

# Computational Intelligence

Winter Term 2018/19

Prof. Dr. Günter Rudolph

Lehrstuhl für Algorithm Engineering (LS 11)

Fakultät für Informatik

TU Dortmund

- Approximate Reasoning
- Fuzzy Control

### So far:

- p: IF X is A THEN Y is B

$$\rightarrow R(x, y) = \text{Imp}( A(x), B(y) )$$

rule as relation; fuzzy implication

- rule: IF X is A THEN Y is B
- fact: X is A'
- conclusion: Y is B'

$$\rightarrow B'(y) = \sup_{x \in X} t( A'(x), R(x, y) )$$

composition rule of inference

### Thus:

- $B'(y) = \sup_{x \in X} t( A'(x), \text{Imp}( A(x), B(y) ) )$

given : fuzzy rule

input : fuzzy set A'

output : fuzzy set B'

here:

$$A'(x) = \begin{cases} 1 & \text{for } x = x_0 \\ 0 & \text{otherwise} \end{cases} \quad \boxed{\text{crisp input!}}$$

$$B'(y) = \sup_{x \in X} t( A'(x), \text{Imp}( A(x), B(y) ) )$$

$$= \begin{cases} \sup_{x \neq x_0} t( 0, \text{Imp}( A(x), B(y) ) ) & \text{for } x \neq x_0 \\ t( 1, \text{Imp}( A(x_0), B(y) ) ) & \text{for } x = x_0 \end{cases}$$

$$= \begin{cases} 0 & \text{for } x \neq x_0 & \text{since } t(0, a) = 0 \\ \text{Imp}( A(x_0), B(y) ) & \text{for } x = x_0 & \text{since } t(a, 1) = a \end{cases}$$

**Lemma:**

- a)  $t(a, 1) = a$
- b)  $t(a, b) \leq \min \{ a, b \}$
- c)  $t(0, a) = 0$

**Proof:**

ad a) Identical to axiom 1 of t-norms.

ad b) From monotonicity (axiom 2) follows for  $b \leq 1$ , that  $t(a, b) \leq t(a, 1) = a$ .  
Commutativity (axiom 3) and monotonicity lead in case of  $a \leq 1$  to  $t(a, b) = t(b, a) \leq t(b, 1) = b$ . Thus,  $t(a, b)$  is less than or equal to  $a$  as well as  $b$ , which in turn implies  $t(a, b) \leq \min\{ a, b \}$ .

ad c) From b) follows  $0 \leq t(0, a) \leq \min \{ 0, a \} = 0$  and therefore  $t(0, a) = 0$ . ■

by a)



## Multiple rules:

IF X is  $A_1$ , THEN Y is  $B_1$

IF X is  $A_2$ , THEN Y is  $B_2$

IF X is  $A_3$ , THEN Y is  $B_3$

...

IF X is  $A_n$ , THEN Y is  $B_n$

X is  $A'$

---

Y is  $B'$

$$\rightarrow R_1(x, y) = \text{Imp}_1( A_1(x), B_1(y) )$$

$$\rightarrow R_2(x, y) = \text{Imp}_2( A_2(x), B_2(y) )$$

$$\rightarrow R_3(x, y) = \text{Imp}_3( A_3(x), B_3(y) )$$

...

$$\rightarrow R_n(x, y) = \text{Imp}_n( A_n(x), B_n(y) )$$

## Multiple rules for crisp input: $x_0$ is given

$$B_1'(y) = \text{Imp}_1(A_1(x_0), B_1(y) )$$

...

$$B_n'(y) = \text{Imp}_n(A_n(x_0), B_n(y) )$$

} aggregation of rules or  
local inferences necessary!

**aggregate!**  $\Rightarrow B'(y) = \text{aggr}\{ B_1'(y), \dots, B_n'(y) \}$ , where  $\text{aggr} = \begin{cases} \min \\ \max \end{cases}$

FITA: “First inference, then aggregate!”

1. Each rule of the form **IF  $X$  is  $A_k$  THEN  $Y$  is  $B_k$**  must be transformed by an appropriate fuzzy implication  $\text{Imp}_k(\cdot, \cdot)$  to a relation  $R_k$  :  

$$R_k(x, y) = \text{Imp}_k( A_k(x), B_k(y) ).$$
2. Determine  $B'_k(y) = R_k(x, y) \circ A'(x)$  for all  $k = 1, \dots, n$  (local inference).
3. Aggregate to  $B'(y) = \beta( B'_1(y), \dots, B'_n(y) ).$

FATI: “First aggregate, then inference!”

1. Each rule of the form **IF  $X$  is  $A_k$  THEN  $Y$  is  $B_k$**  must be transformed by an appropriate fuzzy implication  $\text{Imp}_k(\cdot, \cdot)$  to a relation  $R_k$  :  

$$R_k(x, y) = \text{Imp}_k( A_k(x), B_k(y) ).$$
2. Aggregate  $R_1, \dots, R_n$  to a **superrelation** with aggregating function  $\alpha(\cdot)$ :  

$$R(x, y) = \alpha( R_1(x, y), \dots, R_n(x, y) ).$$
3. Determine  $B'(y) = R(x, y) \circ A'(x)$  w.r.t. superrelation (inference).

