Fuzzy Logic

- Introduction
- Fuzzy Inference Systems
- Examples

Fuzzy Logic

- Introduction
  - What is Fuzzy Logic?
  - Applications of Fuzzy Logic
  - Classical Control System vs. Fuzzy Control
- Developing a Fuzzy Control System
- Examples
- Theory of Fuzzy Sets
- Fuzzy Inference Systems

menu

Topics

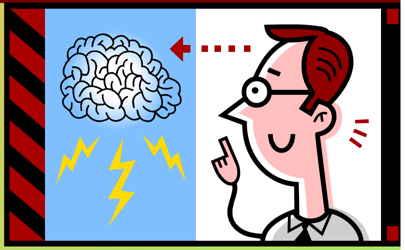
Basics
Control Systems
Computations
Inverted Pendulum
Mamdani
Sugeno
Fuzzy Sets
Defuzzification
Mem. Fcns

- Introduction
- Basic Algorithm
- Control Systems
- Sample Computations
- Inverted Pendulum
- Fuzzy Inference Systems
  - Mamdani Type
  - Sugeno Type
- Fuzzy Sets & Operators
- Defuzzification
- Membership Functions

menu

Fuzzy Logic

What is Fuzzy Logic?



A computational paradigm that is based on how humans think

Fuzzy Logic looks at the world in **imprecise terms**, in much the same way that our brain takes in information (e.g. temperature is hot, speed is slow), then responds with **precise actions**.

The human brain can reason with uncertainties, vagueness, and judgments. Computers can only manipulate precise valuations. Fuzzy logic is an attempt to combine the two techniques.

“Fuzzy” – a misnomer, has resulted in the mistaken suspicion that FL is somehow less exacting than traditional logic

Fuzzy Logic

What is Fuzzy Logic?

FL is in fact, **a precise problem-solving methodology**.

It is able to simultaneously handle numerical data and linguistic knowledge.

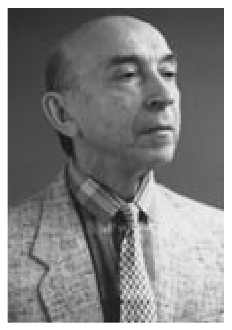
A technique that facilitates the control of a complicated system without knowledge of its mathematical description.

Fuzzy logic differs from classical logic in that statements are no longer black or white, true or false, on or off.
In traditional logic an object takes on a value of either zero or one.
In fuzzy logic, a statement can assume any real value between 0 and 1, representing the degree to which an element belongs to a given set.

menu

Fuzzy Logic

History of Fuzzy Logic



Professor Lotfi A. Zadeh

http://www.cs.berkeley.edu/~zadeh/

In **1965**, **Lotfi A. Zadeh** of the University of California at Berkeley published "Fuzzy Sets," which laid out the mathematics of fuzzy set theory and, by extension, fuzzy logic. Zadeh had observed that conventional computer logic couldn't manipulate data that represented subjective or vague ideas, so he created fuzzy logic to allow computers to determine the distinctions among data with shades of gray, similar to the process of human reasoning.

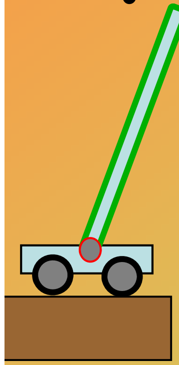
Source: August 30, 2004
(Computerworld)

http://www.computerworld.com/news/2004/story/0,11280,95282,00.html

Pioneering works

20 years later after its conception

- Interest in fuzzy systems was sparked by **Seiji Yasunobu** and **Soji Miyamoto** of **Hitachi**, who in **1985** provided simulations that demonstrated the superiority of fuzzy control systems for the **Sendai railway**. Their ideas were adopted, and fuzzy systems were used to control accelerating and braking when the line opened in **1987**.



- Also in **1987**, during an international meeting of fuzzy researchers in Tokyo, **Takeshi Yamakawa** demonstrated the use of fuzzy control, through a set of simple dedicated fuzzy logic chips, in an "**inverted pendulum**" experiment. This is a classic control problem, in which a vehicle tries to keep a pole mounted on its top by a hinge upright by moving back and forth.

- Observers were impressed with this demonstration, as well as later experiments by **Yamakawa** in which he mounted a wine glass containing water or even a live mouse to the top of the pendulum. The system maintained stability in both cases. Yamakawa eventually went on to organize his own fuzzy-systems research lab to help exploit his patents in the field.

http://en.wikipedia.org/wiki/Fuzzy_control_system

menu

Meeting Lotfi in Germany

My Fuzzy Logic-based Researches

- Robot Navigation
  - Real-time path-planning (Hybrid Fuzzy A*)
- Machine Vision
  - Real-time colour-object recognition
  - Colour correction
  - Fuzzy Colour Contrast Fusion
  - Fuzzy-Genetic Colour Contrast Fusion

9th Fuzzy Days (2006), Dortmund, Germany

Meeting Prof. Yamakawa in Japan

ICONIP 2007, Kitakyushu, Japan

menu

Fuzzy Logic

Introduction of FL in the Engineering world (1990’s),

Fuzzy Logic is one of the most talked-about technologies to hit the embedded control field in recent years. It has already transformed many product markets in Japan and Korea, and has begun to attract a widespread following In the United States. Industry watchers predict that fuzzy technology is on its way to becoming a multibillion-dollar business.

Fuzzy Logic enables low cost microcontrollers to perform functions traditionally performed by more powerful expensive machines enabling lower cost products to execute advanced features.

Intel Corporation's Embedded Microcomputer Division Fuzzy Logic Operation

MCS® 96/296 Microcontrollers

Designed to Meet Your Needs

Motorola 68HC12 MCU

http://www.intel.com/design/mcs96/designex/2351.htm

menu

Sample Applications

In the city of Sendai in Japan, a 16-station subway system is controlled by a fuzzy computer (Seiji Yasunobu and Soji Miyamoto of Hitachi) – the ride is so smooth, riders do not need to hold straps

Nissan – fuzzy automatic transmission, fuzzy anti-skid braking system

CSK, Hitachi – Hand-writing Recognition

Sony - Hand-printed character recognition

Ricoh, Hitachi – Voice recognition

Tokyo’s stock market has had at least one **stock-trading portfolio based on Fuzzy Logic** that outperformed the Nikkei exchange average

Sample Applications

NASA has studied fuzzy control for **automated space docking**: simulations show that a fuzzy control system can greatly reduce fuel consumption

Canon developed an **auto-focusing camera** that uses a charge-coupled device (CCD) to measure the clarity of the image in six regions of its field of view and use the information provided to determine if the image is in focus. It also tracks the rate of change of lens movement during focusing, and controls its speed to prevent overshoot.

The camera's fuzzy control system uses **12** inputs: 6 to obtain the current clarity data provided by the CCD and 6 to measure the rate of change of lens movement. The output is the position of the lens. The **fuzzy control system** uses **13 rules** and requires **1.1 kilobytes** of memory.

http://en.wikipedia.org/wiki/Fuzzy_system

Sample Applications

For washing machines, Fuzzy Logic control is almost becoming a standard feature

fuzzy controllers to load-weight, fabric-mix, and dirt sensors and automatically set the wash cycle for the best use of power, water, and detergent.

GE WPRB9110WH Top Load Washer

Haier ESL-T21 Top Load Washer

LG WD14121 Front Load Washer

Miele WT945 Front Load All-in-One Washer / Dryer

AEG LL1610 Front Load Washer

Zanussi ZWF1430W Front Load Washer

Others: Samsung, Toshiba, National, Matsushita, etc.

Control Systems

- Conventional Control vs. Fuzzy Control

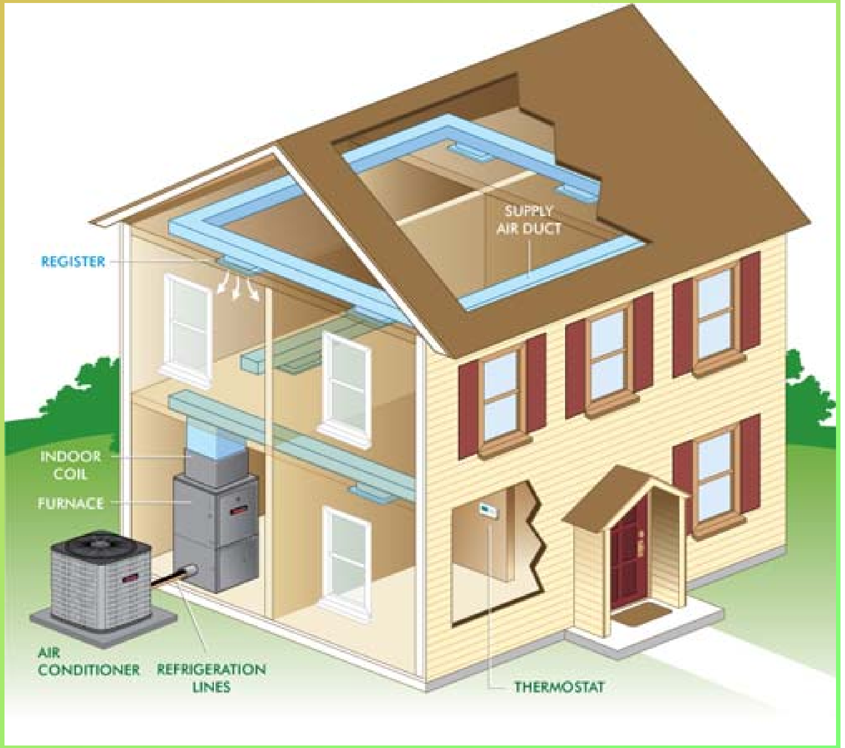
Control Systems in General

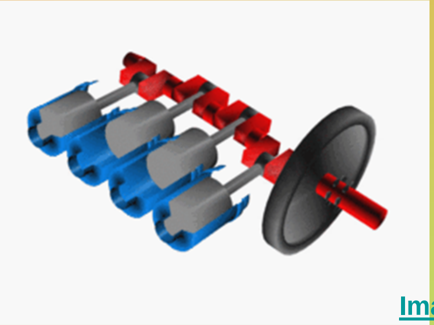
Objective

The aim of any control system is **to produce a set of desired outputs for a given set of inputs**.

Samples

A household thermostat takes a temperature input and sends a control signal to a furnace.





Crankshaft (red), pistons (gray) in their cylinders (blue), and flywheel (black)

Image: http://en.wikipedia.org/wiki/Crankshaft

A car engine controller responds to variables such as engine position, manifold pressure and cylinder temperature to regulate fuel flow and spark timing.

Conventional Control vs. Fuzzy

Look-up table

In the simplest case, a controller takes its cues from a look-up table, which tells what output to produce for every input or combination of inputs.

Sample

The table might tell the controller,
“**IF temperature is 85, THEN increase furnace fan speed to 300 RPM.**”

Drawbacks

The problem with the tabular approach is that the **table can get very long**, especially in situations where there are many inputs or outputs. And that, in turn, **may require more memory than the controller can handle**, or more than is cost-effective.

Tabular control mechanisms may also give a **bumpy, uneven response**, as the controller jumps from one table-based value to the next.

menu

Conventional Control vs. Fuzzy

Mathematical formula

The usual alternative to look-up tables is to have the controller execute a mathematical formula – **a set of control equations that express the output as a function of the input**.

Ideally, these equations represent an accurate model of the system behaviour.

For example:

$$\left[m \frac{\partial^2}{\partial t^2}(x + l \sin \theta) \right] l \cos \theta - \left[m \frac{\partial^2}{\partial t^2}(l \cos \theta) \right] l \sin \theta = mgl \sin \theta$$

Downside

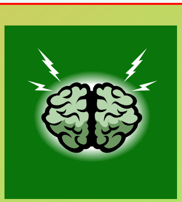
The formulas can be very complex, and working them out in real-time may be more than an affordable controller (or machine) can manage.

Conventional Control vs. Fuzzy

Downside of Mathematical modeling approach

It may be difficult or impossible to derive a workable mathematical model in the first place, making both tabular and formula-based methods impractical.

Though an automotive engineer might understand the general relationship between say, ignition timing, air flow, fuel mix and engine RPM, the exact math that underlies those interactions may be completely obscure.



Why use Fuzzy Logic?

FL overcomes the disadvantages of both table-based and formula-based control.

Fuzzy has **no unwieldy memory requirements** of look-up tables,
and **no heavy number-crunching demands** of formula-based solutions.

Conventional Control vs. Fuzzy

Why use Fuzzy Logic?

FL can make development and implementation much simpler.

It needs no intricate mathematical models, only a practical understanding of the overall system behaviour.

FL mechanisms can result to **higher accuracy** and **smoother control** as well.

Fuzzy Logic Explained

Fuzzy Set Theory

FL differs from orthodox logic in that it is multivalued.

Fuzzy deals with degrees of truth and degrees of membership.

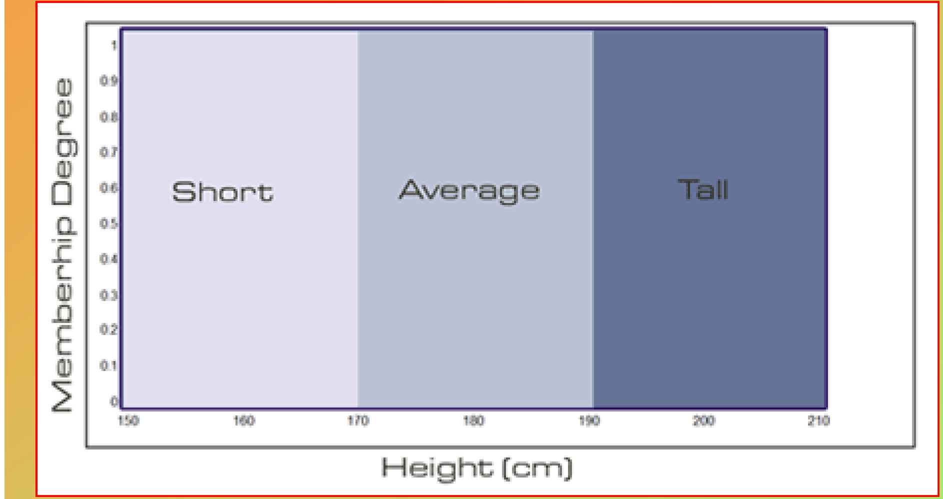
Fuzzy Logic Explained

Fuzzy Set Theory

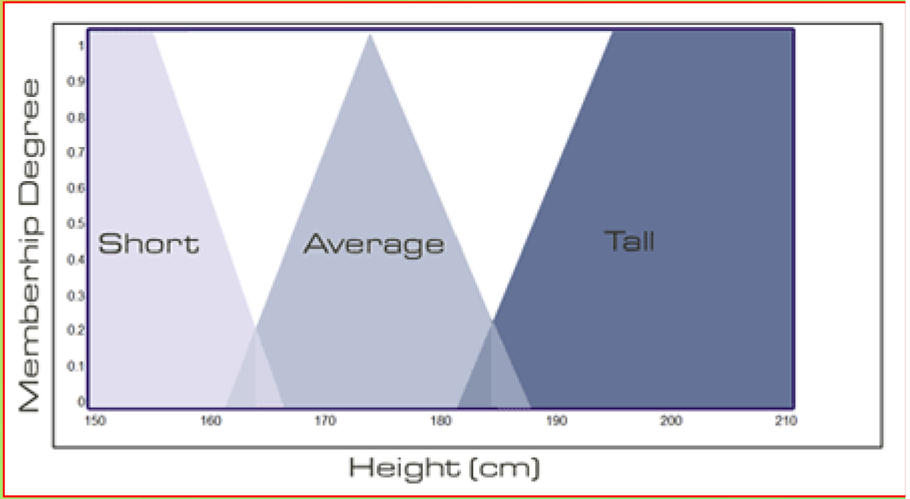
Is a man whose height is 5’ 11-1/2” average or tall?

