SOLVE_ROBOT

Table of Contents


```
Escreve a solução para o manipulador planar RRR com base no objetivo
instituido a partir do sistema da estação
```

Calling Syntax

```
function [near,far,sol]=solve_robot(goal,current,trelw,srelb,L,thetalim)
```

I/O Variables

```
|IN Double Matrix| *goal*: goal objective of the robot  Homogeneous
Transformation Matrix 4x4
|IN Double Matrix| *wrelb*: W rel to B  Homogeneous Transformation
Matrix 4x4
|IN Double Array| *current*: joint angle degrees  [theta1 theta2 theta3]
[degrees degrees degrees]
|IN Double Matrix| *srelb*: S rel to B  Homogeneous Transformation
Matrix 4x4
|IN Double Array| *L*: link lenghts  [L1 L2 L3] [meters meters meters]
|IN Double Array| *thetalim*: Matrix of joint limits in degrees

|OU Double Array| *near*: Homogeneous Transformation Matrix 4x4
|OU Double Array| *far*: Homogeneous Transformation Matrix 4x4
|OU Bool| *sol*: boolean represent solution existence
```

Example

```
L = [0.5 0.3];
current = [45 30 -10];
thetalim = [-170 170;-170 170;-170 170];
srelb = utoi([-0.1, 0.3, 30]);
goal = utoi([0, 0, -90]);
trelw = utoi([0.1 0.2 30]);
[near,far,sol] = solve_robot(goal,current,trelw,srelb,L,thetalim);
```

Hypothesis

```
RRR planar robot.
```

Function

```
function
 [near,far,sol]=solve_robot(goal,current,trelw,srelb,L,thetalim)
    wrelt = tinvert(trelw);
    wrels = tmult(goal,wrelt);
    wrelb = tmult(srelb,wrels);
    [near,far,sol] = invkin(wrelb,current,L,thetalim);
end
```

```
near =

   91.9547  107.4576 -289.4123


far =

  161.7851 -107.4576 -144.3275


sol =

     1
```

Published with MATLAB® R2016a