

**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS**



**SEM0530 – Problemas de Engenharia Mecatrônica II
1º Semestre de 2022**

Prática 6 – Transformação de vetores

Integrantes:

Matheus Della Rocca Martins

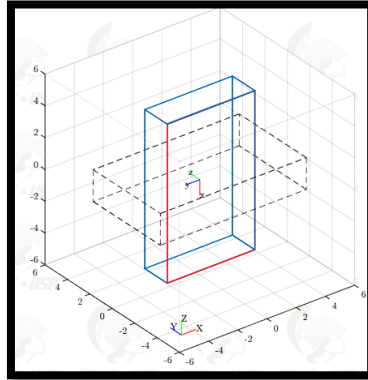
Nº USP : 12549731

São Carlos
15/07/2022

1. OBJETIVOS

O Problema fornecido propõe a simulação da rotação de um corpo rígido para uma sequência pré-determinada de rotações.

Figura 1: Problema Proposto



Fonte: Figura retirada dos slides do professor Marcelo A. Trindade

2. RESOLUÇÃO NO MATLAB

Para atingir os objetivos propostos utilizou-se o software MATLAB com base nos códigos fornecidos pelo professor:

Dados: $\theta = -30,25$ graus ($N=31$) = -0,5279621rad;

$4\theta = -2,11185$ rad.

Inicialmente, para que a sequência de rotações ocorresse na ordem correta (z,y,x) foi adotou-se rotord = 321 que implica em $R = R_z \cdot R_y \cdot R_x$. Além disso, definiu-se phi com dois passos (ambos negativos para possibilitar o cálculo das rotações, uma vez que, o ângulo de rotação fornecido também foi negativo) e intervalos diferentes, para as rotações em z e y adotou-se um passo 4 vezes menor que para a rotação em x, desta forma, a rotação em x ocorre com velocidade 4 vezes maior que as outras.

Figura 3,4: Script MATLAB

```
Editor - C:\Users\Matheus\Downloads\rot3dmain_aula.m
fcnrot3d.m  rot3dmain_aula.m  +
1  % Função de Rotacao 3D em 3 eixos fixados
2  function rot3dmain(rot)
3
4  rot = 2;
5  if rot==1
6  % Rotacao R = Rotacao de Rotacao z y x em 3 eixos fixados
7  rotord=321; % Ordem de rotacao
8  phi=[0:-pi/400:(-0.5279621)];
9  phizv=[phi phi(end)+0*phi phi(end)+0*phi]; % Rotacao de z
10 phiyv=[0*phi phi phi(end)+0*phi]; % Rotacao de y
11 phi=[0:-pi/100:(-2.11185)]; % Rotacao de x
12 phixv=[0*phi 0*phi phi]; % Rotacao de x
13 elseif rot==2
14 % Rotacao R = Rotacao de Rotacao y z x em 3 eixos fixados
15 rotord=321;
16 phi=[0:-pi/400:(-0.5279621)];
17 phizv=[phi];
18 phiyv=[phi];
19 phi=[0:-pi/100:(-2.11185)]; % Rotacao de x
20 phixv=[phi];
21 end
22
23 for id=1:length(phixv)
24     phix=phixv(id); phiy=phiyv(id); phiz=phizv(id);
25     Rx = [1 0 0; 0 cos(phix) -sin(phix); 0 sin(phix) cos(phix)];
```

```
23 - for id=1:length(phixv)
24 -     phix=phixv(id); phiy=phiyv(id); phiz=phizv(id);
25 -     Rx = [1 0 0; 0 cos(phix) -sin(phix); 0 sin(phix) cos(phix)];
26 -     Ry = [cos(phiy) 0 sin(phiy); 0 1 0; -sin(phiy) 0 cos(phiy)];
27 -     Rz = [cos(phiz) -sin(phiz) 0; sin(phiz) cos(phiz) 0; 0 0 1];
28 -     if rotord==321 R(:, :, id)=Rz*Ry*Rx;
29 -     elseif rotord==123 R(:, :, id)=Rx*Ry*Rz;
30 -     end
31 - end
32
33 - fcnrot3d(R)
34
```

