Methods of Soft Control

(Methoden der Soft-Control)

Prof. Dr. Ping Zhang

Institute for Automatic Control

WS 2017/18

Organisation of this course

Chapter 1: Introduction

Chapter 2: Fuzzy control

Chapter 3: Neural networks

Chapter 4: Evolutionary algorithms

Example of fuzzy controller

- Recall the example of fuzzy controller used to control the amount of cooling medium in a drilling machine:

Rule 1: IF $Speed = very\ low$,
THEN $Amount\ of\ Cooling\ medium = very\ little$.

Rule 2: IF $Speed = low$,
THEN $Amount\ of\ Cooling\ medium = little$.

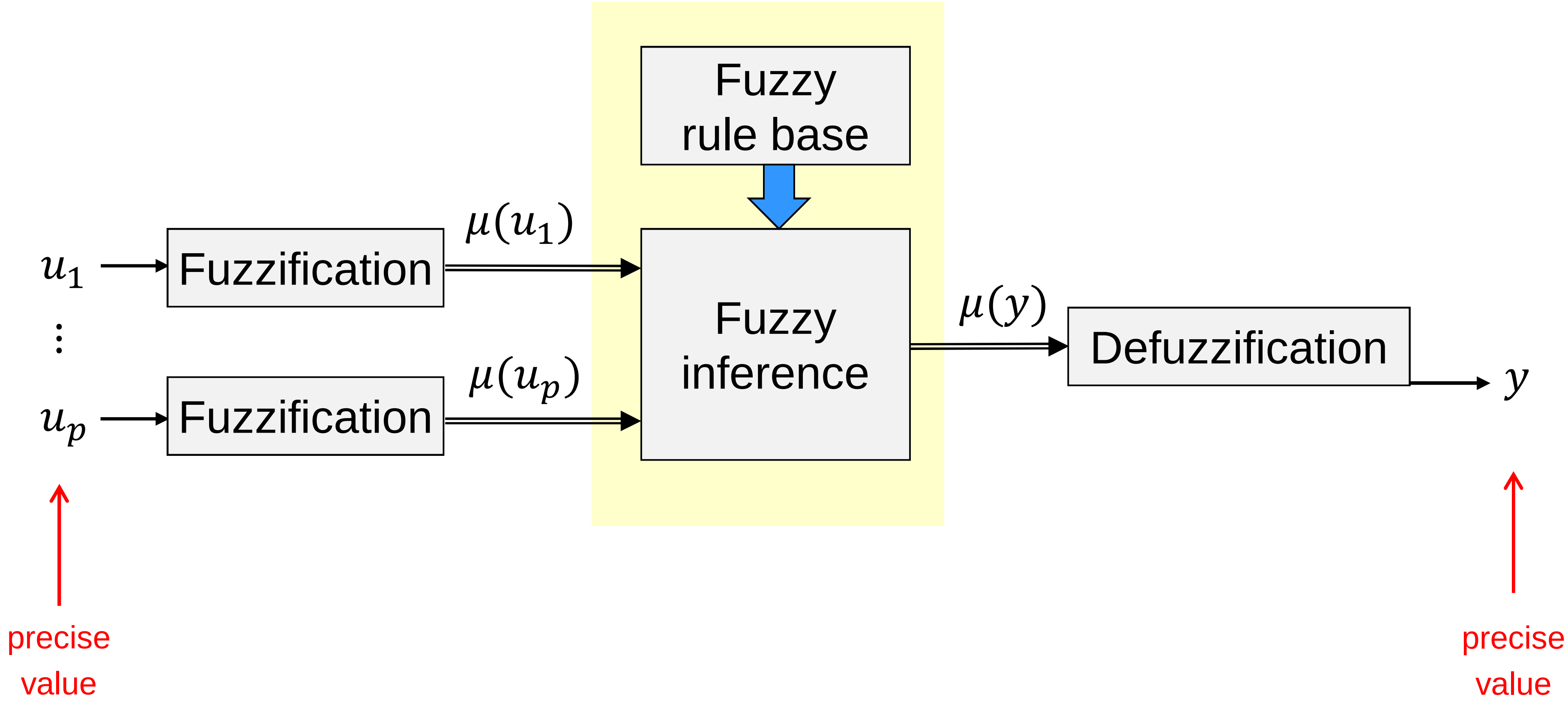
Rule 3: IF $Speed = middle$,
THEN $Amount\ of\ Cooling\ medium = normal$.

Rule 4: IF $Speed = high$,
THEN $Amount\ of\ Cooling\ medium = much$.

Rule 5: IF $Speed = very\ high$,
THEN $Amount\ of\ Cooling\ medium = very\ much$.

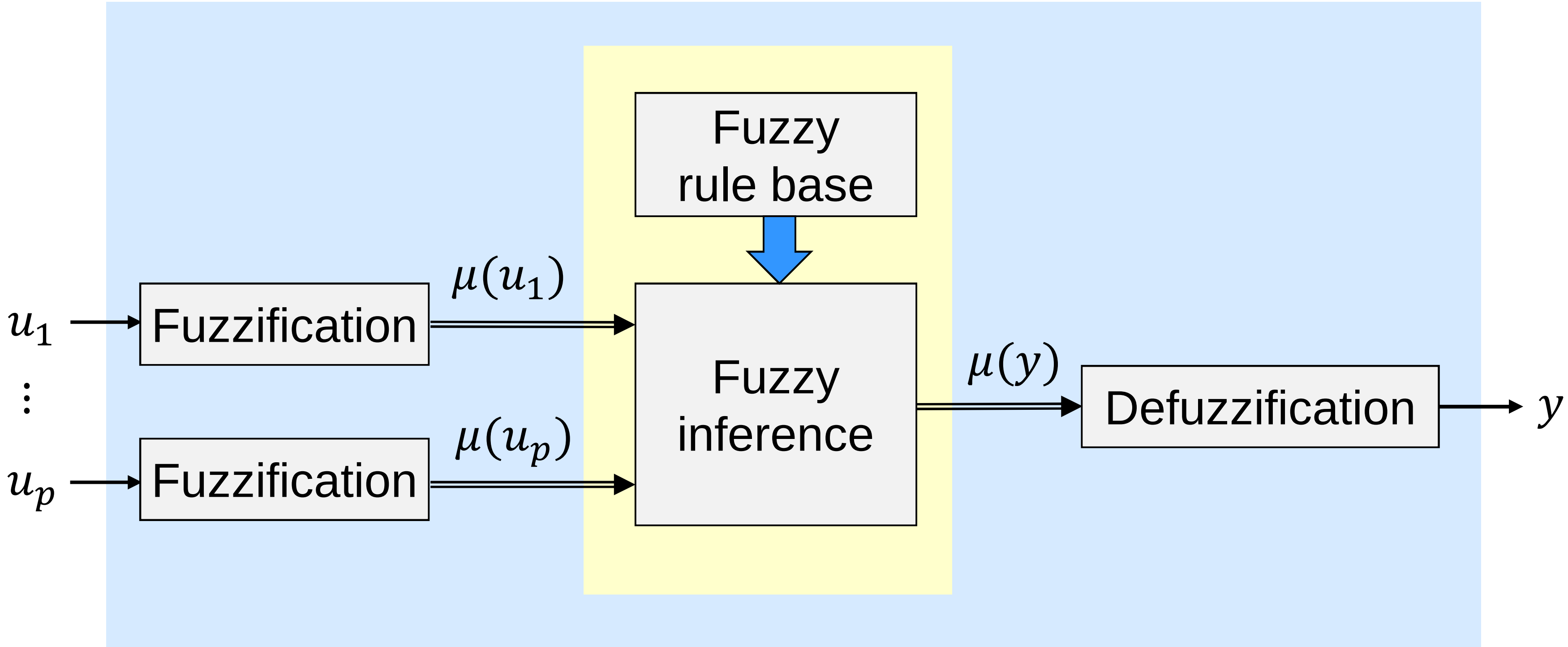
Fuzzy function block according to Mamdani

General structure of fuzzy function block



Fuzzy function block according to Mamdani

General structure of fuzzy function block



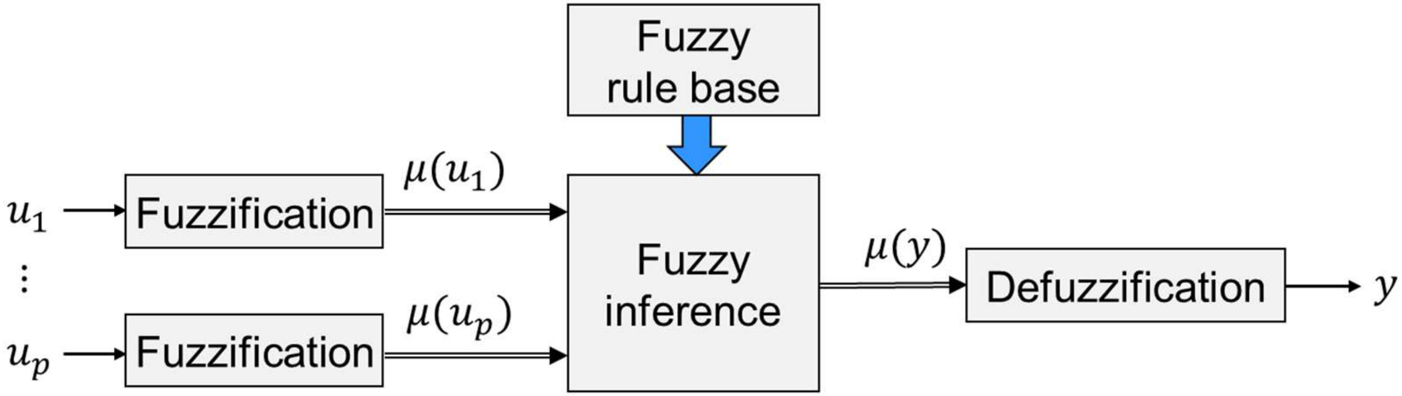
$$y = f\left(u_1, \cdots, u_p\right)$$

Fuzzy function block according to Mamdani

Important parameters in the fuzzy function block:

- The number of inputs p
- For each input u_j $(j = 1, \cdots, p)$, the number of linguistic terms n_j and the membership functions $\mu_{U_{j1}}(u_j), \cdots, \mu_{U_{jn_j}}(u_j)$
- For output y, the number of linguistic terms n_y and $\mu_{Y_1}(y), \cdots, \mu_{Y_{n_y}}(y)$
- The number n_R of fuzzy rules in the form of

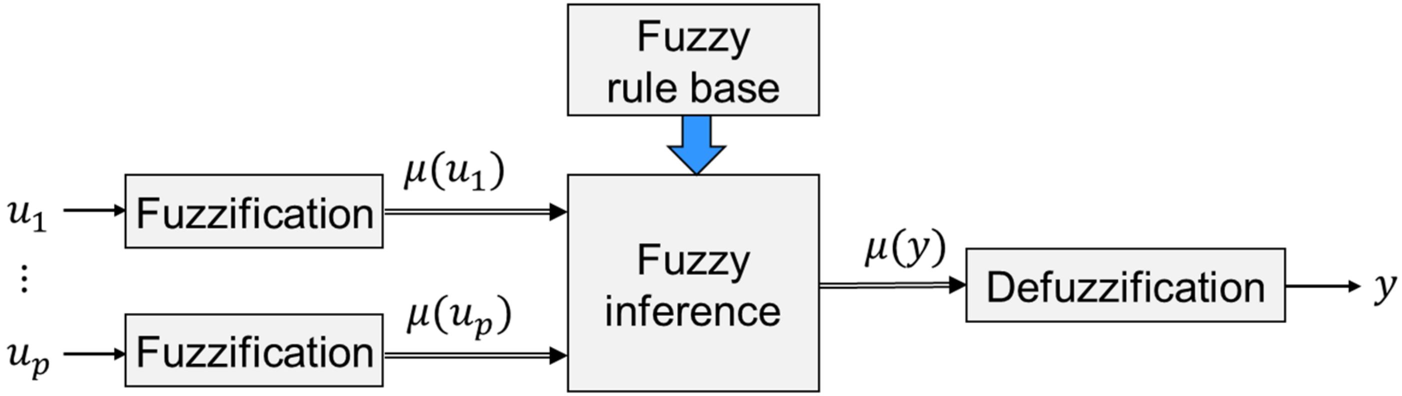
IF $U_1 = U_{1i_1}$ AND $\cdots$ AND $U_p = U_{pi_p}$, THEN $Y = Y_l$



Fuzzy function block according to Mamdani

Operators involved in the fuzzy function block:

Step	Often used operators
Fuzzy implication	min (clipping), algebraic product (scaling)
Aggregation of sub-premises (AND/OR)	min/max, algebraic product/sum
Accumulation of conclusions	max, algebraic sum
Defuzzification	CoA, MoM, SoM, LoM



Fuzzy function block according to Mamdani

Completeness of fuzzy rule base

- The rule base is called **complete**, if for any combination of the input values, there is always a rule activated.
- If the rule base is not complete, then it may happen that no rule can be activated for some combination of the input variables.
- If the completeness of the rule base can not be achieved, then it should be specified which value the output should take in such cases, for instance,
  - $y = y_{\mathrm{default}}$ with pre-specified default value
  - $y(k) = y(k-1)$, i.e. keep the old value

Design of a fuzzy function block

Design procedures

Step 1: Determine the input and output signals

Step 2: For each signal, determine the linguistic terms used to describe it

Step 3: Define the membership functions of the fuzzy sets corresponding to each linguistic term

Step 4: Generate the fuzzy rule base **according to experience and knowledge**

Step 5: Select operators for aggregation of sub-premises, fuzzy implication, accumulation of conclusions and defuzzification

Step 6: Simulation, test and tuning

Design of a fuzzy function block

Key of the design:

- **Generate fuzzy rule base**
  - in form of „IF ⋯ THEN ⋯ „ rules
  - often got by reconstructing the behaviour of experienced operators
- **Determine the shape and parameters of membership functions**
  - For each linguistic variable, the number of membership functions is often not more than 9.
  - For a signal, usually the sum of all membership functions is equal to 1 for any given signal value, i.e.

$$\sum_{i=1}^{n} \mu_{X_i}(x) = 1\,, \qquad \forall\, x$$

Fuzzy Takagi-Sugeno (T-S) dynamic model

- Fuzzy T-S models are developed to describe **complex nonlinear** MIMO dynamic systems.
- The fuzzy T-S model provides a basis for **systematic** analysis and design of controllers (using Lyapunov theory and LMI technique).

