SOFT COMPUTING

FUZZY CONTROLLER

Fuzzy Logic Controller

- Applications of Fuzzy logic
- Fuzzy logic controller
- Modules of Fuzzy logic controller
- Approaches to Fuzzy logic controller design
    - Mamdani approach
    - Takagi and Sugeno’s approach

Applications of Fuzzy Logic

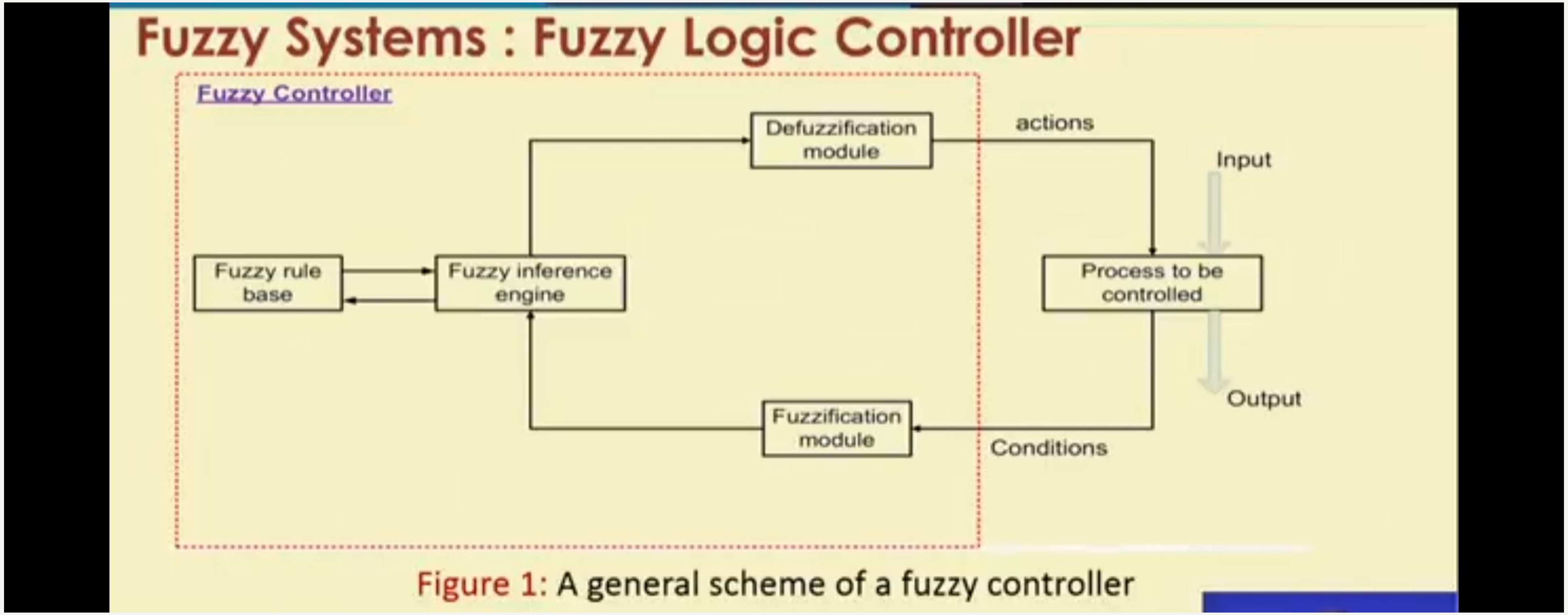
Fuzzy Systems : Fuzzy Logic Controller

- Concept of fuzzy theory can be applied in many applications, such as fuzzy reasoning, fuzzy clustering, fuzzy programming, etc.
- Out of all these applications, fuzzy reasoning, also called "fuzzy logic controller (FLC)" is an important application.
- Fuzzy logic controllers are special expert systems. In general, a FLC employs a knowledge base expressed in terms of a fuzzy inference rules and a fuzzy inference engine to solve a problem.

Fuzzy Systems : Fuzzy Logic Controller

- We use FLC where an exact mathematical formulation of the problem is not possible or very difficult.

- These difficulties are due to non-linearities, time-varying nature of the process, large unpredictable environment disturbances, etc.


