```
clear all; close all; clc;

L = [.3 .2 .1];         % Kinematic Values
T = [1 1 1] * pi/50;    % Initial Pose
```

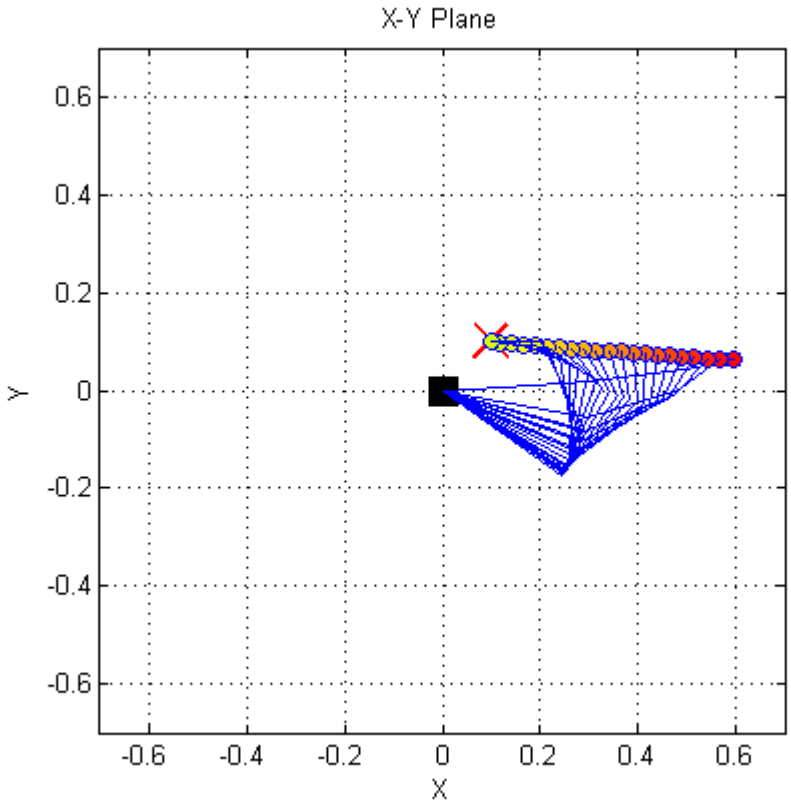
Set 1:

```
close all; clc;

G = [.1 .1]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
   -0.2676    2.2922    1.1246

** Error (Euclidean Distance to Gaol): 0.0016688 m (in 98 steps)
```



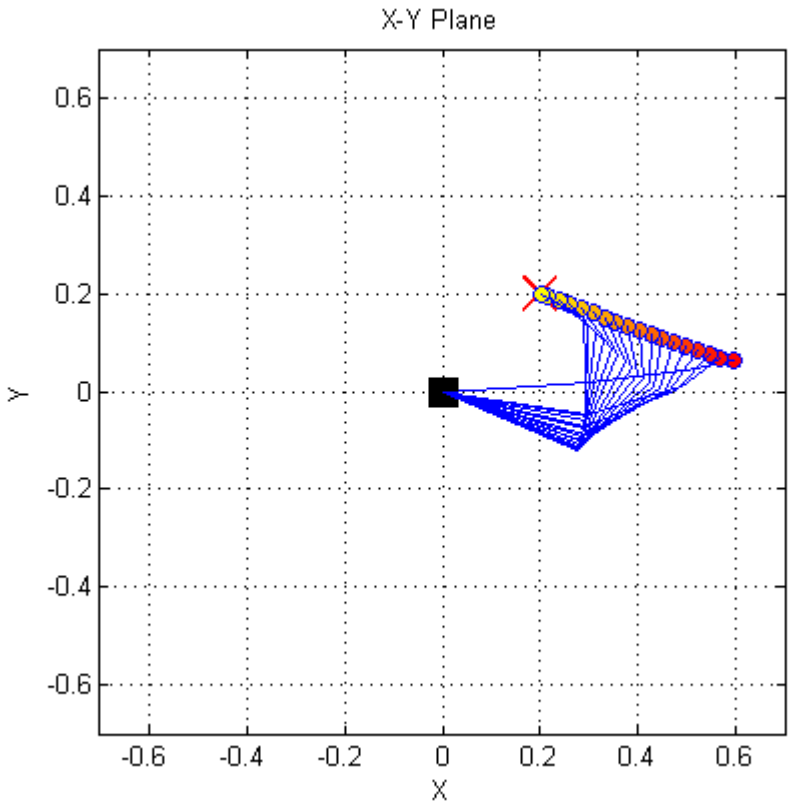
Set 2:

```
close all; clc;

G = [.2 .2]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
   -0.1685    1.7802    1.0059

** Error (Euclidean Distance to Gaol): 0.0010788 m (in 83 steps)
```



