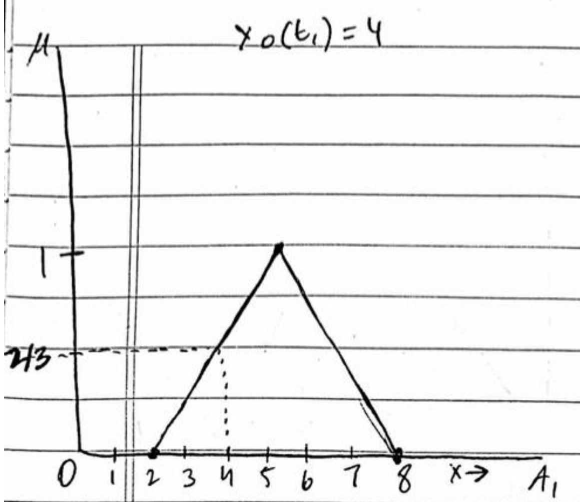
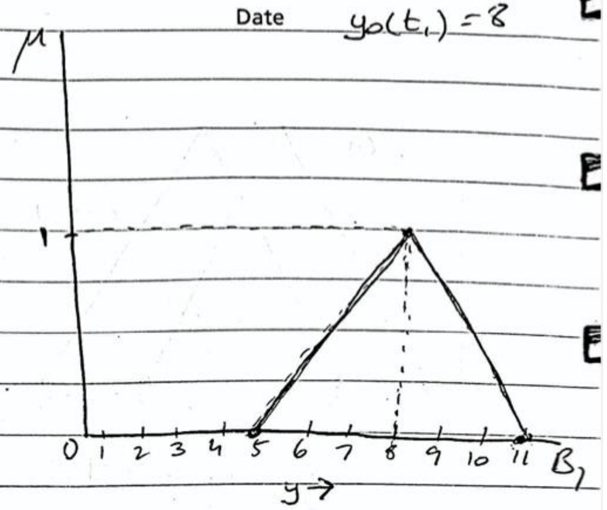
Question 1

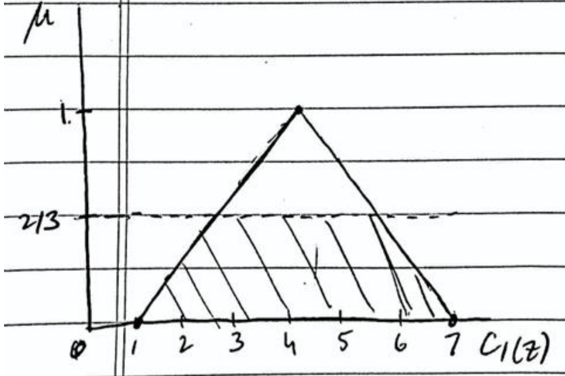


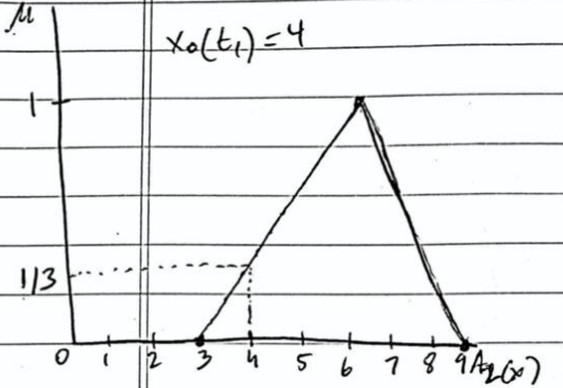


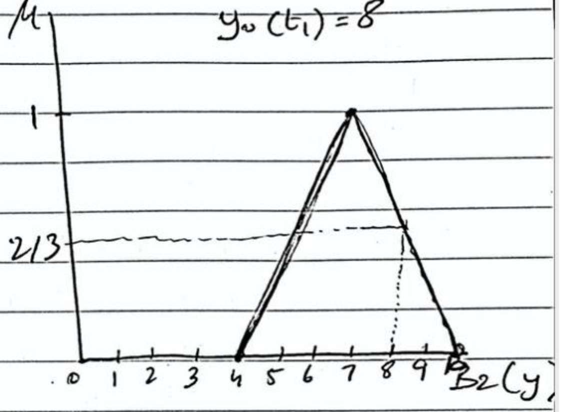
$$\mu_{A1} = \begin{cases} \frac{2}{3} & 2 \leq x \leq 5 \\ \frac{4}{3} & 5 \leq x \leq 8 \end{cases}$$

$$\mu_{B1} = \begin{cases} 1 & 5 \leq y \leq 8 \\ 1 & 8 \leq y \leq 11 \end{cases}$$