A fuzzy system might say that he is partly medium and partly tall.

Boolean representation



Fuzzy representation



http://blog.peltarion.com/2006/10/25/fuzzy-math-part-1-the-theory/

In fuzzy terms, the height of the man would be classified within a range of [0, 1] as average to a degree of 0.6, and tall to a degree of 0.4.

Fuzzy Logic Explained

Fuzzy Set Theory

Is a man whose height is 5’ 11-1/2” medium or tall?

A fuzzy system might say that he is partly medium and partly tall.

In other words, FL recognizes not only clear-cut, black-and-white alternatives, but also the infinite gradations in between.

Fuzzy reasoning eliminates the vagueness by assigning specific numbers to those gradations. These numeric values are then used to derive exact solutions to problems.

In fuzzy terms, the height of the man would be classified within a range of [0, 1] as medium to a degree of 0.6, and tall to a degree of 0.4.

Excerpts from History

Fuzzy Set Theory

"So far as the laws of mathematics refer to reality, they are not certain. And so far as they are certain, they do not refer to reality."

Albert Einstein
Theoretical Physicist and Nobel laureate
"Geometrie und Erfahrung," Lecture to Prussian Academy, 1921

Most things in nature cannot be characterised with simple or convenient shapes or distributions.

Membership functions characterize the fuzziness in a fuzzy set – whether the elements in the set are discrete or continuous – in a graphical form for eventual use in the mathematical formalisms of fuzzy set theory.

The statement above from Einstein, attests to the fact that few things in real life are certain or can be conveniently reduced to the axioms of mathematical theories and models.

Source: Fuzzy Logic with Engineering Applications by Tim Ross

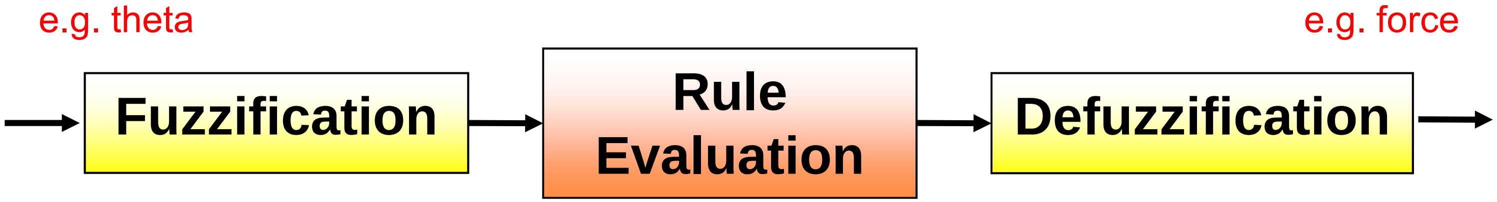
Fuzzy Inference Process

- What are the steps involved in creating a Fuzzy Control System?

menu

Fuzzy Inference Process

Fuzzy Inference Process



e.g. theta

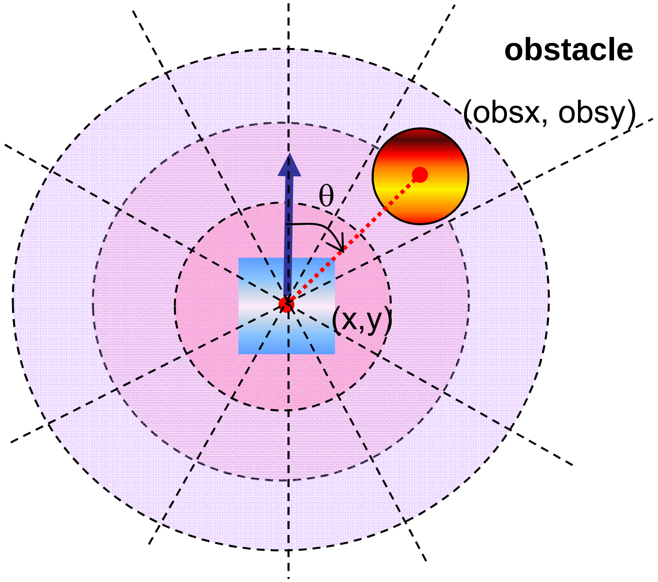
Fuzzification → **Rule Evaluation** → **Defuzzification** →

e.g. force

Fuzzification: Translate input into truth values
Rule Evaluation: Compute output truth values
Defuzzification: Transfer truth values into output

Obstacle Avoidance Problem

Robot Navigation



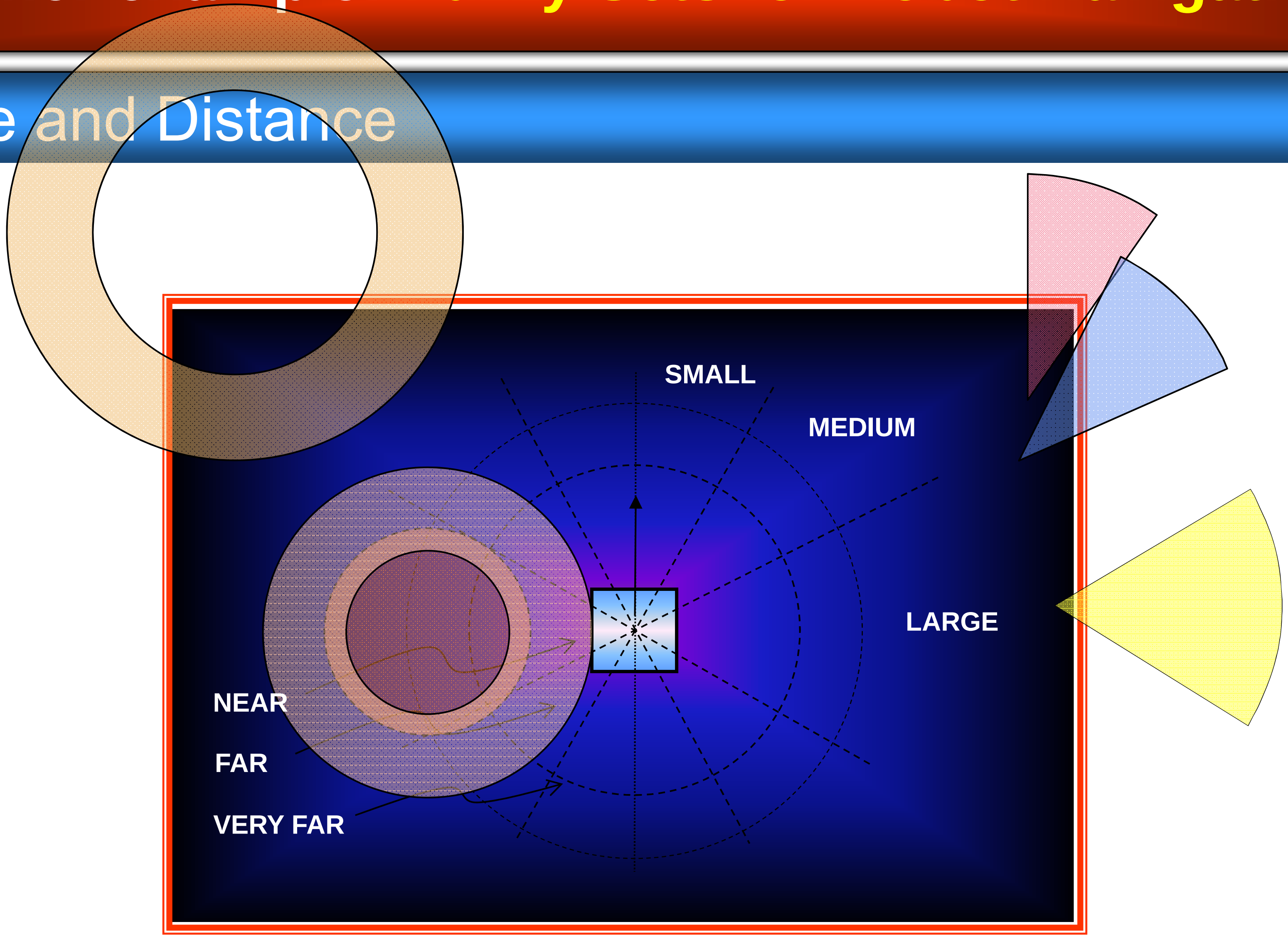
Can you describe how the robot should turn based on the position and angle of the obstacle?

Obstacle Avoidance & Target Pursuit

ⓧ Demonstration

Another example: Fuzzy Sets for Robot Navigation

Angle and Distance



- Sub ranges for angles & distances overlap

Fuzzy Systems *for* Obstacle Avoidance

Vision System

Nearest Obstacle (Distance and Angle)

Fuzzy System 3 (Steering)

	NEAR	*FAR*	*VERY FAR*
SMALL	Very Sharp	Sharp Turn	Med Turn
MEDIUM	Sharp Turn	Med Turn	Mild Turn
LARGE	Med Turn	Mild Turn	Zero Turn

e.g. If the *Distance* from the Obstacle is *NEAR* and the *Angle* from the Obstacle is *SMALL*
Then turn *Very Sharply*.

Angle

Fuzzy System 4 (Speed Adjustment)

	NEAR	*FAR*	*VERY FAR*
SMALL	Very Slow	Slow Speed	Fast Fast
MEDIUM	Slow Speed	Fast Speed	Very Fast
LARGE	Fast Speed	Very Fast	Top Speed

e.g. **If** the *Distance* from the Obstacle is *NEAR* and the *Angle* from the Obstacle is *SMALL*
Then move *Very Slowly.*

Speed

Hand-Simulation

Fuzzy Control

Different stages of Fuzzy control

1. Fuzzification

Input variables are assigned degrees of membership in various classes

e.g. A temperature input might be graded according to its degree of coldness, coolness, warmth or heat.

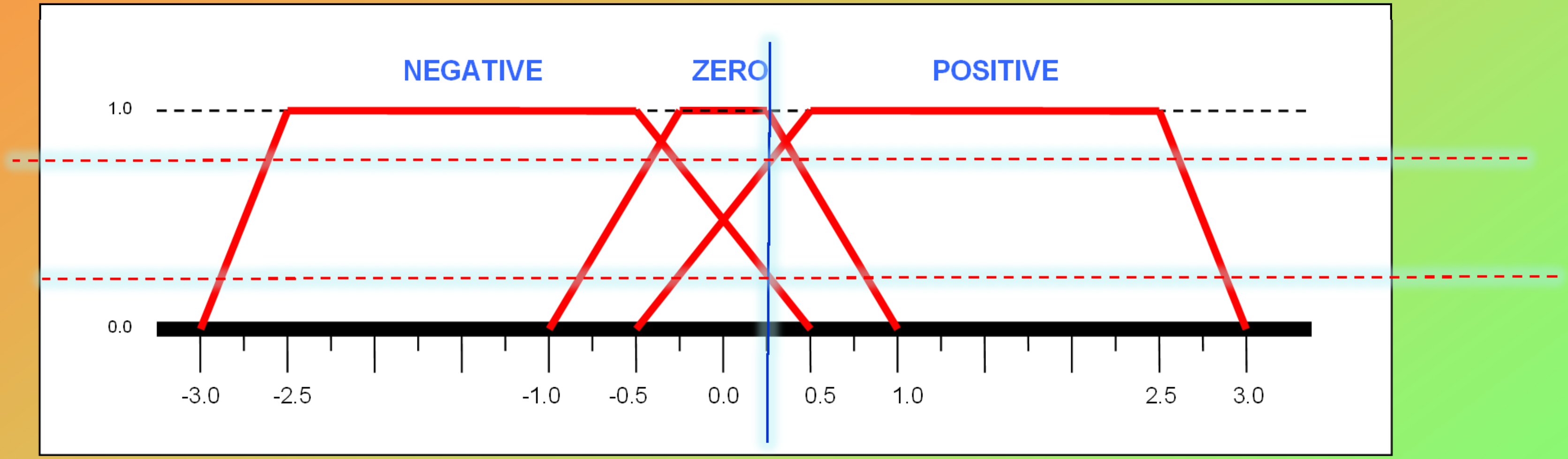
The purpose of **fuzzification** is to **map the inputs** from a set of sensors (or features of those sensors) **to values from 0 to 1** using a set of input membership functions.

We will see a complete example of the steps involved later.

Fuzzification

Fuzzification Example

Fuzzy Sets = { Negative, Zero, Positive }



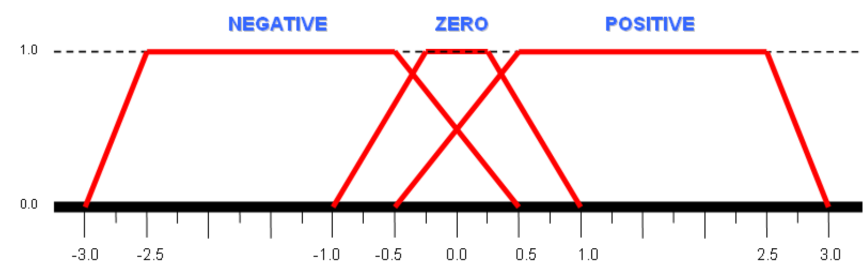
Assuming that we are using trapezoidal membership functions.

Crisp Input: **x = 0.25**

What is the degree of membership of x in each of the Fuzzy Sets?

Sample Calculations

Crisp Input: $x = 0.25$



$F_{zero}(0.25)$

$$F_{ZE}(0.25) = \max\left(\min\left(\frac{x-a}{b-a},1,\frac{d-x}{d-c}\right),0\right)$$

$$= \max\left(\min\left(\frac{0.25-(-1)}{-0.25-(-1)},1,\frac{1-0.25}{1-0.25}\right),0\right)$$

$$= \max\left(\min\left(1.67,1,1\right),0\right)$$

$$= 1$$

$F_{positive}(0.25)$

$$F_P(0.25) = \max\left(\min\left(\frac{0.25-(-0.5)}{0.5-(-0.5)},1,\frac{3-0.25}{3-0.25}\right),0\right)$$

$$= \max\left(\min\left(0.75,1,5.5\right),0\right)$$

$$= 0.75$$

$F_{negative}(0.25)$

$$F_N(0.25) = \max\left(\min\left(\frac{0.25-(-3)}{-2.5-(-3)},1,\frac{0.5-0.25}{0.5-(-0.5)}\right),0\right)$$

$$= \max\left(\min\left(6.5,1,0.25\right),0\right)$$

$$= 0.25$$

menu

Sample Calculations

Crisp Input: $y = -0.25$

$F_{zero}(-0.25)$

$$F_{ZE}(-0.25) = \max\left(\min\left(\frac{-0.25-(-1)}{-0.25-(-1)},1,\frac{1-(-0.25)}{1-0.25}\right),0\right)$$

$$= \max\left(\min\left(1,1,1.67\right),0\right)$$

$$= 1$$

$F_{positive}(-0.25)$

$$F_{P}(-0.25) = \max\left(\min\left(\frac{-0.25-(-3)}{0.5-(-0.5)},1,\frac{3-(-0.25)}{3-2.5}\right),0\right)$$

$$= \max\left(\min\left(0.25,1,6.5\right),0\right)$$

$$= 0.25$$

$F_{negative}(-0.25)$

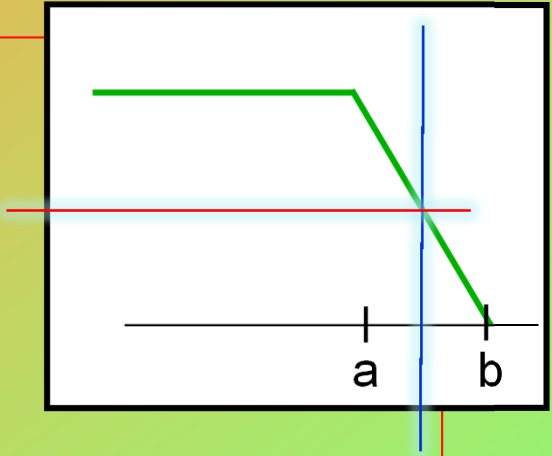
$$F_{N}(-0.25) = \max\left(\min\left(\frac{-0.25-(-3)}{-2.5-(-3)},1,\frac{0.5-(-0.25)}{0.5-(-0.5)}\right),0\right)$$

$$= \max\left(\min\left(5.5,1,0.75\right),0\right)$$

$$= 0.75$$

menu

Trapezoidal Membership Functions



LeftTrapezoid

Left_Slope = 0
Right_Slope = 1 / (A - B)

