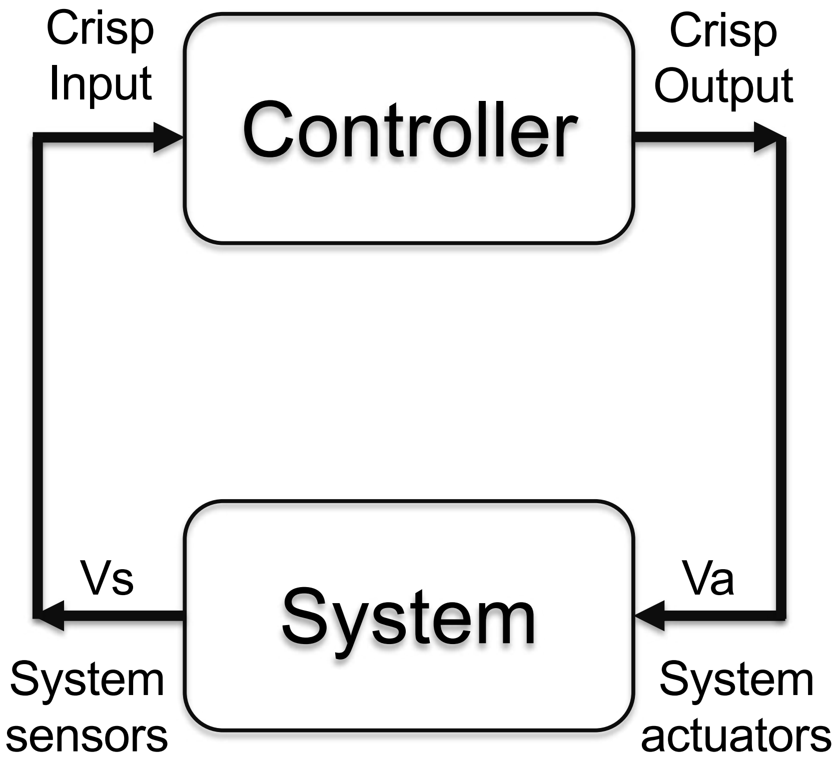
An introduction to Fuzzy Logic

An “easy & natural” way to control a system

Vincent Rodin

- Lab-STICC, UMR 6285, CNRS,
- Computer Science Dept,
- University of Brest, France

Motivations: control system



e.g.: camera control
distance & light (Vs)
➔ focus & aperture (Va)

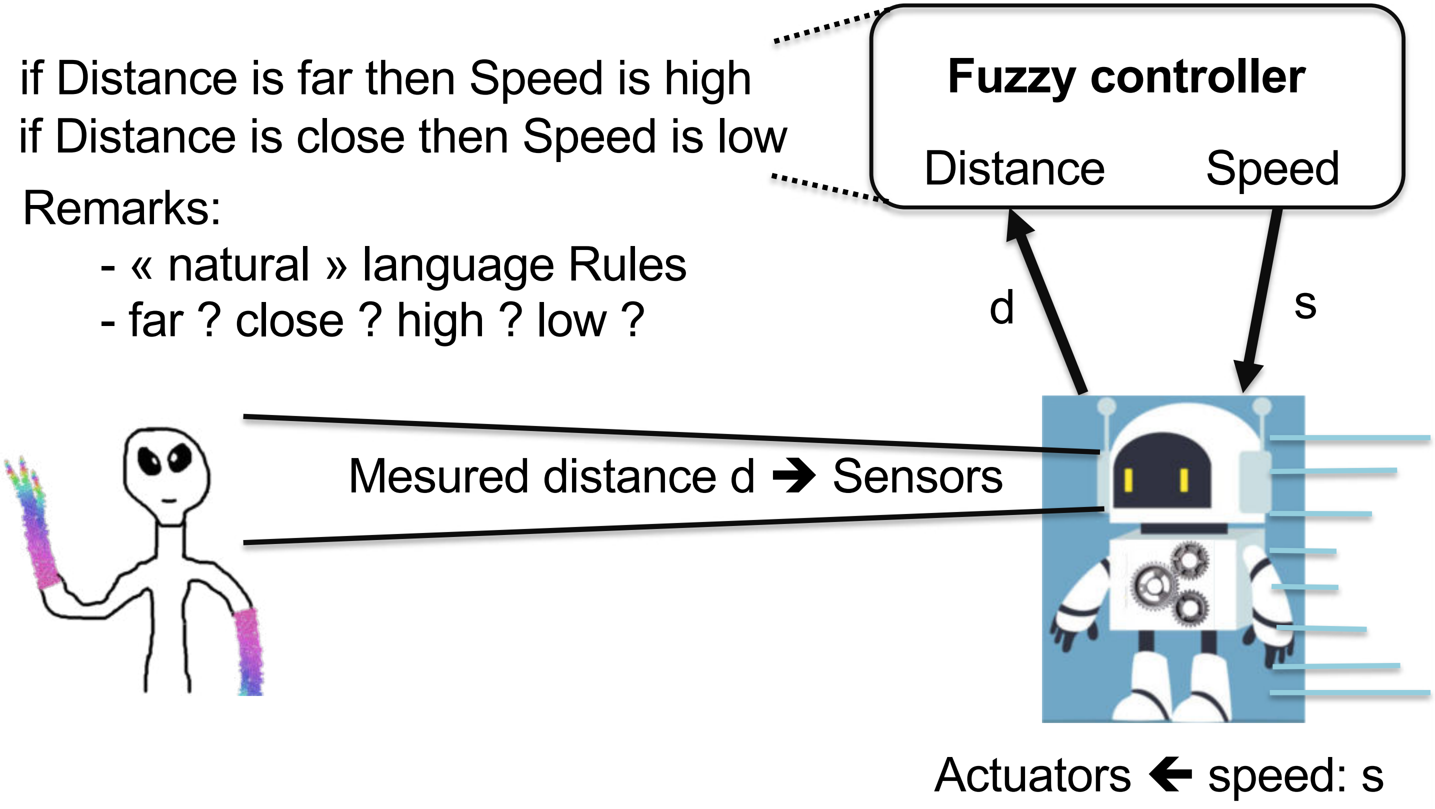
- Classical approach
  - Mathematical function
  - Not always possible
- Fuzzy approach
  - knowledge & experience
    ➔ « Natural » control
  - Not accurate
    ➔ uncertainly ⬅
    but acceptable !!

Fuzzy controller: an example

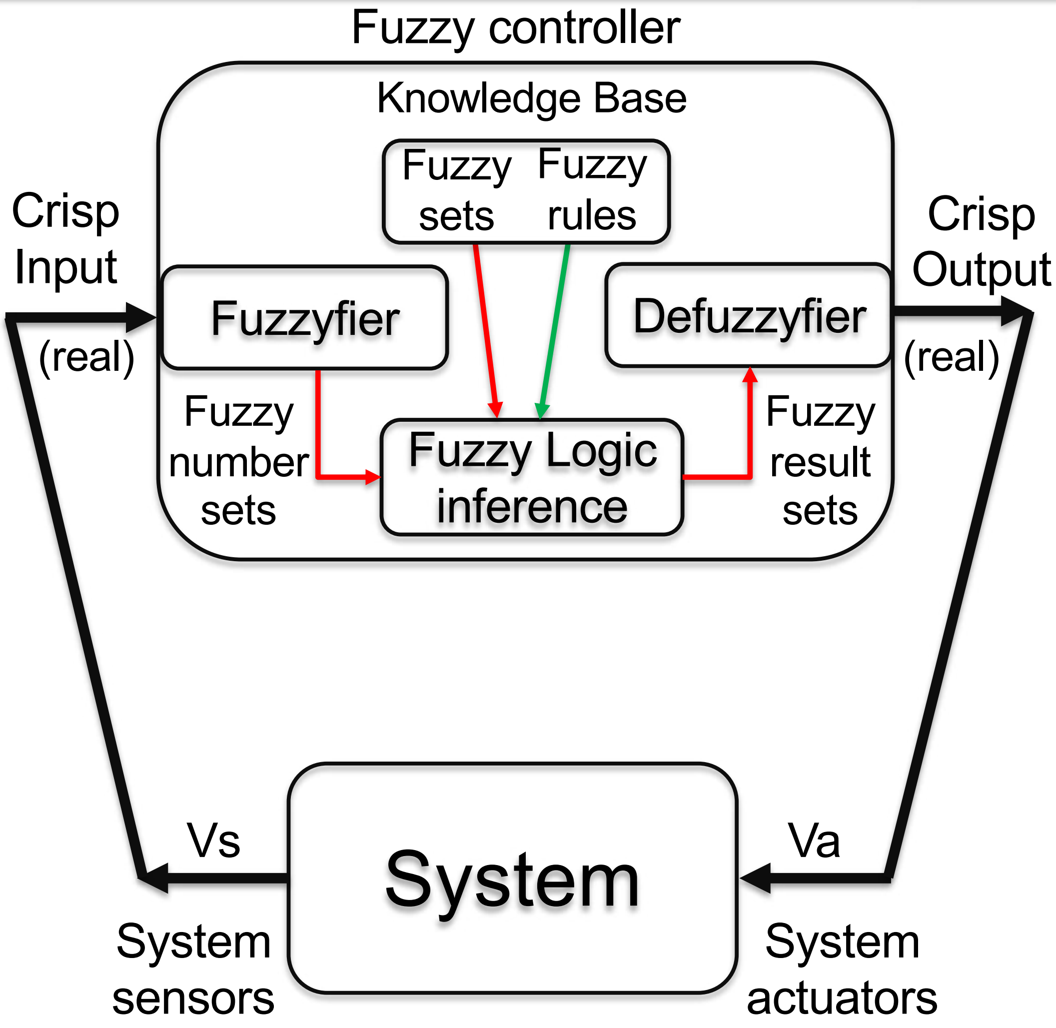
if Distance is far then Speed is high
if Distance is close then Speed is low

Remarks:

- « natural » language Rules
- far ? close ? high ? low ?



Fuzzy controller: structure



Four main parts

- <u>Knowledge base</u> (**expert**)
  If-Then Fuzzy “natural” rules and Fuzzy sets

- <u>Fuzzyfication module</u>
  Crisp **Input** Values (Vs)
  ➔ Fuzzy **input** sets
- <u>Fuzzy Inference engine</u>
  Knowledge base & Fuzzy **input** sets
  ➔ Human like reasonning
  ➔ Fuzzy **output** sets
- <u>Defuzzyfication module</u>
  Fuzzy **output** sets
  ➔ Crisp **output** Values (Va)

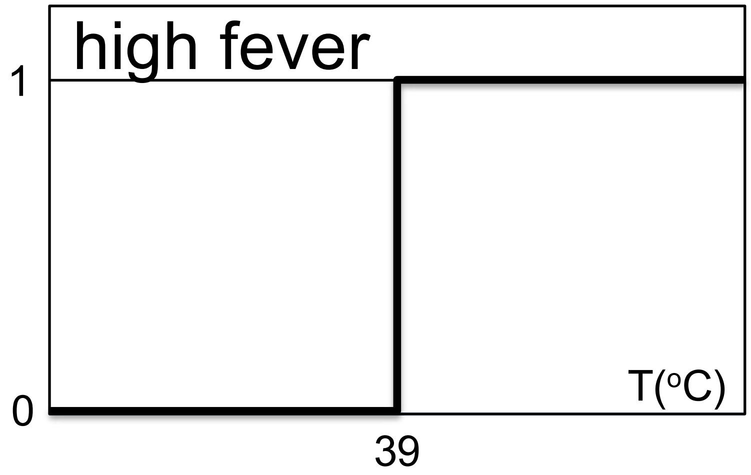
Road map

Fuzzy Logic: a short history

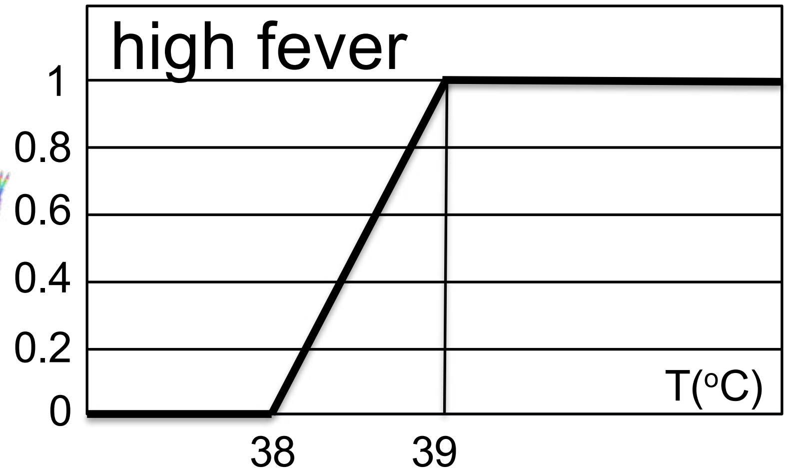
- 1965: Lotfi Zadeh (Berkeley University) defined the Fuzzy set theory ➔ core concepts of fuzzy logic
- 1973: Lotfi Zadeh proposed to apply fuzzy logic to system control
- 1974: Abe Mamdani (London University) proposed a fuzzy steam engine control (first « industrial » application)
- 1985: First general public products (Japan): Cameras, washing machines, etc. with « Fuzzy Logic Inside »
- 1990's: Widespread usage (daily life products….)
  ➔ WARP: Weight Associative Rule Processor

Fuzzy reasoning: a need of fuzzy sets

Classical set



Fuzzy set



Let us consider a patient with a **temperature 38.9 ^{O}C**
and the rule : **If Fever is high then Sickness is Covid-19**

Classical logic
➔ No covid-19

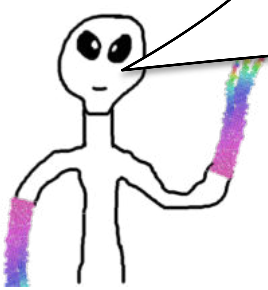
Fuzzy logic
➔ Possibly (0.9) Covid-19

Classical logic do not consider uncertainly of the real world

Fuzzy reasoning: warnings

Fuzzy Logic approach is:
- different from a classical scientific method (a priori)
- more pragmatic than deterministic

➔ Do not be too cartesian to use fuzzy logic
➔ Be intuitive (“expert”/human reasoning)
➔ Accept non-perfect results...



