Fuzzy Logic Control

Mamdani and Takagi-Sugeno Controllers

MSc IS/ISR

Introduction I

- The most successful area of fuzzy set theory is *fuzzy control.*
- While classical controllers are based on mathematical models of the controlled processes, fuzzy controllers are based on knowledge elicited from human operators formulated in terms of a set of *fuzzy control rules (*fuzzy IF-THEN rules)

R: IF (fuzzy criteria) THEN (fuzzy conclusion)

- Such a rule can be seen as a causal relation between measurements and control values of the process.

Introduction II

- Both antecedents and consequents can involve several linguistic variables. If this is the case, the system is called a multi-input-multi-output (MIMO) fuzzy system. Such systems have several input variables and output variables.
- Multi-input-single-output (MISO) systems has several input variables but only one output variable. These are very common and the focus of our session. An example of a MISO system is as follows.

$$R1: IF\ x\ is\ A_1\ AND\ y\ is\ B_1\ THEN\ z\ is\ C_1$$

$$R2: IF\ x\ is\ A_2\ AND\ y\ is\ B_2\ THEN\ z\ is\ C_2$$

$$\ldots$$

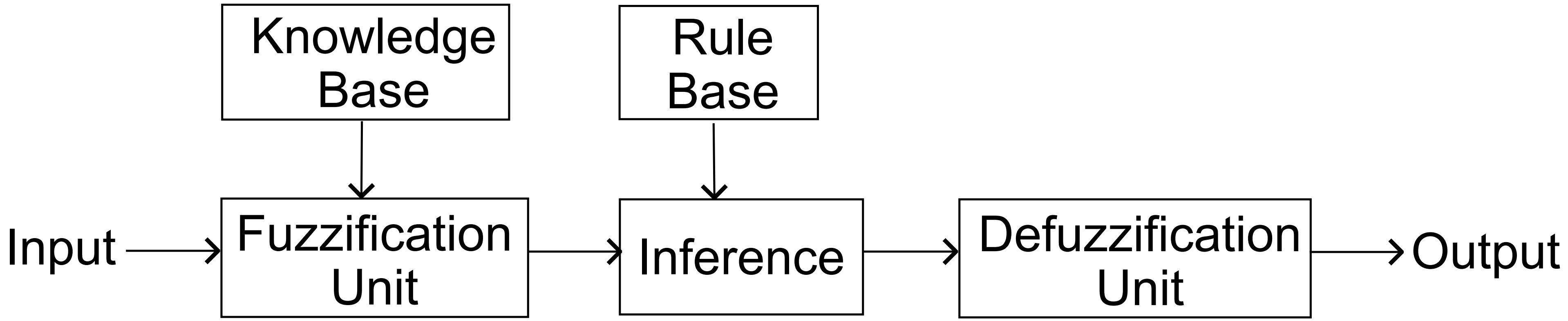
$$Rn: IF\ x\ is\ A_n\ AND\ y\ is\ B_n\ THEN\ z\ is\ C_n.$$

- Here x and y are input variables and z is the output variable, while

A_i, B_i and C_i (i=1,…,n) are linguistic variables.

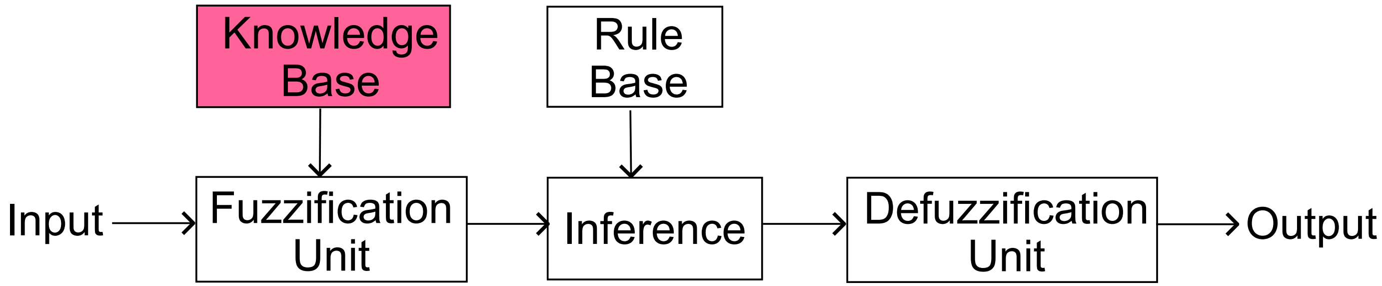
Overview of a Mamdani Fuzzy Controller

- A fuzzy controller consists respectively of fuzzification, rule base, inference mechanism and defuzzification.



