

# Robótica de Manipulação, 2º Relatório: Dinâmica e Controlo

Instituto Superior Técnico  
MEMec

Maio 2021



## **Grupo 18**

Catarina Pires, N° 90230

Ricardo Henriques, N° 90349

**Professores:** João Reis, André Carvalho

**Versão de *Matlab* usada:** *Matlab R2020 a*

# Índice

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Ficheiros de <i>Matlab</i></b>	<b>2</b>
<b>3</b>	<b>Propriedades Físicas do UR5</b>	<b>2</b>
<b>4</b>	<b>Dinâmica de Corpo Rígido</b>	<b>6</b>
4.1	Formulação de Newton-Euler . . . . .	7
4.2	Implementação em <i>Simulink</i> . . . . .	8
4.2.1	Verificação pelo Balanço de Energias . . . . .	9
<b>5</b>	<b>Controlo Descentralizado</b>	<b>10</b>
5.1	Pior Caso de Inércia . . . . .	10
5.2	Introdução Teórica do Controlador . . . . .	10
5.3	Projeto do Controlador . . . . .	11
5.4	Implementação do Controlador . . . . .	12
<b>6</b>	<b>Controlo Centralizado</b>	<b>13</b>
6.1	Projeto do Controlador Centralizado para Dinâmica Inversa . . . . .	13
6.2	Implementação do Controlador . . . . .	14
<b>7</b>	<b>Desempenho dos Controladores</b>	<b>15</b>
7.1	Trajetória para o <i>end-effector</i> . . . . .	15
7.2	Controlador Descentralizado . . . . .	18
7.3	Controlador Centralizado . . . . .	19
7.4	Comparação entre os Dois Controladores . . . . .	21
<b>8</b>	<b>Conclusões</b>	<b>21</b>

# 1 Introdução

No âmbito da unidade curricular Robótica de Manipulação, este projeto consiste na análise do robot UR5 da Universal Robots. O foco deste relatório será na análise de dinâmica e controlo do manipulador. Primeiramente foi feita uma estimativa das propriedades físicas do robot em estudo, sendo que de seguida recorreu-se à formulação de Newton-Euler para obter a dinâmica gravitacional, a matriz de massas e o vetor de forças fictícias para o manipulador. Para controlar o braço robótico serão implementados um controlador descentralizado, e um controlador de dinâmica inversa centralizado. O desempenho destes controladores será testado quando é aplicada uma trajectória ao manipulador.

## 2 Ficheiros de *Matlab*

Para além dos ficheiros da *toolbox* descritos no relatório anterior, foram criados dois ficheiros *.m* essenciais para o funcionamento dos modelos em simulink.

- ***Dynamics\_G18.m***: Este script deve ser corrido antes de qualquer um dos ficheiros de Simulink, uma vez que define a dinâmica gravitacional, a matriz de massa e as velocidades fictícias do manipulador UR5, e os ganhos dos controladores centralizado e descentralizado.
- ***Newton\_Euler\_G18***: A partir desta função obtém-se a formulação Newton-Euler do manipulador para o modelo de dinâmica de corpo rígido.

## 3 Propriedades Físicas do UR5

Antes de se proceder à análise dinâmica do UR5, é necessário obter uma estimativa dos valores da massa, centro de massa e tensor de inércia para cada *link* do manipulador. Para tal procedeu-se à simplificação de cada parte do robot em sólidos geométricos mais simples, nomeadamente cilindros maciços para ter em conta os motores e mecanismos no interior do braço robótico. Após alguma pesquisa sobre o robot UR5, determinou-se que a massa total do robot é igual a 18.4 kg, e que os materiais do exterior do robot são alumínio, de densidade  $\rho = 2600\text{kg/m}^3$ , e *PP Plastic*, de densidade  $\rho = 913\text{kg/m}^3$ , e que o diâmetro da base é igual a 149 mm. Todas as outras medidas foram estimadas utilizando o software *MeshLab*.

Tendo em conta os valores encontrados, procederam-se às seguintes simplificações para cada *link*:

- ***Link 0***

Este *link* não é relevante para a dinâmica do robot, uma vez que se encontra sempre fixo à superfície onde o robot se encontra, e é admitido de que esta superfície também é fixa. Por esta razão, o centro de massa e o tensor de inércia do *link 0* não serão apresentados. Contudo, foi estimada a massa do mesmo para que, no final, seja possível comparar a massa total obtida para o manipulador com o valor da massa real.

Para o cálculo da massa do *link 0*, este foi partido em dois sólidos simples: um cilindro de diâmetro,  $d = 10\text{mm}$ , e comprimento,  $L = 35\text{mm}$ , que representa a saliência que se pode ver na figura 1; e outro cilindro que representa a base, com diâmetro,  $d = 149\text{mm}$ . Esta base pode ser decomposta em vários elementos, contudo foi admitida a sua altura médio,  $L = 20\text{mm}$ . O *link 0* simplificado é apresentado na figura 1.

A massa para cada uma das partes foi calculada utilizando a fórmula  $m = \frac{\pi d^2 L \rho}{4}$ , sendo que  $\rho$  é a massa volúmica do material do *link 0*. Atendendo que o material é alumínio, a massa é dada por,

$$m_0 = 0.00715 + 0.915 = 0.919kg$$

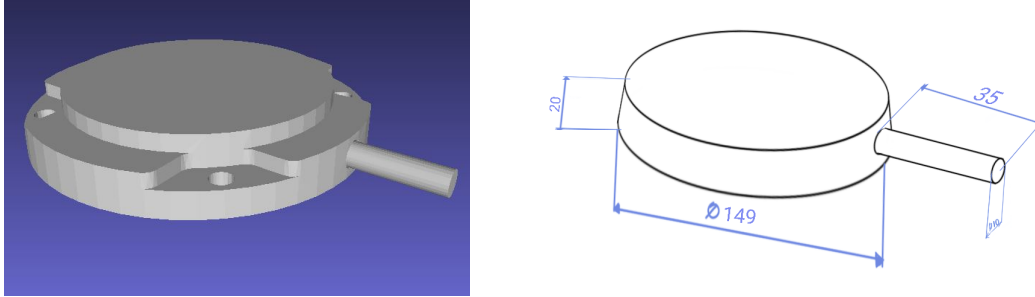


Figura 1: *Link 0* e respectiva simplificação

#### • Link 1

O *link 1*, figura 2, foi aproximado a um cilindro de PP Plastic,  $\rho = 913kg/m^3$ , com diâmetro  $d = 127mm$ , e altura  $L = 120mm$ , figura 2. A sua massa e tensor de inércias são dados por,

$$m_1 = 1.388kg \quad , \quad \bar{I}_1 = \begin{bmatrix} 0.00306 & 0 & 0 \\ 0 & 0.0028 & 0 \\ 0 & 0 & 0.00306 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 1* são:  $r_{1,c_1}^1 = [0; -0.005; 0]$

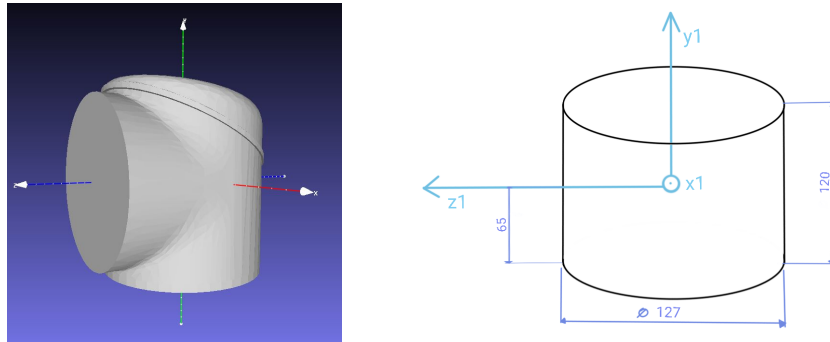


Figura 2: *Link 1* e respectiva simplificação

#### • Link 2

O *link 2*, figura 3, foi aproximado a três cilindros dois cilindros semelhantes ao que se utilizou para o *link 1*, e um cilindro de alumínio de diâmetro  $d = 100mm$ , e altura  $L = 298mm$ . Assim, este link é simétrico, logo o seu centro de massa coincide com o centro de massa do cilindro de alumínio, sendo que para os outros dois cilindro foi necessário aplicar o teorema dos eixos paralelos para calcular as suas inércias no centro de massa do *link*.

