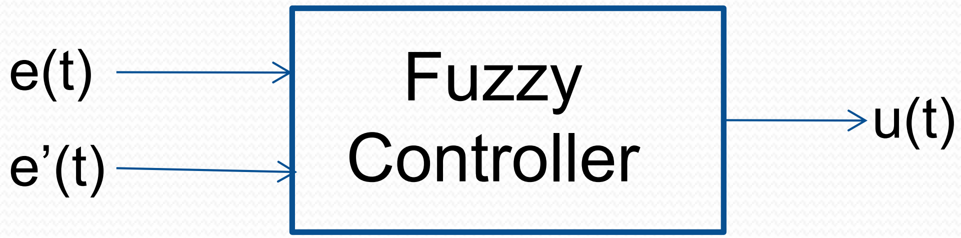
Fuzzy Adaptive Control

Ning Xiong

Mälardalen University

Computer Discrete Control



- But, a computer can only offer discrete control. Every sampling period, the computer read the sensor data and then makes the decision to a new updated action.

- Sensing and decision at time steps kT (k=0, 1, 2,….). The length of sampling period T=20, 30, 50ms, …..

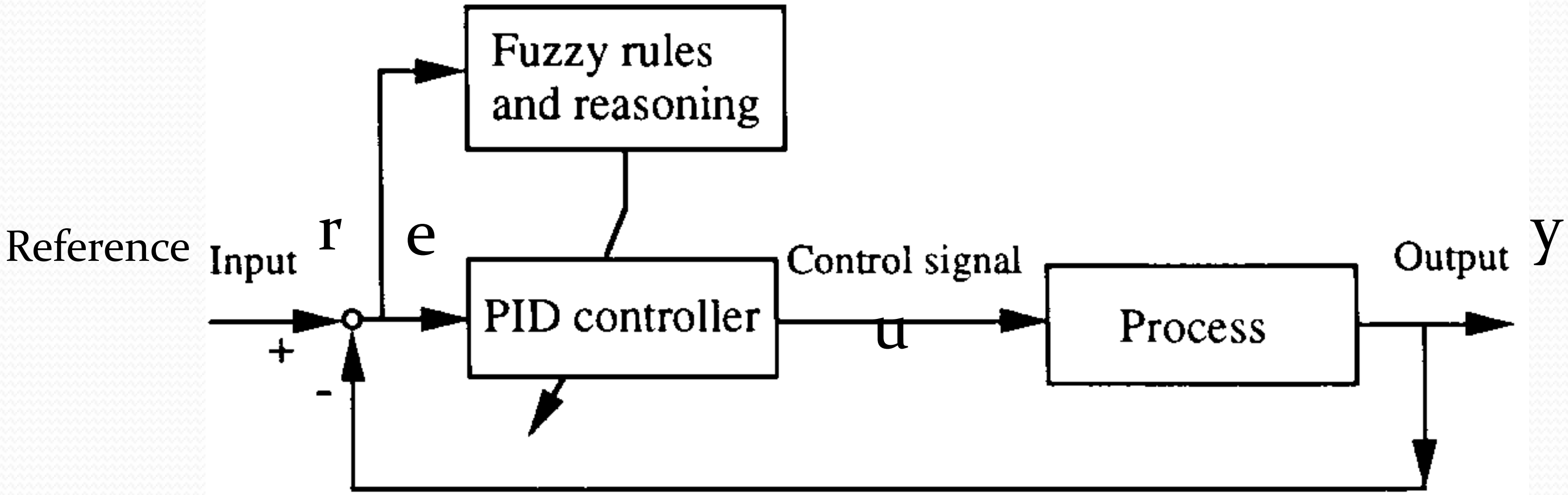
$$e'(kT) = \frac{e(kT) - e(kT - T)}{T}$$

Outline

- Fuzzy approach to adapting parameters of a conventional controller
  - Fuzzy gain tuning for PID controllers (fuzzy PID control)
- On-line adaptation of fuzzy controllers
  - Fuzzy reference model learning control
- Offline "adaptation" or learning of fuzzy controllers (offline learning and optimization)

Fuzzy approach to controller parameter adaptation (Fuzzy PID)