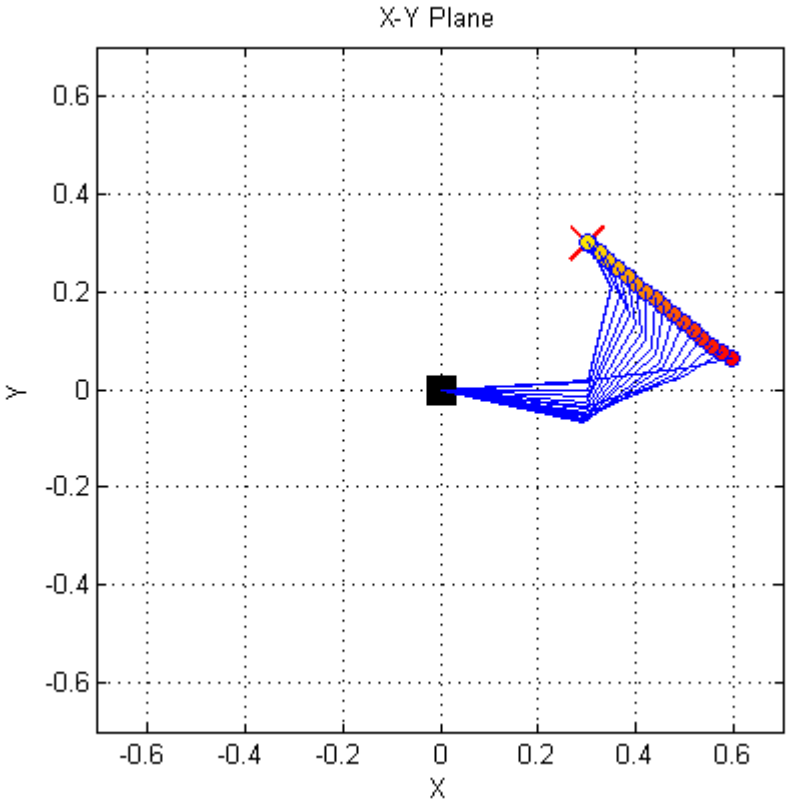
Set 3:

```
close all; clc;

G = [.3 .3]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
    0.0596    1.2635    0.7479

** Error (Euclidean Distance to Gaol): 0.00074182 m (in 76 steps)
```



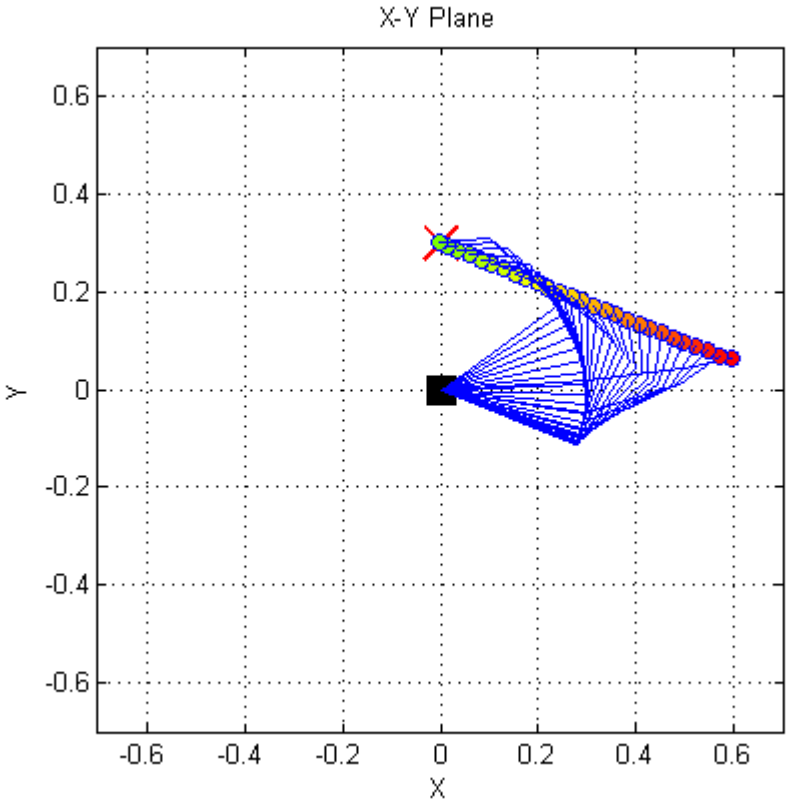
Set 4:

```
close all; clc;

G = [.0 .3]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
    0.6129    1.7805    0.8311

** Error (Euclidean Distance to Gaol): 0.00091374 m (in 128 steps)
```