Fuzzy Takagi-Sugeno (T-S) dynamic model

- A fuzzy T-S model is a **fuzzy rule base** in the form of

Rule i: IF Z_1 is Z_{1i_1} AND ⋯ AND Z_p is Z_{pi_p}, THEN

$$x(k+1) = A_i x(k) + B_i u(k) + E_{di} d(k)$$

$$y(k) = C_i x(k) + D_i u(k) + F_{di} d(k)$$

$i = 1,2,\cdots,n_R$

$z_1,\cdots,z_p$: measurable signals,

$Z_{1i_1}(i_1 = 1,\cdots,n_1),\cdots,Z_{pi_p}(i_p = 1,\cdots,n_p)$: fuzzy sets,

n_R: the number of rules,

x: state vector, u: input vector

d: disturbance vector, y: output vector.

- Each rule represents indeed a local linear model of the system.
- The local models are smoothly connected by fuzzy membership functions.

Fuzzy Takagi-Sugeno (T-S) dynamic model

- The fuzzy T-S model can be re-written as

$$x(k+1) = \frac{\sum_{i=1}^{n_R} (A_i x(k) + B_i u(k) + E_{di} d(k)) \mu_{Ri}(z)}{\sum_{i=1}^{n_R} \mu_{Ri}(z)}$$

$$y(k) = \frac{\sum_{i=1}^{n_R} (C_i x(k) + D_i u(k) + F_{di} d(k)) \mu_{Ri}(z)}{\sum_{i=1}^{n_R} \mu_{Ri}(z)}$$

$$\mu_{Ri}(z) = \prod_{j=1}^{p} \mu_{Z_{ji_j}}(z_j)$$

or equivalently

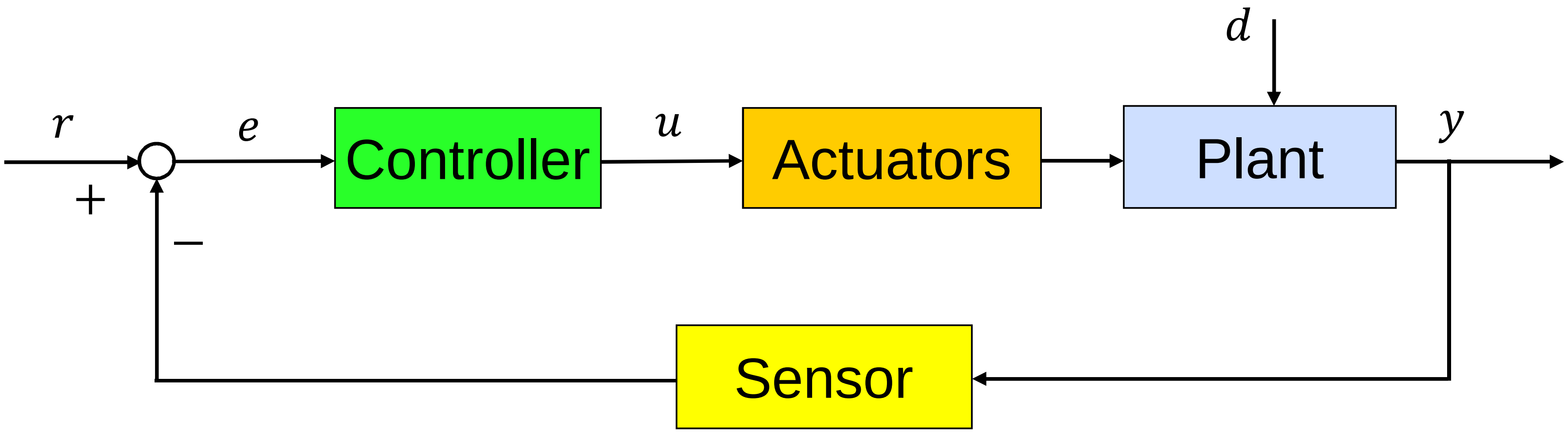
$$x(k+1) = A(\mu)x(k) + B(\mu)u(k) + E_d(\mu)d(k)$$

$$y(k) = C(\mu)x(k) + D(\mu)u(k) + F_d(\mu)d(k)$$

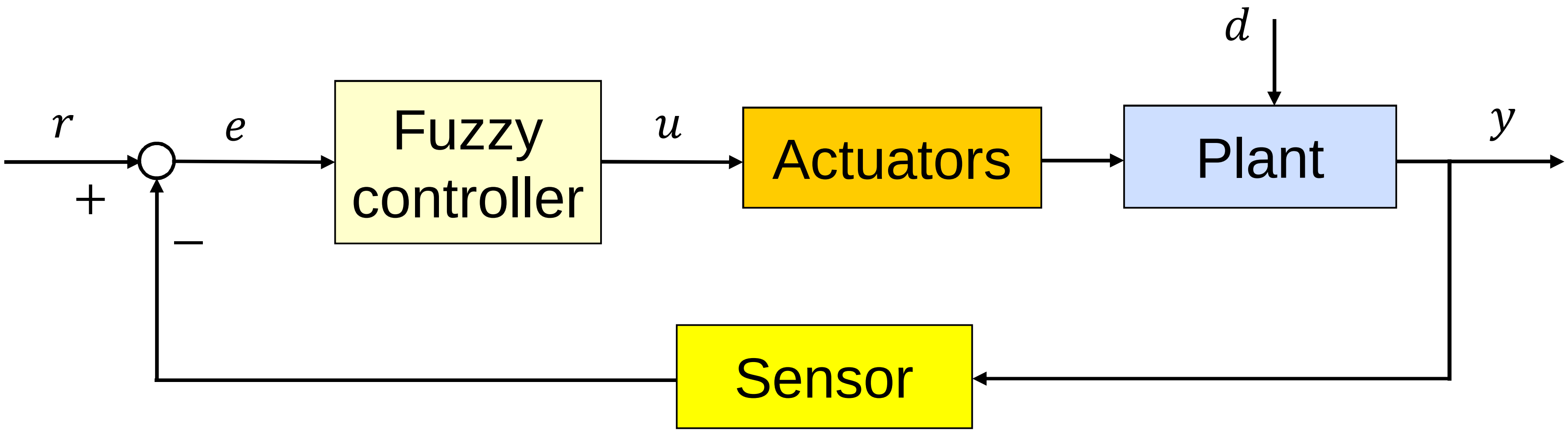
$$A(\mu) = \sum_{i=1}^{n_R} \mu_i A_i, \quad B(\mu) = \sum_{i=1}^{n_R} \mu_i B_i, \quad \cdots$$

$$\mu_i = \frac{\mu_{Ri}(z)}{\sum_{i=1}^{n_R} \mu_{Ri}(z)}, \quad \mu_{Ri}(z) = \prod_{j=1}^{p} \mu_{Z_{ji_j}}(z_j)$$

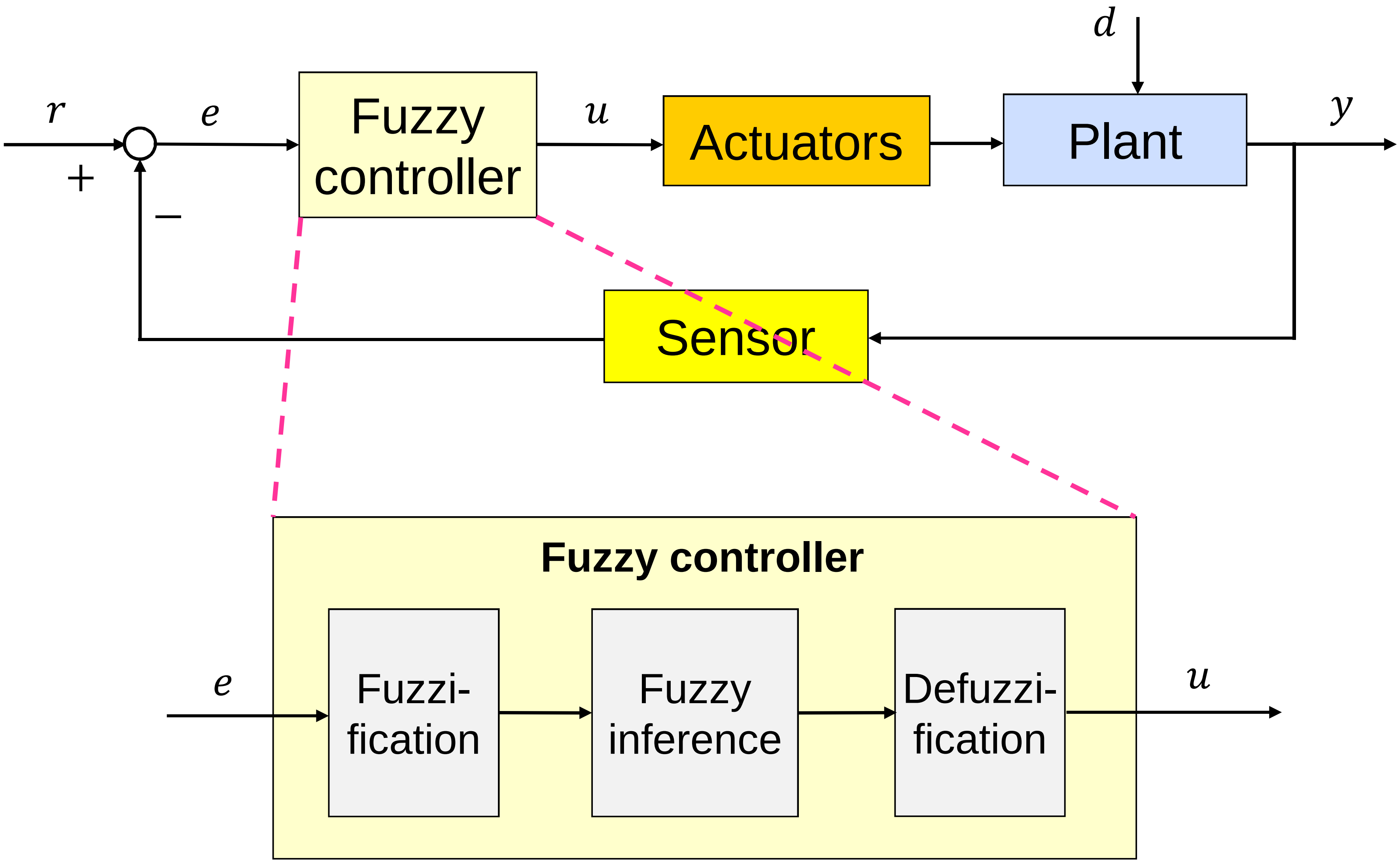
Review: Basic structure of control systems



