

plot_uform_transformation

Arquivo complementar. Recebe uma transformacao homogenea no formato do usuario [x y theta] com relacao ao sistema de coordenadas da origem do grafico e plota o sistema de coordenadas apos a transformacao

Calling Syntax	1
I/O Variables.....	1
Example	1
Hypothesis.....	2
Limitations.....	2
Version Control	2
Function.....	2
Validity.....	2
Main Calculations.....	2
Output Data.....	2

Calling Syntax

`plot_uform_transformation(uform)`

`plot_uform_transformation(uform, color)`

`plot_uform_transformation(uform, color, length)`

`plot_uform_transformation(uform, color, length, width)`

I/O Variables

IN Double Array **uform** *User Form*: Vetor [x y theta] que representa uma transformacao homogenea no formato do usuario

IN String **color** *Plot Colot*: String para cores padrao do matlab

IN Double Scalar **length** *Vector Size*: Multiplicador do tamanho da linha que indica o versor da transformacao

IN Double Scalar **width** *Vector Line Width*: Parametro de espessura da linha que indica o versor da transformacao

Example

```
brela = [1.0 2.0 30]
```

```
color = 'r'
```

```
plot_uform_transformation(brela, 'g', 0.5, 3.0)
```

Hypothesis

A transformacao expressa em *uform* refere-se ao centro (0,0) do plano cartesiano plotado

Limitations

Limita-se ao plano XY para transformacoes com relacao ao sistema de origem cartesiano (0,0)

Version Control

1. Gustavo Gransotto & Lucas Sponchiado (2020/09/23) - Versão inicial
2. Gustavo Gransotto & Lucas Sponchiado (2020/09/26) - Compatibilizacao R2015a
3. Gustavo Gransotto & Lucas Sponchiado (2020/10/01) - Alteracoes na Publicacao

Function

```
function plot_uform_transformation(uform, color, length, width)
```

Validity

```
if nargin == 1
    color = 'r';
    length = 2;
    width = 2.5;
elseif nargin == 2
    length = 2;
    width = 2.5;
elseif nargin == 3
    width = 2.5;
elseif nargin > 4 || nargin < 1
    errID = 'myComponent:InputError';
    throw(MException(errID,'Invalid number of arguments'));
end
```

Main Calculations

```
hold on;
axis equal;

x = uform(1);
y = uform(2);
theta = uform(3);

quiver(x, y, length*sind(-theta), length*cosd(-theta), ...
    color, 'Linewidth', width);
quiver(x, y, length*sind(-theta + 90), length*cosd(-theta + 90), ...
    color, 'Linewidth', width);
```

Output Data

```
end
```

