

01_02_Fuzzy System Constructions and Fuzzy Control

01_02_Fuzzy System Constructions and Fuzzy Control

1. Fuzzy System Constructions

- Structures and Parameters

- From prior knowledge

- From Data

- Overall Process

2. Constructions From Data

- Overall Model

- Least-Squares Estimation of Singletons

 - Model

 - Method

- Least-Squares Estimation of TS Consequents

 - Model

 - Method

- Fuzzy Clustering way

 - Rule-Based Interpretation of fuzzy clusters

 - Fuzzy Clusters with TS-Model: An example

2. Fuzzy-Control

- Motivation

- Overview

- Fuzzy Controllers

- Mamdani Controller

 - Rule Base

- TS Controller

 - A simple example

- Fuzzy Supervisory Control

 - Supervisory Controller

 - Fuzzy Controller ways

3. Basic Process of Designing Fuzzy Controller

- Process

- Example 1: Fuzzy Friction Compensation

- Example 2: Supervisory fuzzy controller

Summary

1. Fuzzy System Constructions

Structures and Parameters

With regard to the design of fuzzy (and also other) models, two basic items are distinguished: the **structure** and the **parameters** of the model.

A model with a **rich structure** is able to approximate **more complicated functions**, but, at the same time, has **worse generalization** properties.

From prior knowledge

The expert knowledge expressed in a **verbal form** is **translated** into a collection of **if-then rules**.

can be seen as "Fuzzy expert systems"

From Data

It is expected that the extracted rules and membership functions can provide an a posteriori interpretation of the system's behavior

Overall Process

1. Select the **input and output** variables, the **structure** of the rules, and the **inference** and **defuzzification** methods.
2. Decide on the **number of linguistic terms** for each variable and define the corresponding **membership functions**.
3. **Formulate the available knowledge** in terms of fuzzy if-then rules
4. **Validate the model** (typically by using data). If the model does not meet the expected performance, iterate on the above design steps.

2. Constructions From Data

Overall Model

a set of N input-output data pairs $(x_i, y_i) | i = 1, 2, \dots, N$ is available for the construction of a fuzzy system.

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]^T, \quad \mathbf{y} = [y_1, \dots, y_N]^T$$

x_k is column vector

Least-Squares Estimation of Singletons

Model

Given A_i and a set of input-output data:

$$\{(\mathbf{x}_k, y_k) \mid k = 1, 2, \dots, N\}$$

Estimate optimal consequent parameters b_i

$$R_i : \text{If } x \text{ is } A_i \text{ then } y = b_i$$

Method

- Compute the membership degrees $\mu_{A_i}(x_k)$
- Normalize

$$\gamma_{ki} = \mu_{A_i}(x_k) / \sum_{j=1}^K \mu_{A_j}(x_k)$$

- put output $y_k = \sum_{i=1}^K \gamma_{ki} b_i$ in a matrix form $\mathbf{y} = \mathbf{\Gamma} \mathbf{b}$
- Least-square Estimate

$$\mathbf{b} = [\mathbf{\Gamma}^T \mathbf{\Gamma}]^{-1} \mathbf{\Gamma}^T \mathbf{y}$$

Least-Squares Estimation of TS Consequents

Model

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \mathbf{\Gamma}_i = \begin{bmatrix} \gamma_{i1} & 0 & \cdots & 0 \\ 0 & \gamma_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \gamma_{iN} \end{bmatrix}$$

$$\boldsymbol{\theta}_i = [\mathbf{a}_i^T \quad b_i]^T, \quad \mathbf{X}_e = [\mathbf{X} \quad \mathbf{1}]$$

Method

Global LS

$$\boldsymbol{\theta}' = [(\mathbf{X}')^T \mathbf{X}']^{-1} (\mathbf{X}')^T \mathbf{y}$$