A massa e tensor de inércia são dados por,

$$m_2 = 1.388 \times 2 + 6.0853 = 8.861kg \quad , \quad \bar{I}_2 = \begin{bmatrix} 0.0137 & 0 & 0 \\ 0 & 0.1803 & 0 \\ 0 & 0 & 0.1798 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 2* são:  $r_{2,c_2}^2 = [-0.2125; \quad 0; \quad 0.109]$

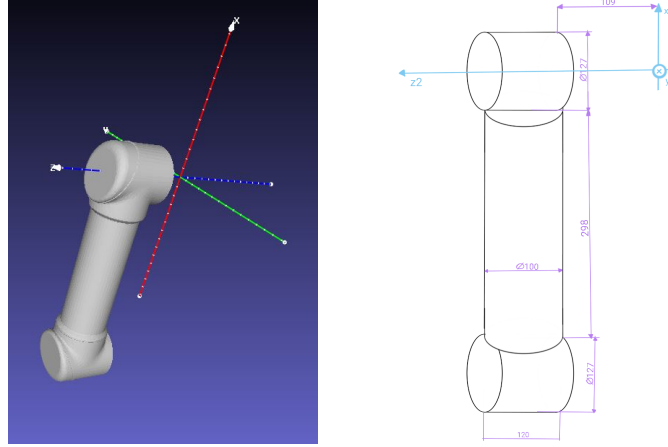


Figura 3: *Link 2* e respectiva simplificação

### • Link 3

O *link 3*, figura 4, foi aproximado a três cilindros. Dois em PP Plastic, sendo um semelhante ao do *link 1*, outro de diâmetro  $d = 74mm$ , e altura  $L = 100mm$ , que depois será semelhante nos *links 4* e *5*, e ainda um outro cilindro de alumínio com diâmetro  $d = 74mm$ , e altura  $L = 343mm$ , figura 4. Neste caso, como o *link* não é simétrico em  $z$ , foi necessário utilizar o teorema dos eixos paralelos para os três cilindros para calcular o tensor de inércia no centro de massa do corpo.

A massa e tensor de inércia são dados por,

$$m_3 = 1.388 + 3.835 + 0.393 = 5.6160kg \quad , \quad \bar{I}_3 = \begin{bmatrix} 0.0062 & 0 & 0 \\ 0 & 0.1373 & 0 \\ 0 & 0 & 0.1369 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 3* são:  $r_{3,c_3}^3 = [-0.2260; \quad 0; \quad 0.0135]$

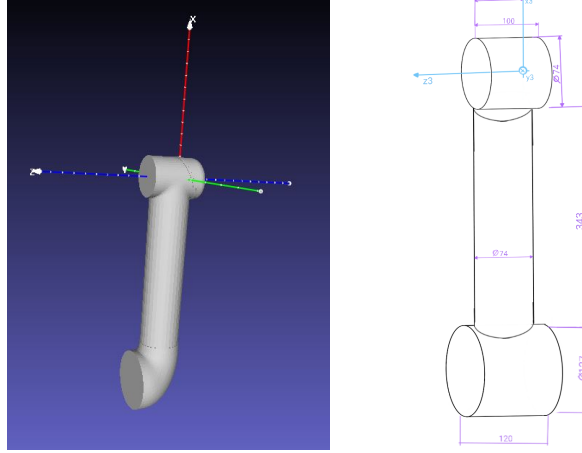


Figura 4: *Link 3* e respectiva simplificação

- **Link 4**

O *link 4*, figura 5, foi aproximado a um cilindros em PP Plastic de diâmetro  $d = 74mm$ , e altura  $L = 100mm$ , figura 5.

A massa e tensor de inércia são dados por,

$$m_4 = 0.393kg \quad , \quad \bar{I}_4 = \begin{bmatrix} 0.0004620 & 0 & 0 \\ 0 & 0.0004620 & 0 \\ 0 & 0 & 0.2690 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 4* são:  $r_{4,c_4}^4 = [0; \quad 0; \quad 0]$

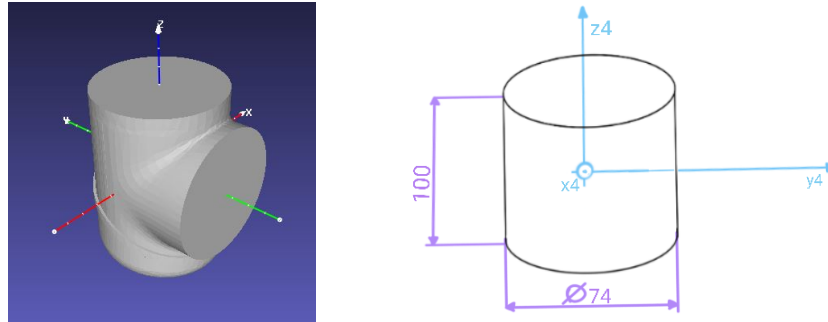


Figura 5: *Link 4* e respectiva simplificação

- **Link 5**

A aproximação para o *link 5* é semelhante à do *link 4*, figuras 6 e 6.

A massa e tensor de inércia são dados por,

$$m_5 = 0.393kg \quad , \quad \bar{I}_5 = \begin{bmatrix} 0.0004620 & 0 & 0 \\ 0 & 0.0004620 & 0 \\ 0 & 0 & 0.2690 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 5* são:  $r_{5,c_5}^5 = [0; \quad 0; \quad 0]$

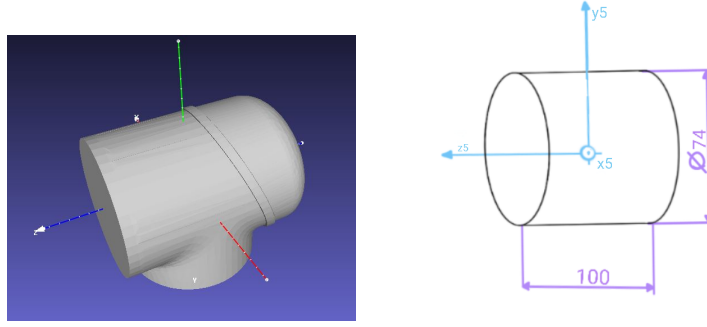


Figura 6: *Link 5* e respectiva simplificação

- ***Link 6***

O *link 6*, figura 7, foi aproximado a um cilindros de alumínio com diâmetro  $d = 75mm$ , e altura  $L = 34mm$ , figura 7.

A massa e tensor de inércia são dados por,

$$m_6 = 0.391kg \quad , \quad \bar{I}_5 = \begin{bmatrix} 0.0015 & 0 & 0 \\ 0 & 0.0015 & 0 \\ 0 & 0 & 0.0003 \end{bmatrix} kgm^2$$

E as coordenadas obtidas para o centro de massa na *frame 6* são:  $r_{6,c_6}^6 = [0; \quad 0; \quad 0]$

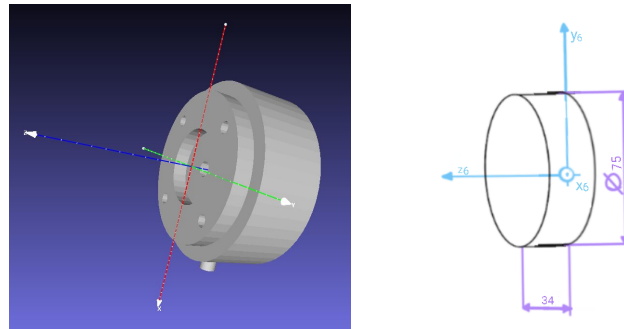


Figura 7: *Link 6* e respectiva simplificação

Das massas obtidas para cada *link* tem-se que a massa total estimada do UR5 é 17.96kg. Tendo em conta que a massa real encontrada é 18.4kg, conclui-se então que os valores obtidos são boas aproximações.

