Fuzzy System Constructions and Fuzzy Control

1. Fuzzy System Constructions

1.1. 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.

1.2. 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"

1.3. From Data

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

1.4. Overal 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

Overal Model

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

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T, \quad \mathbf{y} = [y_1, \ldots, y_N]^T \tag{1}$$

x_k is column vector

Least-Squares Estimation of Singletons

Model

Given A_i and a set of input–output data:

$$\{(\boldsymbol{x}_k, y_k) \mid k = 1, 2, \ldots, N\} \tag{2}$$

Estimate optimal consequent parameters b_i

$$R_i : \text{ If } x \text{ is } A_i \text{ then } y = b_i \tag{3}$$

Method

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

$$\gamma_{ki} = \mu_{A_i}(x_k) / \sum_{j=1}^{K} \mu_{A_j}(x_k) \tag{4}$$

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

$$\boldsymbol{b} = \left[\boldsymbol{\Gamma}^T \boldsymbol{\Gamma}\right]^{-1} \boldsymbol{\Gamma}^\top \boldsymbol{y} \tag{5}$$

Least-Squares Estimation of TS Consequents

Model

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \boldsymbol{\Gamma}_i = \begin{bmatrix} \gamma_{i1} & 0 & \cdots & 0 \\ 0 & \gamma_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \gamma_{iN} \end{bmatrix} \tag{6}$$

$$\boldsymbol{\theta}_i = \left[\boldsymbol{a}_i^T \quad b_i \right]^T, \quad \boldsymbol{X}_e = [\boldsymbol{X} \quad \mathbf{1}] \tag{7}$$

Method

<u>Global LS</u>

$$\boldsymbol{\theta}' = \left[(\boldsymbol{X}')^T \boldsymbol{X}' \right]^{-1} (\boldsymbol{X}')^T \boldsymbol{y} \tag{8}$$

$$\begin{aligned} \mathbf{X}' &= [\boldsymbol{\Gamma}_1 \mathbf{X}_e \, \boldsymbol{\Gamma}_2 \mathbf{X}_e \ldots \boldsymbol{\Gamma}_c \boldsymbol{X}_e] \\ \boldsymbol{\theta}' &= [\boldsymbol{\theta}_1^T \boldsymbol{\theta}_2^T \ldots \boldsymbol{\theta}_c^T] \end{aligned} \tag{9}$$

Fuzzy Clustering way

Rule-Based Interpretation of fuzzy clusters

- Cluster
- Interpretation

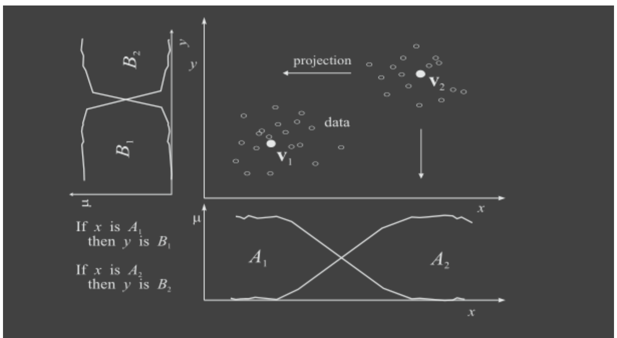


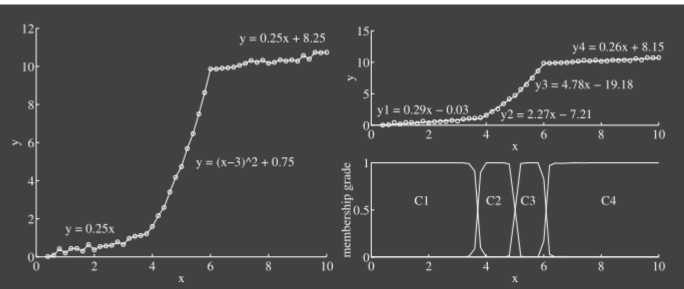
Figure 5.4.: Rule-based interpretation of fuzzy clusters.

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:

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



(a) A nonlinear function (5.10). (b) Cluster prototypes and the corresponding fuzzy sets.