1. Which principle is better? FITA or FATI?

2. Equivalence of FITA and FATI ?

**FITA:**

$$\begin{aligned} B'(y) &= \beta( B_1'(y), \dots, B_n'(y) ) \\ &= \beta( R_1(x, y) \circ A'(x), \dots, R_n(x, y) \circ A'(x) ) \end{aligned}$$

**FATI:**

$$\begin{aligned} B'(y) &= R(x, y) \circ A'(x) \\ &= \alpha( R_1(x, y), \dots, R_n(x, y) ) \circ A'(x) \end{aligned}$$



**special case:**

$$A'(x) = \begin{cases} 1 & \text{for } x = x_0 \\ 0 & \text{otherwise} \end{cases}$$

crisp input!

**On the equivalence of FITA and FATI:**

**FITA:**

$$B'(y) = \beta( B_1'(y), \dots, B_n'(y) )$$

$$= \beta( \text{Imp}_1(A_1(x_0), B_1(y)), \dots, \text{Imp}_n(A_n(x_0), B_n(y)) )$$

**FATI:**

$$B'(y) = R(x, y) \circ A'(x)$$

$$= \sup_{x \in X} t( A'(x), R(x, y) ) \quad (\text{from now: special case})$$

$$= R(x_0, y)$$

$$= \alpha( \text{Imp}_1( A_1(x_0), B_1(y) ), \dots, \text{Imp}_n( A_n(x_0), B_n(y) ) )$$

**evidently:** sup-t-composition with arbitrary t-norm and  $\alpha(\cdot) = \beta(\cdot)$

- **AND-connected premises**

IF  $X_1 = A_{11}$  AND  $X_2 = A_{12}$  AND ... AND  $X_m = A_{1m}$  THEN  $Y = B_1$

...

IF  $X_n = A_{n1}$  AND  $X_2 = A_{n2}$  AND ... AND  $X_m = A_{nm}$  THEN  $Y = B_n$

reduce to single premise for each rule k:

$$A_k(x_1, \dots, x_m) = \min \{ A_{k1}(x_1), A_{k2}(x_2), \dots, A_{km}(x_m) \} \quad \text{or in general: t-norm}$$

- **OR-connected premises**

IF  $X_1 = A_{11}$  OR  $X_2 = A_{12}$  OR ... OR  $X_m = A_{1m}$  THEN  $Y = B_1$

...

IF  $X_n = A_{n1}$  OR  $X_2 = A_{n2}$  OR ... OR  $X_m = A_{nm}$  THEN  $Y = B_n$

reduce to single premise for each rule k:

$$A_k(x_1, \dots, x_m) = \max \{ A_{k1}(x_1), A_{k2}(x_2), \dots, A_{km}(x_m) \} \quad \text{or in general: s-norm}$$

**important:**

- if rules of the form **IF X is A THEN Y is B** interpreted as logical implication  
⇒  $R(x, y) = \text{Imp}(A(x), B(y))$  makes sense
- we obtain:  $B'(y) = \sup_{x \in X} t(A'(x), R(x, y))$   
⇒ the worse the match of premise  $A'(x)$ , the larger is the fuzzy set  $B'(y)$   
⇒ follows immediately from axiom 1:  $a \leq b$  implies  $\text{Imp}(a, z) \geq \text{Imp}(b, z)$

**interpretation of output set  $B'(y)$ :**

- $B'(y)$  is the set of values that are still possible
- each rule leads to an additional restriction of the values that are still possible  
⇒ resulting fuzzy sets  $B'_k(y)$  obtained from single rules must be mutually intersected!  
⇒ aggregation via  $B'(y) = \min \{ B_1'(y), \dots, B_n'(y) \}$

**important:**

- if rules of the form **IF X is A THEN Y is B** are not interpreted as logical implications, then the function  $\text{Fct}(\cdot)$  in

$$R(x, y) = \text{Fct}( A(x), B(y) )$$

can be chosen as required for desired interpretation.

- frequent choice (especially in fuzzy control):

-  $R(x, y) = \min \{ A(x), B(x) \}$  Mamdani – “implication“

-  $R(x, y) = A(x) \cdot B(x)$  Larsen – “implication“

⇒ of course, they are no implications but specific t-norms!

⇒ thus, if relation  $R(x, y)$  is given,  
then the *composition rule of inference*

$$B'(y) = A'(x) \circ R(x, y) = \sup_{x \in X} \min \{ A'(x), R(x, y) \}$$

still can lead to a conclusion via fuzzy logic.