## 4 Dinâmica de Corpo Rígido

A dinâmica do manipulador pode ser expressa pela seguinte equação:

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau(q, \dot{q}, \ddot{q}) \quad (1)$$

$$B(q) = \frac{d\tau}{d\ddot{q}} \quad (2)$$

$$g(q) = \tau(q, 0, 0) \quad (3)$$

$$C(q, \dot{q})\dot{q} = \phi = \tau(q, \dot{q}, 0) - g(q) \quad (4)$$

De forma a calcular os termos da equação da dinâmica de corpo rígido,  $g(q)$ ,  $B(q)$ , e  $\phi(q)$ , para o robot em estudo, utilizou-se a formulação de Newton-Euler. Esta é baseada no balanço de todas as forças a actuar na junta genérica do manipulador. Este balanço leva a uma série de equações cuja estrutura permite o cálculo de uma solução de forma recursiva através de dois passos:

1. **Forward Recursion:** Usada para calcular as velocidades e acelerações das juntas.
2. **Backward Recursion:** Usada para calcular forças e momentos nas juntas.

#### 4.1 Formulação de Newton-Euler

Para esta formulação foram usadas as seguintes variáveis:

Tabela 1: Variáveis da Formulação de Newton-Euler

Variáveis	
$q_i$	Orientação da junta i
$\dot{q}_i$	Velocidade da junta i
$\ddot{q}_i$	Aceleração da junta i
$z_0$	Posição do eixo z no referencial 0
$R_i^{i-1}$	Matriz de rotação da junta i em relação ao referencial i-1
$\omega_i^{i-1}$	Velocidade angular da junta i
$\dot{\omega}_i^{i-1}$	Aceleração angular da junta i
$r_{i-1,i}^i$	Vetor de distância do referencial i-1 até referencial i, no referencial i
$r_{i,C_i}^i$	Vetor de distância do referencial i até ao centro de massa i, no referencial i
$r_{i-1,C_i}^i$	Vetor de distância do referencial i-1 até ao centro de massa i, no referencial i
$\ddot{p}_i^i = \ddot{r}_{0,i}^i$	Aceleração linear da junta i em relação ao referencial 0
$\ddot{p}_{C_i}^i = \ddot{r}_{0,C_i}^i$	Aceleração linear do centro de massa i em relação ao referencial 0
$f_i^{i-1}$	Força da junta i
$\mu_i^{i-1}$	Momento da junta i
$I_i^{i-1}$	Tensor de inércia do link i

##### 1. Forward Recursion:

###### • Velocidade Angular

$$\omega_i^i = R_i^{i-1T} (\omega_{i-1}^{i-1} + \dot{\theta}_i z_0) \quad (5)$$

Para as juntas 1 a 6 de revolução.

###### • Aceleração Angular

$$\dot{\omega}_i^i = R_i^{i-1T} (\dot{\omega}_{i-1}^{i-1} + \ddot{\theta}_i z_0) + \dot{\theta}_i S(\omega_{i-1}^{i-1}) z_0 \quad (6)$$

Para as juntas 1 a 6 de revolução.

###### • Aceleração Linear

$$\ddot{p}_i^i = R_i^{i-1T} \ddot{p}_{i-1}^{i-1} + S(\dot{\omega}_i^i) r_{i-1,i}^i + S(\omega_i^i) S(\omega_i^i) r_{i-1,i}^i \quad (7)$$

Para as juntas 1 a 6 de revolução.



- **Aceleração Linear do Centro de Massa**

$$\ddot{p}_{C_i}^i = \ddot{p}_i^i + S(\dot{\omega}_i^i) r_{i,C_i}^i + S(\omega_i^i) S(\omega_i^i) r_{i,C_i}^i \quad (8)$$

Para as juntas 1 a 6 de revolução.

## 2. Backward Recursion:

- **Força**

$$f_i^i = m_i \ddot{p}_{C_i}^i + R_{i+1}^i f_{i+1}^{i+1} \quad (9)$$

- **Momento**

$$\mu_i^i = \bar{I}_i^i \dot{\omega}_i^i + S(\dot{\omega}_i^i) \bar{I}_i^i \omega_i^i + R_{i+1}^i \mu_{i+1}^{i+1} + S(r_{i-1,C_i}^i) f_i^i + S(r_{i,C_i}^i) R_{i+1}^i f_{i+1}^{i+1} \quad (10)$$

## 3. Projectão nos Eixos da Junta:

$$\tau_i = f_i^{iT} R_i^{i-1T} z_0 \quad (11)$$

Para as juntas 1 a 6 de revolução.

É de salientar que, uma vez que o manipulador em estudo apresenta apenas juntas de revolução, apenas são apresentadas neste relatório as equações referentes às mesmas. Contudo, o ficheiro *Newton\_Euler\_G18.m* apresenta a formulação de Newton-Euler para qualquer manipulador.

## 4.2 Implementação em Simulink

A partir do algoritmo da formulação de Newton-Euler, descrito na secção anterior, criou-se a função de *Matlab*, *Newton\_Euler\_G18.m*. Esta recebe como argumentos de entrada a matriz de Denavit-Hartenberg (DH) do manipulador, as propriedades de massa de cada *link*, os vetores de velocidade e aceleração nas juntas, e os vetores de velocidade e aceleração angular e aceleração linear no *link* 0 ( $\omega_0^0$ ,  $\dot{\omega}_0^0$  e  $\ddot{p}_0^0$ ), e como argumento de saída  $\tau$ .

Esta função foi utilizada para obter a dinâmica da gravidade,  $g(q)$ , a dinâmica de aceleração, cuja derivada em ordem a  $\ddot{q}$  é igual à matriz de massa,  $B(q)$ , e a dinâmica de velocidade,  $\phi(q)$ . Para se obter  $g(q)$  correu-se a função *Newton\_Euler\_G18.m* usando apenas a entrada da aceleração linear no *link* 0, a matriz DH e as propriedades de massa do robot. O vetor  $\ddot{p}_0^0$  depende de da aceleração da gravidade, e, para o UR5, é igual a,

$$\ddot{p}_0^0 = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \quad (12)$$

Para a obtenção da dinâmica de aceleração foram utilizadas as entradas  $\ddot{q}$ ,  $\dot{\omega}_0^0$ , e  $\ddot{p}_0^0$ , a matriz DH, e as propriedades de massa do manipulador. Finalmente, a dinâmica de velocidade foi obtida utilizando as entradas  $\dot{q}$  e  $\omega_0^0$ , juntamente com a matriz DH e as propriedades de massa do robot.

Uma vez calculados  $g(q)$ ,  $B(q)$  e  $\phi(q)$  recorreu-se à função do *Matlab*, *matlabFunctionBlock*, para criar um bloco de *Simulink* que recebe as velocidades em cada junta e a sua configuração, e devolve os vetores  $g(q)$  e  $\phi(q)$  e a matriz  $B(q)$  (bloco da dinâmica de corpo rígido).

Em seguida procedeu-se então à implementação da equação da dinâmica no *Simulink*. Primeiramente, equação 1 foi manipulada de forma a ter-se  $\ddot{q}$  em função de  $B$ ,  $\phi$ , e  $g$ ,

$$\ddot{q} = B^{-1}(\tau - \phi - g) \quad (13)$$

De seguida,  $\ddot{q}$  é integrado duas vezes para se obterem as velocidades e configurações em cada junta, as quais são a entrada do bloco de dinâmica do corpo rígido. A implementação em *Simulink* encontra-se representada na figura 8.