Knowledge-Base

The Knowledge-base of a FLC aims to provide the information for the proprer functioning of the fuzzification module, the rule base, and the defuzzifcation module.

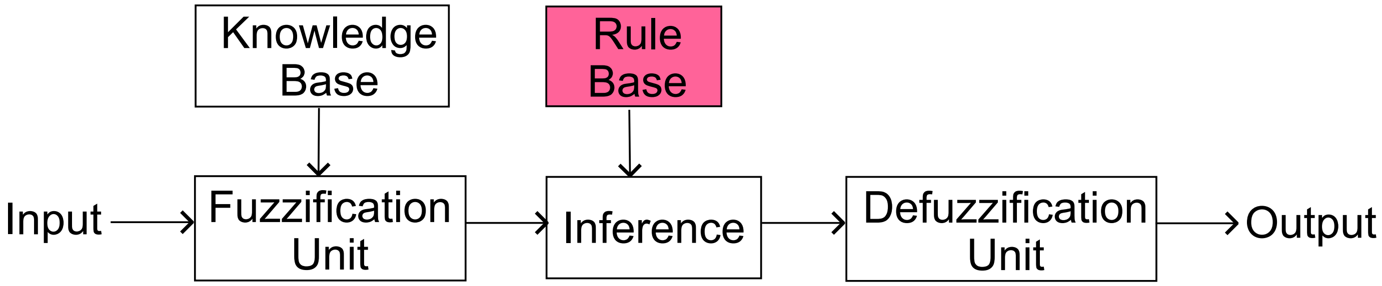


This information includes:

- Domains of the input and output variables.
- Fuzzy sets (memberhip functions) representing the meaning of the linguistic values of the input and output variables.
  - The most popular membership functions are triangular, trapezodial and bell-shaped functions.
  - These are described by means of a small number of parameter values.

Rule Base

The Rule base aims to represent in a structure way the knowledge (of an experienced operator and/or control engineer) in the form of a set of *if-then* rules (through interviewing and verbalisation of experience-based knowledge)

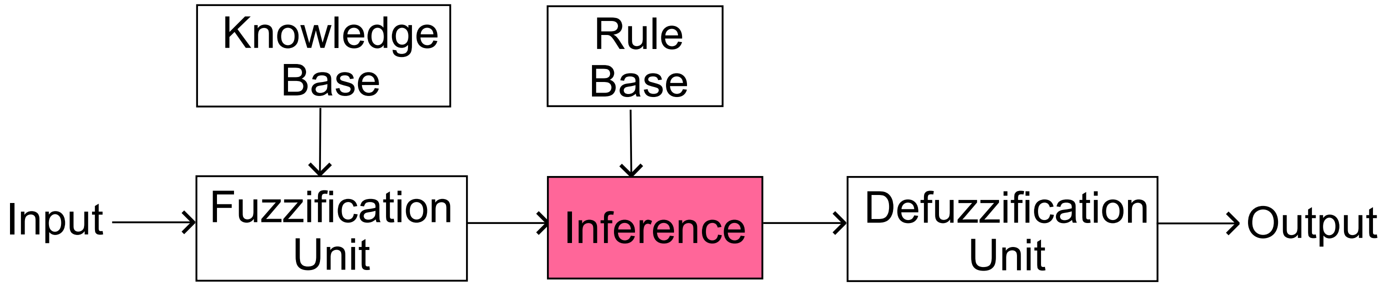


Includes:

- Fuzzy partitions: Partition of the domains of the input and output variables (choice of term-sets -number and ranges of linguistic values- for input and output variables).
- Choice of the content of the rule-antecedent and the rule-consequent.
- Derivation of the set of rules.

Inference Engine

The basic function of the inference engine is to compute the overall value of the output variable based on the individual contributions of each rule in the rule base.