**example:** [JM96, S. 244ff.]

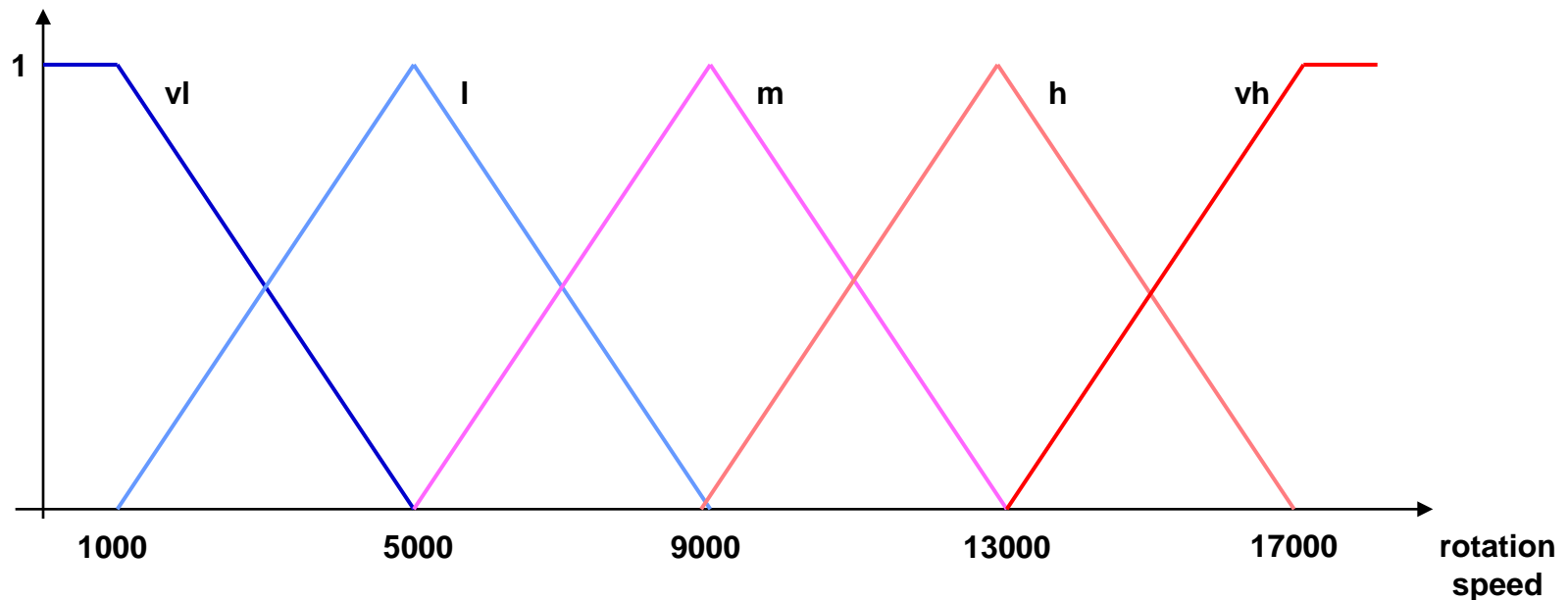
industrial drill machine → control of cooling supply

modelling

linguistic variable : **rotation speed**

linguistic terms : ***very low, low, medium, high, very high***

ground set :  $X$  with  $0 \leq x \leq 18000$  [rpm]



**example:** (continued)

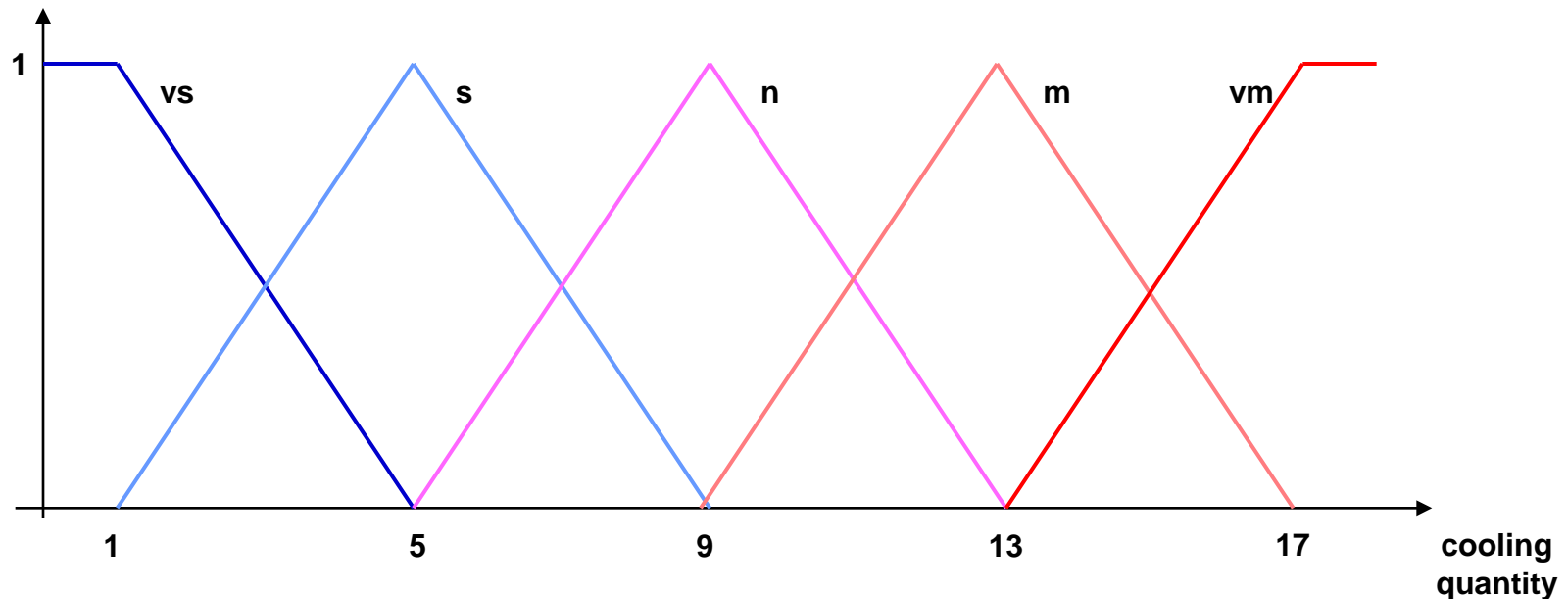
industrial drill machine → control of cooling supply

modelling

linguistic variable : **cooling quantity**

linguistic terms : ***very small, small, normal, much, very much***

ground set :  $Y$  with  $0 \leq y \leq 18$  [liter / time unit]