Take min of 2/3 or 1, min is $\frac{2}{3}$. Project this onto $C_1(z)$



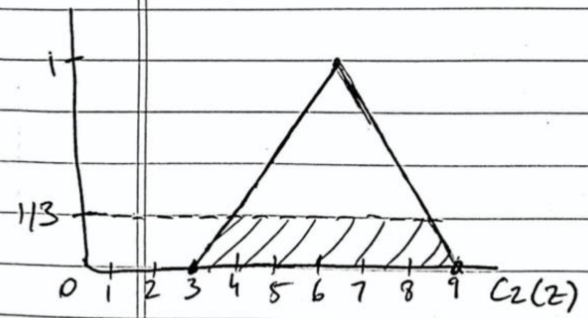


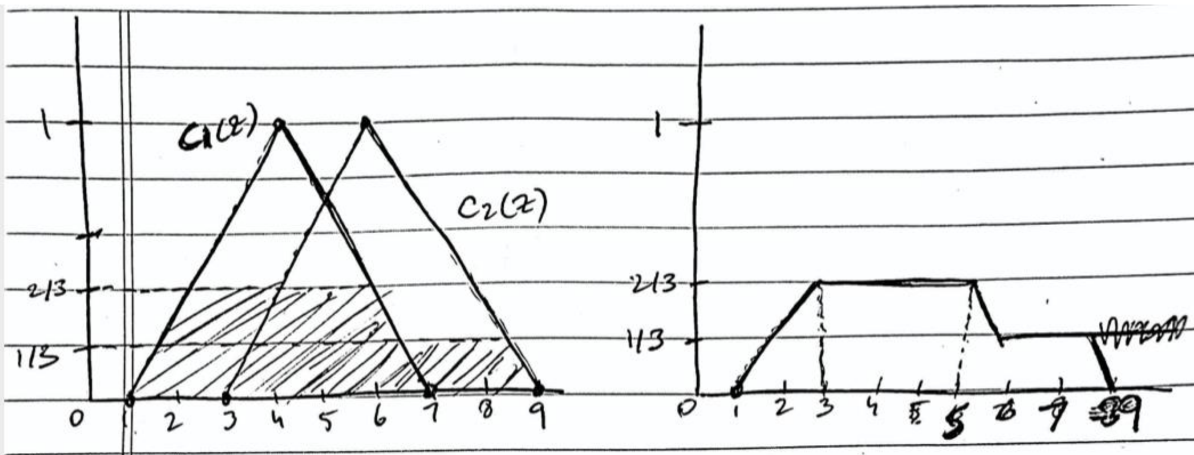


~~Taking~~

$$\mu_{A_2} = \begin{cases} \frac{1}{3} & 3 \le x \le 6 \\ \frac{5}{3} & 6 < x \le 9 \end{cases} \qquad \mu_{B_2} = \begin{cases} \frac{4}{3} & 4 \le y \le 7 \\ \frac{2}{3} & 7 < y \le 10 \end{cases}$$

Take min of $\frac{2}{3}$ or $\frac{1}{3}$, min is $\frac{1}{3}$. Project this on $C_2(z)$





Using MOM method to defuzzify:
2 max values are located at 3 and 5

$$t_1 = \frac{3+5}{2}$$

$$= 4$$

Using LOM method, the output at $t_1 = 5$

Question 2

2) Fuzzy set about the nominal x_{56} :

$$\underset{\sim}{A} = \left\{ \frac{.2}{1.7} + \frac{.4}{1.8} + \frac{.6}{1.9} + \frac{.8}{2.0} + \frac{.6}{2.1} + \frac{.4}{2.2} + \frac{.2}{2.3} \right\}$$

Accelerator bias :

$$\underset{\sim}{B} = \left\{ \frac{.1}{.25} + \frac{.4}{.27} + \frac{.9}{.3} + \frac{.4}{.33} + \frac{.1}{.35} \right\}$$

a. classical implication operator $\mu_R = \max[\min(\mu_A, \mu_B), (1-\mu_A)]$

Find relation R for IF $\underset{\sim}{A}$ Then $\underset{\sim}{B}$

$\min(\mu_A, \mu_B)$ can be obtained using cartesian product :

$$\begin{array}{cc|ccccc|}
 & & & & \mu_B & & \\
 & & 0.1 & 0.4 & 0.9 & 0.4 & 0.1 \\
\hline
 & 0.2 & 0.1 & 0.2 & 0.2 & 0.2 & 0.1 \\
 & 0.4 & 0.1 & 0.4 & 0.4 & 0.4 & 0.1 \\
 & 0.6 & 0.1 & 0.4 & 0.6 & 0.4 & 0.1 \\
\mu_A & 0.8 & 0.1 & 0.4 & 0.8 & 0.4 & 0.1 \\
 & 0.6 & 0.1 & 0.4 & 0.6 & 0.4 & 0.1 \\
 & 0.4 & 0.1 & 0.4 & 0.4 & 0.4 & 0.1 \\
 & 0.2 & 0.1 & 0.2 & 0.2 & 0.2 & 0.1 \\
\hline
\end{array}$$

$A' = (1 - \mu_A)$

$$A' = \left\{ \frac{1-0.2}{1.7} + \frac{1-0.4}{1.8} + \frac{1-0.6}{1.9} + \frac{1-0.8}{2.0} + \frac{1-0.6}{2.1} + \frac{1-0.4}{2.2} + \frac{1-0.2}{2.3} \right\}$$

$$= \left\{ \frac{0.8}{1.7} + \frac{0.6}{1.8} + \frac{0.4}{1.9} + \frac{0.2}{2.0} + \frac{0.4}{2.1} + \frac{0.6}{2.2} + \frac{0.8}{2.3} \right\}$$

To convert A' to a 2D matrix, we obtain its cylindrical extension:

$$E[A'] = \begin{bmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix}$$

$$\therefore R \overset{\text{max cols}}{\underset{\wedge}{=}} \left[E[A'] \vee \min(\mu_A, \mu_B) \right]$$

max columns
$$\begin{bmatrix} \begin{bmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix} & \begin{bmatrix} 0.1 & 0.2 & 0.2 & 0.2 & 0.1 \\ 0.1 & 0.4 & 0.4 & 0.4 & 0.1 \\ 0.1 & 0.4 & 0.6 & 0.4 & 0.1 \\ 0.1 & 0.4 & 0.8 & 0.4 & 0.1 \\ 0.1 & 0.4 & 0.6 & 0.4 & 0.1 \\ 0.1 & 0.4 & 0.4 & 0.4 & 0.1 \\ 0.1 & 0.2 & 0.2 & 0.2 & 0.1 \end{bmatrix} \end{bmatrix}$$

$$R = \begin{bmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.8 & 0.4 & 0.2 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix}$$

ii.