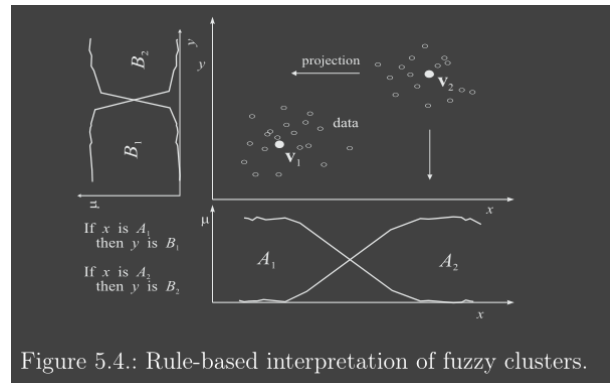
$$\mathbf{X}' = [\mathbf{\Gamma}_1 \mathbf{X}_e \mathbf{\Gamma}_2 \mathbf{X}_e \dots \mathbf{\Gamma}_c \mathbf{X}_e]$$

$$\boldsymbol{\theta}' = [\boldsymbol{\theta}_1^T \boldsymbol{\theta}_2^T \dots \boldsymbol{\theta}_c^T]$$

Fuzzy Clustering way

Rule-Based Interpretation of fuzzy clusters

- Cluster
- Interpretation

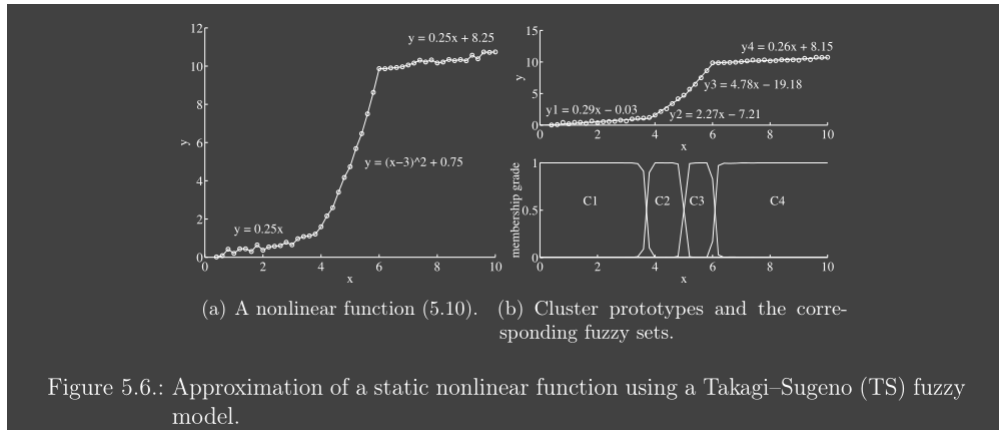


Fuzzy Clusters with TS-Model: An example

- cluster
- use linear prototypes to approximate subspaces

Example 5.2 Consider a nonlinear function $y = f(x)$ defined piece-wise by:

$$\begin{aligned} y &= 0.25x, & \text{for } x \leq 3 \\ y &= (x-3)^2 + 0.75, & \text{for } 3 < x \leq 6 \\ y &= 0.25x + 8.25, & \text{for } x > 6 \end{aligned} \quad (5.10)$$



X_1 : If x is C_1 then $y = 0.29x - 0.03$
 X_2 : If x is C_2 then $y = 2.27x - 7.21$
 X_3 : If x is C_3 then $y = 4.78x - 19.18$
 X_4 : If x is C_4 then $y = 0.26x + 8.15$

$$\theta_i = [a_i^T \quad b_i]^T, \quad \mathbf{X}_e = [\mathbf{X} \quad \mathbf{1}]$$

2. Fuzzy-Control

Motivation

The underlying principle of **knowledge-based (expert) control** is to **capture and implement experience and knowledge** available from experts (e.g., process operators)