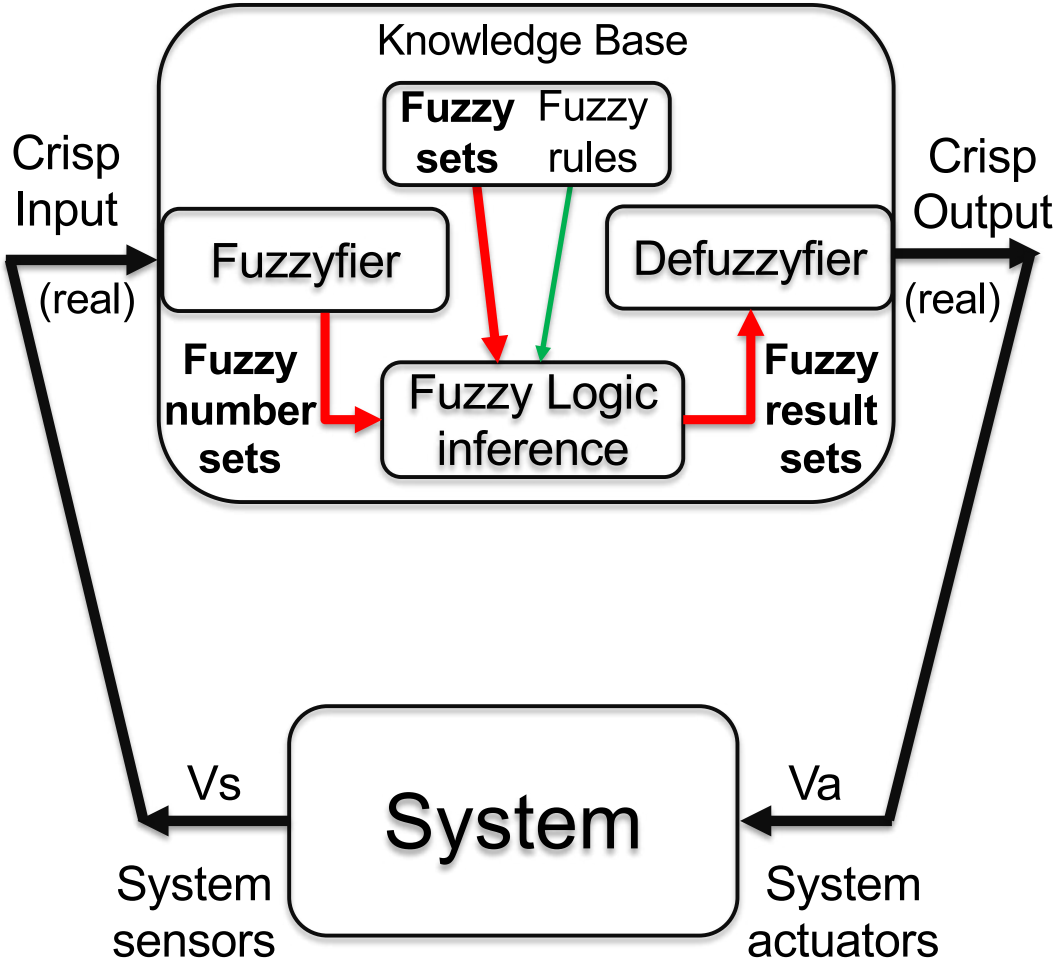
Fuzzy logic considers uncertainly of the real world

Fuzzy reasoning: applications

A wide range of applications

- Decision making (Business, defense, etc.)
- Diagnosis (medical area, fault detection, etc.)
- Database interface (fuzzy objects & fuzzy queries)
- Pattern recognition (medical, defense, autonomous cars, etc.)
- Robotics (robot arm & flexible link control, etc.)
- Industrial Process Control (cement kiln heat control, etc.)
- Daily life product control (air conditioning, camera, etc.)
- Etc.

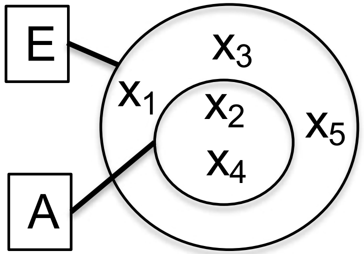
Fuzzy sets:
the main concept !



~~Fuzzy~~ Classical sets: representation

Classical set theory: $E = \{x_1, x_2, x_3, x_4, x_5\}$ Universe of discourse

$A = \{x_2, x_4\}$ A subset of E ($A \subset E$)



$\mu_A(x_1) = 0$
$\mu_A(x_2) = 1$
…

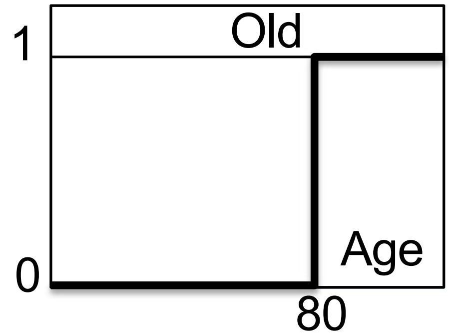
Let μ_A the membership function of set A

$\forall x \in E, \ \boldsymbol{\mu_A(x) \in \{0, 1\}}$

$$\mu_A(x) = \begin{cases} 0, & \text{if } x \notin A \\ 1, & \text{if } x \in A \end{cases}$$

$\mu_A(x)$: membership (value) of x for set A

Classical set example



Too precise for reasoning !!

Fuzzy sets: representation

Fuzzy set therory: E Universe of discourse

A subset of E ($A \subset E$)

Possible interpretation	
$\mu_A(x)$	$x \in A$
1	yes
[0.6;1[	possibly yes
[0.3;0.6[	cannot say
]0;0.3[	possibly no
0	no

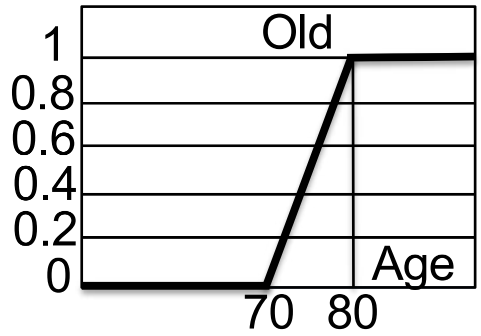
Let μ_A the membership function of A

$\forall x \in E, \ \boldsymbol{\mu_A(x) \in [0, 1]}$

$\mu_A(x)$: membership degree of x for set A

If $\mu_A(x)=0.5$, x belongs to A with a membership degree of 50%

Fuzzy set example

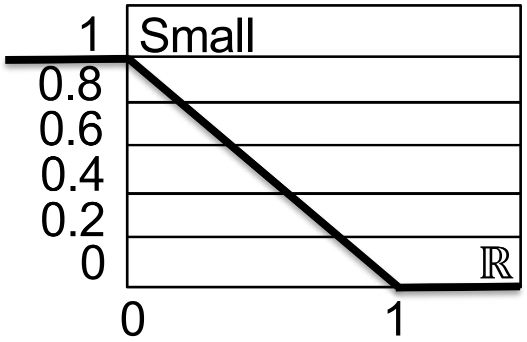


Uncertainly for reasoning !!

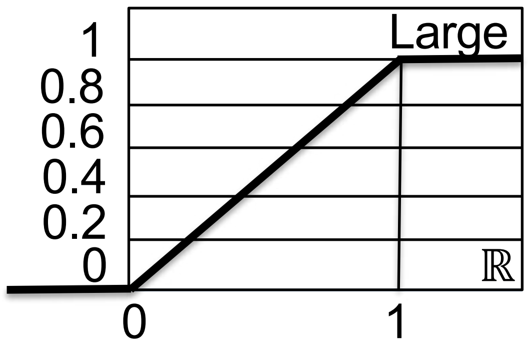
Fuzzy sets: representation

Examples:

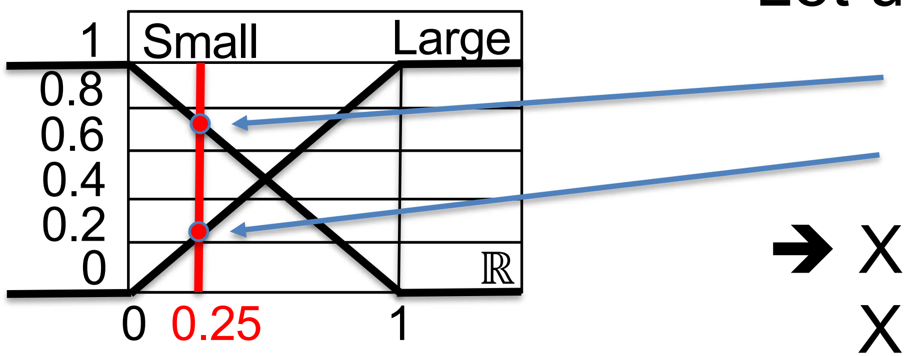
Fuzzy set Small



Fuzzy set large



2-set fuzzy partition of ℝ



Let us consider X=0.25

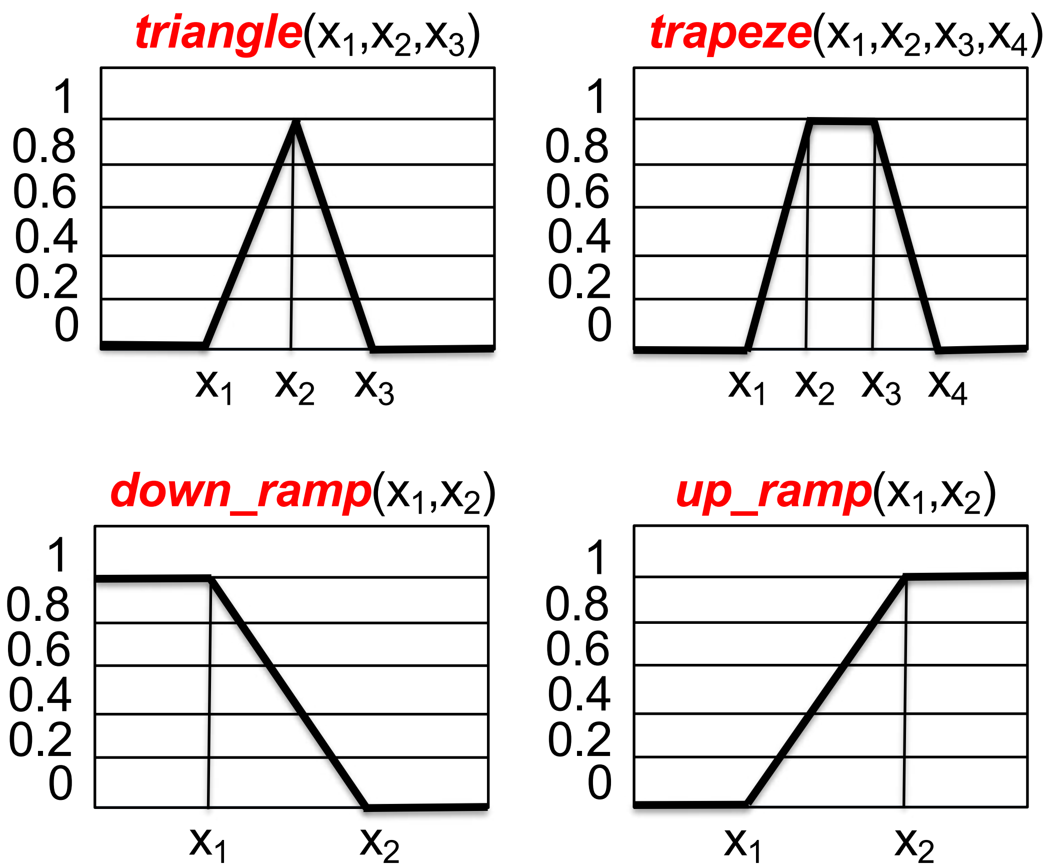
$\mu_{\text{Small}}(X) = 0.75$

$\mu_{\text{Large}}(X) = 0.25$

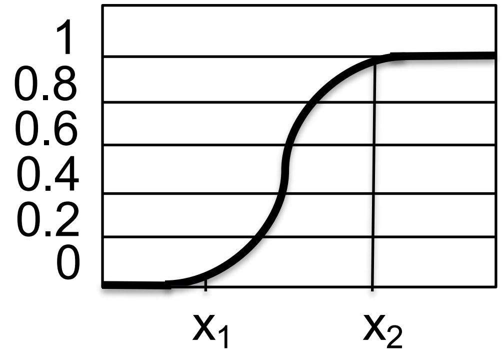
➔ X is Small at 75%

X is Large at 25%

Fuzzy sets: mostly used representations



Remark 1:
due to computation, smooth shapes



are very less used

Remark 2:
classical sets are special cases of fuzzy sets

Fuzzy sets: fuzzy operators

Fuzzy operators define relation between sets
➔ Intersection (and), union (or), complement (not), etc.

Triangular norm and co-norms are used…

Definition 1, a triangular norm is a function $*$: [0,1]x[0,1] ➔ [0,1]

Some t-norm properties:
Limit: $0*a = a*0 = 0$; $1*a = a*1 = a$
Commutativity: $a*b = b*a$
Associativity: $(a*b)*c = a*(b*c)$

t-norme examples:
min: min(a,b) ;
product: a • B ;
etc.

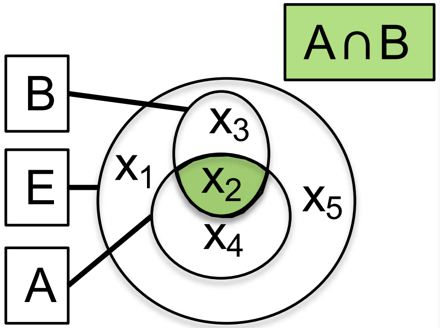
Definition 2, a triangular co-norm is a function $\dot{+}$: [0,1]x[0,1] ➔ [0,1]

Some t-conorm properties:
Limit: $0\dot{+}a = a\dot{+}0 = a$; $1\dot{+}a = a\dot{+}1 = 1$
Commutativity: $a\dot{+}b = b\dot{+}a$
Associativity: $(a\dot{+}b)\dot{+}c = a\dot{+}(b\dot{+}c)$

t-conorm examples:
max: max(a,b) ;
sum: $a \oplus b = \min(1, a+b)$;
etc.

Fuzzy sets: intersection (and)

Classical sets

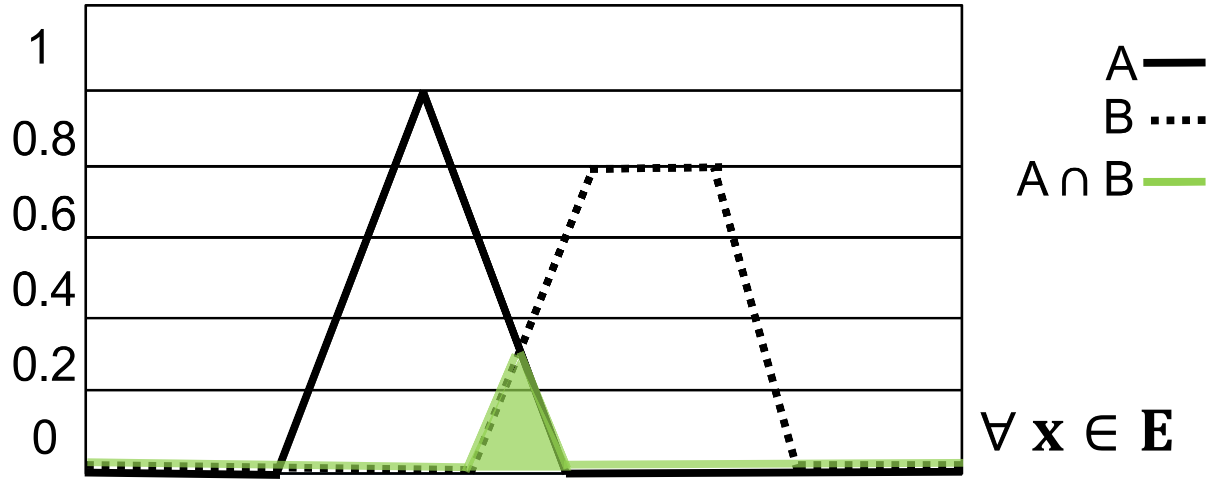


$$A \cap B: \ \forall x \in E, \ \mu_{A\cap B}(x) = \mu_A(x) * \mu_B(x)$$

If t-norm $* \equiv \min$ then $A \cap B$:

$$\forall x \in E, \ \mu_{A\cap B}(x) = \min(\mu_A(x), \mu_B(x))$$

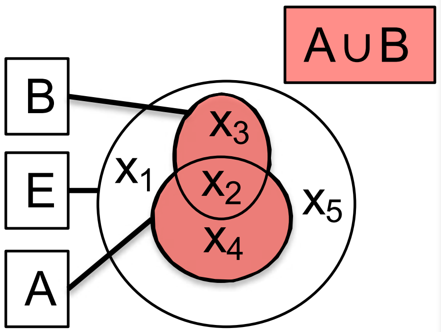
Example



Other t-norms can also be chosen….