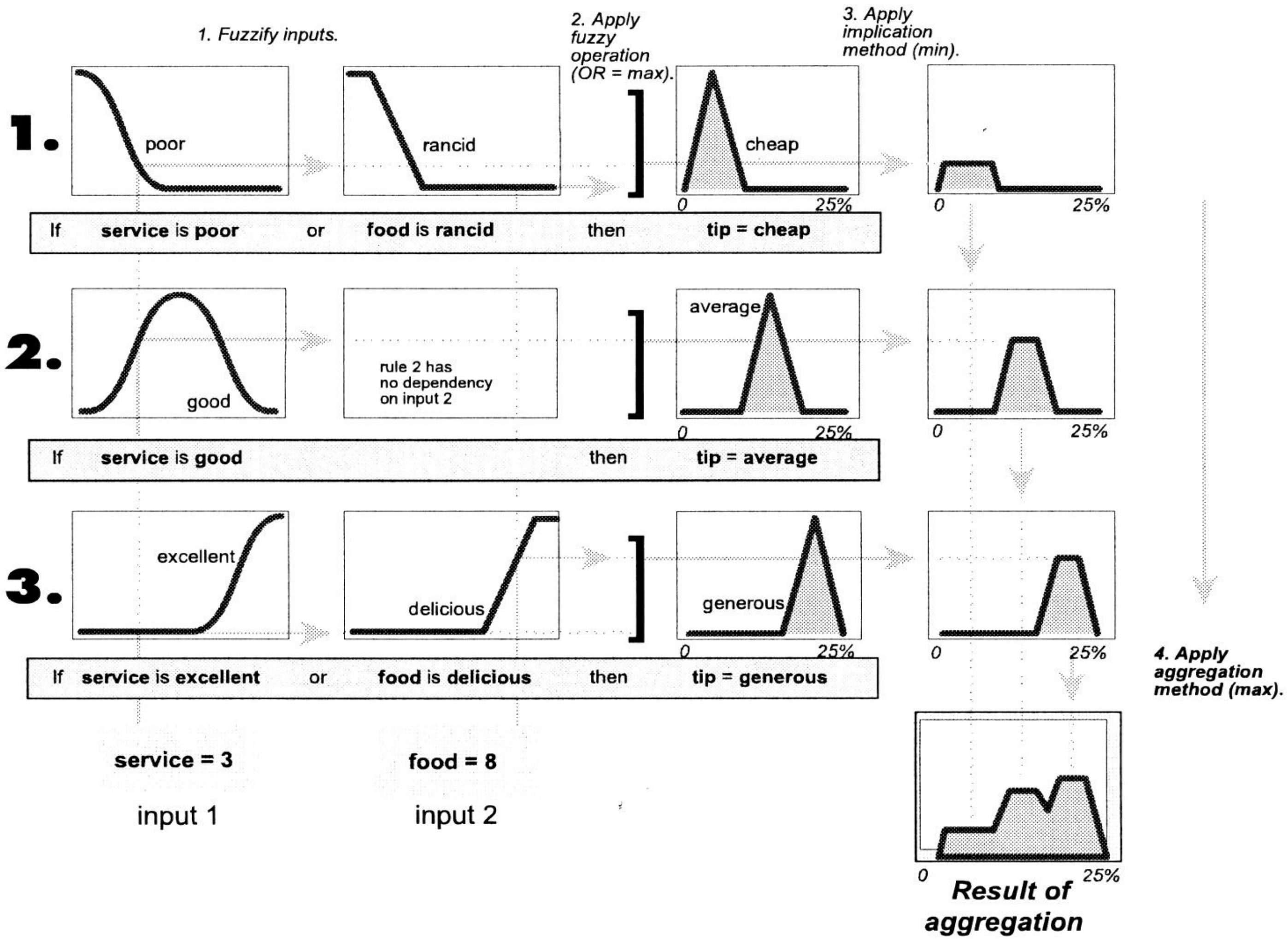
The **fuzzification unit** computes for each rule the matching of the crisp values of the input variables to each rule-antecedent – so a degree of satisfaction for each fuzzy rule is stablished, which is known as the firing level of the rule. Only rules with positive firing level are activated/considered.

The **inference module** uses **Mamdani implication** to derive the **output fuzzy set of the output variable** for each fired-rule.

The output fuzzy sets of all fired-rules are subsequently combined using a union type operator – usually maximum t-conorm.

This is illustrated with the tip example of Matlab in next slide

Inference Engine II



1. **Fuzzification** -- $Input_1$ and $Input_2$ values are mapped to the membership functions of the antecedent of each rule. Membership values in each linguistic labels are computed.
2. **Inference**

   2.1 Minimum (MIN) is applied for AND; maximum for OR, to the membership values of the labels of antecedent in a rule to derive its firing level.