Fuzzy Systems : Fuzzy Logic Controller
Fuzzy Controller
Defuzzification module
actions
Input
Fuzzy rule base
Fuzzy inference engine
Process to be controlled
Output
Fuzzification module
Conditions
Figure 1: A general scheme of a fuzzy controller

Fuzzy Systems : Fuzzy Logic Controller

A general fuzzy controller consists of four modules:

- a fuzzy rule base,
- a fuzzy inference engine,
- a fuzzification module, and
- a defuzzification module.

Fuzzy Systems : Fuzzy Logic Controller

As shown in Figure 1, a fuzzy controller operates by repeating a cycle of the following four steps :

- **Step 1:** Measurements (inputs) are taken of all variables that represent relevant condition of controller process.
- **Step 2:** These measurements are converted into appropriate fuzzy sets to express measurements uncertainties. This step is called fuzzification.

Fuzzy Systems : Fuzzy Logic Controller

- **Step 3:** The fuzzified measurements are then used by the inference engine to evaluate the control rules stored in the fuzzy rule base. The result of this evaluation is a fuzzy set (or several fuzzy sets) defined on the universe of possible actions.

- **Step 4:** This output fuzzy set is then converted into a single (crisp) value (or a vector of values). This is the final step called defuzzification. The defuzzified values represent actions to be taken by the fuzzy controller.

Fuzzy Systems : Fuzzy Logic Controller

There are manly two approaches of FLC.

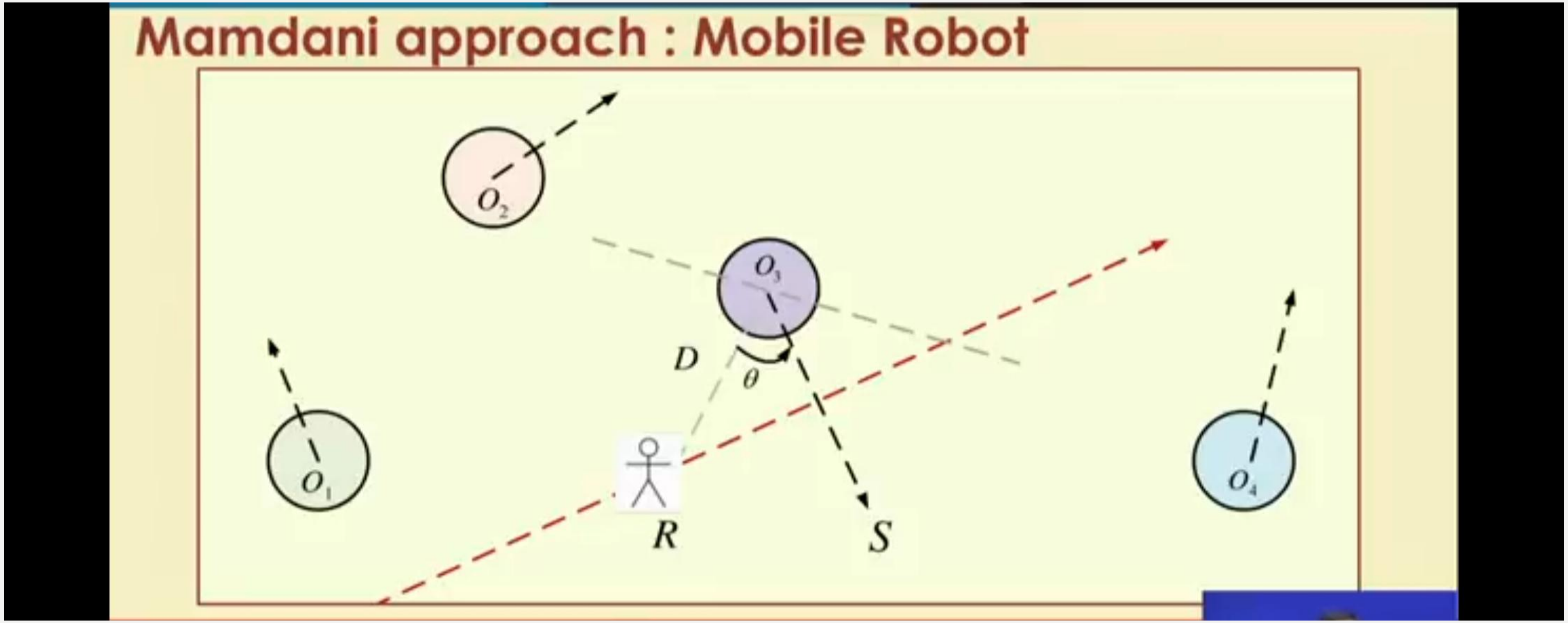
- Mamdani approach
- Takagi and sugeno's approach

