



Robótica Espacial

Aula prática nº 1

Transformações geométricas elementares
Transformações geométricas homogéneas a 2D e 3D

Vitor Santos

Universidade de Aveiro

12 Feb 2025

- 1 Recomendações iniciais
- 2 Exercício 1 - Representação de um polígono simples
- 3 Exercício 2 - Múltiplos objetos por transformação geométrica
- 4 Exercício 3 e 4 - Pré- e pós multiplicações em coordenadas homogéneas
- 5 Exercício 5 - Transformações homogéneas a 3D
- 6 Exercício 6 - Movimentar objetos por animação
- 7 Exercício 7 - Primeiro contacto com o modelo do UR3

Recomendações para realizar os exercícios

- Para dispor de funções adicionais, instalar a Toolbox proprietária:
 - Robotics Toolbox for Matlab Release 10, Peter Corke, 2020 (RTB)
 - Está disponível na página do-elearning. Seguir as instruções.
- Criar uma pasta por aula (Aula1, Aula2, etc.)
- Há duas metodologias principais para resolver os exercícios em cada aula:
 - Cada exercício está num ficheiro diferente:
 - a1ex1.m , a1ex2.m, etc.
 - Vários exercícios no mesmo ficheiro:
 - aula1.m, etc.
- Os dois métodos têm vantagens e desvantagens.

Caso dos exercícios num único ficheiro

- Na opção de usar todos os exercícios de uma aula no mesmo ficheiro, os exercícios devem ficar em secções distintas, demarcadas com comentários duplos, como no seguinte exemplo com dois exercícios:

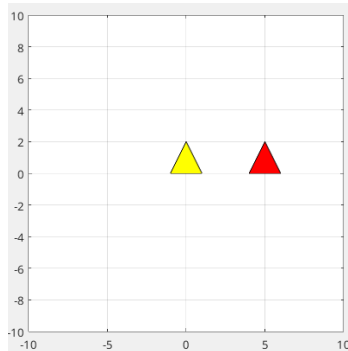
```
1 %% Ex1
2 - Z=zeros(100,200);
3 - imshow(Z)
4
5 %% Ex2
6 - X=ones(100,200);
7 - imshow(X)
8
```

- As secções podem ser executadas individualmente (CTRL+ENTER).
 - NB. Pode haver o risco de interferência de variáveis entre as diversas secções. Se não for para usar variáveis de umas secções para outras, recomenda-se um comando “clear” no início da secção para as limpar.
- Em certos casos será útil criar funções para código usado frequentemente. Nesse caso, as funções deverão ficar cada uma num ficheiro separado.
 - N.B.** É possível criar funções com nomes iguais a funções já existentes no MATLAB. Será executada a primeira que surgir no path. Este é um ponto a ter em atenção porque pode gerar resultados inesperados!

Exercício 1 - Representação de um polígono simples

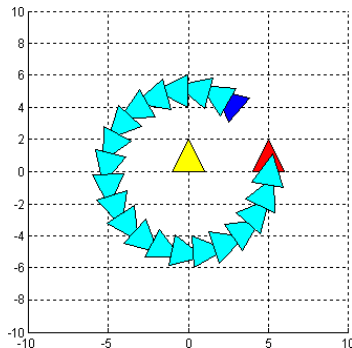
Fazer um script onde:

- Se represente um triângulo $A1$ amarelo com os vértices:
 - $P1 = [-1 \ 0]^T$ (em matlab: $P1 = [-1 \ 0]'$)
 - $P2 = [1 \ 0]^T$
 - $P3 = [0 \ 2]^T$
 - $A1 = [P1 \ P2 \ P3]$
- Usar a função `fill` do MATLAB
 - `h = fill(A1(1,:), A1(2,:), 'y');`
- Criar e representar um novo triângulo (vermelho) $A2$ por translação de $A1$ com o vetor v : $A2 = A1 + v$
 - $v = [5 \ 0]^T$
- Usar uma janela entre -10 e $+10$ nos dois eixos
 - `axis([-10 10 -10 10])`



Exercício 2 - Múltiplos objetos por transformação geométrica

- Criar o triângulo $A3$ por rotação de $A2$ de um ângulo de 50° .
 - Usar uma matriz de transformação geométrica apropriada (2×2) para obter $A3$ ou em alternativa usar a função `rot2()` da RTB. $A3 = \text{rot2}(50^\circ) \times A2$
 - Atender aos ângulos em graus ou radianos.
- Gerar uma sequência de triângulos obtidos pela rotação sucessiva de $A2$ em N valores incrementais a começar em 60 e a terminar em 350. (Experimentar com $N = 10, 20, 50$).
 - Usar um ciclo `for`
- Modificar o programa de forma a fazer uma animação suave alterando as propriedades do objeto `h` em vez de desenhar os triângulos todos como aconteceria repetindo o `fill()`.
 - `h.XData=...` % ajuste das coordenadas X
 - `h.YData=...` % ajuste das coordenadas X



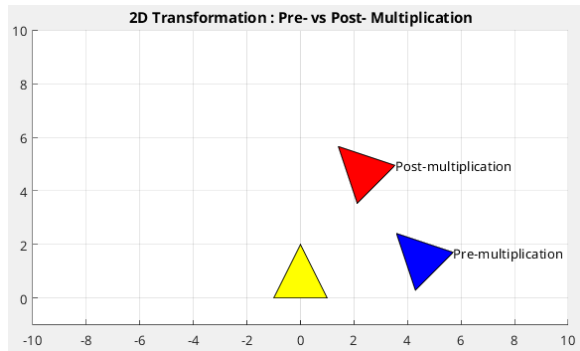
Exercício 3 - Polígono simples e transformações homogêneas