Fuzzy sets: union (or)

Classical sets

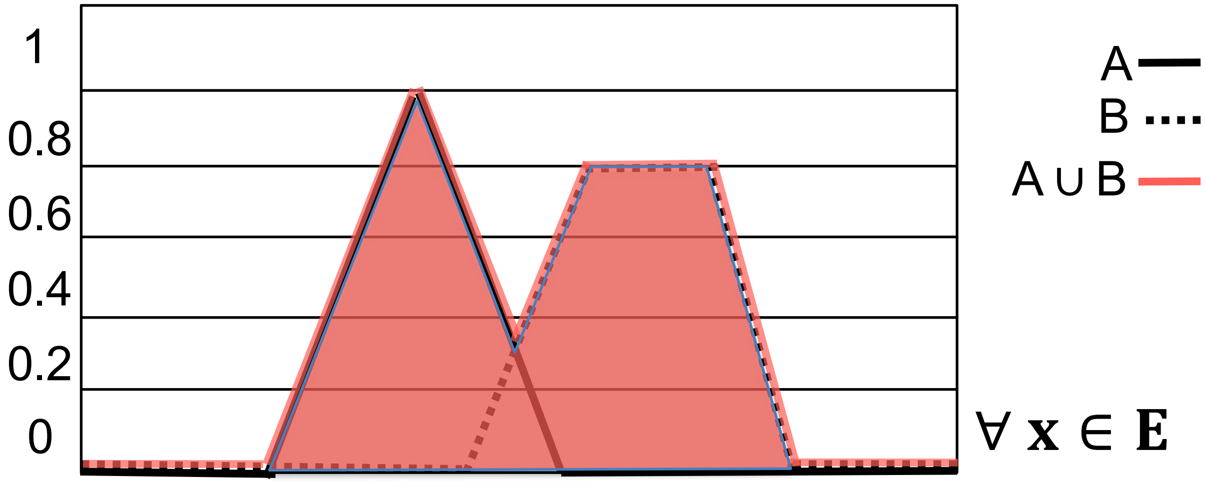


$$A \cup B : \ \forall x \in E, \ \mu_{A\cup B}(x) = \mu_A(x) \dot{+} \mu_B(x)$$

If t-conorm $\dot{+} \equiv \max$ then $A \cup B$:

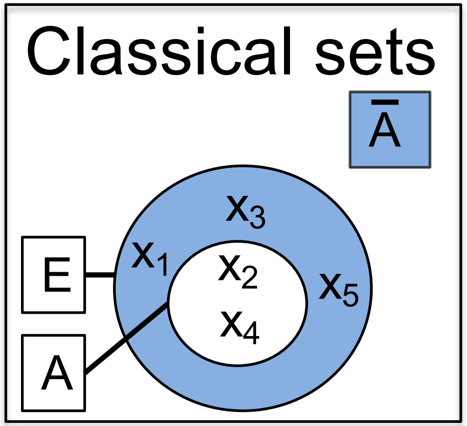
$$\forall x \in E, \ \mu_{A\cup B}(x) = \max(\mu_A(x), \mu_B(x))$$

Example



Other t-conorms can also be chosen….

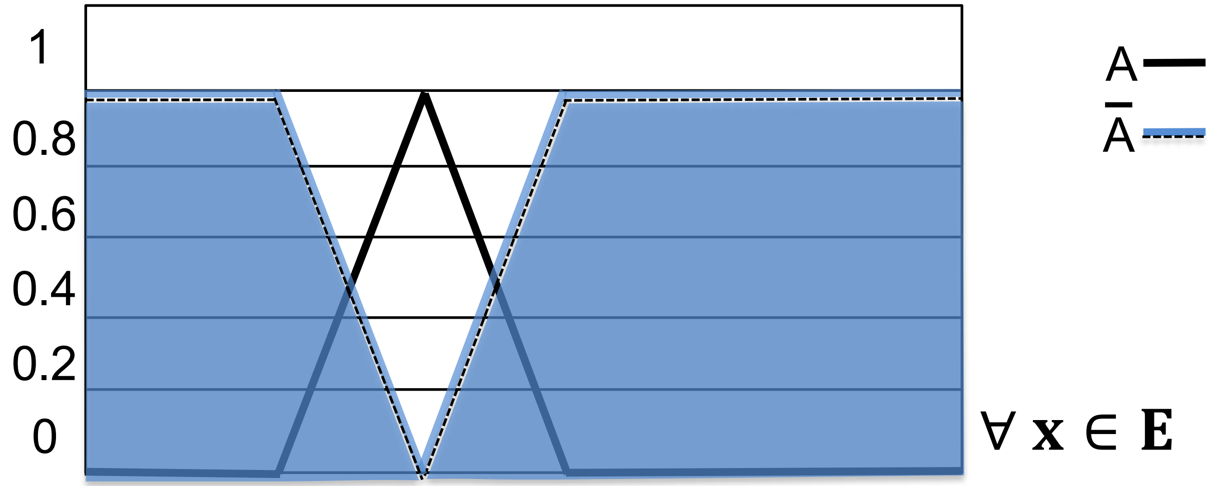
Fuzzy sets: complement (not)



$$\bar{A}: \forall x \in E, \mu_{\bar{A}}(x) = 1 - \mu A(x)$$

Used to describe a set negation

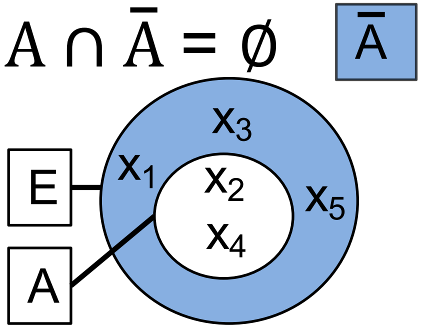
Example



Fuzzy sets: intersection & complement

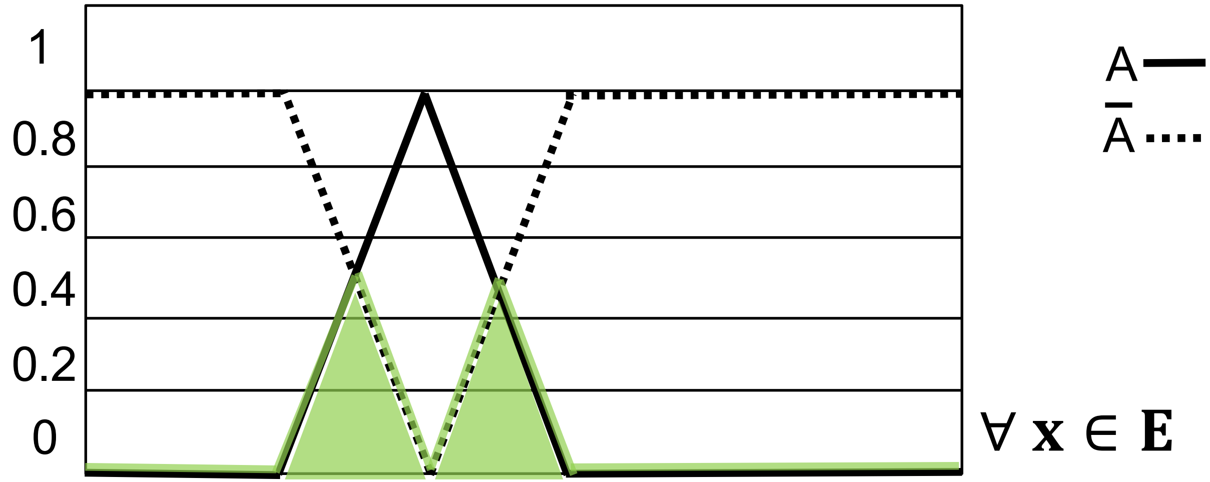
Classical sets

$A \cap \bar{A} = \emptyset$



Remark: with fuzzy sets, $A \cap \bar{A} = \emptyset$?

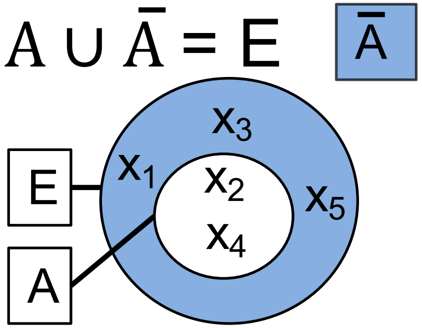
➔ $\mathbf{A \cap \bar{A} \neq \emptyset}$



Fuzzy sets:
union & complement

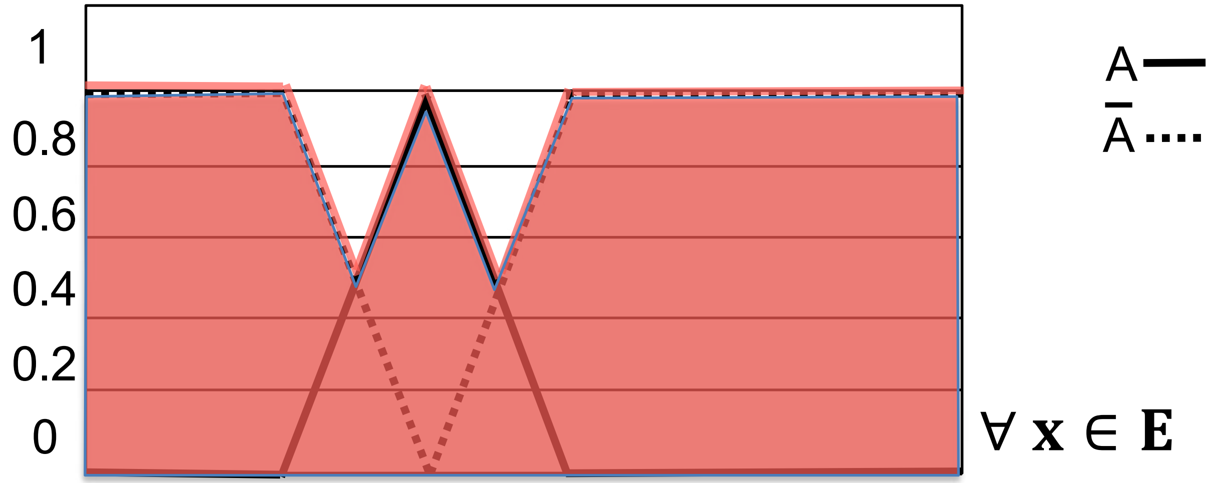
Classical sets

$A \cup \bar{A} = E$



Remark: with fuzzy sets, $A \cup \bar{A} = E$?

➔ $\mathbf{A \cup \bar{A} \neq E}$

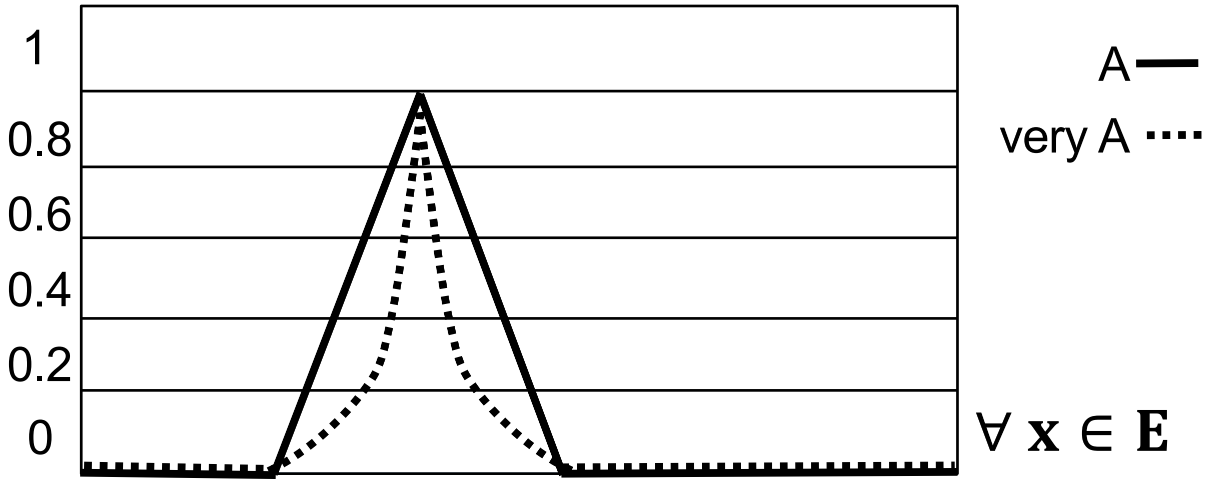


Fuzzy sets: modifier (very)

$$\text{very A: } \forall\, x \in E, \quad \mu_{\text{very A}}(x) = \mu_A(x)^2$$

Used to describe a more specific set

Example



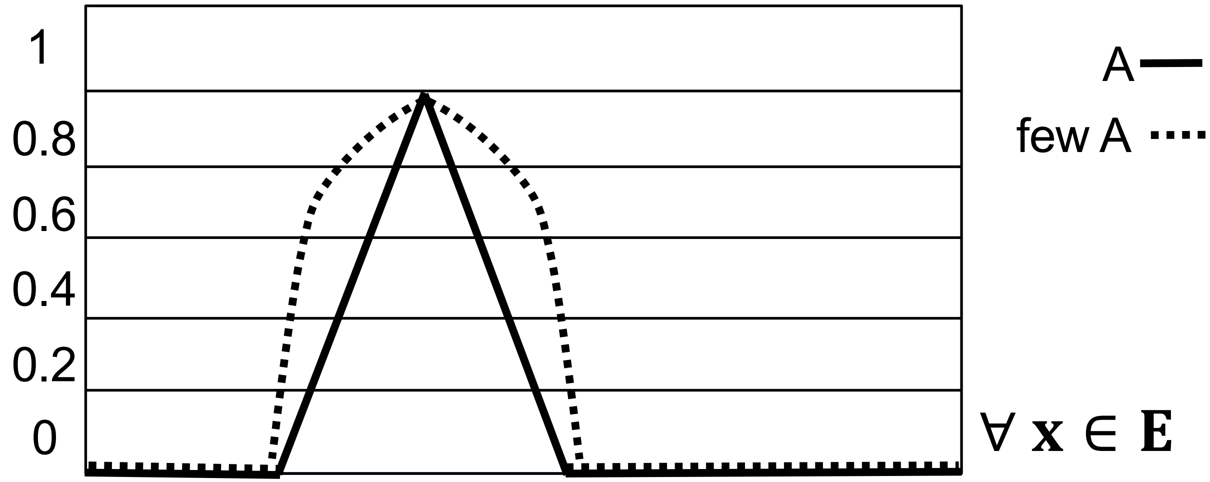
Remember: $\forall\, x \in E, \quad \boldsymbol{\mu_A(x)} \in [\mathbf{0}, \mathbf{1}]$