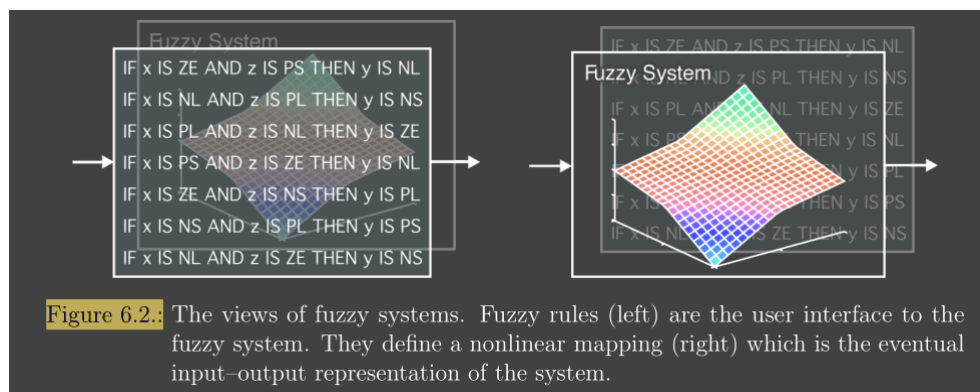
Overview

Three aspects of a fuzzy controller

1. approximation capability
2. efficiency
3. transparent

Two view of a fuzzy controller

- fuzzy if-then rules
- non-linear mapping



Fuzzy Controllers

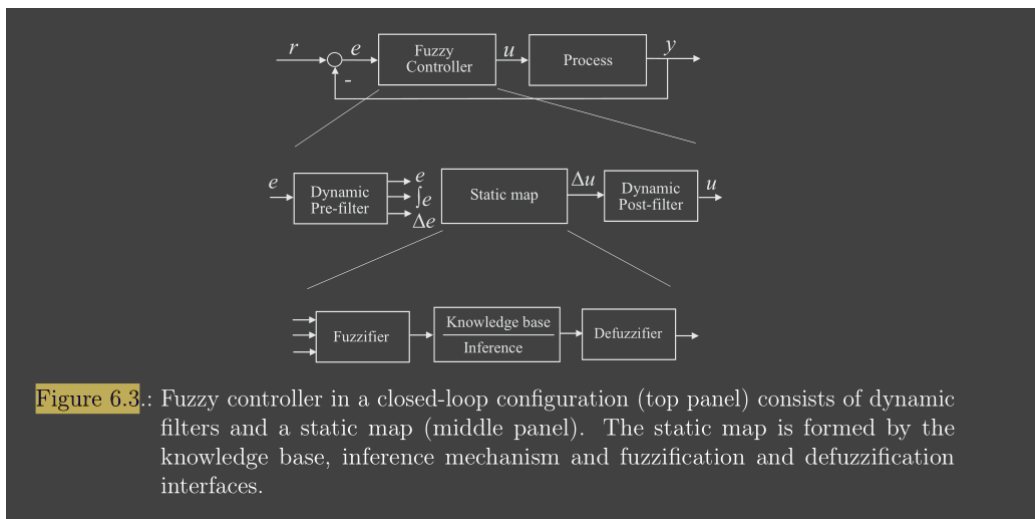
A fuzzy controller is a controller that contains a (nonlinear) mapping that has been defined by using fuzzy if-then rules.

- In most cases a fuzzy controller is used for direct feedback control.

Most often used fuzzy controller:

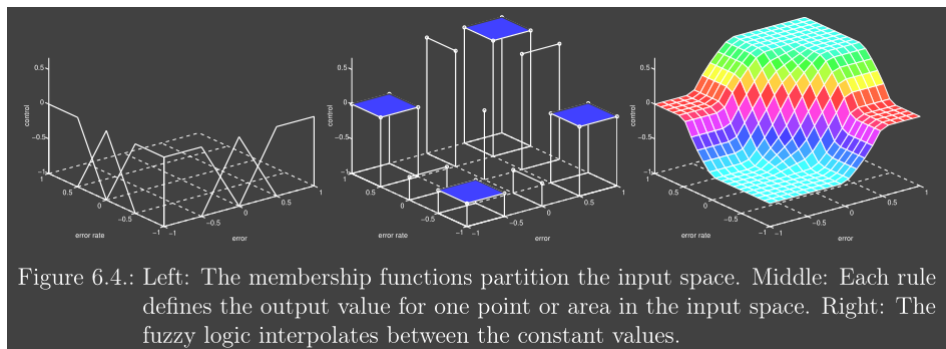
- Mamdani
- TS model

Mamdani Controller



Rule Base

\mathcal{R}_i : If x_1 is $A_{i1} \dots$ and x_n is A_{in} then u is B_i , $i = 1, 2, \dots, K$

