(\alpha_i) & 0 \\ 0 & s(\alpha_i) & c(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_i^{i-1} = \begin{bmatrix} c(\theta_i) & -s(\theta_i)c(\alpha_i) & s(\theta_i)s(\alpha_i) & a_i c(\theta_i) \\ s(\theta_i) & c(\theta_i)c(\alpha_i) & -c(\theta_i)s(\alpha_i) & a_i s(\theta_i) \\ 0 & s(\alpha_i) & c(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.42)$$

A matriz A_i (Equação (2.42)) será a matriz de transformação homogênea do $frame_i$ para o $frame_{i-1}$ após a determinação dos parâmetros DH definidos a seguir:

$$\theta_i \rightarrow \text{ângulo} \quad (2.43)$$

$$d_i \rightarrow \text{excentricidade} \quad (2.44)$$

$$a_i \rightarrow \text{comprimento} \quad (2.45)$$

$$\alpha_i \rightarrow \text{torção} \quad (2.46)$$

Diferentemente do que ocorre na grande maioria da literatura introduziremos a teoria da notação DH, de forma eficiente, através de dois exemplos.

Exemplo 2.5:

Considere o sistema da Figura 12 constituído por um pistão e *links* interligados por juntas de revolução e prismática.

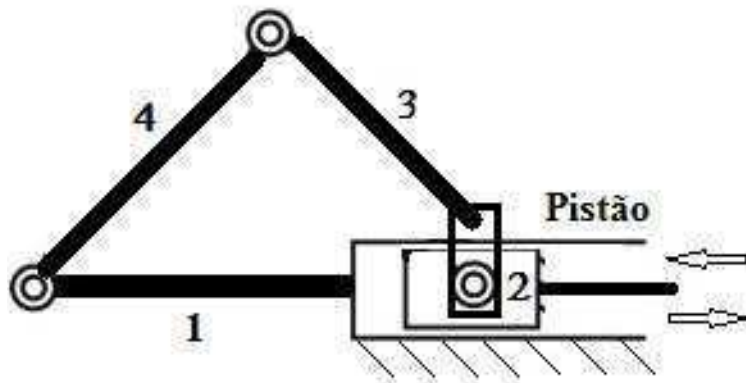


Figura 12 – Sistema constituído de 1 junta prismática e três de revolução

Dessa forma, encontre a matriz de transformação homogênea entre *frames* valendo-se da notação de *Denavit-Hartenberg*.

Resolução

- Passo 01: Identificar quais os tipos de juntas, sendo assim, observando a Figura 12, percebemos que os *links* estão conectados através juntas P_1 (junta do tipo prismática), R_2 , R_3 e R_4 (juntas do tipo revolução).
- Passo 02: Nas juntas prismáticas traça-se os eixos z na direção do seu eixo de deslocamento, ao passo que nas juntas de revolução traça-se os eixos z na direção do seu eixo de rotação. Assim, obtém-se a Figura 13.

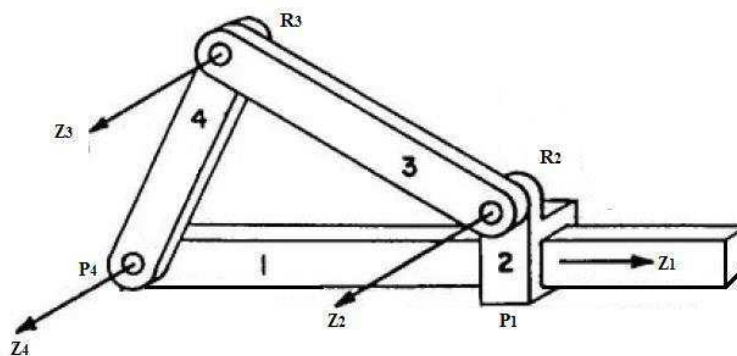
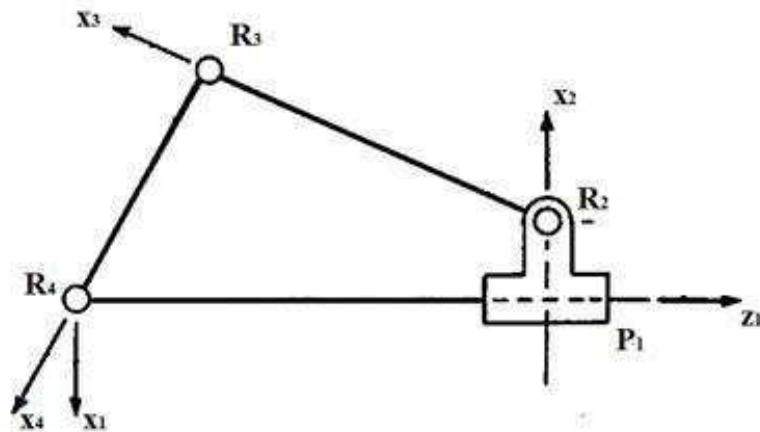
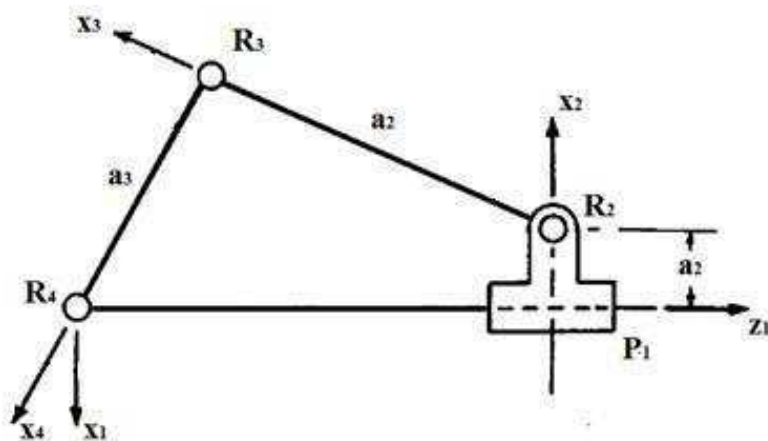


Figura 13 – Posicionamento dos eixos z

- Passo 03: Percebe-se para a presente estrutura que o ângulo entre z_1 e z_2 é $\frac{\pi}{2}$, ou seja, $\alpha_1 = \frac{\pi}{2}$; o ângulo entre z_2 e z_3 é 0, ou seja, $\alpha_2 = 0$; o ângulo entre z_3 e z_4 é 0, ou seja, $\alpha_3 = 0$; e, por fim, o ângulo entre z_4 e z_1 é $\frac{\pi}{2}$ ou seja, $\alpha_4 = \frac{\pi}{2}$.
- Passo 04: O eixo x_1 é definido ao longo da direção perpendicular a z_4 e z_1 ; o eixo x_2 é definido ao longo da direção perpendicular a z_1 e z_2 ; o eixo x_3 é definido ao longo da direção perpendicular a z_2 e z_3 ao longo do *link* 3; o eixo x_4 é definido ao longo da direção perpendicular a z_3 e z_4 ao longo do *link* 4. Assim, obtém-se a Figura 14.
- Passo 05: Os comprimentos a_i serão dados pela menor distância entre os prolongamentos de dois eixos z consecutivos, assim, tal distância entre z_4 e z_1 é zero, ou seja $a_4 = 0$; tal distância entre z_1 e z_2 é a_1 ; tal distância entre z_2 e z_3 é a_2 ; tal distância entre z_3 e z_4 é a_3 . Assim, obtém-se a Figura 15.

Figura 14 – Posicionamento dos eixos x Figura 15 – Ilustração contendo os comprimentos a_i

- Passo 06: Os ângulos entre eixos x consecutivos são designados pela letra θ_i . Assim, para a presente estrutura o ângulo entre x_1 e x_2 é $\theta_1 = \pi$; os ângulos entre x_2 e x_3 , x_3 e x_4 e x_4 e x_1 são θ_2 , θ_3 e θ_4 . Assim, obtém-se a Figura 16.
- Passo 07: Os comprimentos d_i serão dados pela menor distância entre os prolongamentos de dois eixos x consecutivos, assim, tal distância entre x_1 e x_4 é d_1 , as demais distâncias entre x_2 e x_3 , x_3 e x_4 e x_4 e x_1 são todas zero, isto é $d_2 = d_3 = d_4 = 0$. Assim, obtém-se a Figura 17.
- Passo 08: Uma vez determinados todos os parâmetros exigidos pela nota-