Fuzzy sets: modifier (few)

$$\text{few A}: \forall x \in E, \quad \mu_{\text{few A}}(x) = \sqrt{\mu_A(x)}$$

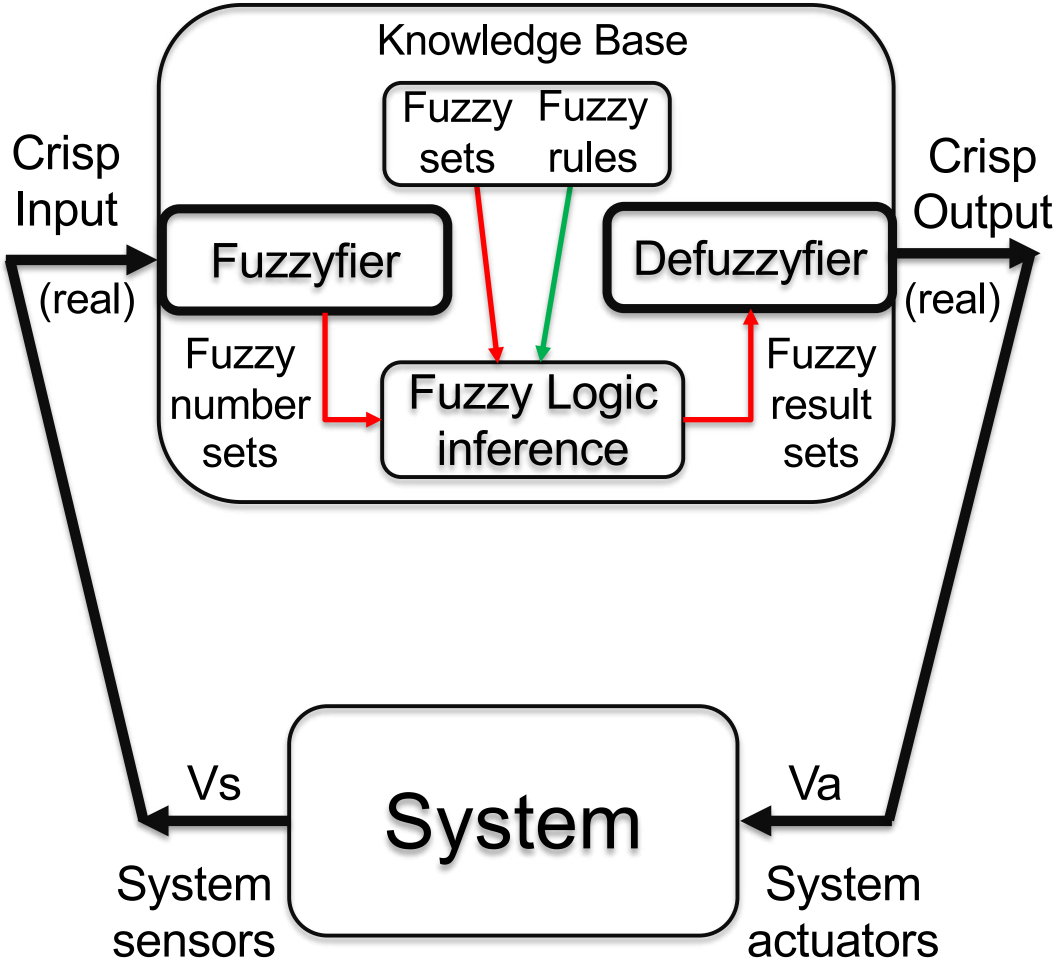
Used to describe a less specific set

Example



Remember: $\forall x \in E, \quad \boldsymbol{\mu_A(x)} \in [\mathbf{0}, \mathbf{1}]$

Fuzzy sets: fuzzyfication/defuzzyfication



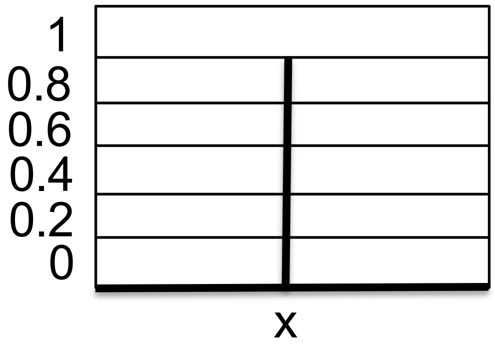
Fuzzy sets: fuzzyfication

Crisp value (real number) from a sensor → **Fuzzyfication module** → Fuzzy set (fuzzy number)

Which fuzzy set can represent a real number x ?

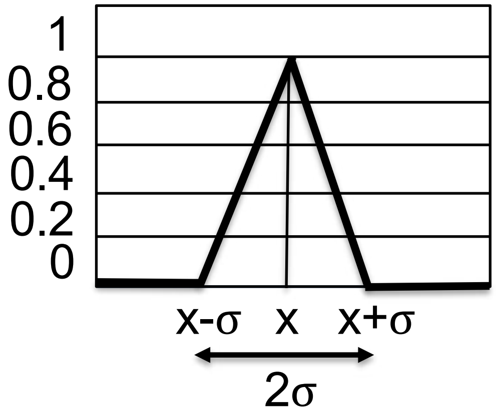
Let us consider:
- x , mesured value by the sensor
- σ, measurement standard deviation

Accurate data

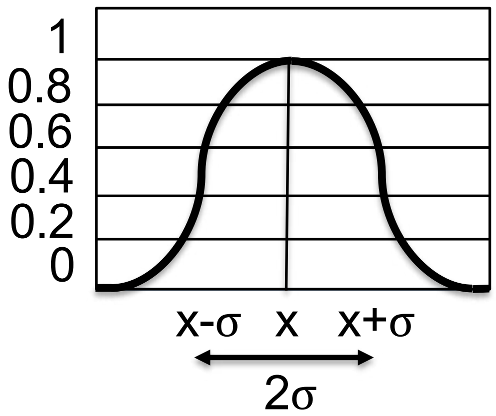


$\sigma = 0 \rightarrow$ Singleton

Imprecise data



Imprecise data

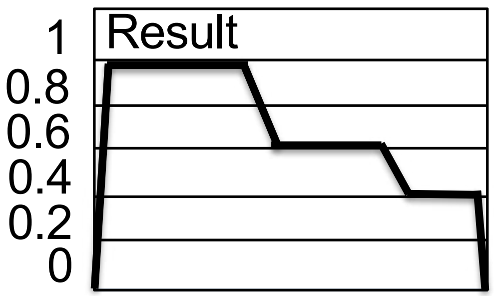


Fuzzy sets: defuzzyfication

Fuzzy set → **Defuzzyfication module** → Crisp value (real number) for an actuator

Which real number x better represents a fuzzy set ?

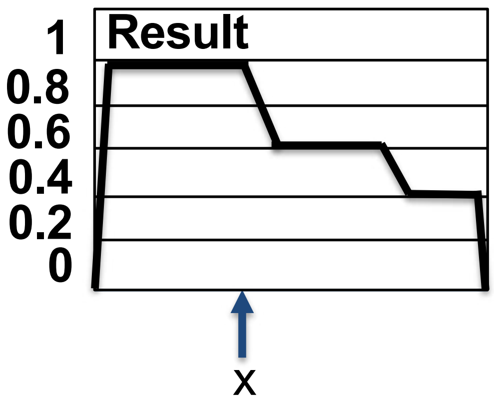
Let Result



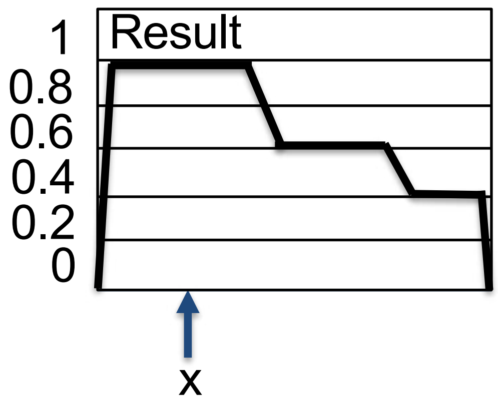
an **inference output** fuzzy set

Two major defuzzyfication methods

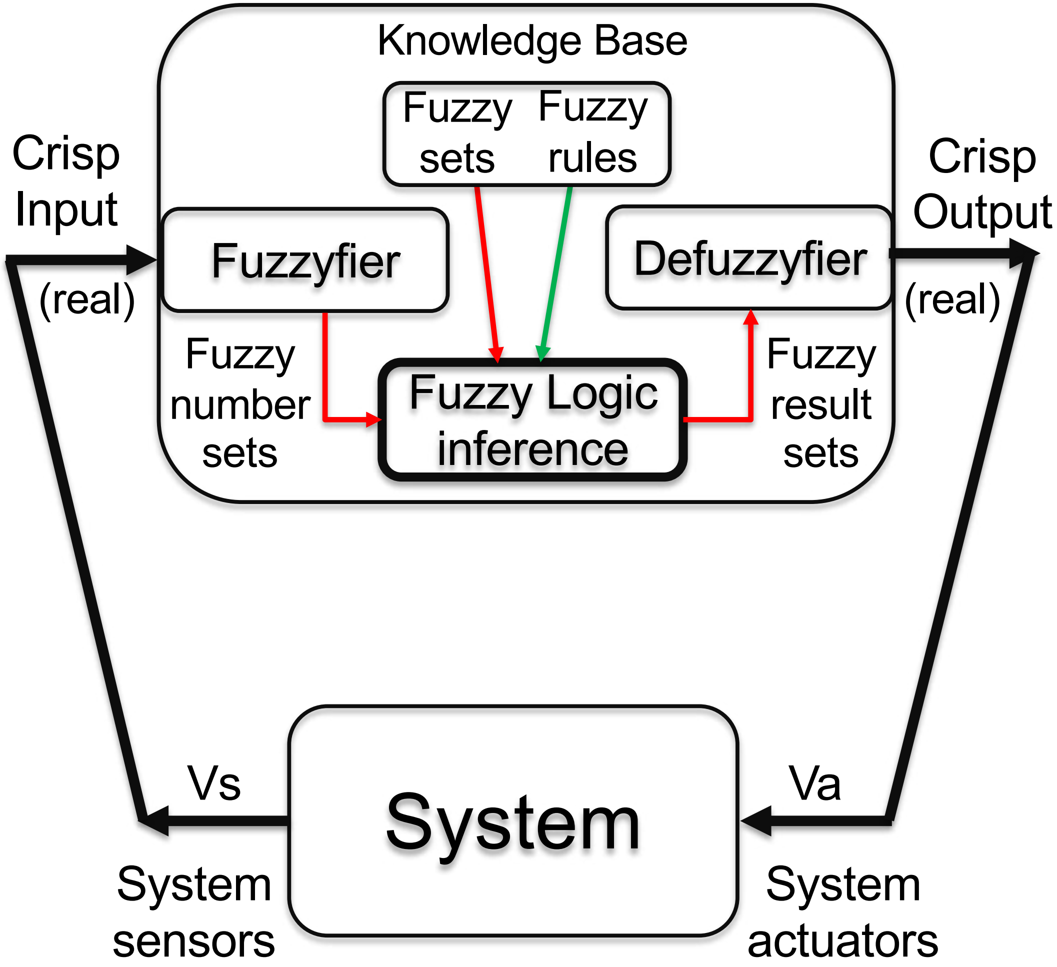
center of gravity



max_average



Fuzzy logic inference: the second main concept !

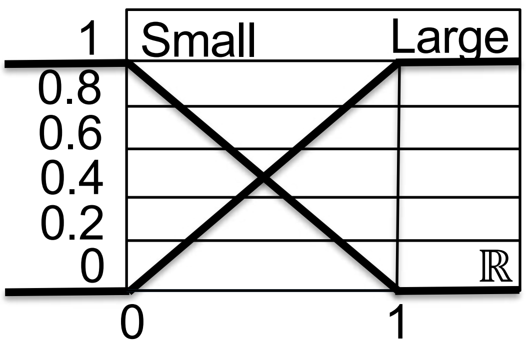


Fuzzy logic inference: example of knowledge base

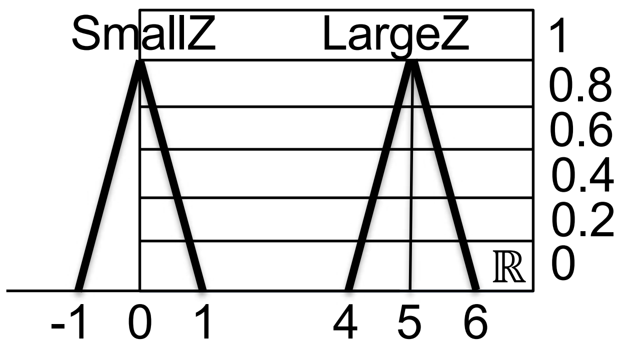
Fuzzy rules: if X is Small then Z is SmallZ (Rule 1)

if X is Large then Z is LargeZ (Rule 2)

Fuzzy input sets

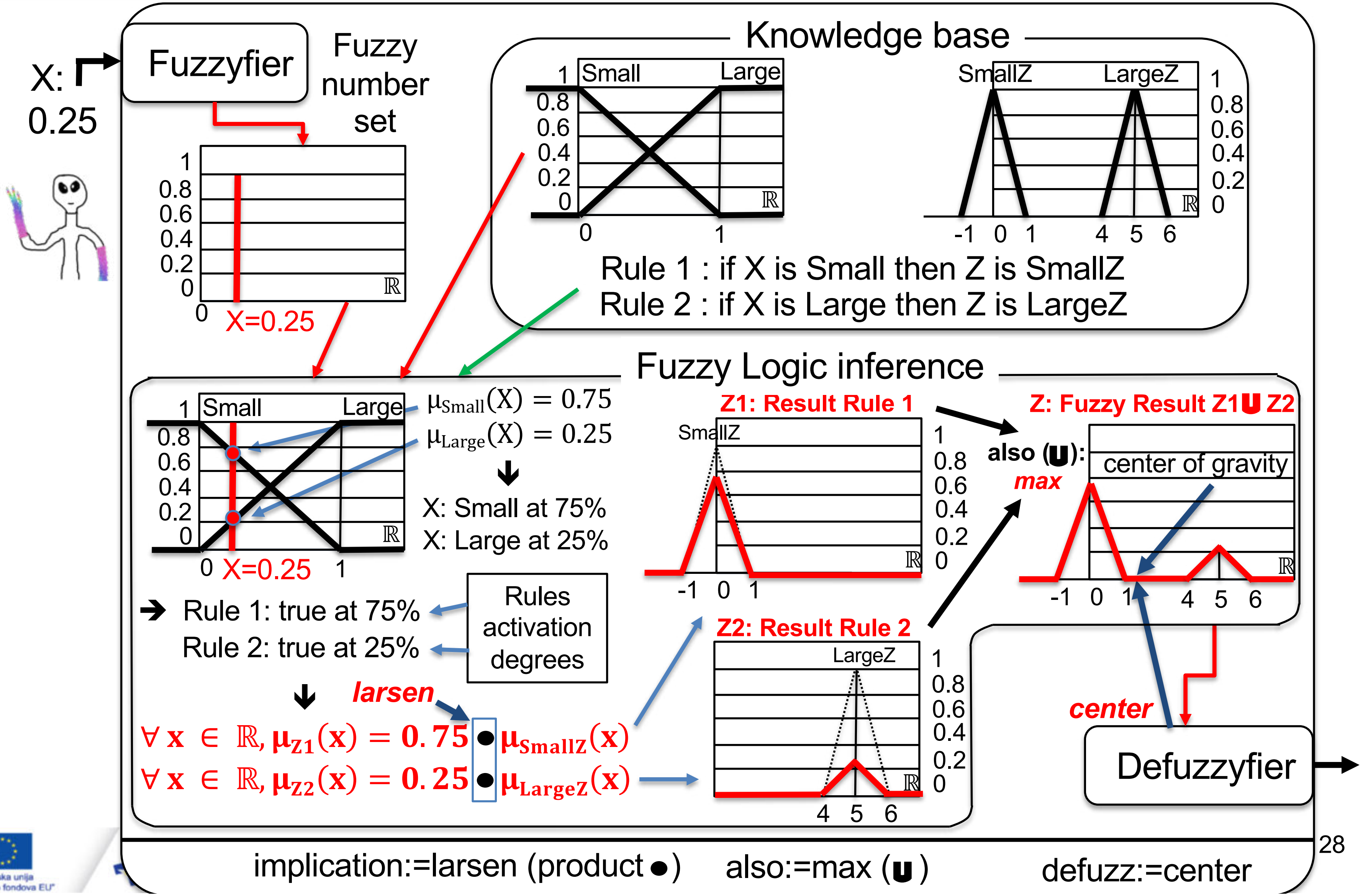


Fuzzy output sets



$X \in \mathbb{R}$ → Fuzzy controller → $Z \in \mathbb{R}$?