Fuzzy Tuning of PID Controller

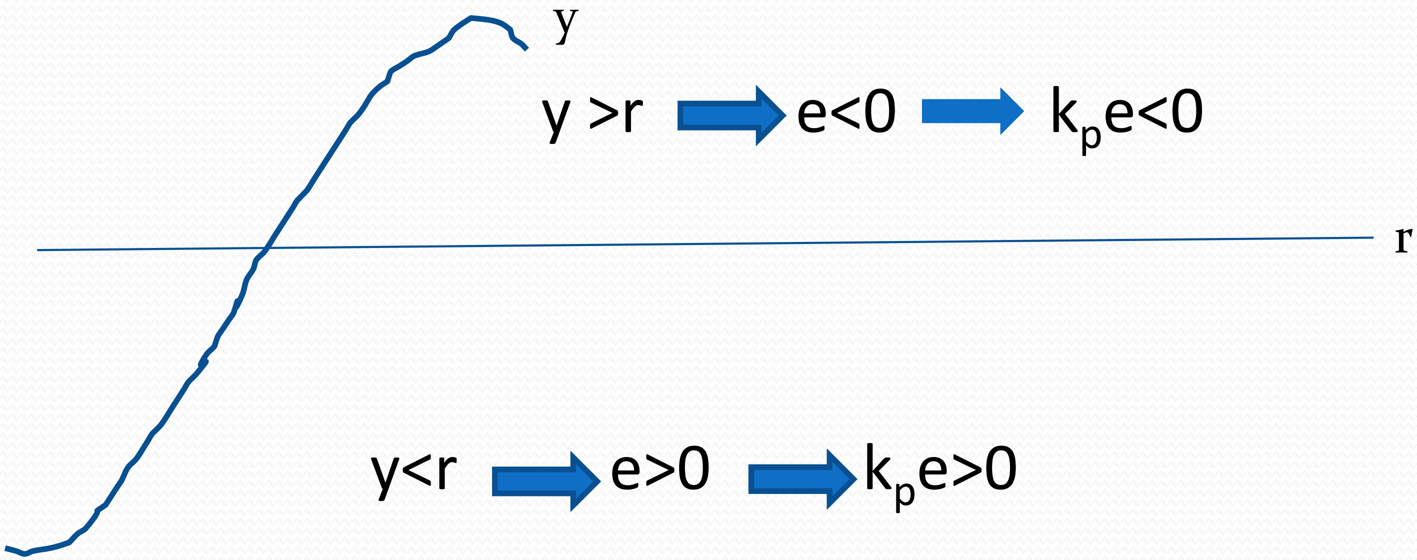


$$u(t) = K_p e(t) + K_i \int_0^t e(t) d(t) + K_d e'(t)$$

$$u(k) = K_p e(k) + K_i T \sum_{j=0}^{K} e(j) + \frac{K_d}{T}[e(k) - e(k-1)]$$

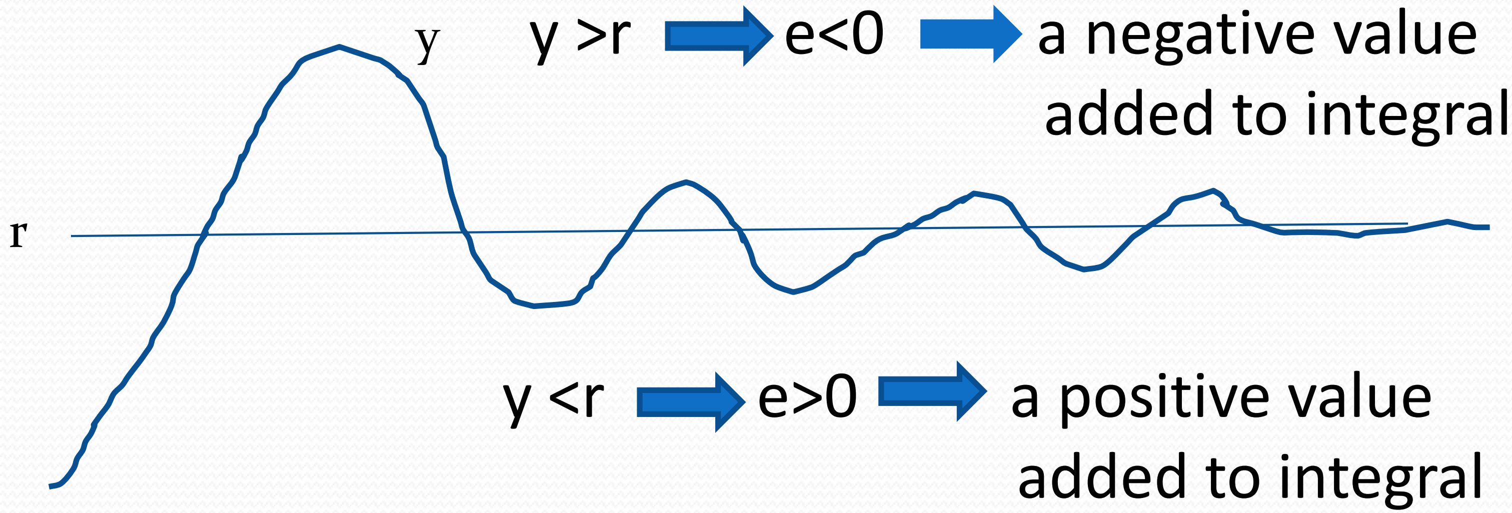
The **proportional gain** K_p, **integral gain** K_i, and **derivative gain** K_d are specified in advance. They remain constant in control process with conventional PID control

The effect of proportional part



Proportional part: typically the main drive in a control loop, reducing a large part of the error.

Effect of the integral part

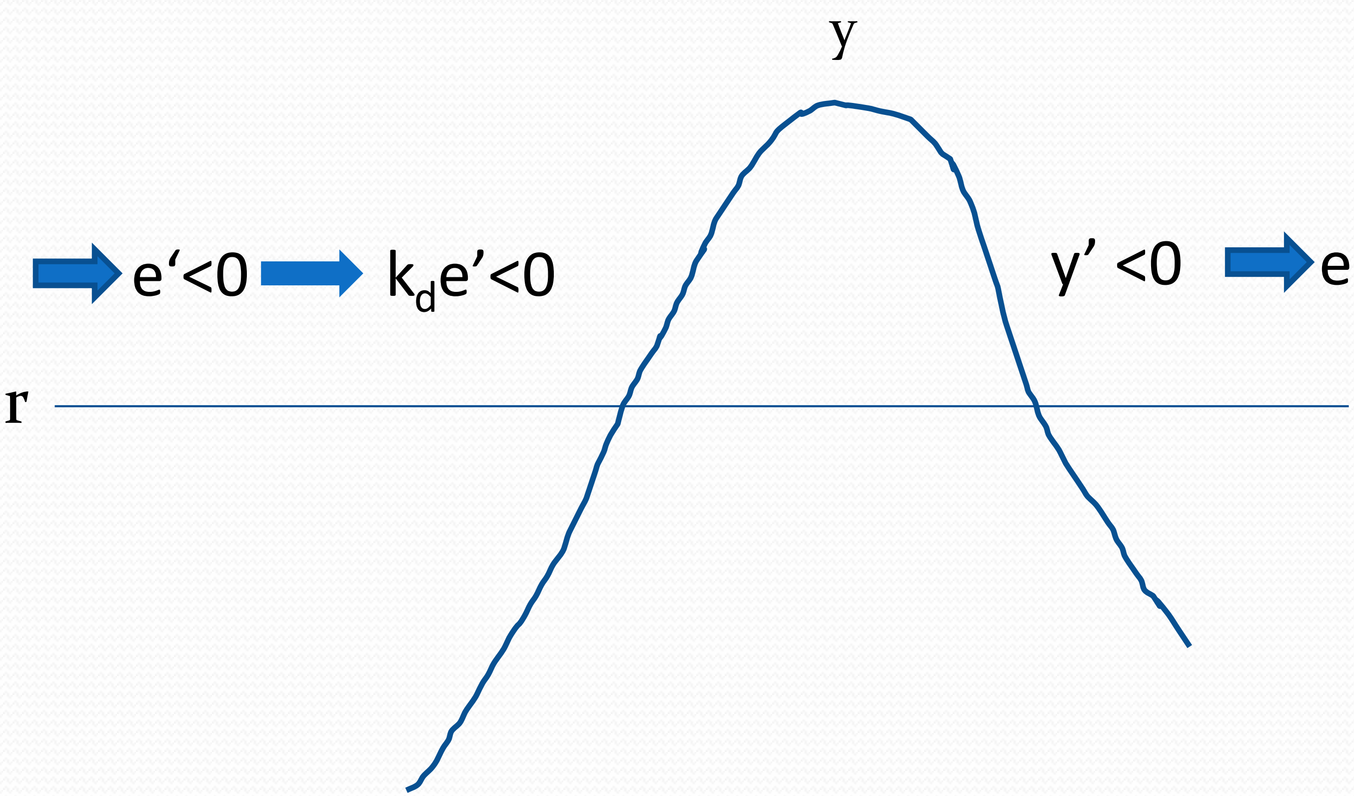


Integral part: Reducing the final error to zero.
Summing even a small error over time produces a drive signal to move the system toward a smaller error.

Effect of the derivative part

$y' > 0 \Rightarrow e' < 0 \Rightarrow k_d e' < 0$

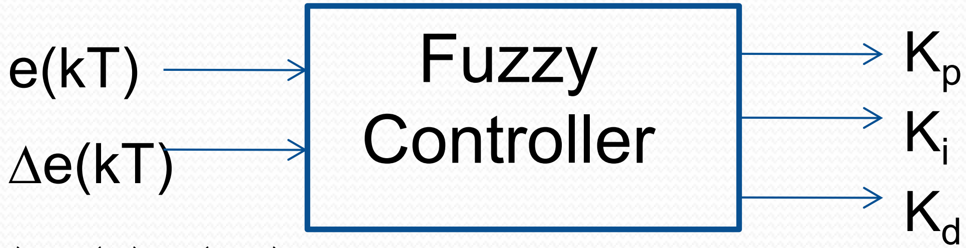
$y' < 0 \Rightarrow e' > 0 \Rightarrow k_d e' > 0$



Derivative part: Curbing quick changes of the output. This helps reduce overshoot. It has no effect on final error.