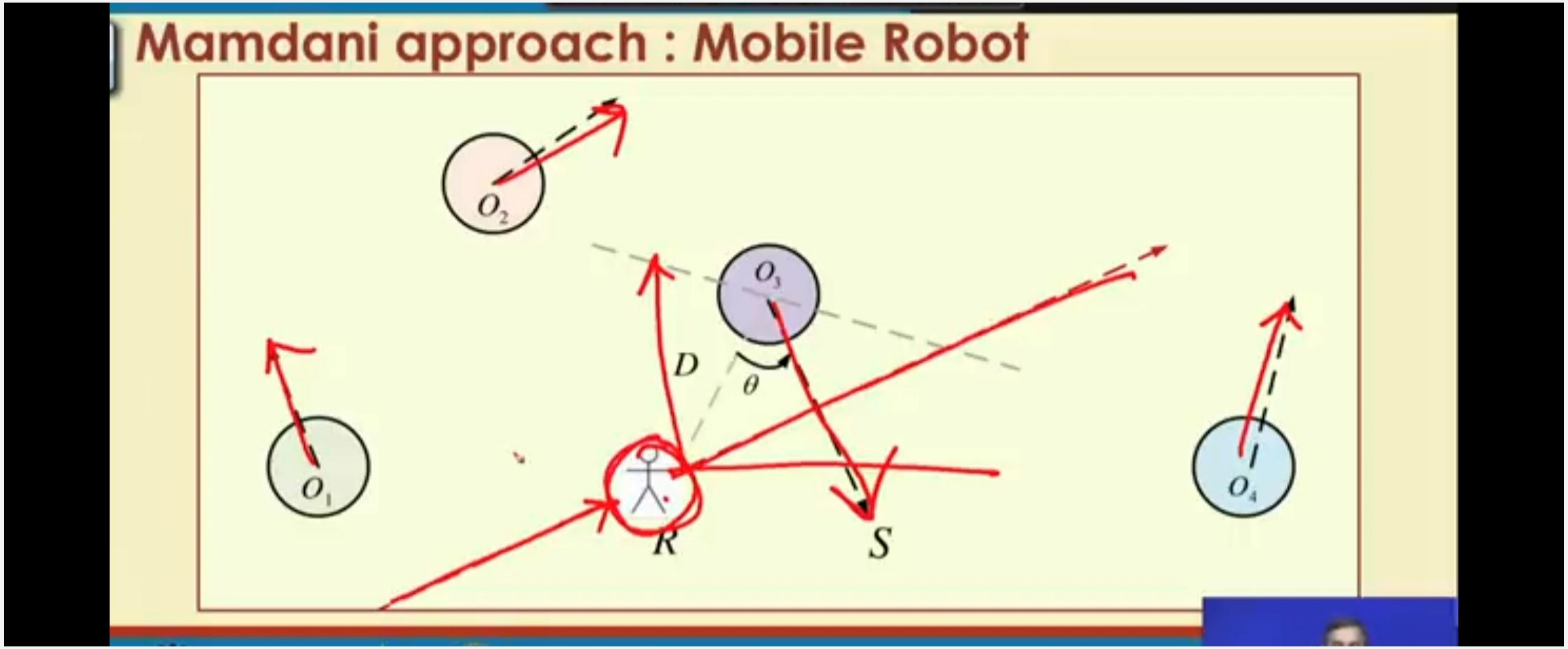
  - Mamdani approach follows linguistic fuzzy modelling and characterized by its high interpretability and low accuracy.
  - On the other hand, Takagi and Sugeno's approach follows precise fuzzy modelling and obtains high accuracy but at the cost of low interpretability.

We illustrate the above two approaches with examples.

Mamdani approach : Mobile Robot

- Consider the control of navigation of a mobile robot in the presence of a number of moving objects.
- To make the problem simple, consider only four moving objects, each of equal size and moving with the same speed.
- A typical scenario is shown in Figure 2.