No script fcnrot3d.m a única alteração feita foi comentar a linha 48 e adicionar view(3) para possibilitar a visualização 3d do gráfico:

Figura 5: Script MATLAB

```
Editor - C:\Users\Matheus\Downloads\fcrot3d.m
Pratica5.m x edo2.m x Untitled x fcrot3d.m x rot3dmain_aula.m x +
31 - set(brd,'linestyle',':', 'color',[1 1 1]*.7)
32 - exx=1; exy=0; exz=0;
33 - eyx=0; eyy=1; eyz=0;
34 - ezx=0; ezy=0; ezz=1;
35 - ex=line([0 exx],[0 exy],[0 exz]); set(ex,'color','r')
36 - ey=line([0 eyx],[0 eyy],[0 eyz]); set(ey,'color','b')
37 - ez=line([0 ezx],[0 ezy],[0 ezz]); set(ez,'color','g')
38 - tx=text(exx,exy,exz,'x');
39 - ty=text(eyx,eyy,eyz,'y');
40 - tz=text(ezx,ezy,ezz,'z');
41 - mxE=mx*.9;
42 - Ex=line([0 exx]-mxE,[0 exy]-mxE,[0 exz]-mxE); set(Ex,'color','r')
43 - Ey=line([0 eyx]-mxE,[0 eyy]-mxE,[0 eyz]-mxE); set(Ey,'color','b')
44 - Ez=line([0 ezx]-mxE,[0 ezy]-mxE,[0 ezz]-mxE); set(Ez,'color','g')
45 - Tx=text(exx-mxE,exy-mxE,exz-mxE,'X');
46 - Ty=text(eyx-mxE,eyy-mxE,eyz-mxE,'Y');
47 - Tz=text(ezx-mxE,ezy-mxE,ezz-mxE,'Z');
48 - rot3d([1 1 1],[1 1 1],[1 1 1]);
49 - view(3)
50 - axis equal,
51 - grid,
```

Figura 6,7: Posição inicial

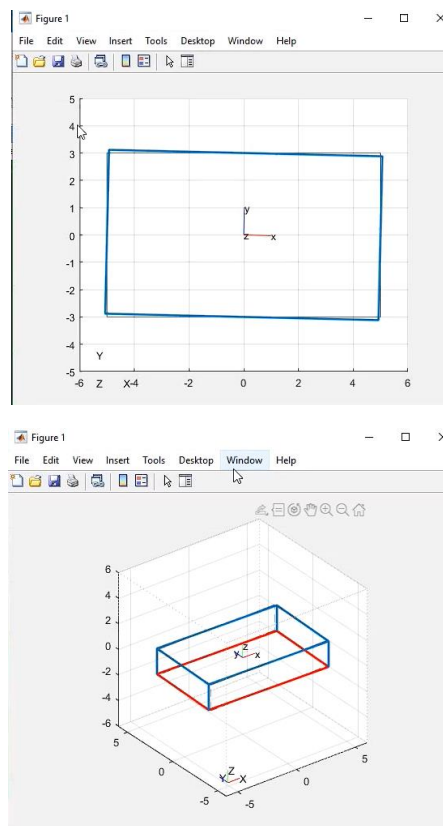


Figura 7,8: Rotação em torno de z

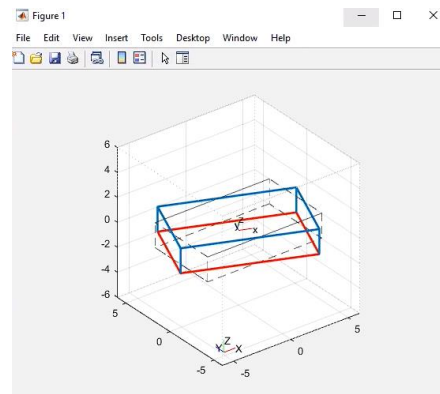
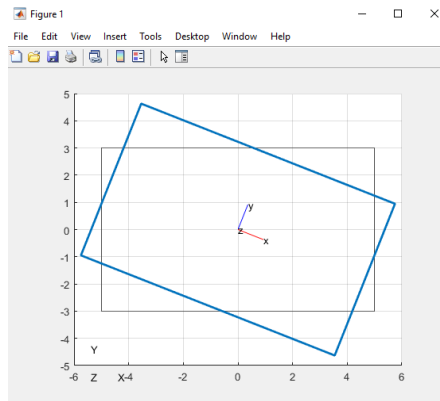