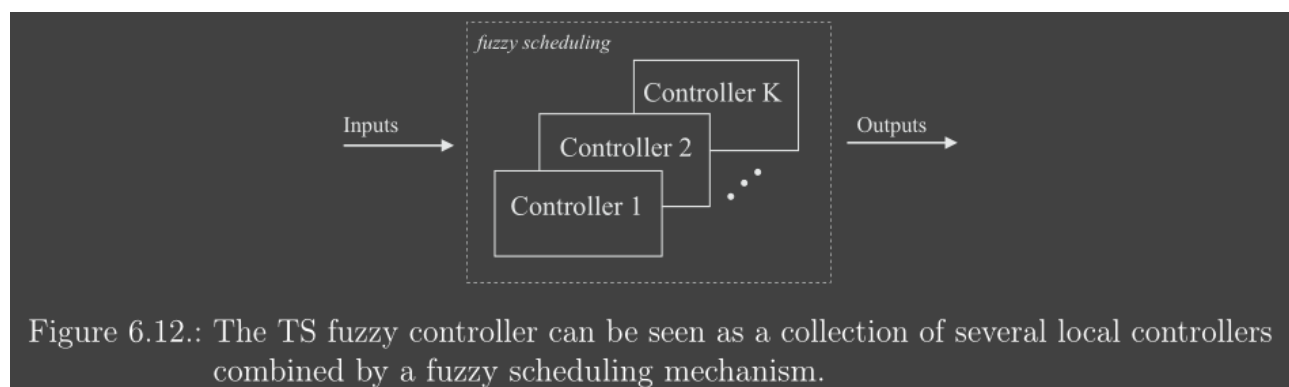


TS Controller

Several linear controllers are defined, each valid in one particular region of the controller's input space. The output is based on:

- Selecting one
- Interpolation

Therefore, the TS controller can be seen as a simple form of supervisory control.



A simple example

$$\begin{aligned}\mathcal{R}_1 &: \text{ If } r \text{ is } Low \text{ then } u_1 = P_{Low}e + D_{Low}\dot{e} \\ \mathcal{R}_2 &: \text{ If } r \text{ is } High \text{ then } u_2 = P_{High}e + D_{High}\dot{e}\end{aligned}$$

$$\begin{aligned}u &= \frac{\mu_{Low}(r)u_1 + \mu_{High}(r)u_2}{\mu_{Low}(r) + \mu_{High}(r)} \\ &= \frac{\mu_{Low}(r)(P_{Low}e + D_{Low}\dot{e}) + \mu_{High}(r)(P_{High}e + D_{High}\dot{e})}{\mu_{Low}(r) + \mu_{High}(r)}\end{aligned}$$

Fuzzy Supervisory Control

Supervisory Controller

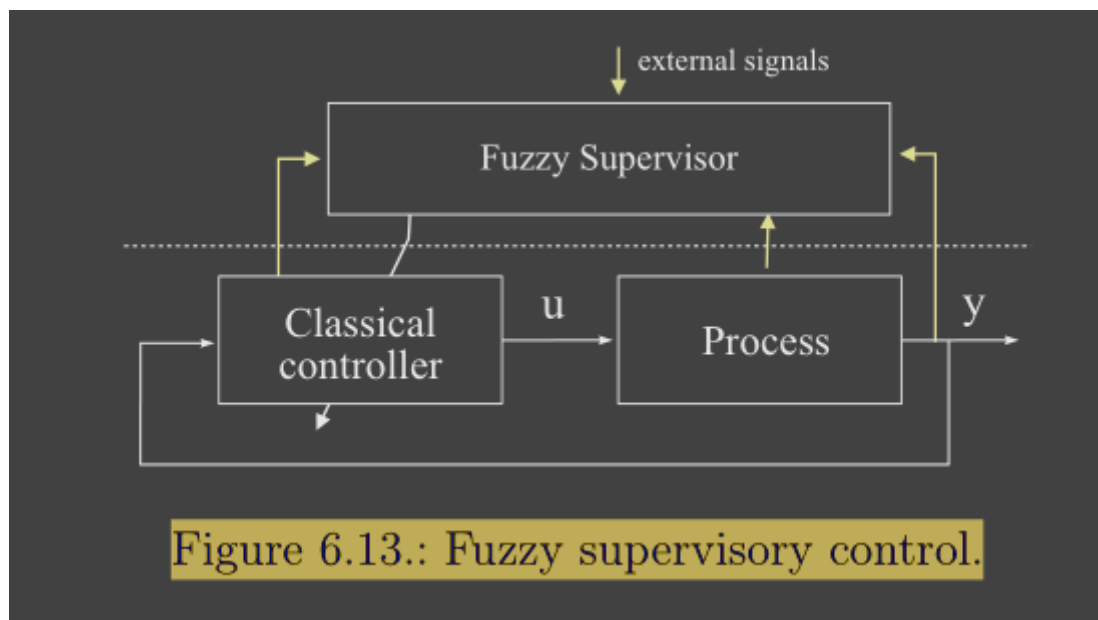
A **supervisory controller** is a **secondary controller** which augments the existing controller so that the control objectives can be met which would not be possible without the supervision.

