Back  Chapter 6, Inverse Kinematics
Graded Quiz • 30 min

Due Oct 1, 11:59 PM +07

## Congratulations! You passed!

Grade received 100%   Latest Submission Grade 100%   To pass 80% or higher

Go to next item

**1.** Use Newton-Raphson iterative numerical root finding to perform two steps of finding the root of

1 / 1 point

$$f(x,y) = \begin{bmatrix} x^2 - 9 \\ y^2 - 4 \end{bmatrix}$$

when your initial guess is $(x^0, y^0) = (1,1)$. Give the result after two iterations $(x^2, y^2)$ with at least 2 decimal places for each element in the vector. You can do this by hand or write a program.

**Write the vector in the answer box and click "Run":**

[1.11,2.22,3.33] for $\begin{bmatrix} 1.11 \\ 2.22 \\ 3.33 \end{bmatrix}$.

```
[3.4,2.05]
```

Run

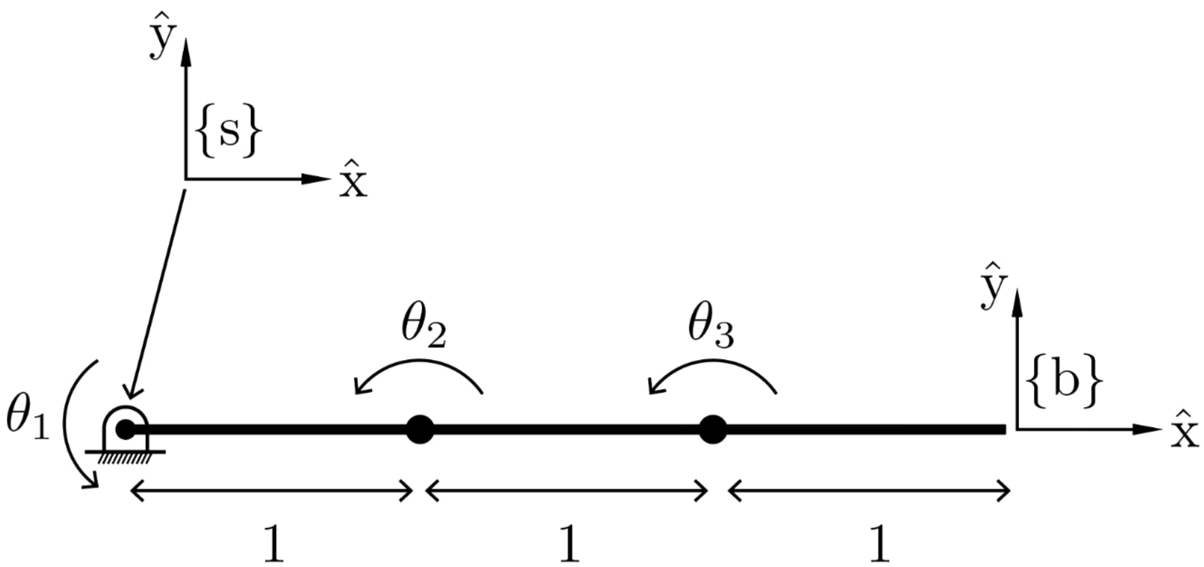
Reset

Correct

Good job!

**2.**

1 / 1 point




Referring to the figure above, find the joint angles $\theta_d = (\theta_1, \theta_2, \theta_3)$ that put the 3R robot's end-effector frame {b} at

$$T(\theta_d) = T_{sd} = \begin{bmatrix} -0.585 & -0.811 & 0 & 0.076 \\ 0.811 & -0.585 & 0 & 2.608 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

relative to the {s} frame, where linear distances are in meters. (The {s} frame is located at joint 1, but it is drawn at a different location for clarity.) The robot is shown at its home configuration, and the screw axis for each joint points toward you (out of the screen). The length of each link is 1 meter. Your solution should use either `IKinBody` or `IKinSpace`, the initial guess $\theta^0 = (\pi/4, \pi/4, \pi/4) = (0.7854, 0.7854, 0.7854)$, and tolerances $\epsilon_\omega = 0.001$ (0.057 degrees) and $\epsilon_v = 0.0001$ (0.1 mm). Give $\theta_d$ as a vector with at least 2 decimal places for each element in the vector. (Note that there is more than one solution to the inverse kinematics for $T_{sd}$, but we are looking for the solution that is "close" to the initial guess $\theta^0 = (\pi/4, \pi/4, \pi/4)$, i.e., the solution that will be returned by `IKinBody` or `IKinSpace`.)

**Write the vector in the answer box and click "Run":**

[1.11,2.22,3.33] for $\begin{bmatrix} 1.11 \\ 2.22 \\ 3.33 \end{bmatrix}$.

```
[0.9252,0.5862,0.6843]
```

Run

Reset

**Correct**

Good job!