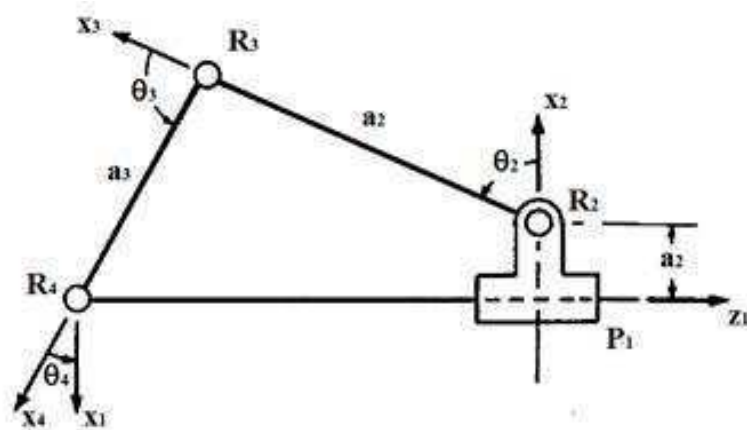


Figura 16 – Ilustração contendo os ângulos θ_i

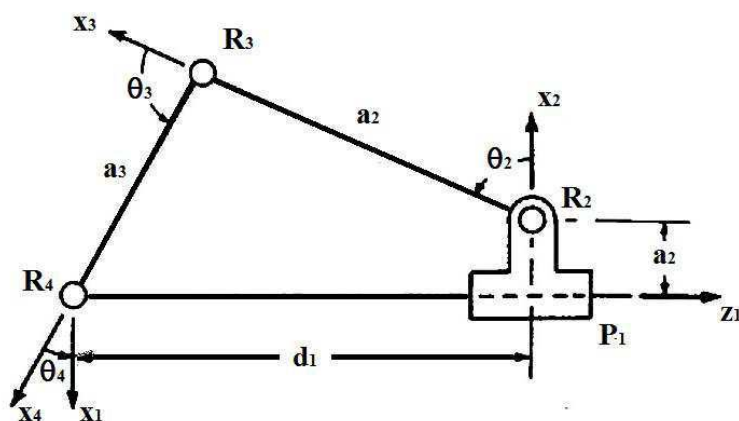


Figura 17 – Ilustração contendo a excentricidade d_1

ção DH, toma-se a expressão em (2.42) para determinar as transformações homogêneas entre \mathbb{S} , isto é,

$$A_2^1 = \begin{bmatrix} c(\pi) & -s(\pi)c(\frac{\pi}{2}) & s(\pi)s(\frac{\pi}{2}) & a_1c(\pi) \\ s(\pi) & c(\pi)c(\frac{\pi}{2}) & -c(\pi)s(\frac{\pi}{2}) & a_1s(\pi) \\ 0 & s(\frac{\pi}{2}) & c(\frac{\pi}{2}) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^1 = \begin{bmatrix} -1 & 0 & 0 & -a_1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c(\theta_2) & -s(\theta_2)c(0) & s(\theta_2)s(0) & a_2c(\theta_2) \\ s(\theta_2) & c(\theta_2)c(0) & -c(\theta_2)s(0) & a_2s(\theta_2) \\ 0 & s(0) & c(0) & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c(\theta_2) & -s(\theta_2) & 0 & a_2c(\theta_2) \\ s(\theta_2) & c(\theta_2) & 0 & a_2s(\theta_2) \\ 0 & s(\theta_2) & c(\theta_2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^3 = \begin{bmatrix} c(\theta_3) & -s(\theta_3)c(0) & s(\theta_3)s(0) & a_3c(\theta_2) \\ s(\theta_3) & c(\theta_3)c(0) & -c(\theta_3)s(0) & a_3s(\theta_2) \\ 0 & s(0) & c(0) & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^3 = \begin{bmatrix} c(\theta_3) & -s(\theta_3) & 0 & a_3c(\theta_3) \\ s(\theta_3) & c(\theta_3) & 0 & a_3s(\theta_3) \\ 0 & s(\theta_3) & c(\theta_3) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1^4 = \begin{bmatrix} c(\theta_4) & -s(\theta_4)c(0) & s(\theta_4)s(0) & a_4c(\theta_4) \\ s(\theta_4) & c(\theta_4)c(0) & -c(\theta_4)s(0) & a_4s(\theta_4) \\ 0 & s(0) & c(0) & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1^4 = \begin{bmatrix} c(\theta_4) & -s(\theta_4) & 0 & a_4c(\theta_4) \\ s(\theta_4) & c(\theta_4) & 0 & a_4s(\theta_4) \\ 0 & s(\theta_4) & c(\theta_4) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Exemplo 2.6:

Considere o Braço Robótico Inteligente AX-12A (Figura 18).



Figura 18 – Braço Robótico Inteligente AX-12A

Dessa forma, encontre a matriz de transformação homogênea entre *frames* valendo-se da notação de *Denavit-Hartenberg*.

Resolução

- Passo 01: Identificar quais os tipos de juntas, sendo assim, observando a Figura 18, percebemos que os *links* estão conectados através das juntas R_0 , R_1 , R_2 e R_3 (juntas do tipo revolução).
- Passo 02: Nas juntas prismáticas traça-se os eixos z na direção do seu eixo de deslocamento, ao passo que nas juntas de revolução traça-se os eixos z na direção do seu eixo de rotação. Assim, obtém-se a Figura 19.
- Passo 03: Percebe-se para a presente estrutura que o ângulo entre z_0 e z_1 é $\frac{\pi}{2}$, ou seja, $\alpha_1 = \frac{\pi}{2}$; o ângulo entre z_1 e z_2 é 0, ou seja, $\alpha_2 = 0$; o ângulo entre z_2 e z_3 é $\frac{\pi}{2}$, ou seja, $\alpha_3 = \frac{\pi}{2}$; e, por fim, o ângulo entre z_3 e z_4 é 0, ou seja, $\alpha_4 = 0$.
- Passo 04: O eixo x_1 é definido ao longo da direção perpendicular a z_4 e z_1 ; o eixo x_2 é definido ao longo da direção perpendicular a z_1 e z_2 ; o eixo x_3

Figura 19 – Posicionamento dos eixos z

é definido ao longo da direção perpendicular a z_2 e z_3 ao longo do *link* 3; o eixo x_4 é definido ao longo da direção perpendicular a z_3 e z_4 ao longo do *link* 4. Assim, obtém-se a Figura 20.

Figura 20 – Posicionamento dos eixos x

- Passo 05: Os comprimentos a_i serão dados pela menor distância entre os prolongamentos de dois eixos z consecutivos, assim, tal distância entre z_0 e z_1 é zero, ou seja $a_0 = 0$; tal distância entre z_1 e z_2 é $a_1 = 17,5\text{cm}$; tal distância entre z_2 e z_3 é $a_2 = 0$; tal distância entre z_3 e z_4 é $a_3 = 0$. Assim, obtém-se a Figura 21.

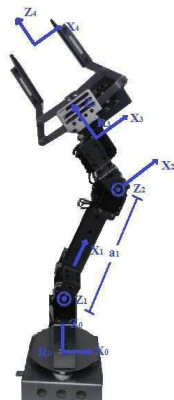


Figura 21 – Ilustração contendo os comprimentos a_i

- Passo 06: Os ângulos entre eixos x consecutivos são designados pela letra θ_i . Assim, para a presente estrutura o ângulo entre x_0 e x_1 é θ_1 ; os ângulos entre x_1 e x_2 é θ_2 , x_2 e x_3 é zero; entre x_3 e x_4 é zero. Assim, obtém-se a Figura 22.

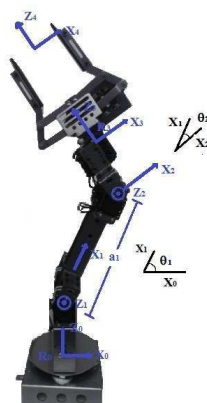


Figura 22 – Ilustração contendo os ângulos θ_i

- Passo 07: Os comprimentos d_i serão dados pela menor distância entre os prolongamentos de dois eixos x consecutivos, assim, tal distância entre x_0 e x_1 é $d_0 = 13$; entre x_1 e x_2 é $d_1 = 0$; entre x_2 e x_3 é $d_2 = 7,5cm$; entre x_3 e x_4 é $d_3 = 17$. Assim, obtém-se a Figura 23.

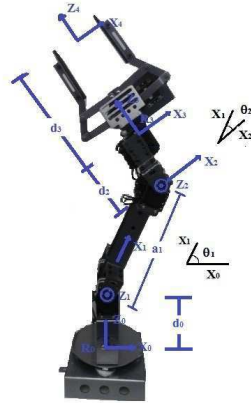


Figura 23 – Ilustração contendo as excentricidades d_i

- Passo 08: Uma vez determinados todos os parâmetros exigidos pela notação DH, toma-se a expressão em (2.42) para determinar as transformações homogêneas entre \mathfrak{S}_i , isto é,

$$A_1^0 = \begin{bmatrix} c(\theta_1) & -s(\theta_1)c(\frac{\pi}{2}) & s(\theta_1)s(\frac{\pi}{2}) & a_0c(\theta_1) \\ s(\theta_1) & c(\theta_1)c(\frac{\pi}{2}) & -c(\theta_1)s(\frac{\pi}{2}) & a_0s(\theta_1) \\ 0 & s(\frac{\pi}{2}) & c(\frac{\pi}{2}) & d_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_1^0 = \begin{bmatrix} c(\theta_1) & 0 & s(\theta_1) & 0 \\ s(\theta_1) & 0 & -c(\theta_1) & 0 \\ 0 & 1 & 0 & 13 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^1 = \begin{bmatrix} c(\theta_2) & -s(\theta_2)c(0) & s(\theta_2)s(0) & a_1c(\theta_2) \\ s(\theta_2) & c(\theta_2)c(0) & -c(\theta_2)s(0) & a_1s(\theta_2) \\ 0 & s(0) & c(0) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2^1 = \begin{bmatrix} c(\theta_2) & -s(\theta_2) & 0 & 17,5c(\theta_2) \\ s(\theta_2) & c(\theta_2) & 0 & 17,5s(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c(\theta_3) & -s(\theta_3)c(\frac{\pi}{2}) & s(\theta_3)s(\frac{\pi}{2}) & a_2c(\theta_2) \\ s(\theta_3) & c(\theta_3)c(\frac{\pi}{2}) & -c(\theta_3)s(\frac{\pi}{2}) & a_2s(\theta_2) \\ 0 & s(\frac{\pi}{2}) & c(\frac{\pi}{2}) & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^2 = \begin{bmatrix} c(\theta_3) & 0 & s(\theta_3) & 0 \\ s(\theta_3) & 0 & -c(\theta_3) & 0 \\ 0 & 1 & 0 & 7,5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^3 = \begin{bmatrix} c(\theta_4) & -s(\theta_4)c(0) & s(\theta_4)s(0) & a_3c(\theta_4) \\ s(\theta_4) & c(\theta_4)c(0) & -c(\theta_4)s(0) & a_3s(\theta_4) \\ 0 & s(0) & c(0) & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4^3 = \begin{bmatrix} c(\theta_4) & -s(\theta_4) & 0 & 0 \\ s(\theta_4) & c(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & 17 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3 Cinemática diferencial e estática

No capítulo anterior foram derivadas as equações de cinemática direta e inversa estabelecendo uma relação entre as variáveis das juntas e a *pose* do elemento terminal, neste capítulo será apresentada a cinemática diferencial que dá ênfase a obtenção das relações entre as velocidades das juntas e as velocidades angular e linear do elemento terminal. Tal mapeamento é descrito por uma matriz, denominada *Jacobiano geométrico* que depende da configuração do manipulador. Por fim é dada uma breve descrição de estática agregada a equação fundamental da estática para manipuladores.