$$A' = \left\{ \frac{0}{1.7} + \frac{.5}{1.8} + \frac{.7}{1.9} + \frac{.95}{2.0} + \frac{.7}{2.1} + \frac{.5}{2.2} + \frac{0}{2.3} \right\}$$

Max-min composition, $T = A' \circ R$

$$T = \underset{\text{rows}}{\max} \left(\underset{\text{columns}}{\min} \left[\begin{bmatrix} 0.0 \\ 0.5 \\ 0.7 \\ 0.95 \\ 0.7 \\ 0.5 \\ 0.0 \end{bmatrix} \begin{bmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.8 & 0.4 & 0.2 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix} \right] \right)$$

$$T = \underset{\text{rows}}{\text{Max}} \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.8 & 0.4 & 0.2 \\ 0.4 & 0.4 & 0.6 & 0.5 & 0.4 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$= [0.5 \quad 0.5 \quad 0.8 \quad 0.5 \quad 0.5]$$

Using min-max composition, $T = [0.5 \; 0.5 \; 0.8 \; 0.5 \; 0.5]$

(ii). Max-product composition, $T = A' \circ R$

$$T = \underset{\text{rows}}{\text{max}} \left([0.0 \; 0.5 \; 0.7 \; 0.95 \; 0.7 \; 0.5 \; 0.0] \begin{bmatrix} 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.2 & 0.4 & 0.8 & 0.4 & 0.2 \\ 0.4 & 0.4 & 0.6 & 0.4 & 0.4 \\ 0.6 & 0.6 & 0.6 & 0.6 & 0.6 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix} \right.$$

$$T = \underset{\text{rows}}{\text{Max}} \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.28 & 0.28 & 0.42 & 0.28 & 0.28 \\ 0.19 & 0.38 & 0.76 & 0.38 & 0.19 \\ 0.28 & 0.28 & 0.42 & 0.28 & 0.28 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$T = \begin{bmatrix} 0.3 & 0.38 & 0.76 & 0.38 & 0.3 \end{bmatrix}$$

using max-product composition,

$$T = \begin{bmatrix} 0.3 & 0.38 & 0.76 & 0.38 & 0.3 \end{bmatrix}.$$

Question 3

This part looks at powerful optimization techniques based on evolutionary computation which uses similar technique of natural selection and the way genetics work.

J.Holland introduced Genetic algorithms in the 1960's with the original intention to study adaptive systems. Genetic algorithms today are mainly used as a search technique to find approximate solutions to different kinds of problems. In intelligent Control (IC) they are mainly used as an optimization technique to find the minimums or maximums of complex equations, or quasi-optimal solutions in a short period of time.

Genetic Programming in 1985 was proposed by N. Crammer which is another kind of evolutionary computation. It is another kind of evolutionary computation algorithm with string bases in genetic algorithm (GA). The difference is that in GA strings of bits representing chromosomes are evolved whereas in genetic programming the whole structure of a computer program is evolved by the algorithm. Due to this structure genetic programming can manage problems that are harder to manipulate by genetic algorithms. Genetic programming has been used in Intelligent Control optimize the sets of rules on fuzzy and neuro-fuzzy controllers.

Genetic algorithms operate its chromosomes in linear structure which be vectors and integer strings, generally one-dimensional arrays. While on the other hand, genetic programming uses the tree shaped chromosomes which are not linear. In generic algorithm the size of chromosomes is fixed however, whereas in Genetic Programming the tree may change in depth and width. It can be a tree with 100 nodes or just a tree with 2 nodes.

APPLICATIONS

Genetic programming has been successfully used as an automatic programming tool, machine learning tool and automatic problem-solving engine. Genetic Programming is especially useful in the domains where the exact form of the solution is not known in advance, or an approximate solution is acceptable. Some applications of genetic programming are curve fitting, data modelling, symbolic regression, feature selection, classification.

As for Genetic Algorithm, using the case of feature selection, how do you select features that are important in prediction of the target variable? You always look at the feature importance of some model, and then manually decide the threshold, and select the features which have importance above that threshold, one of the most advanced algorithms for feature selection is genetic algorithm. Engineering Design has relied heavily on computer modelling and simulation to make design cycle process fast and economical and the Genetic algorithm is at the heart of the optimization process. Other areas include traffic and shipment routing and robotics to name a few.

PSEUDOCODE / STEPS

Genetic Algorithm

Genetic Algorithm starts with an initial population (which may be generated at random or seeded by other heuristics), select parents from this population for mating. Apply crossover and mutation operators on the parents to generate new off-springs. These off-springs replace the existing individuals in the population and the process repeats.

GA() steps:

- initialize population
- find fitness of population
- while (termination criteria is reached) do
  - parent selection
  - crossover with probability pc
  - mutation with probability pm
  - decode and fitness calculation
  - survivor selection
  - find best
- return best

Genetic Programming

Genetic programming (GP) creates random programs and assigns them a task of solving a problem. The fitness function describes how well they perform their task. Crossover "breeds" two programs together (swaps their code). Mutation introduces random changes in some programs.

GP() steps:

- Generate an initial population of random computer programs.
- Execute each program in the population and assign it a fitness value according to how well it solves the problem.
- Create a new population of computer programs.
- Copy the best existing programs.