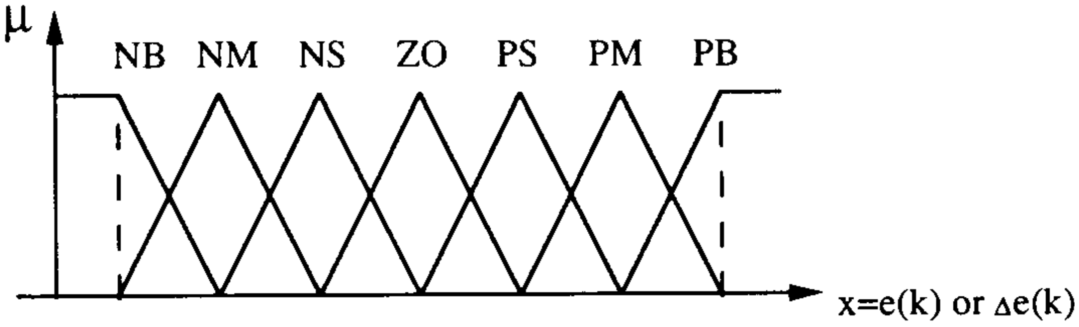
Fuzzy Tuning of PID

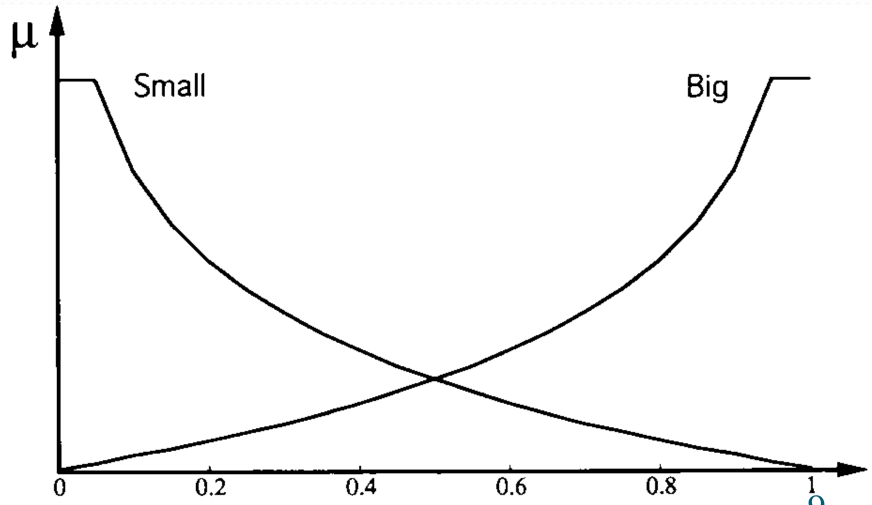
Using fuzzy controller to enable varying PID gains in terms of the situation



$\Delta e(kT) = e(k) - e(k-1)$

Fuzzy membership functions for gains





Fuzzy Gain Tuning Rules

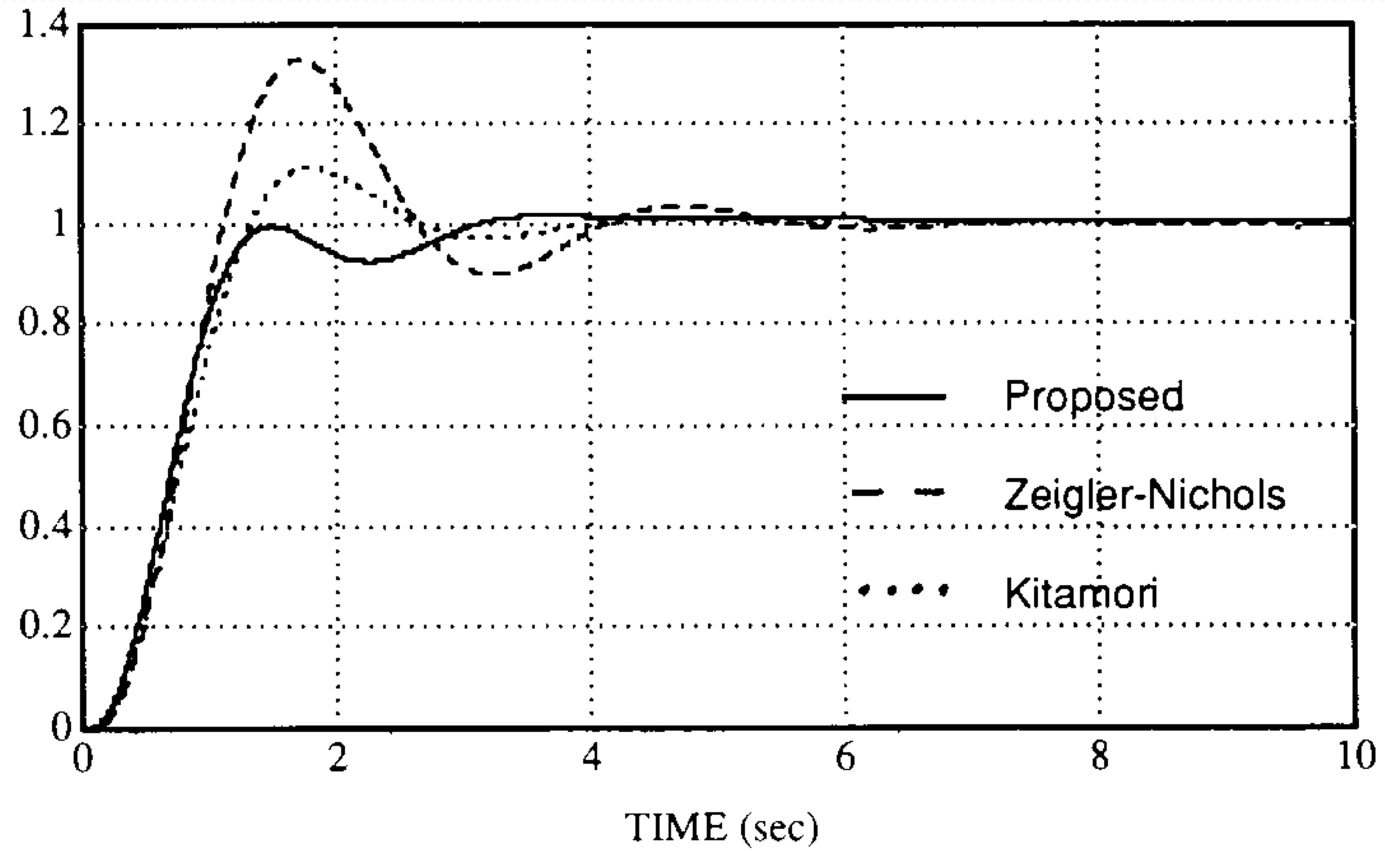
TABLE I
FUZZY TUNING RULES FOR K'_p

		$\Delta e(k)$						
		NB	NM	NS	ZO	PS	PM	PB
	NB	B	B	B	B	B	B	B
	NM	S	B	B	B	B	B	S
	NS	S	S	B	B	B	S	S
$e(k)$	ZO	S	S	S	B	S	S	S
	PS	S	S	B	B	B	S	S
	PM	S	B	B	B	B	B	S
	PB	B	B	B	B	B	B	B

Interested students (likely with Electrical Engineering background) can find further information from the paper:

"Fuzzy gain scheduling of PID controllers", IEEE Trans. Systems, Man & Cybernetics, No. 5, 1993, pp. 1392-1398.