- For instance, adjust the parameters of a low-level controller according to the process information

Fuzzy Controller ways

static or dynamic behavior of the low-level control system can be modified in order to cope with process nonlinearities or changes in the operating or environmental conditions.

An **advantage** of a supervisory structure is that it can be added to already existing control systems.

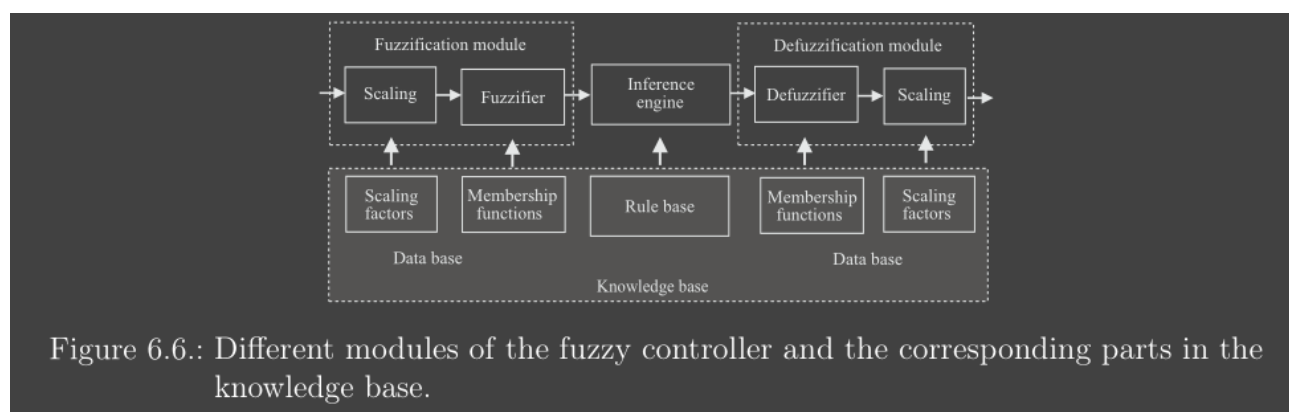


Many processes in the industry are controlled by PID controllers, they tried to retune PD when the operating conditions change, For example:

If process output is *High*
 then reduce proportional gain *Slightly* and
 increase derivative gain *Moderately*.