**example:** (continued)

industrial drill machine → control of cooling supply

rule base

|                             |                         |                                 |                          |
|-----------------------------|-------------------------|---------------------------------|--------------------------|
| <b>IF rotation speed IS</b> | <b><i>very low</i></b>  | <b>THEN cooling quantity IS</b> | <b><i>very small</i></b> |
|                             | <b><i>low</i></b>       |                                 | <b><i>small</i></b>      |
|                             | <b><i>medium</i></b>    |                                 | <b><i>normal</i></b>     |
|                             | <b><i>high</i></b>      |                                 | <b><i>much</i></b>       |
|                             | <b><i>very high</i></b> |                                 | <b><i>very much</i></b>  |



sets  $S_{vl}, S_l, S_m, S_h, S_{vh}$

“rotation speed”



sets  $C_{vs}, C_s, C_n, C_m, C_{vm}$

“cooling quantity”

**example:** (continued)

industrial drill machine → control of cooling supply

1. **input:** crisp value  $x_0 = 10000 \text{ min}^{-1}$  (not a fuzzy set!)

→ **fuzzyfication** = determine membership for each fuzzy set over  $X$

→ yields  $S' = (0, 0, \frac{3}{4}, \frac{1}{4}, 0)$  via  $x \alpha ( S_{vl}(x_0), S_l(x_0), S_m(x_0), S_h(x_0), S_{vh}(x_0) )$

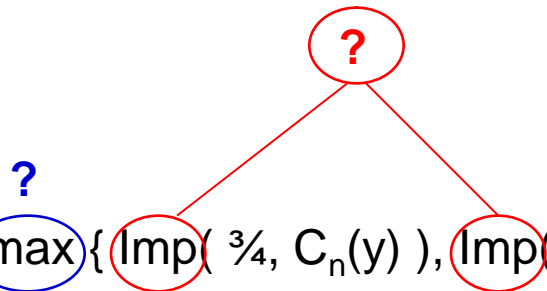
2. FITA: locale **inference** ⇒ since  $\text{Imp}(0,a) = 0$  we only need to consider:

$$S_m: C'_n(y) = \text{Imp}( \frac{3}{4}, C_n(y) )$$

$$S_h: C'_m(y) = \text{Imp}( \frac{1}{4}, C_m(y) )$$

3. **aggregation:**

$$C'(y) = \text{aggr} \{ C'_n(y), C'_m(y) \} = \text{max} \{ \text{Imp}( \frac{3}{4}, C_n(y) ), \text{Imp}( \frac{1}{4}, C_m(y) ) \}$$





**example:** (continued)

industrial drill machine → control of cooling supply

$$C'(y) = \max \{ \text{Imp}( \frac{3}{4}, C_n(y) ), \text{Imp}( \frac{1}{4}, C_m(y) ) \}$$

in case of control task typically no logic-based interpretation:

→ max-aggregation and

→ relation  $R(x,y)$  not interpreted as implication.

often:  $R(x,y) = \min(a, b)$  „Mamdani controller“

**thus:**

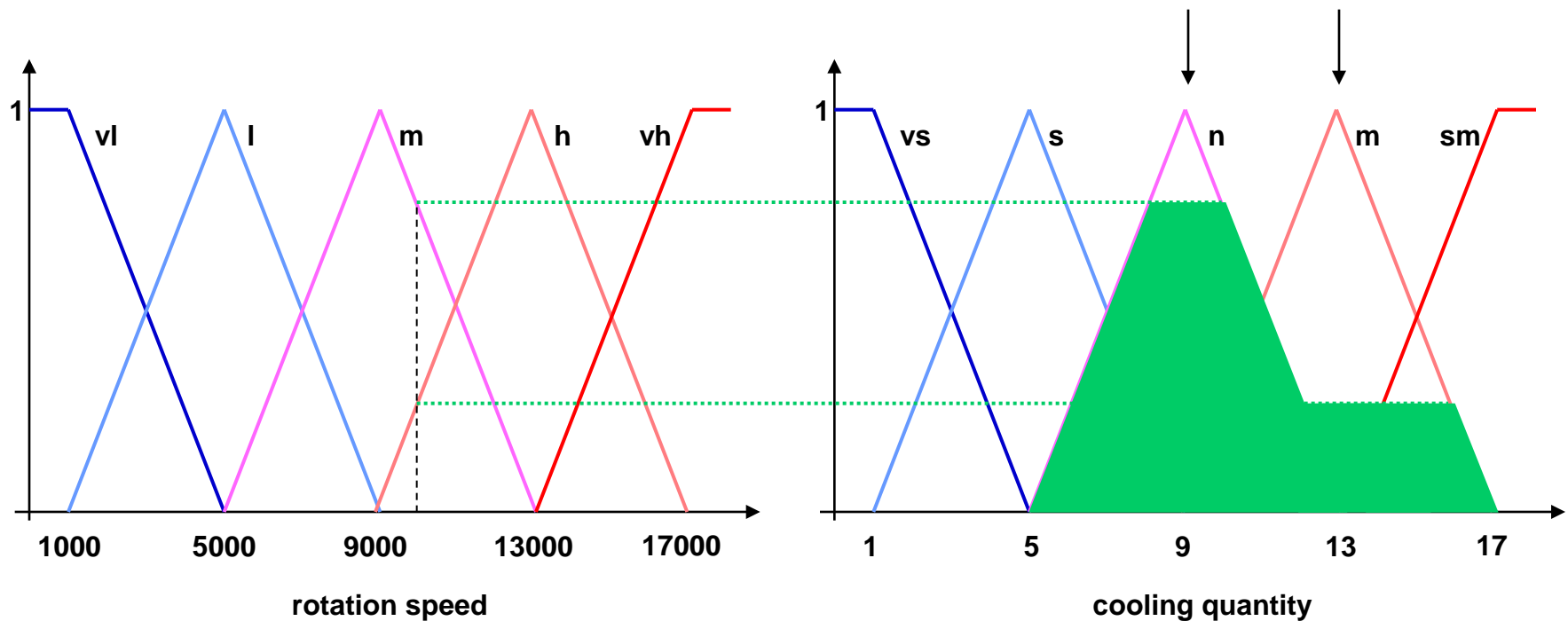
$$C'(y) = \max \{ \min \{ \frac{3}{4}, C_n(y) \}, \min \{ \frac{1}{4}, C_m(y) \} \}$$