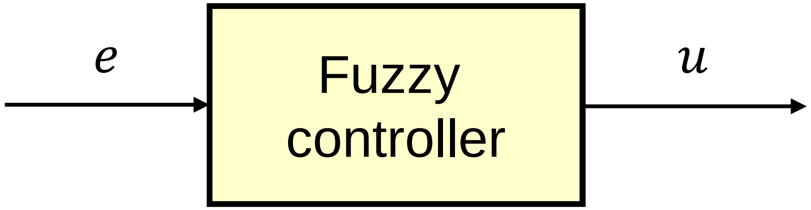
Fuzzy controller



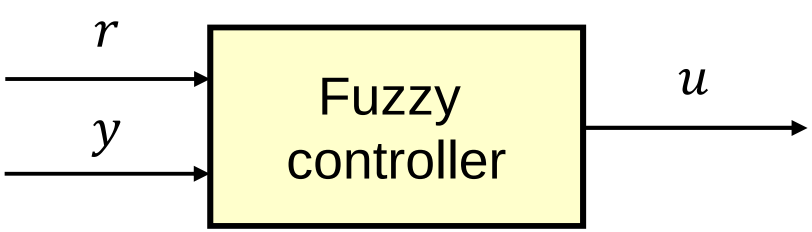
Fuzzy controller



Fuzzy P-controller



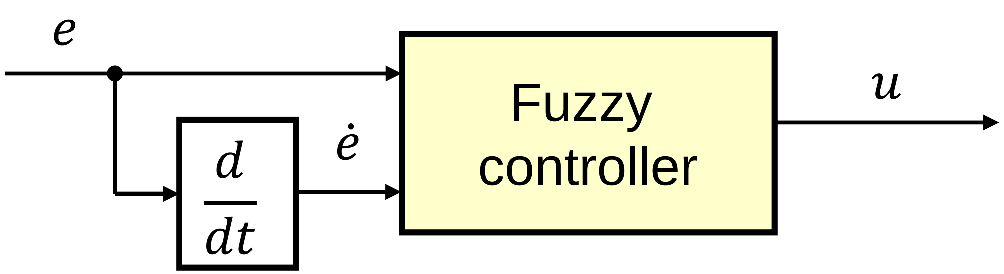
$$u = f(e)$$



$$u = f(r, y)$$

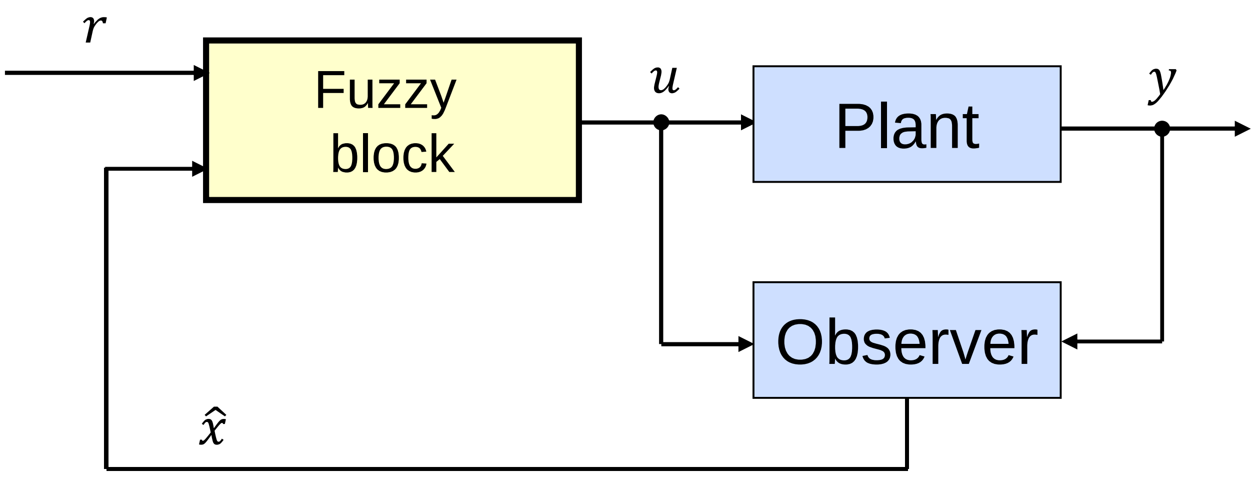
Fuzzy controller

Fuzzy PD-controller



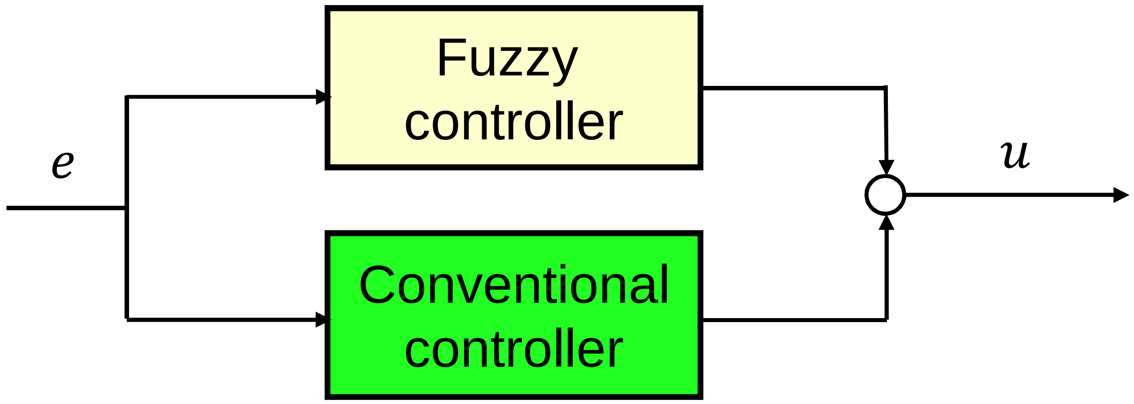
$$u = f(e, \dot{e})$$

Fuzzy state feedback controller

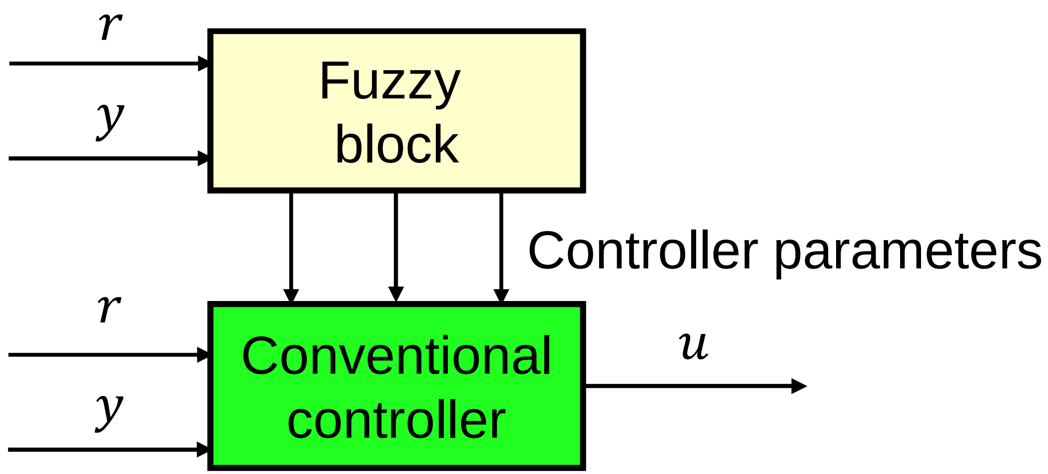


Fuzzy controller

Hybrid controller

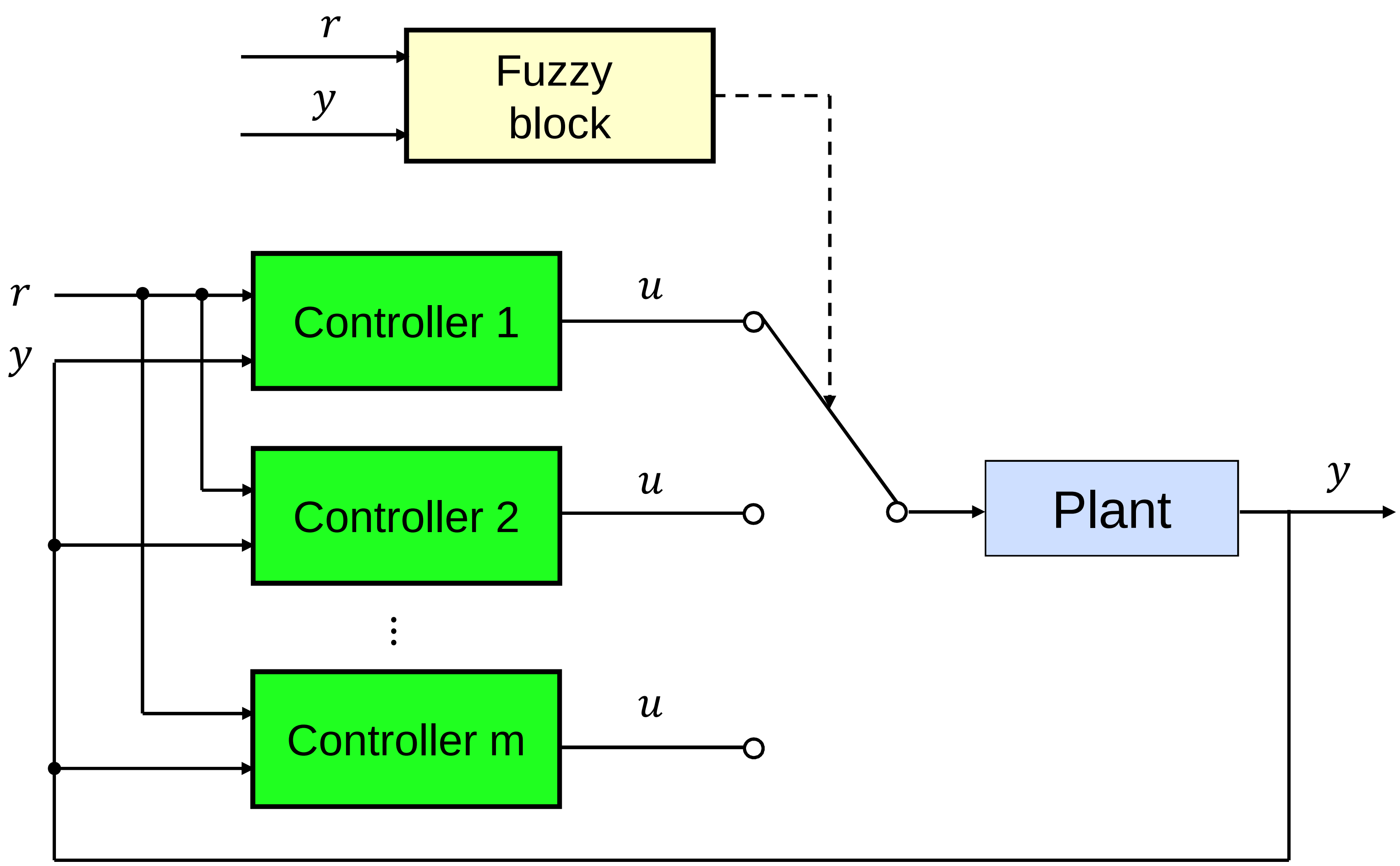


Fuzzy adaption of controller parameter



Fuzzy controller

Fuzzy switching among controllers



Design of fuzzy controllers

Case 1: Experience of operators is available

The rule base can be determined by interviewing experienced operators.

Case 2: The conventional linear controller works partly

At first approximate the conventional linear controller. Then tune the rules in the range where the conventional controller doesn‘t have a satisfactory performance.

Case 3: A good model of the plant is available

The parameters of the fuzzy controller can be optimized with the help of simulations.

Case 4: Local linear models of the plants at different working points are available

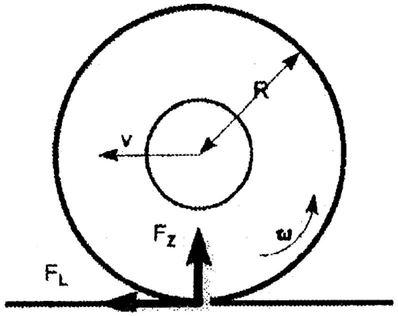
Design a conventional linear controller based on each local model. Consider fuzzy adaption of controller parameters or fuzzy switching rules among controllers. Alternatively, consider fuzzy T-S model based design.

Case 5: The plant model is not available

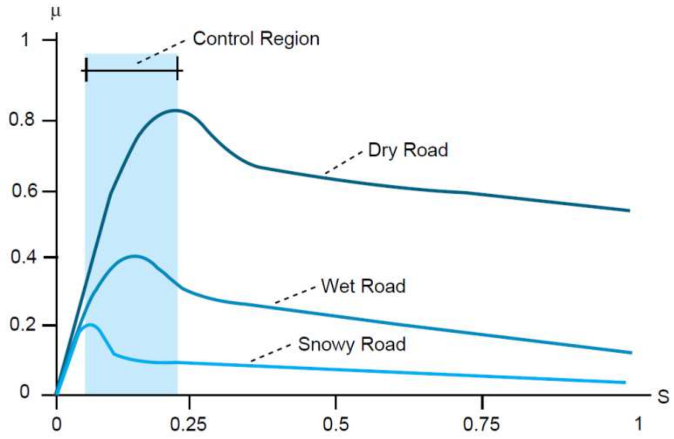
Online tuning or online optimization of fuzzy controller on the real plant may be possible. Plan it outside of the production time to avoid problems that may be caused by incorrect fuzzy controller parameters.