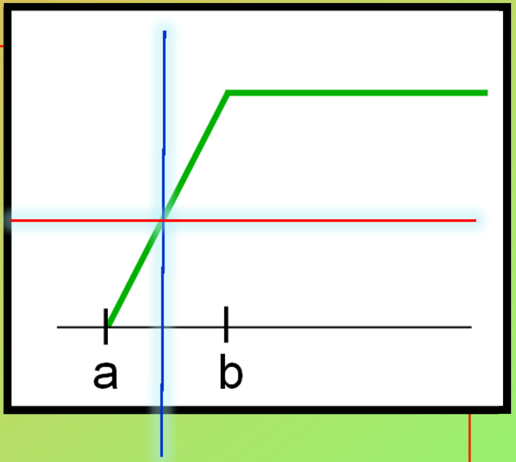
CASE 1: X < a
Membership Value = 1

CASE 2: X >= b
Membership Value = 0

CASE 3: a < x < b
Membership Value = Right_Slope * (X - b)

menu

Trapezoidal Membership Functions



RightTrapezoid

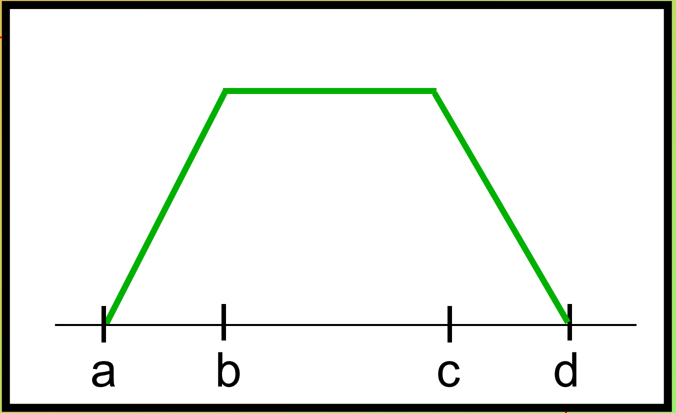
Left_Slope = 1 / (B - A)
Right_Slope = 0

CASE 1: X <= a
Membership Value = 0

CASE 2: X >= b
Membership Value = 1

CASE 3: a < x < b
Membership Value = Left_Slope * (X - a)

Trapezoidal Membership Functions



Regular Trapezoid

Left_Slope = 1 / (B - A)
Right_Slope = 1 / (C - D)

CASE 1: X <= a Or X >= d
Membership Value = 0

CASE 2: X >= b And X <= c
Membership Value = 1

CASE 3: X >= a And X <= b
Membership Value = Left_Slope * (X - a)

CASE 4: (X >= c) And (X <= d)
Membership Value = Right_Slope * (X - d)

menu

Fuzzy Control

Different stages of Fuzzy control

2. Rule Evaluation

Inputs are applied to a set of **if/then** control rules.

e.g. **IF** temperature is very hot, **THEN** set fan speed very high.

Fuzzy Control

Different stages of Fuzzy control

Fuzzy rules are always written in the following form:

If (**input1** *is membership function1)* ***and/or***
(**input2** *is membership function2)* ***and/or*** *….*

Then *(output is output membership function).*

For example, one could make up a rule that says:

if *temperature is high* ***and*** *humidity is high* ***then*** *room is hot.*

menu

Fuzzy Control

Different stages of Fuzzy control

2. Rule Evaluation

Inputs are applied to a set of **if/then** control rules.

The results of various rules are *summed together* to generate a set of “fuzzy outputs”.

FAMM

Outputs
NL=-5
NS=-2.5
ZE=0
PS=2.5
PL=5.0

		x		
		N	ZE	P
y	N	NL	NS	NS
	ZE	NS	ZE	PS
	P	PS	PS	PL

W1	**W4**	**W7**
W2	**W5**	**W8**
W3	**W6**	**W9**

Fuzzy Control

Rule Evaluation Example

Assuming that we are using the **conjunction operator (AND)** in the antecedents of the rules, we calculate the **rule firing strength Wn**.

FAMM

y \ x	N	ZE	P
N	NL	NS	NS
ZE	NS	ZE	PS
P	PS	PS	PL

W1	**W4**	**W7**
W2	**W5**	**W8**
W3	**W6**	**W9**

$$W_1 = \min\left[F_N(0.25), F_N(-0.25)\right] = \min\left[0.25, 0.75\right] = 0.25$$

$$W_2 = \min\left[F_N(0.25), F_{ZE}(-0.25)\right] = \min\left[0.25, 1\right] = 0.25$$

$$W_3 = \min\left[F_N(0.25), F_P(-0.25)\right] = \min\left[0.25, 0.25\right] = 0.25$$

$$W_4 = \min\left[F_{ZE}(0.25), F_N(-0.25)\right] = \min\left[1, 0.75\right] = 0.75$$

$$W_5 = \min\left[F_{ZE}(0.25), F_{ZE}(-0.25)\right] = \min\left[1, 1\right] = 1$$

$$W_6 = \min\left[F_{ZE}(0.25), F_P(-0.25)\right] = \min\left[1, 0.25\right] = 0.25$$

$$W_7 = \min\left[F_P(0.25), F_N(-0.25)\right] = \min\left[0.75, 0.75\right] = 0.75$$

$$W_8 = \min\left[F_P(0.25), F_{ZE}(-0.25)\right] = \min\left[0.75, 1\right] = 0.75$$

$$W_9 = \min\left[F_P(0.25), F_P(-0.25)\right] = \min\left[0.75, 0.25\right] = 0.25$$

Does a FAMM need to be a square?

Is it possible to use more than 2 input parameters for a FAMM?

menu

Fuzzy Control

Different stages of Fuzzy control

3. Defuzzification

Fuzzy outputs are combined into discrete values needed to drive the control mechanism

(e.g. A cooling fan)

We will see a complete example of the steps involved later.

Fuzzy Control

Defuzzification Example

Assuming that we are using the center of mass defuzzification method.

$$\mathrm{OUTPUT} = \frac{(W_1 \cdot NL + W_2 \cdot NS + W_3 \cdot PS + W_4 \cdot NS + W_5 \cdot ZE + W_6 \cdot PS + W_7 \cdot NS + W_8 \cdot PS + W_9 \cdot PL)}{\sum_{i=1}^{9} W_i}$$

$$= \frac{\left(0.25 \cdot (-5) + 0.25 \cdot 2.5 + 0.25 \cdot 2.5 + 0.75 \cdot 2.5 + 1 \cdot 0 + 0.25 \cdot 2.5 + 0.75 \cdot 2.5 + 0.75 \cdot 2.5 + 0.25 \cdot 5\right)}{\left(0.25 + 0.25 + 0.25 + 0.75 + 1 + 0.25 + 0.75 + 0.75 + 0.25\right)}$$

$$\boxed{= -1.25/\ 4.5\ \ = -0.278}$$

FAMM

Outputs

NL=-5

NS=-2.5

ZE=0

PS=2.5

PL=5.0

W1	**W4**	**W7**
W2	**W5**	**W8**
W3	**W6**	**W9**

x

y

	N	ZE	P
N	NL	NS	NS
ZE	NS	ZE	PS
P	PS	PS	PL

menu

Summary of Steps

To compute the output of this FIS given the inputs, one must go through six steps:

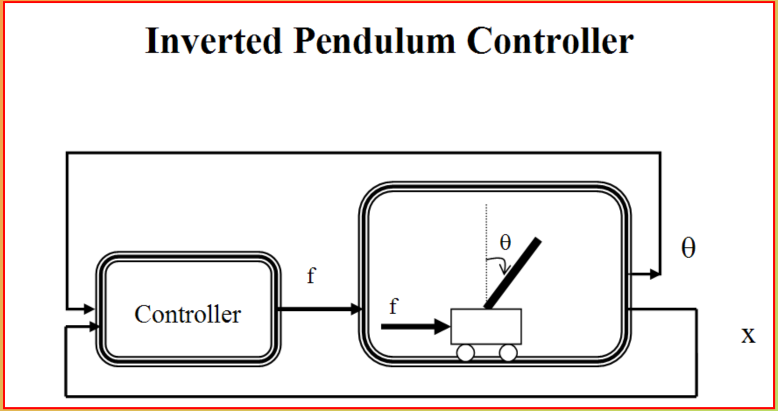
1. determining a set of fuzzy rules
2. fuzzifying the inputs using the input membership functions,
3. combining the fuzzified inputs according to the fuzzy rules to establish a rule strength,
4. finding the consequence of the rule by combining the rule strength and the output membership function (if it’s a mamdani FIS),
5. combining the consequences to get an output distribution, and
6. defuzzifying the output distribution (this step applies only if a crisp output (class) is needed).

menu

EXAMPLE

Inverted Pendulum Problem

Inverted Pendulum Problem



A Classic test case in embedded control

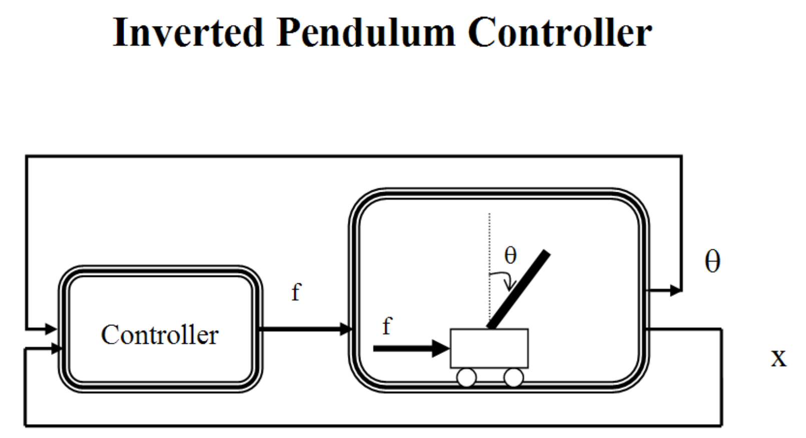
A pole with a weight on top is mounted on a motor-driven cart. The pole can swing freely, and the cart must move back and forth to keep it vertical.

A controller monitors the angle and motion of the pole and directs the cart to execute the necessary balancing movements.

A Glimpse at History: International Conference in Tokyo (1987) **Takeshi Yamakawa** demonstrated the use of fuzzy control, through a set of simple dedicated fuzzy logic chips, in an "inverted pendulum" experiment. (Later experiments: mounted a wine glass containing water or even a live mouse to the top of the pendulum).

menu

Inverted Pendulum Problem



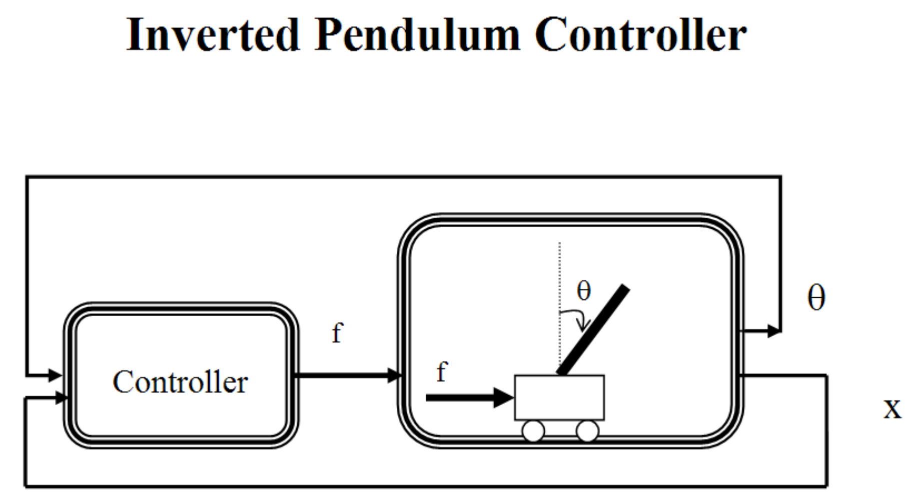
Inverted Pendulum Controller

Conventional mathematical solution

The solution uses a second-order differential equation that describes cart motion as a function of pole position and velocity:

$$\left[m\frac{\partial^2}{\partial t^2}(x + l\sin\theta) \right] l\cos\theta - \left[m\frac{\partial^2}{\partial t^2}(l\cos\theta) \right] l\sin\theta = mgl\sin\theta$$

Inverted Pendulum Problem



Sensed values:
X – position of object with respect to the horizontal axis
θ - angle of pole relative to the vertical axis

Derived values:
X' - Velocity along the x-axis
θ' - Angular velocity

Input variables: sensed and derived values

Controller output: **F** – force to be applied to the cart

menu

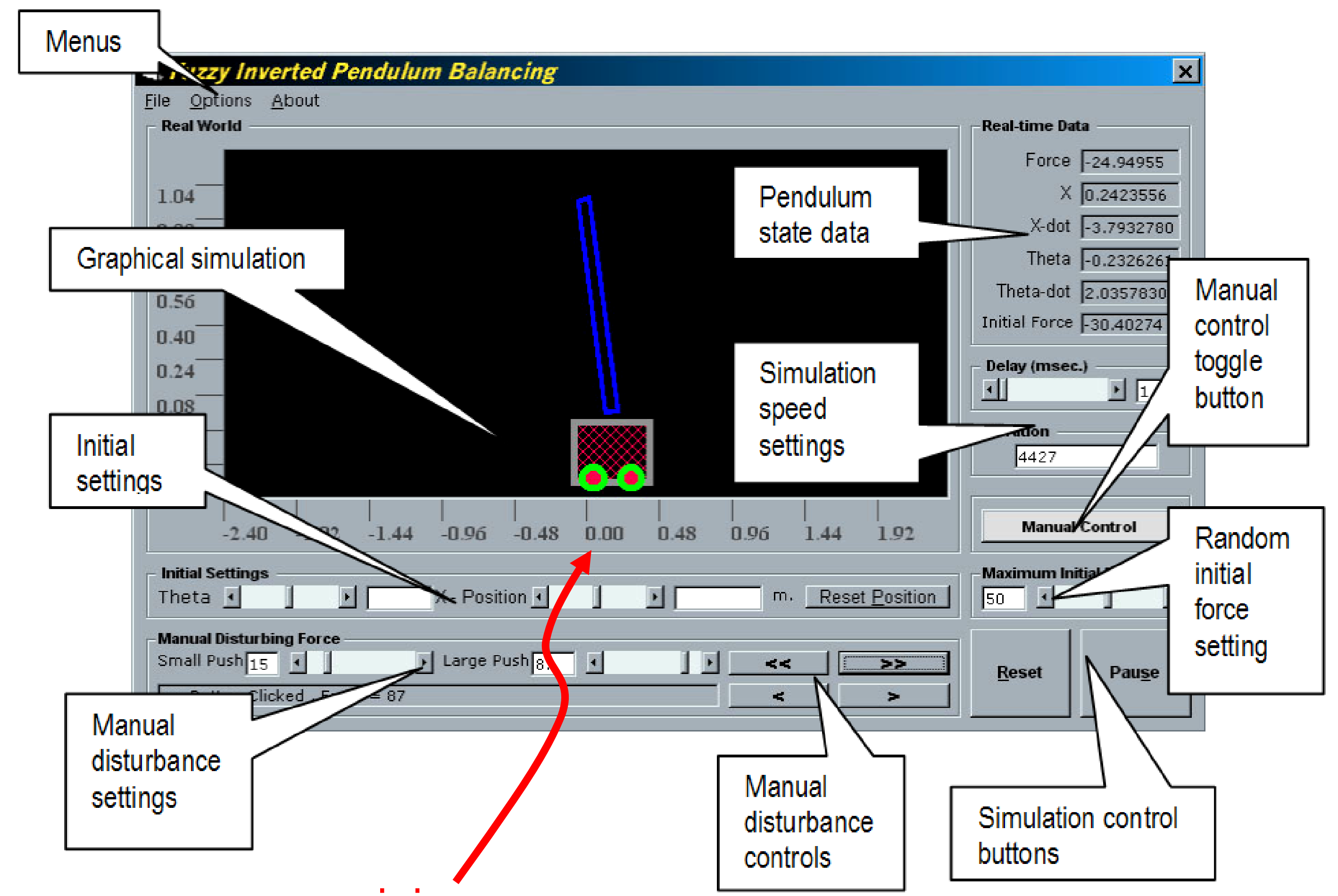
Derived Input Values

We can derive new input values for our Fuzzy Control System using Physics equations.