   2.2. The firing level of each rule truncates the membership function of the output variable fuzzy label in the consequent. This is the output fuzzy set of the rule.

   2.3. Each rule has a fuzzy output, which are all combined with to get the final output using the MAX (result of aggregation) operator.

Note: A rule with an OR can be split in two rules

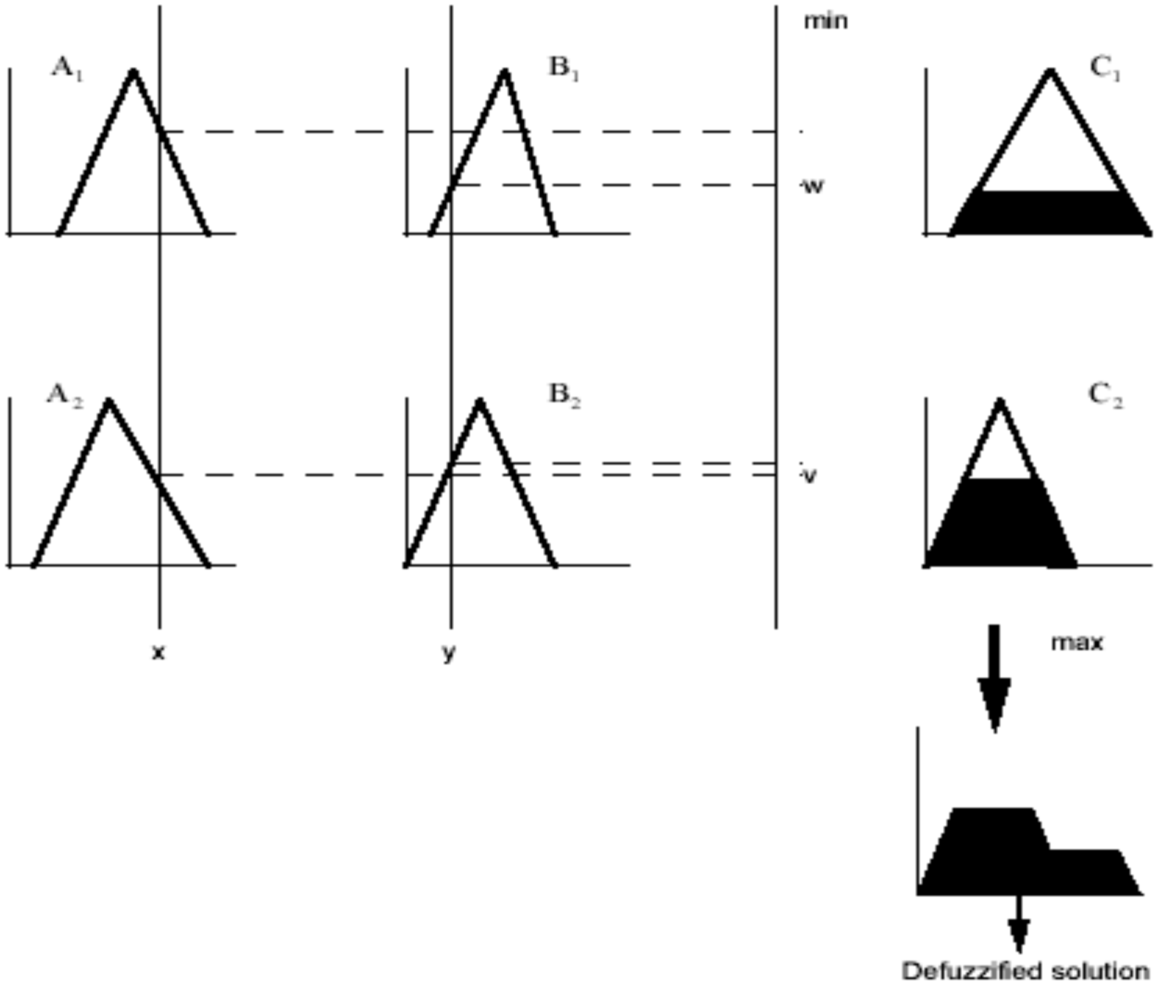
"If service is poor OR food is rancid then tip is cheap " can be split in two rules:

"If service is poor then tip is cheap " OR "If food is rancid then tip is cheap "

So, in 1. above the MIN is always used. This is referred to as the MAX-MIN Mamdani inference model.