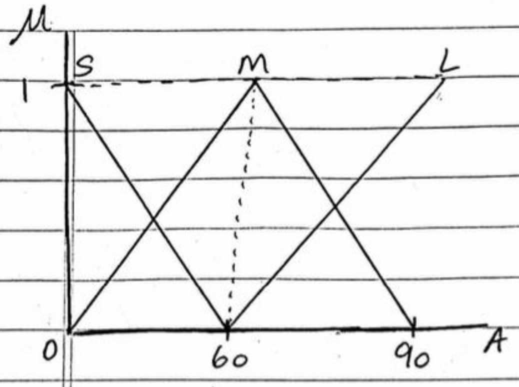
- Create new computer programs by mutation.
- Create new computer programs by crossover (sexual reproduction).

Question 4

Part 1

The membership functions for angle can be seen below.

Date



To find membership functions, use formula $\frac{y_2-y_1}{x_2-x_1} = \frac{y-y_1}{x-x_1}$ for each state

Small: [0,60]

Med: [0,60], [60,90]

Large: [60,90]

$$\mu_S = \frac{0-1}{60-0} = \frac{y-1}{x-0}$$

$$\frac{-x}{60} = y-1$$

$$\boxed{\mu_S = \frac{-x+60}{60}}$$

$$\mu_L: \frac{1-0}{90-60} = \frac{y-0}{x-60}$$

$$\frac{1}{30} = \frac{y}{x-60}$$

$$\boxed{\mu_L = \frac{x-60}{30}}$$

$$\mu_m = \frac{1-0}{60-0} = \frac{y-0}{x-0}$$

$$\boxed{\mu_m = \frac{x}{60} \quad [0,60]}$$

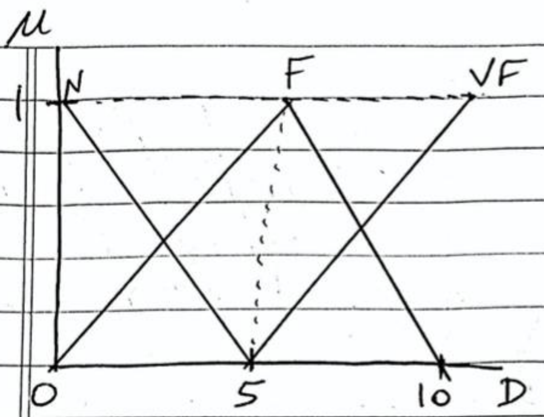
$$\mu_m = \frac{0-1}{90-60} = \frac{y-1}{x-60}$$

$$\frac{-1}{30} = \frac{y-1}{x-60}$$

$$\boxed{\mu_m = \frac{-x+90}{30} \quad [60,90]}$$

The membership functions for distance can be seen below.

Date



$$\mu_N: \frac{0-1}{5-0} = \frac{y-1}{x-0}$$

$$\frac{-1}{5} = \frac{y-1}{x}$$

$$\boxed{\mu_N = \frac{-x+5}{5}}$$

$$\mu_F: \frac{1-0}{5-0} = \frac{y-0}{x-0}$$

$$\frac{1}{5} = \frac{y}{x}$$

$$\boxed{\mu_F = \frac{x}{5} \quad [0,5]}$$

$$\mu_{VF}: \frac{1-0}{10-5} = \frac{y-0}{x-5}$$

$$\frac{1}{5} = \frac{y}{x-5}$$

$$\boxed{\mu_{VF} = \frac{x-5}{5}}$$

$$\mu_F: \frac{0-1}{10-5} = \frac{y-1}{x-5}$$

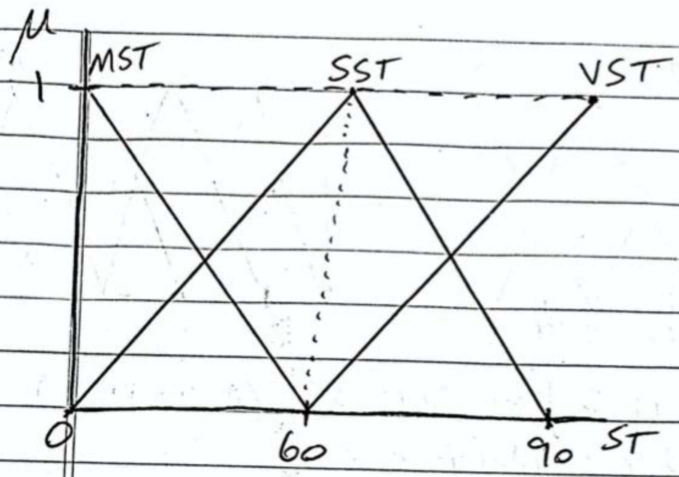
$$\frac{-1}{5} = \frac{y-1}{x-5}$$

$$\boxed{\mu_F = \frac{-x+10}{5} \quad [5,10]}$$

N: [0,5]
F: [0,5], [5,10]
VF: [5,10]

The membership functions for steering can be seen below.