Example: Fuzzy control of ABS braking

Reference: Mauer, IEEEFS95



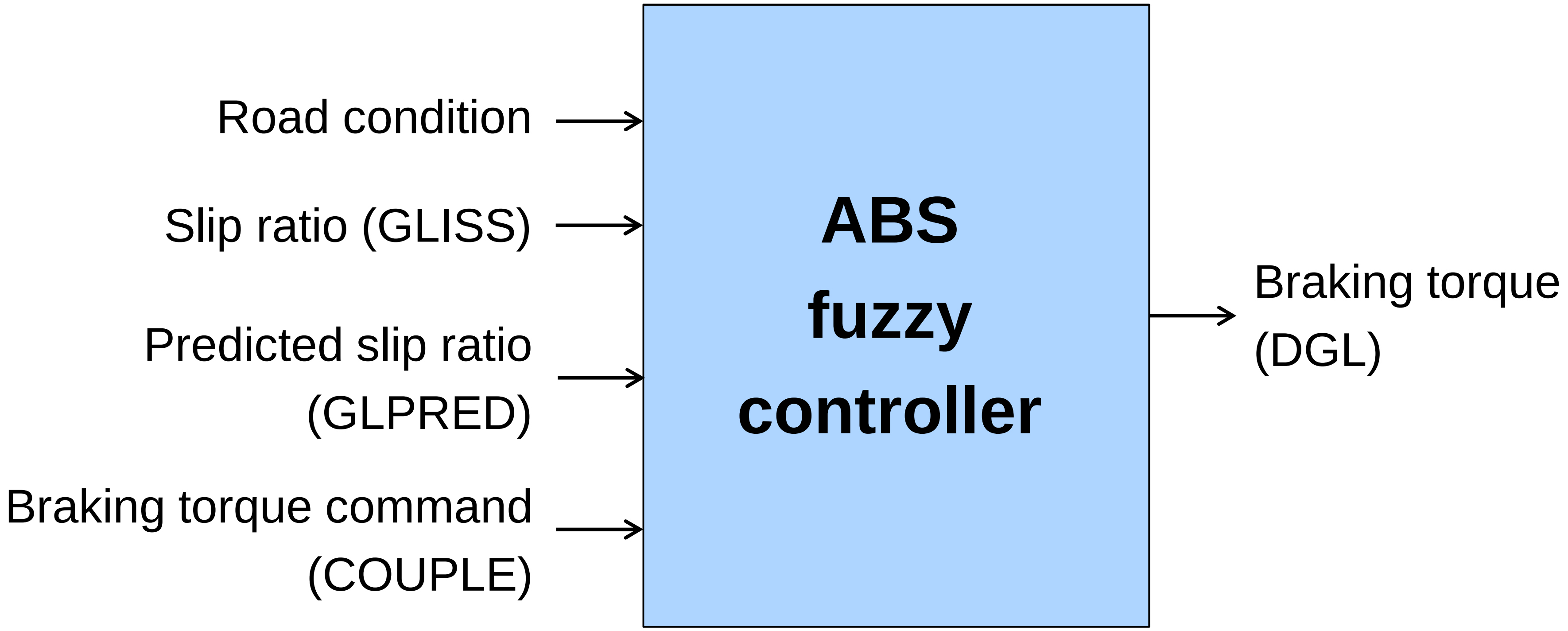
v : velocity of wheel
R : radius of wheel
$\mathbf{F_z}$: wheel load
$\mathbf{F_L}$: longitudinal force
ω : angular velocity of wheel



$$Slip = \frac{vehicle\ speed\ - wheel\ speed}{vehicle\ speed}$$

Example: Fuzzy control of ABS braking

Fuzzy controller structure



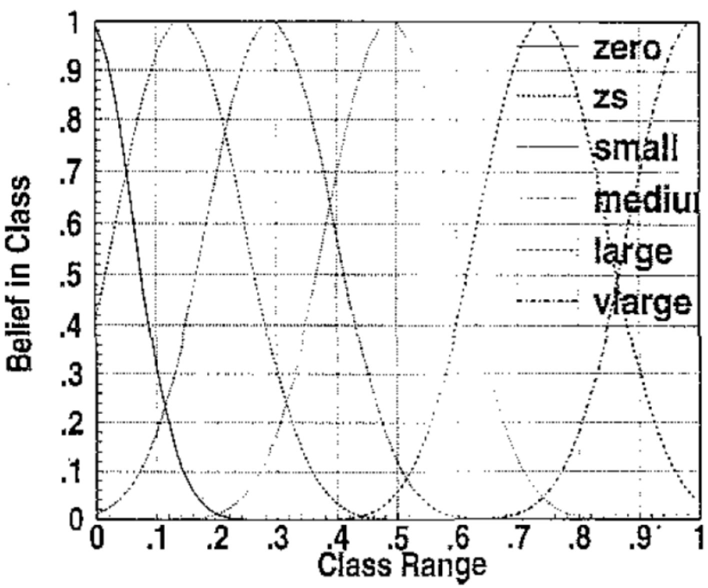
Example: Fuzzy control of ABS braking

Fuzzy rule base

```
1. RULE DRY1;
   IF DRY IS TRUE AND;
     GLPRED IS NOT VLARGE;
   THEN DGL IS LARGE;
     RULE_1;
2. RULE DRY2;
   IF GLISS IS LARGE AND;
     DRY IS TRUE AND;
     COUPLE IS LARGE;
   THEN DGL IS MEDIUM;
     RULE_2;
3. RULE DRY3;
   IF GLISS IS SMALL AND;
     DRY IS TRUE AND;
     COUPLE IS LARGE AND;
     GLPRED IS NOT VLARGE;
   THEN DGL IS LARGE;
     RULE_3;
4. RULE DRY4;
   IF GLISS IS MEDIUM AND;
     DRY IS TRUE AND;
     GLPRED IS NOT VLARGE AND;
     COUPLE IS LARGE;
   THEN DGL IS LARGE;
     RULE_4;
5. RULE ICE7;
   IF ICE IS TRUE AND;
     GLISS IS ZS AND;
     COUPLE IS ZS;
   THEN DGL IS ZS;
     RULE_7;
6. RULE ICE5;
   IF GLISS IS ZERO AND;
     ICE IS TRUE;
   THEN DGL IS SMALL;
     RULE_5;
7. RULE ICE8;
   IF GLISS IS SMALL AND;
     ICE IS TRUE;
   THEN DGL IS ZERO;
     RULE_8;
8. RULE BLOCKAGE;
   IF GLISS IS VLARGE AND;
     GLPRED IS VLARGE;
   THEN DGL IS ZERO;
     RULE_9;
9. RULE WET10;
   IF WET IS TRUE AND;
     GLISS IS ZS AND;
     GLPRED IS NOT LARGE;
   THEN DGL IS SMALL;
     RULE_10;
10. RULE WET11;
    IF WET IS TRUE AND;
      GLISS IS SMALL;
    THEN DGL IS ZS;
      RULE_11;
11. RULE WET12;
    IF WET IS TRUE AND;
      GLISS IS ZERO AND;
      GLPRED IS NOT LARGE;
    THEN DGL IS SMALL;
      RULE_12.
```