Fuzzy logic inference: example of a (larsen: •) controller

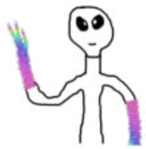


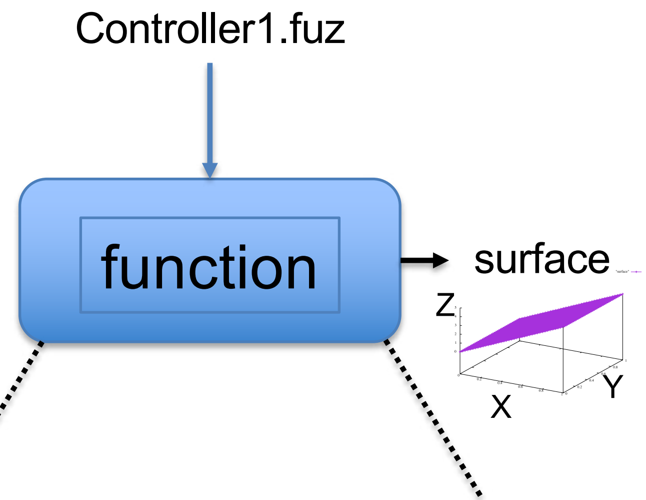
Fuzzy logic inference: example of a (larsen: •) controller

Controller1.fuz

```
configuration                  // A larsen controller
{
  and := min;                  // min, product
  or := max;                   // max, sum
  defuzz := center;            // center, max_average
  implication := larsen;       // larsen, mamdani
  also := max;                 // max, sum
}
linguistic
{
  // Input sets: Linguistic values for X and Y (Y not used)
  Small := down_ramp(0.0, 1.0);
  Large := up_ramp(0.0, 1.0);

  // Output sets: Linguistic values Z
  SmallZ  := triangle(-1, 0,1);
  LargeZ  := triangle(4,5,6);
}
rules
{
  if X is Small then Z is SmallZ;
  if X is Large then Z is LargeZ;
}
```

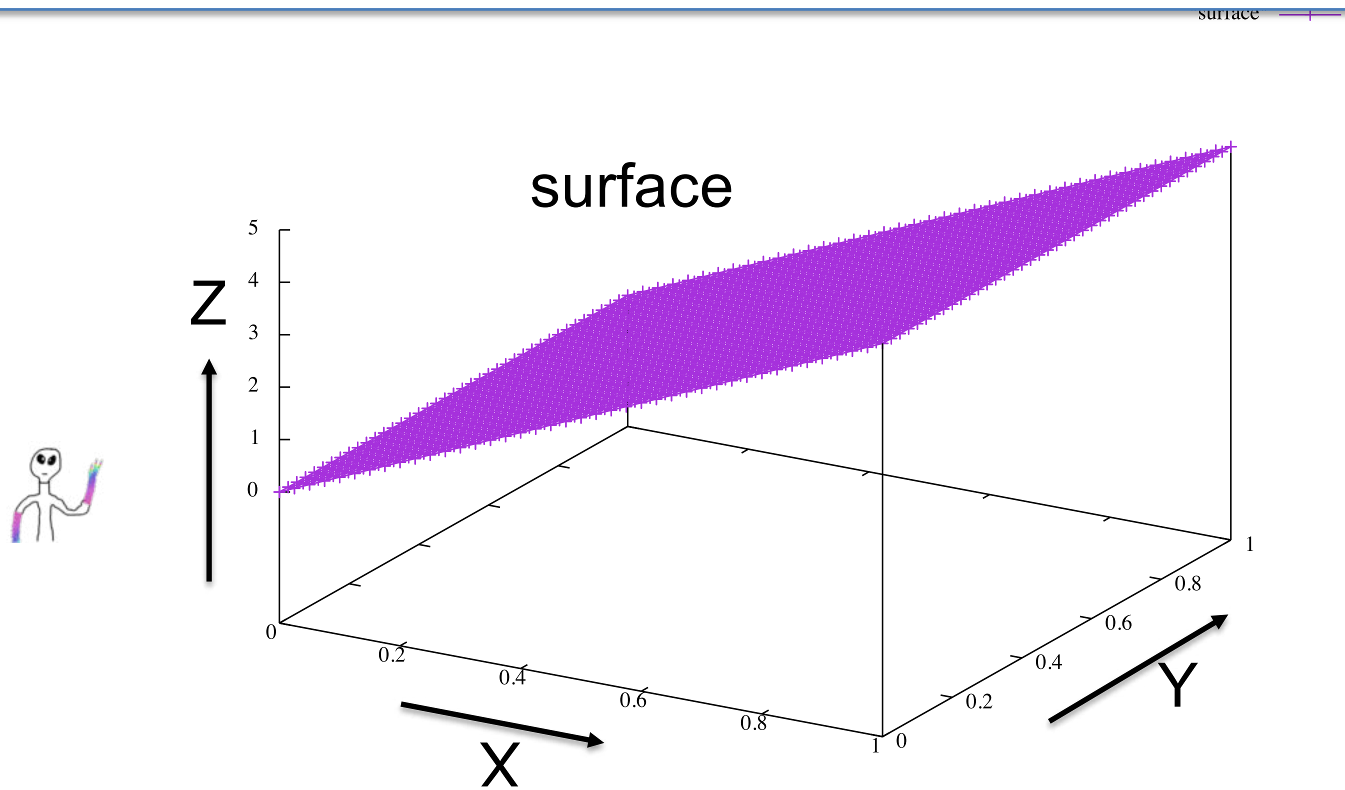




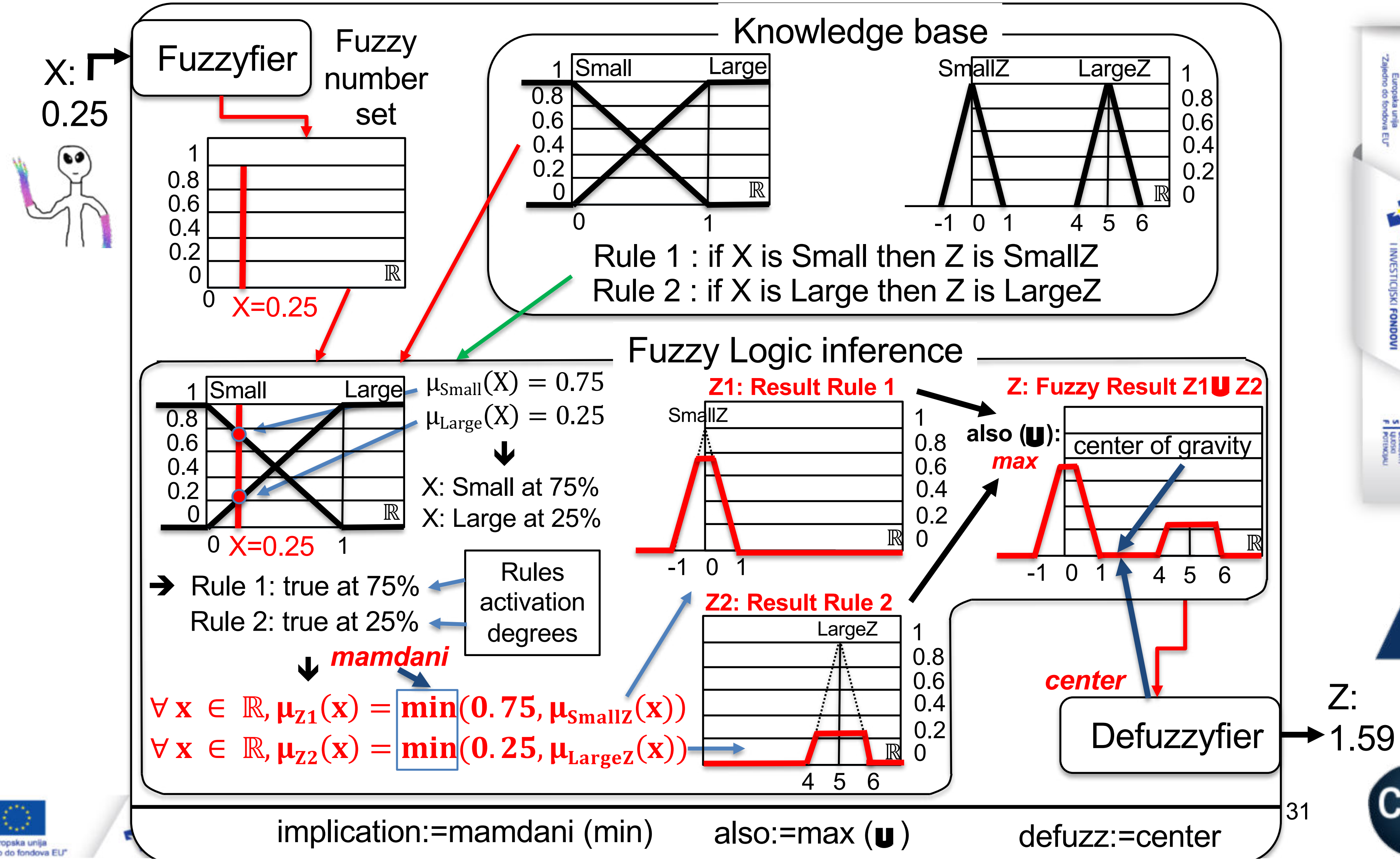

```
For(X=0;X<=1.0;X=X+0.1)
{
  For(Y=0;Y<=1.0;Y=Y+0.1)
  {
    X → Fuzzy controller Controller1.fuz → Z
    Y →
  }
}
```

Fuzzy logic inference: example of a (larsen: •) controller

Terminal
student@student ~/Desktop/FuzzyLogic/FuzzyFonction $ ./function Controller1.fuz



Fuzzy logic inference: example of a (mamdani: min) controller

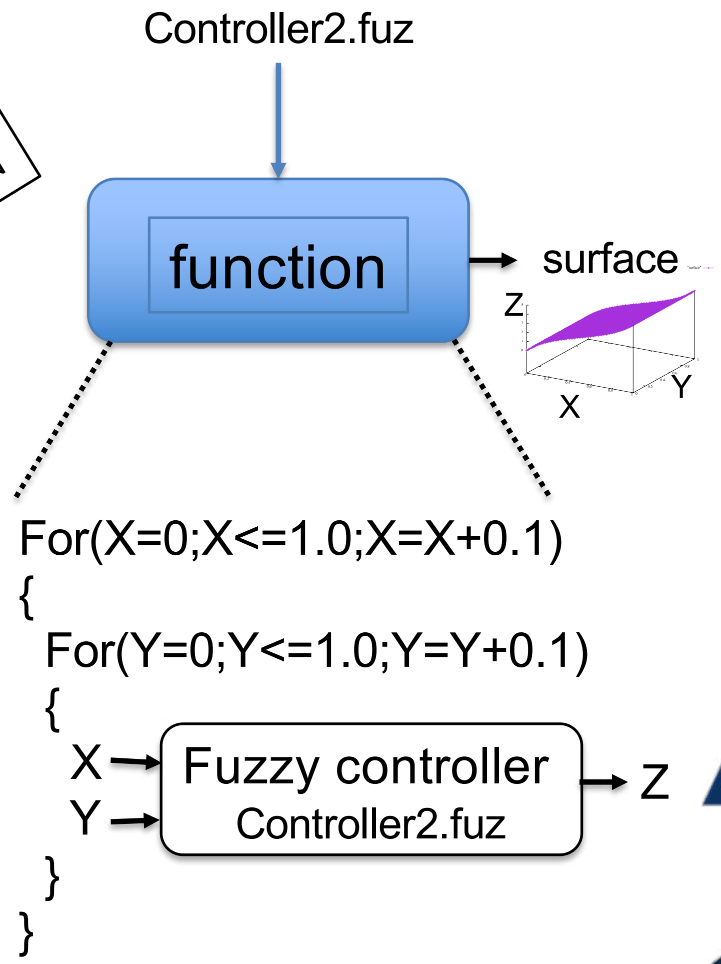


Fuzzy logic inference: example of a (mamdani: min) controller

Controller2.fuz

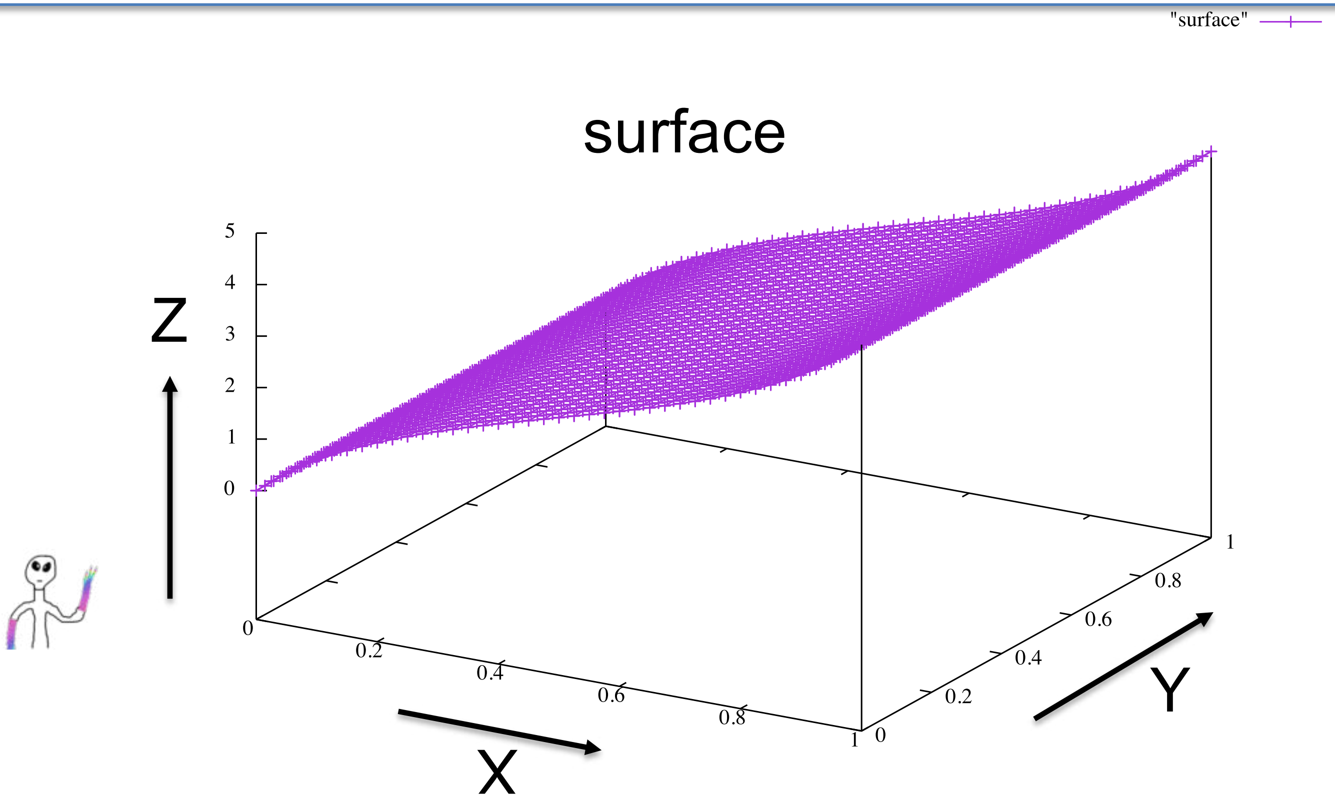
```
configuration                      // A mamdani controller
{
  and := min;                      // min, product
  or := max;                       // max, sum
  defuzz := center;                // center, max_average
  implication := mamdani;          // larsen, mamdani
  also := max;                     // max, sum
}
linguistic
{
  // Input sets: Linguistic values for X and Y (Y not used)
  Small := down_ramp(0.0, 1.0);
  Large := up_ramp(0.0, 1.0);

  // Output sets: Linguistic values Z
  SmallZ  := triangle(-1, 0,1);
  LargeZ  := triangle(4,5,6);
}
rules
{
  if X is Small then Z is SmallZ;
  if X is Large then Z is LargeZ;
}
```



Fuzzy logic inference: example of a (mamdani: min) controller

Terminal
student@student ~/Desktop/FuzzyLogic/FuzzyFonction $ ./function Controller2.fuz

