MA410 Practical: Fuzzy Logic - Washing Machine Example

The washing machine is one of the first devices to use fuzzy logic.

The idea is to identify the appropriate time needed to wash the load, given the dirtiness and the volume of the load.

Rules

The following table shows the rule base for the washing time problem:

load volume \ *load dirtiness*	***vd***	***md***	***ld***	***nd***
fl	vlot	vlot	lot	lit
ml	vlot	mt	mt	lit
ll	lot	lot	lit	lit

where

vd: very dirty
md: medium dirty
ld: lightly dirty
nd: not dirty

fl: full load
ml: medium load
ll: low load

vlot: very long time
lot: long time
mt: medium time
lit: little time

The rule table consists of 12 rules.
For example, the entry in the second row and the third column of the table specifies the rule:

if load volume ***is*** medium load
and load dirtiness ***is*** lightly dirty
then washing time ***is*** medium time

The fuzzy sets for load dirtiness, volume and washing time respectively are based on the linear fit function $ax + b$, and are defined based on the following tables:

Load Dirtiness (LD)		***nd***	***ld***	***md***	***vd***
	$\mu(LD) = 0$ if $LD \leq$		1	2	6
	$\mu(LD) = 1$ if $LD =$	0	3	5	10
	$\mu(LD) = 0$ if $LD \geq$	2	5	7	

Load Volume (LV)		***ll***	***ml***	***fl***
	$\mu(LV) = 0$ if $LV \leq$		2	6
	$\mu(LV) = 1$ if $LV =$	0	5	10
	$\mu(LV) = 0$ if $LV \geq$	4	8	

Washing Time (WT)		***lit***	***mt***	***lot***	***vlot***
	$\mu(WT) = 0$ if $WT \leq$		20	50	90
	$\mu(WT) = 1$ if $WT =$	10	50	80	120
	$\mu(WT) = 0$ if $WT \geq$	30	60	100	

Tasks

1. Based on the fuzzy set tables above, draw three individual graphs for load dirtiness (LD), load volume (LV) and washing time (WT).

2. (a) Name and explain the four steps of fuzzy inference based on the Mamdani method.

   (b) Use this method and the rules given in order to compute the washing time WT necessary to wash the load for:

   I. $LD = 9, LV = 3$

   II. $LD = 3, LV = 7$

   when (i) clipping is used and (ii) scaling is used for aggregate sets.

3. How would you create the output sets WT for Sugeno's method. Draw the graph comprising the fuzzy sets for WT and explain your reasoning.

4. Calculate the output given the same inputs above if Sugeno's method was used instead of Mamdani inference.

5. (a) Download the files `washing.pl` and `fuzzyfns.pl` from the course webpage.

   (b) Add in the following predicates to reflect the membership functions given:

   i. `medium_dirty(MuD, D)` : `MuD` = degree of membership associated with crisp input `D`.

   ii. `full_load(MuV, V)` : `MuV` = degree of membership associated with crisp input `V`.

   iii. `long_time(MuT, T)` : `MuT` = degree of membership associated with crisp input `T`.

   iv. Add in the missing rules from the file (`if X then Y`, etc.)

   v. What prolog function allows you to enter in rules in such a natural language way, i.e. in the form "`if X then Y.`"?

   vi. Use the incredibly long-named predicate:

   `fuzzify_ruleevaluate_aggregate_defuzzify(INPS, CorS, START, END, STEP, COG)`

   where

   - `INPS` : of the form `[ [linguistic variable 1, crisp value 1], [linguistic variable 2, crisp value 2], ...]` ,

     e.g. `[[load_volume, 1], [load_dirtiness, 2]]`

   - `CorS` : argument that takes value `clip` or `scale`.
   - `START` : point where to start summing riemann integral in order to calculate centre of gravity
   - `END` : point where to end summing riemann integral in order to calculate centre of gravity
   - `STEP` : size of steps (width of blocks) for riemann integration
   - `COG` : centre of gravity (what we're looking for)

   vii. Solve 2(b) I. and II. (i) & (ii) using this function.

   (c) i. Create your own fuzzy-logic based system using the file `fuzzyfns.pl`.

   ii. Calculate various centres of gravity for different crisp values.

   iii. E-mail me your fuzzy-logic idea, the code used and some sample inputs & outputs.