The MAX-DOT is also possible if using product instead of minimum

Inference Engine III



1. $Input_1$ x and $Input_2$ y are mapped to the membership functions of the antecedent of each rule. For rule 1 input x is mapped to A_1 while input y is mapped to B_1. The MIN of the two membership values is kept (we are assuming there is an AND in the antecedent of rule 1).
2. This firing level of antecedent of rule 1 is used to truncate the membership function of C_1, which is the output fuzzy set of rule 1 for input (x,y).
3. This is done for rule 2 (assuming AND in antecedent), we compute output truncating C_2.
4. The rules are combined assuming an OR operation, i.e. R_1 OR R_2 is computed. Therefore, the outputs of both rules are combined using the MAX operator.
5. The final step consist in obtaining the defuzzified solution

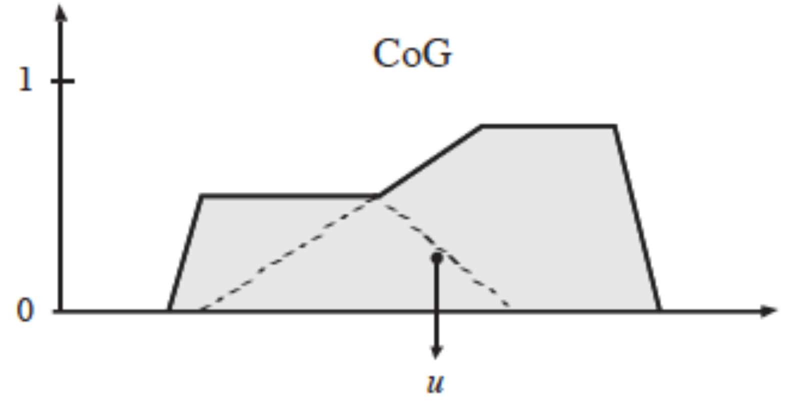
Defuzzification

- In the Mamdani FIS, the final output is a fuzzy set.
- For most applications there is a need for a ‘crisp’ decision.
- This is where defuzzification (as its name implies) reduces the fuzzy set to a single number.
- There are several defuzzification techniques available.

Defuzzification Methods I

- Centre-of-Gravity (CoG), also known as the Centre of Area (CoA), finds the centre of gravity of the final output fuzzy set.
- In the discrete case, we have

$$u = \frac{\sum_{k=1}^{l} u_k \cdot \mu_U(u_k)}{\sum_{k=1}^{l} \mu_U(u_k)}$$



- In the continuous case,

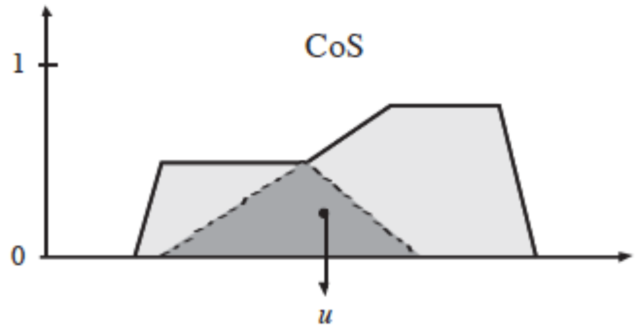
$$u = \frac{\int_{\mathcal{U}} v \cdot \mu_U(v)\ \mathrm{d}v}{\int_{\mathcal{U}} \mu_U(v)\ \mathrm{d}v}.$$

- This is done for you by Matlab.

Defuzzification Methods II

- Centre-of-Sums (CoS) is like CoG but much more efficient to implement.
- In CoS, the final output fuzzy set is not computed. Instead, each fired-rule output fuzzy set is processed individually. So, overlapping areas are counted more than once.
- In the discrete case, we have

$$u = \frac{\sum_{k=1}^{l} u_k \cdot \sum_{i=1}^{n} \mu_{U_i}(u_k)}{\sum_{k=1}^{l} \sum_{i=1}^{n} \mu_{U_i}(u_k)}$$