Figure 5.6.: Approximation of a static nonlinear function using a Takagi–Sugeno (TS) fuzzy model.

$$\begin{aligned} X_1 &: \textbf{If } x \text{ is } C_1 \textbf{ then } y = 0.29x - 0.03 \\ X_2 &: \textbf{If } x \text{ is } C_2 \textbf{ then } y = 2.27x - 7.21 \\ X_3 &: \textbf{If } x \text{ is } C_3 \textbf{ then } y = 4.78x - 19.18 \\ X_4 &: \textbf{If } x \text{ is } C_4 \textbf{ then } y = 0.26x + 8.15 \end{aligned}$$

2. Fuzzy-Control

2.1. Motivation

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

2.2. 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

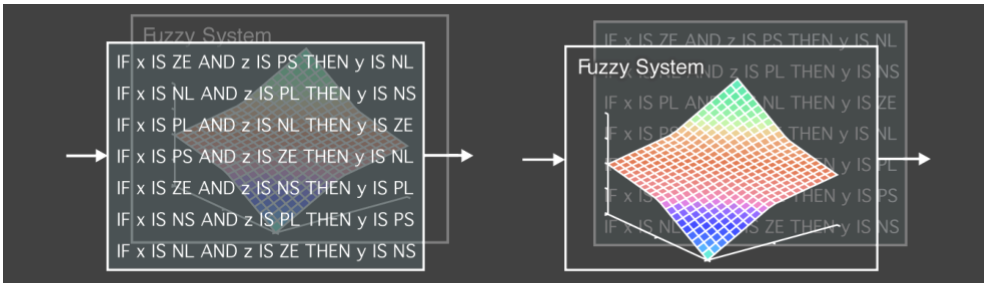


Figure 6.2.: The views of fuzzy systems. Fuzzy rules (left) are the user interface to the fuzzy system. They define a nonlinear mapping (right) which is the eventual input–output representation of the system.

2.3. 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

2.4. Mamdani Controller

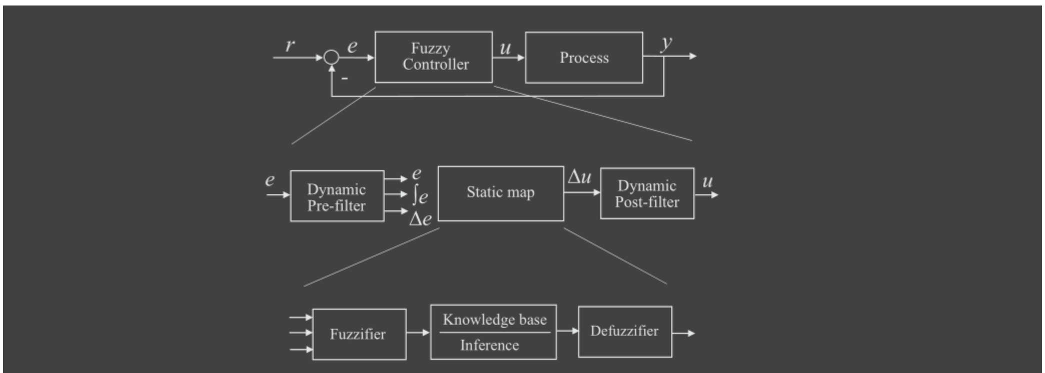


Figure 6.3.: Fuzzy controller in a closed-loop configuration (top panel) consists of dynamic filters and a static map (middle panel). The static map is formed by the knowledge base, inference mechanism and fuzzification and defuzzification interfaces.

Rule Base

$$\mathcal{R}_i: \text{ If } x_1 \text{ is } A_{i1} \ldots \text{ and } x_n \text{ is } A_{in} \text{ then } u \text{ is } B_i, \quad i = 1, 2, \ldots, K \tag{10}$$

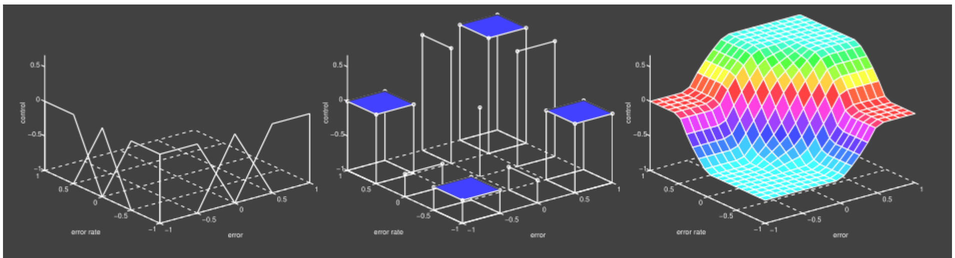


Figure 6.4.: Left: The membership functions partition the input space. Middle: Each rule defines the output value for one point or area in the input space. Right: The fuzzy logic interpolates between the constant values.