3. Basic Process of Designing Fuzzy Controller

Process



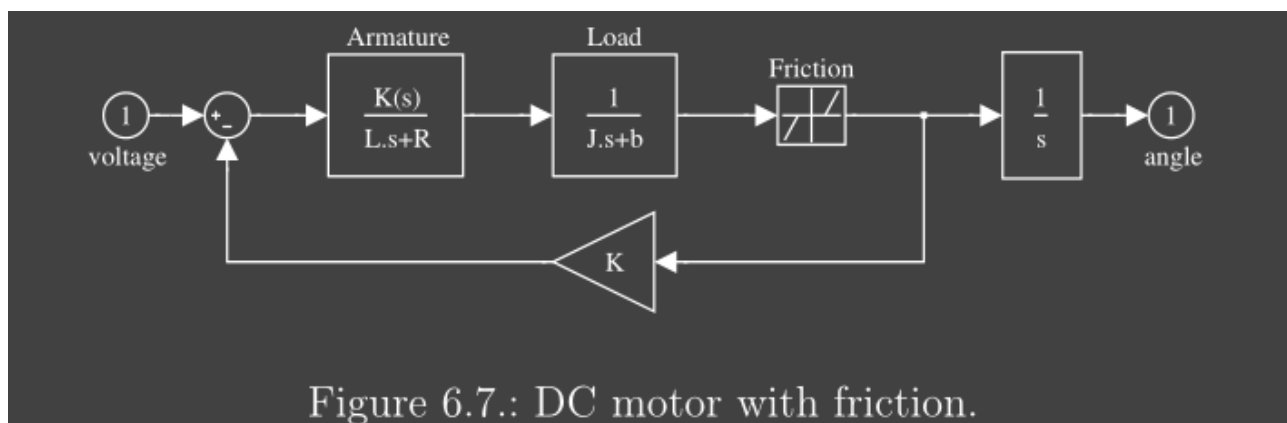
- decide **how many linguistic terms** per input variable will be used

- A good choice may be to **start with a few terms** (e.g. 2 or 3 for the inputs and 5 for the outputs) and increase these numbers when needed.
- Decide **Membership Functions** and **Scaling Factors**
 - Membership Functions based on expert's knowledge, If such knowledge is not available, membership functions of the same shape, **uniformly distributed over the domain** can be used as an initial setting and can be tuned later.
- Design the **Rule Base**
 - based entirely on the expert's intuitive knowledge and experience.
 - uses a fuzzy model of the process from which the controller rule base is derived.
- **Tune** the controller
 - The scaling factors → the most global effect
 - the membership functions → more localized effect

Or another way can be followed:

- Initialized a fuzzy controller by using an existing linear control law
- Then Tune it

Example 1: Fuzzy Friction Compensation



- First **Mimic** the linear controller

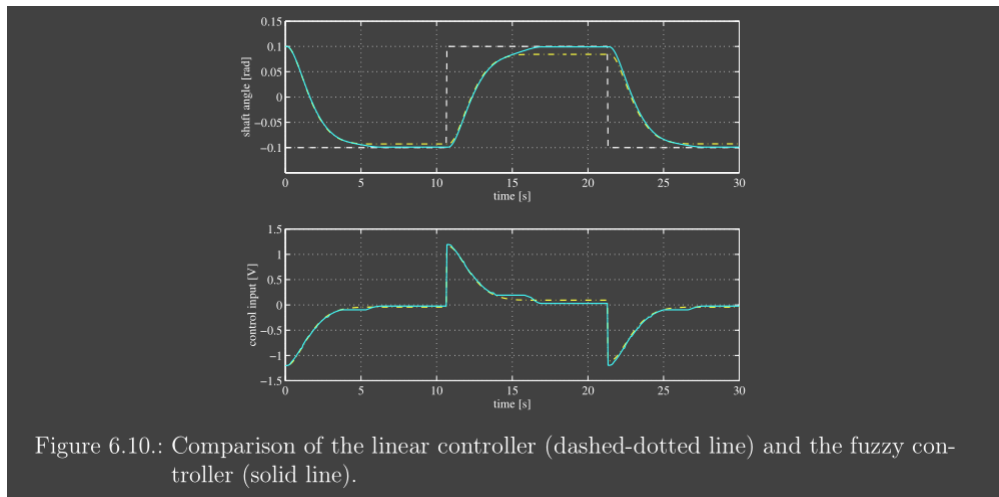
```

If error is Zero
    then control input is Zero;
If error is Positive Big
    then control input is Positive Big;
If error is Negative Big
    then control input is Negative Big;
  