Mamdani approach : Mobile Robot
O2
O3
D
θ
O1
R
S
O4



Mamdani approach : Mobile Robot
O2
O3
O1
O4
D
θ
R
S

Mamdani approach : Mobile Robot

- We consider two parameters : D, the distance from the robot to an object and θ the angle of motion of an object with respect to the robot.
- The value of these parameters with respect to the most critical object will decide an output called deviation (δ).
- We assume the range of values of D is $[0.1, \dots . 2.2]$ in meter and θ is $[-90, \dots, 0, \dots 90]$ in degree.
- After identifying the relevant input and output variables of the controller and their range of values, the Mamdani approach is to select some meaningful states called "linguistic states" for each variable and express them by appropriate fuzzy sets.

Linguistic States

For the current example, we consider the following linguistic states for the three parameters.

Distance is represented using four linguistic states:

- VN : Very Near
- NR : Near
- VF : Very Far
- FR : Far

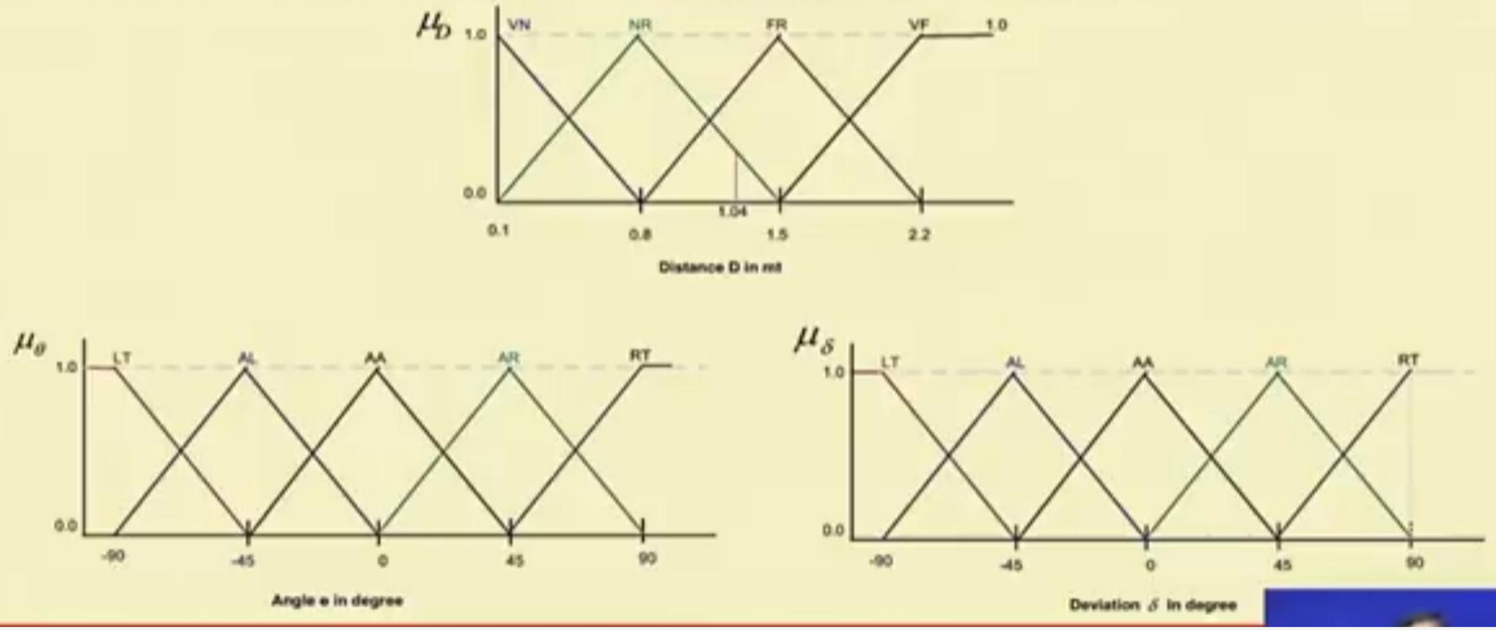
Linguistic States

Angle (for both angular direction (θ) and deviation (δ)) are represented using five linguistic states:

- LT : Left
- AL : Ahead Left
- AA : Ahead
- AR : Ahead Right
- RT : Right

Linguistic States

Three different fuzzy sets for the three different parameters are given below (Figure 3).



Fuzzy rule base

- Once the fuzzy sets of all parameters are worked out, our next step in FLC design is to decide fuzzy rule base of the FLC.
- The rule base for the FLC of mobile robot is shown in the form of a table below.