3.1 Jacobiano geométrico

Para fins de entendimento considere um manipulador de n graus de liberdade. Sua equação de cinemática direta pode ser escrita conforme a Equação 3.1

$$\mathbf{T}_e(\mathbf{q}) = \begin{bmatrix} \mathbf{R}_e(\mathbf{q}) & \mathbf{p}_e(\mathbf{q}) \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (3.1)$$

em que $\mathbf{q} = [q_1 \dots q_n]^T$ é o vetor de variáveis das juntas. Sendo $\dot{\mathbf{p}}_e$ a velocidade linear e $\boldsymbol{\omega}_e$ a velocidade angular do elemento terminal do manipulador demonstrar-se-a que suas relações com as velocidades das juntas são:

$$\dot{\mathbf{p}}_e = \mathbf{J}_P(\mathbf{q}) \dot{\mathbf{q}} \quad (3.2)$$

$$\boldsymbol{\omega}_e = \mathbf{J}_O(\mathbf{q}) \dot{\mathbf{q}} \quad (3.3)$$

que pode ser reescrita como

$$\mathbf{v}_e = \begin{bmatrix} \dot{\mathbf{p}}_e \\ \boldsymbol{\omega}_e \end{bmatrix} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}} \quad (3.4)$$

que representa a *equação cinemática diferencial* do manipulador na qual a matriz \mathbf{J} é denotada como **Jacobiano geométrico**

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_O \end{bmatrix} \quad (3.5)$$

Antes de se calcular o Jacobiano geométrico, serão elucidadas algumas características das matrizes de rotação assim como importantes resultados da cinemática do corpo rígido.

3.2 Derivada de uma matriz de rotação

Seja uma matriz de rotação variável no tempo $\mathbf{R} = \mathbf{R}(t)$. Uma vez que elas são todas matrizes ortogonais, temos que

$$\mathbf{R}\mathbf{R}(t) = \mathbf{I} \quad (3.6)$$

diferenciando com respeito ao tempo, chega-se a identidade

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + \mathbf{R}(t)\dot{\mathbf{R}}^T(t) = \mathbf{0} \quad (3.7)$$

denotando

$$\mathbf{S}(t) = \dot{\mathbf{R}}(t)\mathbf{R}^T(t) \quad (3.8)$$

chegar-se-á

$$\mathbf{S} + \mathbf{S}^T = \mathbf{0} \quad (3.9)$$

Por (3.9) constata-se que \mathbf{S} é uma matriz anti-simétrica. Além disso, pós-multiplicando a Equação (3.8) por $\mathbf{R}(t)$ chega-se a

$$\dot{\mathbf{R}} = \mathbf{S}(t)\mathbf{R} \quad (3.10)$$

Tal equação (3.10) permite obter a derivada de uma matriz R como uma função dela própria. Agora, considere um vetor constante \mathbf{p}' e o vetor $\mathbf{p}(t) = \mathbf{R}(t).\mathbf{p}'$. Sua derivada em relação ao tempo será então

$$\dot{\mathbf{p}}(t) = \dot{\mathbf{R}}(t)\mathbf{p}' \quad (3.11)$$

que, valendo-se de (3.10), pode ser reescrita como

$$\dot{\mathbf{p}}(t) = \mathbf{S}(t)\mathbf{R}(t)\mathbf{p}' \quad (3.12)$$

Se o vetor $\boldsymbol{\omega}(t)$ denota a *velocidade angular* do *frame* $\mathbf{R}(t)$ em relação ao *frame de referência* no tempo t , temos da mecânica que

$$\dot{\mathbf{p}}(t) = \boldsymbol{\omega}(t) \times \mathbf{R}(t)\mathbf{p}' \quad (3.13)$$

Comparando as equações (3.12) e (3.13) percebe-se que o operador \mathbf{S} descreve o produto vetorial entre $\boldsymbol{\omega}$ e o vetor $\mathbf{R}(t)\mathbf{p}'$. Em outras palavras, a matriz \mathbf{S} é

$$\mathbf{S} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (3.14)$$

Exemplo 3.1:

Sejam as matrizes de rotação em x e y obtidas no Exemplo 2.2. Se φ e ϑ são funções do tempo t calcula-se as respectivas derivadas das matrizes em relação a t . Dado:

$$R_x(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\varphi) & -s(\varphi) \\ 0 & s(\varphi) & c(\varphi) \end{bmatrix}, \quad R_y(\vartheta) = \begin{bmatrix} c(\vartheta) & 0 & s(\vartheta) \\ 0 & 1 & 0 \\ -s(\vartheta) & 0 & c(\vartheta) \end{bmatrix} \quad (3.15)$$

Resolução

- Para R_x :

$$\mathbf{S}(t) = \dot{\mathbf{R}}(t)\mathbf{R}^T(t)$$

$$\mathbf{S}(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\dot{\varphi}s(\varphi) & -\dot{\varphi}c(\varphi) \\ 0 & \dot{\varphi}c(\varphi) & -\dot{\varphi}s(\varphi) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\varphi) & s(\varphi) \\ 0 & -s(\varphi) & c(\varphi) \end{bmatrix}$$

$$\mathbf{S}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\dot{\varphi} \\ 0 & \dot{\varphi} & 0 \end{bmatrix}$$

De acordo com a Equação (3.14) $\boldsymbol{\omega} = [\dot{\varphi} \ 0 \ 0]^T$ neste caso.

- Para R_y :

$$\mathbf{S}(t) = \dot{\mathbf{R}}(t)\mathbf{R}^T(t)$$

$$\mathbf{S}(t) = \begin{bmatrix} -\dot{\vartheta}s(\vartheta) & 0 & \dot{\vartheta}c(\vartheta) \\ 0 & 0 & 0 \\ -\dot{\vartheta}c(\vartheta) & 0 & -\dot{\vartheta}s(\vartheta) \end{bmatrix} \begin{bmatrix} c(\vartheta) & 0 & -s(\vartheta) \\ 0 & 1 & 0 \\ s(\vartheta) & 0 & c(\vartheta) \end{bmatrix}$$

$$\mathbf{S}(t) = \begin{bmatrix} 0 & 0 & \dot{\vartheta} \\ 0 & 0 & 0 \\ -\dot{\vartheta} & 0 & 0 \end{bmatrix}$$

De acordo com a Equação (3.14) $\boldsymbol{\omega} = [\dot{\varphi} \ 0 \ 0]^T$ neste caso.

Derivando a Equação (2.29) em relação ao tempo obtém-se

$$\dot{\mathbf{p}}^0 = \dot{\mathbf{o}}_1^0 + \mathbf{R}_1^0 \dot{\mathbf{p}}^1 + \mathbf{S}(\boldsymbol{\omega}_1^0) \mathbf{R}_1^0 \mathbf{p}^1 \quad (3.16)$$

por fim, denotando o vetor $\mathbf{R}_1^0 \dot{\mathbf{p}}^1$ por r_1^0 , ou seja,

$$\dot{\mathbf{p}}^0 = \dot{\mathbf{o}}_1^0 + \mathbf{R}_1^0 \dot{\mathbf{p}}^1 + \boldsymbol{\omega}_1^0 \times \mathbf{r}_1^0 \quad (3.17)$$

3.3 Velocidades dos links

Seja um *link* genérico i pertencente a um manipulador com uma cadeia cinemática aberta. De acordo com a convenção Denavit-Hartenberg adotada na seção 2.8, o *link* i conecta a juntas i e $i+1$; O *frame* i é ligado ao *link* i e origem sobre a junta $i+1$, enquanto o *frame* $i-1$ tem sua origem sobre a junta i , conforme Figura.