Example: Fuzzy control of ABS braking

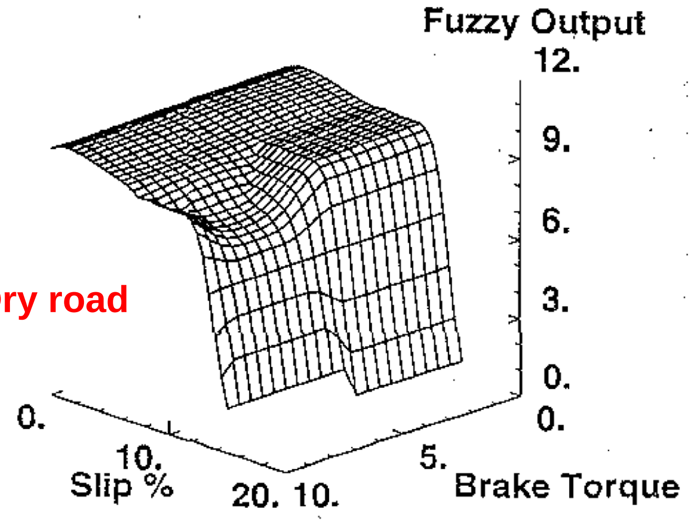
Membership functions



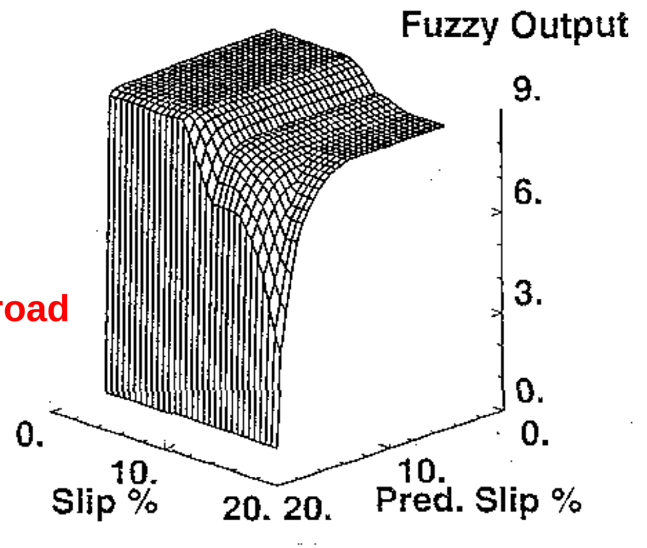
Example: Fuzzy control of ABS braking

Controller output

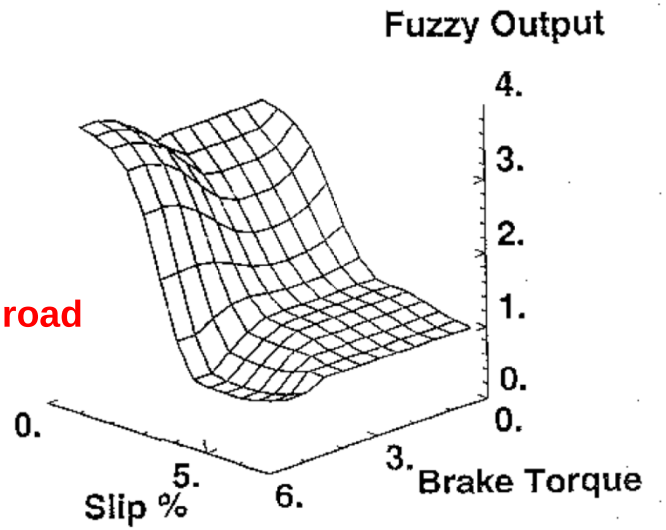
Dry road



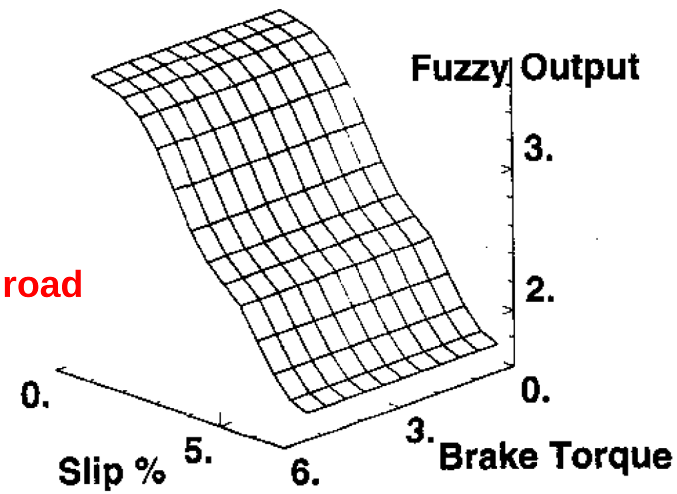
Dry road



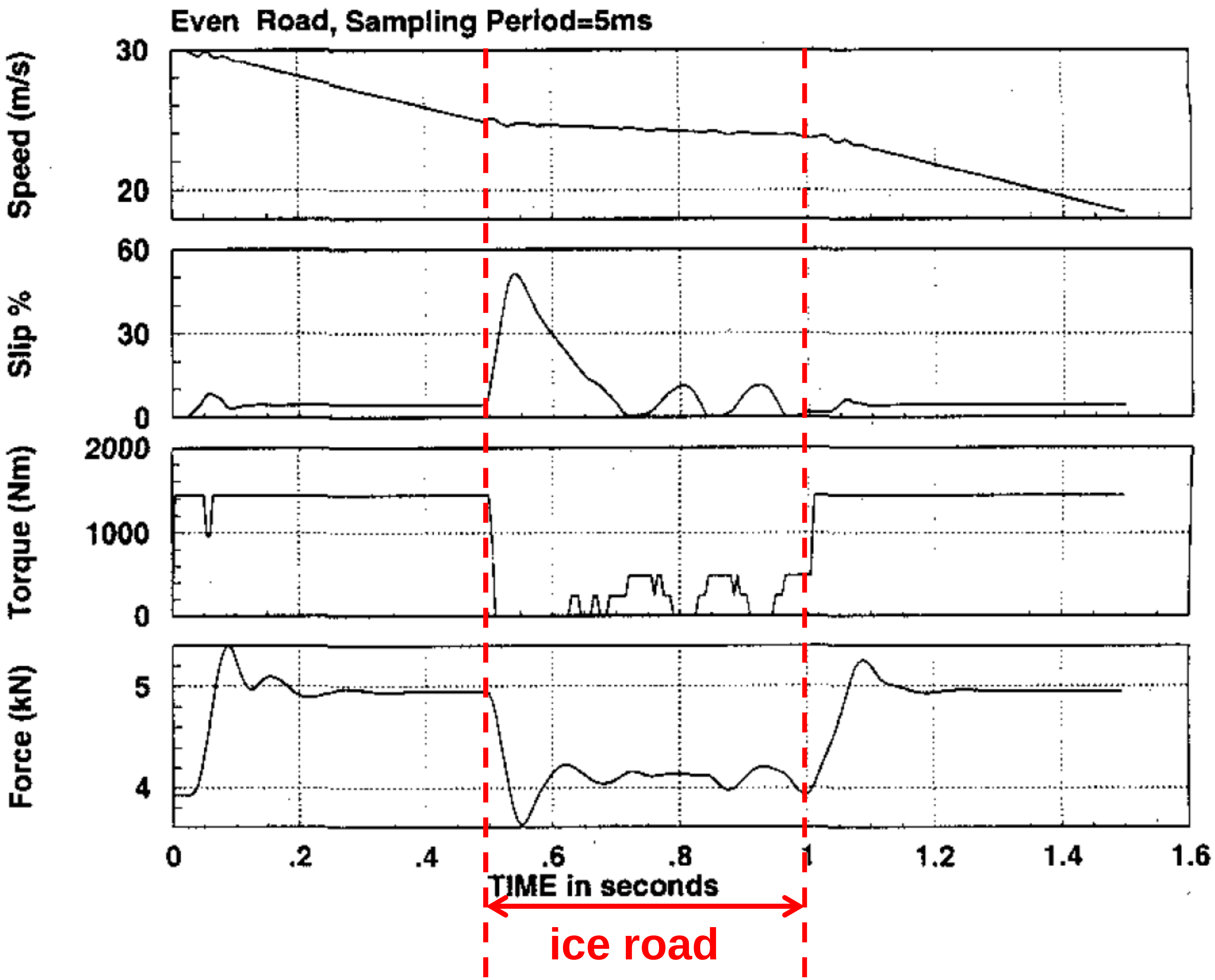
Ice road



Wet road

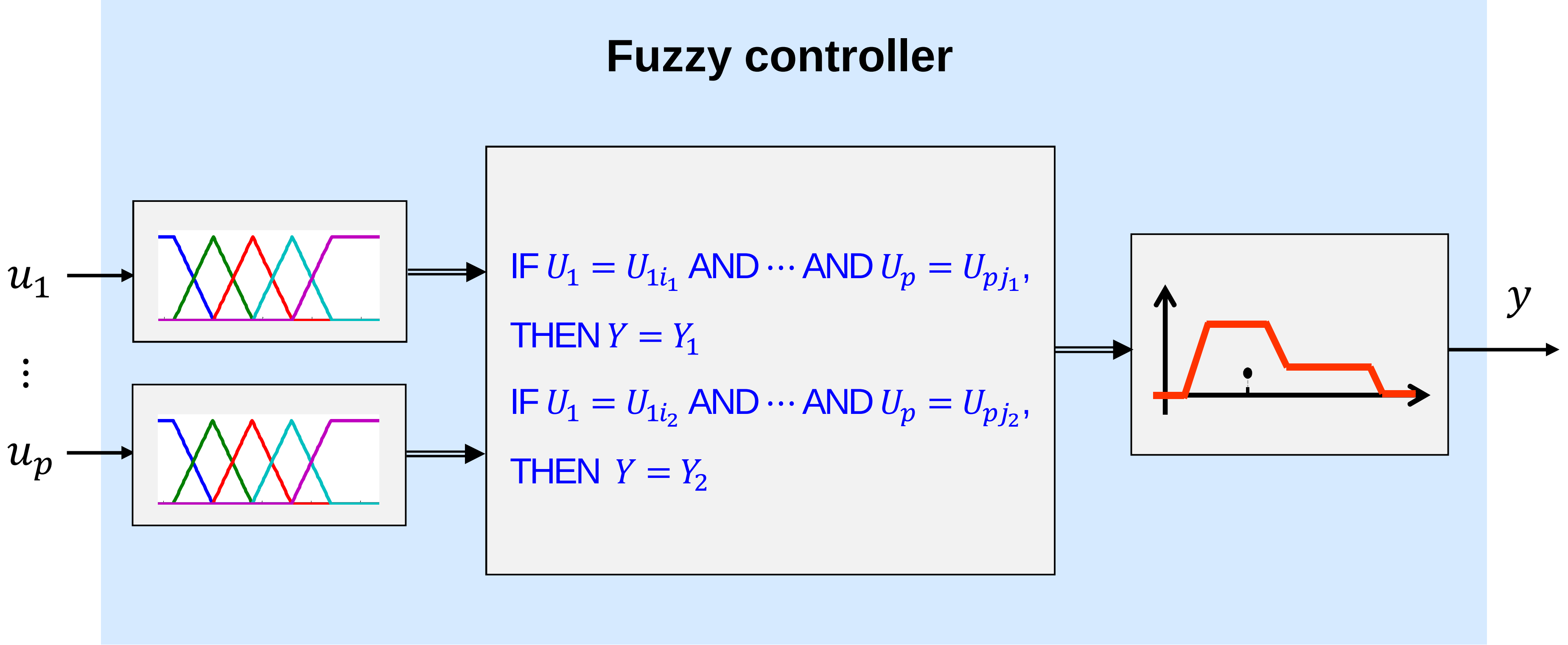


Example: Fuzzy control of ABS braking



Summary

- Fuzzy control can be applied to systems whose model is not available or whose dynamics is too complex.
- The fuzzy rules are often obtained based on experience or knowledge.



Organisation of this course

Chapter 1: Introduction

Chapter 2: Fuzzy control

Chapter 3: Neural networks

Chapter 4: Evolutionary algorithms

Chapter 3

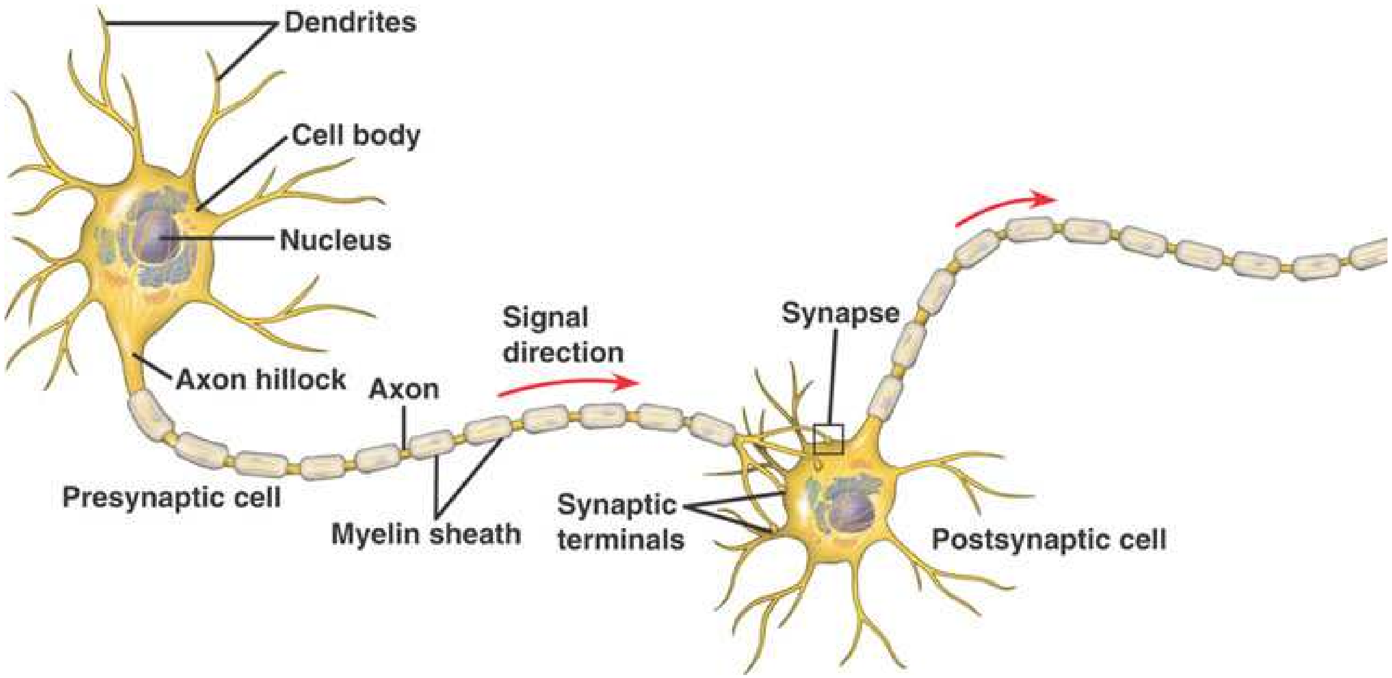
Neural Networks

Literature

- S. Haykin. Neural networks and learning maschines. Pearson, 2009.
- S. Russell, P. Norvig. Artificial Intelligence – a modern approach. Pearson, 2010.

Biological neurons

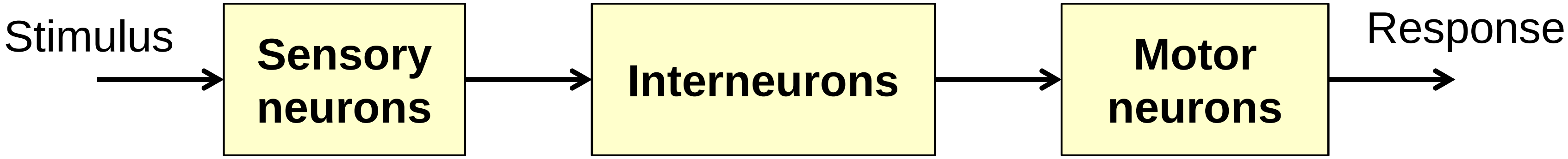
- A neuron is the basic functional unit of the nervous system.
- The **dendrites** receive impulses from other neurons and conduct them to the cell body.
- The **axon** conduct impulses from the cell body towards other neurons.
- The axons ends in **synapses**. The synapses are contact points to the next neighbouring neurons.



Biological neurons

According to the functions, the neurons can be classified into:

- **Sensory neurons**: receive sensory signals from skin, eyes, ears, tongue, etc, and send them via axons to the central nervous system.
- **Motor neurons**: conduct the commands to the muscles to make them contract or relax.
- **Interneurons (called also association neurons)**: interconnect different neurons, for instance, the neurons in the brain or in the spinal cord.

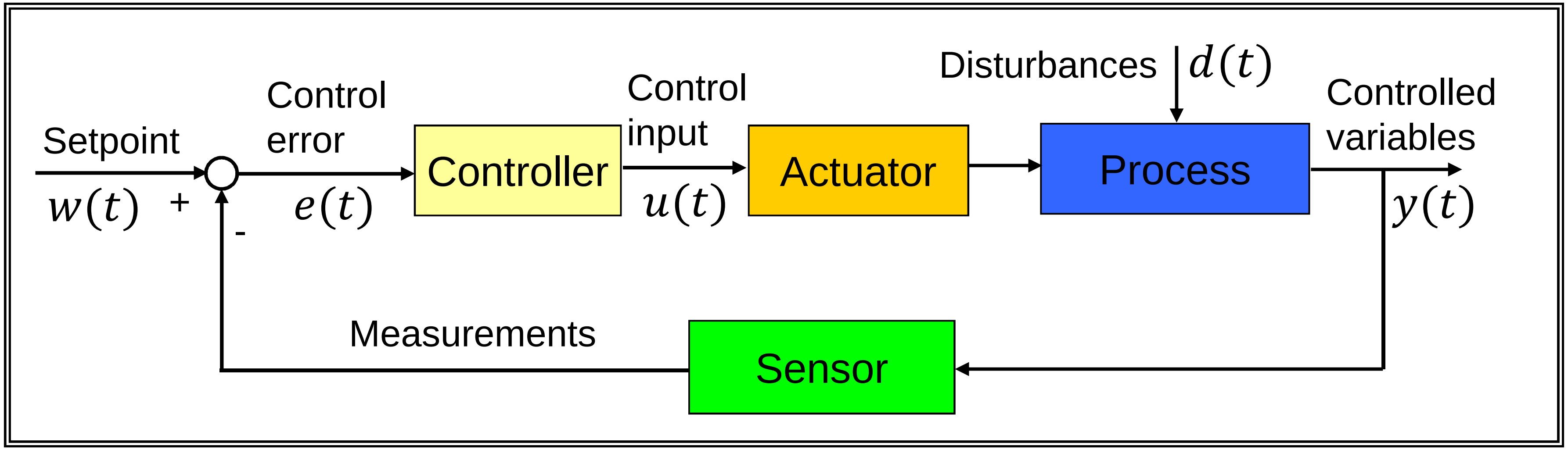


About 10^{12} neurons in the brain!