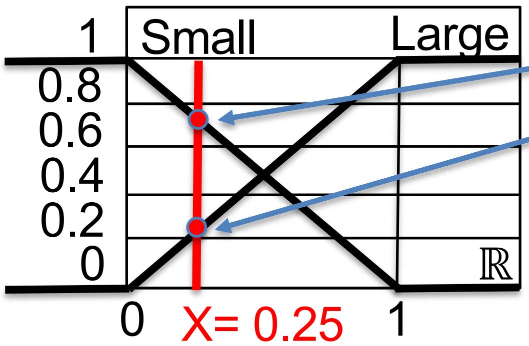


Fuzzy logic inference: very important remarks (1)

Implication: **larsen vs mamdani**

Example: if X is Small then Z is SmallZ (Rule 1)

if X is Large then Z is LargeZ (Rule 2)

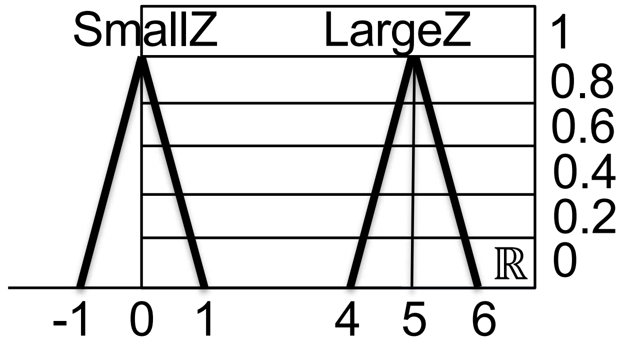


$\mu_{Small}(X) = 0.75$

$\mu_{Large}(X) = 0.25$

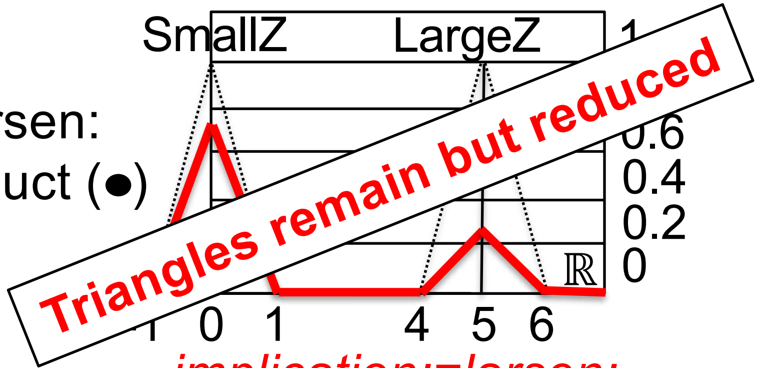
➔ Rule 1 true at 75%

Rule 2 true at 25%



larsen:

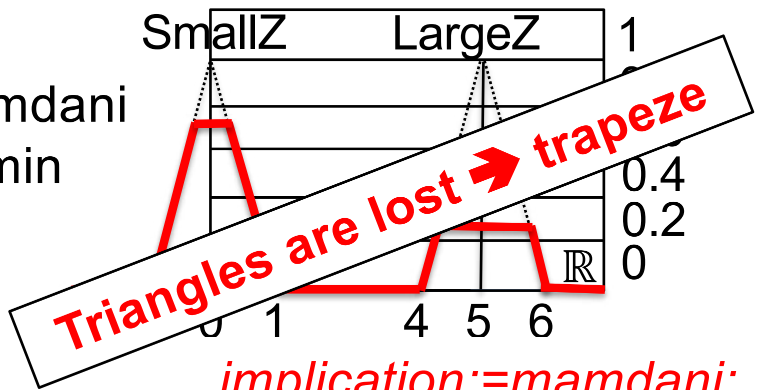
product (●)



implication:=larsen;

mamdani

min



implication:=mamdani;

Fuzzy logic inference: very important remarks (2)

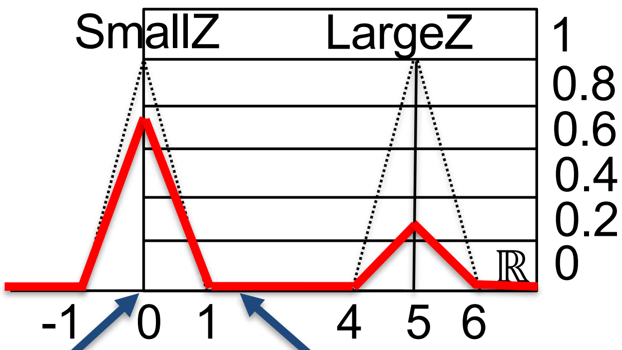
Defuzzyfication: **center vs max_average**

Example:

if X is Small then Z is SmallZ (Rule 1)

if X is Large then Z is LargeZ (Rule 2)

Larsen: Product (●)



defuzz:=max_average;
➔ only one maxima in 0
➔ 0

defuzz:=center;
➔ center of gravity
➔ 1.25

Fuzzy logic inference: very important remarks (3)

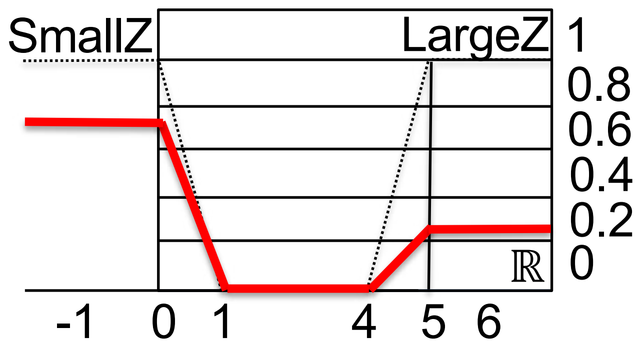
Defuzzyfication: **center & max_average**

Example: if X is Small then Z is SmallZ (Rule 1)
if X is Large then Z is LargeZ (Rule 2)

Larsen: Product (●)

For output fuzzy sets (here SmallZ and LargeZ):
Do not use ramps (open fuzzy sets) !

defuzz:=center;   ➔ For an open surface, impossible to compute the center of gravity
defuzz:=max_average;   ➔ For an open surface, impossible to compute the average of maxima



Fuzzy logic inference: very important remarks (4.1)

Merging rules results: **max vs sum**

Example: if X is Small then Z is SmallZ (Rule 1)

if X is Large then Z is LargeZ (Rule 2)

Larsen: Product (●)

- In a fuzzy controller, all the rules are evaluated (no priority)
- For a specific output variable (here Z), their results are merged
- Usually the results are merge with a t-conorm (max or sum)
- t-conorms properties: associativity + commutativity

➔ **rules evaluation order not important !**

Fuzzy logic inference: very important remarks (4.2)

Merging rules results: **max vs sum**

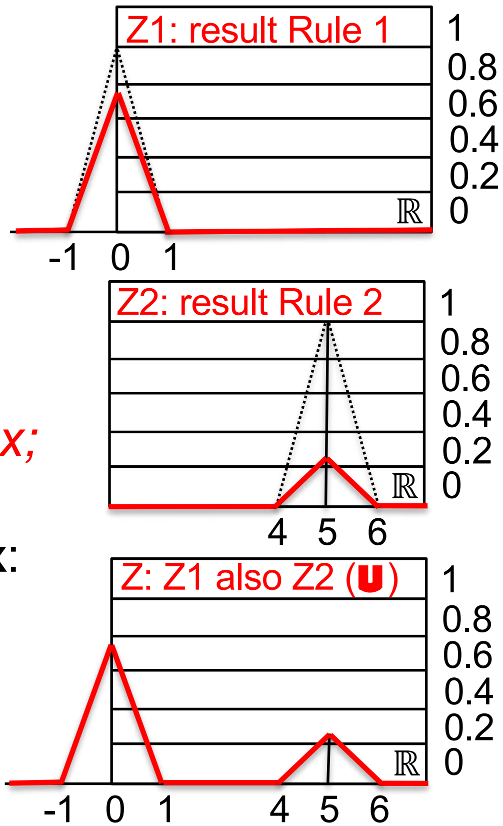
Example:

if X is Small then Z is SmallZ (Rule 1)
if X is Large then Z is LargeZ (Rule 2)

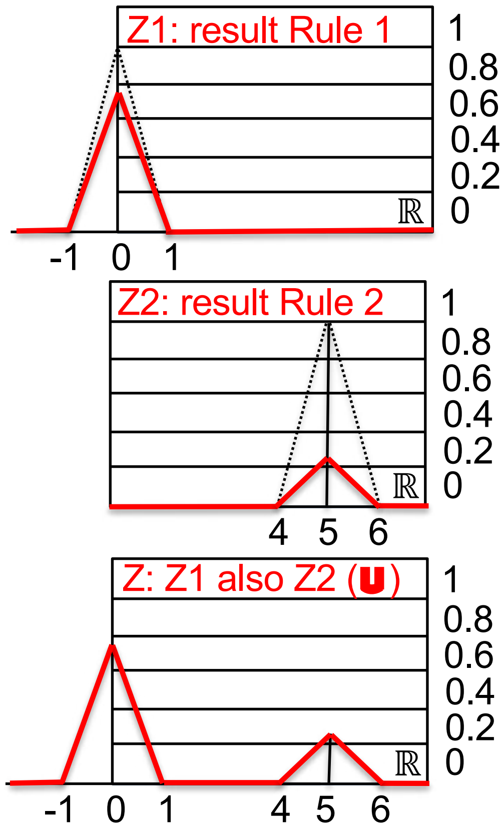
Larsen: Product (●)

also:=max;

with max: max(a,b)



Here, same results but..



also:=sum;

with sum: $a \oplus b = \min(1, a+b)$

Fuzzy logic inference: very important remarks (4.3)

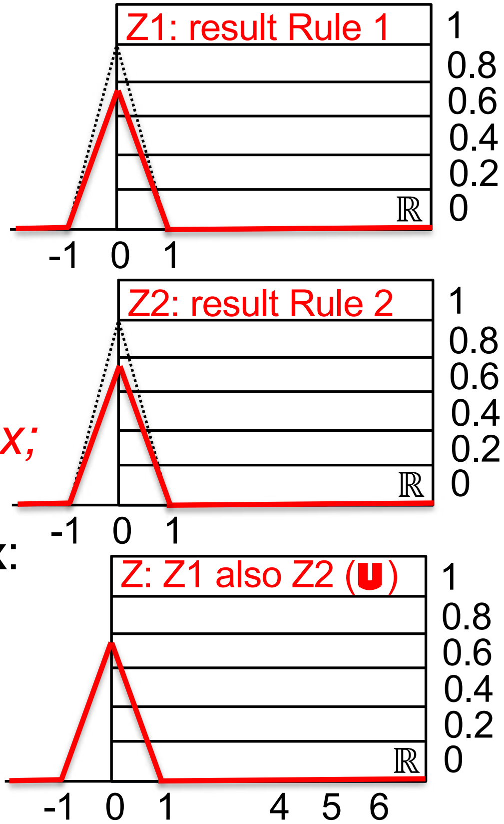
Merging rules results:

max vs sum

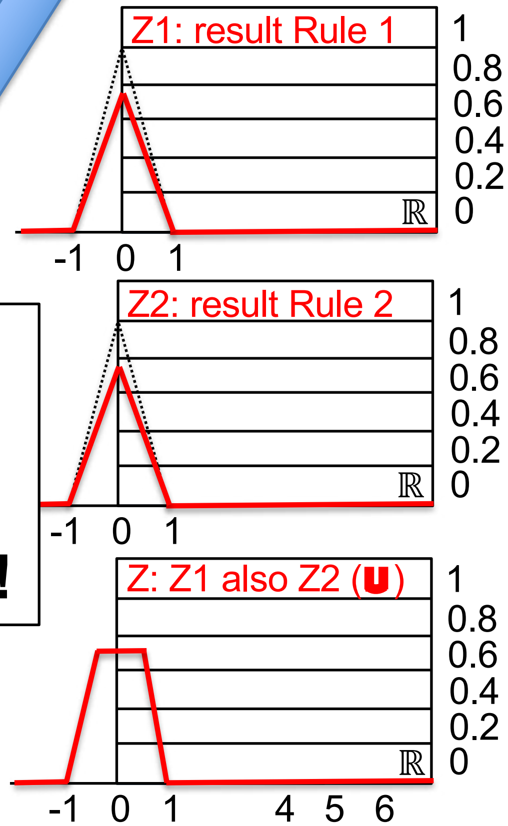
With sum:
If 2 rules give the same result
➔ amplified controller response

also:=max;

with max:
max(a,b)



In some cases, could be different !



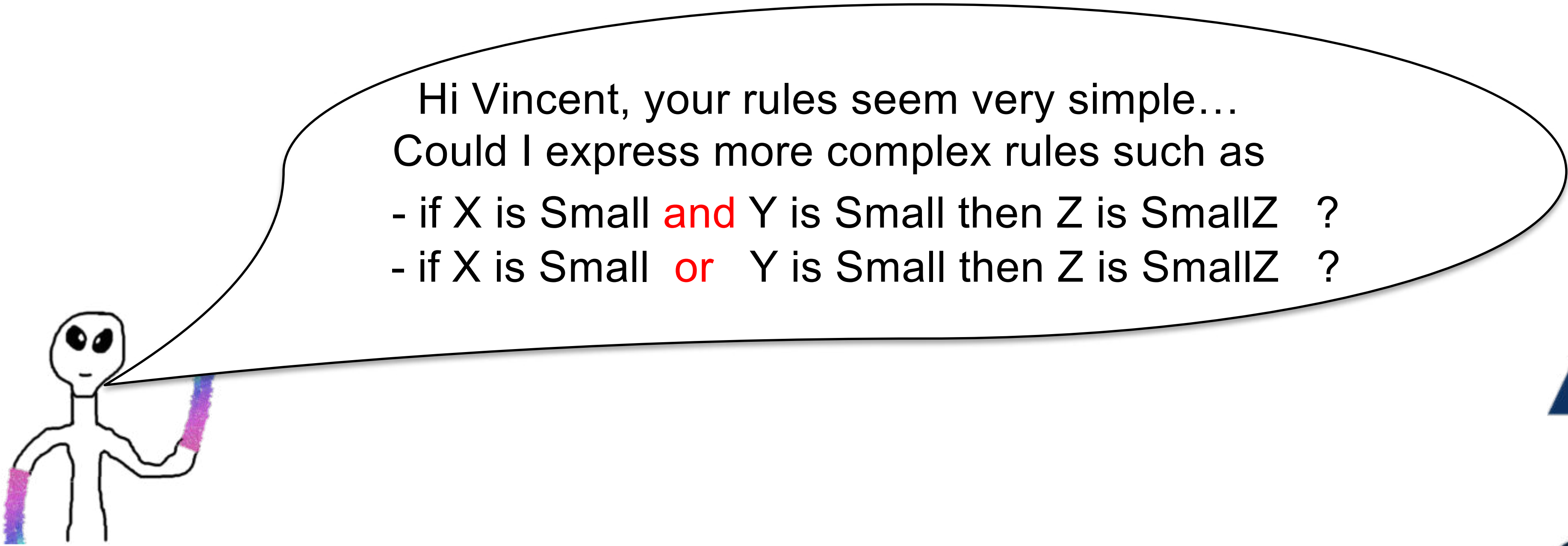
also:=sum;

with sum:
$a \oplus b = \min(1, a+b)$

Fuzzy logic inference: very important remarks (5.1)

Till now: if X is Small then Z is SmallZ (Rule 1)

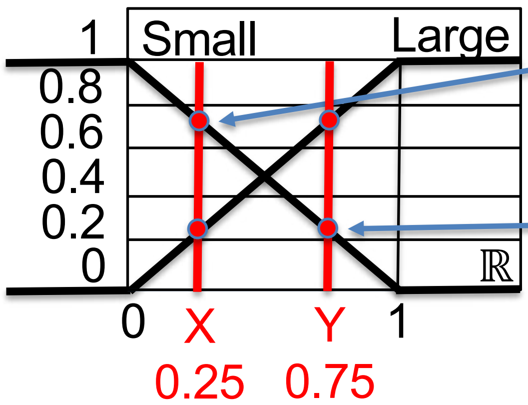
if X is Large then Z is LargeZ (Rule 2)



Fuzzy logic inference: very important remarks (5.2)

How to evaluate : if X is Small and Y is Small then Z is SmallZ ?

