COMPUTATIONAL INTELLIGENCE ASSIGNMENT

By

Aaron Ward

Supervisor(s):

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
B.SC IN COMPUTING AND INFORMATION TECHNOLOGY
AT
INSTITUTE OF TECHNOLOGY BLANCHARDSTOWN
DUBLIN, IRELAND
2017

Declaration

I herby certify that this material, which I now submit for assessment on the programme of study leading to the award of **B.Sc in Computing and Information Technology** in the Institute of Technology Blanchardstown, is entirely my own work except where otherwise stated, and has not been submitted for assessment for an academic purpose at this or any other academic institution other than in partial fulfillment of the requirements of that stated above.

Dated: 2017

Author: ______________________
Aaron Ward

Introduction

The purpose of this assignment is to implement a fuzzy logic controller using a feed back system. In this case, the fuzzy logic controller is a water pumping system that controls a bidirectional pump to maintain water levels of a reserve between 40% and 70%. There are two main factors that affect the water levels: The demand for water for appliances, which drains the tank of water and rainfall that may occur, that fills the tank with water.

This report will detail and explain the choices made when designing the linguistic variable and rule terms. Diagrams will be used to illustrate the linguistic variables for further explanation.

Assignment

The initial steps of implementing a fuzzy logic system are as follows:

- Crisp inputs are converted to set of fuzzy linguistic variable, linguistic terms and membership functions in what is known as fuzzification.
- Based on a set of rules, an inference is made
- Lastly, defuzzification is implemented by using the membership functions to provide crisp results.

Linguistic variables are natural language values assigned to ranges of numeric inputs or outputs. These linguistic variables can have a set of real life terms to used to define a portion of the overall variable. For example, the linguistic variable **height** may have the terms short, medium or tall etc..

Linguistic Variables

Based on the requirements of the system, linguistic variables need to be defined for the fuzzy logic system. We know that there are two input variables: ”Level” and ”Demand”. Additionally, an output variable is specified to represent the command given to the pumping system, which will be defuzzified to a final crisp output. Each of these variables should be segmented into the appropriate terms.

Level

With this variable, five terms have been used to define the ranges of the level in the water. The terms are as follows: **Level(l) = very_low, low, average, high, very_high.**
The membership functions are based on the ranges specified in the assignment document. Level of the water in the tank can range from between 0 and 100. The type of membership functions used are trapezoidal and triangle. The trapezoidal function are used on the most extreme cases such as very_low and very_high and triangle are used on the low, average and high. See figure 1 for graphical representation.

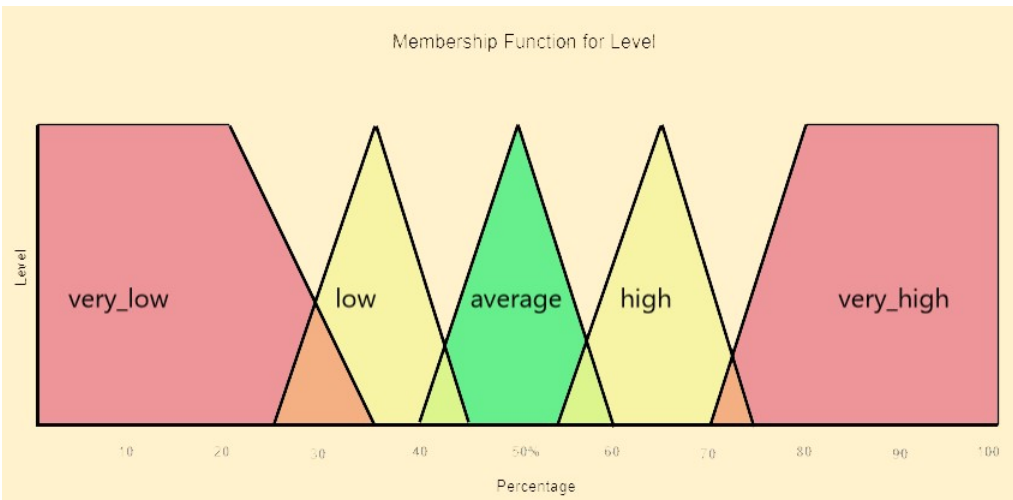

Figure 1: Membership functions for Level

Demand

Like the level variable, five linguistic terms were made. The terms are as follows: **demand(d) = very_low, low, middle, high, very_high**. The demand variable has a range of -1 and 1.5. The same structure of two trapezoids and three triangle functions were used.

