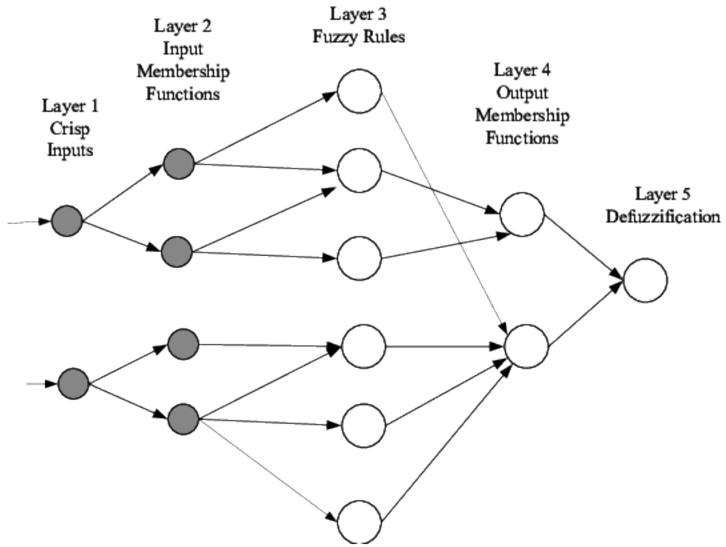
Machine intelligence

Fuzzy logic control and Neuro-fuzzy systems



Dr. Ahmad Al-Mahasneh

PID-like FLC

PID-like FLC: A further option to obtain a better performance in respect of rise time, settling time, overshoot and steady-state error is to develop a proportional-integral-derivative (PID)-like FLC. The basic idea of a PID controller is to choose the control law by considering the error e, change of error Δe and integral of error Σe (or $\int_0^t e dt$). The PID-type fuzzy controller is described by

$$u_{\text{PID}} = k_P \cdot e + k_d \cdot \Delta e + k_I \cdot \int_0^t e \cdot dt \tag{3.40}$$

By replacing the integral of error term $\int_0^t edt$ with the sum of error term Σe, the PID-type fuzzy controller in discrete time is described by

$$u_{\text{PID}} = k_P \cdot e + k_d \cdot \Delta e + k_I \cdot \Sigma e \tag{3.41}$$

The fuzzy control rule corresponding to the PID controller (as shown in Figure 3.31) has the form

$$\text{If } e \text{ is } A_i \text{ and } \Delta e \text{ is } B_j \text{ and } \Sigma e \text{ is } C_k \text{ then } u \text{ is } D_l \tag{3.42}$$

where $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$, $k = 1, \ldots, n_3$ and $l = 1, \ldots, m$. Theoretically, the number of rules to cover all possible input combinations and variations for a three-term fuzzy controller is $n_1 \times n_2 \times n_3$, where n_1, n_2 and n_3 are the number of linguistic labels of the three input variables.

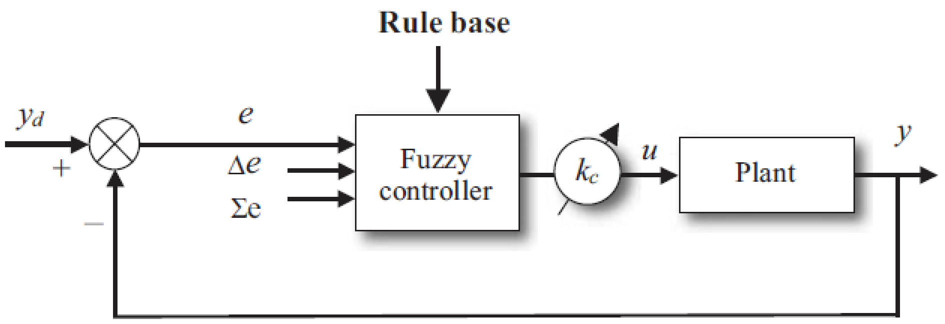


Figure 3.31 PID-type FLC with error, change of error and sum of error

PID-like FLC

- Generally, a **PD-type two-term fuzzy controller cannot eliminate steady-state error** whereas a **PI-type two-term fuzzy controller can eliminate steady-state error but it has a slower response due to the integral term in the control variable.**
- In order to meet the design criteria of fast rise time, minimum overshoot, shorter settling time and zero steady-state error, a further option is to develop a PID-type FLC which enables fast rise time, smaller overshoot and settling time from the PD part and minimum steady-state error from the PI part of the PID controller.
- The generic fuzzy PID controller is a four-dimensional (three-input single-output) fuzzy system. The basic idea of a PID controller is to choose the control law by considering the error *e*, change of error *delta* *e* and integral of error or sum of error.