A sample calculation of some of the derived Values: angular velocity (theta’)

theta	time	theta'
2	1	
10	2	8
30	3	20
40	4	10
47	5	7
32	6	-15
28	7	-4
19	8	-9

Inverted Pendulum Problem



Fuzzy Inverted Pendulum Balancing

Real-time Data

Force -24.94955

X 0.2423556

X-dot -3.7932780

Theta -0.2326261

Theta-dot 2.0357830

Initial Force -30.40274

Delay (msec.)

4427

Initial Settings

Theta

X-Position

m.

Maximum Initial

50

Manual Disturbing Force

Small Push 15

Large Push 8

Clicked F = 87

Menus

Graphical simulation

Initial settings

Pendulum state data

Simulation speed settings

Manual control toggle button

Random initial force setting

Manual disturbance settings

Manual disturbance controls

Simulation control buttons

origin

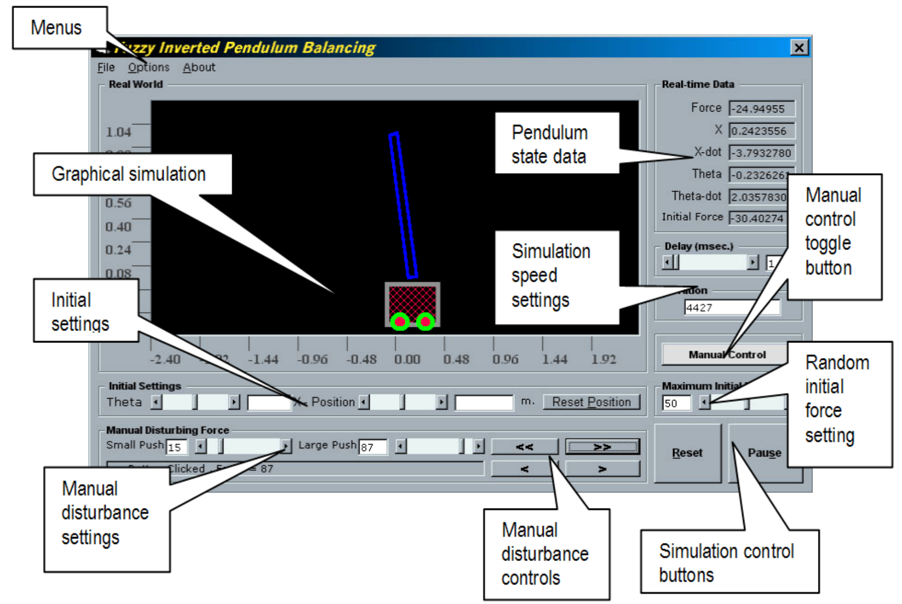
Parameters for a Fuzzy System

Once you have determined the appropriate inputs and outputs for your application, there are three steps to designing the parameters for a fuzzy system:

1. specify the fuzzy sets to be associated with each variable.
2. decide on what the fuzzy rules are going to be.
3. specify the shape of the membership functions.

menu

Fuzzy Sets



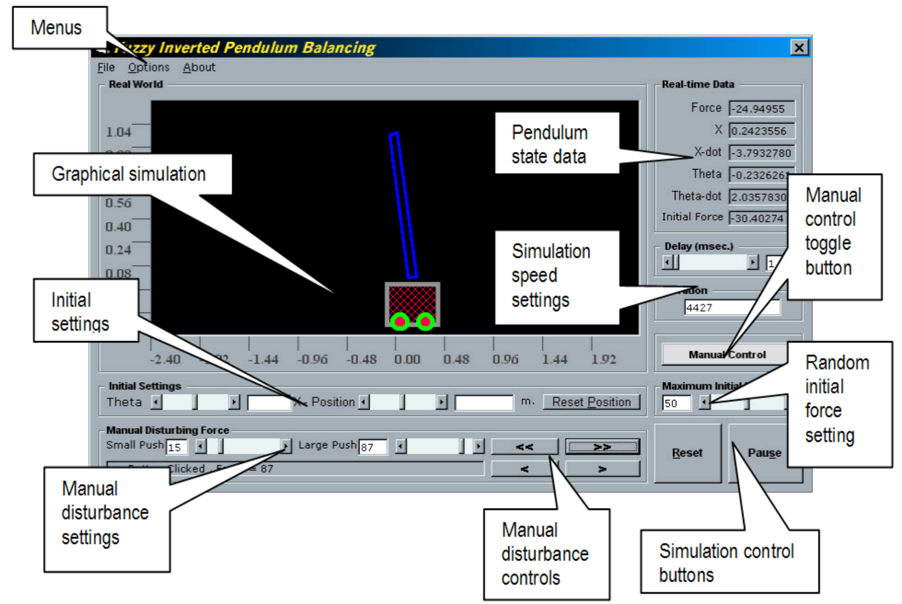
Force -24.94955
X 0.2423556
X-dot -3.7932780
Theta -0.232626
Theta-dot 2.0357830
Initial Force -30.40274
4427
Maximum Initial 50
Small Push 15
Large Push 87

We might begin designing a fuzzy system by subdividing the two input variables (pole angle and angular velocity) into membership sets.

The **angle** could be described as:

1. Inclined to the Left (N).
2. Vertical (Zero).
3. Inclined to the Right (P).

Fuzzy Sets



Force -24.94955
X 0.2423556
X-dot -3.7932780
Theta -0.232626
Theta-dot 2.0357830
Initial Force -30.40274
4427
Maximum Initial 50
Small Push 15
Large Push 87

The **angular velocity** could be described as:

1. Falling to the Left (N).
2. Still (Zero).
3. Falling to the Right (P).

menu

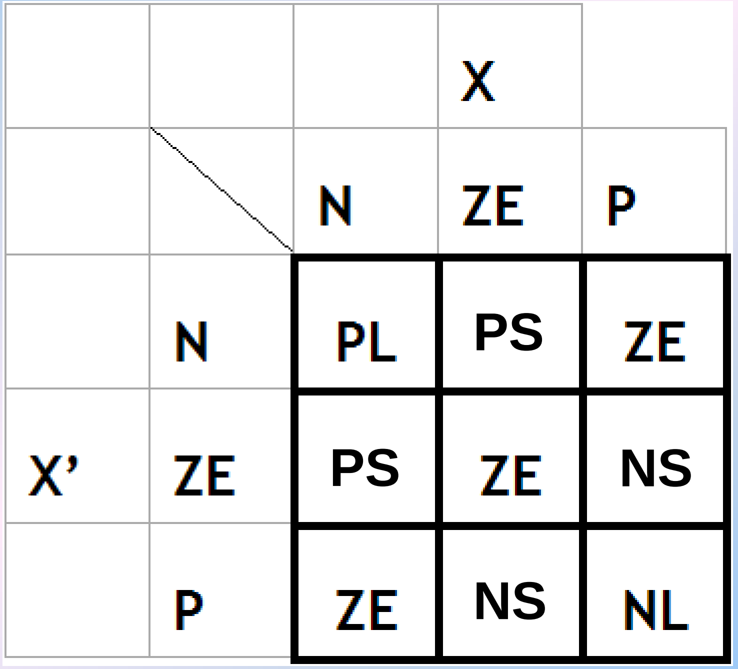
Fuzzy Rules (FAMM 1)

Fuzzy rule base and the corresponding FAMM for the **velocity and position** vectors of the inverted pendulum-balancing problem

1. IF cart is on the left AND cart is going left THEN largely push cart to the right
2. IF cart is on the left AND cart is not moving THEN slightly push cart to the right
3. IF cart is on the left AND cart is going right THEN don't push cart
4. IF cart is centered AND cart is going left THEN slightly push cart to the right
5. IF cart is centered AND cart is not moving THEN don't push cart
6. IF cart is centered AND cart is going right THEN slightly push cart to the left
7. IF cart is on the right AND cart is going left THEN don't push cart
8. IF cart is on the right AND cart is not moving THEN push cart to the left
9. IF cart is on the right AND cart is going right THEN largely push cart to the left

Position vs. Velocity (FAMM 1)

If the cart is too near the end of the path, then regardless of the state of the broom angle push the cart towards the other end.



			X	
		N	ZE	P
	N	PL	PS	ZE
X’	ZE	PS	ZE	NS
	P	ZE	NS	NL

3 x 3 FAMM

menu

Fuzzy Rules (FAMM 2)

Fuzzy rule base and the corresponding FAMM for the **angle and angular velocity** vectors of the inverted pendulum-balancing problem

1. IF pole is leaning to the left AND pole is dropping to the left THEN largely push cart to the left
2. IF pole is leaning to the left AND pole is not moving THEN slightly push cart to the left
3. IF pole is leaning to the left AND pole is moving to the right THEN don’t push the cart

…. and so on, and so forth

….

….

Angle vs. Angular Velocity (FAMM 2)

If the broom angle is too big or changing too quickly, then regardless of the location of the cart on the cart path, push the cart towards the direction it is leaning to.

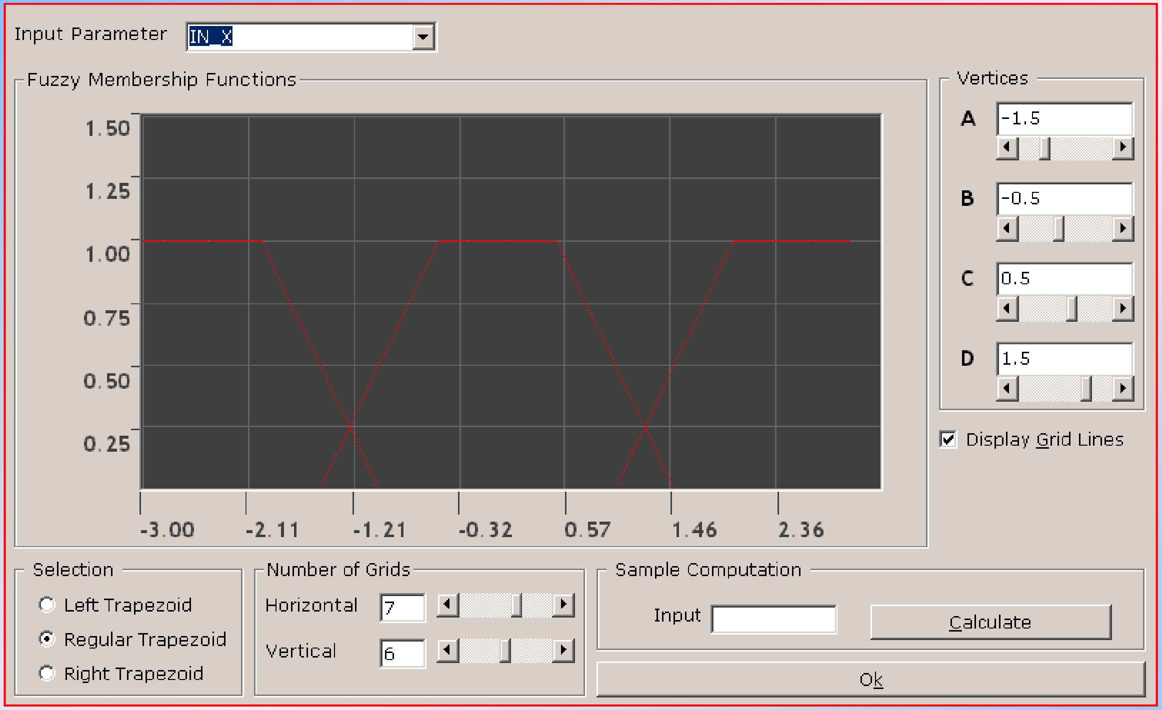
			θ	
		N	ZE	P
	N	NL	NM	ZE
θ'	ZE	NS	ZE	PS
	P	ZE	PM	PL

3 x 3 FAMM

menu

Membership Functions

Membership Functions for the Cart Position x



Input Parameter: IN_X

A: -1.5

B: -0.5

C: 0.5

D: 1.5

☑ Display Grid Lines

○ Left Trapezoid
● Regular Trapezoid
○ Right Trapezoid

Horizontal: 7

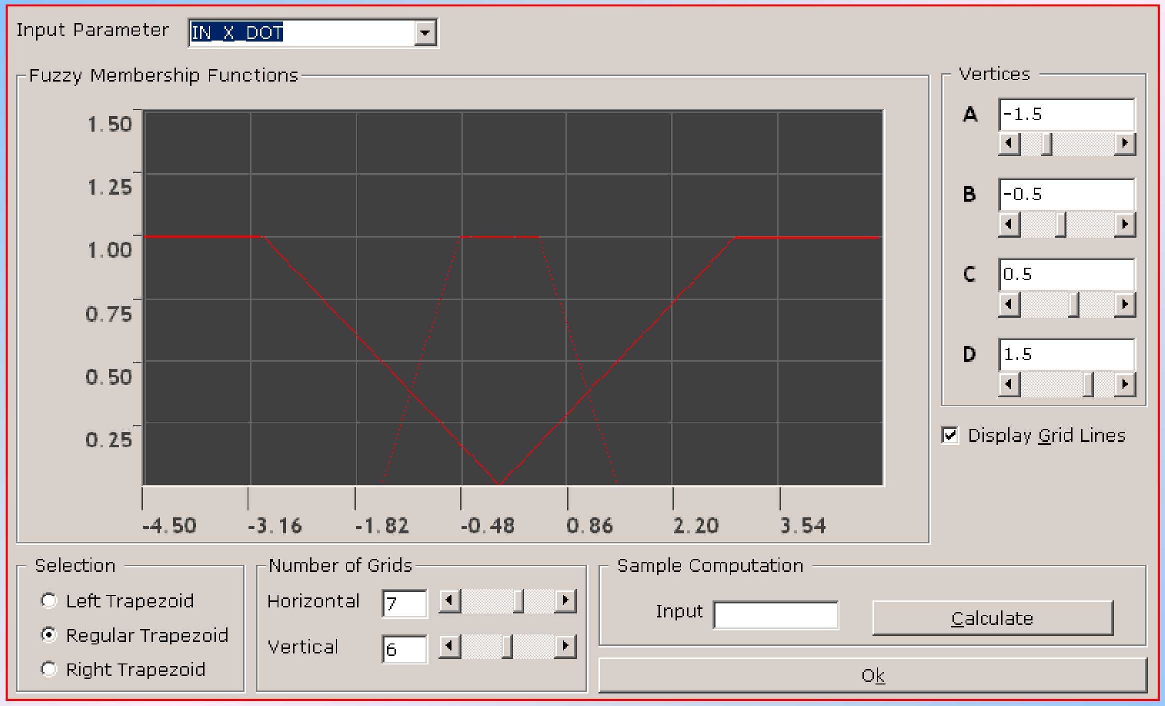
Vertical: 6

Take note of the position of the origin.

Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -2	A = -1.5	A = 1
B = -1	B = -0.5	B = 2
C = 0	C = 0.5	C = 0
D = 0	D = 1.5	D = 0

Membership Functions

Membership Functions for the Cart Velocity


Input Parameter: IN_X_DOT

Vertices
A: -1.5
B: -0.5
C: 0.5
D: 1.5

☑ Display Grid Lines

Selection
○ Left Trapezoid
● Regular Trapezoid
○ Right Trapezoid

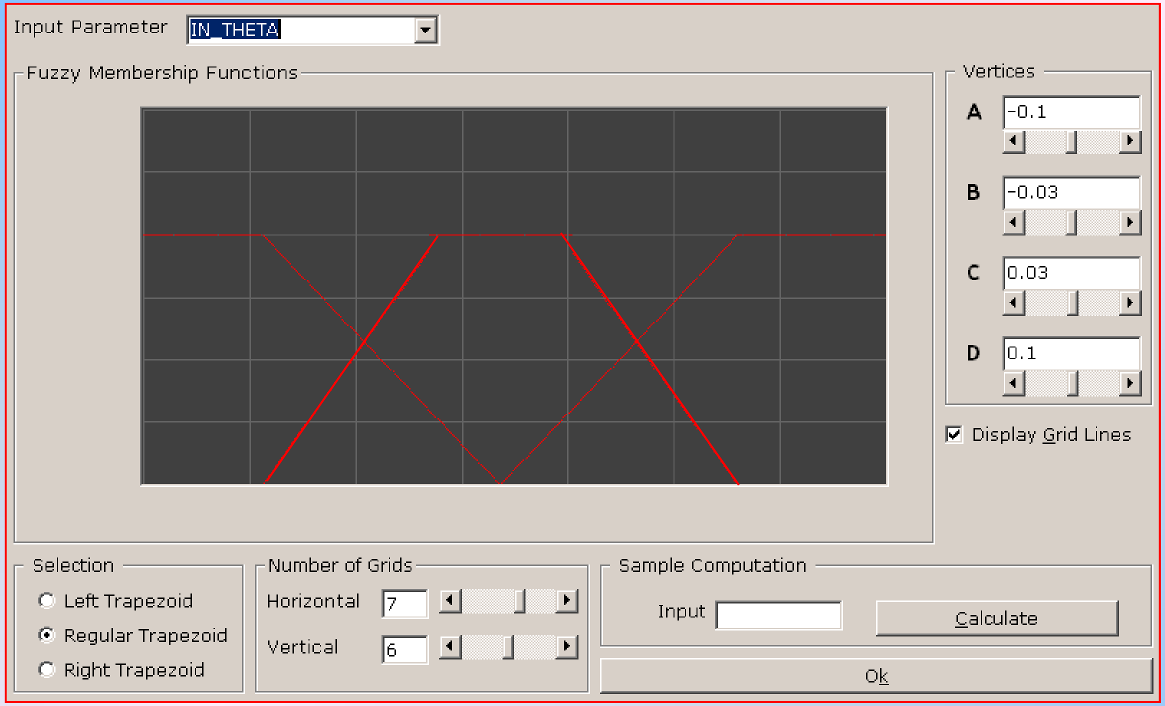
Number of Grids
Horizontal: 7
Vertical: 6

Sample Computation
Input:

Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -3	A = -1.5	A = 0
B = 0	B = -0.5	B = 3
C = 0	C = 0.5	C = 0
D = 0	D = 1.5	D = 0

Membership Functions

Membership Functions for the Pole Angle



Input Parameter: IN_THETA

A: -0.1
B: -0.03
C: 0.03
D: 0.1

☑ Display Grid Lines

Selection
○ Left Trapezoid
● Regular Trapezoid
○ Right Trapezoid

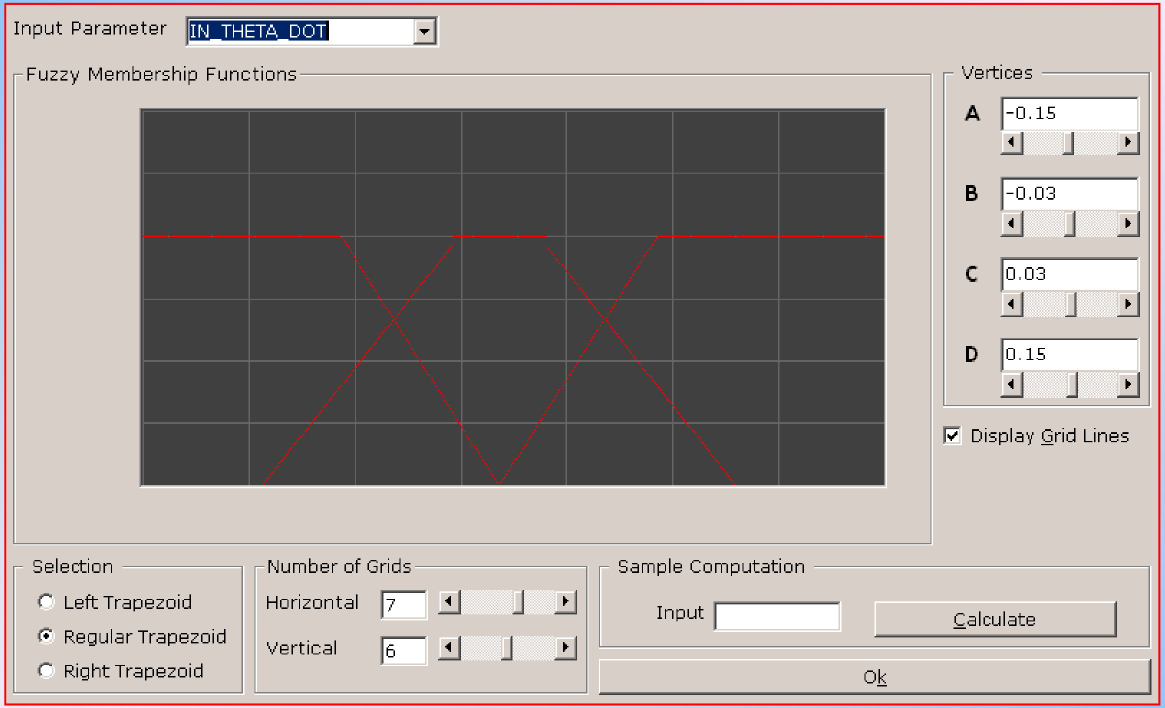
Number of Grids
Horizontal: 7
Vertical: 6

Sample Computation
Input:

Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -0.1	A = -0.1	A = 0
B = 0	B = -0.03	B = 0.1
C = 0	C = 0.03	C = 0
D = 0	D = 0.1	D = 0

Membership Functions

Membership Functions for the Broom Angular Velocity



Input Parameter: IN_THETA_DOT

A: -0.15
B: -0.03
C: 0.03
D: 0.15

☑ Display Grid Lines

○ Left Trapezoid
● Regular Trapezoid
○ Right Trapezoid

Horizontal: 7
Vertical: 6

Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -0.1	A = -0.15	A = 0
B = 0	B = -0.03	B = 0.1
C = 0	C = 0.03	C = 0
D = 0	D = 0.15	D = 0

- OBSTACLE AVOIDANCE - FUZZY LECTURE.doc
- Lec2 - Fuzzy Logic - FuzzyEqns.doc

Fuzzy Rule

If Distance is NEAR **and** Angle is SMALL **Then** Turn Sharp Left.

F_{NEAR}(Distance) = degree of membership of the given distance in the Fuzzy Set NEAR

Could be a constant or another MF

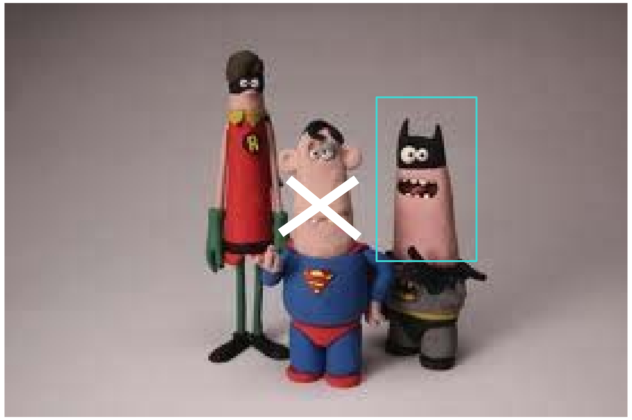
F_{SMALL}(Angle) = degree of membership of the given angle in the Fuzzy Set SMALL

menu

Fuzzy System Example

Autofocusing System

Autofocusing System



Cameras are usually equipped with an auto-focusing feature that estimates the distance to the center of a finder’s view.

This method, however, would not work all the time, as the object of interest may not always be at the center of the view.

menu

Autofocusing System

Cameras are usually equipped with an auto-focusing feature that estimates the distance to the center of a finder's view.

This method, however, would not work all the time, as the object of interest may not always be at the center of the view.

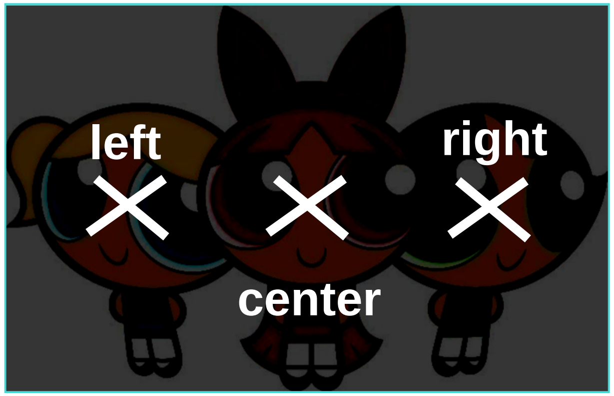
Better focusing results can be achieved by utilising several measures of distance from the scene.

menu

Autofocusing System

Objective

Determine the object distance using three distance measures for an automatic camera focusing system.



3 Distance Measures

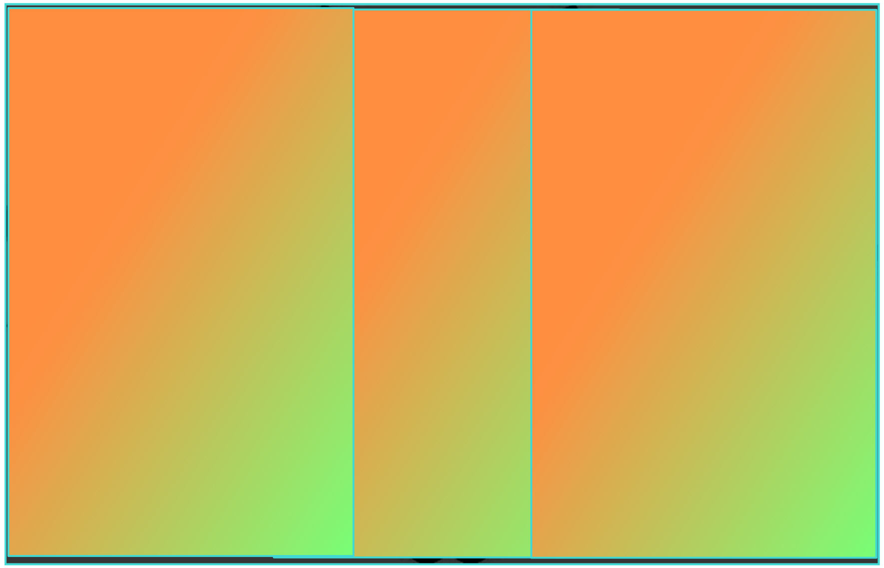
Autofocusing System

Objective

Determine the object distance using three distance measures for an automatic camera focusing system.

Definition of Input/Output Variables

Inputs to the fuzzy inference system are 3 distance measures at **left**, **center**, **right** points in the finder view. Outputs are the plausibility values associated with these 3 points. The point with the highest plausibility is deemed to be the object of interest. Its distance is then forwarded to the automatic focusing system.



3 Distance Measures

menu

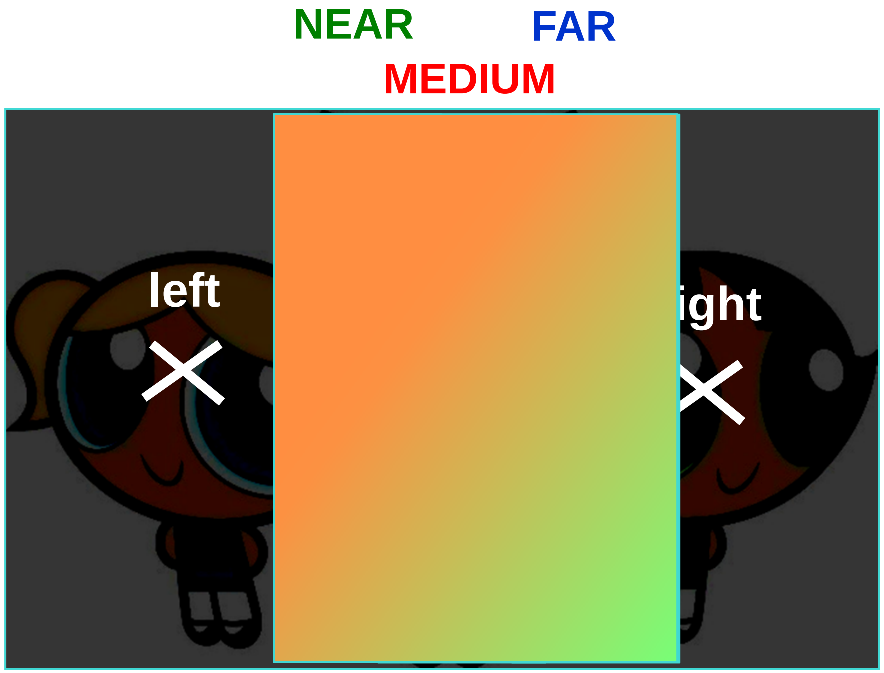
Autofocusing System

Objective

Determine the object distance using three distance measures for an automatic camera focusing system.

Definition of Input/Output Variables

Inputs to the fuzzy inference system are 3 distance measures at **left**, **center**, **right** points in the finder view. Outputs are the plausibility values associated with these 3 points. The point with the highest plausibility is deemed to be the object of interest. Its distance is then forwarded to the automatic focusing system.



3 Distance Measures

menu

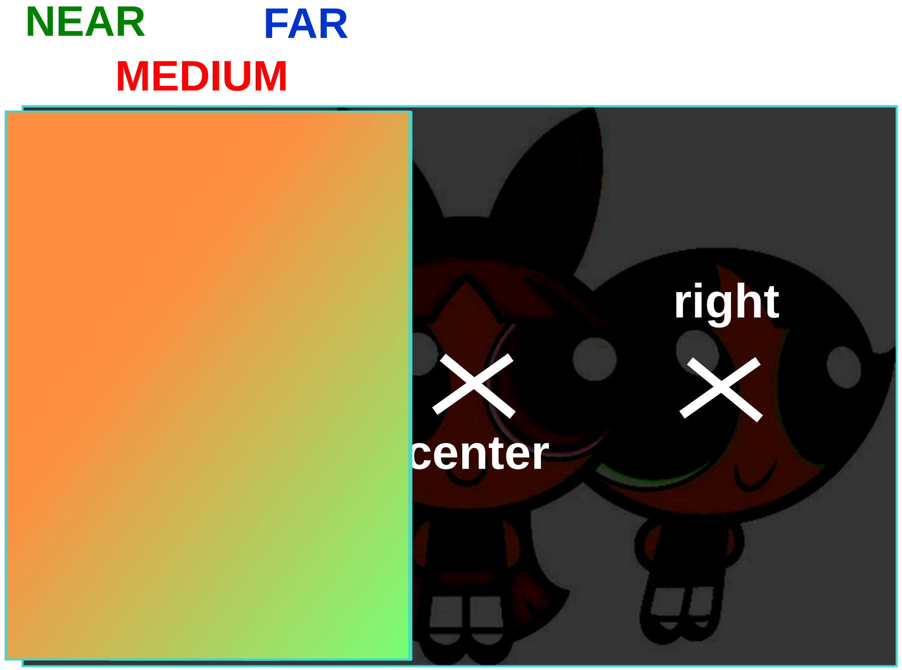
Autofocusing System

Objective

Determine the object distance using three distance measures for an automatic camera focusing system.

