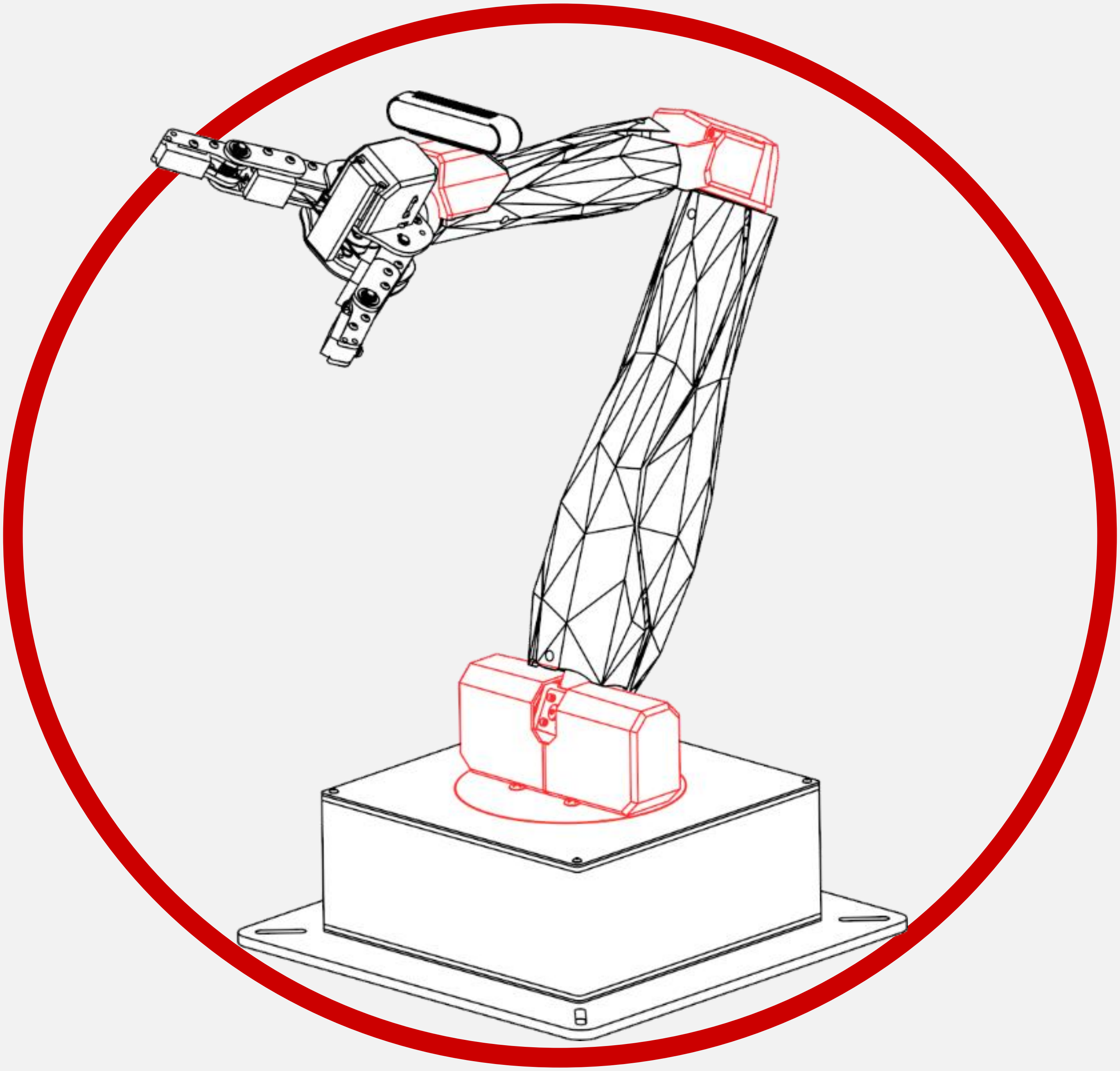


# QArm

## Singularity Avoidance

V1.0 – 13th February 2025

Quanser Consulting Inc. info@quanser.com
119 Spy Court Phone : 19059403575
Markham, Ontario Fax : 19059403576
L3R 5H6, Canada printed in Markham, Ontario.



**Caution**

**This equipment is designed to be used for educational and research purposes and is not intended for use by the public.** The user is responsible for ensuring that the equipment will be used by technically qualified personnel only. Users are responsible for certifying any modifications or additions they make to the default configuration.