Improved Performance with Fuzzy Tuning



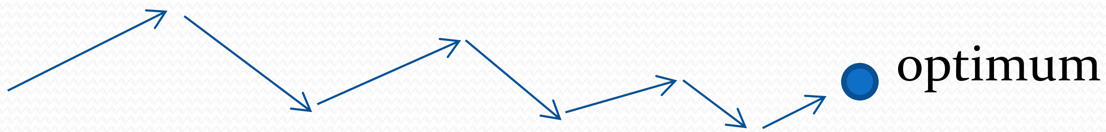
Merits with fuzzy tuning: smaller overshoot, faster reaching stable status without error

Inspiration from Fuzzy PID?

Can we use fuzzy to adjust parameters of a procedure/method during its operation?

Examples:

- Adjust the mutation rate of GA with generations?
- Adjust the step size during a search process?



Fuzzy is like soy sauce for food, can make system more elegant. It is rewarding to use fuzzy techniques in many application areas

Online adaptation of fuzzy controller

Why Needing Online Adaptation for Fuzzy Controller ?

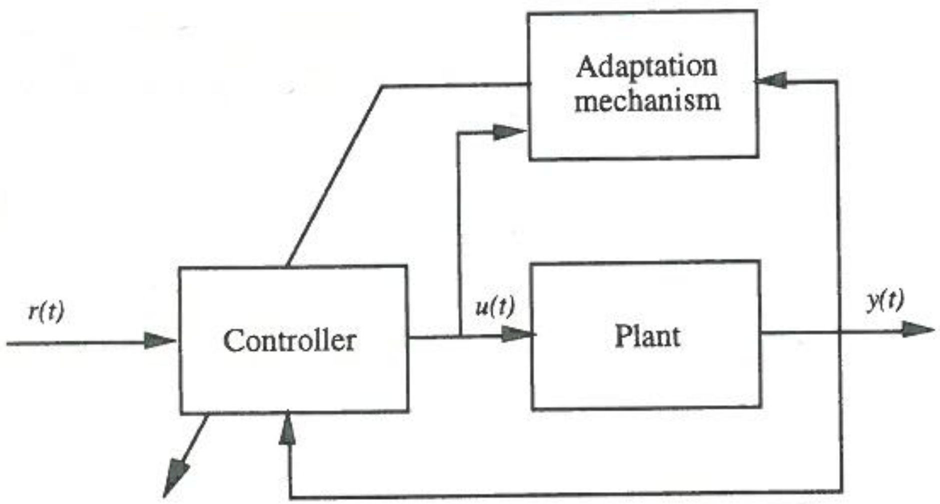
1. Human knowledge and experience is neither exact nor perfect

2. Sometimes humans can not precisely express their experience and knowledge

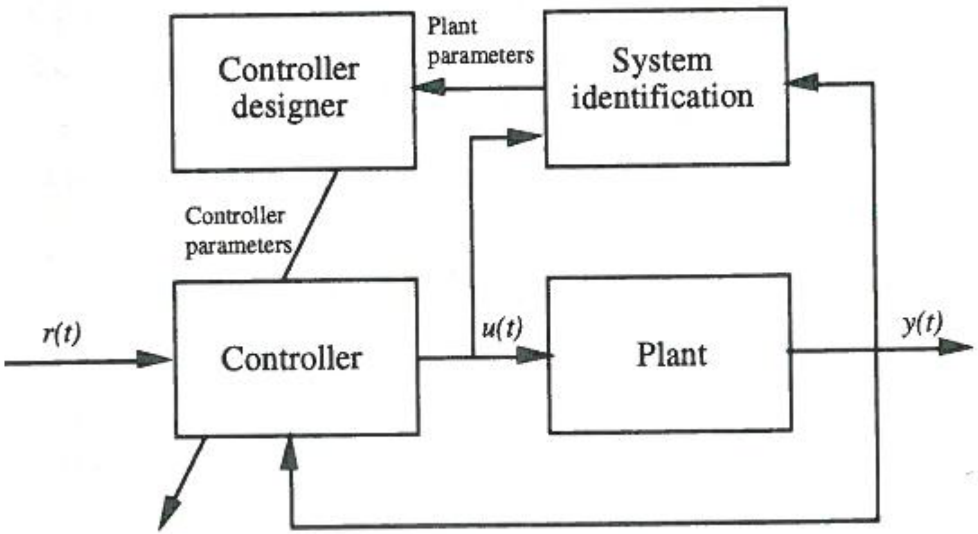
3. Environment /plant can change over time, fuzzy controller needs to adjust its strategy in response to such changes.