→ graphical illustration

**example:** (continued)

industrial drill machine → control of cooling supply

$$C'(y) = \max \{ \min \{ \frac{3}{4}, C_n(y) \}, \min \{ \frac{1}{4}, C_m(y) \} \}, x_0 = 10000 \text{ [rpm]}$$



### **open and closed loop control:**

affect the dynamical behavior of a system  
in a desired manner

- **open loop control**

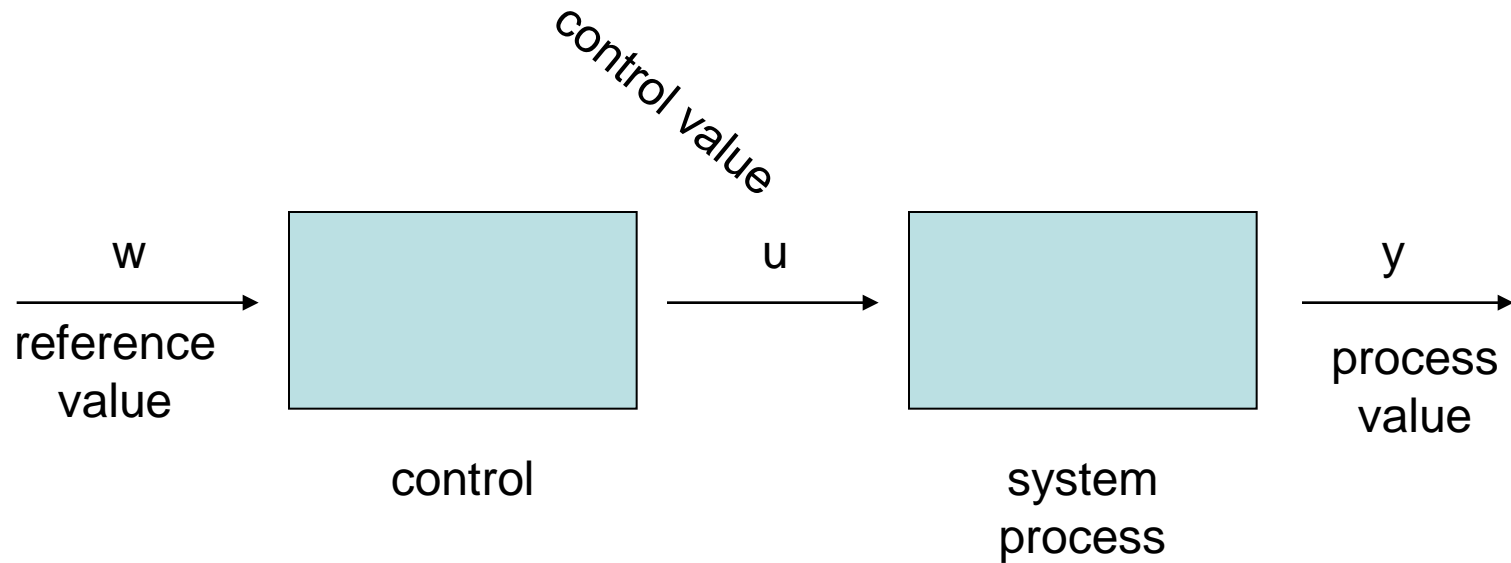
control is aware of reference values and has a model of the system  
⇒ control values can be adjusted,  
such that process value of system is equal to reference value