# Singularity Avoidance

## What is Sigularity Avoidance?

One of the major problems that affects the operation of a robotic manipulator in task space is kinematic singularities. A singularity is a joint configuration that prevents the end-effector of a manipulator from moving in one or more directions. This loss of motion reduces the degrees-of-freedom of the manipulator in task-space. Once you have identified the singularities that affect the operation of your system, you must devise a strategy to avoid them. In this lab you will explore a singularity avoidance strategy that prevents one of the known singularities of the QArm, which is placing the end-effector directly above the base joint. The strategy involves placing a singularity avoidance volume – in the shape of a cylinder - around the base joint. If the manipulator is commanded a trajectory that intersects with the avoidance volume, it will track the trajectory until it reaches the avoidance area, then tracks the surface of the avoidance volume until the end-effector is clear of the area, hence avoiding the problematic region.

## Singularity Avoidance

Figure 1 shows the proposed cylindrical avoidance volume with radius R placed around the base joint of the manipulator. When operating the manipulator in task-space, we will devise a strategy that prevents the end-effector from entering the volume.

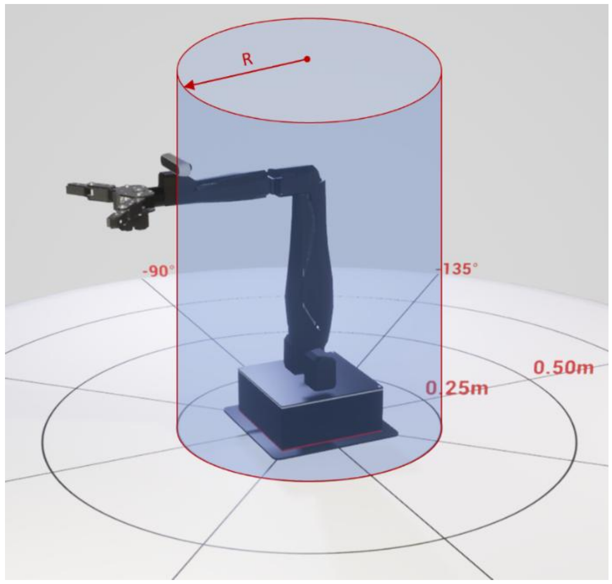



Figure 1: Singularity avoidance cylinder

Let us consider the example where the end-effector is commanded a linear trajectory that passes through the avoidance volume. This is schematically illustrated in Figure 2, which shows trajectory L intersecting with a horizontal cross section of the avoidance volume (a circle of radius R). One strategy is to allow the end-effector to track the linear commanded trajectory until it reaches the avoidance area at the **entry point** $\left[pT_x, pT_y\right]_{t=0}$. At this point the end-effector is command to track the surface of the avoidance volume (a circular trajectory) until it exits the area at the **exit point** $[x_f, y_f]$, after which it resumes tracking the original trajectory.

We can implement this strategy using the finite state machine (FSM) shown in Figure 3. The FSM consist of the following four states:

1. *State 1 – Normal Operation*: In this state the end-effector follows the desired linear trajectory. This state will be active prior to the end-effector reaching the singularity avoidance volume, as well as after it has steered clear of it. This serves as the initial state of the FSM.

2. *State 2 – Find Exit Point and Rate:* Once the end-effector is commanded a position that falls on or inside the avoidance (entry point), the FSM changes to this state. State 2 determines two key elements: the exit point at which the end-effector need to stop following the surface of the avoidance volume $[x_f, y_f]$, as well the rate at which the base joint needs to rotate as the end-effector follows the surface of the avoidance volume. Note from the diagram that this state only executes once.

3. *State 3 – Move Toward Exit Point at Rate:* Once we have determined the exit point as well at the rate at which the base joint needs to rotate, we remain in State 3, which keeps the end-effector moving along the surface of the avoidance volume, until it reaches the exit point.

4. *State 4 – Terminate:* This state prevents a false positive that incorrectly jumps to state 2. As soon as the measured position reaches *close enough* to the exit point, leaving state 3 to go back to state 1 might return a TRUE value for the commanded trajectory being on the circle or within in. Instead, moving to state 4 to wait until the desired trajectory is outside the avoidance volume prevents the loop.