Two Types of Adaptive Fuzzy Control

Direct adaptive control



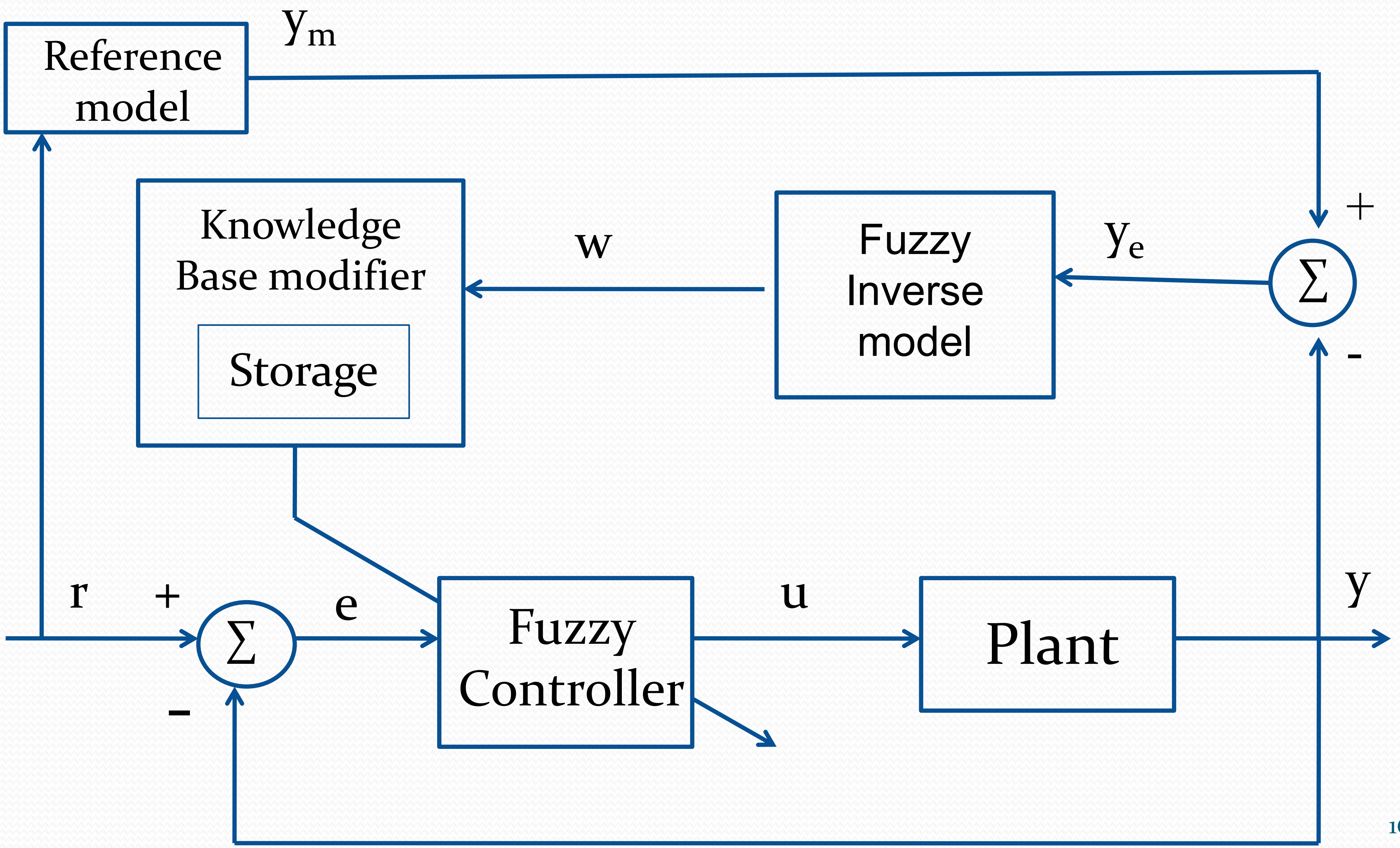
Indirect adaptive control



Utilize output signal to adapt

Utilize parameters of the plant estimated via on-line system identification

Fuzzy Reference Model Learning Control



Fuzzy Reference Model Learning Control

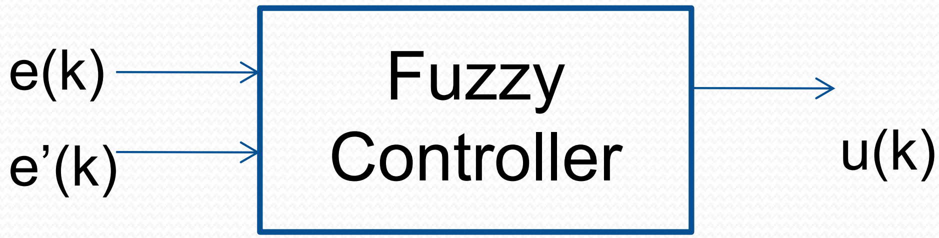
• Fuzzy controller: decides u(*k*) based on e(*k*) and e'(*k*). Its goal is to control the plant to track the reference input

• Reference model: gives the requirement of how to track the reference input. It specifies the desired system output y_m at every time step given the reference signal

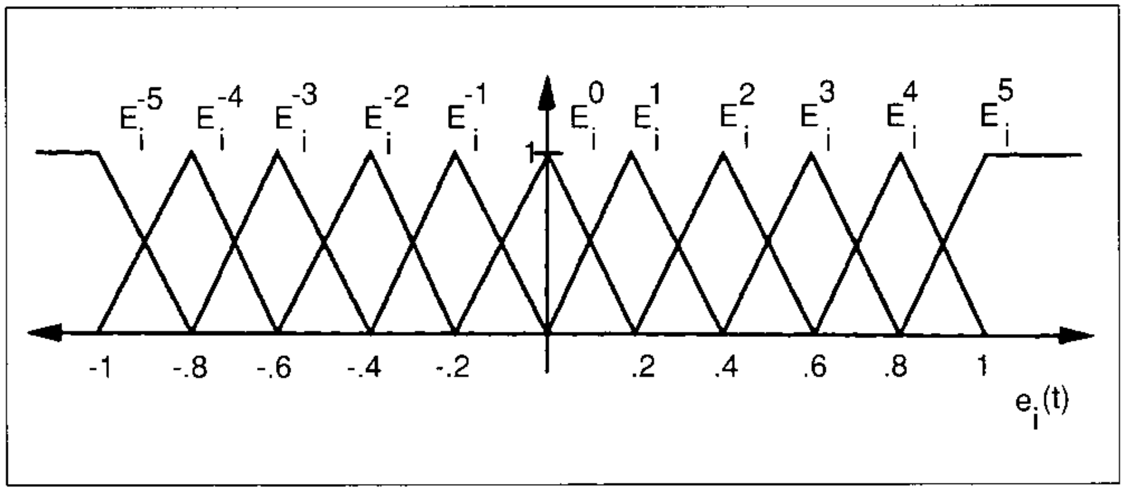
• Fuzzy inverse model: decides the value of change w for the controller output *u* such that the difference y_e between the desired output y_m from the reference model and the actual output y should have been reduced to zero. It has the same structure as the fuzzy controller

• Knowledge-base modifier: modifies the the fuzzy rules to realize the required change for output *u*.

Fuzzy Controller



Fuzzy membership function for inputs

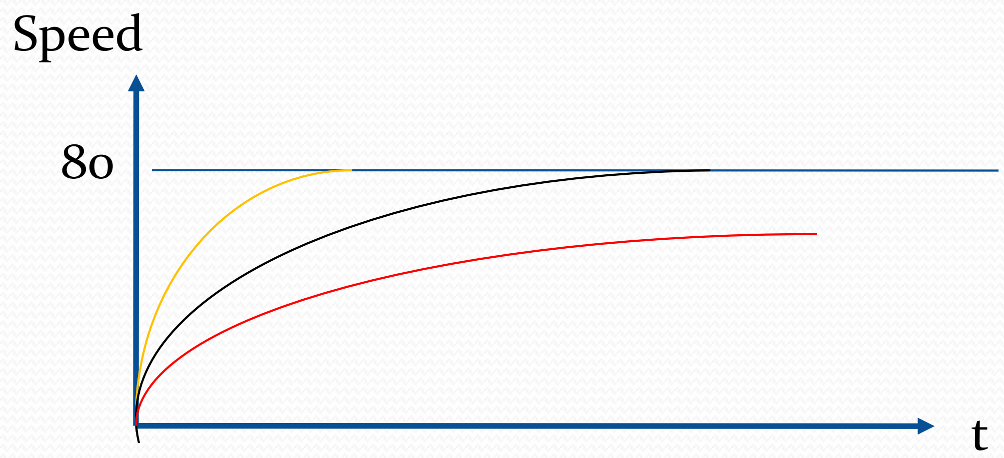


Initial conclusions of 121 rules:

- Best guess using available information
- Membership functions with centers at zero (no idea at all)

Reference Model

A reference model gives a feasible specification of how to best track the input reference signal



Reference Model Formulation

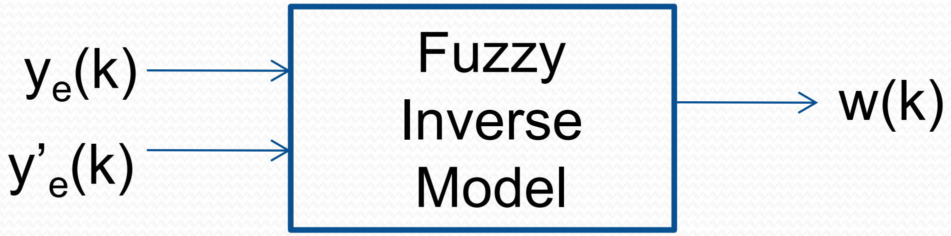
Reference model shows the desired relation between plant output y(t) and reference input r(t), or how should *y* change in response to *r* .