problem: noise! ⇒ deviation from reference value not detected

- **closed loop control**

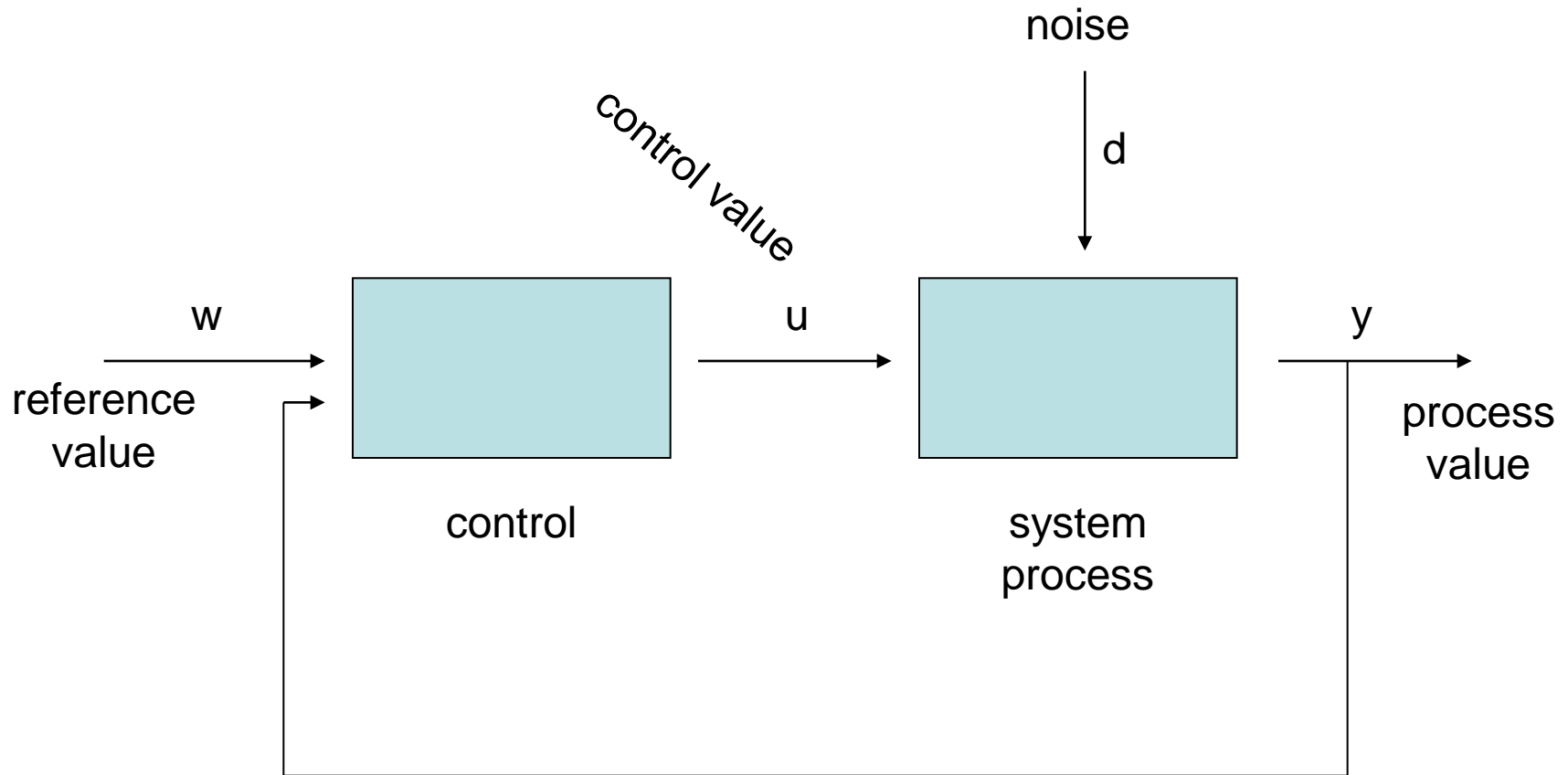
now: detection of deviations from reference value possible  
(by means of measurements / sensors)  
and new control values can take into account the amount of deviation

## open loop control



assumption: undisturbed operation  $\Rightarrow$  process value = reference value

closed loop control



control deviation = reference value – process value

### **required:**

model of system / process

→ as differential equations or difference equations (DEs)

→ well developed theory available

### **so, why fuzzy control?**

- there exists no process model in form of DEs etc.  
(operator/human being has realized control by hand)
- process with high-dimensional nonlinearities → no classic methods available
- control goals are vaguely formulated („soft“ changing gears in cars)

## fuzzy description of control behavior

IF X is  $A_1$ , THEN Y is  $B_1$

IF X is  $A_2$ , THEN Y is  $B_2$

IF X is  $A_3$ , THEN Y is  $B_3$

...

IF X is  $A_n$ , THEN Y is  $B_n$

X is  $A'$

---

Y is  $B'$

} similar to approximative reasoning

but fact  $A'$  is not a fuzzy set but a crisp input

→ actually, it is the current process value

fuzzy controller executes inference step

→ yields fuzzy output set  $B'(y)$

but crisp control value required for the process / system

→ defuzzification (= “condense” fuzzy set to crisp value)

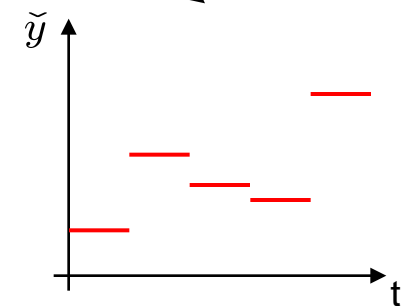
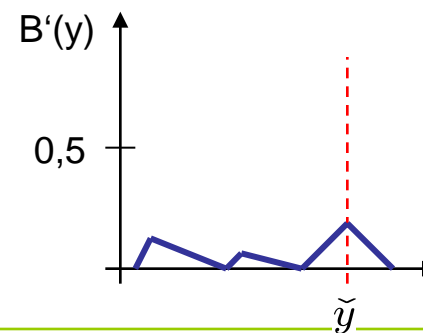
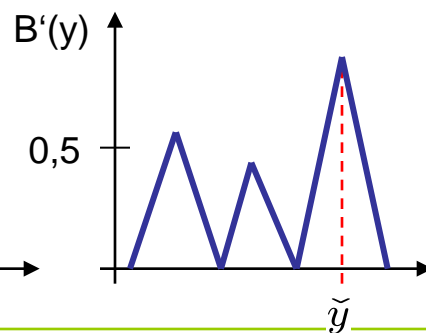
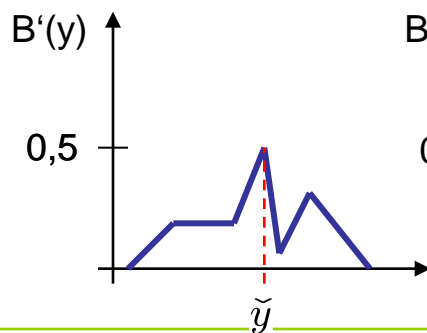
## defuzzification