Definition of Input/Output Variables

Inputs to the fuzzy inference system are 3 distance measures at **left**, **center**, **right** points in the finder view. Outputs are the plausibility values associated with these 3 points. The point with the highest plausibility is deemed to be the object of interest. Its distance is then forwarded to the automatic focusing system.



3 Distance Measures

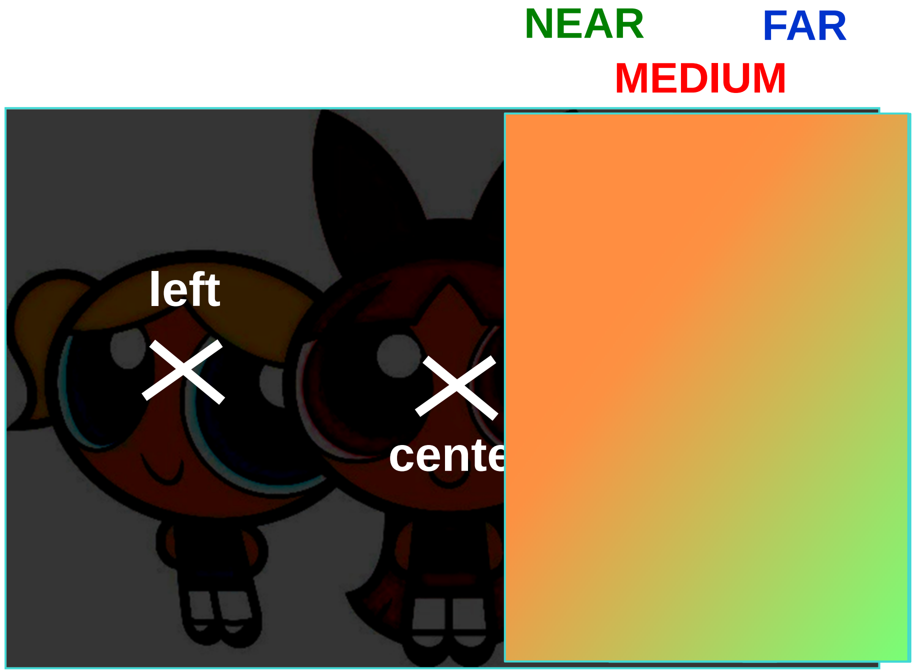
Autofocusing System

Objective

Determine the object distance using three distance measures for an automatic camera focusing system.

Definition of Input/Output Variables

Inputs to the fuzzy inference system are 3 distance measures at **left**, **center**, **right** points in the finder view. Outputs are the plausibility values associated with these 3 points. The point with the highest plausibility is deemed to be the object of interest. Its distance is then forwarded to the automatic focusing system.

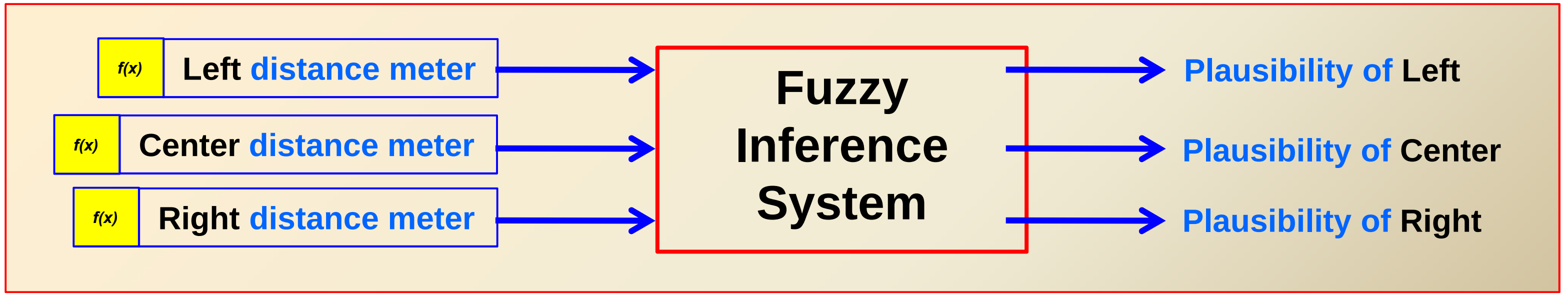


3 Distance Measures

menu

Autofocusing System

Fuzzy Inference



Each **input variable**, representing distance, has 3 fuzzy sets:

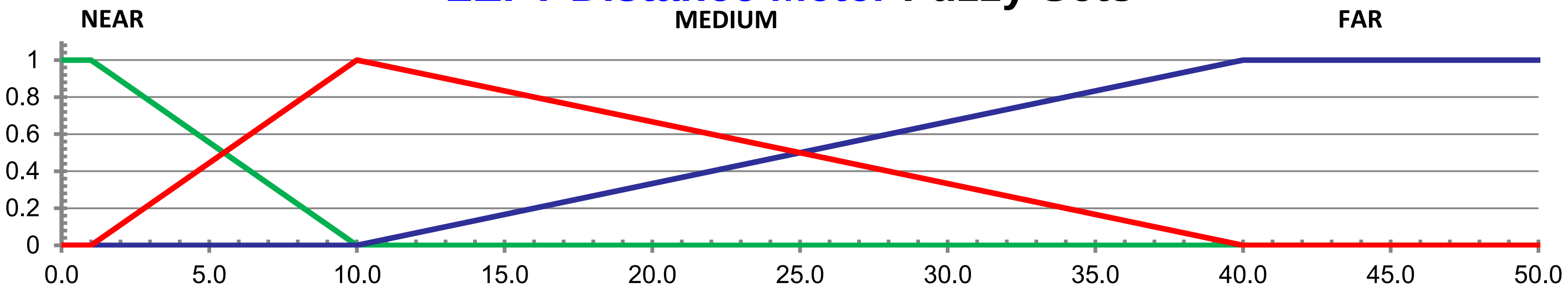
- **Near**
- **Medium**
- **Far**

Each **output variable**, representing plausibility, has 4 fuzzy sets:

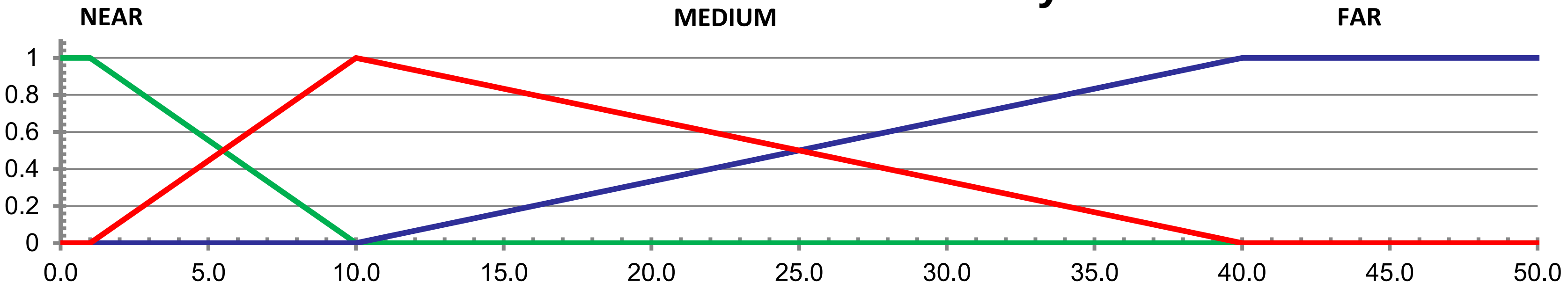
- **Low**
- **Medium**,
- **High**
- **Very High**

Input Variables’ Fuzzy Sets

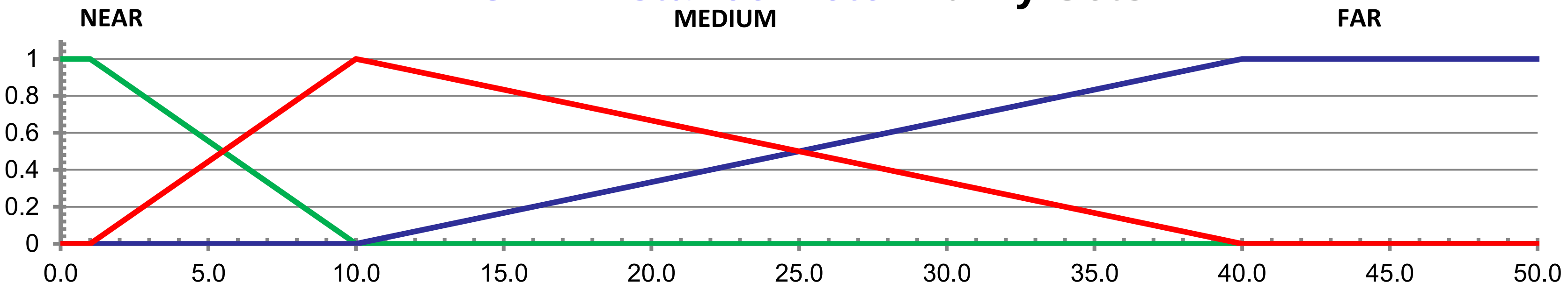
LEFT Distance Meter Fuzzy Sets



CENTER Distance Meter Fuzzy Sets



RIGHT Distance Meter Fuzzy Sets



menu

Definition of Output Variables

Output variable: **Plausibility of Left**:

- **Low** = 0.3
- **Medium** = 0.5
- **High** = 0.8
- **Very High** = 1.0

Output variable: **Plausibility of Center**:

- **Low** = 0.3
- **Medium** = 0.5
- **High** = 0.8
- **Very High** = 1.0

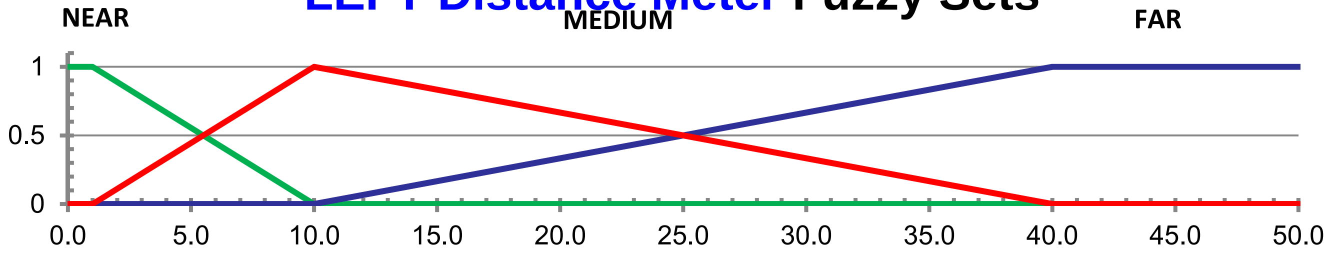
Output variable: **Plausibility of Right**:

- **Low** = 0.3
- **Medium** = 0.5
- **High** = 0.8
- **Very High** = 1.0

menu

Rules

LEFT Distance Meter Fuzzy Sets



The guiding principle for establishing rules of this autofocusing system is that the likelihood of an object being at medium distance (typically 10 meters) is high, and becomes very low as the distance increases (say, more than 40 meters).

If (Left is **Near**) **then** Plausibility of Left is **Medium**.

If (Center is **Near**) **then** Plausibility of Center is **Medium**.

If (Right is **Near**) **then** Plausibility of Right is **Medium**.

If (Left is **Near**) and (Center is **Near**) and (Right is **Near**) **then** Plausibility of Center is **High**.

If (Left is **Near**) and (Center is **Near**) **then** Plausibility of Left is **Low**.

If (Right is **Near**) and (Center is **Near**) **then** Plausibility of Right is **Low**.

If (Left is **Far**) **then** Plausibility of Left is **Low**.

If (Center is **Far**) **then** Plausibility of Center is **Low**.

If (Right is **Far**) **then** Plausibility of Right is **Low**.

If (Left is **Far**) and (Center is **Far**) and (Right is **Far**) **then** Plausibility of Center is **High**.

If (Left is **Medium**) **then** Plausibility of Left is **High**.

If (Center is **Medium**) **then** Plausibility of Center is **High**.

If (Right is **Medium**) **then** Plausibility of Right is **High**.

If (Left is **Medium**) and (Center is **Medium**) and (Right is **Medium**) **then** Plausibility of Center is **Very High**.

If (Left is **Medium**) and (Center is **Medium**) **then** Plausibility of Left is **Low**.

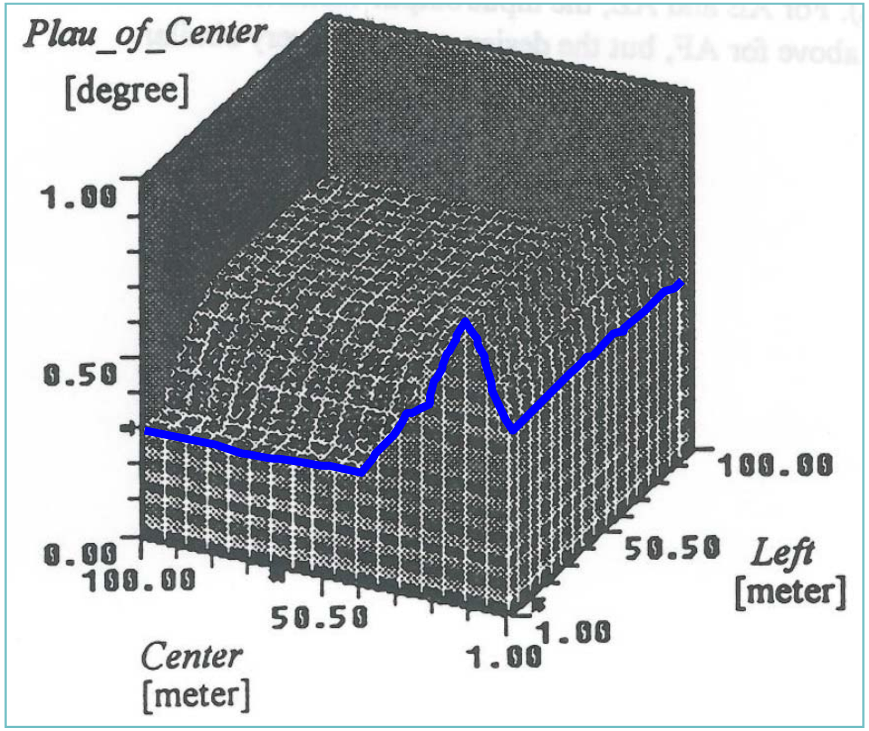
If (Right is **Medium**) and (Center is **Medium**) **then** Plausibility of Right is **Low**.

If (Left is **Medium**) and (Center is **Far**) **then** Plausibility of Center is **Low**.