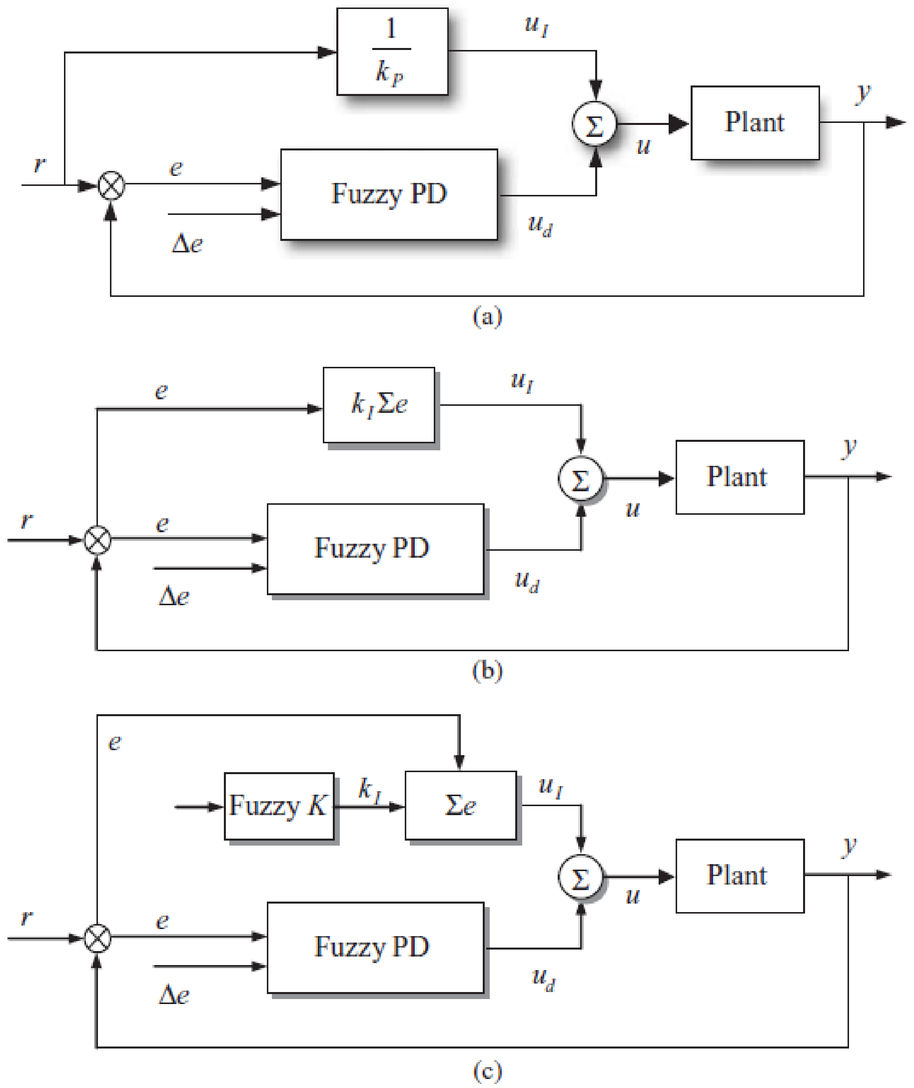


Figure 3.32 Different PID-type FLC configurations. (a) Fuzzy PD with steady-state gain control; (b) Fuzzy PD with integral action control; (c) Fuzzy PD with fuzzified k_I

Modular Fuzzy Controller

A modular structure of FLCs with minimum number of input/output variables can reduce the number of rules.