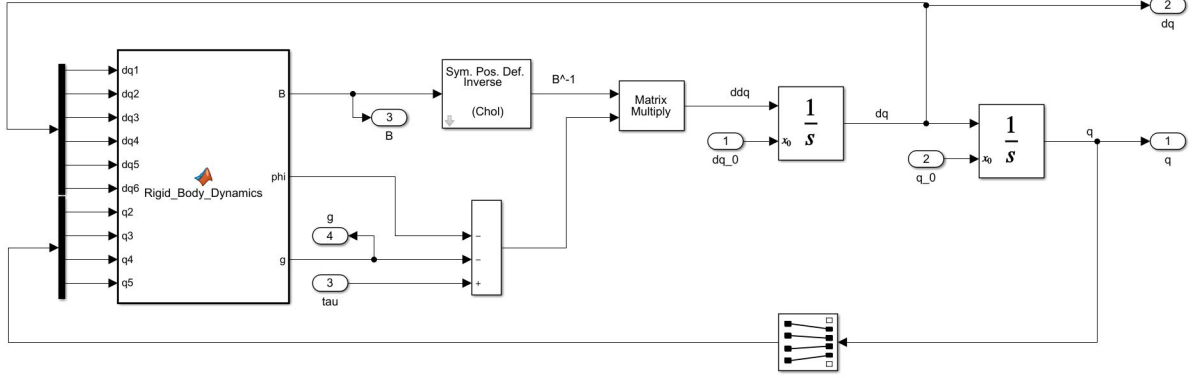


Figura 8: Implementação em *Simulink* da dinâmica direta do manipulador

#### 4.2.1 Verificação pelo Balanço de Energias

Para validar os resultados obtidos para o modelo de dinâmica de corpo rígido fez-se uma verificação pelo balanço de energia. Pelo Princípio de Conservação de Energia tem-se que a energia mecânica do sistema tem de ser sempre constante, sendo que esta é igual à soma da energia potencial e da energia cinética. A energia cinética do sistema é dada por,

$$T = \frac{1}{2} \times \dot{q}^T B \dot{q} \quad (14)$$

A energia potencial é obtida a partir da integração da potência das forças externas, cuja expressão é dada por,

$$\dot{U} = (g - \tau)^T \times \dot{q} \quad (15)$$

Na implementação em *Simulink* foram criados dois blocos do tipo *MatlabFunction*, que contêm as equações 14 e 15, figura 9, que se encontram num subsistema que recebe o  $\tau$  que também entra no bloco de dinâmica direta, e ainda recebe deste os vetores  $g$  e  $\dot{q}$ , e a matriz  $B$ , figura 10. O valor de  $\tau$  utilizado na verificação é igual às velocidades em cada junta multiplicadas por um ganho de amortecimento.

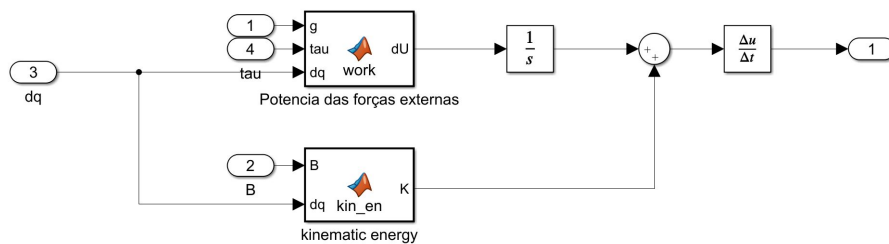


Figura 9: Cálculo da Energia Mecânica

Uma vez que a energia mecânica tem de permanecer constante, para a validar o modelo procedeu-se à sua derivação e verificou-se que esta, de facto, dava zero, figura 10.

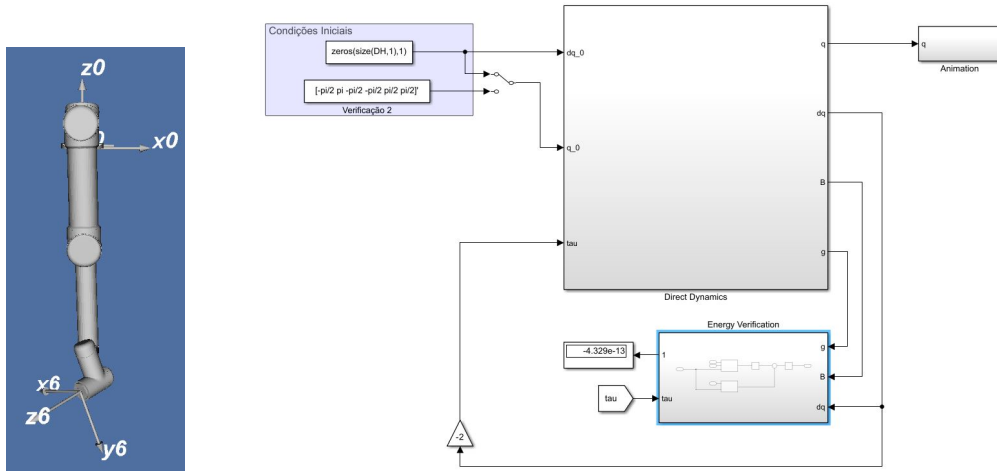


Figura 10: Validação do modelo

## 5 Controlo Descentralizado

### 5.1 Pior Caso de Inércia

No pior caso de inércia, ou caso de inércia máxima, os centros massa dos *links* encontram-se o mais afastados possível da *frame 0*, consequentemente aumentando a sua inércia e dificultando o seu movimento. Desta forma, a sua análise é essencial para a regulação dos ganhos dos controladores. Os valores máximos de inércia são obtidos da matriz de massa  $B(q)$ , cujos valores da diagonal representam a inércia média em cada junta. É de salientar que, uma vez que o manipulador pode mover-se nas três dimensões do espaço, haverá mais do que uma solução possível.

Relativamente ao manipulador UR5, a solução para o pior caso de inércia obtida encontra-se representada na 2. Note-se que as configurações das juntas 1 e 6 não tomam nenhum papel na variação da inércia do robot, e como tal, a matriz  $B(q)$  é independente das configurações  $q_1$  e  $q_6$ .

Tabela 2: Configurações para o pior caso de inércia

	1	2	3	4	5	6
$q_i$	-	0	0	-111	-90	-
$B_{ii}$	4.818	4.696	0.9608	0.0121	0.004561	0.000275

### 5.2 Introdução Teórica do Controlador

A abordagem de controlo descentralizado encara o manipulador como um conjunto de  $n$  sistemas independentes ( $n$  juntas) e controla o eixo de cada junta como um sistema SISO. Os efeitos de acoplamento entre as juntas devido às diferentes configurações do manipulador durante um movimento são assumidas como perturbações. Uma vez que o erro de seguimento tem de ser corrigido é possível usar uma compensação gravítica em vez de uma acção integradora dado que no caso de a velocidade não ser muito alta o erro resulta da acção da gravidade. Desse modo, ao fazer um controlo de força na junta é possível projectar um controlador proporcional derivativo (PD) que

permite executar a previsão do erro da resposta do sistema com compensação gravítica. As figuras seguintes representam o modelo do controlo descentralizado e do controlador PD:

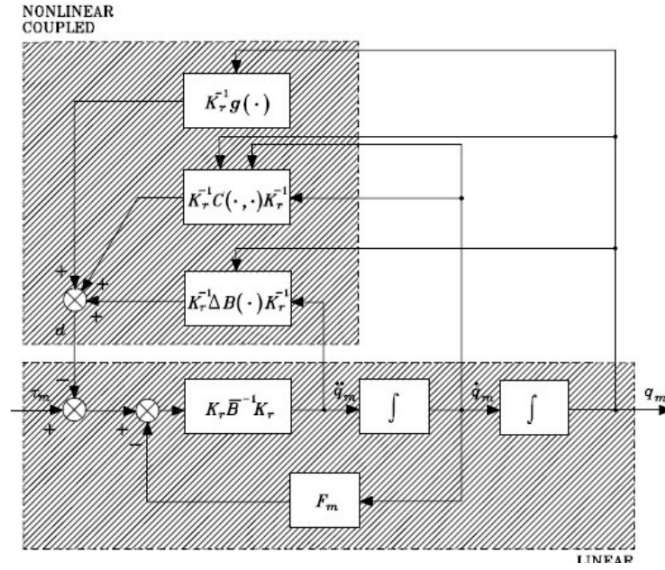


Figura 11: Modelo de um Controlador Descentralizado

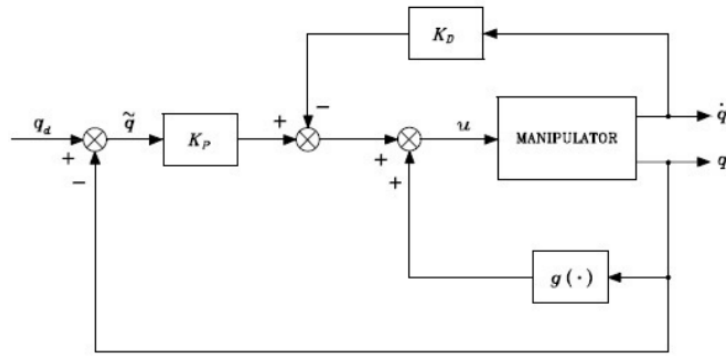


Figura 12: Diagrama de Blocos de um Controlador Proporcional Derivativo

### 5.3 Projeto do Controlador

A equação de movimento do manipulador é dada por:

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} = \tau \quad (16)$$

Com a compensação gravítica consegue-se eliminar a dinâmica devido à gravidade. Desta forma, a equação do projeto do controlador PD é a seguinte:

$$\hat{g}(q) + -K_D\dot{q} + K_P\tilde{q} = \tau \quad (17)$$

O termo  $\hat{g}(q)$  é uma estimativa e assumindo que  $\hat{g}(q) = g(q)$ , é possível combinar as equações 16 e 17 e chegando à seguinte equação:

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} = -K_D\dot{q} + K_P\tilde{q} \quad (18)$$

Uma vez que  $\tilde{q} = q_d - q$ , é possível obter a seguinte expressão através da transformada de laplace:

$$Bs^2Q(s) + CQ(s)s = K_PQ_d(s) - K_D(s)s \iff Q(s)(Bs^2 + (C + K_D)s + K_P) = K_PQ_d(s) \quad (19)$$

$$G(s) = \frac{Q(s)}{Q_d(s)} = \frac{K_P}{Bs^2 + (C + K_D)s + K_P} \quad (20)$$

Dividindo a equação anterior pelo termo B e desprezando o termo não linear C, uma vez que para velocidades baixas  $C = 0$ , é possível observar que a função de transferência é do tipo:

$$G(s) = \frac{K_P}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (21)$$

Desta forma pode-se calcular os ganhos:

$$\frac{K_P}{B} = \omega_n^2 \iff K_P = \omega_n^2 B \quad (22)$$

$$\frac{K_D}{B} = 2\xi\omega_n \iff K_D = 2\xi\omega_n B \quad (23)$$

## 5.4 Implementação do Controlador

Considerando o caso do amortecimento crítico ( $\xi = 1$ ) foram testados 3 casos diferentes, um com ganhos mais baixos, um com ganhos intermédios e outra com ganhos mais elevados, através da variação do valor de  $\omega_n$ :

Tabela 3: Valores dos Ganhos com  $\omega_n = 20rad/s$

Valores dos Ganhos com $\omega_n = 20rad/s$		
n	$K_P$	$K_D$
1	1927.2	192.72
2	1878.4	187.84
3	384.32	38.432
4	4.84	0.48
5	1.8244	0.1824
6	0.11	0.011

Tabela 4: Valores dos Ganhos com  $\omega_n = 60rad/s$

Valores dos Ganhos com $\omega_n = 60rad/s$		
n	$K_P$	$K_D$
1	17345	578.16
2	16906	563.52
3	3458.9	115.296
4	43.56	1.452
5	16.4196	0.5473
6	0.99	0.033

Tabela 5: Valores dos Ganhos com  $\omega_n = 100rad/s$

Valores dos Ganhos com $\omega_n = 100rad/s$		
n	$K_P$	$K_D$
1	48180	963.6
2	46960	939.2
3	9608	192.16
4	121	2.42
5	45.61	0.9122
6	2.75	0.055

Estes ganhos foram implementados no seguinte modelo de simulink e foram usados para testar a trajetória que será abordada posteriormente:

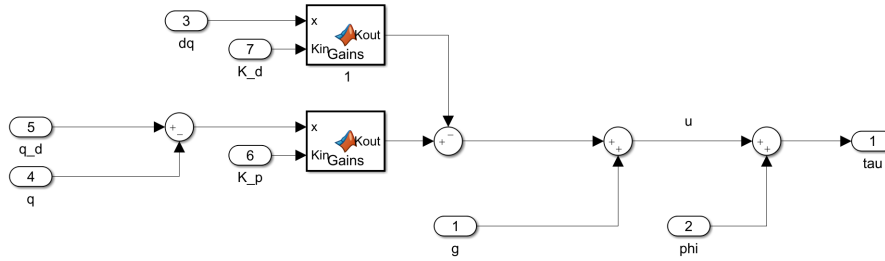


Figura 13: Implementação em Simulink do Controlador PD

## 6 Controlo Centralizado

Quando são necessárias grandes velocidades operacionais ou actuação de accionamento directo, os termos de acoplamento não linear têm uma grande influência no desempenho do sistema. Desta forma, ao contrário do que se faz na abordagem de controlo descentralizado, considerar os efeitos dos componentes dos coeficiente de fricção viscosa como perturbações pode gerar grandes erros no seguimento de referências. Neste método de controlo projectam-se algoritmos que tiram partido de um conhecimento detalhado da dinâmica do manipulador, de forma a compensar os termos de acoplamento não linear do modelo.

### 6.1 Projeto do Controlador Centralizado para Dinâmica Inversa

De forma a modelar o controlador centralizado é preciso descrever o modelo dinâmico considerando os termos não lineares ( $n(q, \dot{q})$ ), sendo a sua formulação a seguinte:

$$B(q)\ddot{q} + n(q, \dot{q}) = u \quad (24)$$

Nesta formulação,  $u$  representa a acção de controlo. De forma a executar uma linearização exata é necessário haver um *feedback* não linear:

$$u = B(q) + n(q, \dot{q}) \quad (25)$$

Nesta formulação  $y = \ddot{q}$ . Desta forma, o problema do controlo centralizado resume-se a encontrar

um controlo estável  $y$ , sendo este:

$$y = -K_P q - K_D \dot{q} + r \iff \ddot{q} + K_D \dot{q} + K_P q = r \quad (26)$$

Os ganhos são definidos da seguinte forma:

$$K_P = \begin{bmatrix} \omega_{n1}^2 & \dots & 0 \\ & \ddots & \\ 0 & \dots & \omega_{nn}^2 \end{bmatrix} \quad (27)$$

$$K_D = \begin{bmatrix} 2\xi_1 \omega_{n1} & \dots & 0 \\ & \ddots & \\ 0 & \dots & 2\xi_n \omega_{nn} \end{bmatrix} \quad (28)$$

Considerou-se que  $\xi = 1$  e que  $\omega_{n1} = \dots = \omega_{nn}$ . Para obter o melhor resultado possível variou-se os valores de  $\omega$ .