**Def:** rule k active  $\Leftrightarrow A_k(x_0) > 0$

- maximum method

- only active rule with largest activation level is taken into account
  - suitable for pattern recognition / classification
  - decision for a single alternative among finitely many alternatives
- selection independent from activation level of rule (0.05 vs. 0.95)
- if used for control: incontinuous curve of output values (leaps)

$$\tilde{y} = \operatorname{argmax} B'(y)$$



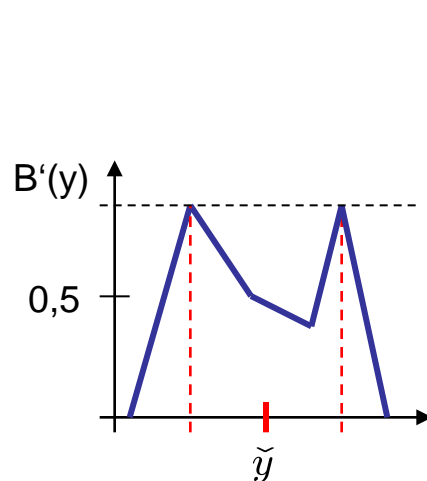


## defuzzification

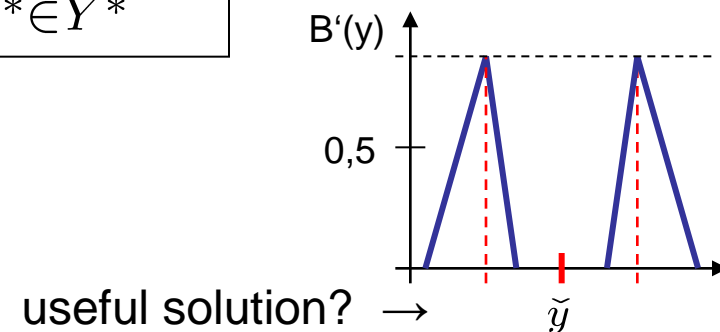
$$Y^* = \{ y \in Y: B'(y) = \text{hgt}(B') \}$$

- maximum mean value method

- all active rules with largest activation level are taken into account
  - interpolations possible, but need not be useful
  - obviously, only useful for neighboring rules with max. activation
- selection independent from activation level of rule (0.05 vs. 0.95)
- if used in control: incontinuous curve of output values (leaps)



$$\tilde{y} = \frac{1}{|Y^*|} \sum_{y^* \in Y^*} y^*$$



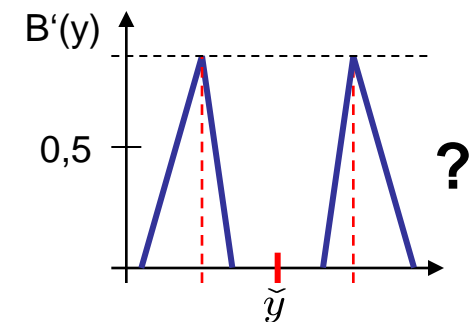
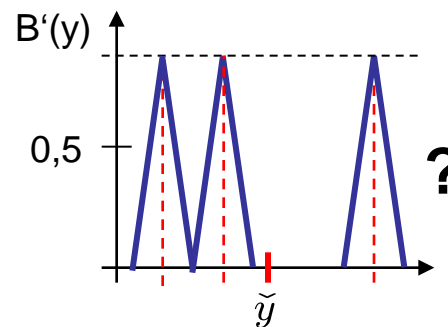
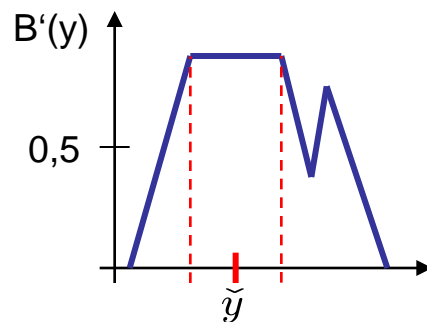
## defuzzification

$$Y^* = \{ y \in Y: B'(y) = \text{hgt}(B') \}$$

- center-of-maxima method (COM)

- only **extreme** active rules with largest activation level are taken into account
  - interpolations possible, but need not be useful
  - obviously, only useful for neighboring rules with max. activation level
- selection independent from activation level of rule (0.05 vs. 0.95)
- in case of control: incontinuous curve of output values (leaps)

$$\tilde{y} = \frac{\inf Y^* + \sup Y^*}{2}$$



## defuzzification

- Center of Gravity (COG)

- all active rules are taken into account

- but numerically expensive ...      ...only valid for HW solution, today!