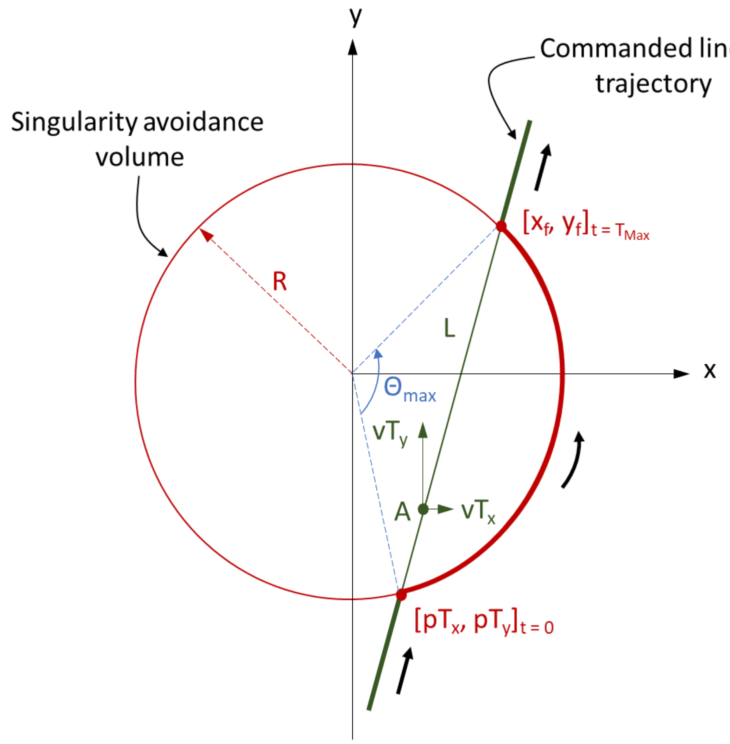
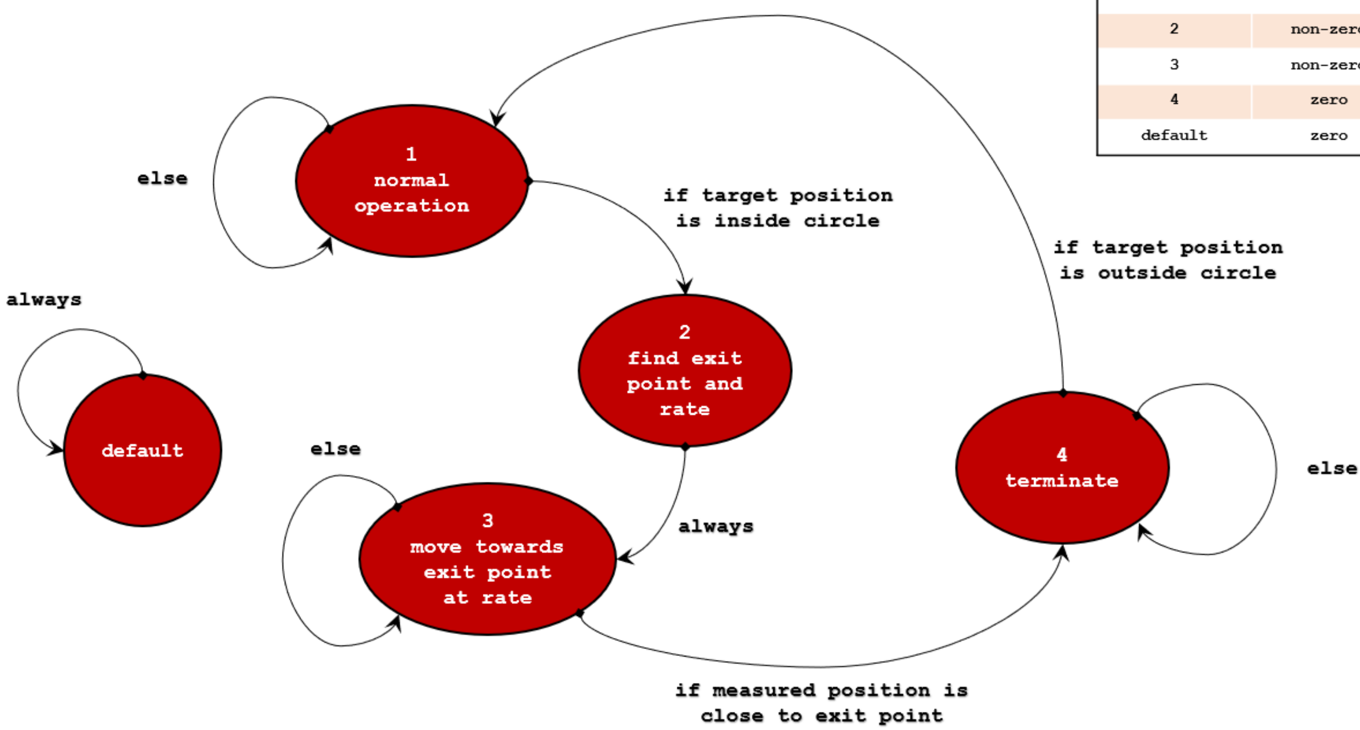