Tabela 6: Valores dos Ganhos

	Ganho Baixo	Ganho Médio	Ganho Elevado
$\omega_n$	20	60	100
$K_P$	400	3600	10000
$K_D$	40	120	200

## 6.2 Implementação do Controlador

A implementação foi baseada no seguinte diagrama de blocos:

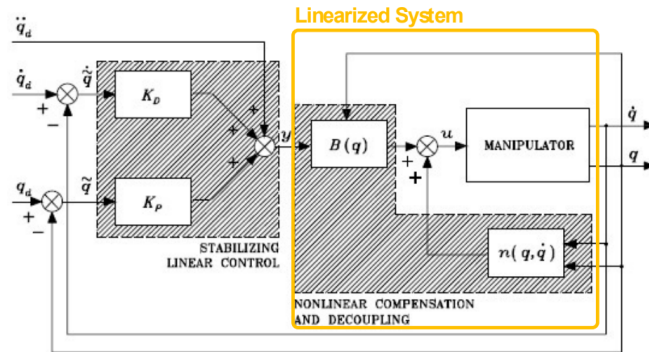


Figura 14: Diagrama de Blocos de um Controlador Centralizado

Os ganhos abordados no ponto 6.1 foram implementados no seguinte modelo de simulink e foram usados para testar a trajetória que será abordada posteriormente:

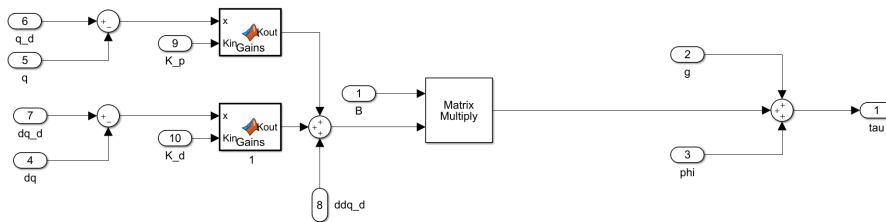


Figura 15: Implementação em Simulink do Controlador Centralizado

## 7 Desempenho dos Controladores

### 7.1 Trajetória para o *end-effector*

Com o objectivo de testar e comparar o desempenho dos controladores projetados, foi escolhida uma trajetória para o *end-effector* de forma a este realizar uma certa tarefa. Esta trajetória foi baseada na utilização do robot UR5 na fábrica da empresa *Nordic Sugar*<sup>[3]</sup>, na Suécia, em que o manipulador participa no processo de análise do açúcar, pegando em recipientes do produto e movendo-os repetidamente entre balanças e filtros.

Na trajetória projetada, o manipulador parte da posição e orientação iniciais,

$$p_a = \begin{bmatrix} -0.400 \\ 0 \\ 0.180 \end{bmatrix}, \quad R_a^0 = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

com as seguintes configurações nas juntas (em graus),

$$q = \begin{bmatrix} -20 \\ 120 \\ 132 \\ -245 \\ -95 \\ 177 \end{bmatrix} \quad (29)$$

Para implementação da trajetória, esta foi dividida em 9 caminhos diferentes (*paths*). Admitindo que  $s$  é o comprimento total da trajetória percorrida no tempo  $t \in \{t_0, t_f\}$  definida pela seguinte expressão,

$$s(t) = s_0 + 3(s_f - s_0)\left(\frac{t}{t_f}\right)^2 - 2(s_f - s_0)\left(\frac{t}{t_f}\right)^3 \quad (30)$$

com  $s_0$  como a posição inicial,  $s_f$  como a posição final, e  $t_f$  como o tempo final, os caminhos que a constituem são definidos da seguinte forma:

- **Subida do *End-effector***

Para  $0 \leq s \leq L$ , sendo  $L$  a distância percorrida verticalmente pelo *end-effector*, a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} 0 \\ -s \\ 0 \end{bmatrix}, \quad R_d^a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} 0 \\ -\dot{s} \\ 0 \end{bmatrix}, \quad \dot{R}_d^a(s) = 0, \quad \omega = 0$$

- **Rotação em torno de  $z_0$  em sentido horário**

Para  $L < s \leq L + \frac{\pi\rho}{2}$ , sendo  $\rho$  o arco de circunferência, a posição e a orientação são definidas por,



$$r_{a,d}^a(s) = \begin{bmatrix} \rho \sin(\frac{s-L}{\rho}) \\ -L \\ -\rho + \rho \cos(\frac{s-L}{\rho}) \end{bmatrix}, \quad R_d^a = \begin{bmatrix} \cos(\frac{s-L}{\rho}) & 0 & \sin(\frac{s-L}{\rho}) \\ 0 & 1 & 0 \\ -\sin(\frac{s-L}{\rho}) & 0 & \cos(\frac{s-L}{\rho}) \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \dot{s} \begin{bmatrix} \cos(\frac{s-L}{\rho}) \\ 0 \\ -\sin(\frac{s-L}{\rho}) \end{bmatrix}, \quad \dot{R}_d^a = \frac{\dot{s}}{\rho} \begin{bmatrix} -\sin(\frac{s-L}{\rho}) & 0 & \cos(\frac{s-L}{\rho}) \\ 0 & 1 & 0 \\ -\cos(\frac{s-L}{\rho}) & 0 & -\sin(\frac{s-L}{\rho}) \end{bmatrix}, \quad \omega = \begin{bmatrix} S(\omega)(3,2) \\ S(\omega)(1,3) \\ S(\omega)(2,1) \end{bmatrix}$$

Onde  $S(\omega)$  é a matriz anti-simétrica definida como  $S(\omega) = \dot{R}_d^a R_d^{aT}$

- **Descida do *End-effector***

Para  $L + \frac{\pi\rho}{2} < s \leq 2L + \frac{\pi\rho}{2}$  a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho \\ s - (2L + \frac{\pi\rho}{2}) \\ -\rho \end{bmatrix}, \quad R_d^a = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} 0 \\ \dot{s} \\ 0 \end{bmatrix}, \quad \dot{R}_d^a(s) = 0, \quad \omega = 0$$

- **Esticar o braço (colocar produto no lugar devido)**

Para  $2L + \frac{\pi\rho}{2} < s \leq 2L + \frac{\pi\rho}{2} + d$ , em que  $d$  é a distância que o braço "se estica", a qual foi admitida como sendo igual a 0.2 . A posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho + (s - (2L + \frac{\pi\rho}{2})) \\ 0 \\ -\rho \end{bmatrix}, \quad R_d^a = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} \dot{s} \\ 0 \\ 0 \end{bmatrix}, \quad \dot{R}_d^a(s) = 0, \quad \omega = 0$$

- **Tempo de espera (libertar o produto no lugar devido)**

Para  $2L + \frac{\pi\rho}{2} + 0.2 < s \leq 2L + \frac{\pi\rho}{2} + 0.2 + \Delta s$ , sendo  $t_{e_0}$  o tempo de início de espera e  $t_{e_f}$  o tempo de fim de espera, então  $\Delta s = s(t_{e_f}) - s(t_{e_0})$ . Este valor foi assumido como igual a 0.3. A posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho + 0.2 \\ 0 \\ -\rho \end{bmatrix}, \quad R_d^a = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = 0 \quad , \quad \dot{R}_d^a(s) = 0 \quad , \quad \omega = 0$$

- **Recolha do *End-effector***

Para  $2L + \frac{\pi\rho}{2} + 0.5 < s \leq 2L + \frac{\pi\rho}{2} + 0.7$  a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho - (s - (2L + \frac{\pi\rho}{2} + 0.7)) \\ 0 \\ -\rho \end{bmatrix} \quad , \quad R_d^a = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} -\dot{s} \\ 0 \\ 0 \end{bmatrix} \quad , \quad \dot{R}_d^a(s) = 0 \quad , \quad \omega = 0$$