Escrever um *script* onde:

- repetindo os elementos do exercício 1, se defina e represente o triângulo $A1$ com os vértices:
 - $P1 = [-1 \ 0]^T$, $P2 = [1 \ 0]^T$, e $P3 = [0 \ 2]^T$
 - $A1 = [P1 \ P2 \ P3]$
- Converter os pontos de $A1$ para formato homogêneo:
 - Pode-se acrescentar uma linha de 1s na matriz $A1$ ou...
 - ... aplicar a função `e2h()`: $A1=e2h(A1)$;
- Para representar o polígono pode-se usar a função `fill` do MATLAB como anteriormente:
 - `h = fill(A1(1,:), A1(2,:), 'y');` ou ...
 - ... usar a mais amigável função `plot_poly`: `h=plot_poly(A1,'fillcolor','y');`
- Usar uma janela entre -10 e $+10$ nos dois eixos
 - `axis([-10 10 -10 10])`
- criar duas transformações homogêneas T e R e verificar o seu valor:
 - T é uma translação de $[5 \ 1]$: Usar a função `transl2()`
 - R é uma rotação de $\pi/4$ rad; Usar a função `trot2()`

Exercício 4 - Representar a pré- e pós-multiplicação de R com T

- A pré-multiplicação de R com T ($T \cdot R$) [é o mais comum]
- A pós-multiplicação de R com T ($R \cdot T$) [tem um significado particular]
- Calcular:
 - `pre_mult_result = T * R * A1;`
 - `pos_mult_result = R * T * A1;`
- Representar o objeto transformado em cada caso e interpretar o resultado.



Exercício 5 - Transformações homogêneas a 3D

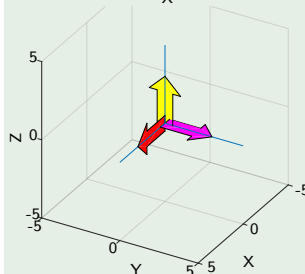
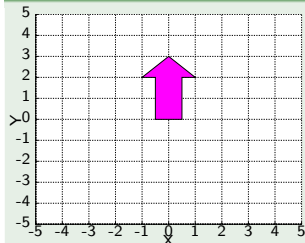
Criar em 3D o objeto P indicado (a seta) no plano XY

- Sugere-se as seguintes coordenadas: $P = \begin{bmatrix} 0.5 & 0.5 & 1 & 0 & -1 & -0.5 & -0.5 \\ 0 & 2 & 2 & 3 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$
- Representá-lo em 3D usando o comando `fill3()` do MATLAB ou a função mais amigável `plot_poly()`.
- Mediante as transformações geométricas `trotx()`, `troty()`, etc., criar e representar réplicas da seta P nos outros dois planos (XZ e YZ).

Notas:

- As transformações requerem coordenadas homogêneas; por isso recomenda-se usar a função `e2h()`, que converte coordenadas euclidianas para homogêneas, antes de aplicar as transformações.
- Todavia, a função `plot_poly()` espera coordenadas euclidianas, logo os resultados das transformações (que são em coordenadas homogêneas) devem ser "des-homogeneizados" usando, por exemplo, a função `h2e()`.
- Os sistemas de eixos podem ser desenhados com `trplot()`.

Seta em XY

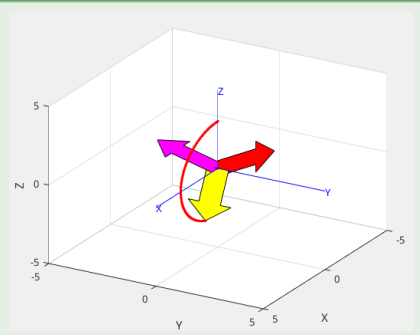


Exercício 6 - Movimentar objetos por animação

Colocar os 3 objetos em movimento simultâneo

- Animar os três objetos (setas) de tal forma que rodem em torno do eixo onde têm a base apoiada:
 - Fazê-los dar 10 voltas completas girando à mesma velocidade.
 - Usar as rotações apropriadas para cada caso ($trotx$, $troty$, $trotz$).
- Ajustar o código para que girem a velocidades diferentes:
 - Por exemplo: $1x$, $2x$ e $3x$;
 - NB. Neste caso só um deles dará 10 voltas porque devem parar todos ao mesmo tempo!

Setas em rotação



Como traçar a trajetória da ponta?

Exercício 7 - O modelo do UR3 em Matlab

- Carregar o modelo do UR3
 - `ur3 = loadrobot('universalUR3', 'DataFormat', 'row');`
- Visualizá-lo na posição de defeito: `juntas=(0,0,0,0,0,0)`
 - `show(ur3)`
- Colocar as 3 juntas primárias em (0, -135, +90) respetivamente
 - `juntas=zeros(1,6);`
 - `juntas(1)=0; juntas(2)=???; juntas(3)=???;`
 - `show(ur3,juntas)`
- Fazer o movimento animado da junta 2 de 0 a -135 em 50 passos
- Excerto relevante do programa (completar)

```
for i = 1: ???  
    clf  
    juntas(2)= ???  
    show(ur3,juntas,'PreservePlot',false,'FastUpdate',true);  
    pause(0.05)  
end
```