Let us consider the case with X=0.25 and Y=0.75

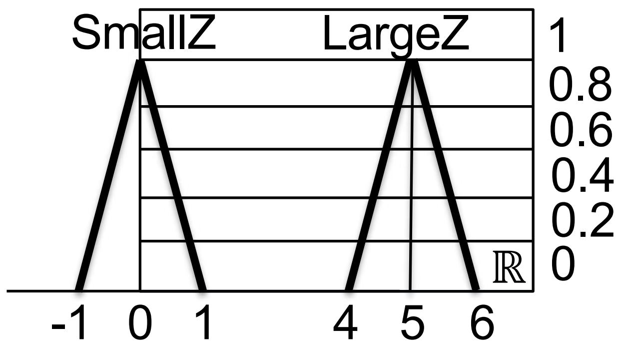


$\mu_{Small}(X) = 0.75$

$\mu_{Large}(X) = 0.25$

$\mu_{Small}(Y) = 0.25$

$\mu_{Large}(Y) = 0.75$



➔ Rule « if X is Small and Y is Small then Z is SmallZ » true at ?? %

Page 16, we saw that and is an intersection expressed by a t-norm $*$

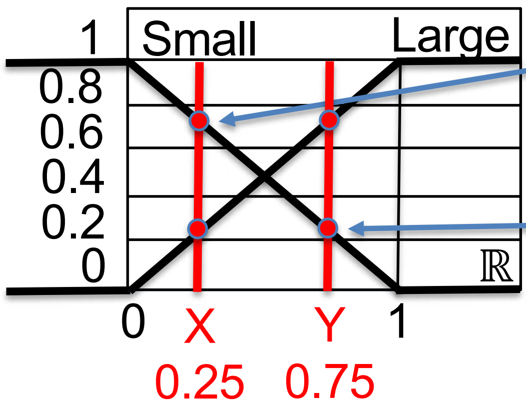
If we consider the t-norm $* \equiv \min$

➔ Rule « if X is Small and Y is Small then Z is SmallZ » true at $\min(\mu_{Small}(X), \mu_{Small}(Y))$ = min(0.75,0.25) = 0.25 ➔ 25 %

Fuzzy logic inference: very important remarks (5.3)

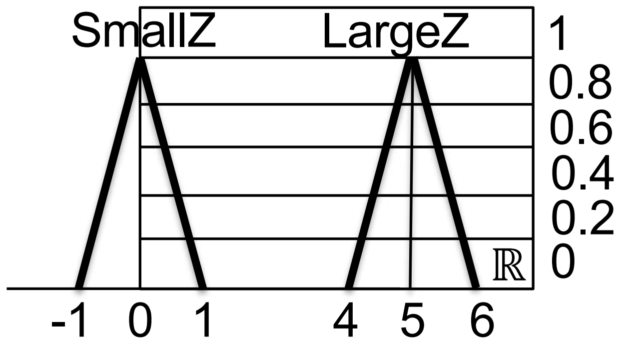
How to evaluate : if X is Small or Y is Small then Z is SmallZ ?

Let us consider the case with X=0.25 and Y=0.75



$\mu_{Small}(X) = 0.75$
$\mu_{Large}(X) = 0.25$

$\mu_{Small}(Y) = 0.25$
$\mu_{Large}(Y) = 0.75$



➔ Rule « if X is Small or Y is Small then Z is SmallZ » true at ?? %

Page 16, we saw that or is an union expressed by a t-conorm $\dot{+}$

If we consider the t-conorm $\dot{+} \equiv \max$

➔ Rule « if X is Small or Y is Small then Z is SmallZ » true at $\max(\mu_{Small}(X), \mu_{Small}(Y))$ = max(0.75,0.25) = 0.75 ➔ 75 %

Fuzzy logic inference: very important remarks (6)

Till now: if X is Small then Z is SmallZ (Rule 1)

if X is Large then Z is LargeZ (Rule 2)

Could we use *not*, *very*, *few* ?

Could we use rules such as:

- if X is *not* Small *and* Y is *very* Small then Z is SmallZ ?
- if X is *few* Small *or* Y is Small then Z is SmallZ ?

Yes !!!

➔ *not* (see page 18)

➔ *very* (see page 21)

➔ *few* (see page 22)

It is also possible **to combine** conclusions with *and*…

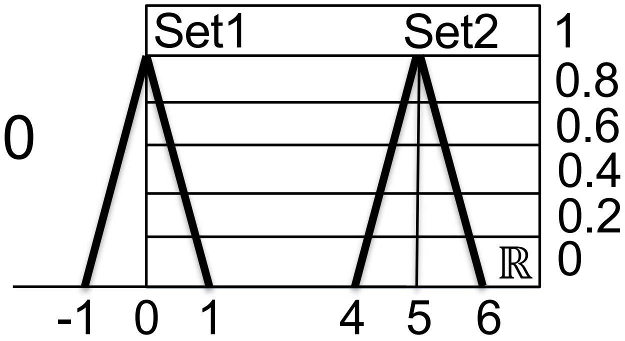
Fuzzy logic inference: very important remarks (7)

For a specific output variable V,
if no rule is active (i.e all the activation degrees are near 0)

➔ The final output set if « flat » near 0

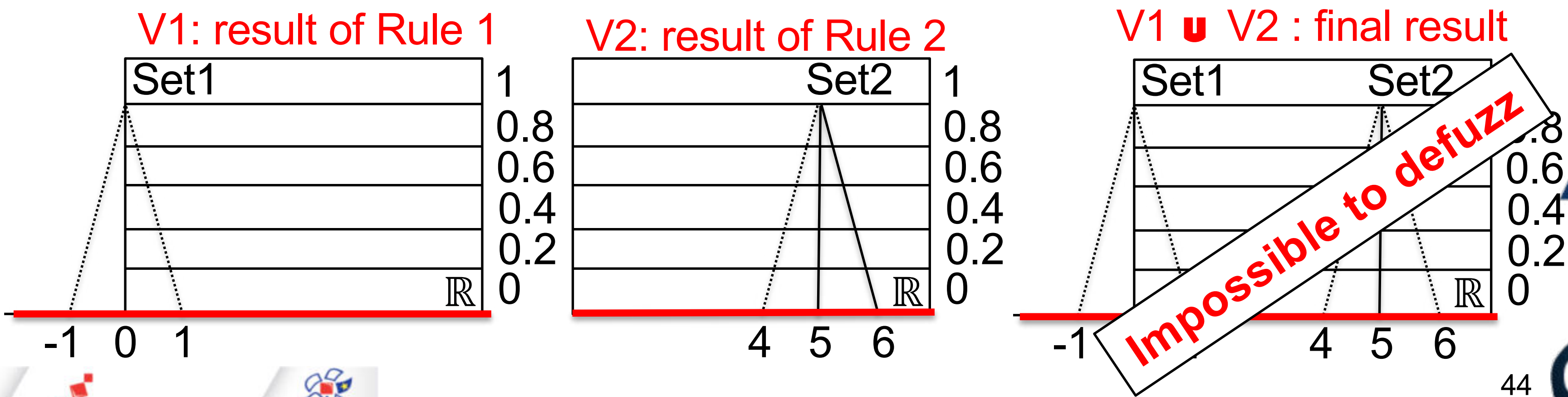
➔ Impossible to defuzz !

➔ By default, the controller will respond 0



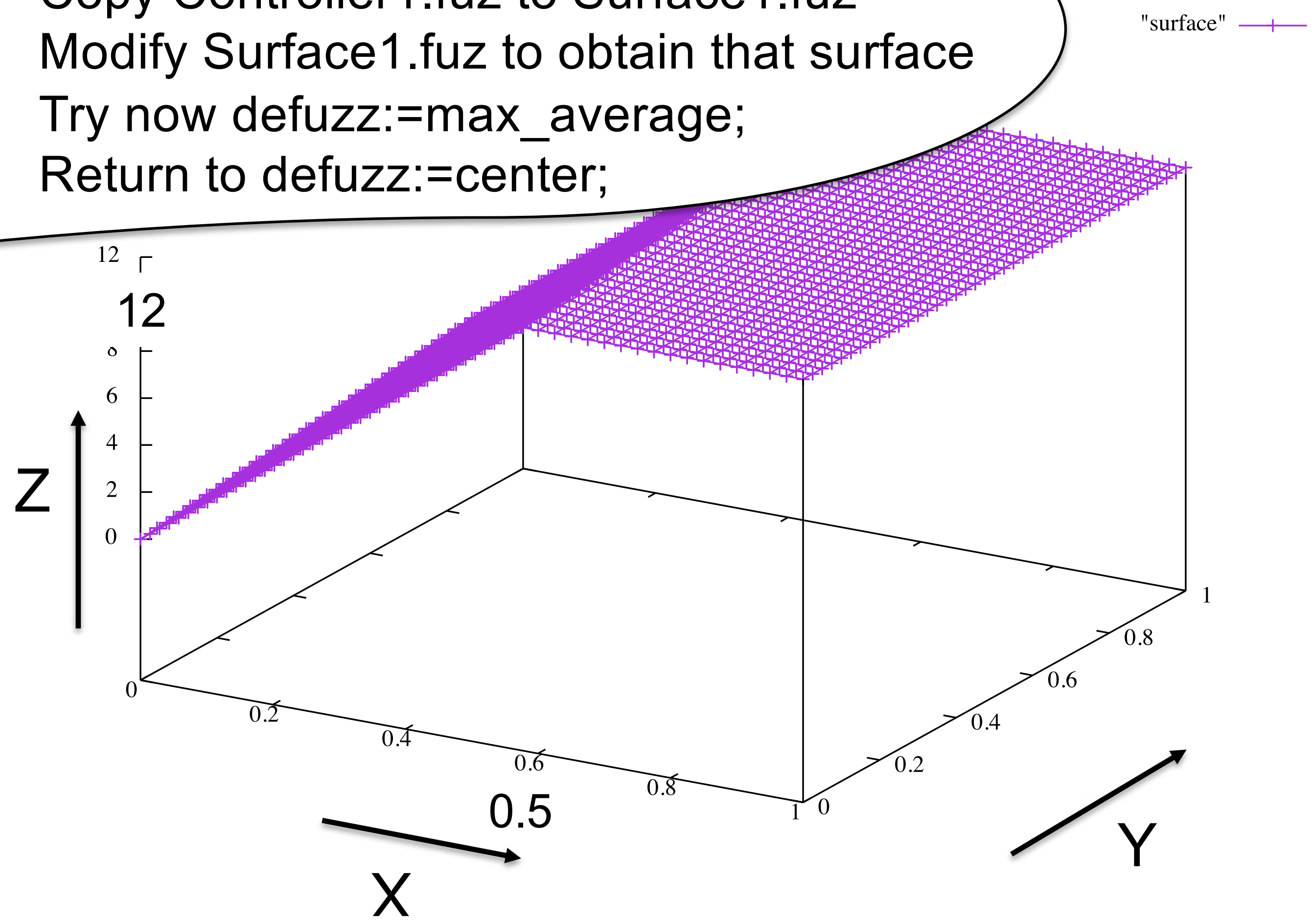
Rule 1: if cond_1 then V is Set1

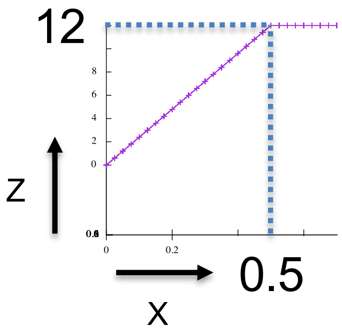
Rule 2: if cond_2 then V is Set2



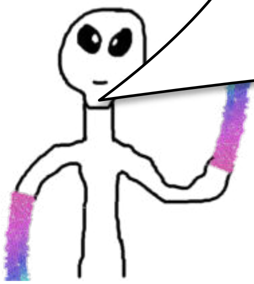
Exercices: Surface1.fuz

1. Copy Controller1.fuz to Surface1.fuz
2. Modify Surface1.fuz to obtain that surface
3. Try now defuzz:=max_average;
4. Return to defuzz:=center;



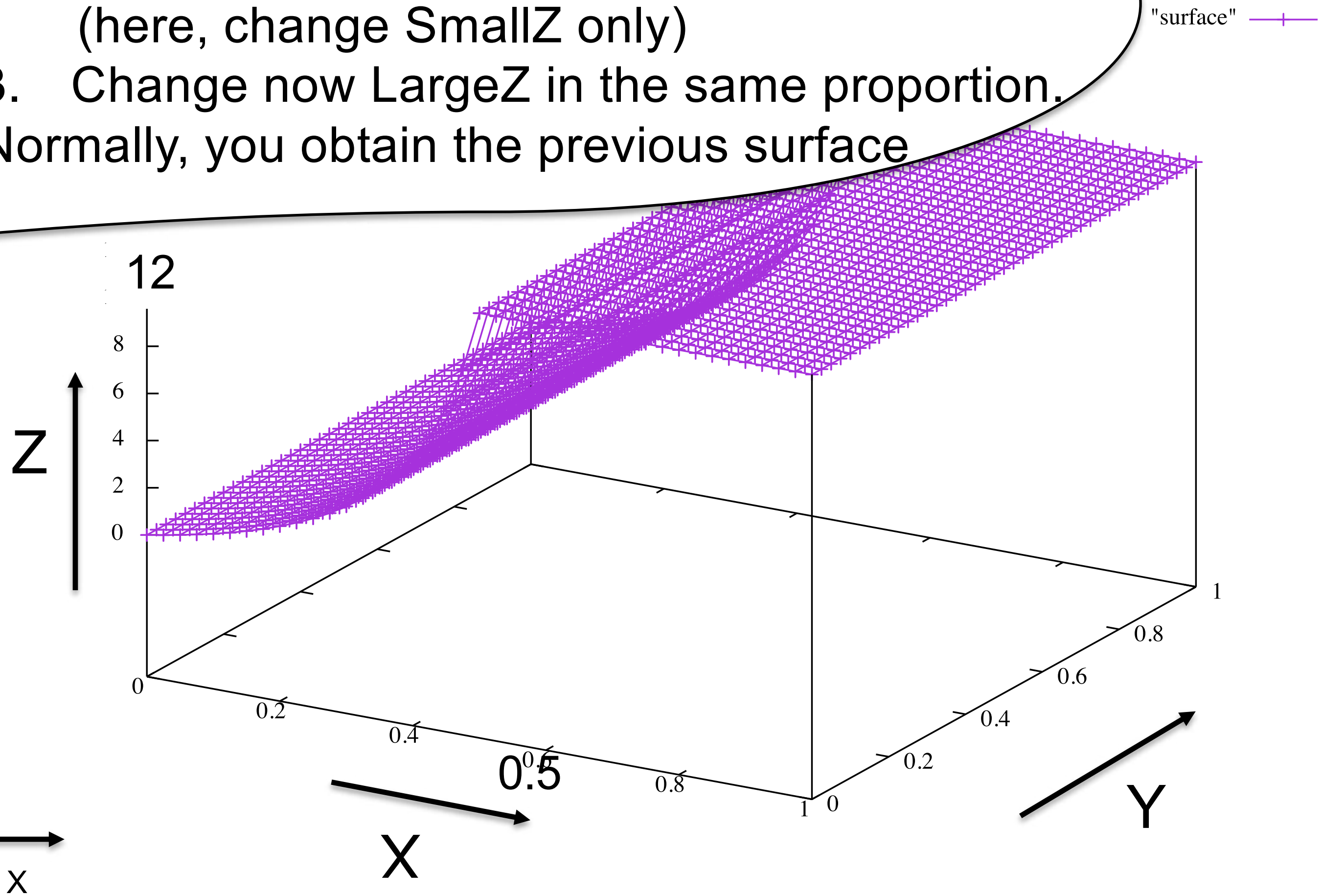


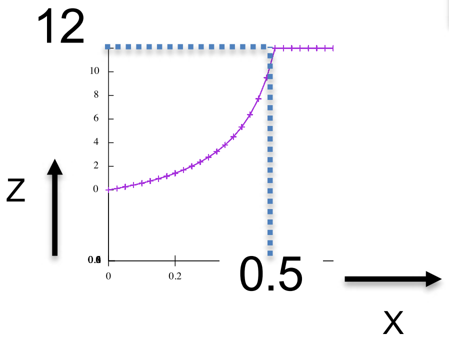
Exercices: Surface2.fuz



1. Copy Surface1.fuz to Surface2.fuz
2. Modify Surface2.fuz to obtain that surface
   (here, change SmallZ only)
3. Change now LargeZ in the same proportion.