Example: $$y'(t) + y(t) = r(t)$$

Discrete form with T=0.1 sec.:

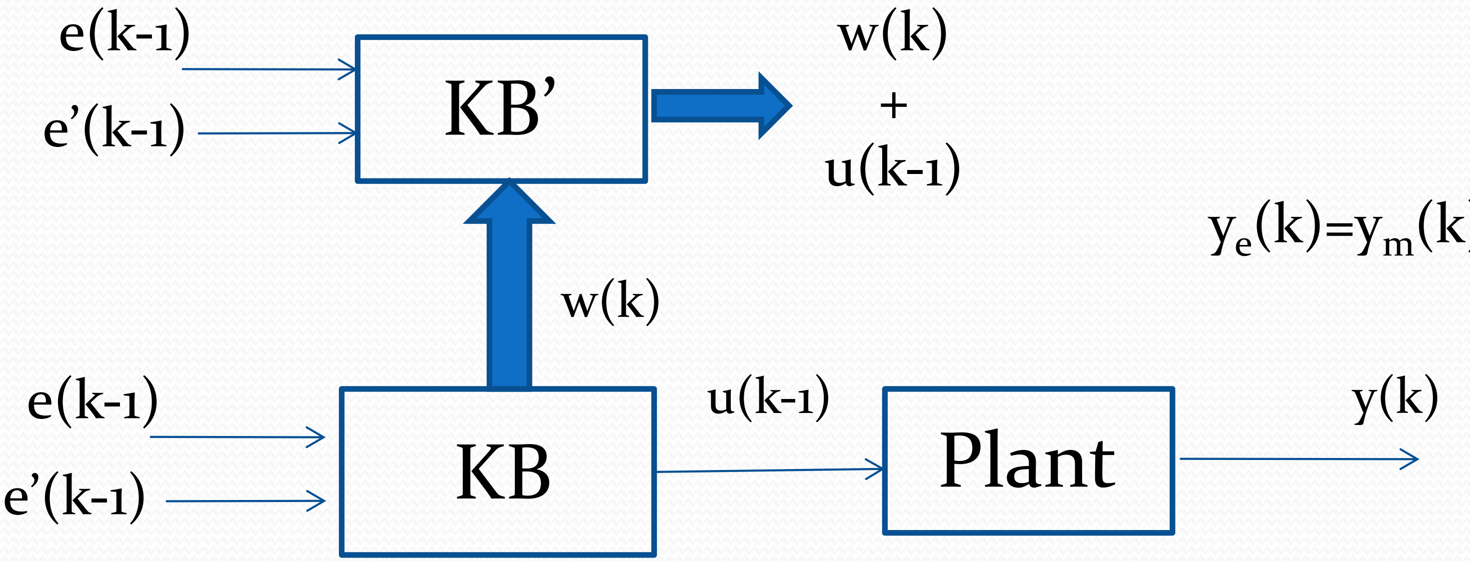
$$y_m(kT + T) = \frac{19}{21} y_m(kT) + \frac{1}{21} r(kT + T) + \frac{1}{21} r(kT)$$

Fuzzy Inverse Model



- Fuzzy Inverse Model: similar to fuzzy controller
- Fuzzy Inverse model: fuzzy knowledge base, inference mechanism, fuzzification, defuzzification

Knowledge Modifier



$y_e(k)=y_m(k)-y(k)$

Focusing on those rules whose firing strength are non-zero, i.e.

$$t_i(e(k-1),e'(k-1))>0$$

Modifying Effective Rules

- Only consider and modify the effective rules

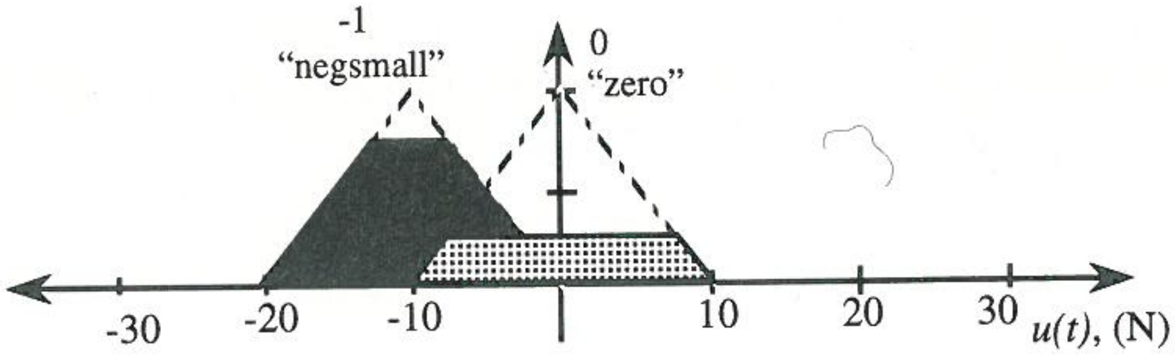
- Only move the consequent membership functions of the effective rules. Their shapes are not changed.

- Suppose b is the center of the output membership function of an effective rule, after time step kT ,its value will be revised according to the following rule:

$$b(KT)=b(KT-T)+w(kT)$$

- If the centers of the membership functions of effective rules are revised in terms of the above rule, the controller output in the preceding step should have been u(kT-T)+w(kT)

Verifying the Modification



Center of Gravity (COG)

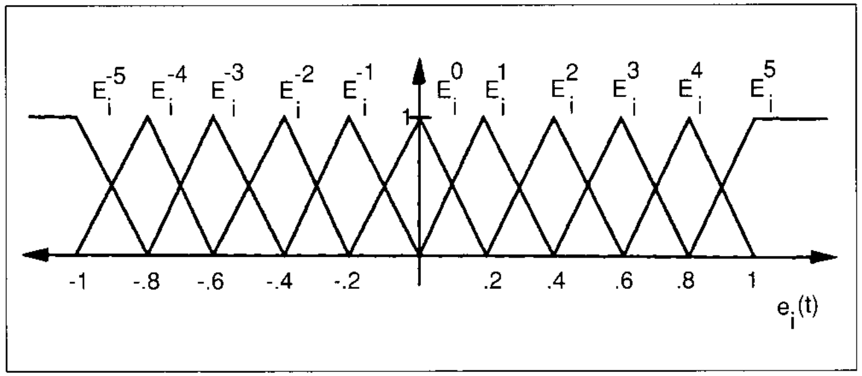
$$u^{crisp} = \frac{\sum_i b_i \int \mu_{(i)}(u)du}{\sum_i \int \mu_{(i)}(u)du}$$

$$u = \frac{b_1 S_1 + b_2 S_2}{S_1 + S_2}$$

$$u' = \frac{(b_1 + w)S_1 + (b_2 + w)S_2}{S_1 + S_2} = u + w$$

Example

Suppose w(k)=0.5, e(k-1)=0.75, e'(k-1)=-0.2



Effective rules:

R_1: if e="3" and e'="-1" then U_1

R_2: if e="4" and e'="-1" then U_2

Modification of centers b1 and b2 for U1 and U2:

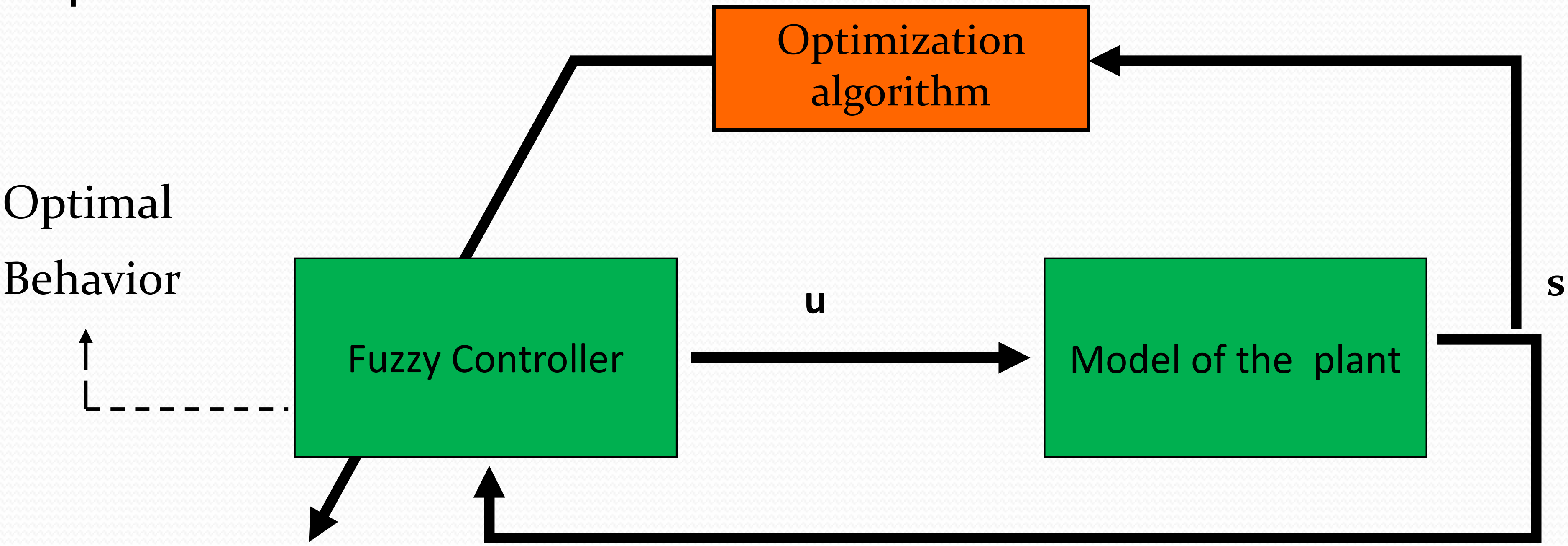
$b_1(k)=b_1(k-1)+0.5$, $b_2(k)=b_2(k-1)+0.5$

Offline adaptation of fuzzy controller

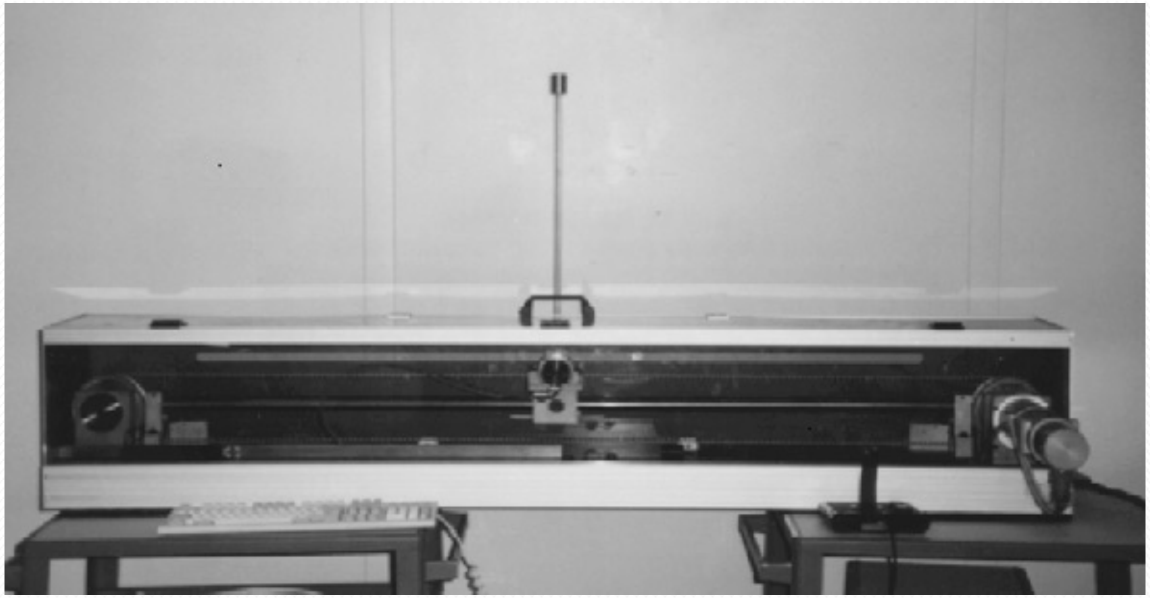
Model-Based Off-Line Fuzzy Learning

- If we know the model of the plant , we can predict the performance of a fuzzy controller by simulation.

- With the assessment of a controller by simulation, we can further apply optimization algorithm to find the best controller parameters