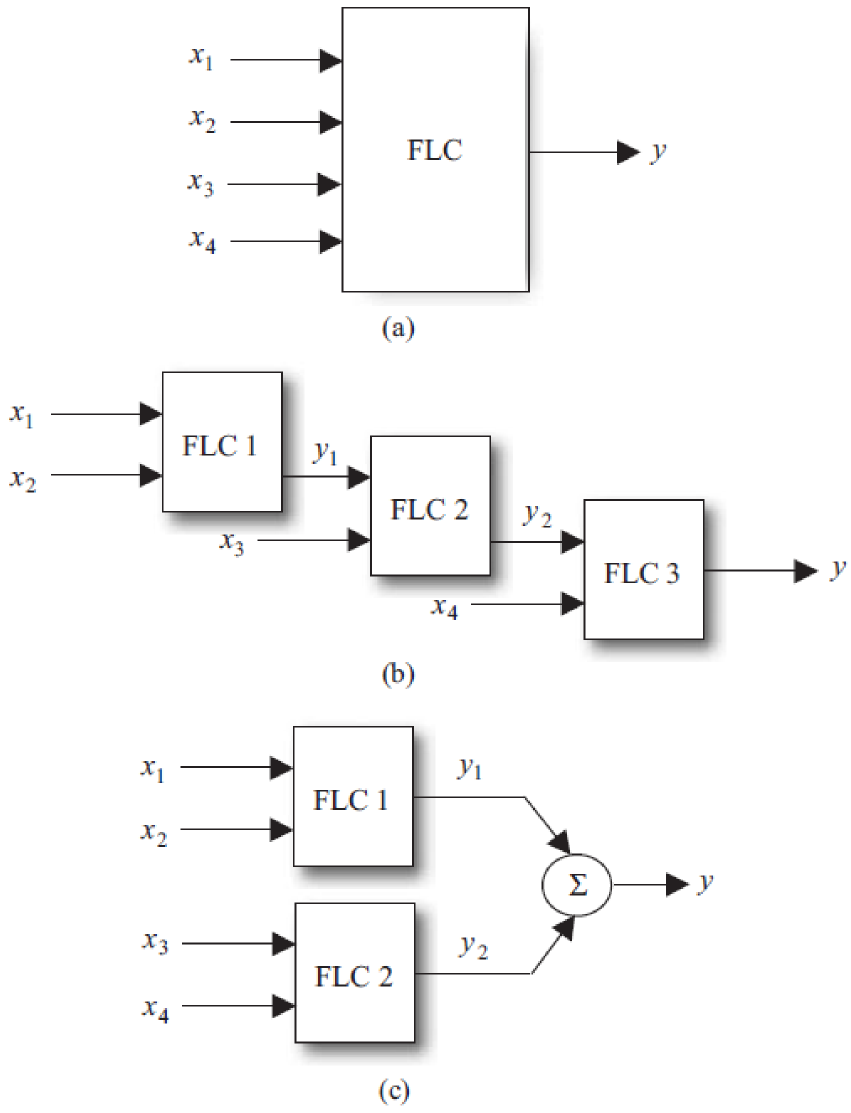
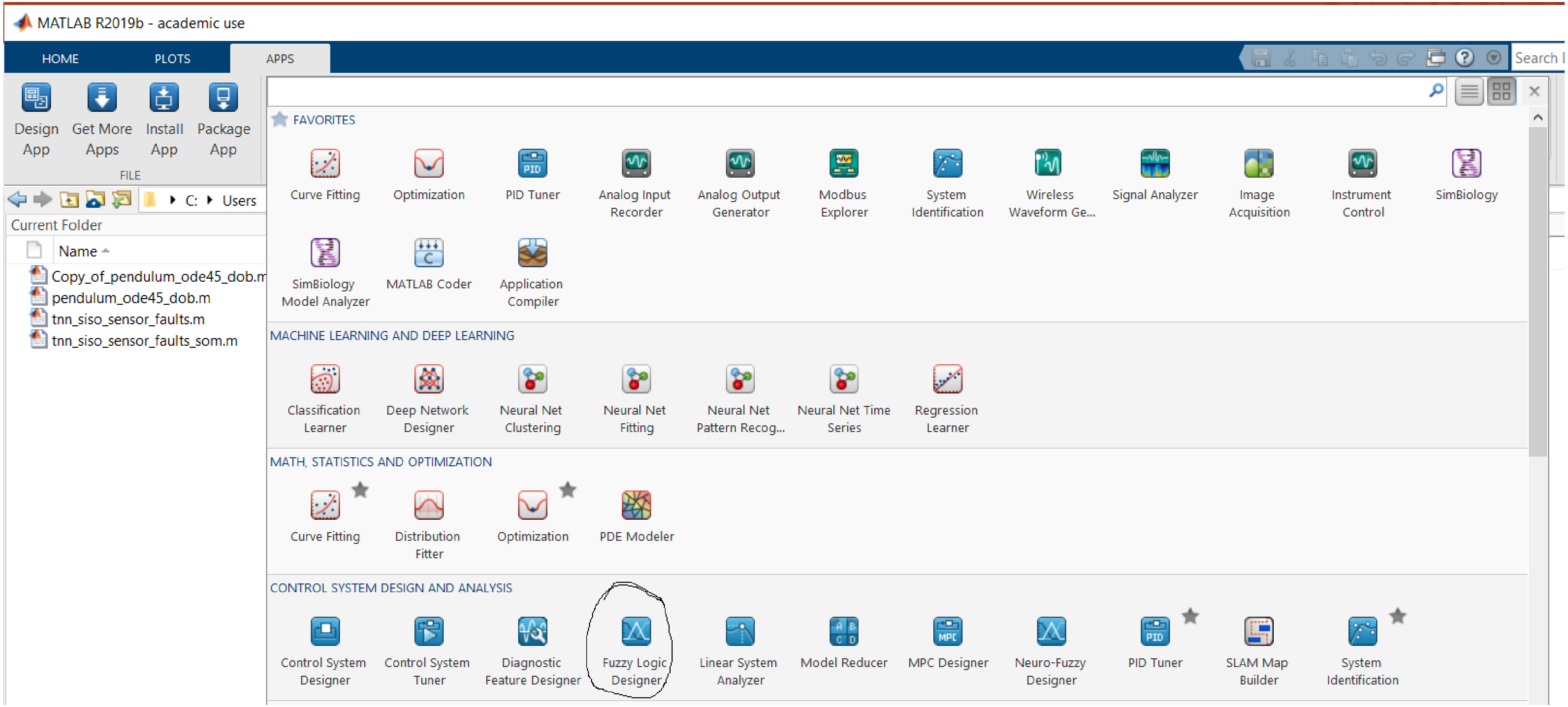