```

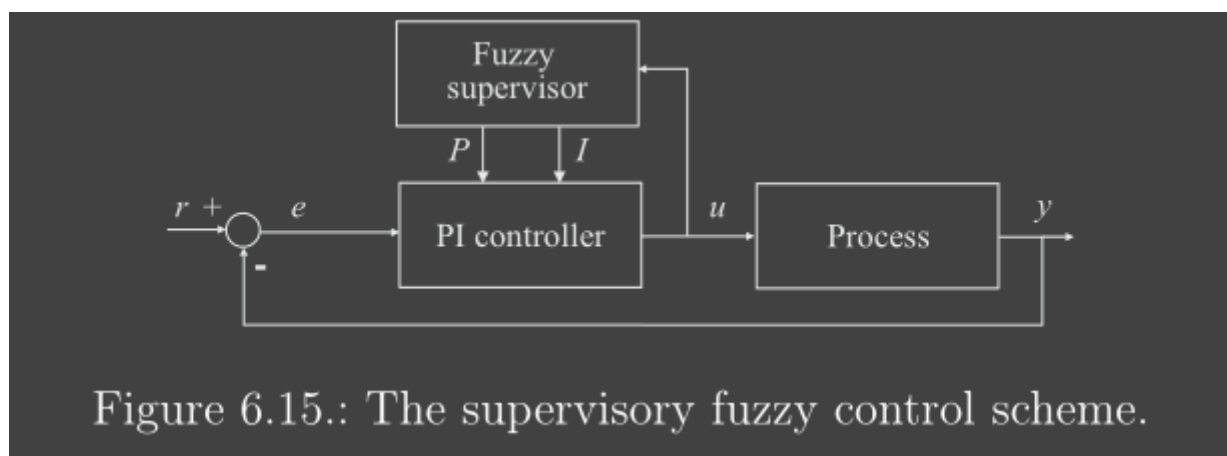
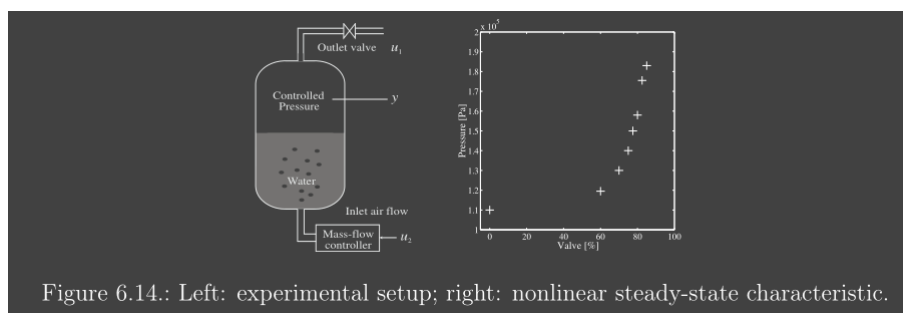
- Then tune it

```

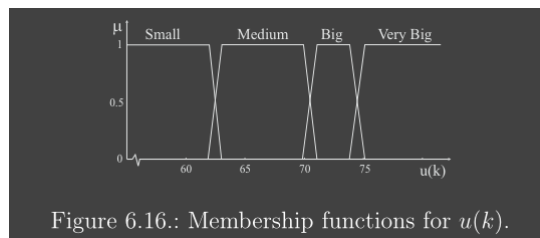
If error is Negative Small
    then control input is NOT Negative Small;
If error is Positive Small
    then control input is NOT Positive Small;
  
```



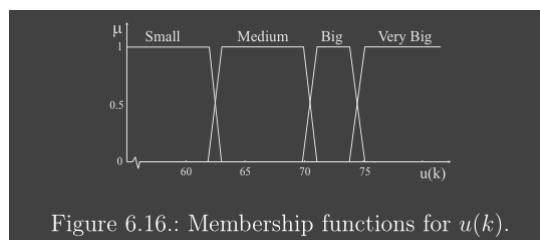
Example 2: Supervisory fuzzy controller



The domain of the valve position (0–100%) was **partitioned into four fuzzy sets** ('Small', 'Medium', 'Big' and 'Very Big'), and define P and I for each set



Gains \ $u(k)$	Small	Medium	Big	Very big
P	190	170	155	140
I	150	90	70	50



Gains \ $u(k)$	Small	Medium	Big	Very big
P	190	170	155	140
I	150	90	70	50

Summary

- A fuzzy logic controller can be seen as a **small real-time expert system** implementing a part of human operator's or process engineer's expertise.
- From the control engineering perspective, a fuzzy controller is a **nonlinear controller**
- **Nonlinear and partially known systems** that pose problems to conventional control techniques can be tackled using fuzzy control.