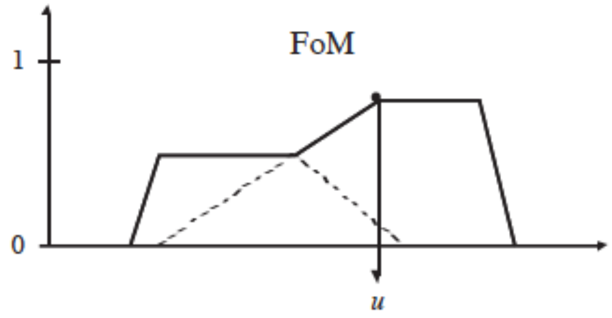


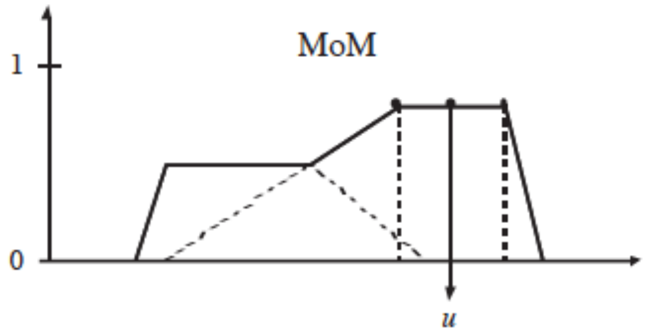
- In the continuous case,

$$u = \frac{\int_{\mathcal{U}} v \cdot \sum_{i=1}^{n} \mu_{U_i}(v)\ \mathrm{d}v}{\int_{\mathcal{U}} \sum_{i=1}^{n} \mu_{U_i}(v)\ \mathrm{d}v}.$$

Defuzzification Methods III

- First-of-Maxima (FoM) is defined as the smallest value in the domain of U with maximal membership value
- Analogously, we can define Last-of-Maxima (LoM).
- Middle-of-Maxima (MoM) is like FoM. Instead of taking the first values with maximal grades of membership, the average of all values with maximal grades is computed.





Fuzzy Logic Control

Mamdani and Takagi-Sugeno Controllers

MSc IS/ISR

The Approach of Takagi and Sugeno

It can be seen as a modification of Mamdani's model

- Fuzzy partition for the input domains have to be specified in the same way
- A fuzzy partition of the output domain is not needed
- Control rules are given in the form

$$IF\ X\ is\ A_1\ AND\ Y\ is\ B\ THEN\ Z = f(X, Y)$$

  where the output is a function of the values that X and Y may take
- Function f is usually assumed to be a linear function

Takagi-Sugeno's Rules

$$R^i: \; If \; x_1 \; is \; A_1^i, \ldots, \; x_n \; is \; A_n^i \; then \; y^i = c_0^i + c_1^i \cdot x_1 + \ldots + c_n^i \cdot x_n$$

- y^i is the output from the i-th rule
- A_j^i is a fuzzy set
- c_j^i is a consequent parameter
- The input-output data is collected while a human performs the process to control
- This data is also used in parameter identification of the above rules

Output Computation

Given input $(x_1^0, x_2^0, \ldots, x_n^0)$ the final output is

$$y = \frac{\sum_{i=1}^{n} w^i \cdot y^{i0}}{\sum_{i=1}^{n} w^i}$$

- y^{i0} is calculated for rule R^i
- The weight w^i is the firing strength of rule R^i (minimum operator)

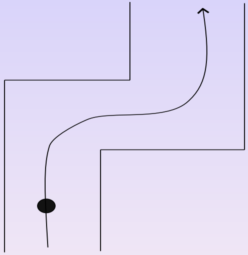
Defuzzification is superfluous for the Takagi-Sugeno fuzzy controller

Algorithm of Identification of Model

1. Choose the premise structure and consequent structure
2. Estimate the parameters of the structure determined in 1
3. Evaluate the model
4. Repeat 1-3 until satisfied with the result

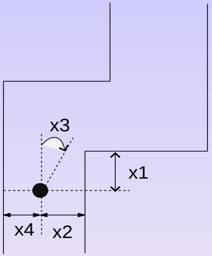
Example - Sugeno 1985

The task is to pass a bend with a model car at a constant speed



Passing a bend

Example - Sugeno 1985



Input variables for the control of the car

- x_1 : distance of the car to the beginning of the bend
- x_2 : distance of the car to the inner barrier
- x_3 : direction (angle) of the car
- x_4 : distance of the car to the outer barrier

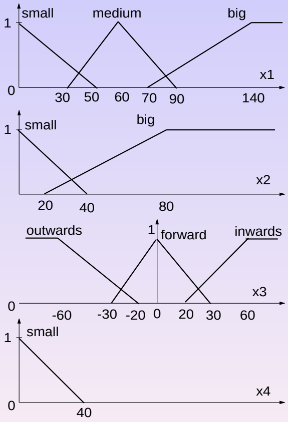
Example - Cont.