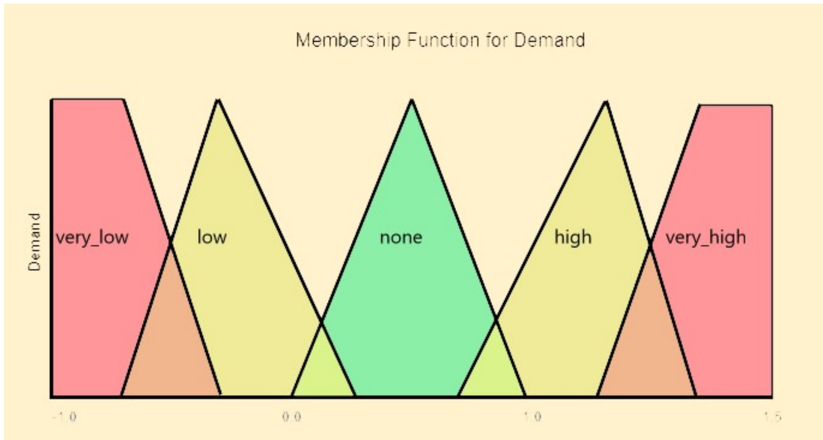

Figure 2: Membership functions for demand

Rule Terms

in order to establish an output variable a set of rules must be defined control the value of the output variable. The three terms used for the output variable are as follows: **Command = pump_in, no_pump, pump_pump out**. The following diagram shows the membership functions of the output variable command.

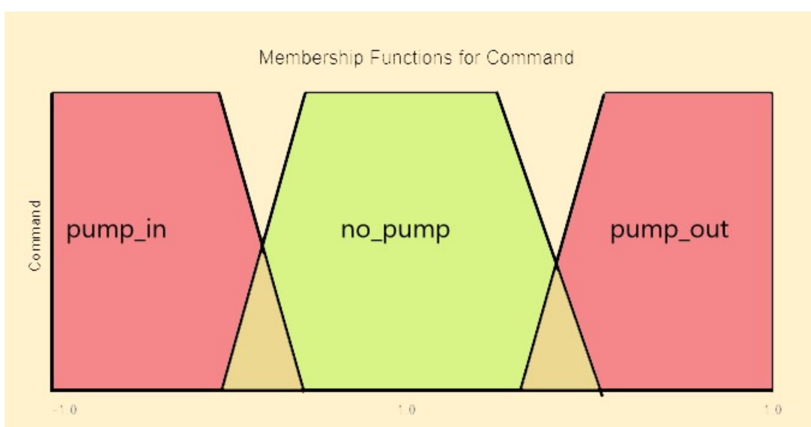

Figure 3: Membership functions for command

A rule matrix will be utilized to illustrate the number of possible rules in the system. The top row signifies the demand variable with the terms it has been provided. Also, the first column displays the Level linguistic variable and it's terms. The intersection terms in the cells contain the command output variable that will be used to control the system.

	Demand				
Level	very_low	low	middle	high	very_high
very_low	pump_in	pump_in	pump_in	pump_in	no_pump
low	pump_in	pump_in	no_pump	pump_out	no_pump
average	no_pump	no_pump	no_pump	pump_out	pump_out
high	no_pump	pump_out	pump_out	pump_out	pump_out
very_high	pump_out	pump_out	pump_out	pump_out	pump_out

Following the implementation of the rule matrix, a set of rules need to be defined using an *IF-THEN* format.

IF (level is very_low) and (demand is very_low or low or middle or high) THEN command is pump_in
IF (level is very_low) and (demand is very_high) THEN command is no_pump
IF (level is low) and (demand is very_low or low) THEN command is pump_in
IF (level is low) and (demand is high or very_high) THEN command is pump_out
IF (level is average) and (demand is very_low or low or middle) THEN command is no_pump
IF (level is average) and (demand is high or very_high) THEN command is pump_out
IF (level is high) and (demand is very_low) THEN command is no_pump
IF (level is high) and (demand is low or middle or high or very_high) THEN command is pump_out
IF (level is very_high) and (demand is very_low or low or middle THEN command is pump_out
IF (level is very_high) and (demand is high or very_high) THEN command is pump_out

After some experimentation with the rules, it appeared that the more rules that were applied, and the more complex the rules were, the worse the system performed. Therefore only four of the rules where kept in the final implementation. Using the check boxes, the controller maintains the level of water in the tank between the 40% and 70% requirement.