	LT	AL	AA	AR	RT
VN	AA	AR	AL	AL	AA
NR	AA	AA	RT	AA	AA
FR	AA	AA	AR	AA	AA
VF	AA	AA	AA	AA	AA

Fuzzy rule base for the mobile robot

Note that this rule base defines 20 rules for all possible instances. These rules are simple rules and take in the following forms.

- **Rule 1**: If (distance is VN) and (angle is LT) Then (deviation is AA)

  .

  .

- **Rule 13**: If (distance is FR) and (angle is AA) Then (deviation is AR)

  .

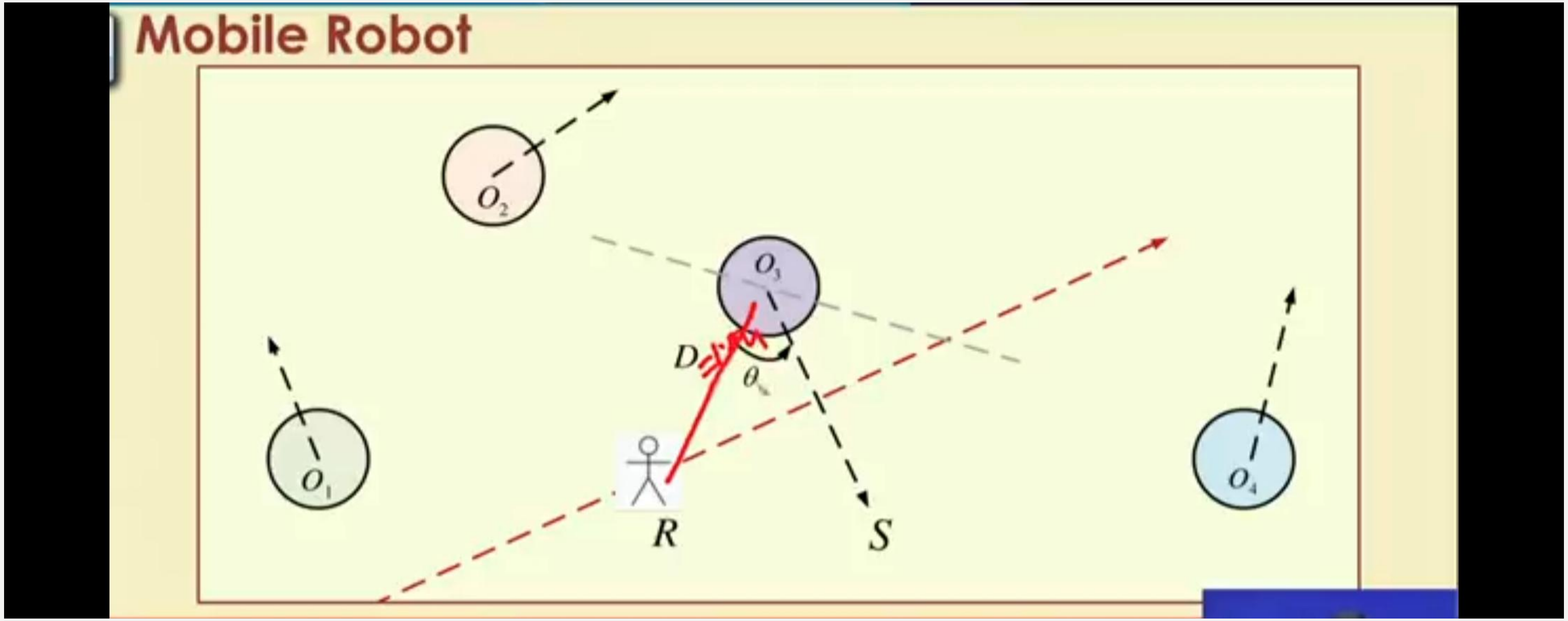
  .