- The control variable y represents the rotation speed of the steering wheel
- The domains of the input variables are:

$$X_1 = [0, 150](cm), \quad X_2 = [0, 150](cm)$$

$$X_3 = [-90, 90](^o), \quad X_4 = [0, 150](cm)$$

Fuzzy Partition of Input Variables



Fuzzy partition of the sets X1, X2, X3, X4

Rules

$$\begin{aligned}
&IF\ x_1\ is\ A,\ x_2\ is\ B,\ x_3\ is\ C\ AND\ x_4\ is\ D\ THEN\\
&y = c_0 + c_1 \cdot x_1 + c_2 \cdot x_2 + c_3 \cdot x_3 + c_4 \cdot x_4\\
&A \in \{small,\ medium,\ big\}\\
&B \in \{small,\ big\}\\
&C \in \{outward,\ forward,\ inwards\}\\
&D \in \{small\}
\end{aligned}$$

Rules in which all four input variables do not appear simultaneously are also allowed

Control Rules for the Car

Rule	x_1	x_2	x_3	x_4	c_0	c_1	c_2	c_3	c_4
R_1	-	-	outwards	small	3.000	0.000	0.000	-0.045	-0.004
R_2	-	-	forward	small	3.000	0.000	0.000	-0.030	-0.090
R_3	small	small	outwards	-	3.000	-0.041	0.004	0.000	0.000
R_4	small	small	forward	-	0.303	-0.026	0.061	-0.050	0.000
R_5	small	small	inwards	-	0.000	-0.025	0.070	-0.075	0.000
R_6	small	big	outwards	-	3.000	-0.066	0.000	-0.034	0.000
R_7	small	big	forward	-	2.990	-0.017	0.000	-0.021	0.000
R_8	small	big	inwards	-	1.500	0.025	0.000	-0.050	0.000
R_9	medium	small	outwards	-	3.000	-0.017	0.005	-0.036	0.000
R_{10}	medium	small	forward	-	0.053	-0.038	0.080	-0.034	0.000
R_{11}	medium	small	inwards	-	-1.220	-0.016	-0.047	-0.018	0.000
R_{12}	medium	big	outwards	-	3.000	-0.027	0.000	-0.044	0.000
R_{13}	medium	big	forwards	-	7.000	-0.049	0.000	-0.041	0.000
R_{14}	medium	big	inwards	-	4.000	-0.025	0.000	-0.100	0.000
R_{15}	big	small	outwards	-	0.370	0.000	0.000	-0.007	0.000
R_{16}	big	small	forward	-	-0.900	0.000	0.034	-0.030	0.000
R_{17}	big	small	inwards	-	-1.500	0.000	0.005	-0.100	0.000
R_{18}	big	big	outwards	-	1.000	0.000	0.000	-0.013	0.000
R_{19}	big	big	forward	-	0.000	0.000	0.000	-0.006	0.000
R_{20}	big	big	inwards	-	0.000	0.000	0.000	-0.010	0.000

Exercise

Assume that the car is 10 cm away from the beginning of the bend ($x_1 = 10$). The distance of the car to the inner barrier is 30 cm ($x_2 = 30$),to the outer barrier 50 cm ($x_4 = 50$), and the car's direction is 'forward' ($x_3 = 0$). Obtain the corresponding output.

Obtain the output when $x_1 = 75,\ x_2 = 35,\ x_3 = -25, x_4 = 45$

Conclusion

- Takagi - Sugeno FLC similar to Mamdani FLC
- Difference: Output
- Linear function of inputs
- Fundamental: Test the system to estimate consequent parameter
- Linguistic variables are essential in FLCs

For Further Reading

Michio Sugeno
An introductory survey of fuzzy control
Information Sciences 36 (1985), 59-83

Hamid R. Berenji
Fuzzy Logic Controllers
In: R. R. Yager, L. A. Zadeh (Eds.) *An Introduction to Fuzzy Logic Applications in Intelligent Systems* Kluwer Academic Publishers (1992)

R. Kruse, J. Gebhardt and F. Klawonn
Foundations of Fuzzy Logic
John Wiley & Sons, Inc., 1994