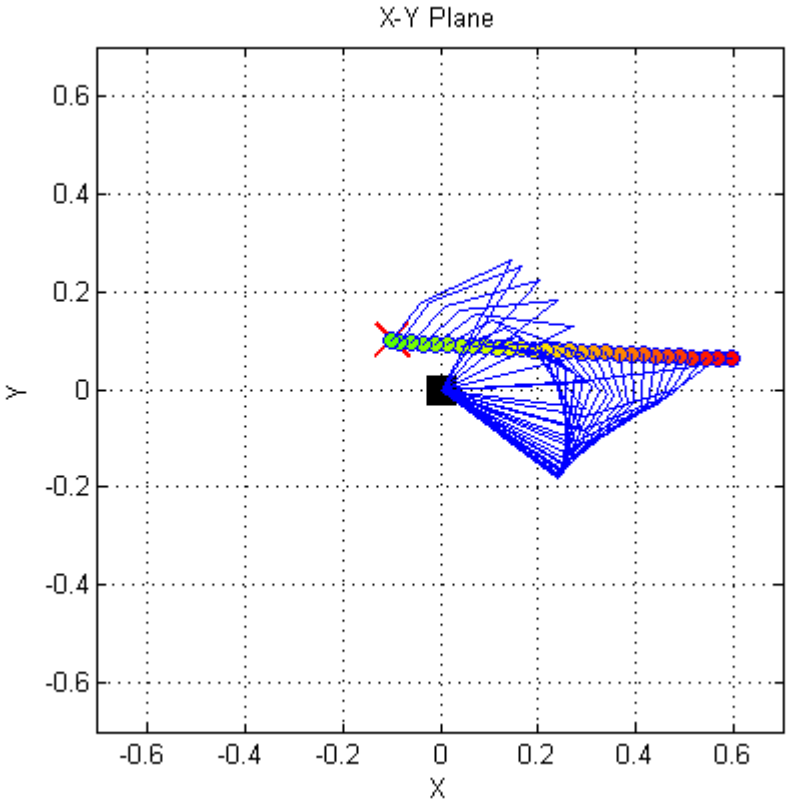
Set 5:

```
close all; clc;

G = [-.1 .1]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
    1.0714    2.5048    0.4832

** Error (Euclidean Distance to Gaol): 0.0015235 m (in 138 steps)
```