If (Right is **Medium**) and (Center is **Far**) **then** Plausibility of Center is **Low**.

menu

Input/Output Response: Plausibility of Center

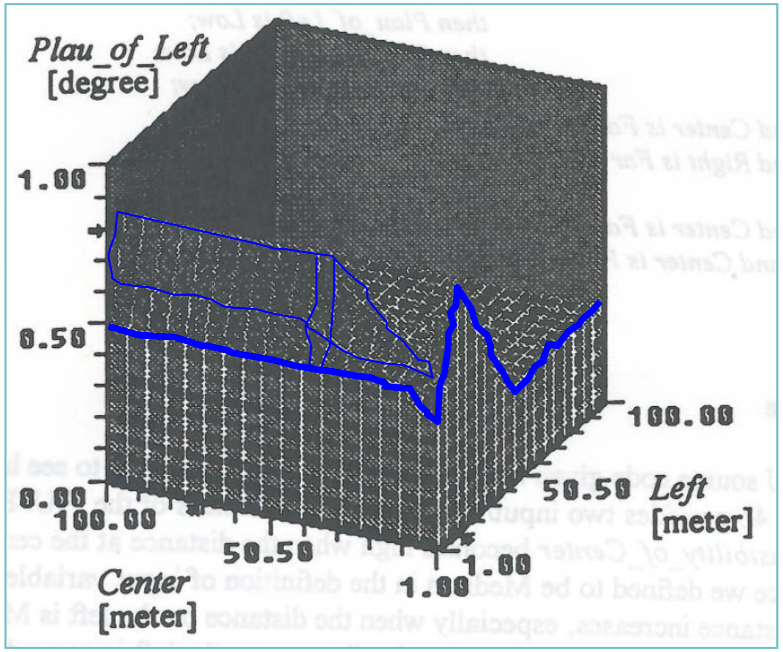


The plausibility of center becomes high when the distance at the center is around 10 meters, a distance we defined to be **Medium** in the definition of **input variables**.

It becomes lower when the distance increases, especially when the distance on the left is Medium.

menu

Input/Output Response: Plausibility of Left



The **plausibility of left** becomes high when the distance on the **left** is around 10 meters, a distance we defined to be **Medium** in the definition of **input variables**, except in the case when the distance at the center is also around 10 meters. In that case, we choose center as the desired object.

The **plausibility of right** is similar.

The **highest output** is the one that is considered. The distance at that point is the focus distance.

menu

More details…

- Fuzzy Inference Systems (FIS)
- Fuzzy Rules
- Fuzzy Combination Operators
- Membership functions

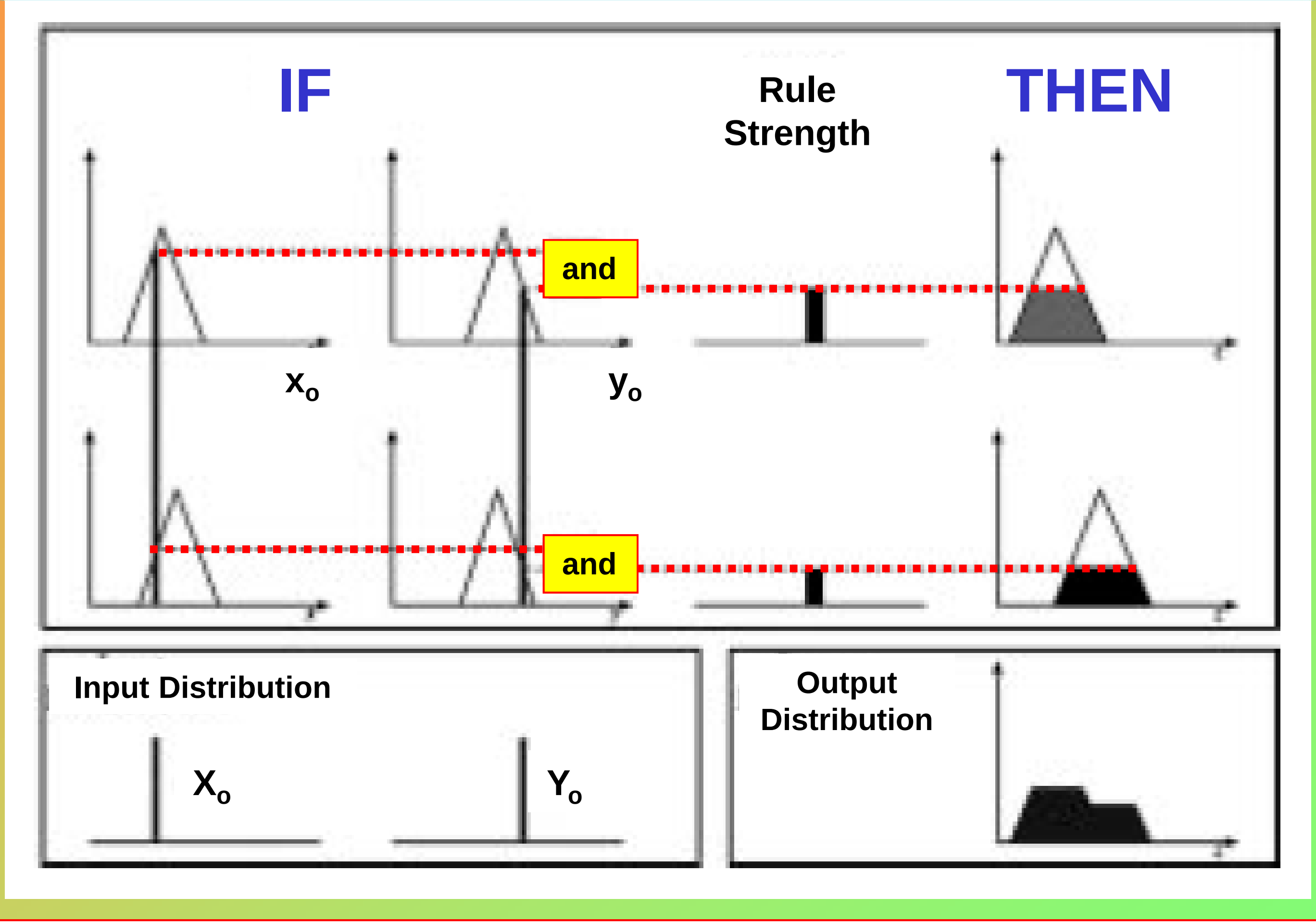
Fuzzy Inference

Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference involves all of the pieces that are described in the previous sections: Membership Functions, Logical Operations, and If-Then Rules.

menu

Mamdani Inference System

Two input, two rule Mamdani FIS with crisp inputs

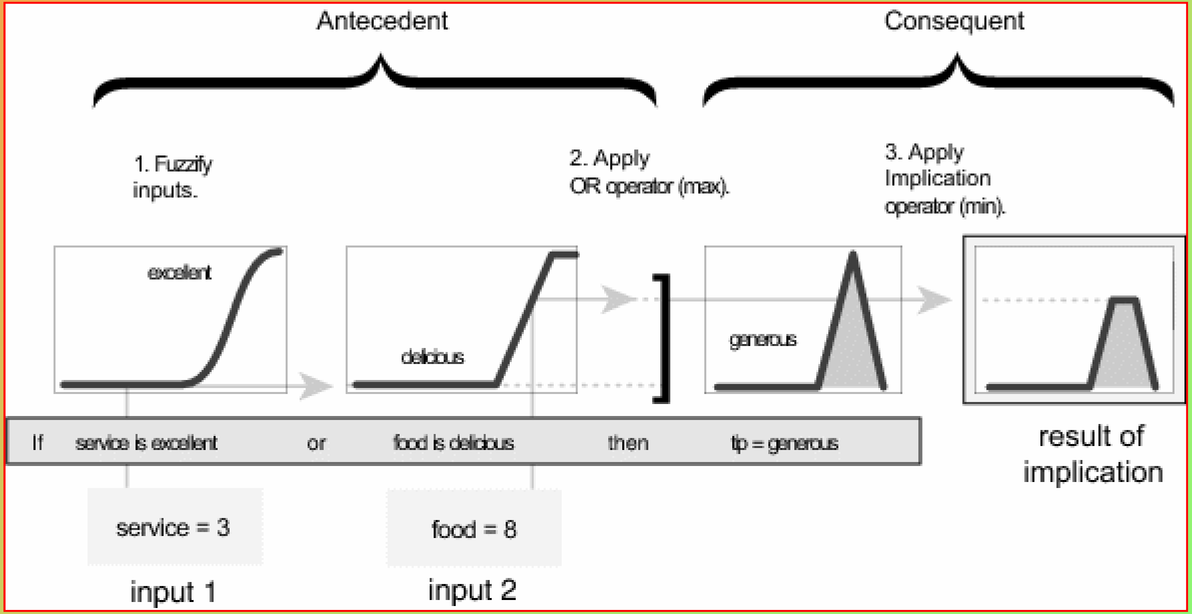


Fuzzy rules are a collection of linguistic statements that describe how the FIS should make a decision regarding classifying an input or controlling an output.

menu

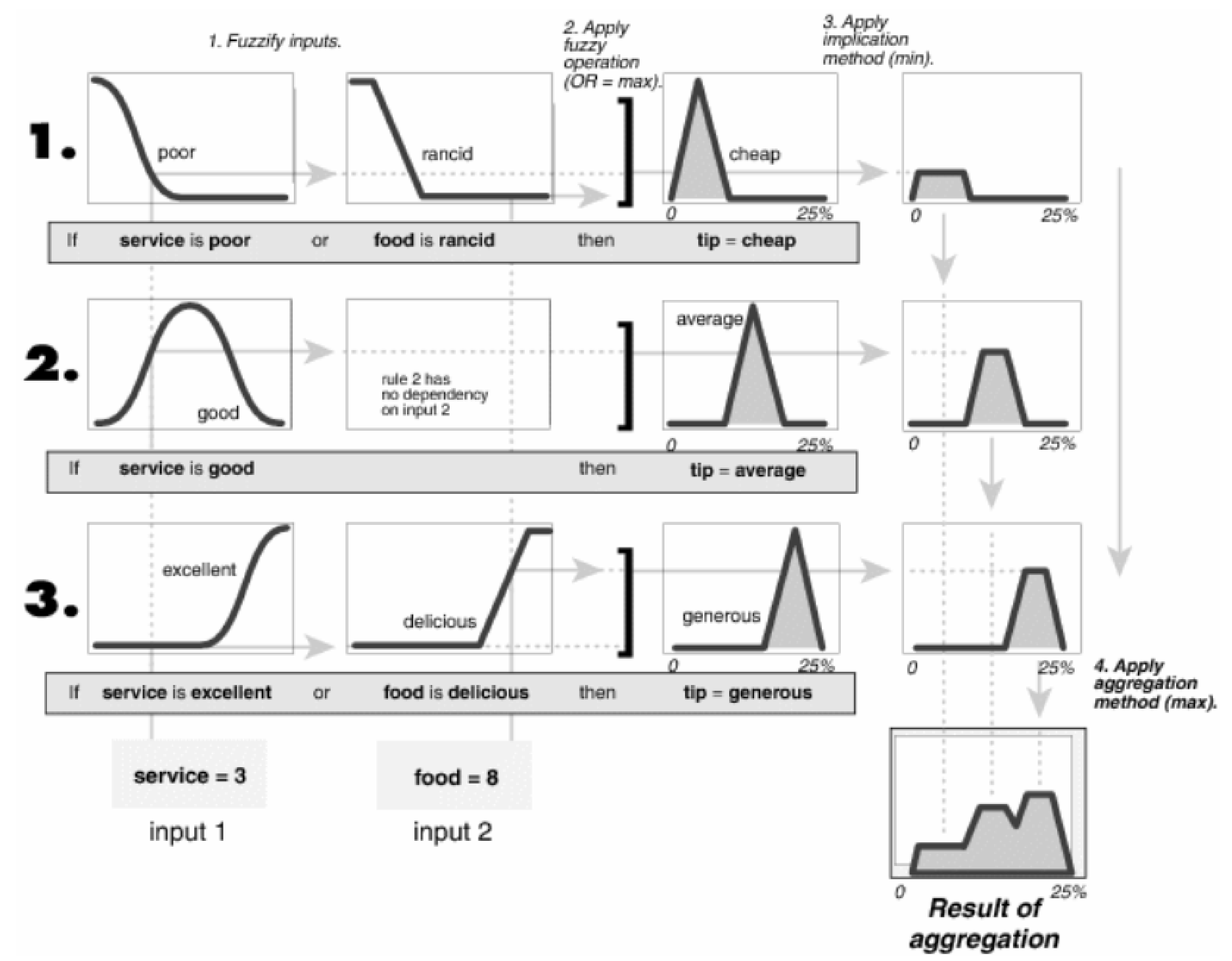
Mamdani FIS

Mamdani-type inference, expects the output membership functions to be fuzzy sets. After the aggregation process, there is a fuzzy set for each output variable that needs defuzzification.



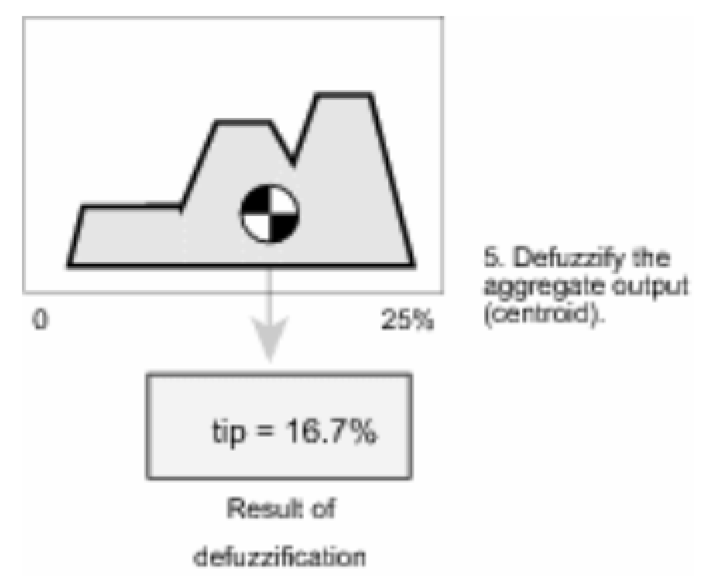
http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/index.html?/access/helpdesk/help/toolbox/fuzzy/bp78l6_-1.html&http://www.mathworks.com/cgi-bin/texis/webinator/search/?db=MSS&prox=page&rorder=750&rprox=750&rdfreq=500&rwfreq=500&rlead=250&sufs=0&order=r&is_summary_on=1&ResultCount=10&query=fuzzy+logic&submitButtonName=Search

Mamdani FIS



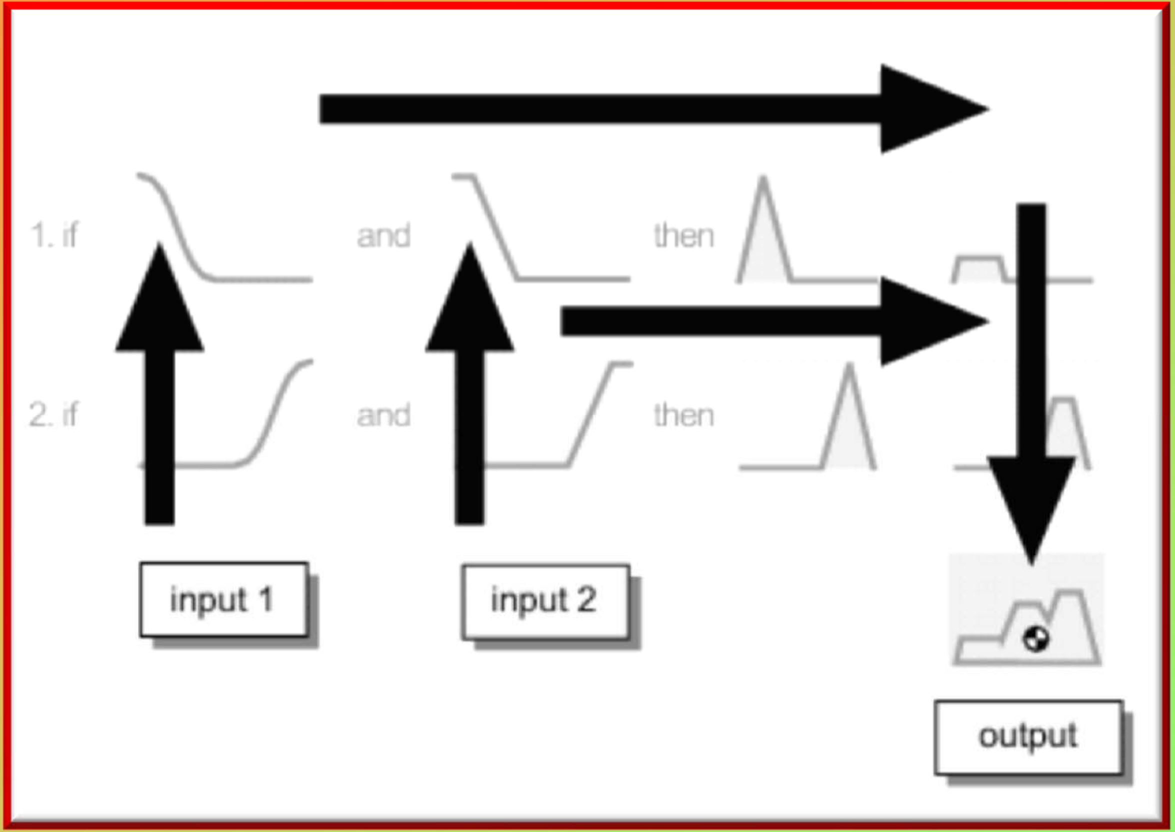
menu

Mamdani FIS



5. Defuzzify the aggregate output (centroid).

Flow of Fuzzy Inference



In this figure, the flow proceeds up from the inputs in the lower left, then across each row, or rule, and then down the rule outputs to finish in the lower right. This compact flow shows everything at once, from linguistic variable fuzzification all the way through defuzzification of the aggregate output.