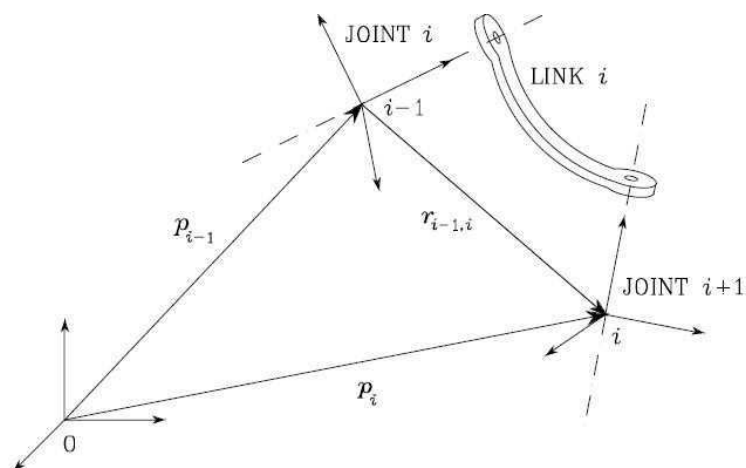


Figura 24 – Caracterização de um *link i* genérico

Assim sendo sejam \mathbf{p}_{i-1} e \mathbf{p}_i os vetores posição das origens dos $\mathbb{E} i-1$ e i , respectivamente. Seja também $r_{i-1,i}^{i-1}$ a posição do *frame i* em relação ao *frame i-1* no *frame i-1*. Então,

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \mathbf{v}_{i-1,i} + \boldsymbol{\omega}_{i-1} \times \mathbf{r}_{i-1,i} \quad (3.18)$$

que fornece a velocidade linear do *link i* como uma função das velocidades de translação e rotação do *link i-1*. Note ainda que $\mathbf{v}_{i-1,i}$ denota a velocidade da origem do *frame i* em relação ao *frame i-1*.

Já para a velocidade angular temos que

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_{i-1} + \boldsymbol{\omega}_{i-1,i} \quad (3.19)$$

em que $\boldsymbol{\omega}_{i-1,i}$ denota a velocidade da origem do *link i* em relação ao *link i-1*.

Obviamente as relações (3.18) e (3.19) depende dos tipos das juntas. É o que tratar-se-á a seguir.

- **Junta prismática** - Uma vez que a orientação do *frame i* não varia em relação ao *frame i-1* temos que $\boldsymbol{\omega}_{i-1,i} = 0$ e assim

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_{i-1} \quad (3.20)$$

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \dot{d}_i \mathbf{z}_{i-1} + \boldsymbol{\omega}_{i-1} \times \mathbf{r}_{i-1,i} \quad (3.21)$$

- Junta de Revolução - As expressões das velocidades angular e linear se tornam respectivamente

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_{i-1} + \dot{\vartheta}_i \mathbf{z}_{i-1} \quad (3.22)$$

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \boldsymbol{\omega}_{i-1} \times \mathbf{r}_{i-1,i} \quad (3.23)$$

3.4 Determinação do Jacobiano

Buscando simplificar, dar-se-á diretamente a expressão geral do Jacobiano, isto é,

$$\begin{bmatrix} \mathbf{J}_{Pi} \\ \mathbf{J}_{Oi} \end{bmatrix} = \begin{cases} \begin{bmatrix} \mathbf{z}_{i-1} \\ 0 \end{bmatrix}, & \text{para uma junta prismática;} \\ \begin{bmatrix} \mathbf{z}_{i-1} \times (\mathbf{p}_e - \mathbf{p}_{i-1}) \\ \mathbf{z}_{i-1} \end{bmatrix}, & \text{para uma junta de revolução;} \end{cases} \quad (3.24)$$

De maneira particular, \mathbf{z}_{i-1} equivale a terceira coluna da matriz de rotação R_{i-1}^0 ao passo que \mathbf{p}_e corresponde aos três primeiros elementos da quarta coluna da matriz de transformação R_e^0 .

Exemplo 3.2

Determine os parâmetros de Denavit-Hartenberg do manipulador planar da Figura 25 assim como a expressão do Jacobiano.

Resolução

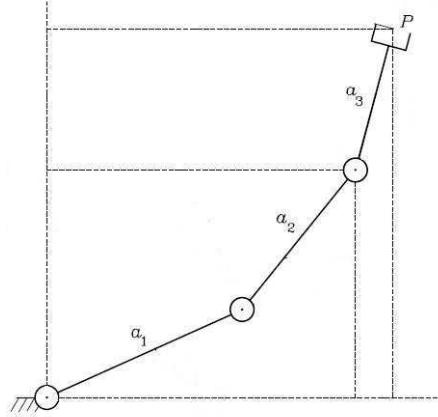
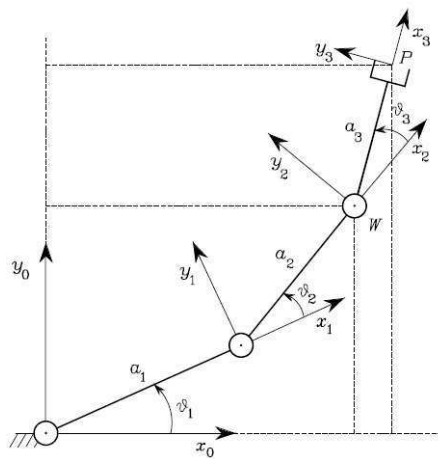
Figura 25 – Manipulador planar dotado de três *links*Figura 26 – Representação dos parâmetros DH do manipulador dotado de três *links*

Tabela 1 – Parâmetros DH do manipulador dotado de três *links*

<i>Link</i>	a_i	α_i	d_i	ϑ_i
1	a_1	0	0	ϑ_1
2	a_2	0	0	ϑ_2
3	a_3	0	0	ϑ_3

Realizando procedimento análogo ao trabalhado na Seção 2.8 obtém-se o seguinte conjunto de parâmetros DH (Tabela 1) e representação na Figura 26.

Observando a Tabela 1 percebe-se que a matriz de transformação homogênea tem a mesma estrutura para todas as juntas, isto é,

$$\mathbf{A}_i^{i-1} = \begin{bmatrix} c(\vartheta_i) & -s(\vartheta_i) & 0 & a_i c(\vartheta_i) \\ s(\vartheta_i) & c(\vartheta_i) & 0 & a_i s(\vartheta_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad i = 1, 2, 3. \quad (3.25)$$

Assim, neste caso, da Equação (3.24) chega-se ao Jacobiano dado por:

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \mathbf{z}_0 \times (\mathbf{p}_3 - \mathbf{p}_0) & \mathbf{z}_1 \times (\mathbf{p}_3 - \mathbf{p}_1) & \mathbf{z}_2 \times (\mathbf{p}_3 - \mathbf{p}_2) \\ \mathbf{z}_0 & \mathbf{z}_1 & \mathbf{z}_2 \end{bmatrix}$$

O cálculo da posição vetorial dos vários *links* leva a

$$\mathbf{p}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{p}_1 = \begin{bmatrix} a_1 c_1 \\ a_1 s_1 \\ 0 \end{bmatrix}$$

$$\mathbf{p}_2 = \begin{bmatrix} a_1 c_1 + a_2 c_{12} \\ a_1 s_1 + a_2 s_{12} \\ 0 \end{bmatrix} \quad \mathbf{p}_3 = \begin{bmatrix} a_1 c_1 + a_2 c_{12} + a_3 c_{123} \\ a_1 s_1 + a_2 s_{12} + a_3 s_{123} \\ 0 \end{bmatrix}$$

Por outro lado,

$$\mathbf{z}_0 = \mathbf{z}_1 = \mathbf{z}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Assim,

$$\mathbf{J} = \begin{bmatrix} -a_1s_1 - a_2s_{12} - a_3s_{123} & -a_2s_{12} - a_3s_{123} & -a_3s_{123} \\ a_1c_1 + a_2c_{12} + a_3c_{123} & a_2c_{12} + a_3c_{123} & a_3c_{123} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Exemplo 3.3

Determine a matriz Jacobiano para o Braço Robótico Inteligente AX-12A tratado anteriormente no Exemplo 2.6.

Resolução

Baseando-se nos resultados obtidos no Exemplo 2.6 chegamos a Tabela 2.

Tabela 2 – Parâmetros DH do manipulador dotado de três *links*

<i>Link</i>	a_i	α_i	d_i	ϑ_i
1	0	$\frac{\pi}{2}$	13	ϑ_1
2	17.5	0	0	ϑ_2
3	0	$\frac{\pi}{2}$	7.5	0
4	0	0	17	0

Assim, neste caso, da Equação (3.24) chega-se ao Jacobiano dado por:

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} \mathbf{z}_0 \times (\mathbf{p}_4 - \mathbf{p}_0) & \mathbf{z}_1 \times (\mathbf{p}_4 - \mathbf{p}_1) & \mathbf{z}_2 \times (\mathbf{p}_4 - \mathbf{p}_2) & \mathbf{z}_3 \times (\mathbf{p}_4 - \mathbf{p}_3) \\ \mathbf{z}_0 & \mathbf{z}_1 & \mathbf{z}_2 & \mathbf{z}_3 \end{bmatrix} \quad (3.26)$$

O cálculo da posição vetorial dos vários *links* leva a

$$\mathbf{p}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{p}_1 = \begin{bmatrix} 0 \\ 0 \\ 13 \end{bmatrix}$$

$$\mathbf{p}_2 = \begin{bmatrix} 17.5c_1c_2 \\ 17.5s_1c_2 \\ 13 + 17.5s_2 \end{bmatrix} \quad \mathbf{p}_3 = \begin{bmatrix} c_1(17.5c_2 + 13c_{23}) \\ s_1(17.5c_2 + 7.5c_{23}) \\ 13 + 17.5s_2 + 7.5s_{23} \end{bmatrix} \quad \mathbf{p}_4 = \begin{bmatrix} c_1(17.5c_2 + 24.5c_{23}) \\ s_1(17.5c_2 + 24.5c_{23}) \\ 13 + 17.5s_2 + 24.5s_{23} \end{bmatrix}$$

Por outro lado,

$$\mathbf{z}_0 = \begin{bmatrix} s_1 \\ -c_1 \\ 0 \end{bmatrix} \quad \mathbf{z}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{z}_2 = \begin{bmatrix} s_3 \\ -c_3 \\ 0 \end{bmatrix} \quad \mathbf{z}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Assim, encontra-se o Jacobiano substituindo os valores acima na Equação (3.26).