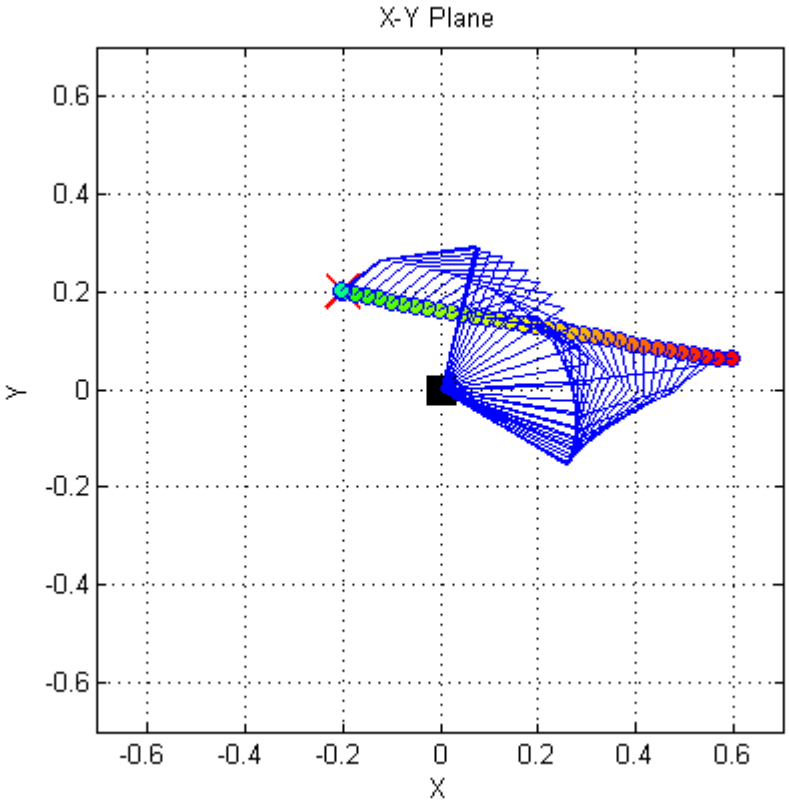
Set 6:

```
close all; clc;

G = [-.2 .2]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
    1.3231    1.9516    0.5561

** Error (Euclidean Distance to Gaol): 0.0019957 m (in 211 steps)
```



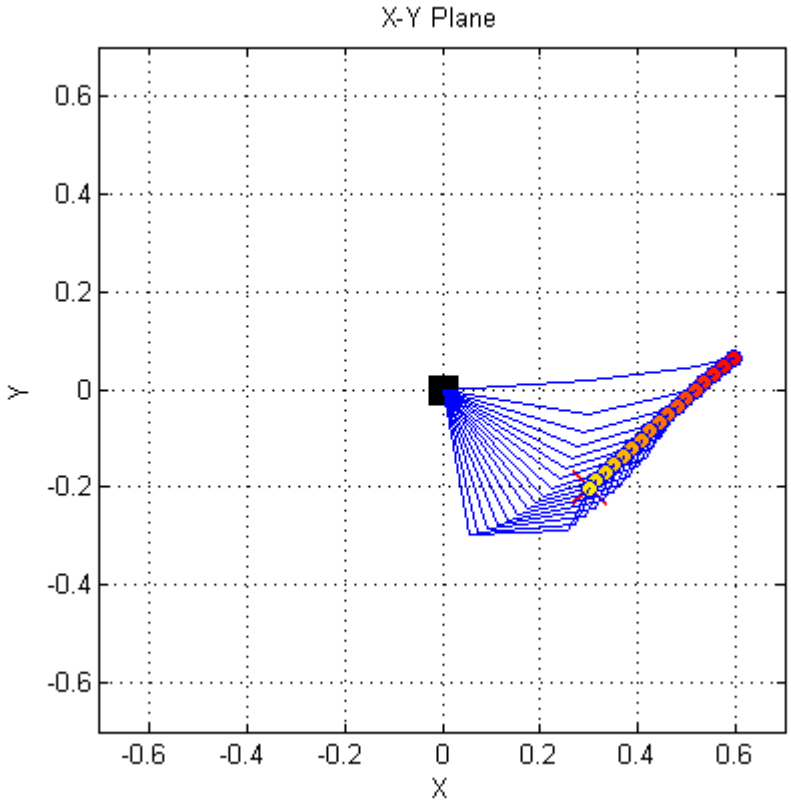
Set 7:

```
close all; clc;

G = [.3 -.2]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
   -1.3874    1.4103    1.0871

** Error (Euclidean Distance to Gaol): 0.0012063 m (in 79 steps)
```