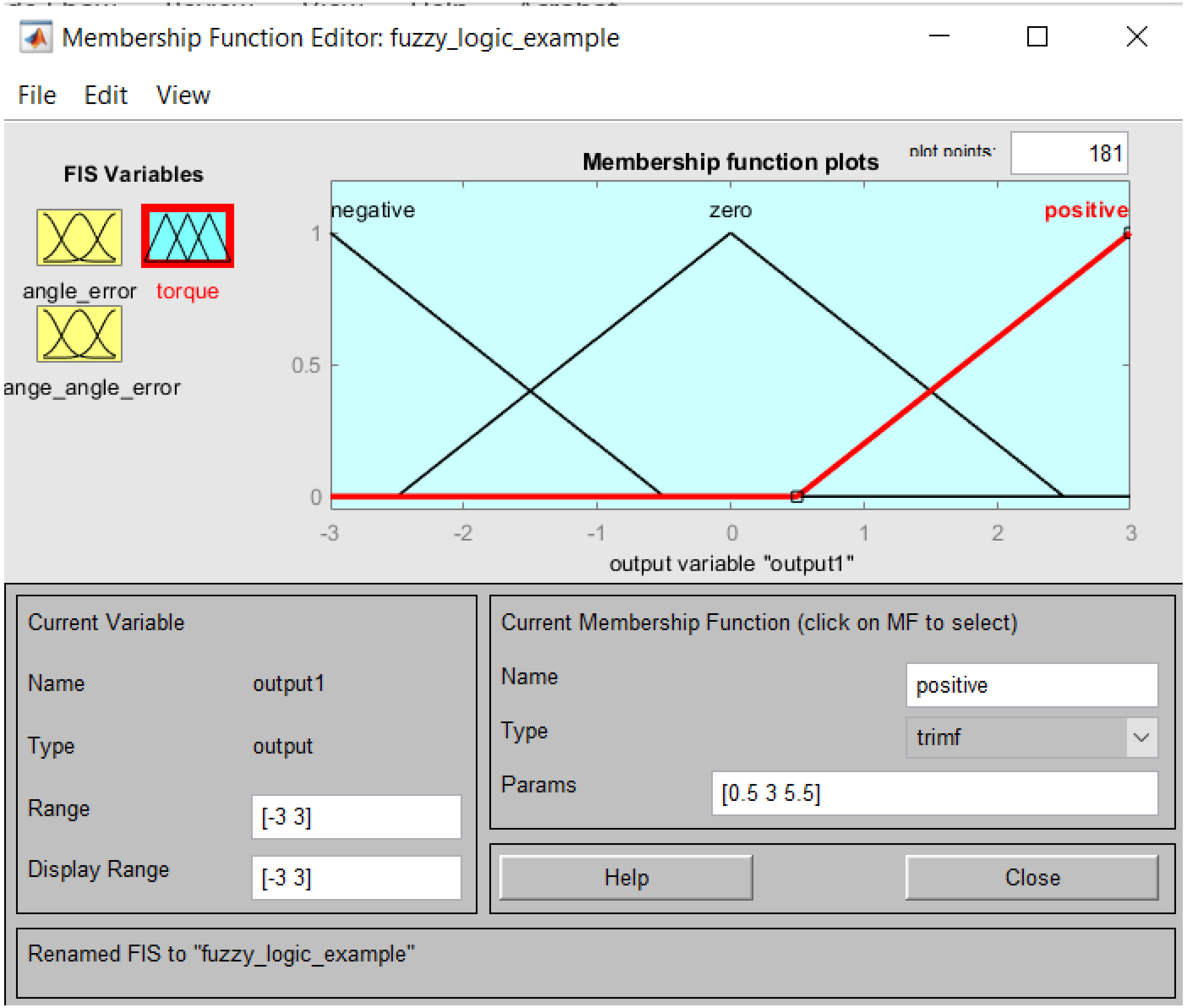
Figure 3.35 Decomposition and modular design of fuzzy controller. (a) Classical monolithic controller; (b) Hierarchical combination of modular FLCs; (c) Parallel combination of modular FLCs

