Chapter 6

Racing on the Rainbow Road

6.1 Overview

3....2....1....Go!

During this module, we'll be taking our Neatokarts on a drive down Rainbow Road. But beware! This avenue of light has no bumpers or guardrails to keep us on track. Instead, we'll need to use our mastery of vector operations, parabolic curves, and odometry to successfully cross the finish line.

Learning Objectives

By the end of this challenge, you should be able to:

1. Compute tangent vectors and normal vectors to parametric curves, and connect these vectors to the position and motion of a mobile robot.
2. Derive a motion model of a wheeled robot.
3. Validate a motion model of a robot empirically.
4. Control a robot using an open-loop control strategy.
5. Map the path of a robot based on encoder values.

6.2 The Challenge

You will need to develop an autonomous pilot for your Neatokart to successfully navigate the cheerful, but treacherous, Rainbow Road from the start to the finish line. Our crack team has analyzed the pre-race track maps, and determined that the centerline of our Rainbow Road is defined by the following parametric curve:

$$\vec{r}(u) = 0.3960\cos(2.65(u+1.4))\hat{\imath} - 0.99\sin(u+1.4)\hat{\jmath},\ u \in [0, 3.2].$$

Note that u is not necessarily equal to time t, but must be some function of t so that you can find your desired velocities. You get to use your analytical skills to determine what is an appropriate choice of $u(t)$.

To approach this challenge, we'll be spending the next few weeks getting familiar with our Neatokarts and crafting a motion model that will help us ultimately design our autonomous pilot. Specifically, the end-goal here is to design an *open-loop controller* – a method of commanding a robot such that the sequence of motor commands (e.g., the velocities for each of the Neatokart's wheels) is set ahead of the race and executed at prescribed times irrespective of where the robot is along the path. Simple, open-loop control is incredibly powerful and used in many modern robotic systems today; and, it should be up to the task of getting us across the Rainbow Road.

6.2.1 A Neatokart for Two

We're asking you to work in groups of two on this project deliverable. Our expectations are that both team members will feel reasonably comfortable with all aspects of this project and will collaborate fully and equitably. You may want to consider taking a pair programming approach as you write Matlab code. Ask the teaching team (especially the CAs) about some best practices for pair programming. At the end of Rainbow Road we'll ask you to complete a short peer- and self- assessment survey.

Check the Rainbow Road Canvas module for specifics on how to form teams.

6.2.2 Neatokart Specifications

For this challenge, the maximum wheel speed of the Neatokart is ± 0.3 m/s and the commanded wheel speeds **should not** exceed this value. The wheelbase for the Neatokart is $d = 0.245$ m.

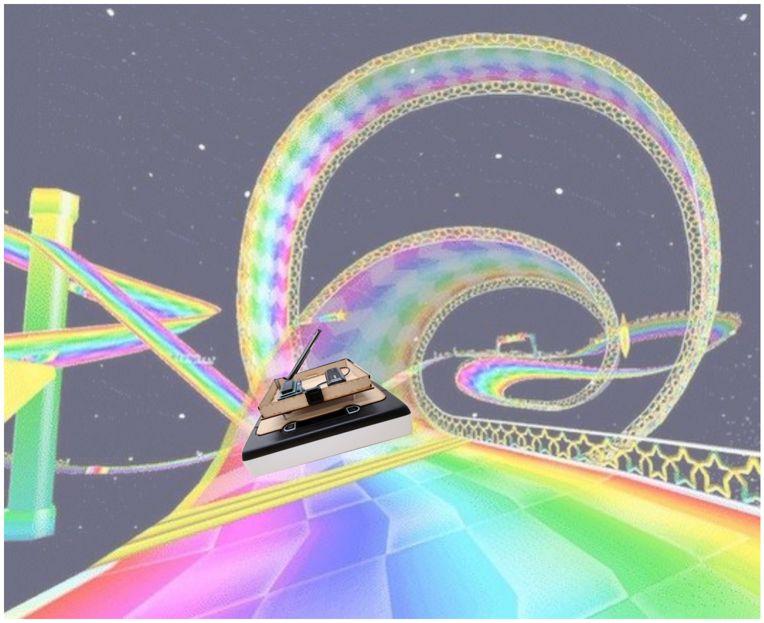

Figure 6.1: A Neatokart hurtling through the Rainbow Road on the way to the finish line.