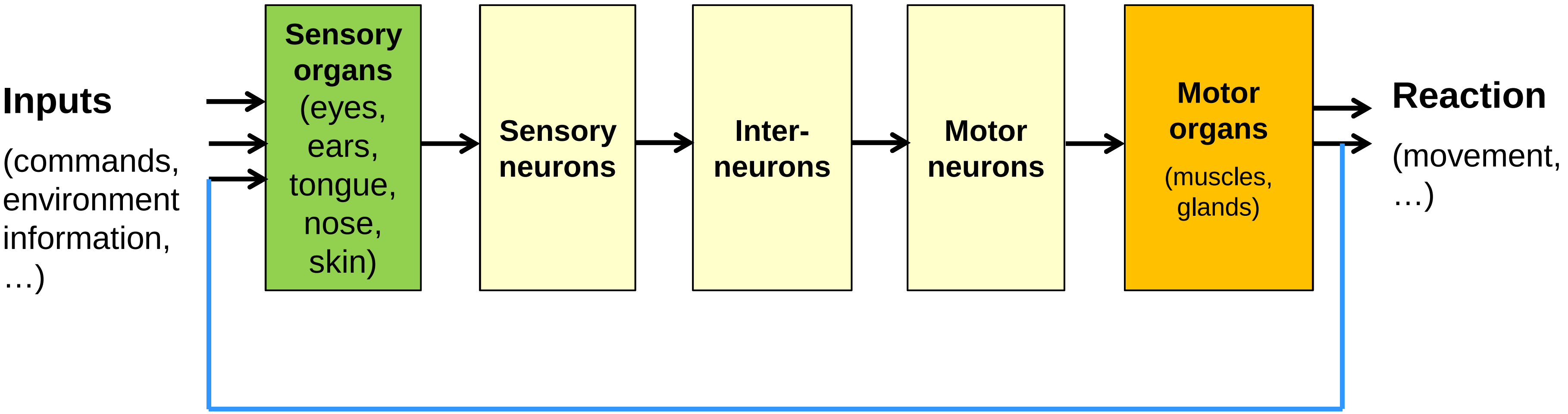
Each neuron may have between 1000 and 10,000 synapses!

Biological neurons

Recall that the general structure of control systems is as follows:



Cooperation among neurons

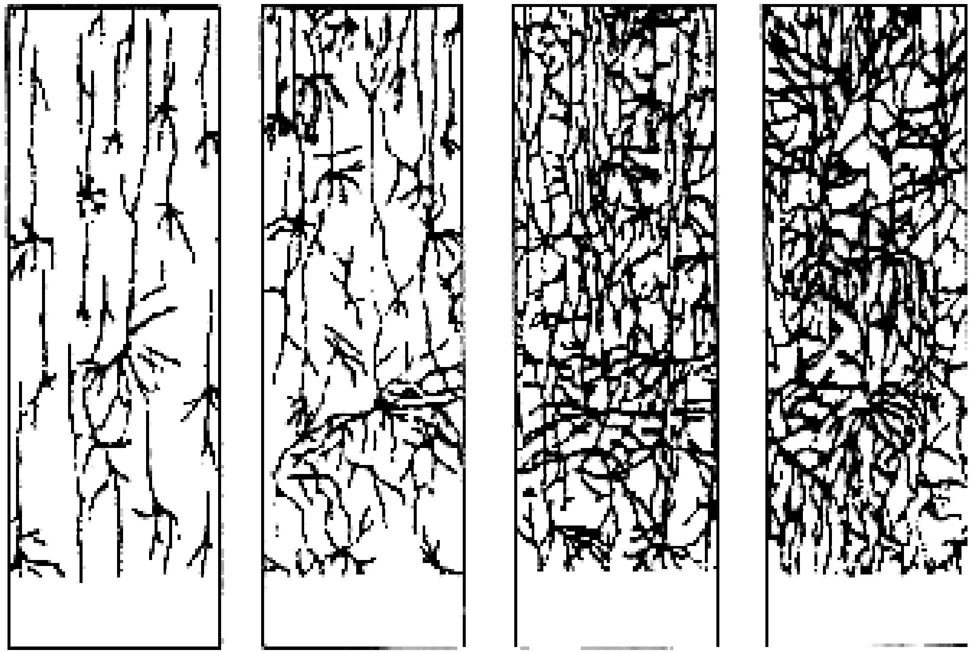


A extremely powerful control network …

Neuroplasticity

- The neurons adapt to the environment and experience!
- The creation of new synapses between neurons and the modification of existing synapses.
- So people can learn, create, but sometimes also forget …

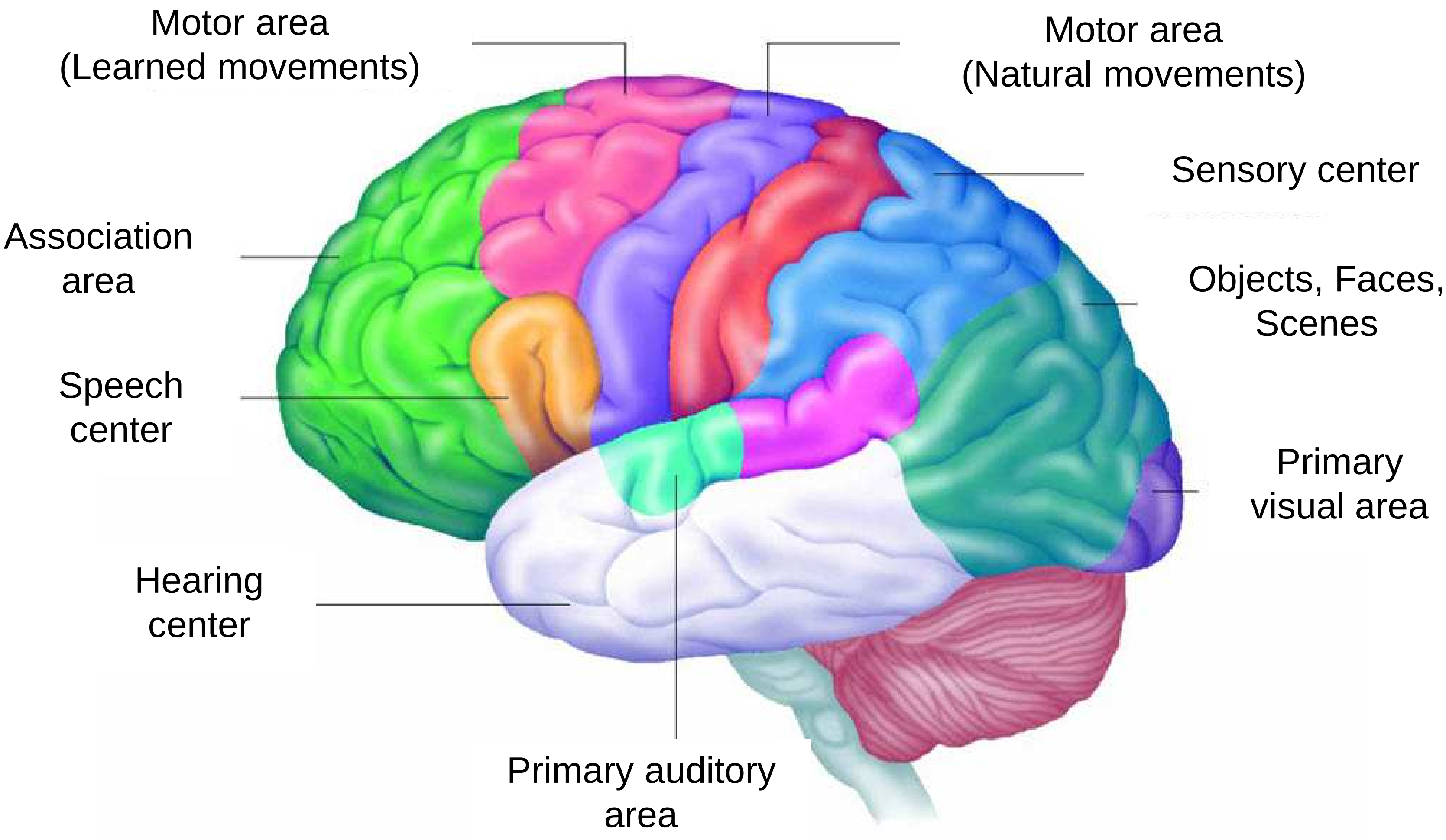
Network of neurons



At birth 3 months 15 months 2 years

The brain is a big network …

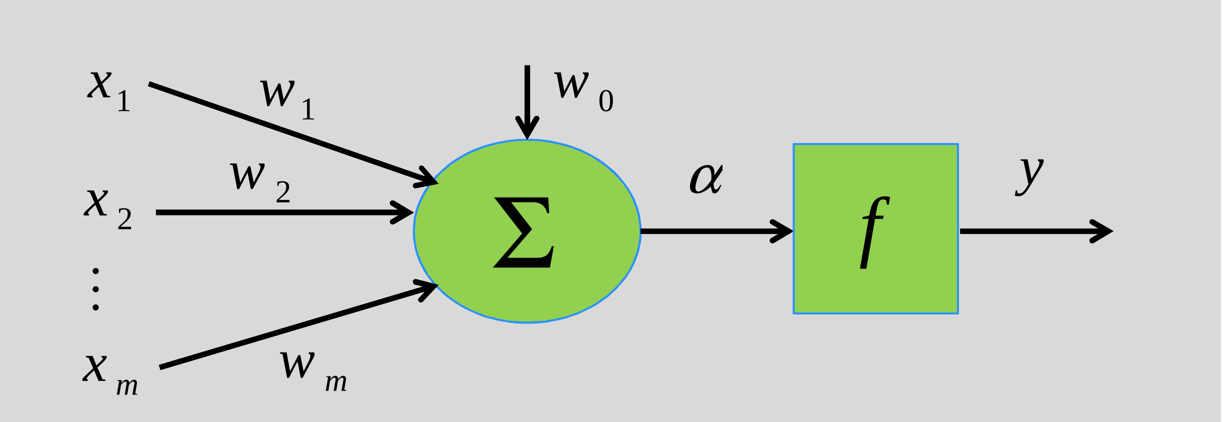
Overview of function areas in brain



Artificial neurons

Basic idea: simulate the function of biological neurons and build networks of neurons similar to the nervous system

General structure of a neuron:



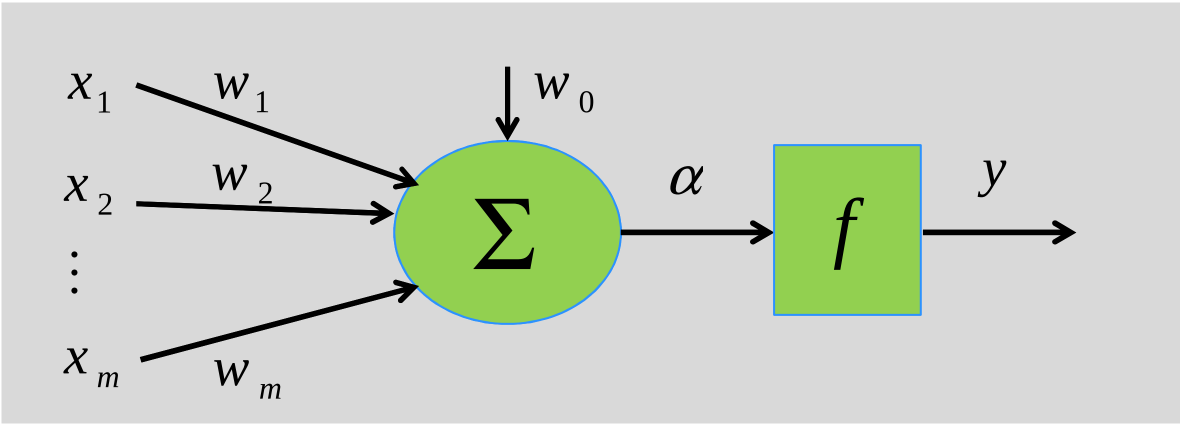
$$y = f(\alpha), \quad \alpha = w_0 + \sum_{j=1}^{m} w_j x_j$$