Date



MST: [0, 60]
SST: [0, 60], [60, 90]
VST: [60, 90]

$$\mu_{MST}: \frac{0-1}{60} = \frac{y-1}{x}$$

$$\boxed{\mu_{MST} = \frac{-x+60}{60}}$$

$$\mu_{VST}: \frac{1-0}{30} = \frac{y-0}{x-60}$$

$$\boxed{\mu_{VST} = \frac{x-60}{30}}$$

$$\mu_{SST}: \frac{1-0}{60} = \frac{y}{x}$$

$$\mu_{SST} = \boxed{\frac{x}{60}\ [0,60]}$$

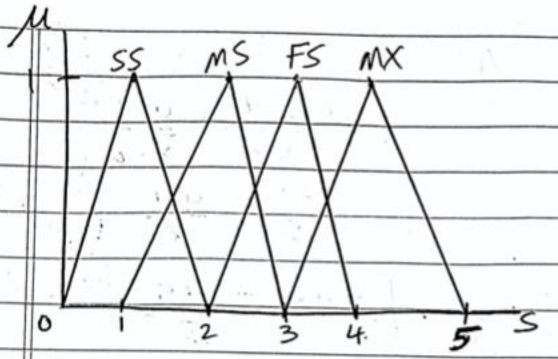
$$\mu_{SST}: \frac{0-1}{30} = \frac{y-1}{x-60}$$

$$\frac{-x+60}{30} = y-1$$

$$\mu_{SST} = \boxed{\frac{-x+90}{30}\ [60,90]}$$

The membership function for speed can be seen below.

Date



0-2: SS

1-3: MS

2-4: FS

3-5: MX

μ_{SS}: $\frac{1-0}{1-0} = \frac{y-0}{x-0}$

$1 = \frac{y}{x}$

$\boxed{\mu_{SS} = x \quad [0,1]}$

μ_{SS}: $\frac{0-1}{2-1} = \frac{y-1}{x-1}$

$-1 = \frac{y-1}{x-1}$

$-x+1 = y-1$

$\boxed{\mu_{SS} = -x+2 \quad [1,2]}$

μ_{MS}: $\frac{1-0}{2-1} = \frac{y-0}{x-1}$

$\frac{x-1}{1} = y$

$\boxed{\mu_{MS} = x-1 \quad [1,2]}$

μ_{MS}: $\frac{0-1}{1} = \frac{y-1}{x-2}$

$-x+2 = y-1$

$\boxed{\mu_{MS} = -x+3 \quad [2,3]}$

μ_{FS}: $\frac{1-0}{1} = \frac{y-0}{x-2}$

$x-2 = y$

$\boxed{\mu_{FS} = x-2 \quad [2,3]}$

μ_{FS}: $\frac{0-1}{1} = \frac{y-1}{x-3}$

$-x+3 = y-1$

$\boxed{\mu_{FS} = -x+4 \quad [3,4]}$

μ_{MX}: $\frac{1-0}{1} = \frac{y-0}{x-3}$

$\boxed{\mu_{MX} = x-3 \quad [3,4]}$

μ_{MX}: $\frac{0-1}{1} = \frac{y-1}{x-4}$

$-x+4 = y-1$

$\boxed{\mu_{MX} = -x+5 \quad [4,5]}$

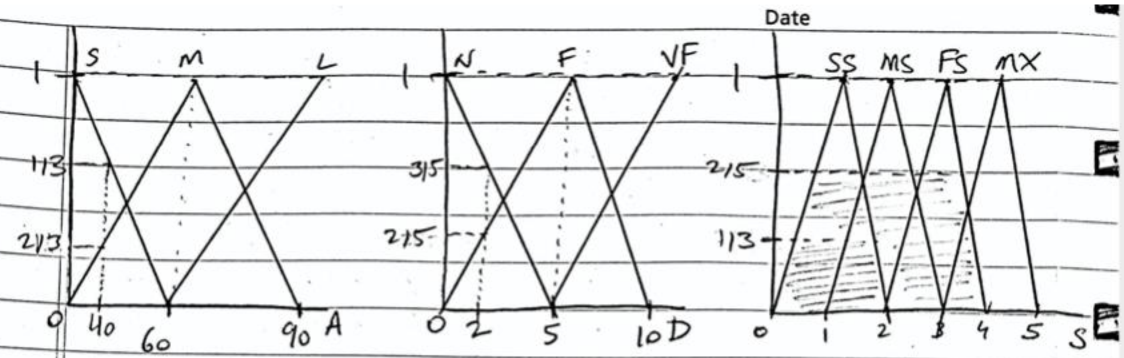
Page

Mamdani Inference System is chosen because the rules created for this problem are not a function of the inputs. Sugeno systems are best used where the consequent in the rule is a function of the antecedent [1]. In this case, the consequent is usually a crisp function while the antecedent is a fuzzy set [1]. Since the output membership function is defined in this problem, it was best to use a Mamdani system. Moreover, Mamdani systems are best used when the rules need to be created based on human knowledge [2]. In this case, the fuzzy quantities needed to be interpreted to make the rule base. The Mean of Maximum method is used to defuzzify the output.

Rule Base:

1. If D is N and A is S then S is SS. This rule indicates that an obstacle is nearby and that the speed needs to be reduced.
2. If D is F and A is M then S is FS. This rule indicates that an obstacle is farther away and that speed can be increased.
3. If D is N and A is S then ST is VST. This rule indicates that an obstacle is near and that a very sharp turn needs to be made in order to avoid the obstacle.
4. If D is F and A is M then ST is SST. This rule indicates that an obstacle is farther away and that a sharp turn to a lesser degree can be made.

We implemented an inference system by hand using sample input values for angle and distance. The output for speed can be seen below. Using sample inputs of Angle as 40 and Distance as 2 and the Mamdani Inference System, the membership value for angle S is 1/3 and the membership value for distance N is 3/5. Take the minimum of these two values and project it on the S plot so that it covers region SS to get the output. The minimum is 1/3. A sample demonstration of this system can be seen below. Please note the graphs are not drawn to scale.