2.5. 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.

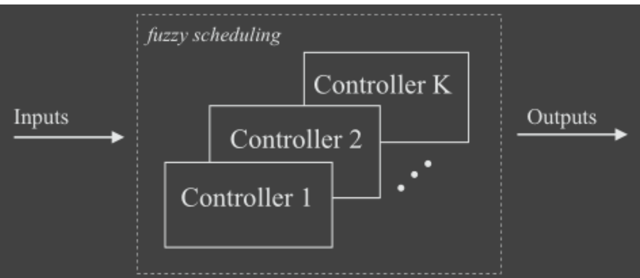


Figure 6.12.: The TS fuzzy controller can be seen as a collection of several local controllers combined by a fuzzy scheduling mechanism.

A simple example

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

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

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.

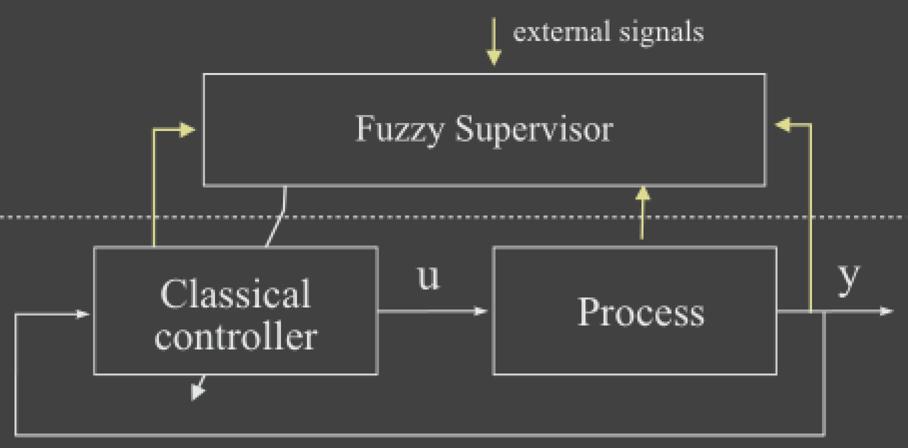


Figure 6.13.: Fuzzy supervisory control.

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

3.1. Process

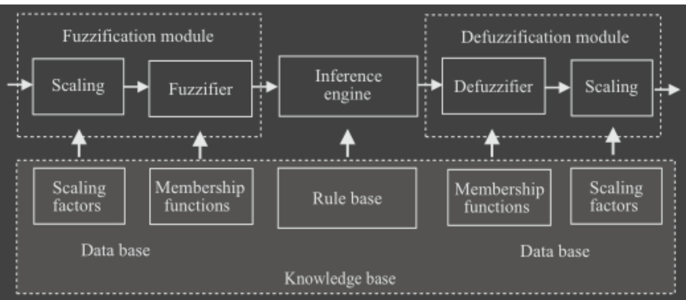


Figure 6.6.: Different modules of the fuzzy controller and the corresponding parts in the knowledge base.