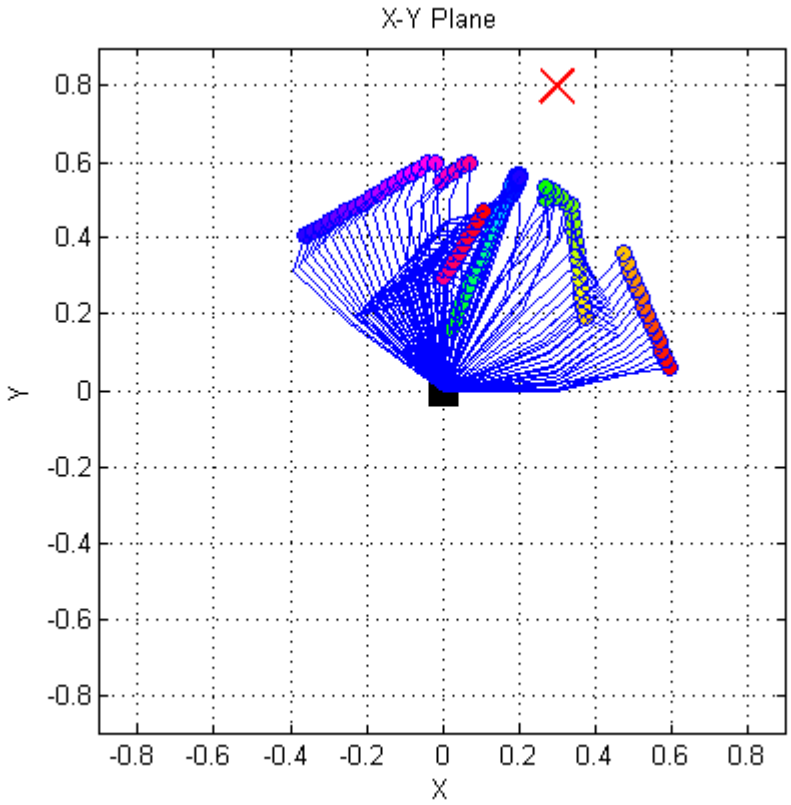
Set 8:

```
close all; clc;

G = [.3 .8]';           % Desired EE location
jacobianIK(L, T, G);
```

```
Warning: Goal is out of reach! But lets try ...
Now up to 500 iterations!
That is the max and cannot reach the goal ...
Perhaps it is out of reach!
*  Joint Space Solution (rad.):
  -60.9124   99.5726   68.4262

** Error (Euclidean Distance to Gaol): 0.37918 m (in 501 steps)
```



Test 1 with 11 DOFs:

```
close all; clc;
L = [.3 .2 .1  .1 .2 .1  .3 .1 .1  .1 .1];          % Kinematic Values
T = [.4 .6 .2  .5 .1 .7  .3 .2 .1  .1 .1] * pi/50;  % Initial Pose

G = [.3 .8]';           % Desired EE location
jacobianIK(L, T, G);
```

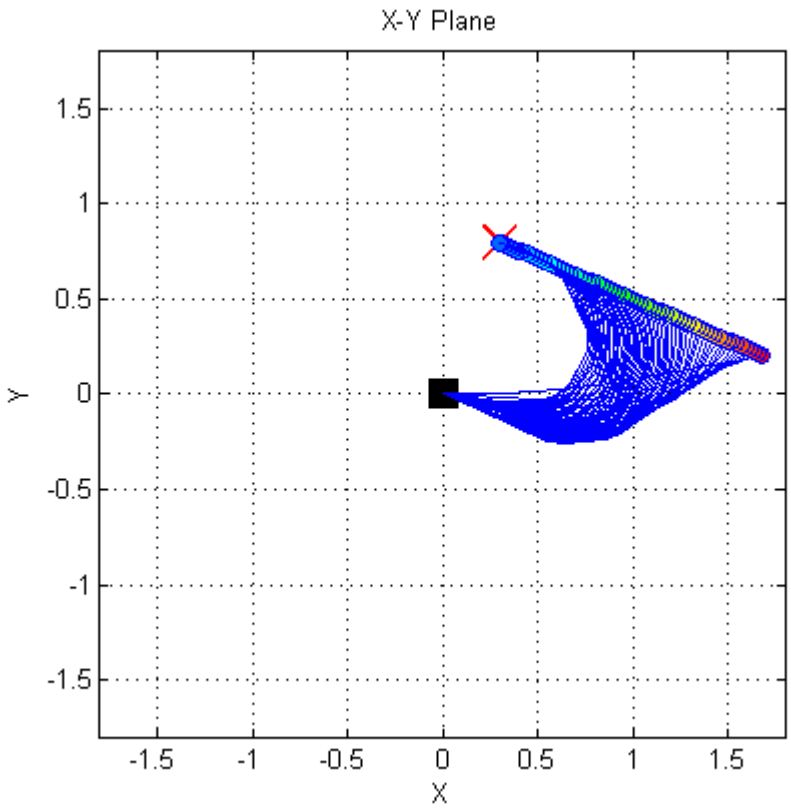
```
*  Joint Space Solution (rad.):
  Columns 1 through 7

   -0.1330    0.1413    0.2878    0.3792    0.4003    0.4858    0.4456

  Columns 8 through 11

    0.3122    0.2453    0.1726    0.0918

** Error (Euclidean Distance to Gaol): 0.0015369 m (in 301 steps)
```



