

Lab 3,4 Question

1 Problem Statement

Design a Proportional Controller for a 2 Degrees of Freedom arm, such that; When given 2 different joint values as a client request, the bot moves it's joints to adjust accordingly.

Also, Analyze and understand how to tune p controller, and what improvements are to be given to it. Write a Report, with the details of the problem and how you approached to solve and tune it; preferably with graphs.

1.1 Details

Using the message `sensor_msgs/JointState` publish to the topic `/set_joint_state` with only two numbers in the position data field. This should, move the joints to their requested position. But this does not solve our problem. We want to control the arm from one position to the other. For this, you need to use a p controller to control each joints based on their current position. How do we do that?

Topic to get current Joint position from is: `/joint_states` with the same message `sensor_msgs/JointState`. To use the package given, run the command:

```
roslaunch arm_sim simenv.launch
```

Make a new package and a script inside of it, to publish to `/set_joint_state` and subscribe to `/joint_states` and finish the controller. **Note: Unless you run the above mentioned file, the topic `/set_joint_state` will not appear.**

1.1.1 Math Required

Just as we covered the math required to make a normal p controller for a differential drive, this is no different. But we are going to address the problem using matrices to make it extensible to n joints.

$$\begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} = (k1 \quad k2) * \begin{pmatrix} \theta_{1reference} - \theta_1 \\ \theta_{2reference} - \theta_2 \end{pmatrix}$$

Finally publish the new θ values after multiplying to the appropriate topic mentioned above.