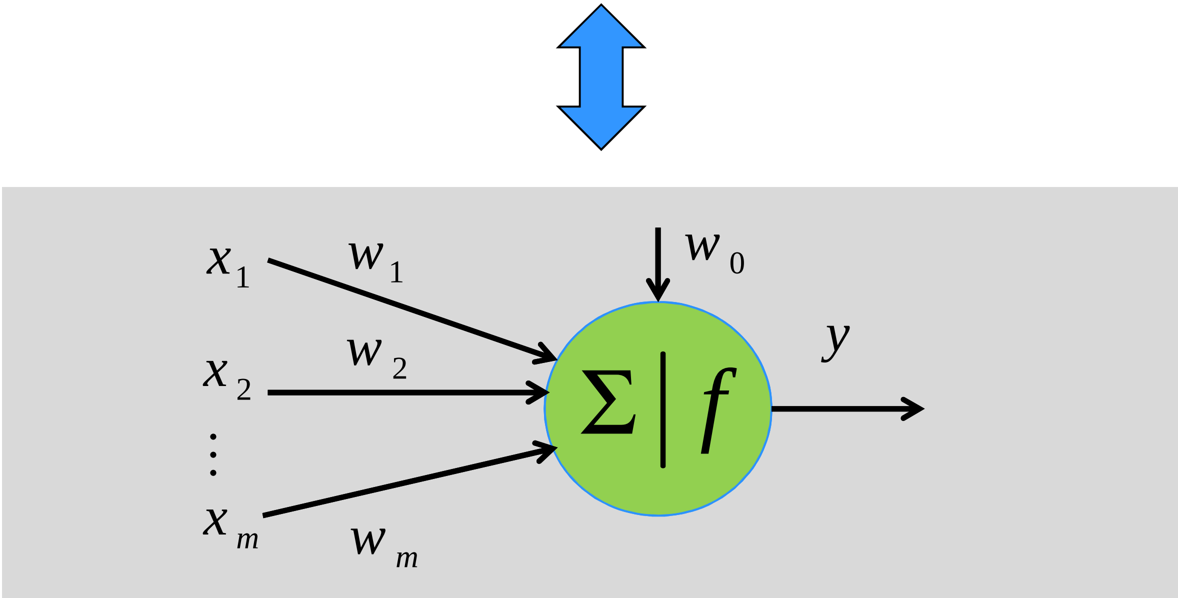
$w_0, w_1, \cdots, w_m$ are constants.

f is called activation function.

Artificial neurons

For convenience, the neuron is often abbreviated as

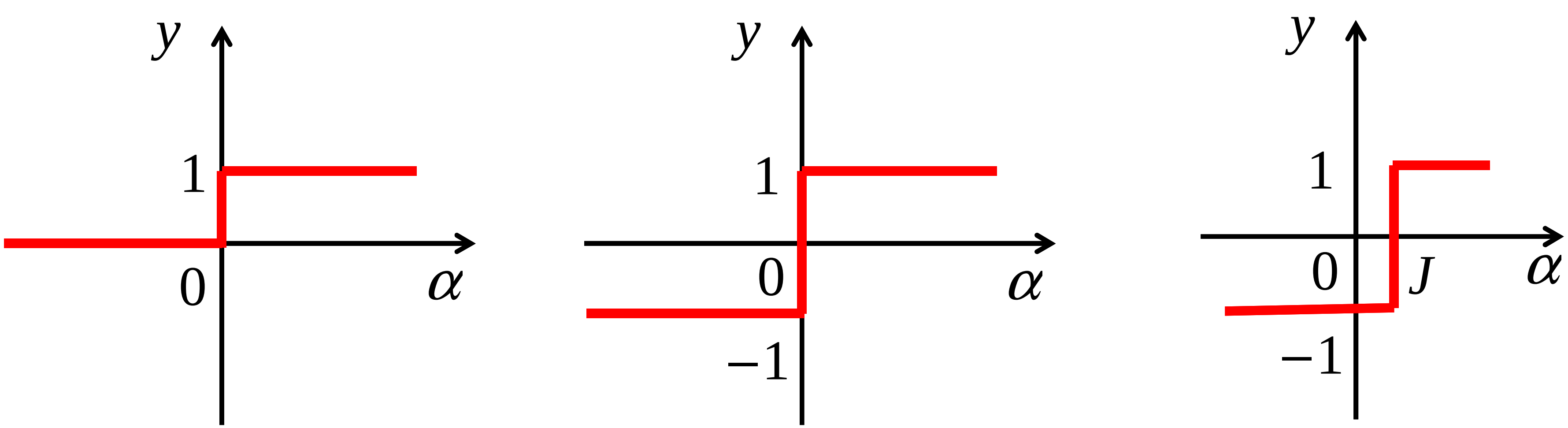




Typical activation functions

Threshold functions (for neurons with binary outputs)

$$y = \begin{cases} 1, & \text{if } \alpha \geq 0 \\ 0, & \text{if } \alpha < 0 \end{cases} \qquad y = \begin{cases} 1, & \text{if } \alpha \geq 0 \\ -1, & \text{if } \alpha < 0 \end{cases} \qquad y = \begin{cases} 1, & \text{if } \alpha \geq J \\ -1, & \text{if } \alpha < J \end{cases}$$



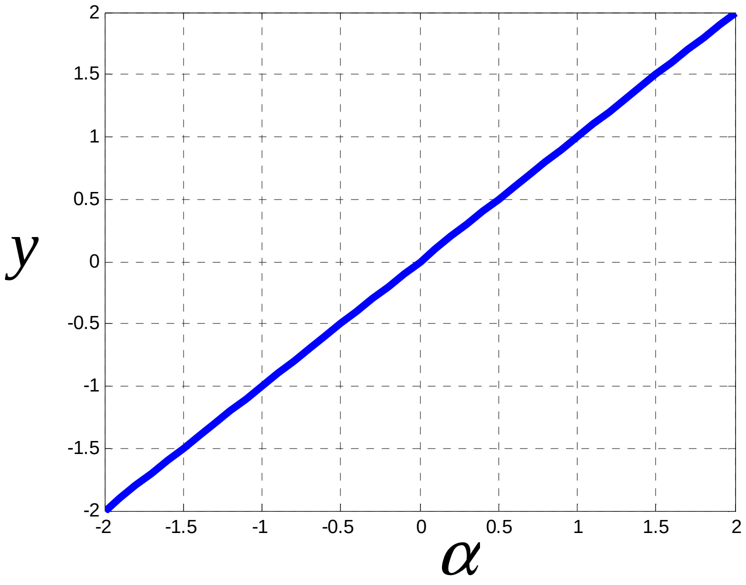
Typical activation functions

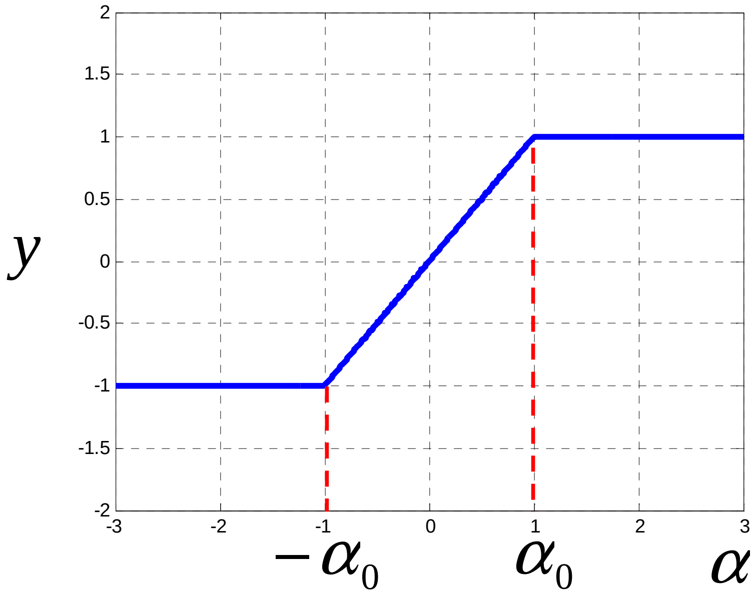
linear function

$$y = \alpha$$

piecewise linear function

$$y = \begin{cases} 1, & \text{if } \alpha > \alpha_0 \\ x/\alpha_0, & \text{if } -\alpha_0 \le \alpha \le \alpha_0 \\ -1, & \text{if } \alpha < -\alpha_0 \end{cases}$$

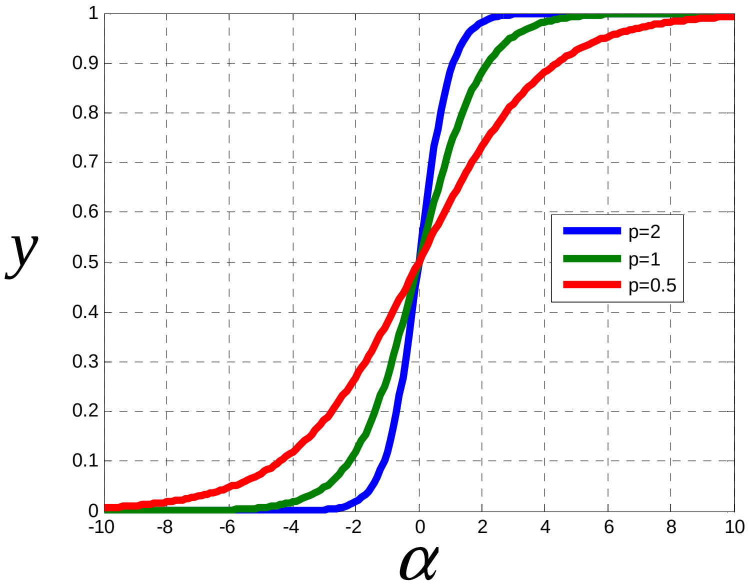




Typical activation functions

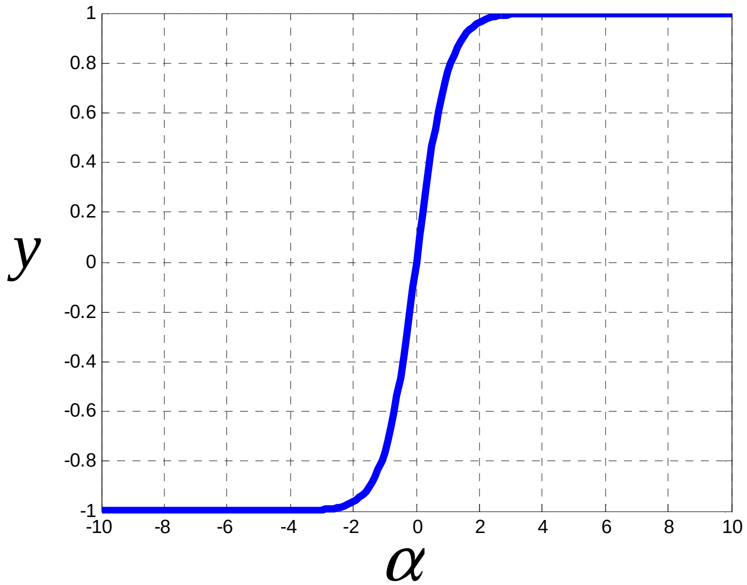
Sigmoid function

$$y = \frac{1}{1+e^{-p\alpha}}$$



Hyperbolic tangent function

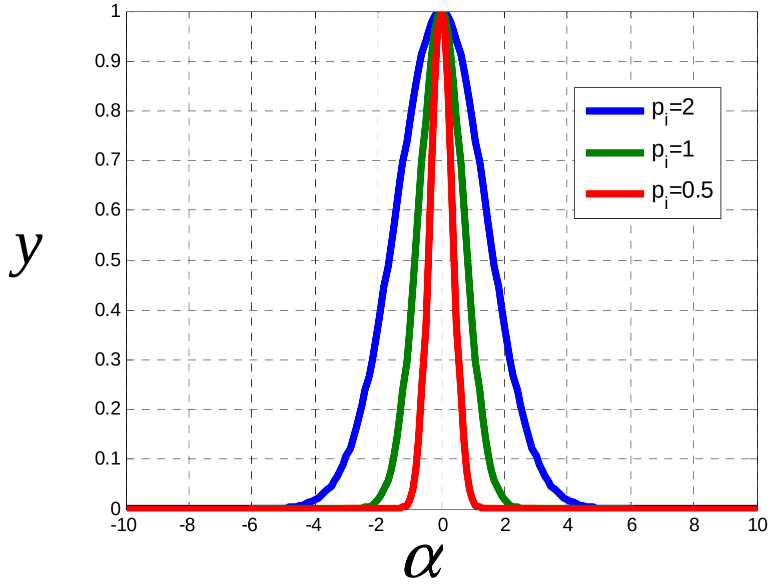
$$y = \frac{e^{\alpha}-e^{-\alpha}}{e^{\alpha}+e^{-\alpha}} = 1-\frac{2}{e^{2\alpha}+1}$$