Test 2 with 11 DOFs:

```
close all; clc;
L = [.3 .2 .1  .1 .2 .1  .3 .1 .1  .1 .1];          % Kinematic Values
T = [.4 .6 .2  .5 .1 .7  .3 .2 .1  .1 .1] * pi/50;  % Initial Pose

G = [-.5 .8]';           % Desired EE location
jacobianIK(L, T, G);
```

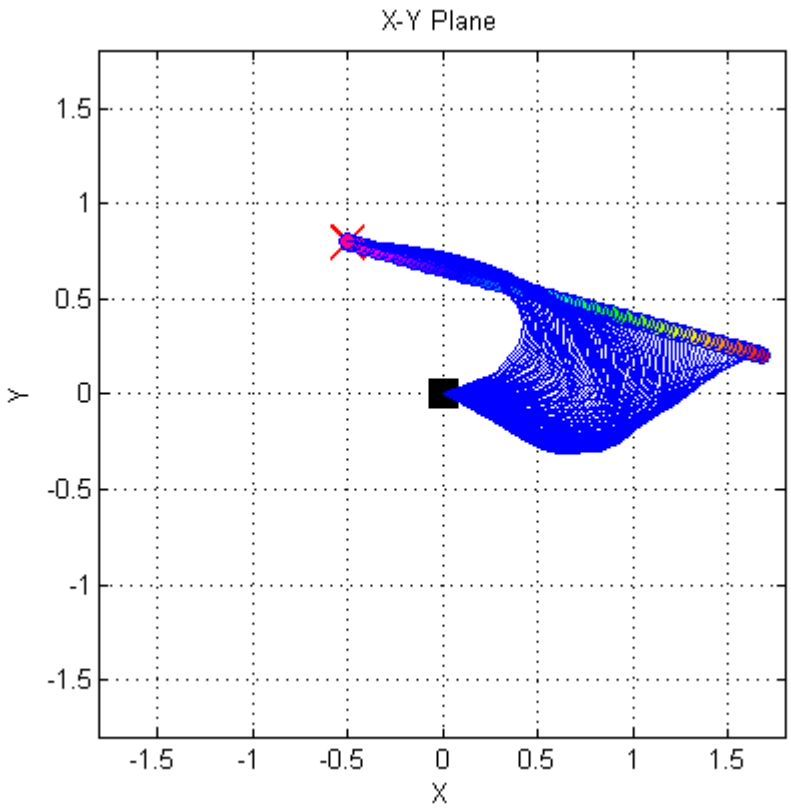
```
*  Joint Space Solution (rad.):
  Columns 1 through 7

    0.3943    0.4980    0.4856    0.4782    0.4003    0.3307    0.2498

  Columns 8 through 11

    0.1184    0.0780    0.0507    0.0277

** Error (Euclidean Distance to Gaol): 0.0018895 m (in 453 steps)
```



Test 3 with 11 DOFs:

```
close all; clc;
L = [.3 .2 .1  .1 .2 .1  .3 .1 .1  .1 .1];          % Kinematic Values
T = [.4 .6 .2  .5 .1 .7  .3 .2 .1  .1 .1] * pi/50;  % Initial Pose

G = [-.5 -1]';           % Desired EE location
jacobianIK(L, T, G);
```

```
*  Joint Space Solution (rad.):
  Columns 1 through 7

   -1.7190   -0.7657   -0.3345   -0.1251    0.0214    0.3916    0.5071

  Columns 8 through 11

    0.7028    0.6599    0.5094    0.2769

** Error (Euclidean Distance to Gaol): 0.0029787 m (in 501 steps)
```