Fuzzy logic MATLAB Toolbox



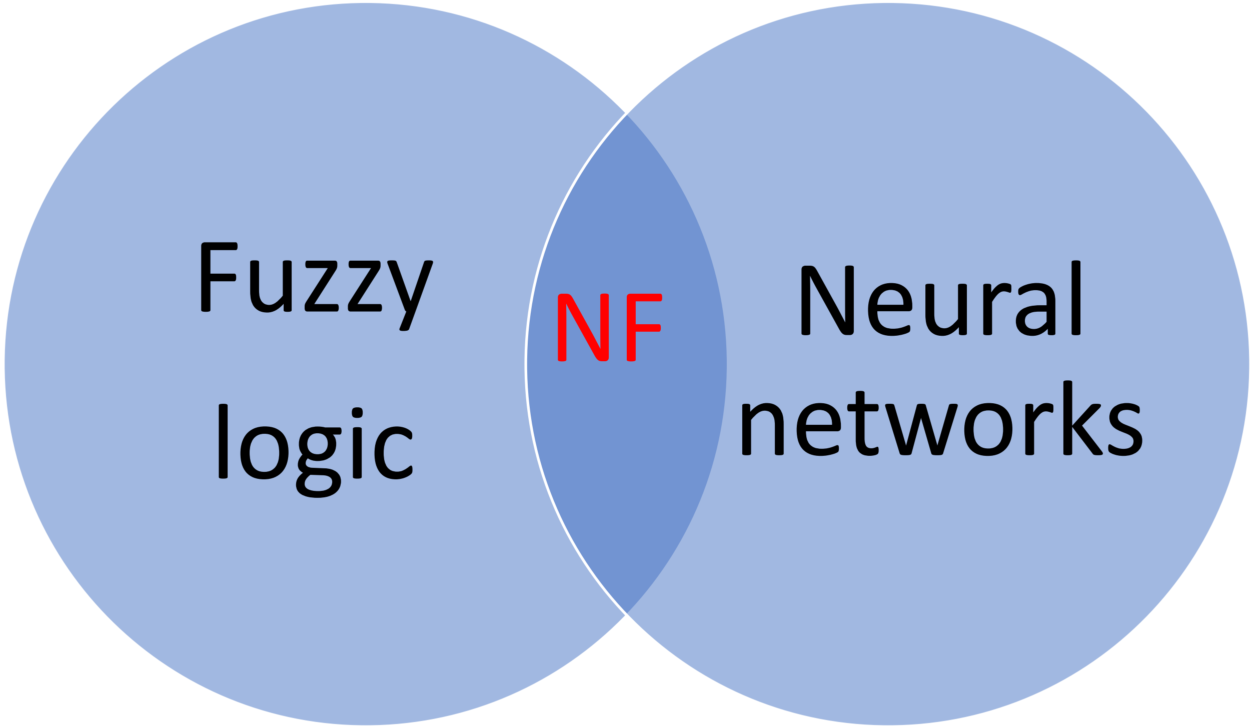
https://www.mathworks.com/products/fuzzy-logic.html

Fuzzy logic MATLAB Toolbox



Introduction

- Neuro-fuzzy (NF) systems include fuzzy logic and neural networks.
- Fuzzy logic systems try to emulate human-like reasoning using linguistic expression
- Neural networks try to emulate the human brain-like learning and storing information on a purely experiential basis.
- A neuro-fuzzy system finds the parameters of a fuzzy system by means of learning methods obtained from neural networks.
- The most important reason for combining neural networks with fuzzy systems is their learning capability.
- Such a combination should be able to learn linguistic rules and/or membership functions or tune existing ones.