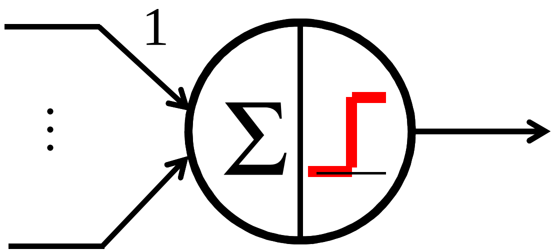
Typical activation functions

Radial basis function

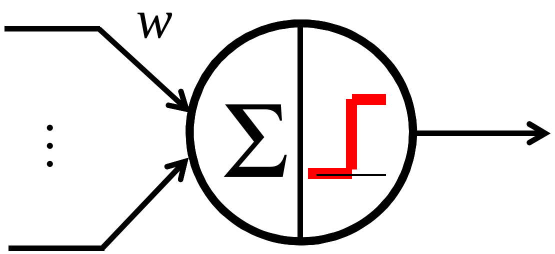
$$y = e^{-\left(\frac{\alpha}{p}\right)^2}$$



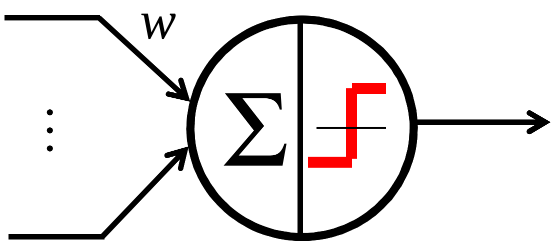
Typical neurons



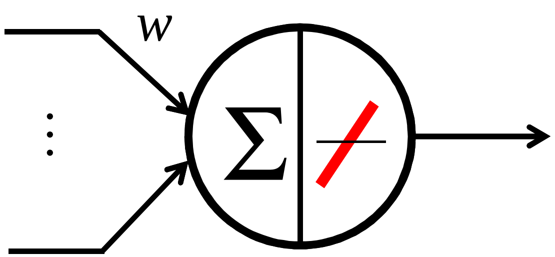
McCulloch-Pitts-Neuron
(no weighting of input signals,
i.e. w = [1 1 … 1])



Perceptron

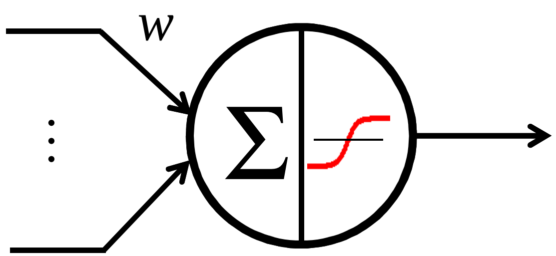


Bipolar perceptron

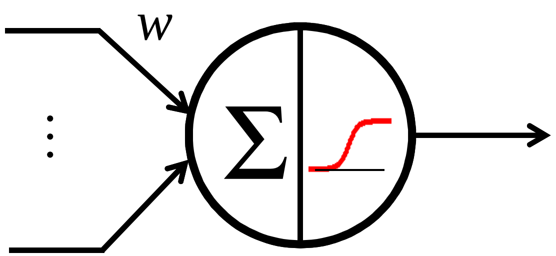


Linear associator

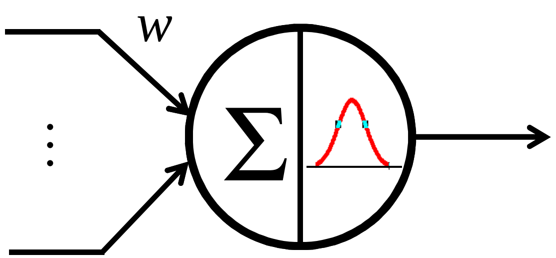
Typical neurons



Bipolar sigmoid neuron



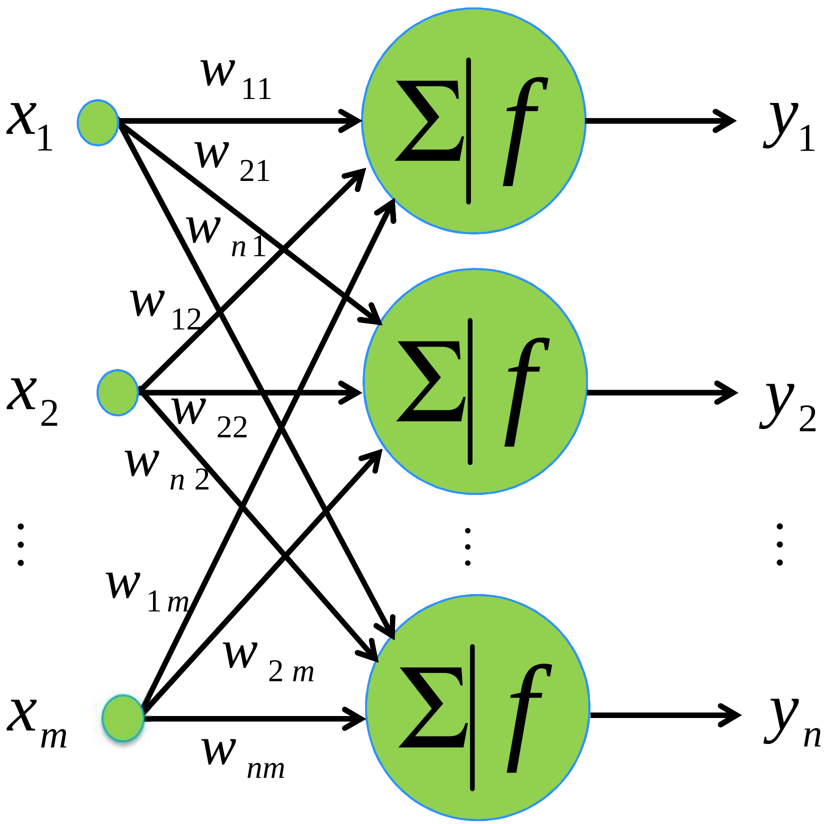
Unipolar sigmoid neuron



RBF neuron

Neural network structures

One network layer with m inputs and n outputs



$$y_1 = f\left(w_{10} + \sum_{j=1}^{m} w_{1j} x_j \right)$$

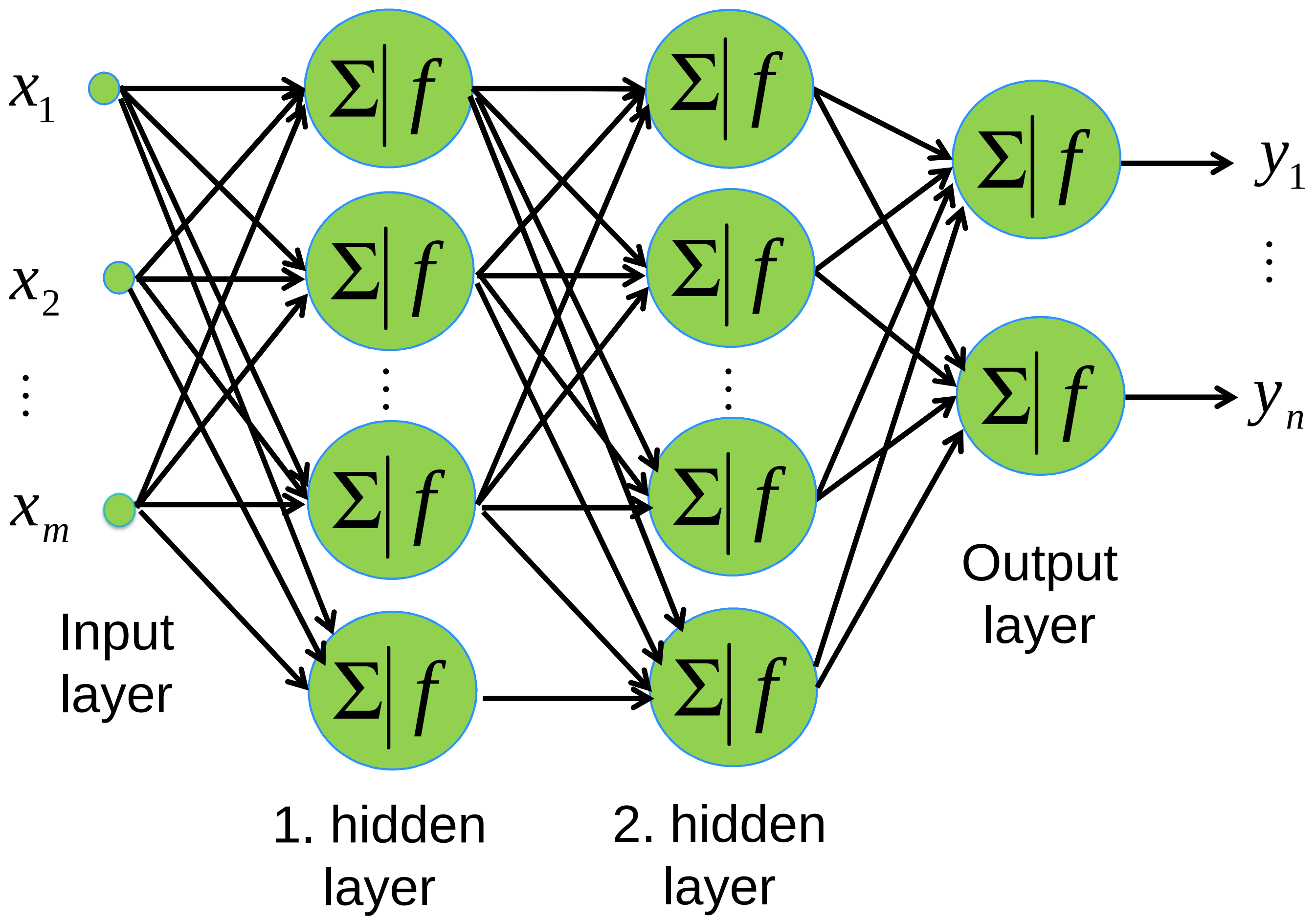
$$y_2 = f\left(w_{20} + \sum_{j=1}^{m} w_{2j} x_j \right)$$

$$\vdots$$

$$y_n = f\left(w_{n0} + \sum_{j=1}^{m} w_{nj} x_j \right)$$

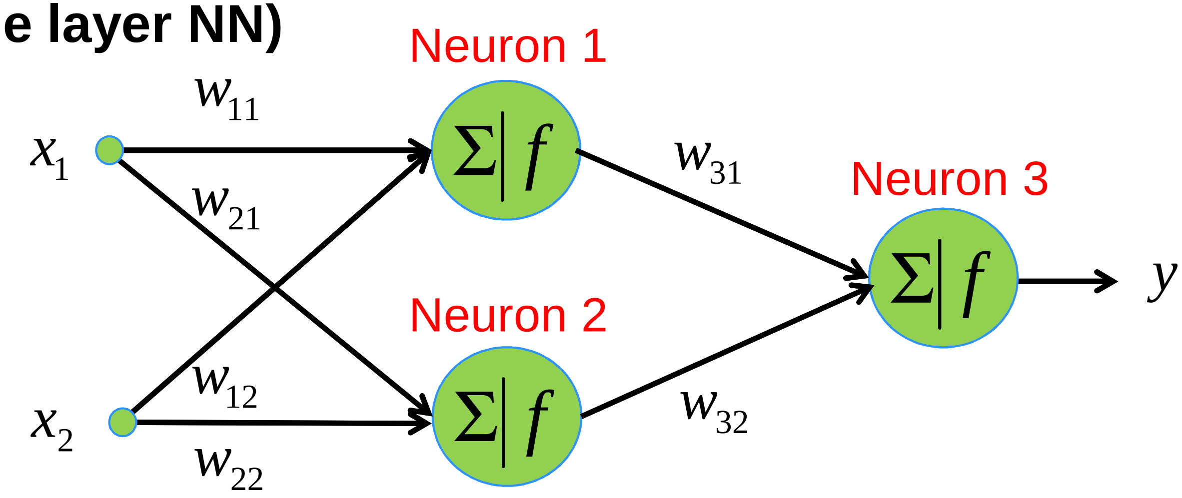
Neural network structures

Feedforward network



Neural network structures

Example 1 (One layer NN)



What is the relation between the NN output y and the NN inputs x_1, x_2?

The outputs of the neurons in the hidden layer are

$$y_1 = f(\alpha_1) = f(w_{11}x_1 + w_{12}x_2)$$

$$y_2 = f(\alpha_2) = f(w_{21}x_1 + w_{22}x_2)$$

The output of the neuron in the output layer is

$$y = y_3 = f(\alpha_3) = f(w_{31}y_1 + w_{32}y_2)$$

$$y = f(w_{31}f(w_{11}x_1 + w_{12}x_2) + w_{32}f(w_{21}x_1 + w_{22}x_2))$$