COMP4033
Fuzzy Logic and Fuzzy Systems
Lab Sheet 2

Notes:

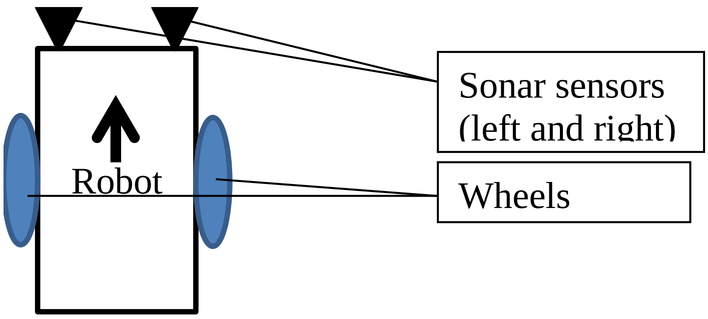
- This Lab Sheet is assessed and is worth 3% of your overall mark.
- It is due in the lab session two weeks after it has been issued (see Moodle). Please notify a member of the module team during the lab session—when you are ready to be assessed.
- The Lab Sheet is designed to help you understand concepts which are critical to the module, thus avoid seeking shortcuts to complete it. Working through the process is the aim.

Tasks:

Your goal is to design a simple fuzzy controller for the scenario below. No experience in robotics is required for this task – the focus is on the systematic design of your controller.

As part of an autonomous mobile robot soccer competition, you are asked to write a (type-1) Fuzzy Controller implementing an obstacle avoidance behaviour for a two-wheeled mobile robot. The robot is equipped with two sonar sensors pointing forward from either side of the robot as shown in the diagram below.

Note: Design the fuzzy system required for the 2-input, 1-output controller, specifically, use two fuzzy sets for each input (the domain for your sonar sensors should be [0 50]) and three fuzzy sets for the steering output (domain = [0, 100], with 0 corresponding to a hard left, and 100 to a hard right turn).



You should achieve the following:

1. As a first step, you should develop a complete fuzzy system for the controller, building on lab 1, i.e. using your own implementation and without using any fuzzy system libraries. Your controller should provide functionality for receiving two inputs (via the command prompt): the distance measured by both sonar sensors. Your program should:
   a. visualise all the fuzzy sets used
   b. visualise the inputs on the antecedent fuzzy sets
   c. visualise the output fuzzy set
   d. return the defuzzified steering output using centroid defuzzification
2. As a second step, use the Matlab fuzzy toolbox or a fuzzy system library in R, Java, or Python, to rebuild your system and compare the outputs for the following inputs: (0,0), (80, 0),

(50,50). You should be ready to discuss the results and whether they are the same, similar, or different.

3. Bonus task: generate the control surface of your own controller developed in Step 1 using your own code.

Tips:

- The aim of this lab is to give you experience in designing fuzzy systems for very different tasks—in this case a controller—one of the traditionally most popular applications of fuzzy systems. Note that while it may seem a very different problem initially, the fuzzy system design is not different to what we discuss in the lectures.
- Consider designing your system on paper or using tools such as the fuzzy toolbox in Matlab (write ‘fuzzy’ on the prompt to start it). Use your design to manually work through the operations, helping you to get a clear picture of the operation of the system—before programming it.

Marking Criteria:

- No engagement during the labs and/or no meaningful progress achieved: 0%
- Some functionality developed, with some tasks incomplete or errors/mistakes. Lacking convincing justification for the implementation presented: 2%
- Working application with all tasks completed and a convincing presentation/justification: 3%
- Bonus task: no marks (but rewarding!)

Half marks may be awarded in borderline cases.