Introduction

Learning in NF systems means:

- Creating a rule base,
- Adjusting MFs from scratch and
- Determining other system parameters.

Table 10.1 Comparison of neural and fuzzy systems

Neural networks (NN)	Fuzzy systems (FS)
• No mathematical model required. • Acquire knowledge usually from samples and knowledge is encoded into the network structure. • Supervised and unsupervised learning algorithms available. • Rules cannot be extracted. • Capable of learning from experiential data.	• No mathematical model required. • Acquire knowledge from domain experts and knowledge is represented by rule base. • No learning algorithms available but simple implementation possible. • Rules must be available. • Capable of working without much *a priori* information.

Neuro-fuzzy systems types

- Cooperative neuro-fuzzy systems,

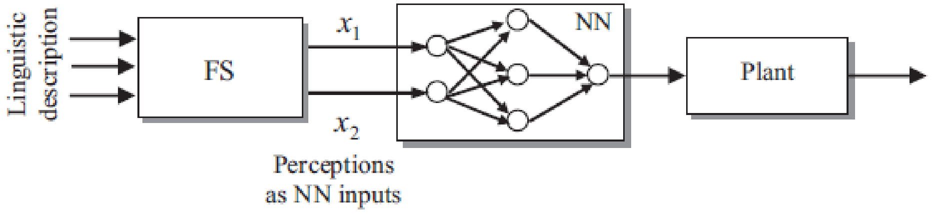


Figure 10.1 Cascade combination of cooperative FS-NN system

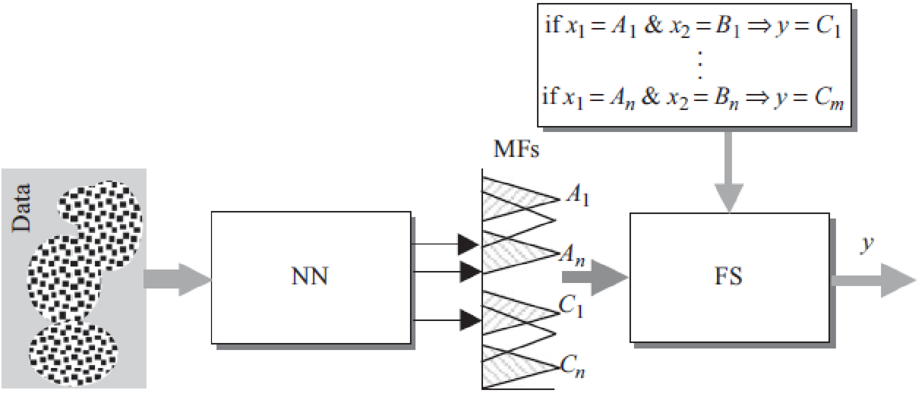


Figure 10.5 NN learning MFs' parameters from available data

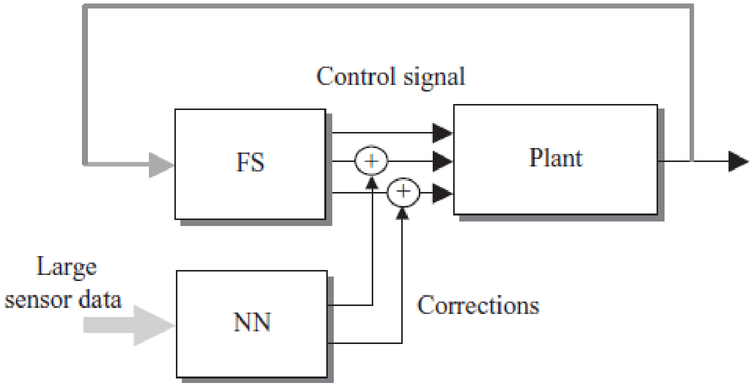


Figure 10.4 Correcting output mechanism

Neuro-fuzzy systems types

- Concurrent neuro-fuzzy systems and

Fuzzy systems and neural networks can work in parallel for a plant without mutual cooperation among themselves.

- For example, some Japanese air conditioners use an FS to prevent a compressor from freezing in the winter and use an NN to estimate index parameters of comfort, known as predictive mean vote (PMV).
- PMV can be defined as a function of room temperature, mean radiant temperature, relative air velocity, humidity, thermal resistance of users' clothing, metabolic rate.
- Some of the PMV parameters cannot be measured using sensors, e.g., thermal resistance of clothing and metabolic rate.