- **Segunda subida do *End-effector***

Para  $2L + \frac{\pi\rho}{2} + 0.7 < s \leq 3L + \frac{\pi\rho}{2} + 0.7$  a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho \\ -(s - (2L + \frac{\pi\rho}{2} + 0.7)) \\ -\rho \end{bmatrix} \quad , \quad R_d^a = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} 0 \\ -\dot{s} \\ 0 \end{bmatrix} \quad , \quad \dot{R}_d^a(s) = 0 \quad , \quad \omega = 0$$

- **Rotação em torno de  $z_0$  em sentido anti-horário**

Para  $3L + \frac{\pi\rho}{2} + 0.7 < s \leq 3L + \pi\rho + 0.7$  a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} \rho \sin(\frac{s-3L-0.7}{\rho}) \\ -L \\ -\rho - \rho \cos(\frac{s-3L-0.7}{\rho}) \end{bmatrix} \quad , \quad R_d^a = \begin{bmatrix} -\cos(\frac{s-3L-0.7}{\rho}) & 0 & \sin(\frac{s-3L-0.7}{\rho}) \\ 0 & 1 & 0 \\ -\sin(\frac{s-3L-0.7}{\rho}) & 0 & -\cos(\frac{s-3L-0.7}{\rho}) \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \dot{s} \begin{bmatrix} \cos(\frac{s-3L-0.7}{\rho}) \\ 0 \\ \sin(\frac{s-3L-0.7}{\rho}) \end{bmatrix} \quad , \quad \dot{R}_d^a = \frac{\dot{s}}{\rho} \begin{bmatrix} \sin(\frac{s-3L-0.7}{\rho}) & 0 & \cos(\frac{s-3L-0.7}{\rho}) \\ 0 & 1 & 0 \\ -\cos(\frac{s-3L-0.7}{\rho}) & 0 & \sin(\frac{s-3L-0.7}{\rho}) \end{bmatrix}$$

$$\omega = \begin{bmatrix} S(\omega)(3,2) \\ S(\omega)(1,3) \\ S(\omega)(2,1) \end{bmatrix}$$

Onde  $S(\omega)$  é a matriz anti-simétrica definida como  $S(\omega) = \dot{R}_d^a R_d^{aT}$

- **Segunda descida do *End-effector***

Para  $3L + \pi\rho + 0.7 < s \leq 4L + \pi\rho + 0.7$  a posição e a orientação são definidas por,

$$r_{a,d}^a(s) = \begin{bmatrix} 0 \\ s - (4L + \pi\rho + 0.7) \\ 0 \end{bmatrix}, \quad R_d^a = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

E as velocidades linear e angular são definidas por,

$$\dot{r}_{a,d}^a(s) = \begin{bmatrix} 0 \\ \dot{s} \\ 0 \end{bmatrix}, \quad \dot{R}_d^a(s) = 0, \quad \omega = 0$$

## 7.2 Controlador Descentralizado

Uma vez definida a trajectória que o *end-effector* deverá seguir, os controladores projectados nas secções 5 e 6 foram postos à prova quando esta é executada a duas velocidades diferentes, uma em 30 segundos e outra a 60 segundos. Nesta secção será analisado o erro da orientação de cada junta quando é utilizado o controlador descentralizado. As tabelas 7 e 8 apresentam os resultados do erro relativo médio para cada junta nos ensaios de 60 e 30 segundos, respetivamente. Na figuras 16 e 17 é apresentado o erro absoluto para os melhores resultados, para cada tempo de duração do ensaio.

Tabela 7: Erro da orientação de cada junta para 60s

Junta	Ganho Baixo ( $\omega_n = 20rad/s$ )	Ganho Médio ( $\omega_n = 60rad/s$ )	Ganho Alto ( $\omega_n = 100rad/s$ )
1	0.0068	0.0021	0.0012
2	0.0035	0.0012	$7.415 \times 10^{-4}$
3	0.0035	0.0012	$7.467 \times 10^{-4}$
4	0.0031	0.0011	$6.486 \times 10^{-4}$
5	$8.6971 \times 10^{-4}$	$2.714 \times 10^{-4}$	$1.604 \times 10^{-4}$
6	$2.799 \times 10^{-5}$	$9.423 \times 10^{-6}$	$5.636 \times 10^{-6}$
Média	0.0030	$9.791 \times 10^{-4}$	$5.851 \times 10^{-4}$

Tabela 8: Erro da orientação de cada junta para 30s

Junta	Ganho Baixo ( $\omega_n = 20rad/s$ )	Ganho Médio ( $\omega_n = 60rad/s$ )	Ganho Alto ( $\omega_n = 100rad/s$ )
1	0.0135	0.0042	0.0025
2	0.0072	0.0025	0.0015
3	0.0071	0.0024	0.0014
4	0.0062	0.0021	0.0013
5	0.0020	$5.819 \times 10^{-4}$	$3.380 \times 10^{-4}$
6	$6.373 \times 10^{-5}$	$2.0512 \times 10^{-5}$	$1.215 \times 10^{-5}$
Média	0.0060	0.0020	0.0012

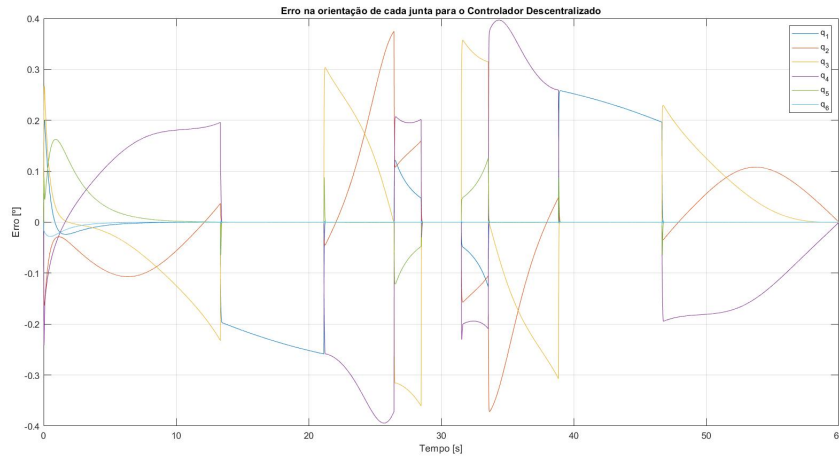


Figura 16: Erro absoluto para os melhores resultados para 60 segundos (Ganho alto)

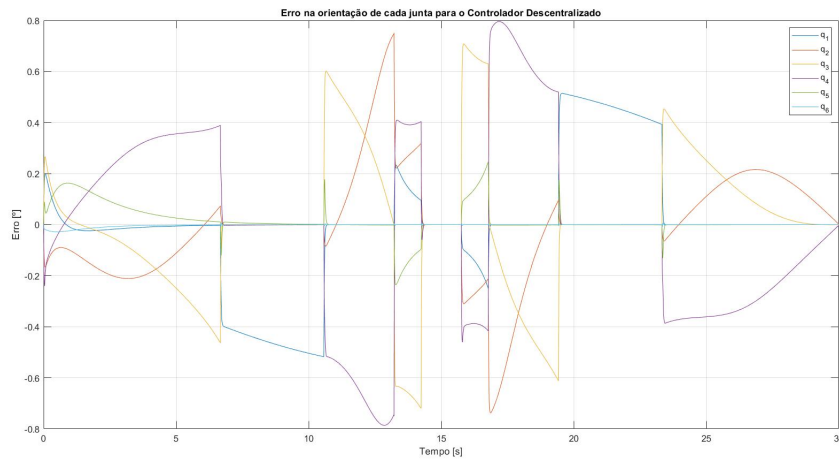


Figura 17: Erro absoluto para os melhores resultados para 30 segundos (Ganho alto)

Analisando os valores das tabelas 7 e 8, é possível concluir que o erro da orientação das juntas diminui com o aumento do ganho do controlador, e com o aumento do tempo de duração. Observando as figuras 16 e 17, que ilustram a variação do erro absoluto ao longo do tempo, para cada junta, observa-se que os valores mais elevados correspondem à transição de um caminho da trajetória para outro. Contudo, os erros observados são baixos o que leva a querer que o controlador foi bem implementado.