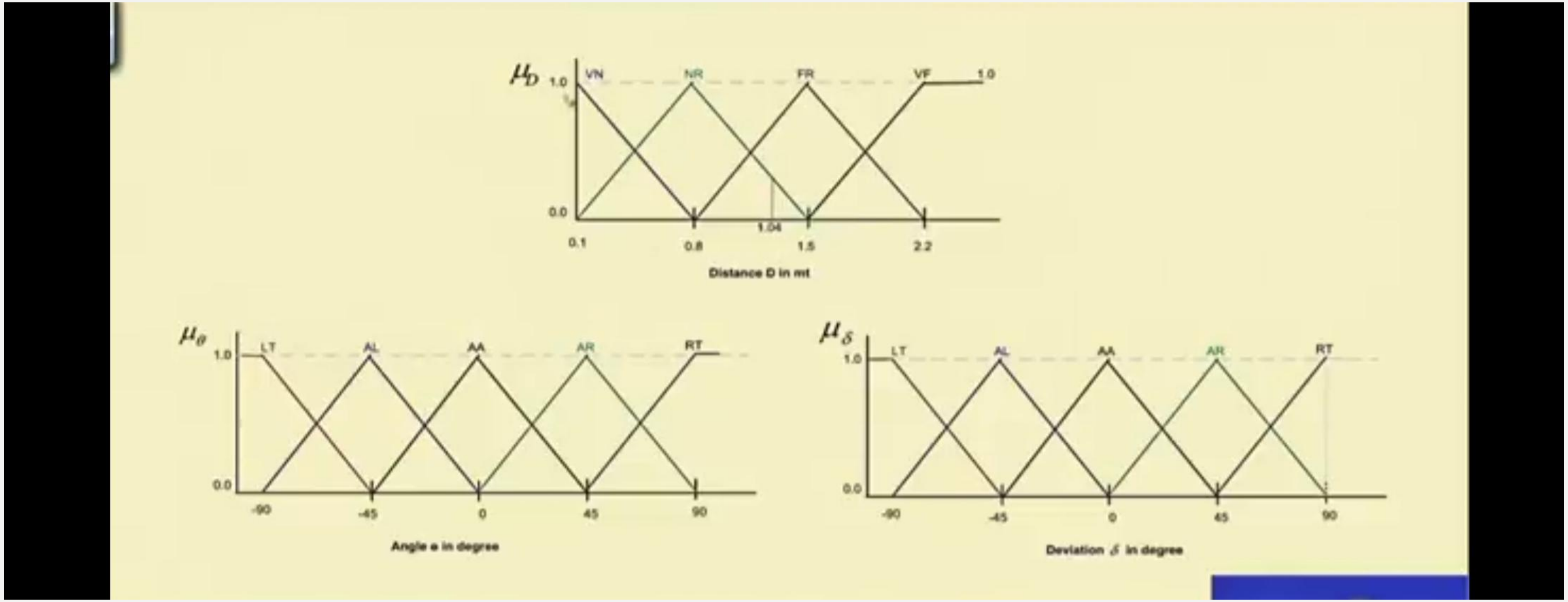
- **Rule 20**: If (distance is VF) and (angle is RT) Then (deviation is AA)

Fuzzification of inputs

- The next step is the fuzzification of inputs. Let us consider, at any instant, the object O_3 is critical to the Mobile Robot and distance $D = 1.04\, m$ and angle $\theta = 30°$.

- For this input, we are to decide the deviation δ of the robot as output.


Mobile Robot
O_2
O_3
O_1
O_4
D
R
S



μ_D
VN
NR
FR
VF
1.0
1.0
0.0
0.1
0.8
1.04
1.5
2.2
Distance D in mt
μ_θ
LT
AL
AA
AR
RT
1.0
0.0
-90
-45
0
45
90
Angle θ in degree
μ_δ
LT
AL
AA
AR
RT
1.0
0.0
-90
-45
0
45
90
Deviation δ in degree

Fuzzification of inputs

- From the given fuzzy sets and input parameters' values, we say that the distance $D = 1.04\ m$ may be called as either NR (near) or FR (far).
- Similarly, the input angle $\theta = 30°$ can be declared as either AA (ahead) or AR (ahead right).

Takagi and Sugeno's approach

- In this approach, a rule is composed of fuzzy antecedent and functional consequent parts.

- Thus, any $i-th$ rule, in this approach is represented by $If\ (x_1\ is\ A_1{}^i)\ and\ (x_2\ is\ A_2{}^i)\ \ldots\ldots\ and\ (x_n\ is\ Ani)$

- Then, $y^i = a_0{}^i + a_1{}^i\, x_1 + a_2{}^i\, x_2 + \cdots + a_n{}^i\, x_n$
- where, $a_0, a_1, a_2, \ldots\ an$ are the co-efficients.