Dessa forma, a fim de uma implementação computacional utilizamos a Equação (3.27).

$$\mathbf{q}(t_{k+1}) = \mathbf{q}(t_k) + \mathbf{J}^{-1}(\mathbf{q}(t_k))v_e(t)\Delta t \quad (3.27)$$

3.5 Estática

O objetivo da estática é determinar a relação entre a força generalizada aplicada ao elemento terminal e as forças aplicadas as juntas - forças para juntas prismáticas e torques para juntas de revolução - com o manipulador em um configuração de equilíbrio.

Sendo assim, seja $\boldsymbol{\tau}$ o vetor de torques nas juntas (forças generalizadas em juntas são normalmente chamadas de *torques*) e $\boldsymbol{\gamma}$ o vetor de forças no elemento terminal (forças generalizadas no elemento terminal são normalmente chamadas de *forces*). Então,

$$\boldsymbol{\tau} = \mathbf{J}^T(\mathbf{q})\boldsymbol{\gamma}_e \quad (3.28)$$

4 Geração de trajetória

Para execução de uma tarefa específica, é importante considerar as principais características do algoritmo de movimento. Dito isso, neste capítulo será dado enfoque a obtenção de entradas de referência para o sistema de controle de movimento que garantam que o manipulador realize o movimento desejado no espaço. Vale ressaltar que aqui a trajetória se refere a história da posição, velocidade e aceleração no tempo para cada grau de liberdade.

4.1 Considerações gerais sobre a descrição e geração de rotas

O problema básico consiste em mover o manipulador de uma posição inicial até uma posição final desejada. Os pontos da trajetória incluem os pontos ao longo da via, inicial e final. Por fim vale ressaltar que geralmente o movimento do manipulador é *uniforme*.

4.2 Técnicas para geração de trajetórias

Considere o problema de mover o elemento de sua posição inicial até uma posição de destino num determinado intervalo de tempo. Da cinemática inversa pode-se obter o conjunto de ângulos para cada junta correspondente a posição e orientação finais. A posição inicial do manipulador também se conhece na forma de um conjunto de ângulos. Assim, se faz necessário uma função para cada articulação, cujo valor em t_0 corresponda a posição inicial e cujo valor em t_f corresponda a posição final desta articulação. Logicamente, existem várias funções uniformes $\theta(t_f)$ como mostra a Figura (27).

Para gerar um movimento uniforme são necessárias quatro restrições sobre $\theta(t)$. Duas dessas consistem no estabelecimento das condições inicial e final, ao passo que as outras duas proveem do fato que a velocidade deve constante. Assim,

$$\theta(0) = \theta_0, \quad \theta(t_f) = \theta_f, \quad \dot{\theta}(0) = 0 \quad e \quad \dot{\theta}(t_f) = 0 \quad (4.1)$$

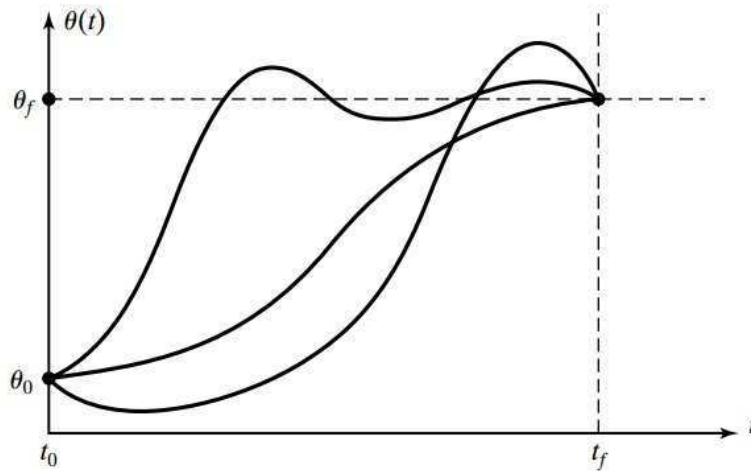


Figura 27 – Possíveis trajetórias para alcançar a posição final

Sendo assim, a fim de satisfazer essas quatro equações podemos tomar um polinômio de ordem mínima igual a três, isto é,

$$\theta(t) = a_0 + a_1t + a_2t^2 + a_3t^3 \quad (4.2)$$

de maneira que a velocidade e a aceleração da junta ao longo dessa trajetória é

$$\dot{\theta}(t) = a_1 + 2a_2t + 3a_3t^2 \quad e \quad \ddot{\theta}(t) = 2a_2 + 6a_3t \quad (4.3)$$

Se combinarmos as Equações (4.2) e (4.3) com as quatro restrições desejadas chegamos a

$$a_0 = \theta_0$$

$$a_1 = 0$$

$$a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0)$$

$$a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0)$$

Dessa forma, podemos determinar o polinômio cúbico que conecta qualquer posição inicial a final desejada.

Agora, reconsidere o caso anterior com a ressalva que a trajetória deva passar por pontos intermediários específicos antes de chegar ao destino final. Como no caso anterior, pode-se chegar a seguinte solução que atende as quatro restrições (Equação (4.2)).

$$a_0 = \theta_0$$

$$a_1 = \dot{\theta}_0$$

$$a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0) - \frac{2}{t_f}\dot{\theta}_0 - \frac{1}{t_f}\dot{\theta}_f$$

$$a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0) + \frac{1}{t_f^2}(\dot{\theta}_f + \dot{\theta}_0)$$

Dessa forma, podemos determinar o polinômio cúbico que conecta qualquer posição inicial a final desejada. Para isso aplica-se a cada segmento a fim de encontrar os polinômios requeridos.

4.3 Implementação das trajetórias no Manipulador Inteligente AX-12A

O problema da implementação das trajetórias foi resolvido de três maneiras:

- Utilizando o software MatLab em conjunto com a ferramenta RVC Tools;
- Utilizando o software MatLab em conjunto com o simulador V-REP;
- Utilizando ROS

Todavia antes de elucidar cada uma das implementação tratar-se-á um pouco sobre o Dynamixel AX-12A utilizado no braço robótico e sobre o controlador USB2Dynamixel.

1. Dynamixel AX-12A

O AX-12A trata-se de um atuador Dynamixel inteligente, modular que incorpora um redutor de velocidade, um motor de precisão DC e um circuito

de controle com a funcionalidade de rede, tudo em um único pacote. Apesar do seu tamanho compacto, o AX-12A pode produzir um binário elevado e é feito com materiais de elevada qualidade de forma a proporcionar a resistência e capacidade de resistência estrutural necessária para suportar grandes forças externas. Além disso, o AX-12A tem a capacidade de detectar e agir de acordo com as condições internas, como mudanças na temperatura interna ou tensão de alimentação.

Possui um controle de precisão de 1024 níveis com faixa de posicionamento de 300° (i.e., uma resolução de 0.2930 graus/nível) possibilidade de trabalhar com velocidades de comunicação de até 1Mb.

Tabela 3 – Características físicas e elétricas do dynamixel AX-12A

<i>Características</i>	Especificações
Peso(g)	55
Taxa de redução	1/254
Alimentação	entre 7 e 10V
Torque máx. (kgf.cm)	12 (7V) e 16.5 (10V)
Resolução	0.2930°
Faixa de operação	0° a 300°
Corrente máx.	900mA
Temperatura de oper.	-5°C 85°C
Tipo de protocolo	Serial Half-Duplex Assín. (8 bit,1 par,sem par.)
ID	254 (0 254 ID)
Velocidade de com.	7343bps 1Mbps

Para comunicar-se estabelece uma conexão serial TTL do tipo Half-Duplex, em que o AX-12A identificado por um ID único atua como um servidor que recebe comandos e retorna respostas ao controlador externo (PC). Na Figura 28 a seguir, é possível contemplar a geometria do AX-12A assim como a disposição de seus terminais.

Na comunicação o controlador principal comunica-se com os Dynamixel enviando e recebendo pacotes de informação. Existem dois tipos de pacotes que são os “Pacotes de Instrução” (enviados do controlador principal para os Dynamixels) e os “Pacotes de Status” (enviados do Dynamixel para o controlador principal). Assim, se o controlador principal envia um pacote

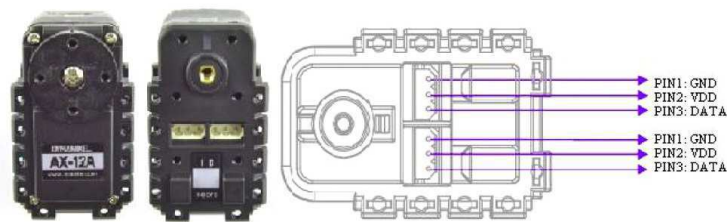


Figura 28 – Aspecto físico e pinagem do AX-12A

de instrução com um ID do conjunto N, o Dynamixel com esse valor de ID retorna o respectivo pacote de status e realiza a instrução requerida. Se mais de uma unidade do Dynamixel possui o mesmo ID então haverá envio simultâneos de pacote gerando problemas de comunicação.