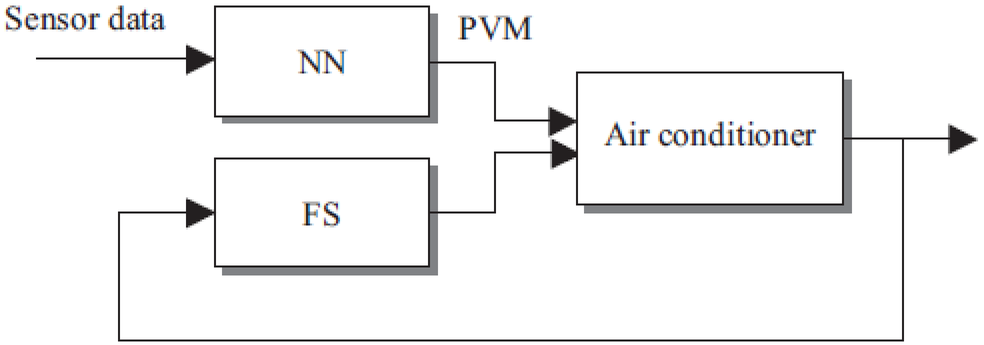


Figure 10.13 FS and NN working concurrently on the same plant

PVM = {room temperature, mean radiant temperature, relative air velocity, humidity, thermal resistance of users' clothing, metabolic rate}

Sensor data = {room temperature, time differential of room temperature, outdoor air temperature, air flow, setting temperature, direction of air flow}

Neuro-fuzzy systems types

- Hybrid neuro-fuzzy systems.

- In any fuzzy system, inferencing using the rule base and defuzzification using different methods such as centre of gravity are the most time-consuming part.
- The idea of a hybrid approach is to interpret a fuzzy system in terms of a neural network.
- The strategy adopted here with a neuro-fuzzy system is, firstly, to replace the rule base with a neural network so that the inference processing is simplified and secondly, to find the parameters of a fuzzy system by means of learning methods obtained from neural networks.
- A common way to apply a learning algorithm to a fuzzy system is to represent it in a special neural-network-like architecture so that a learning algorithm, such as backpropagation, can be used to train the system.