- 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 Function**s 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 $\rightarrow$ the most global effect
    - the membership functions $\rightarrow$ more localized effect

Or another way can be followed:

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

Example 1: Fuzzy Friction Compenstation

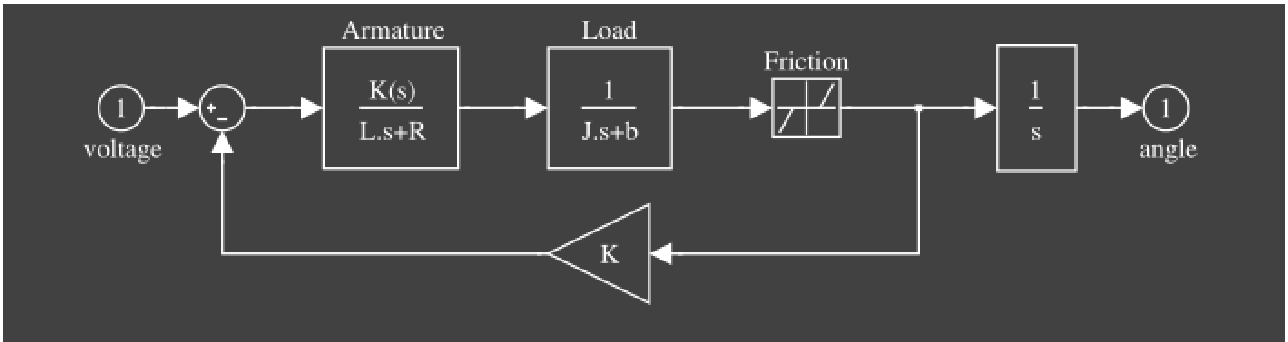


Figure 6.7.: DC motor with friction.

- 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;
```

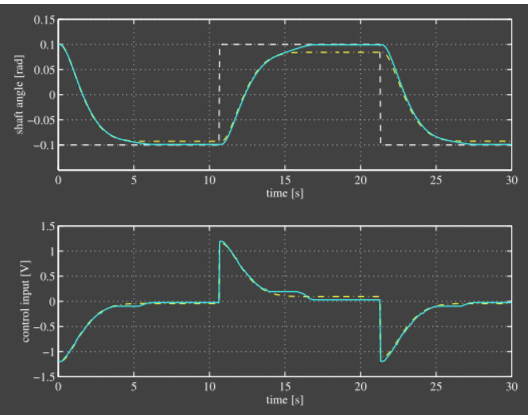


Figure 6.10.: Comparison of the linear controller (dashed-dotted line) and the fuzzy controller (solid line).

Example 2: Supervisory fuzzy controller

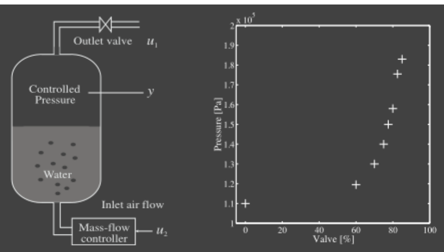


Figure 6.14.: Left: experimental setup; right: nonlinear steady-state characteristic.

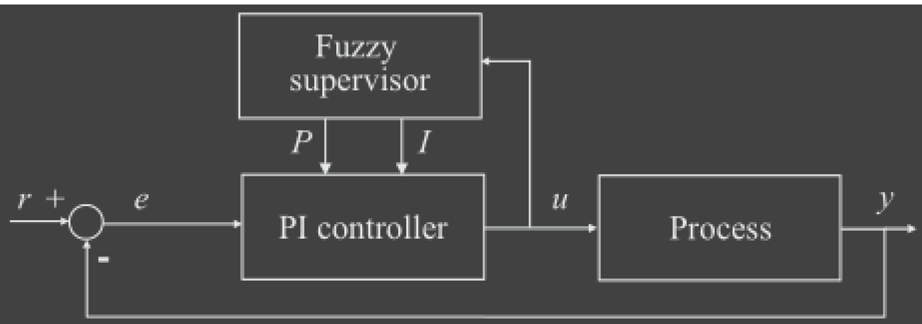


Figure 6.15.: The supervisory fuzzy control scheme.

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

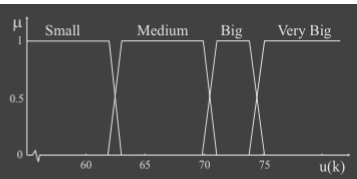


Figure 6.16.: Membership functions for $u(k)$.

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

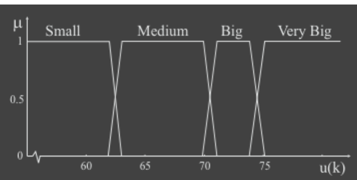


Figure 6.16.: Membership functions for $u(k)$.

Gains $\backslash u(k)$	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.