2. USB2Dynamixel O dispositivo utilizado para controlar os dynamixels presentes no braço robótico foi o USB2Dynamixel a partir da porta USB de um computador. Tal dispositivo possui conexão 3P para dynamixels da série AX assim como 4P para dispositivos das séries DX e RX. Além disso, o USB2Dynamixel pode ser utilizada para transformar uma porta USB em uma porta serial para um PC sem uma porta serial tal como em um notebook.

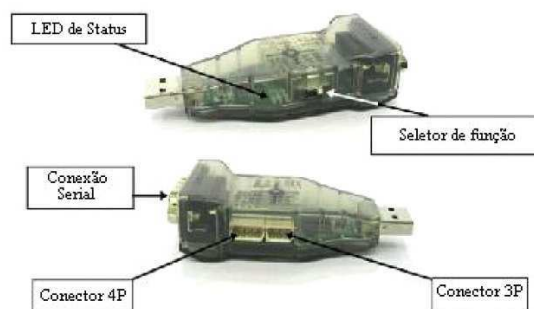


Figura 29 – Aspecto físico do USB2Dynamixel

A Figura 31 mostra a pinagem do USB2Dynamixel.



Figura 30 – Conexão entre o USB2Dynamixel e o AX-12A

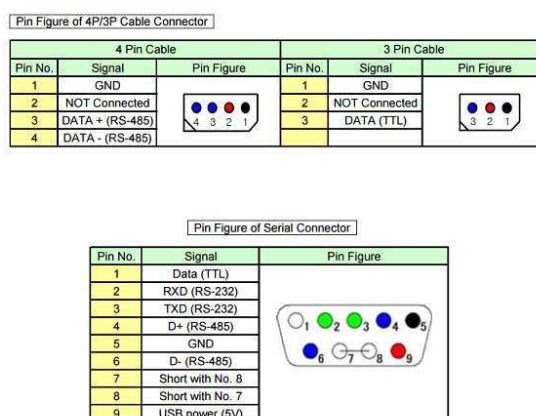


Figura 31 – Pinagem do USB2Dynamixel

4.3.1 Implementação via MatLab e RVC Tools

Para estabelecer a comunicação entre o MatLab e os AX-12A deve-se utilizar a biblioteca Dynamixel SDK cuja descrição é dada a seguir.

1. Dynamixel SDK

A fim de realizar o controle utilizou-se a biblioteca SDK, escrita em C, que apesar de seu número limitado de funções facilitou consideravelmente tal controle. O fato dela ser escrita em C e ser padronizada garante a portabilidade e versatilidade frente a diversos controladores em uma mesma aplicação. Antes de exemplificar o uso é importante estar atento aos passos necessários para a correto uso da biblioteca Dynamixel SDK descritos a seguir:

- Disponível na página <http://www.robotis.us/dynamixel-sdk/>
- Extrair o arquivo *dxl_sdk_win32_v1_02.zip*
- Copiar para a pasta do script, que fará o controle do braço, os arquivos *dynamixel.dll*, *dynamixel.h*, *ControlTable.m* e *MyDynamixel.m*.

As principais funções contidas na biblioteca são:

- *Exit()* – Fecha a comunicação entre o PC e os *dynamixel*'s;
- *Init()* – Inicia a comunicação entre o PC e os *dynamixel*'s;
- *addDevice()* – Instancia *dynamixel*;
- *removeDevice()* – Destrói a instancia do *dynamixel*;
- *writeAngle()* – Altera a posição angular do servo;
- *setSpeed()* – Altera a velocidade do servo.

Para o efetivo controle do braço se fez necessário incluir/elaborar algumas funções dentro do script *MyDynamixel.m* descritas a seguir

- *readAngle()* – Lê a posição angular do servo;
- *movendoouparado()* – Informa se o eixo do servo está em movimento ou não;
- *Torque()* – Ativa ou desativa o torque do servo;
- *LedOn()* – Aciona led do AX-12A;
- *LedOff()* – Desliga led do AX-12A.

Uma vez configurada a biblioteca, elaborou-se um script no MatLab tendo em vista apresentar as principais funções da biblioteca acrescido de comentários explicativos, dado a seguir.

```
clc %Limpa a Janela de comandos
clear %Limpa o Workspace
usb2dynamixel_id = 1; %número atribuído ao USB2Dynamixel que enviará e
%receberá dados dos dynamixels
liec = MyDynamixel(dynamixel_id); %instancia o USB2Dynamixel no código
```

```

%e o nomeia como liec
liec.portNum = 3; %informa o número da porta ao qual o USB2Dynamixel
%está conectado
liec.baudNum = 1; %informa o valor do baudrate, no caso 1MB
liec.init(); %inicia a comunicação entre o MatLab e o USB2Dynamixel físico
liec.addDevice(1); %Instancia o dynamixel 1(no caso um AX – 12A)
liec.addDevice(7);
liec.setSpeed('id', 1, 'RPM', 10); %configura a velocidade do dynamixel 1 como
%10 cujo valor máximo é 16
liec.setSpeed('id', 7, 'RPM', 10);
while(1)
ufcg = input('Digite a posição angular desejada em graus : '); %Captura o
%valor da posição angular desejada
if strcmp(ufcg, 'dee') %Se ufcg for igual a string "dee"
break; %Encerre o loop
end
liec.writeAngle('id', 1, 'deg', ufcg); %Gireodynamixel01atéaposição"ufcg"
%graus
liec.writeAngle('id', 7, 'deg', ufcg); %Gireodynamixel07atéaposição"ufcg"
%graus
end

```

2. RVC Tools

Essa *toolbox* fornece muitas funções frequentemente utilizadas no campo da robótica, incluindo ferramentas envolvendo cinemática, dinâmica e geração de trajetórias. Com ela pode-se para simular e analisar resultados de experimentos feitos com robôs reais. Escrita em C, a *toolbox* é baseada em métodos muito gerais garantindo assim uma eficiente e versátil representação dos manipuladores, tais manipuladores correspondem a objetos no *MatLab*. Os *scripts* são de fácil entendimento, talvez em virtude da alta eficiência computacional. Além disso, a biblioteca possui funções capazes de lidar com vetores, transformações homogêneas necessários para representar em um espaço tridimensional a posição e a orientações dos links.

Disponível no site <http://petercorke.com/Home.html> a toolbox é de fácil utilização bastante para utilizar suas funções carregar e executar o arquivo *startup_vc.m* no *script* a ser elaborado. A Figura 32 a seguir ilustra um manipulador criado com a *RVC Tools*.

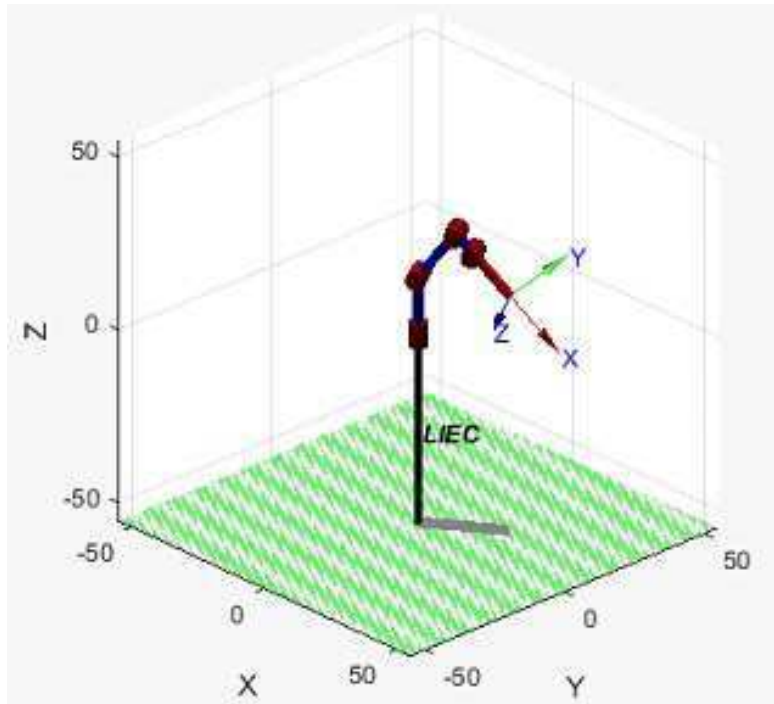


Figura 32 – Manipulador criado com RVC Tools

Obtidos os valores dos parâmetros DH para o Braço Robótico AX-12A implementou-se o seguinte código para uso da ferramenta RVC Tools:

```
clear

run('startup_vc')

L(1) = Link('d', 16.5, 'a', 0, 'alpha', pi/2); L(2) = Link('d', 0, 'a', 16.7, 'alpha', 0); L(3) =
Link('d', 0, 'a', 7, 'alpha', 0); L(4) = Link('d', 0, 'a', 15, 'alpha', 0);
BRACO = SerialLink(L, 'name', 'LIEC');
BRACO.plot([0pi/200])
```

O resultado de sua implementação na ferramenta *RVC Tools* encontram-se logo a seguir:

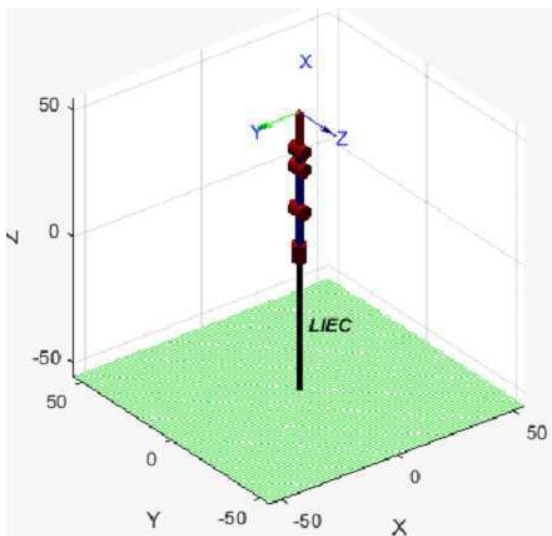


Figura 33 – Representação do braço robótico inteligente AX-12A usando RVC Tools

No intuito de facilitar os testes com o braço robótico elaborou-se uma interface valendo-se do tipo GUIDE disponível no MatLab, tal interface pode ser visualizada na figura a seguir.

