



# **Robótica Industrial**

## **Aula prática nº 7**

**Cinemática Inversa: Robôs planares e redundâncias.**

Vítor Santos

Universidade de Aveiro

30 Out 2023

# Sumário

- 1 Cinemática Inversa do robô RR planar
- 2 Funções auxiliares `InitRobot()`, `RobotEndPath()`
- 3 Caminhos nas várias redundâncias
- 4 Cinemática inversa do robô RRR planar

# Exercício 1 - Cinemática Inversa do robô RR planar

## Escrever a função de cinemática inversa do robô RR planar

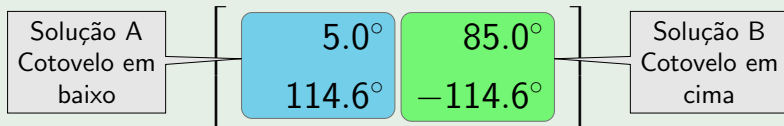
- $Q = \text{invkinRR}(x, y, L1, L2)$
- Contemplar as redundâncias, retornando a matriz:
  - $Q = [q1A \ q1B; \ q2A \ q2B]$ ,
  - sendo a solução A a do cotovelo em baixo.

$$\theta_2 = \pm \arccos \frac{x^2 + y^2 - L_1^2 - L_2^2}{2L_1L_2}$$

$$\theta_1 = \arctan \left[ \frac{y(L_1 + L_2 \cos \theta_2) - xL_2 \sin \theta_2}{x(L_1 + L_2 \cos \theta_2) + yL_2 \sin \theta_2} \right]$$

## Exemplo

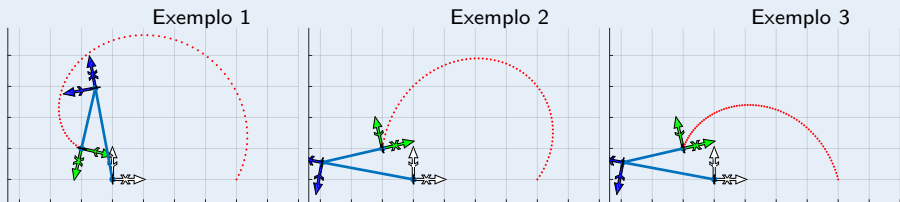
- Para um robô com  $L1=3$  e  $L2=2$ , a expressão  $\text{invkinRR}(2,2,L1,L2)$  deverá dar aproximadamente (convertido em graus):



## Exercício 2 - Animação de RR planar

### Ilustrar o movimento animado do robô RR planar

- Com  $L1=3$  e  $L2=2$
  - entre os pontos  $p1=[4;0]$  e  $p2=[-1; 1]$
  - usando a redundância do cotovelo em baixo ( $1^a$  redundância).
- 
- Porém, é possível usar as diversas combinações de redundâncias nos pontos de partida e chegada.
  - Nessa linha, determinar com que redundâncias se começou e acabou nos exemplos ilustrados de seguida:



## Exercício 3 - Função InitRobot()

**Criar** `[H,h,P,AAA]=InitRobot(QQ, NN, DH, jTypes, sScale)`

- Para facilitar os procedimentos de preparação e representação inicial de um robô, recomenda-se criar uma função que aceita os elementos base e devolve *handles* gráficos e o cálculo do movimento para a animação e outras operações de simulação.
- Usar as funções criadas anteriormente:
  - `MQ=LinspaceVect(Qi,Qf,N)`
  - `MDH=GenerateMultiDH(DH,MQ,jTypes)`
  - `AAA=CalculateRobotMotion(MDH)`
  - `Org=LinkOrigins(AA)`
  - `h=DrawLinks(Org)`
  - `[P,F]=seixos3(sScale)`
  - `H=DrawFrames(AA,P,F)`

# Argumentos e valores de retorno de InitRobot()

```
[H,h,P,AAA]=InitRobot(QQ, NN, DH, jTypes, sScale)
```

Argumentos de entrada:

- QQ - matriz de colunas das posições de junta (pelo menos duas)
- NN - número de pontos de cada segmento do movimento
- DH - Matriz-base dos parâmetros cinemáticos
- jTypes - vetor com o tipo de juntas (0=rot, 1=prism; opcional)
- sScale - Fator de escala dos seixos3() (opcional: 1 por defeito)

Valores de retorno:

- H - *Handles* gráficos dos sistemas de eixos
- h - *Handle* gráfico dos elos do robô
- P - Objeto do seixos3() na escala usada
- AAA - *Superhipermatriz* com as transformações das posições do robô

## Exercício 3b - Teste da função InitRobot

### Exemplo para testar a função InitRobot()

Animar um RR planar entre a posição (4,0) e a posição (-2,2) usando a redundância de cotovelo para baixo:

```
L1=4; L2=2; %Link lengths
DH=[0 L1 0 0; 0 L2 0 0]; %Base DH matrix
Qi = invkinRR( 4, 0, L1, L2); %Starting configuration
Qf = invkinRR(-2, 2, L1, L2); %Ending configuration
QQ=[Qi(:,1) Qf(:,1)]; %Use redundancy 1
axis equal; axis([-6 6 -3 6 -6 6]); %Fix/adjust axis
hold on; grid on; %Hold on for graphics
xlabel('X');ylabel('Y');zlabel('Z');%Put labels

[H,h,P,AAA]=InitRobot(QQ,50,DH); %Do it all ;- )

pause %Press key to continue
AnimateRobot(H,AAA,P,h,0.05,1); %Do the animation
```

## Exercício 4 - RobotEndPath

### Criar a função `[X,Y,Z]=RobotEndPath(AAA)`

- A função deve determinar o caminho que o robô irá percorrer no movimento.
- `[X,Y,Z]`: vetores das coordenadas sucessivas do *end-effector* do robô.
- `AAA`: a sequência de hipermatrizes calculadas por `CalculateRobotMotion`.
- As coordenadas são a extremidade do robô, ou seja, a 4ª coluna do produto acumulado das transformações geométricas de todos os elos.

### Sugestão parcial de código a completar:

```
function [X,Y,Z]=RobotEndPath(AAA)
for n=1:size(AAA,4)
    A= %...
    for j=1:size(AAA,3)
        A=A*%...
    end
    X=%...;
    Y=%...;
    Z=%...;
end
```



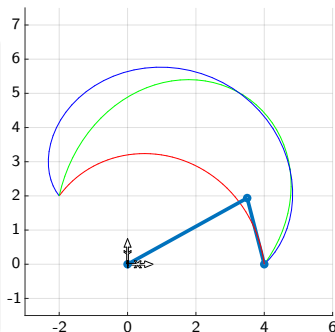
# Exercício 5 - Caminhos nas várias redundâncias

## Representar os diversos caminhos do end-effector de um RR planar

- O robô tem  $L_1=4$ ,  $L_2=2$  e movimenta-se do ponto  $[4,0]$  até  $[-2,2]$  nas 3 seguintes combinações:
  - 1 Percurso todo na redundância 2 (cotovelo para cima).
  - 2 Começar na redundância 1 e terminar na redundância 2
  - 3 Começar na redundância 2 e terminar na redundância 1

## Alguns passos do processo

```
Qi=invkinRR(4,0,L1,L2);  
Qf=invkinRR(-2,2,L1,L2);  
MQ=LinspaceVect(Qi(:,2),Qf(:,2),50);  
MDH=GenerateMultiDH(DH,MQ);  
AAA=CalculateRobotMotion(MDH);  
  
[x,y,z]=RobotEndPath(AAA);  
plot3(x,y,z,'r-');  
%...
```



## Exercício 6 - Cinemática inversa do RRR planar

**Escrever a função**  $Q = \text{invkinRRR}(x, y, \phi, L1, L2, L3)$

- Esta função calcula a cinemática inversa do RRR planar.
- $x, y, \phi$  - coordenadas do *end-effector*
- $L1, L2, L3$  - comprimentos dos elos
- $Q$  - vetor com o valor das três juntas com uma redundância por coluna

### Metodologia:

- Usar a cinemática inversa do RR planar para o punho ( $P_{w_x}, P_{w_y}$ ) que, recorde-se, é dado por:
  - $P_{w_x} = x - L_3 \cos \phi$
  - $P_{w_y} = y - L_3 \sin \phi$

# Exercício 7 - Exemplo de aplicação no RRR planar

## Animar RRR planar num movimento específico

- Simular um movimento de elevação do *end-effector* mas forçando o 3º elo sempre na horizontal.
- Os comprimentos dos elos são  $L1=2$ ,  $L2=1$ ,  $L3=1$
- O ponto de partida é  $[2, -1]$  e o de chegada  $[2, 2.5]$
- O movimento deve ser feito com o cotovelo em baixo