Figure 2: Trajectory L intersecting with the singularity avoidance volume (top view)




| state | speed |
|---|---|
| 1 | zero |
| 2 | non-zero |
| 3 | non-zero |
| 4 | zero |
| default | zero |

Figure 3: Finite state machine that implements a singularity avoidance strategy

## Calculating Exit Point and Rate

Let us have a closer look at how the coordinates of the exit point and rate of rotation is calculated in State 2 of the FSM. We can express the circular trajectory of the avoidance volume cross-section as shown in Figure 1.2 as follows:

$$x^2 + y^2 = R^2 \tag{1}$$

Furthermore, we can express the linear trajectory *L* as a function of time (t) in parametric form as follows:

$$\begin{aligned} x &= pT_x|_{t=0} + vT_x|_{t=0}\, t \\ x &= pT_x|_{t=0} + vT_x|_{t=0}\, t \\ y &= pT_y|_{t=0} + vT_y|_{t=0}\, t \end{aligned} \tag{2}$$

where $pT_x|_{t=0}$ and $pT_y|_{t=0}$ are the coordinates of the entry point at which the trajectory intersects with the avoidance region at time $t = 0$, and $vT_x|_{t=0}$ and $vT_y|_{t=0}$ are the velocity components of the end-effector at this instant. Here, the $p$ and $v$ denote position and velocity, respectively, and the T denotes that it is a target trajectory.

Since this calculation is carried out in state 2 only once, we can drop the $|_{t=0}$ flag. Substituting Equation 2 into Equation 1 (i.e. intersecting the equations of the circle and the line), and rearranging the equation yields:

$$
\begin{aligned}
&(pT_x + vT_x t)^2 + \left(pT_y + vT_y t\right)^2 = R^2 \\
&pT_x^2 + vT_x^2 t^2 + 2pT_x vT_x + pT_y^2 + vT_y^2 t^2 + 2pT_y vT_y = R^2 \\
&\left(vT_x^2 + vT_y^2\right)t^2 + 2\left(vT_x pT_x + vT_y pT_y\right)t + \left(pT_x^2 + pT_y^2 - R^2\right) = 0
\end{aligned} \tag{3}
$$

Solving the latter equation, yields the following two answers:

$$
t = 0, T_{max} \tag{4}
$$

where $T_{max}$ is the time it will take for the targe trajectory to reach the exit point. This is also the time we must rotate the base to get the end-effector to the exit point as well. Knowing the angle of the arc which the end effector needs to travel ($\Theta_{max}$), we can calculate the rate of rotation of the base joint as follows:

$$
rate = \frac{\theta_{max}}{T_{max}} \tag{5}
$$

Note that the angle $\Theta_{\mathrm{max}}$ can be calculated using the difference between the arctangents on the entry and exit points. Finally, calculate the coordinates of the exit point $(x_f, y_f)$ by substituting $T_{max}$ into Equation 2:

$$
\begin{aligned}
x_f &= pT_x + vT_x\, T_{max} \\
y_f &= pT_y + vT_y\, T_{max}
\end{aligned} \tag{6}
$$

## Background

The QArm content contains 3 labs that focus on velocity manipulation. They focus on tool manipulation, singularity identification and singularity avoidance respectively. This lab focuses on controlling the arm to avoid singularity positions so the arm always knows what position to go to.

## Getting started

The goal of this lab is to understand how to avoid singularity positions based on the singularity identification lab before.

Ensure you have completed the following labs

- **Kinematic Manipulation Labs**
- **Tool Manipulation**

- **Singularity Identification**

Before you begin this lab, ensure that the following criteria are met.

- The QArm has been setup and tested. See the QArm Quick Start Guide for details on this step.
- You are familiar with the basics of Simulink. See the Simulink Onramp for more help with getting started with Simulink.