Visando um melhor entendimento da interface haverá uma descrição do funcionamento dos principais botões a partir daqui.

- Botão “Iniciar Comunicação” O botão “Iniciar Comunicação” instancia um objeto do tipo MyDynamixel e atribui a ele o nome LIEC. Além disso, ele configura a porta e a velocidade da comunicação serial, inicia a comunicação e carrega os sete atuadores AX-12A do braço robótico. Por fim, exibe uma mensagem informando que a comunicação foi realizada com sucesso;
- O botão "Encerrar Comun." Libera o conjunto de atuadores carregados no botão "Iniciar Comun." e exibe uma mensagem confirmando que a sessão foi encerrada;

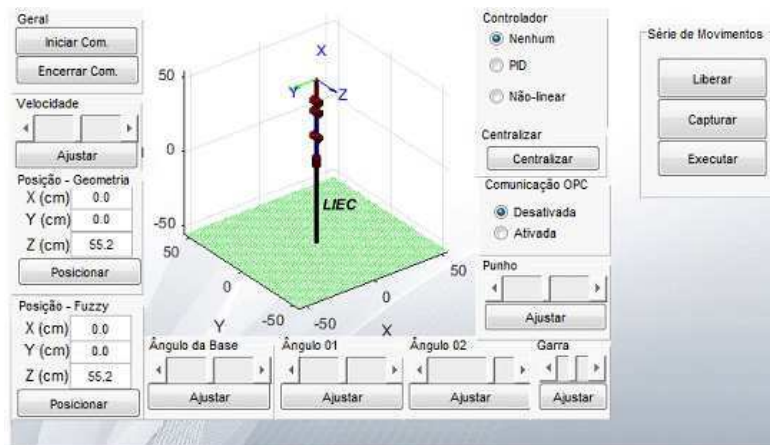


Figura 34 – Interface elaborada no GUIDE do MatLab

- Botão “Ajustar” da seção Velocidade Esse botão configura a velocidade do conjunto de atuadores em um valor entre 0 e 16, de forma que todos os AX-12A possuam a mesma velocidade.
- Botão “Posicionar” Independentemente do posicionamento ser pelo Método geométrico ou Fuzzy possui a mesma função que é posicionar o braço robótico levando em conta os valores das coordenadas X, Y e Z fornecidas pelo usuário. Além disso o movimento é reproduzido na tela da interface de maneira síncrona. A seguir é dado o código utilizado para implementar essa operação valendo-se do método geométrico.
- Botão “Centralizar” Esse botão fixa os ângulos dos atuadores em 150° de forma a obter a posição com maior coordenada Z.
- Botão “Liberar” Esse botão trata de desativa o torque dos AX-12A de forma que seja possível movimentar o braço livremente sem qualquer torque de resistência proveniente dos motores. Para isso foi preciso criar uma função Torque que ativa ou desativa o torque dos atuadores de acordo com a precisão.
- Botão “Capturar” Esse botão captura as posições angulares do AX-12A e armazena em um arquivo de texto.

- Botão “Executar” Esse botão executa o conjunto de leituras feitas pelo botão capturar.

4.3.2 Implementação via MatLab e V-REP

Além das simulações feitas com a ferramenta RVC Tools foram feitas algumas simulações com o ambiente de simulação V-REP de fácil integração com o MatLab. Para isso, inicialmente elaborou-se um modelo do braço de forma a atender as características físicas com relativa precisão, conforme ilustra a Figura 35.

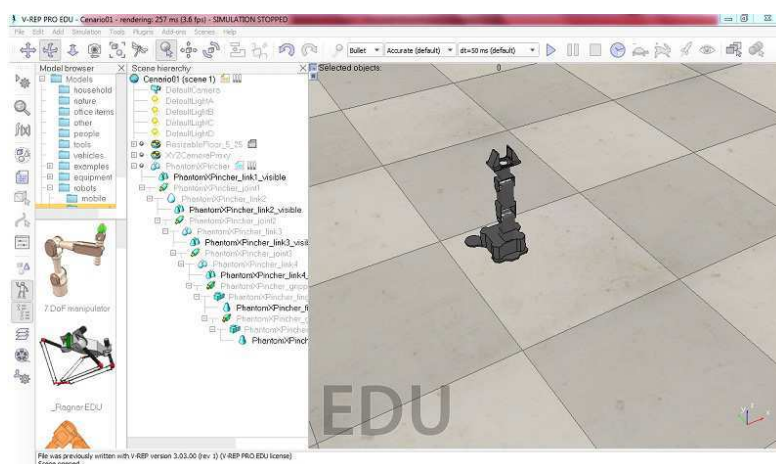


Figura 35 – Implementação do braço robótico inteligente AX-12A com V-REP

O controle simultâneo do V-REP e do braço robótico real se mostrou efetivo mas deixou a desejar no que tange velocidade de processamento, fato esperado em virtude da memória solicitado pelo processo de comunicação com o V-REP. Mesmo assim, pode-se dizer que o V-REP se aproxima mais da realidade em termos de design e pode muito bem ser integrado no controle do braço robótico usando linguagens mais leves como C, Lua, Python e ROS.

4.3.3 Implementação via ROS

Antes de mostrar a implementação será descrito um pouco sobre o ROS e como instalá-lo.

- ROS

ROS (Robot Operating System), ou Sistema Operacional de Robôs, disponibiliza bibliotecas e ferramentas para auxiliar desenvolvedores de software a criar aplicações robóticas. Ele fornece abstração de hardware, device drivers, bibliotecas, visualizadores, transmissão de mensagens, gerenciamento de pacotes e muito mais. ROS é licenciado sob uma licença livre (open source), a licença BSD.

- Como instalá-lo?

Execute os seguintes comandos no terminal Linux para instalar a **versão Kinetic do ROS**:

```
- sudo sh -c 'echo "debhttp : //packages.ros.org/ros/ubuntu$(lsb_release - sc)main" > /etc/apt/sources.list.d/ros - latest.list'
- sudo apt - key adv --keyserver hkp : //ha.pool.sks - key servers.net : 80 --recv - key 0xB01FA116
- sudo apt - get update
- sudo apt - get install ros - kinetic - desktop - full
- sudo rosdep init
- rosdep update
- echo "source /opt/ros/kinetic/setup.bash" > ~/.bashrc
- source ~/.bashrc
- sudo apt - get install python - rosinstall
```

Valendo-se do ROS, foi possível implementar um modelo 3D do Braço Robótico Inteligente AX-12A ilustrado na Figura 36.

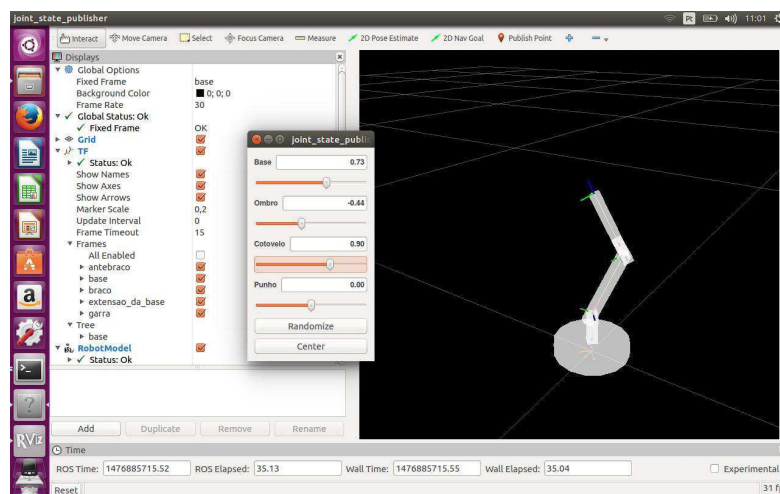


Figura 36 – Interface elaborada no RVIZ do ROS

5 Modelagem e identificação no Manipulador Inteligente AX-12A

A derivação de um modelo dinâmico de um manipulador é uma tarefa importantíssima para a simulação do movimento, análise estrutural e projeto de algoritmos de controle. Simular o movimento do manipulador permite que estratégias de controle e técnicas de movimento sejam testadas sem a necessidade de um sistema físico.

Todavia nem sempre é viável aferir todas as grandezas físicas do sistema sendo assim necessário realizar um processo de identificação de cada uma das juntas.

5.1 Modelagem

A modelagem mecânica do braço robótico foi realizada considerando a atuação de um servo por vez. Além disso desprezou-se as vibrações presentes na garra do braço robótico. De forma que os modelos obtidos foram para cada articulação encontram-se ilustrados nas Figuras 37, 38 e 39.