### 7.3 Controlador Centralizado

Nesta secção será analisado o erro da orientação e de cada junta quando é utilizado o controlador centralizado. As tabelas 9 e 10 apresentam os resultados do erro relativo médio para cada junta nos ensaios de 60 e 30 segundos, respectivamente. Na figuras 18 e 19, é apresentado o erro absoluto para os melhores resultados, para cada tempo de duração do ensaio.

Tabela 9: Erro da orientação de cada junta para 60s

Junta	Ganho Baixo ( $\omega_n = 20rad/s$ )	Ganho Médio ( $\omega_n = 60rad/s$ )	Ganho Alto ( $\omega_n = 100rad/s$ )
1	$7.100 \times 10^{-5}$	$2.552 \times 10^{-5}$	$5.412 \times 10^{-6}$
2	$1.725 \times 10^{-5}$	$5.934 \times 10^{-6}$	$2.220 \times 10^{-6}$
3	$3.690 \times 10^{-5}$	$1.705 \times 10^{-5}$	$3.600 \times 10^{-6}$
4	$1.715 \times 10^{-5}$	$7.052 \times 10^{-6}$	$1.970 \times 10^{-6}$
5	$7.047 \times 10^{-6}$	$2.026 \times 10^{-6}$	$1.170 \times 10^{-6}$
6	$3.179 \times 10^{-7}$	$7.076 \times 10^{-8}$	$2.352 \times 10^{-8}$
Média	$2.4942 \times 10^{-5}$	$9.6097 \times 10^{-6}$	$2.3975 \times 10^{-6}$

Tabela 10: Erro da orientação de cada junta para 30s

Junta	Ganho Baixo ( $\omega_n = 20rad/s$ )	Ganho Médio ( $\omega_n = 60rad/s$ )	Ganho Alto ( $\omega_n = 100rad/s$ )
1	$2.555 \times 10^{-4}$	$3.422 \times 10^{-5}$	$1.484 \times 10^{-5}$
2	$6.201 \times 10^{-5}$	$1.210 \times 10^{-5}$	$5.563 \times 10^{-6}$
3	$1.551 \times 10^{-4}$	$2.770 \times 10^{-5}$	$1.145 \times 10^{-5}$
4	$6.763 \times 10^{-5}$	$1.230 \times 10^{-5}$	$6.100 \times 10^{-6}$
5	$2.738 \times 10^{-5}$	$5.720 \times 10^{-6}$	$2.901 \times 10^{-6}$
6	$5.372 \times 10^{-7}$	$1.126 \times 10^{-7}$	$3.990 \times 10^{-8}$
Média	$9.4681 \times 10^{-5}$	$1.5357 \times 10^{-5}$	$6.8154 \times 10^{-6}$

Analisando os valores das tabelas 9 e 10, é possível concluir que o erro da orientação das juntas diminui com o aumento do ganho do controlador sendo que o melhor resultado obtido foi para  $\omega_n = 100rad/s$  em ambos os casos. Os valores do erro da orientação obtidos são relativamente baixos o que nos permite concluir que o controlador foi bem aplicado. Ao analisar as duas trajetórias pode-se concluir que foram obtidos erros de orientação maiores na trajetória mais rápida, e com os maiores ganhos. De seguida são apresentados gráficos dos erros absolutos para os melhores resultados da orientação.

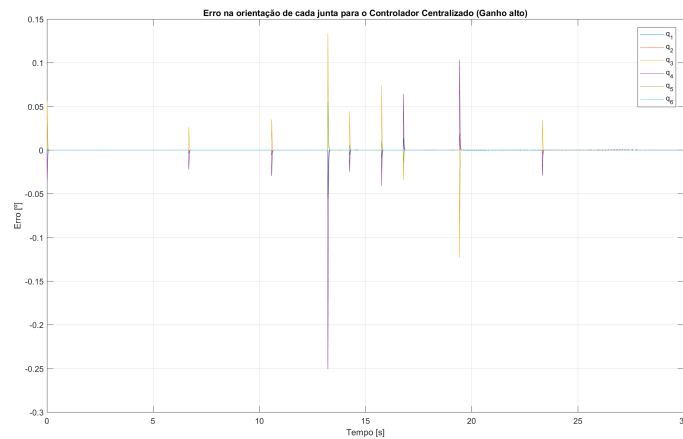


Figura 18: Erro na configuração das juntas para o melhor resultado para 30s

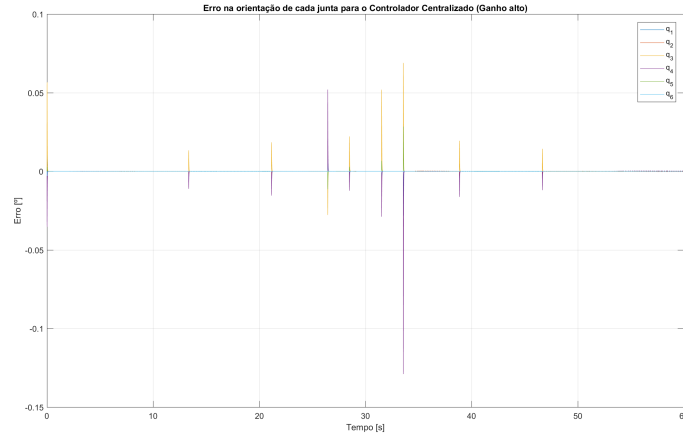


Figura 19: Erro na configuração das juntas para o melhor resultado para 60s

Observando as figuras 18, 19, que ilustram a variação do erro absoluto ao longo do tempo, para cada junta, conseguimos observar que os valores mais elevados correspondem à transição de um caminho da trajetória para outro.

## 7.4 Comparação entre os Dois Controladores

Tabela 11: Erro da orientação de cada junta para 60s

	$\omega_n = 20rad/s$	$\omega_n = 60rad/s$	$\omega_n = 100rad/s$
Controlador Descentralizado	0.0030	$9.791 \times 10^{-4}$	$5.851 \times 10^{-4}$
Controlador Centralizado	$2.4942 \times 10^{-5}$	$9.097 \times 10^{-6}$	$2.3975 \times 10^{-6}$

Tabela 12: Erro da orientação de cada junta para 30s

	$\omega_n = 20rad/s$	$\omega_n = 60rad/s$	$\omega_n = 100rad/s$
Controlador Descentralizado	0.0060	0.0020	0.0012
Controlador Centralizado	$9.4681 \times 10^{-5}$	$1.5357 \times 10^{-5}$	$6.8154 \times 10^{-6}$

Observando as tabelas anteriores é possível concluir que o controlador centralizado apresenta resultados muito mais favoráveis que o descentralizado, havendo uma diferença de duas ordens de grandeza.

## 8 Conclusões

Nesta fase final do projeto foi implementado com sucesso o modelo da dinâmica do corpo rígido usando o método de Newton-Euler, confirmado pelo balanço das energias e foi desenhada uma trajetória para um procedimento real do funcionamento do robot. Para esta trajetória conseguimos implementar um controlador descentralizado PD e um controlador centralizado e conseguimos perceber o funcionamento de cada um destes e a sua implementação. Em suma, a aplicação dos conceitos abordados foi fundamental para a sua compreensão bem como para perceber o funcionamento do robot.

## Referências

- [1] *Robotics - Modelling, Planning and Control*, B. Siciliano, L. Sciavicco, L. Villani and G. Oriolo 2009 Springer-Verlag
- [2] Apontamentos da disciplina Robótica de Manipulação: Jorge Martins 2009 IST 6
- [3] Nordic Sugar, *The UR5 makes all employees robot experts*. - Universal Robots,  
<https://www.universal-robots.com/case-stories/nordic-sugar/>