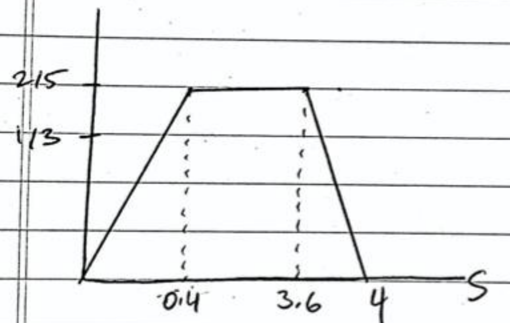
A=40

D=2

If D is N and A is S then S is SS

If D is F and A is M then S is FS

Output



To defuzzify, use MoM method:

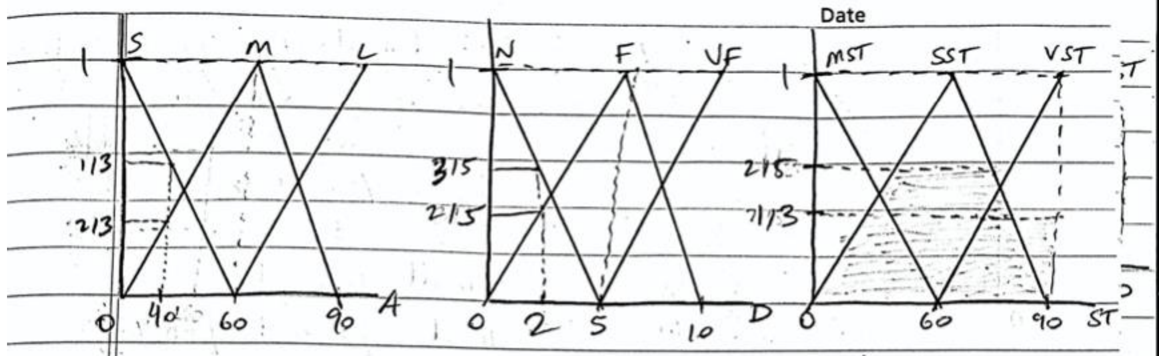
$$S = \frac{0.4+3.6}{2}$$

S = 2 at A=40, D=2

If D is N and A is S then S is SS
If D is F and A is M then S is FS

First Rule:
Using sample inputs of Angle as 40 and Distance as 2 and the Mamdani Inference System, the membership value for angle S is 1/3 and the membership value for distance N is 3/5. Take the minimum of these two values and project it on the S plot so that it covers region SS to get the output. The minimum is 1/3.

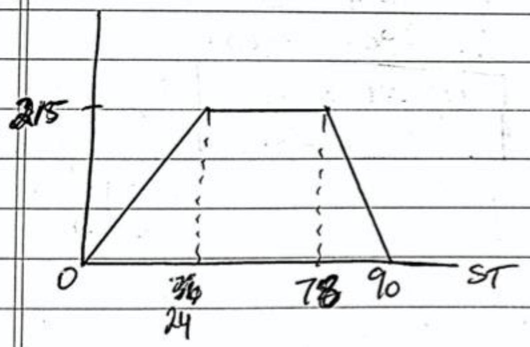
Second Rule:
Similarly, the value for angle M is 2/3 and the value for distance F is 2/5. Take the minimum of these and project it on the S plot so that it covers the FS region to get the output. The minimum is 2/5.
Similarly, the output for steering can be seen below.