- borders cannot appear in output ( ∃ work-around )

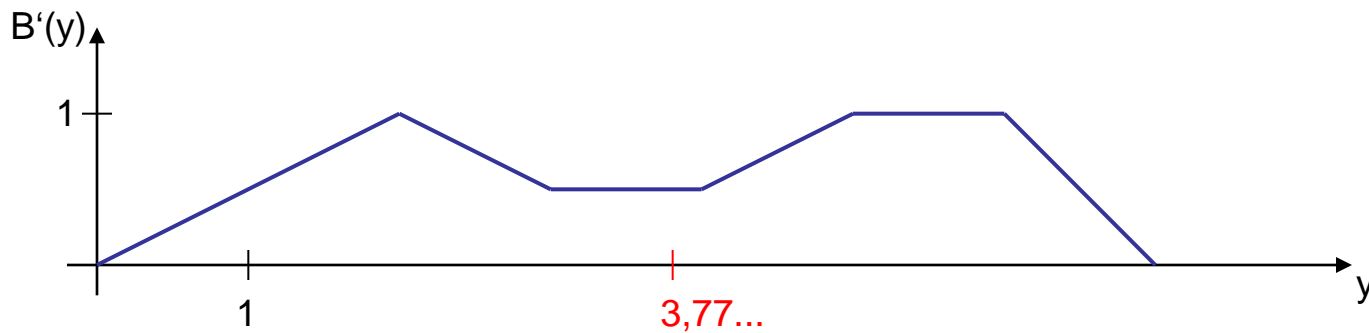
- if only single active rule: independent from activation level

- continuous curve for output values

$$\check{y} = \frac{\int y \cdot B'(y) dy}{\int B'(y) dy}$$

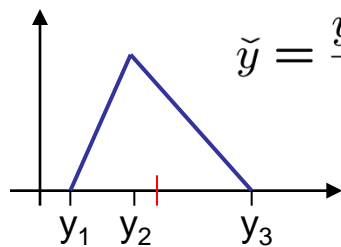
## Excursion: COG

$$\tilde{y} = \frac{\int y \cdot B'(y) dy}{\int B'(y) dy}$$



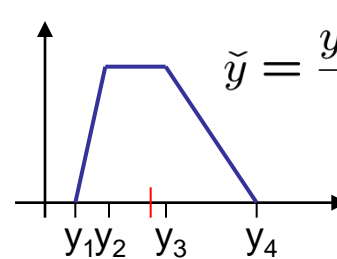
pendant in  
probability theory:  
expectation value

triangle:

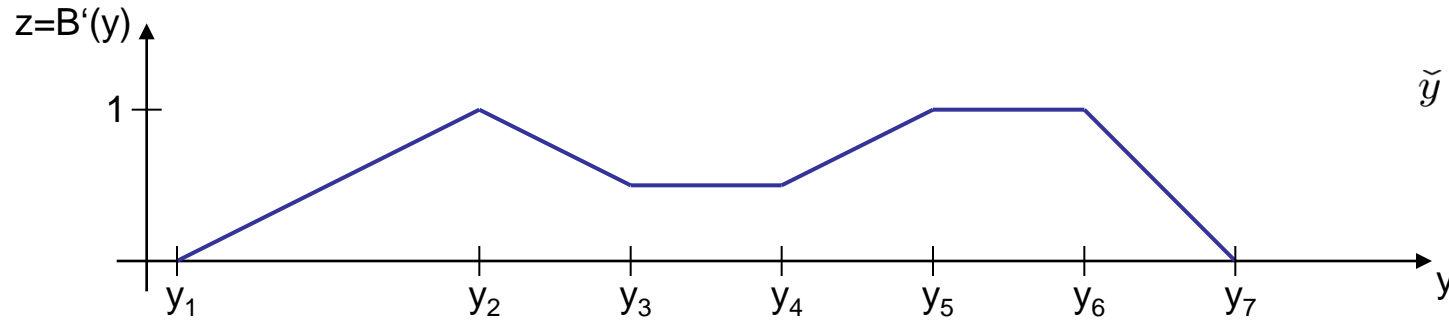


$$\tilde{y} = \frac{y_1 + y_2 + y_3}{3}$$

trapezoid:



$$\tilde{y} = \frac{y_4^2 + y_3^2 - y_2^2 - y_1^2 + y_3y_4 - y_1y_2}{3(y_4 + y_3 - y_2 - y_1)}$$



$$\tilde{y} = \frac{\int y \cdot B'(y) dy}{\int B'(y) dy}$$

assumption: fuzzy membership functions piecewise linear

output set  $B'(y)$  represented by sequence of points  $(y_1, z_1), (y_2, z_2), \dots, (y_n, z_n)$

⇒ area under  $B'(y)$  and weighted area can be determined additively piece by piece

⇒ linear equation  $z = m y + b$  ⇒ insert  $(y_i, z_i)$  and  $(y_{i+1}, z_{i+1})$

⇒ yields  $m$  and  $b$  for each of the  $n-1$  linear sections

$$\Rightarrow F_i = \int_{y_i}^{y_{i+1}} (m y + b) dy = \frac{m}{2} (y_{i+1}^2 - y_i^2) + b(y_{i+1} - y_i)$$

$$\Rightarrow G_i = \int_{y_i}^{y_{i+1}} y (m y + b) dy = \frac{m}{3} (y_{i+1}^3 - y_i^3) + \frac{b}{2} (y_{i+1}^2 - y_i^2)$$

$$\tilde{y} = \frac{\sum_i G_i}{\sum_i F_i}$$

## Defuzzification

- Center of Area (COA)
  - developed as an approximation of COG
  - let  $\hat{y}_k$  be the COGs of the output sets  $B'_k(y)$ :

$$\tilde{y} = \frac{\sum_k A_k(x_0) \cdot \hat{y}_k}{\sum_k A_k(x_0)}$$

### how to:

assume that fuzzy sets  $A_k(x)$  and  $B_k(x)$  are triangles or trapezoids

let  $x_0$  be the crisp input value

for each fuzzy rule “IF  $A_k$  is X THEN  $B_k$  is Y“

determine  $B'_k(y) = R(A_k(x_0), B_k(y))$ , where  $R(.,.)$  is the relation

find  $\hat{y}_k$  as center of gravity of  $B'_k(y)$