Takagi and Sugeno's approach

- The weight of $i-th$ rule can be determined for a set of inputs $x_1, x_2, x_3, \dots x_n$ as follows.
- $w^i = \mu^i_{A_1}(x_1) \times \mu^i_{A_2}(x_2) \times \cdots \times \mu^i_{A_n}(x_n)$
- where $A_1, A_2, \dots A_n$ indicates membership function distributions of the linguistic hedges used to represent the input variables and μ denotes membership function value.

Takagi and Sugeno's approach

- $y^i = a_0{}^i + a_1{}^i x_1 + a_2{}^i x_2 + \cdots + a_n{}^i x_n$
- $w^i = \mu^i_{A_1}(x_1) \times \mu^i_{A_2}(x_2) \times \cdots \times \mu^i_{A_n}(x_n)$

- The combined action then can be obtained as

$$y = \frac{\sum_i^k w^i y^i}{\sum_i^k w^i}$$

where k denotes the total number of rules

Illustration:

Given the distribution functions for I_1 and I_2 as below.

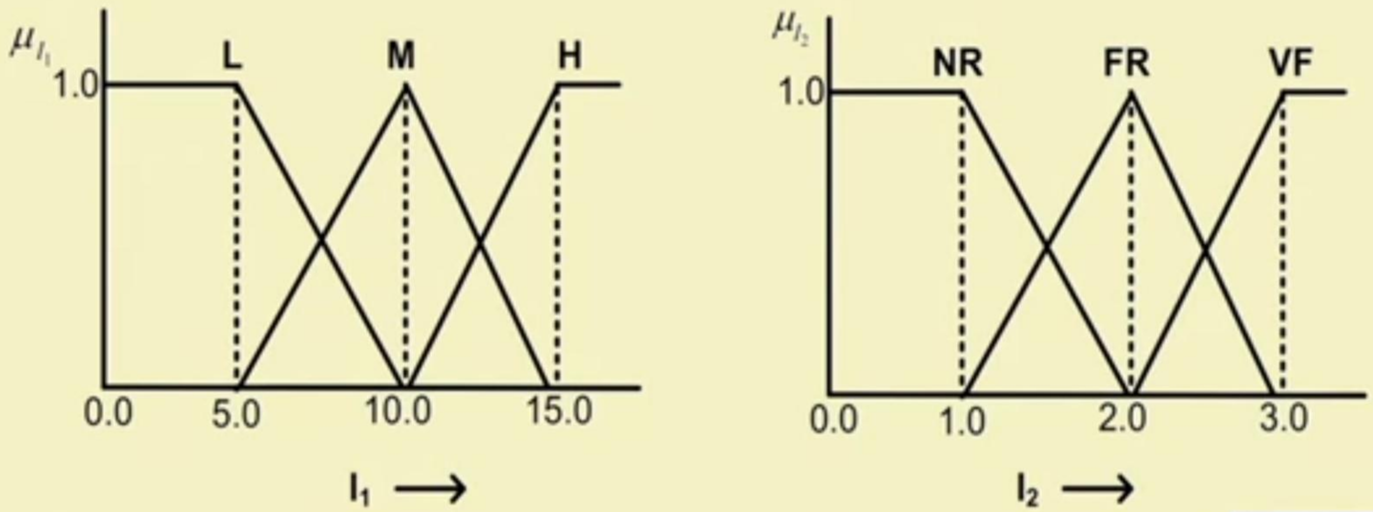


Illustration:

The output of any $i-th$ rule can be expressed by the following.

$$y^i = f(I_1, I_2) = a_j^i I_1 + b_k^i I_2;$$

where, $j, k = 1{,}2{,}3$.

Suppose:

$a_1^i = 1, a_2^i = 2, a_3^i = 3$ if $I_1 = L, M \; and \; H$, respectively.

$b_1^i = 1, b_2^i = 2, b_3^i = 3$ if $I_2 = NR, FR, and \; VF$, respectively.

We have to calculate the output of FLC for $I_1 = 6.0$ and $I_2 = 2.2$

Solution:

c) For the input set, following four rules can be fired out of all 9 rules.

- **R1**: I_1 is L and I_2 is FR
- **R2**: I_1 is L and I_2 is VF
- **R3**: I_1 is M and I_2 is FR
- **R4**: I_1 is M and I_2 is VF

Solution:

f) Therefore, the output *y* of the controller can be determined as follows.

$$y = \frac{w^1y^1 + w^2y^2 + w^3y^3 + w^4y^4}{w^1 + w^2 + w^3 + w^4}$$

$$\mathbf{y = 12.04}$$