Robotics workshop

For students from National Formosa University @ UTSA

Workshop Outcomes

At the end of this 1.5 day workshop students will be able to:

1. Basic usage of MATLAB using command window
2. Program scripts in MATLAB
3. Create animations in MATLAB
4. Learn about forward and inverse kinematics
5. Model and simulate a planar manipulator
6. Learn about numerical integration in MATLAB
7. Model and simulate a differential drive car
8. Construct a simple video game based on differential drive car to move around obstacles.

Equipment list

You need a fairly recent version of MATLAB (say, last 5 years).

1 Basics of MATLAB

1. Basics (see MATLAB-Basics.pdf and folder 1.basics)
2. Scripts (see MATLAB-Scripts.pdf and folder 1.basics)
3. Animation (see MATLAB-Animations.pdf and folder 1.basics)

Exercise: Create an animated face with eyes, nose, and mouth. Make atleast 1 body part to move (e.g., blinking eye, mouth opens/closes. *Hint: You can modify test_animation3.m*

2 Planar manipulator / Forward and inverse kinematics

1. Forward and inverse kinematics of a two-link manipulator (see coordinate frames and manipulator notes in robotics_notes.pdf)
2. MATLAB simulation (see folder 2.manipulator)

Exercise: Draw the cardioid given by $x(\phi) = a(2\cos\phi - \cos 2\phi)$ and $y(\phi) = a(2\sin\phi - \sin 2\phi)$ where $0 \leq \phi \leq 2\pi$. Also see `https://en.wikipedia.org/wiki/Cardioid`. The center of the cardiod should be at $x = 0.5$ and $y = 0.5$. What is the biggest value of 'a' that you can achieve? *Hint: You can modify manipulator_inverse_circle.m*

3 Differential drive robot / Ordinary Differential Equations

1. Forward kinematics of a differential drive robot (see differential drive robot notes in robotics_notes.pdf)
2. MATLAB simulation (see folder 3.differential_drive)

Exercise: Generate commands for the car to generate your initials. *Hint: See lines 26 to 41 in diff_drive_main.m*

4 Project – Video game

1. Simple car program (see folder 4.project)

Run the MATLAB code *car_game.m* and play with the car controls by using the left/right and up/down controls. The heart of the code are lines 8 to 12. The function *initData* initializes the data, *initFigure* initializes the animation, thereafter, the functions *moveCar* and *refreshPlot* are run in the loop till the game ends.

Exercise: Your goal is to add more features to the game to make it a fun play. You can start adding obstacles by writing appropriate commands in the functions *initData*, *initFigure*, and *moveCar*. We have done this for the obstacle in the middle. Feel free to add more obstacles and then keep adding features. Let others play the game you have modified.

Here is an example game: https://youtu.be/9EMhdv0IAAQ (18 seconds)