A = 40
D = 2

If D is N and A is S then ST is VST
If D is F and A is M then ST is SST

Output



To defuzzify, use MoM method

$ST = \frac{24+78}{2}$

= 51 at A = 40, D = 2

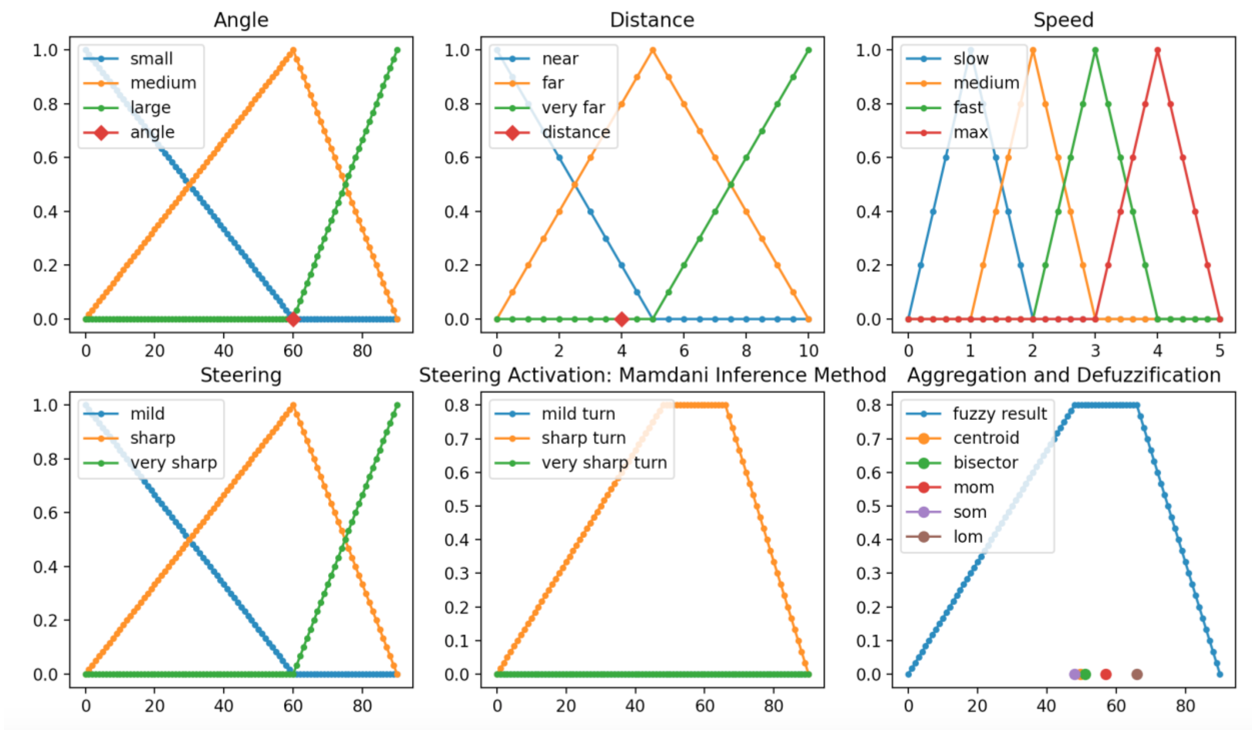
If D is N and A is S then ST is VST

If D is F and A is M then ST is SST

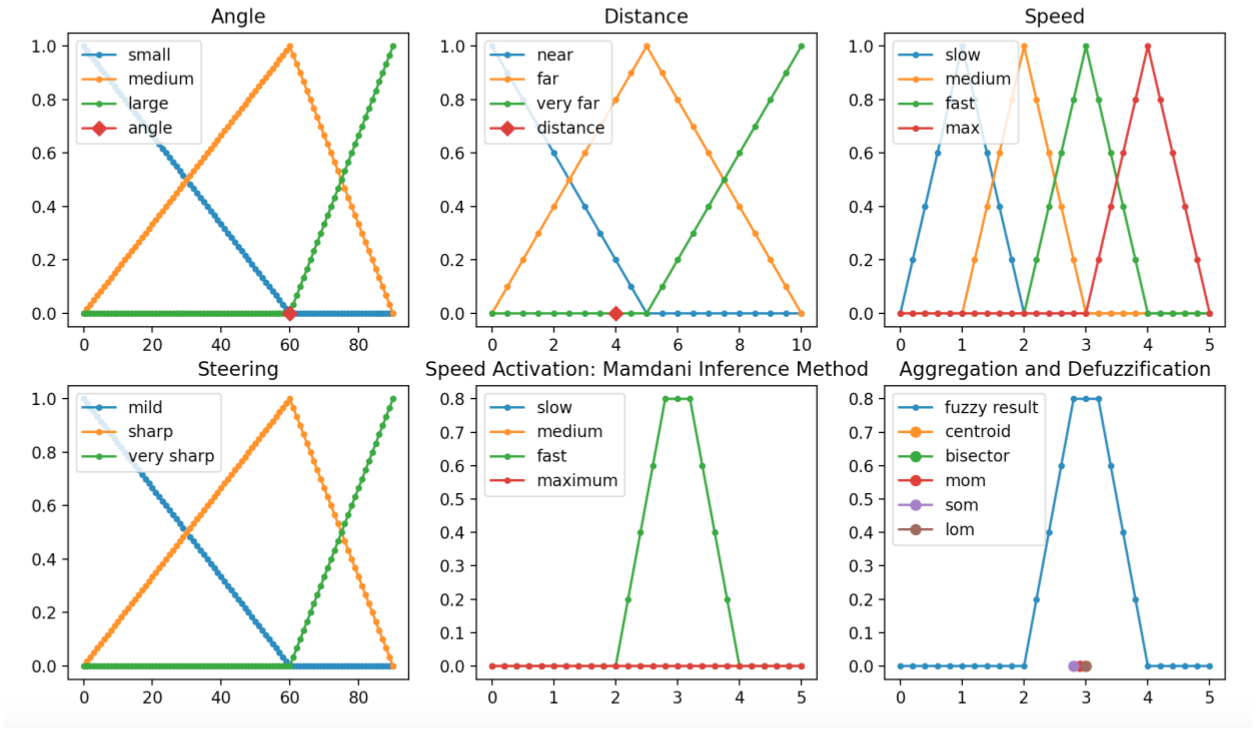
First Rule:
Using sample inputs of Angle as 40 and Distance as 2 and the Mamdani Inference System, the membership value for angle S is 1/3 and the membership value for distance N is 3/5. Take the minimum of these two values and project it on the ST plot so that it covers region VST to get the output. The minimum is 1/3.

Second Rule:
Similarly, the value for angle M is 2/3 and the value for distance F is 2/5. Take the minimum of these and project it on the ST plot so that it covers the SST region to get the output. The minimum is 2/5.