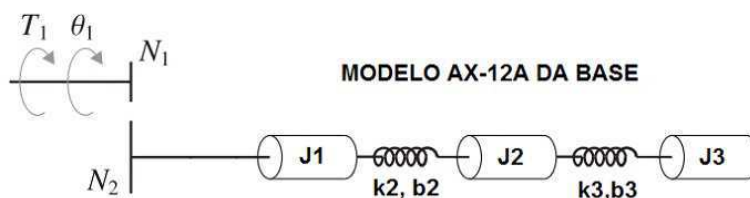


Figura 37 – Modelagem mecânica da base

5.2 Identificação

Demonstrar-se-a o desenvolvimento da função de transferência para o caso do cotovelo como forma de justificar a escolha de modelos de 2ª ordem numa eventual

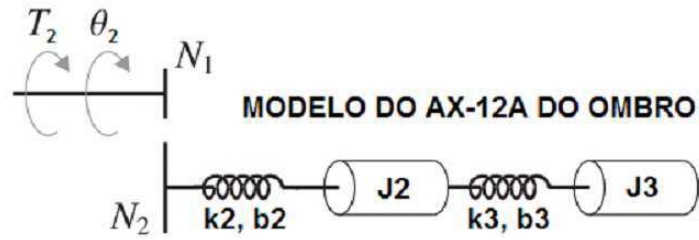


Figura 38 – Modelagem mecânica do ombro

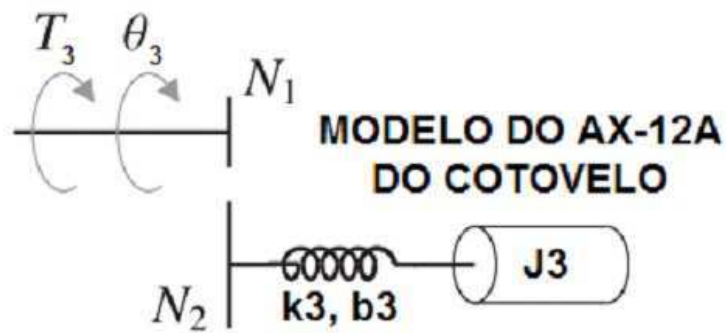


Figura 39 – Modelagem mecânica do cotovelo

identificação.

$$\theta_3 = \frac{N_1}{N_2} \theta_3 \quad (5.1)$$

$$T_3 = \frac{N_2}{N_1} T_3 \quad (5.2)$$

$$T_3 - k_3(\theta_3 - 0) - b_3(\dot{\theta}_3 - 0) = J_3 \ddot{\theta}_3 \quad (5.3)$$

Substituindo as equações (5.1) e (5.2) em (5.3) e rearranjando no *domínio S* chega-se a:

$$\frac{\theta_3(s)}{T_3(s)} = \frac{\left(\frac{N_2}{N_1}\right)^2}{J_2 s + b_3 s + k_3} \quad (5.4)$$

É importante frizar que a taxa de redução de cada servo, isto é $\frac{N_1}{N_2}$ é igual a $\frac{1}{254}$.

Sendo assim, no intuito de realizar a identificação submeteu-se cada articulação a um deslocamento de 70° no sentido positivo (anti-horário) de rotação. E se fez a leitura da posição angular para uma velocidade equivalente a 25% da máxima até o sistema entrar em regime permanente. Logo, para o servo da base obteu-se a seguinte resposta:

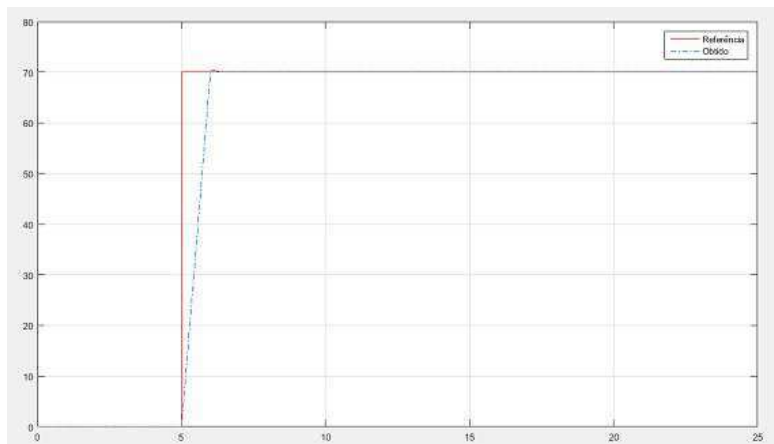


Figura 40 – Resposta do servo da base.

Para o conjunto de servos localizado no ombro observou-se o seguinte comportamento:

Para o conjunto de servos localizado no cotovelo observou-se o seguinte comportamento:

Como era de se esperar a oscilação do cotovelo tomou aspectos intermediários entre a resposta da base e do ombro. Valendo-se da Toolbox *Ident* do *MatLab*, chegou-se as seguintes funções de transferência:

$$H_{base}(s) = \frac{18.940}{s^2 + 3.557s + 19.104} \quad (5.5)$$

$$H_{ombro}(s) = \frac{37.152}{s^2 + 13.072s + 37.930} \quad (5.6)$$

$$H_{cotovelo}(s) = \frac{60.524}{s^2 + 12.343s + 61.002} \quad (5.7)$$

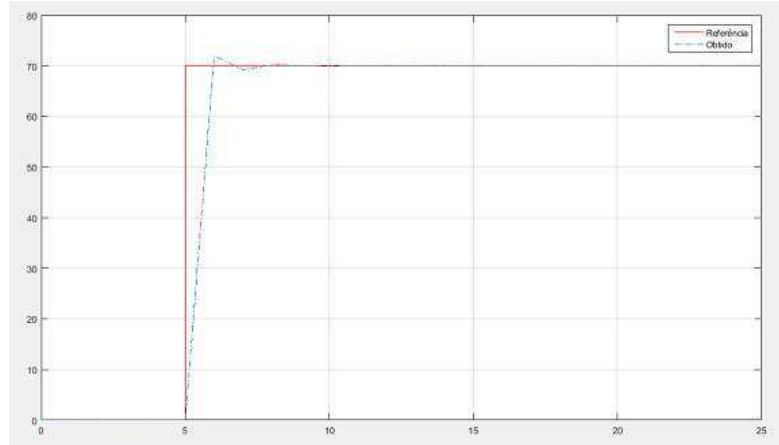


Figura 41 – Resposta do servo pertencente ao ombro

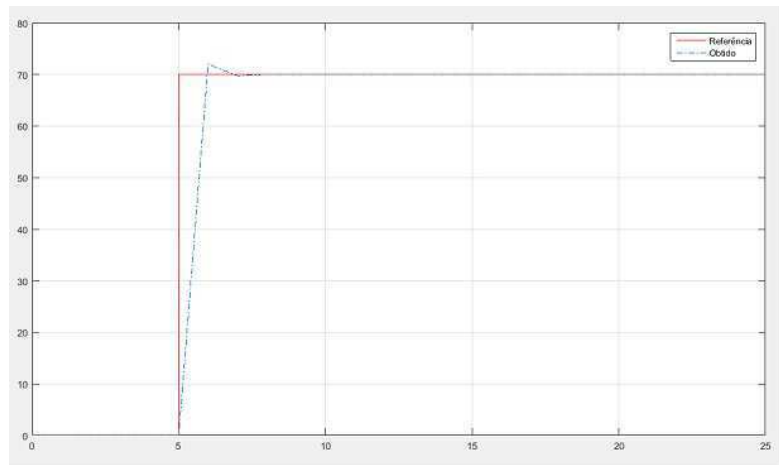


Figura 42 – Resposta do servo do cotovelo

6 Considerações Finais

Com base nos testes realizados é possível sugerir algumas melhorias. Tais melhorias são elucidadas nos pontos a seguir:

- Concluir a etapa de controle P, PI e PID;
- Instalar um sensor *Leap Motion*;
- Instalar uma câmera o outro objeto de forma que seja possível identificar a posição de objetos;
- Implementar a interface em C++ de forma a maximizar a compatibilidade com a biblioteca SDK;
- Usar *MoveIt* no caso da implementação em ROS

Por fim, conclui-se que há um vasto conteúdo de estudo no que tange a área de manipuladores, exigindo uma matemática consideravelmente apurada. Além disso a partir das três implementações de trajetórias foi possível vislumbrar que o melhor conjunto foi o implementado com ROS, uma vez que exigiu uma pequeníssima taxa de processamento da máquina quando comparada aos outros dois conjuntos.

Referências

1. SICILIANO, Bruno et al. **Robotics: Modelling, Planning and Control**. 1. ed. Glasgow: Springer, 2009. 644 p.
2. DENAVIT, J.; HARTENBERG, R.S.; EVANSTON, III. **A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices**. *Trans ASME*, New York, n.23, p. 215-221, mar. 1955.
3. KUIPERS, Jack B. **Quaternions and Rotation Sequences**. Coral Press, Sofia, p. 127-143, set. 2000.
4. EBERLY, David. **Euler Angle Formulas**. LLC, New York, p. 1-18, mar. 2014.
5. BRUYNINCKX, Herman. **Robot Kinematics and Dynamics**. 2. ed. Leuven: Editora da Universidade Católica de Leuven, 2010. 184 p.
6. SPONG, Mark W.; HUTCHINSON, Seth; Vidyasagar, M. **Robot Modeling and Control**. 1. ed. New York: 2015. 419 p.
7. CRAIG, John J. **Robótica**. 3. ed. Cidade do México: 2006. 410 p.
8. ROS. **Tutoriais ROS**. 2015. <<http://www.ros.org/>>. Acesso 12 de Outubro de 2016.

9. V-REP. **Tutoriais V-REP**. 2014.

<<http://www.coppeliarobotics.com/>>. Acesso 14 de Setembro de 2016.