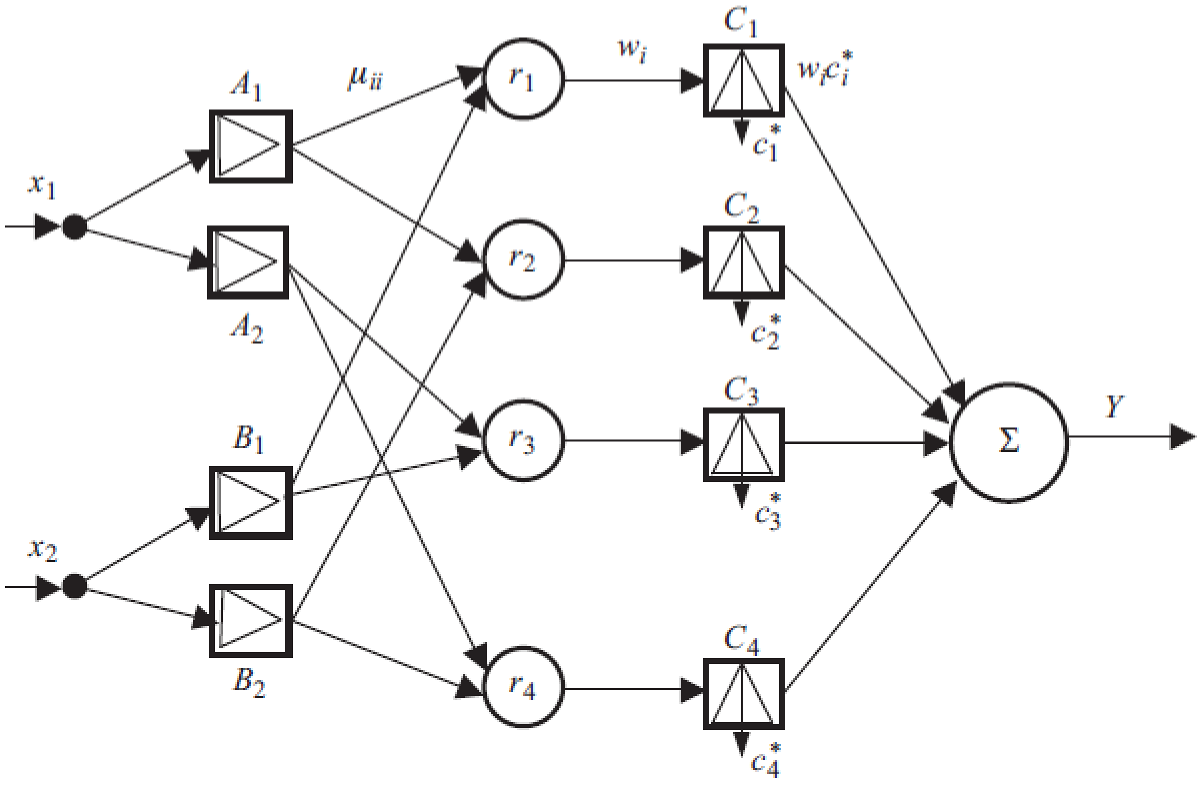


Figure 10.14 Hybrid Mamdani-type fuzzy-neural system

The ANFIS System

- Adaptive Network-based Fuzzy Inference System
- Neuro-fuzzy system that can identify parameters by using supervised learning methods
- Sugeno-type fuzzy system with learning capabilities
- First order model

IF x is A_1 AND y is B_2 THEN $f_1 = p_1 x + q_1 y + r_1$
IF x is A_2 AND y is B_2 THEN $f_2 = p_2 x + q_2 y + r_2$

The reasoning mechanism for this model is:

$$f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \bar{w}_1 + \bar{w}_2.$$

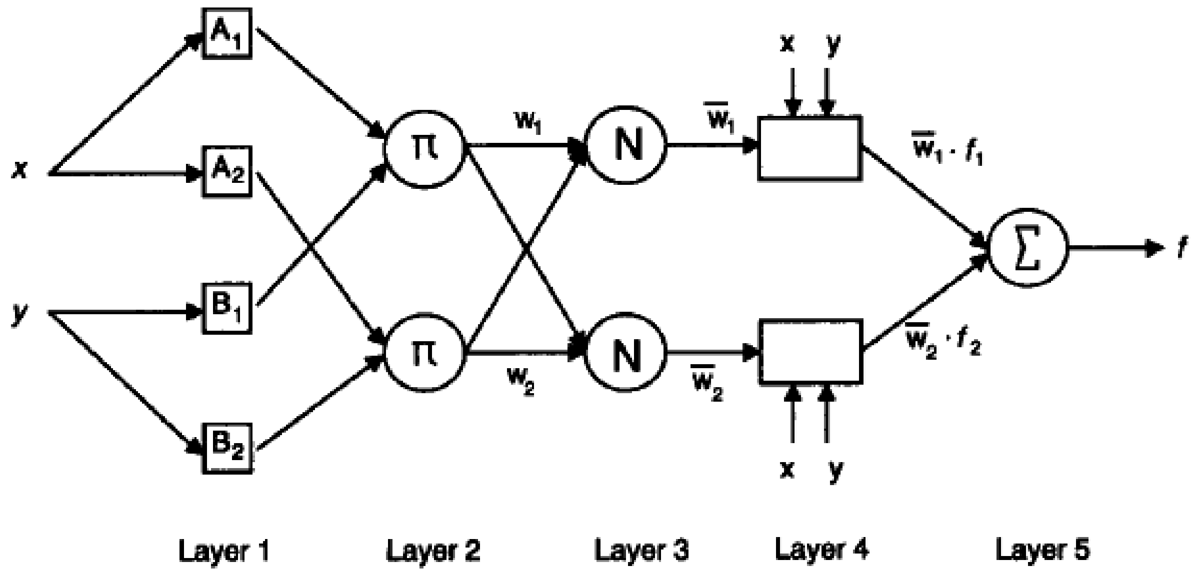
- Nodes have the same function for a given layer but are different from one layer to the next

ANFIS System

IF x is A_1 AND y is B_2 THEN $f_1 = p_1 x + q_1 y + r_1$
IF x is A_2 AND y is B_2 THEN $f_2 = p_2 x + q_2 y + r_2$

The reasoning mechanism for this model is:

$$f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \bar{w}_1 + \bar{w}_2.$$



ANFIS System

- Learning algorithm is a hybrid supervised method based on gradient descent and Least-squares.
- Forward phase: signals travel up to layer 4 and the relevant parameters are fitted by least squares
- Backward phase: the error signals travel backward and the premise parameters are updated as in backpropagation
- Neuro-Fuzzy toolbox Matlab

ANFIS System

- Since a wide class of fuzzy controllers can be transformed into equivalent adaptive networks, ANFIS can be used for building intelligent controllers that is:
- Controllers that can reason with simple fuzzy inference and that are able to learn from experience in the ANN style