Control Both Position and Angle

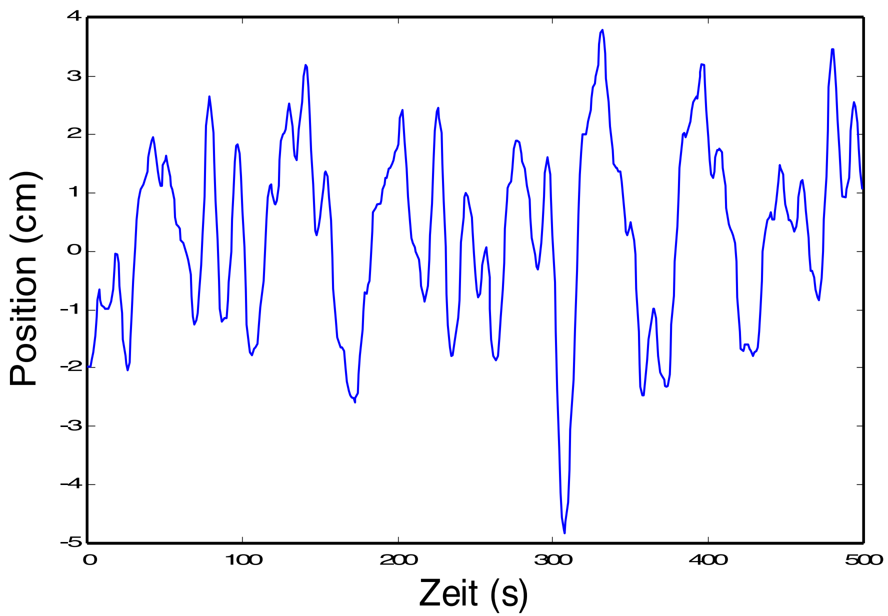


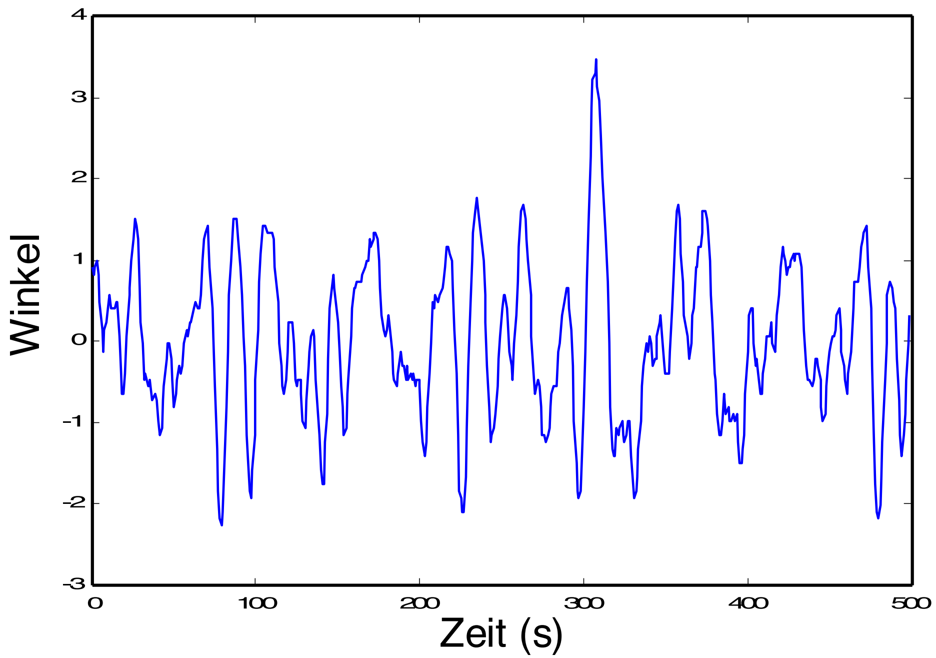
- Objectives:

1) Keep the cart at the center
2) Keep the pendulum upstraight

- Learning is based on the model of the pendulum equipment

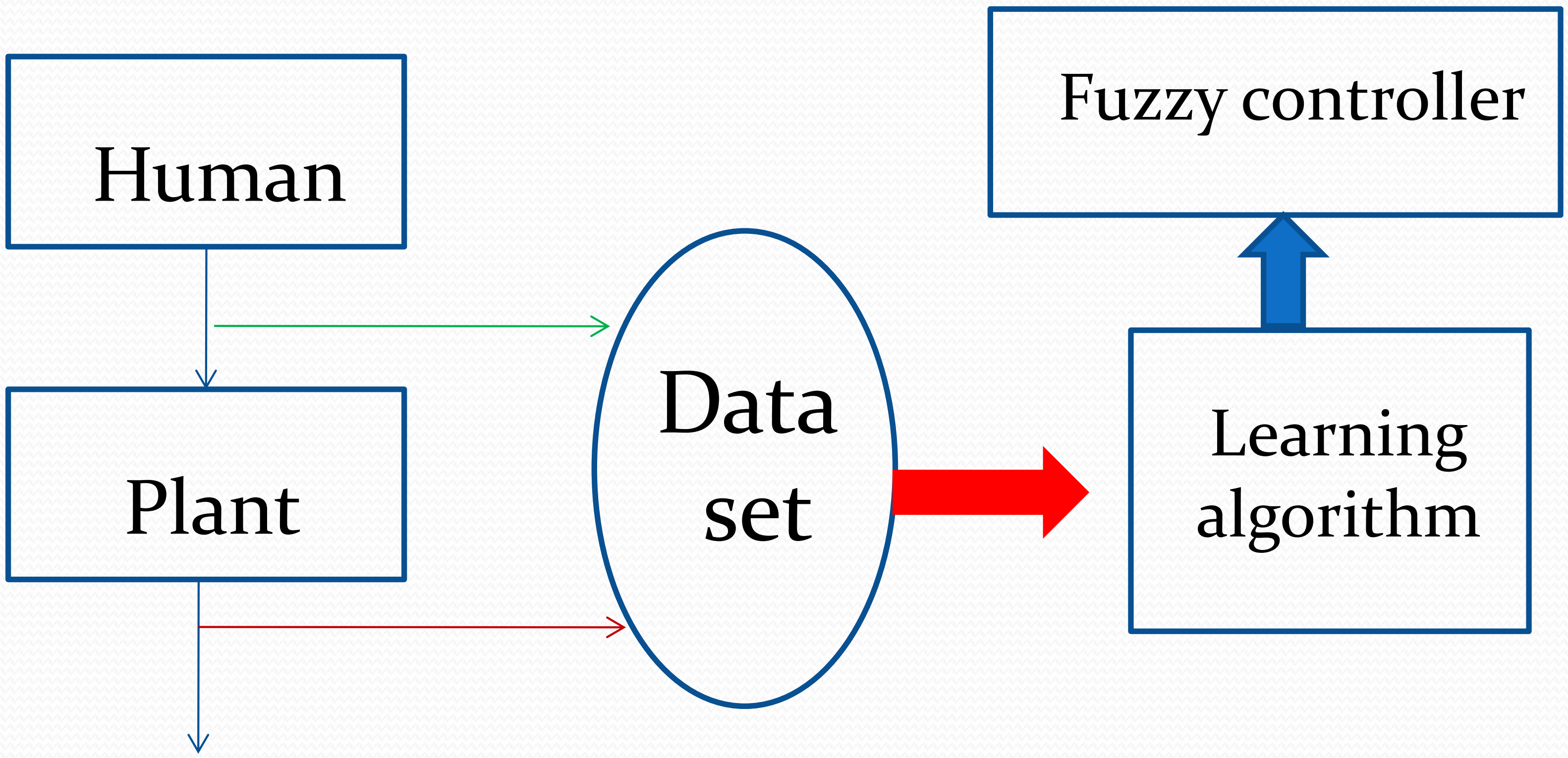
Performance in Real Balancing





Ning Xiong, Lothar Litz, "Reduction of fuzzy control rules by means of premise learning – Method and case study", Fuzzy Sets and Systems, Vol. 132, 2002, pp. 217-231.

Data-Based Fuzzy Controller Learning



A Simple Method to Extract Rules from Data

L.X. Wang and J.M. Mendel, ”Generating fuzzy rules by learning from examples”, IEEE Trans. Systems, Man, & Cybernetics, Vol. 22, No. 6, 1992, pp. 1414-1427.

Wang-Mendel Method:

1. Create a fuzzy rule from every training example
2. Assess evey created rule with a truth value
3. Redudance and conflict removal

The rules created by Wang-Mendel method can be treated as a good starting point for further optimization of membership functions

Recommendation for Reading

- It is important to carefully study and understand the content in the slides

- Optional reading:
  "Fuzzy gain scheduling of PID controllers", IEEE Trans. Systems, Man & Cybernetics, No. 5, 1993, pp. 1392-1398. Available in the blackboard

- Compulsory reading:
"Fuzzy model reference learning control for cargo ship steering", IEEE Control Systems Magazine, 1993. Available in the blackboard.