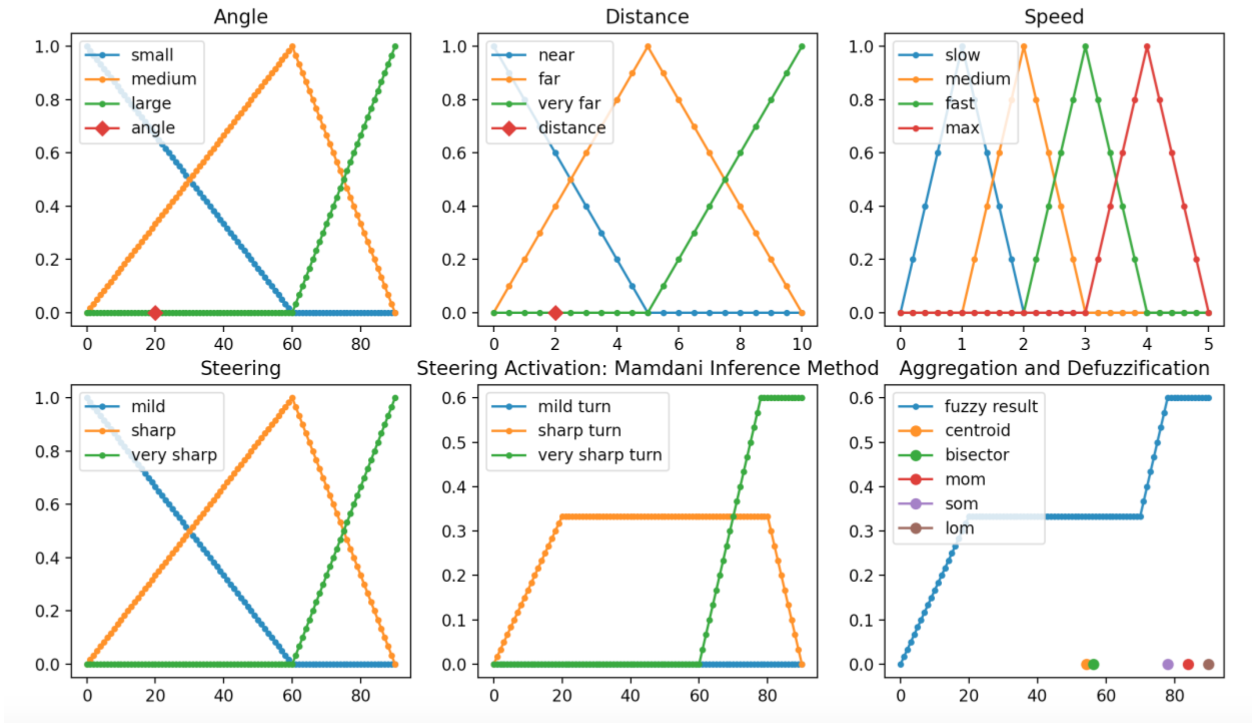
Moreover, a Python implementation can be seen below. The first implementation uses Angle as 60 and Distance as 4. As visible, the membership functions along with the output plots can be seen. The output for steering can be seen. Using the MOM method to defuzzify, the output for steering is 57.



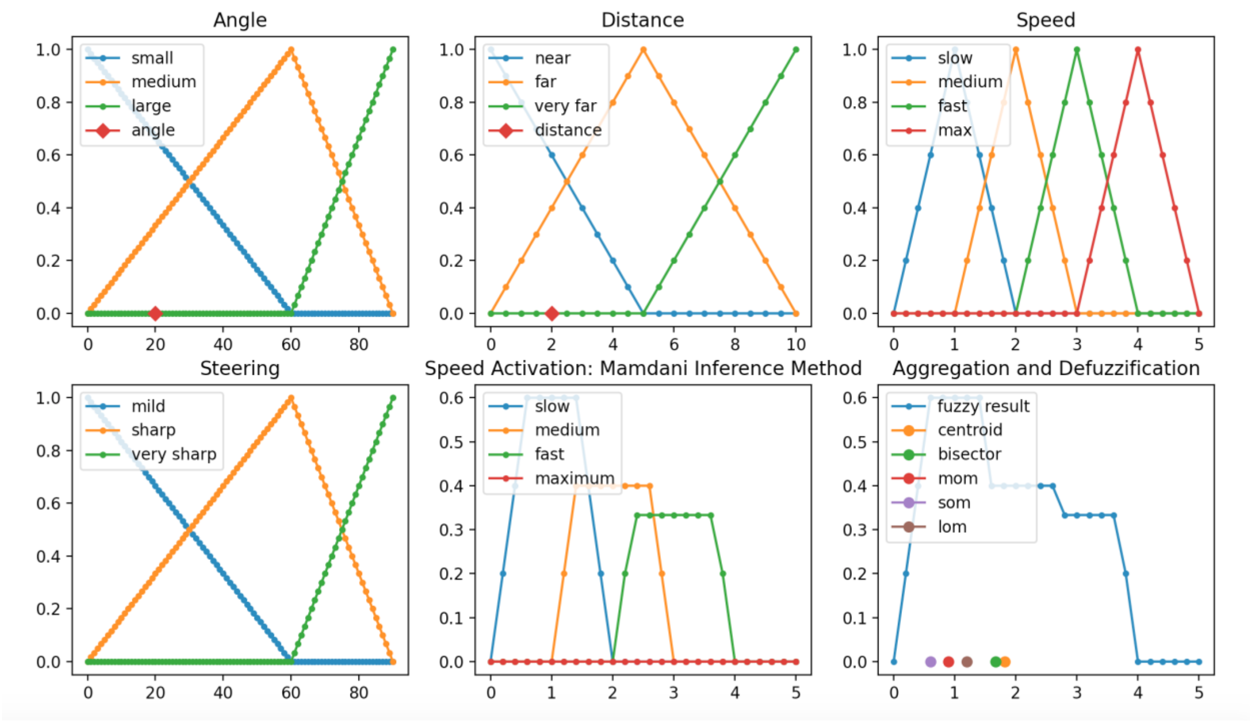
The speed output can be seen below. Using the MOM method, the output is 2.9.



The second implementation uses angle as 20 and distance as 2. The output for steering can be seen below. Using the MOM method, the output is 84.



The output for speed can be seen below. Using the MOM method to defuzzify, the output is 0.9.



References

1.Comparison Between Mamdani and Sugeno Fuzzy Inference System. (2020, May 26). Retrieved July 31, 2021, from https://www.geeksforgeeks.org/comparison-between-mamdani-and-sugeno-fuzzy-inference-system/

2. Select a Web SiMamdani and Sugeno Fuzzy Inference Systemste. (n.d.). Retrieved July 31, 2021, from https://www.mathworks.com/help/fuzzy/types-of-fuzzy-inference-systems.html