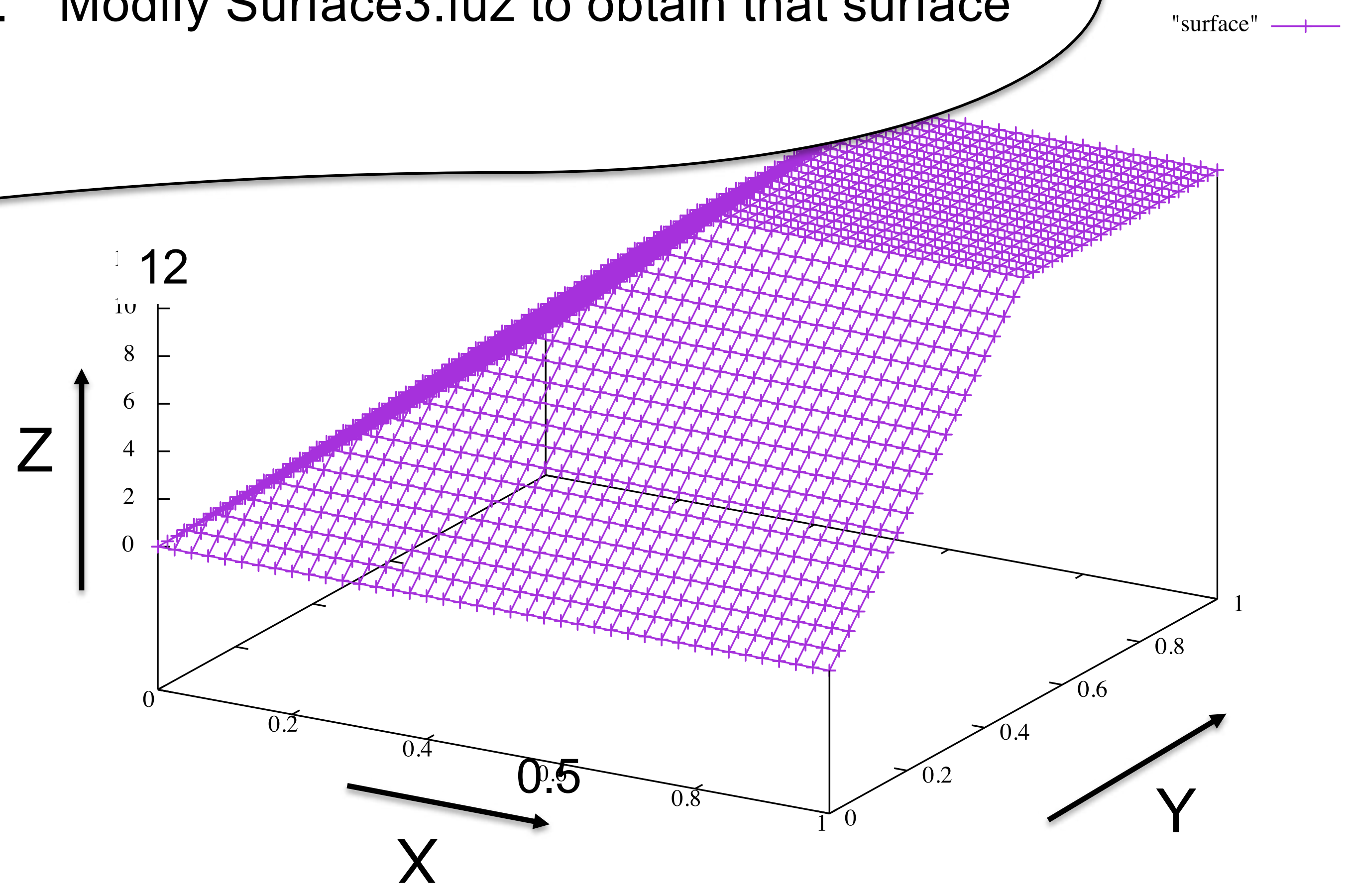
Normally, you obtain the previous surface



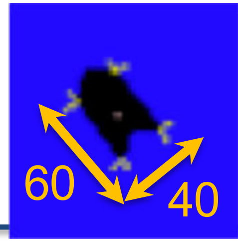


Exercices: Surface3.fuz

1. Copy Surface1.fuz to Surface3.fuz
2. Modify Surface3.fuz to obtain that surface



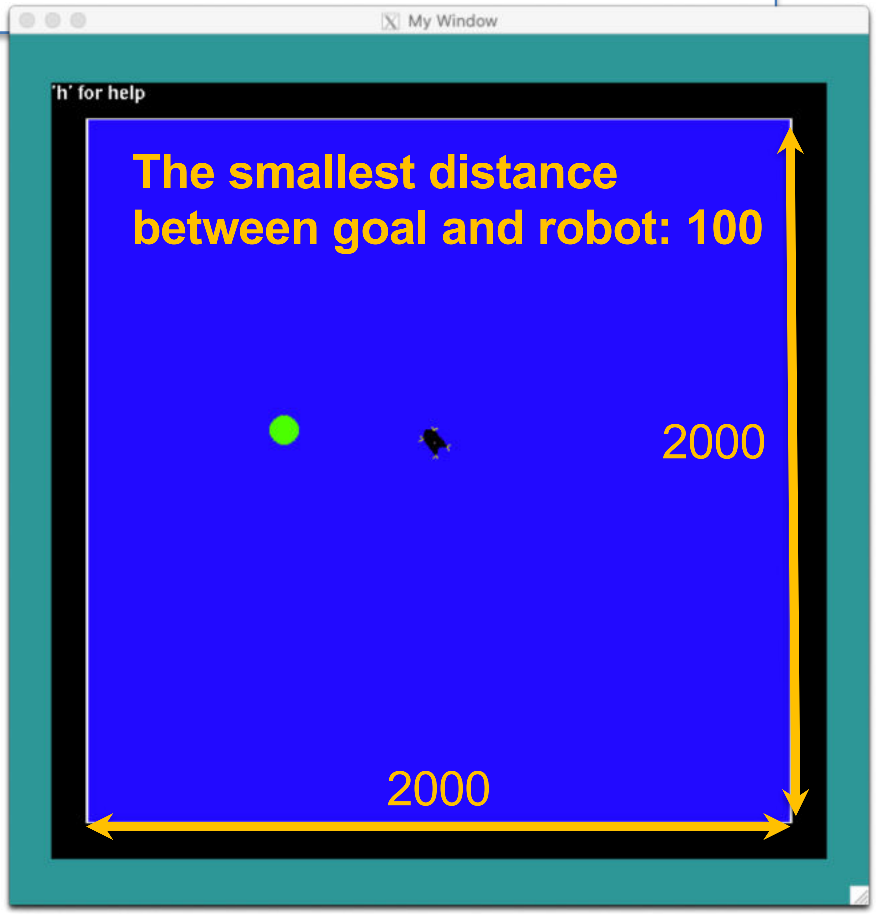
Exercices: a fuzzy Robot



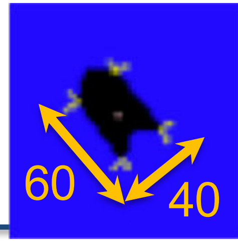
Terminal
student@student ~/Desktop/FuzzyLogic/FuzzyRobot $ ./robot map0.des direction.fuz

Help when mouse pointer inside this green/black/blue window :

's' : start robot
'g' : new goal
'v' : view detection
'i' : informations (distances,...)
'c' : fuzzy controller informations
'a' : autoscale
' ' : pause
'h' : help
'q' : quit



Exercices: a fuzzy Robot various informations



Informations about the robot:

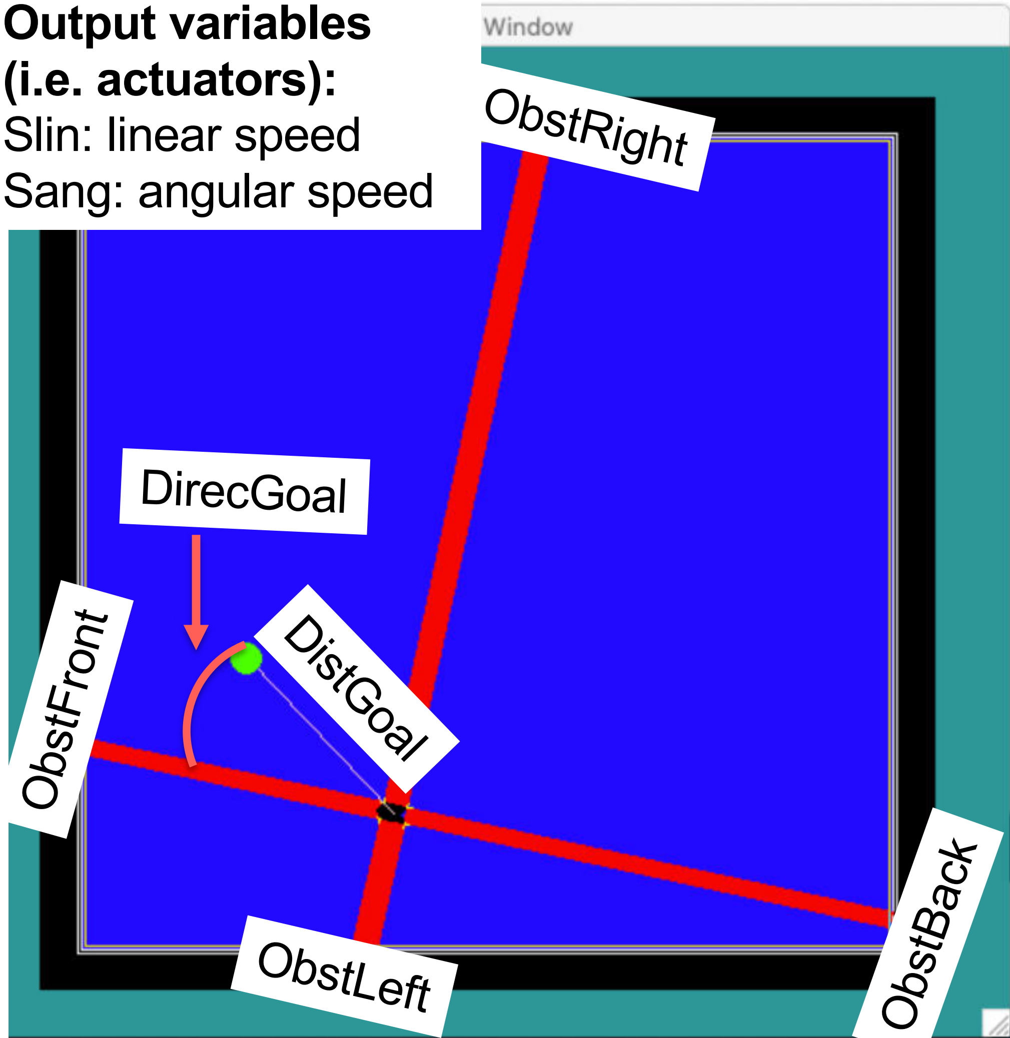
- The robot is oriented
- The maximum angular speed is $100^{O}.s^{-1}$
  Acceleration: $70^{O}.s^{-2}$
- The maximum linear speed is is 150 cm s^{-1}.
  Acceleration: $70cm.s^{-2}$
- The robot receives commands 10 times per second

Input variables (i.e. sensors):

DistGoal: distance to the goal
DirecGoal: orientation to the goal
ObstFront: distance to a front obstacle
ObstBack: distance to a back obstacle
ObstRight: distance to a right obstacle
ObstLeft: distance to a left obstacle
InSlin: current linear speed
InSang; current angular speed

Output variables (i.e. actuators):

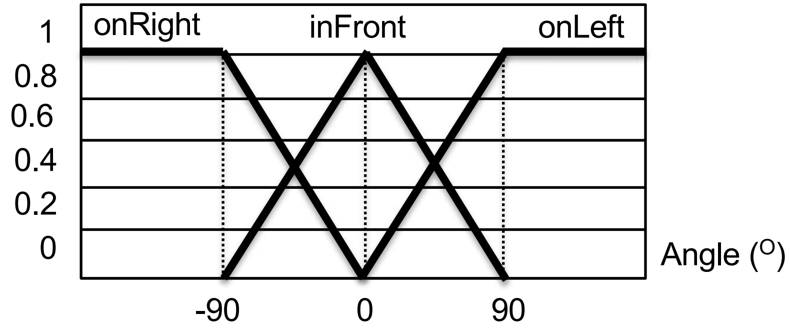
Slin: linear speed
Sang: angular speed



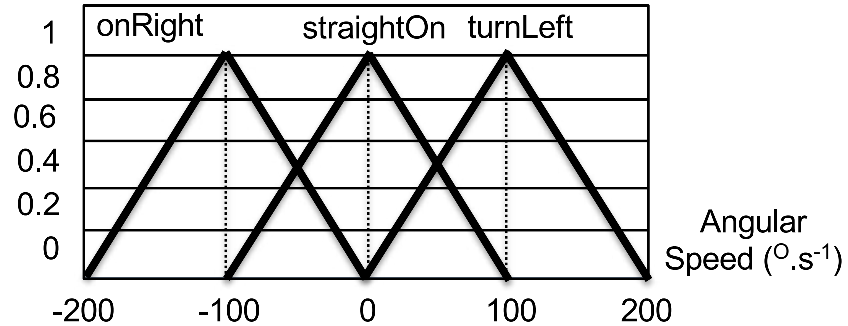
Exercices: a fuzzy Robot manage rotation: direction.fuz

```
configuration                       // A larsen controller
{
  and :=  min;                      // min, product
  or := max;                        // max, sum
  defuzz := center;                 // center, max_average
  implication := larsen;            // larsen, mamdani
  also := max;                      // max, sum
}
Linguistic
{
// Linguistic Values for DirecGoal (Input variable)
onRight := rampe_bas(-90.0,0);
inFront := triangle(-90.0,0.0,90.0);
onLeft := rampe_haut(0.0,90.0);
// Linguistic Values for Sang (Output variable)
turnRight := triangle(-200.0,-100.0,0.0);
straightOn := triangle(-100.0,0,100.0);
turnLeft := triangle(0.0,100.0,200.0);
}
Rules
{
// Orientation to the goal only
if DirecGoal is inFront then Sang is straightOn;
if DirecGoal is onLeft  then Sang is turnLeft;
if DirecGoal is onRight then Sang is turnRight;
}
```

For DirecGoal an **Input variable (sensor)**



For Sang an **Output variable (actuator)**



The robot must now go to the goal !!
➔ Manage Linear speed Slin