Figura 7: Rotação em torno de y

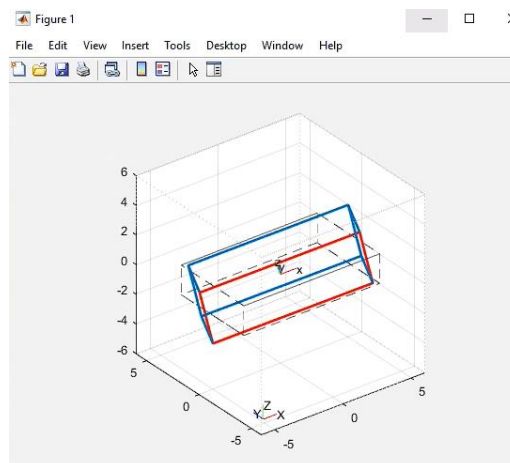
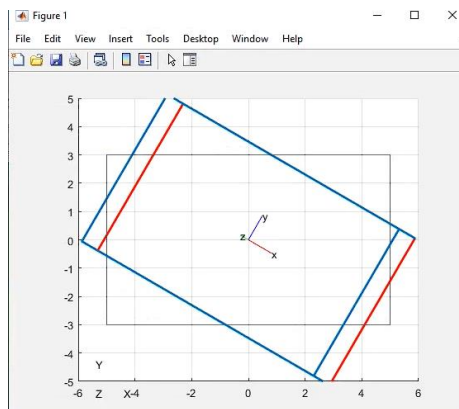
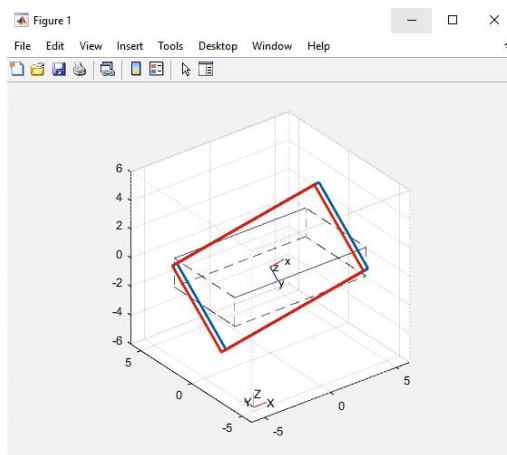
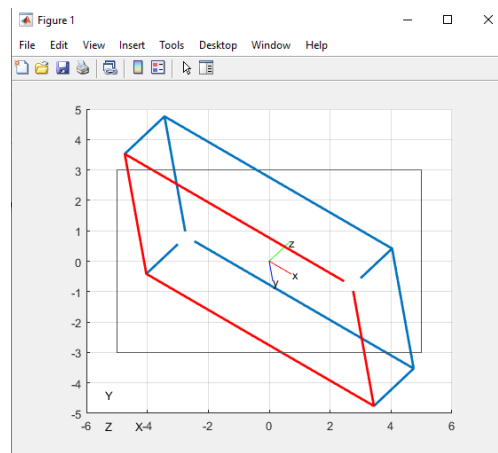


Figura 8: Rotação em torno de x



3. CONCLUSÕES

A realização do trabalho proposto possibilitou o entendimento sobre a de métodos numéricos através do MATLAB para realização de sequência de rotações, simultâneas ou não, sobre um corpo rígido. Por fim, foi possível visualizar essas rotações de forma bidimensional e tridimensional.

Essa prática abrange uma área de conhecimento com grande aplicabilidade para cursos de engenharia. Matrizes de rotação podem ser utilizadas, por exemplo, em esquemas de controle de formações de *swarm* de drones e, a sua visualização facilitaria o processo de desenvolvimento do projeto.