6.3 A Roadmap to Success!

As we work through in-class and homework activities, we'll be systematically building out our autonomous pilot by defining a motion model for our Neatokart, analyzing our encoder values and estimating our Neatokart's position, and relating that to velocity (linear and angular). For this challenge, we want to leverage all of this information to command our Neatokart to travel a given parametric curve given a set of velocity commands we design. Let's consider our design process step by step:

Exercise 6.1

1. Plot the parametric curve that defines the centerline of the Rainbow Road.
2. On the same figure, plot the unit tangent and normal vectors at several points along the curve.

Exercise 6.2

1. Given the parametric curve, compute and plot your robot's planned linear speed and angular velocity at its center of mass, as a function of time, for the Rainbow Road.
2. Compute and plot your robot's left and right wheel velocities as a function of time for the Rainbow Road.

Exercise 6.3

Write a program to drive your Neatokart from the start to the finish line on Rainbow Road. To do this, you will need to send the appropriate control signal to your robot's wheels based on the time elapsed since the start of the path. Don't forget that you will command the left and right wheel velocities. **Take a video of your robot crossing the Rainbow Road.**

Exercise 6.4

Comparing your Measured and Experimental Path. Cross the Rainbow Road and capture the encoder values as your Neatokart traverses the path. Convert those measurements to coordinates and headings for the robot and plot the robot's measured journey. Include the predicted and experimental unit tangent vectors at various points along the curve (note: do not include an arrow for every time step or your plot will be too cluttered). Your plot should include the planned (theoretical) path plotted with a solid line, the experimental path plotted with a dashed line, and the tangent vectors for both. Make sure your plot includes appropriate units, labels, and legends.

6.4 Deliverables

Exercise 6.5

Your submission should contain the following components (5 pages maximum):

1. A single page description of your methodology for solving the challenge, explicitly identifying and explaining the assumptions, key steps, equations, and parameters needed to produce the plots and video identified above.
    - See the Rainbow Road assignment in Canvas for the grading rubric. This will clarify what you should include in this single page description.
2. A single page for each of the four plots detailed above in each of the 4 exercises with appropriate captions (i.e. four pages, each with one figure and caption). Note that a caption is a descriptive sentence describing what is being communicated via the plot (so not just a title).
    - The four plots are also listed on the Rainbow Road assignment in Canvas.
3. Links to both your code and a youtube video of your Neatokart in action (include a link in your submission). Bonus points (not really) for high production value like this.

6.5 Optional Extensions

Exercise 6.6

1. Develop a parameterization that permits you to drive over the Rainbow Road in the least amount of time (don't forget about the Neatokart speed limit!).
2. Develop a parameterization that permits you to drive over the Rainbow Road at a constant linear speed.
3. Try running the Neatokart over different floor surfaces, e.g. wood, tile, carpet. Do you expect any differences? Think about the friction! Do you see any differences in the encoder data?

4. One weakness of the open-loop control approach is that it doesn't take into account that the robot doesn't (and cannot) instantaneously do what you tell it to do. One possible way to remedy this is to monitor the robot's position over time using live readings of the wheel encoders. In class, you derived a method for tracking the robot's position and orientation given measurements of its wheel velocities. We call this estimate of the robot's position its odometry. By comparing the robot's position as determined by its odometry with the desired position (given by $\vec{r}(t)$), you can try to correct your robot's motion to more faithfully follow the path. How you accomplish this exactly is up to your own creativity and analytical skills.