Mamdani FIS

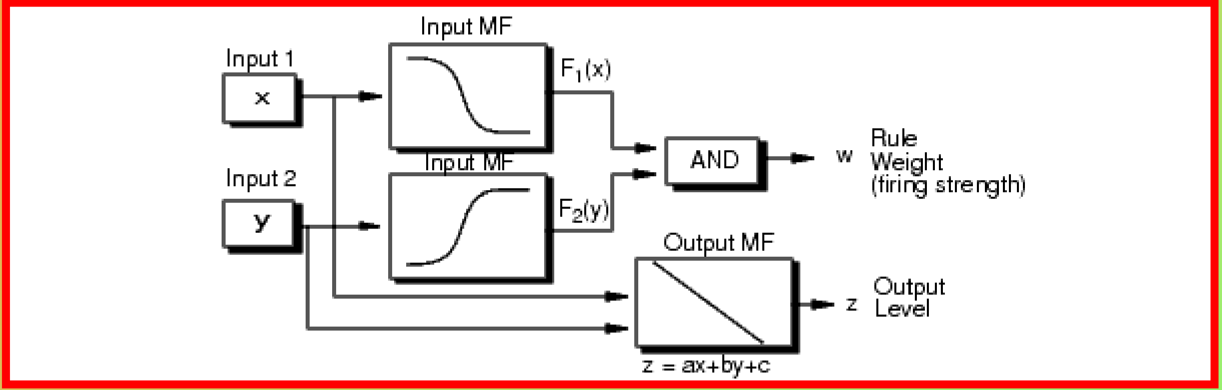
OUTPUT MEMBERSHIP FUNCTION

- It is possible, and in many cases much more efficient, to use a **single spike** as the **output membership function** rather than a distributed fuzzy set.
- This type of output is sometimes known as a **singleton output membership function**, and it can be thought of as a pre-defuzzified fuzzy set.
- It enhances the efficiency of the defuzzification process because it greatly simplifies the computation required by the more general Mamdani method, which finds the **centroid of a 2-D function**.
- Rather than integrating across the two-dimensional function to find the centroid, you use the **weighted average of a few data points**.

menu

Sugeno FIS

Sugeno FIS is similar to the Mamdani method in many respects. The first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same. The main difference between Mamdani and Sugeno is that the Sugeno output membership functions are either linear or constant.



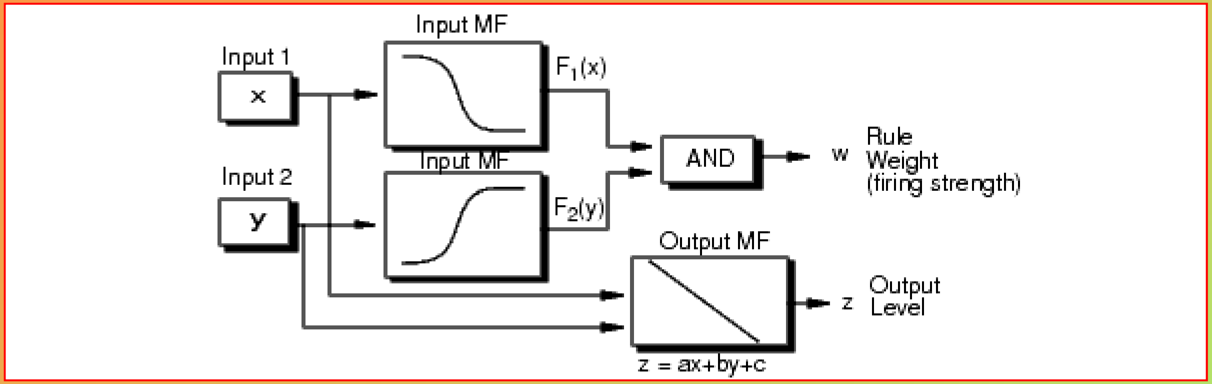
A typical rule in a Sugeno fuzzy model has the form:

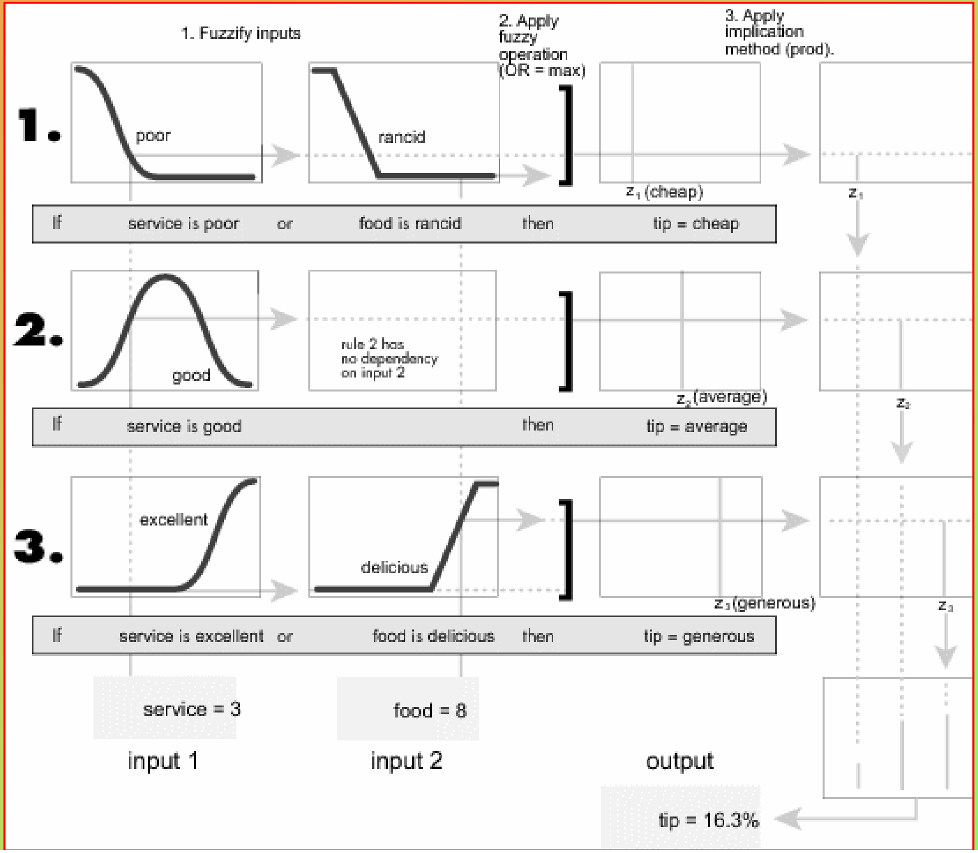
If Input 1 = x and Input 2 = y, then Output is **z = ax + by + c**

For a **zero-order Sugeno model**, the output level z is a **constant** (a=b =0).

menu

Sugeno FIS



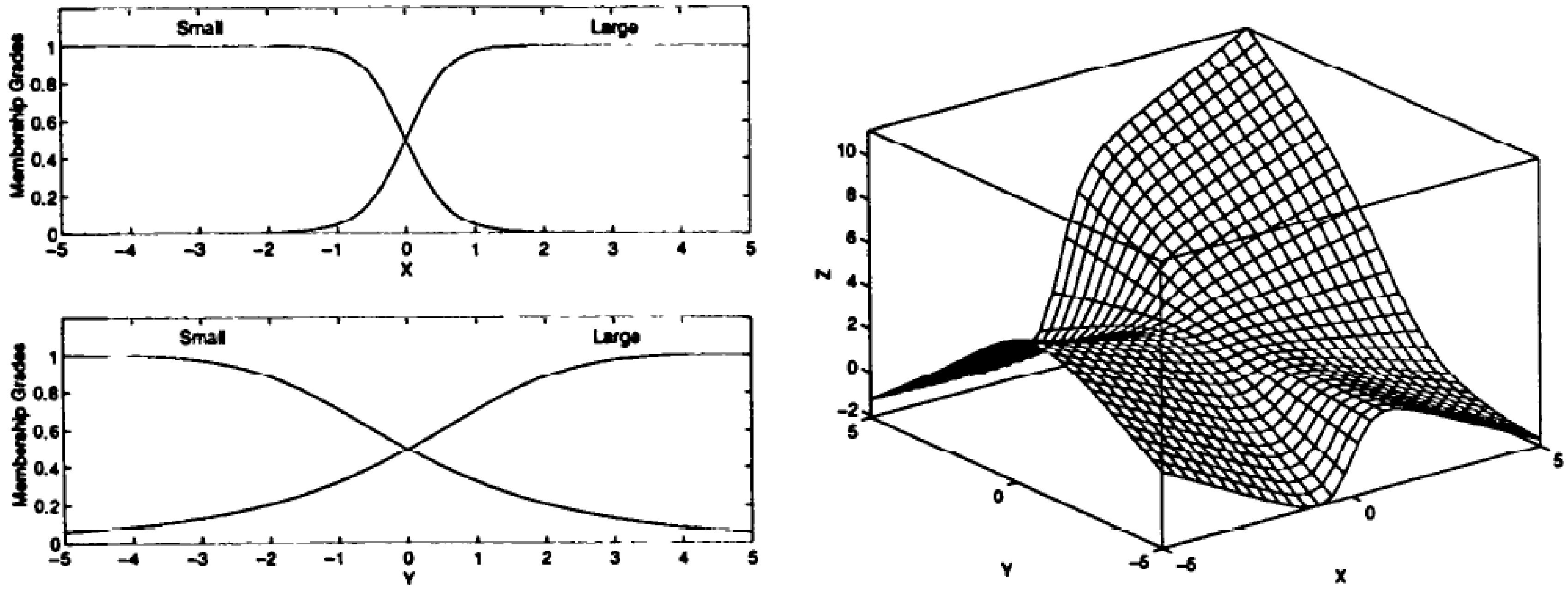


$$\text{Final Output} = \frac{\sum_{i=1}^{N} w_i z_i}{\sum_{i=1}^{N} w_i}$$

menu

2-input, single-output Sugeno fuzzy model

$$\begin{cases} \text{If } X \text{ is small and } Y \text{ is small then } z = -x + y + 1. \\ \text{If } X \text{ is small and } Y \text{ is large then } z = -y + 3. \\ \text{If } X \text{ is large and } Y \text{ is small then } z = -x + 3. \\ \text{If } X \text{ is large and } Y \text{ is large then } z = x + y + 2. \end{cases}$$



(a) (b)

FIS: Sugeno vs. Mamdani

Advantages of the Sugeno Method

It is computationally efficient.
It can be used to model any inference system in which the output membership functions are either linear or constant.
It works well with linear techniques (e.g., PID control).
It works well with optimization and adaptive techniques.
It has guaranteed continuity of the output surface.
It is well suited to mathematical analysis.

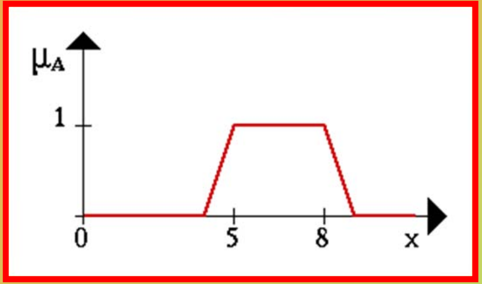
Advantages of the Mamdani Method

It is intuitive.
It has widespread acceptance.
It is well suited to human input.

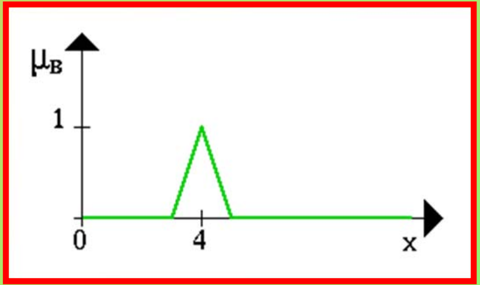
Fuzzy Sets

We will use the following fuzzy sets in explaining the different fuzzy operators that follows next.

Examples:



Fuzzy Set **A**



Fuzzy Set **B**

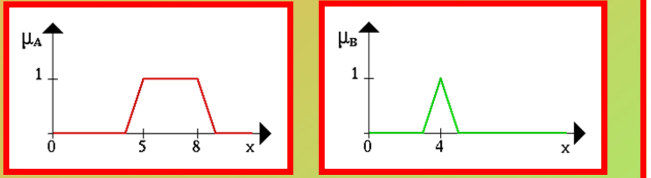
menu

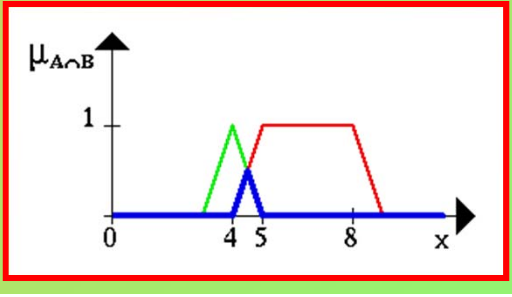
Fuzzy combinations (T-norms)

In making a fuzzy rule, we use the concept of “**and**”, “**or**”, and sometimes “**not**”. The sections below describe the most common definitions of these “fuzzy combination” operators. Fuzzy combinations are also referred to as “**T-norms**”.

Fuzzy “and”

Example:





Intersection of A and B

The fuzzy “and” is written as:

$$u_{A \cap B} = T(u_A(x), u_B(x))$$

where μ_A is read as “the membership in class A” and μ_B is read as “the membership in class B”.

Fuzzy “and”

There are many ways to compute “**and**”. The two most common are:

Zadeh - min(μ_A(x), μ_B(x))

This technique, named after the inventor of fuzzy set Theory; it simply computes the “and” by taking the **minimum** of the two (or more) membership values. This is the most common definition of the fuzzy “and”.

Product - μ_A(x) * μ_B(x)

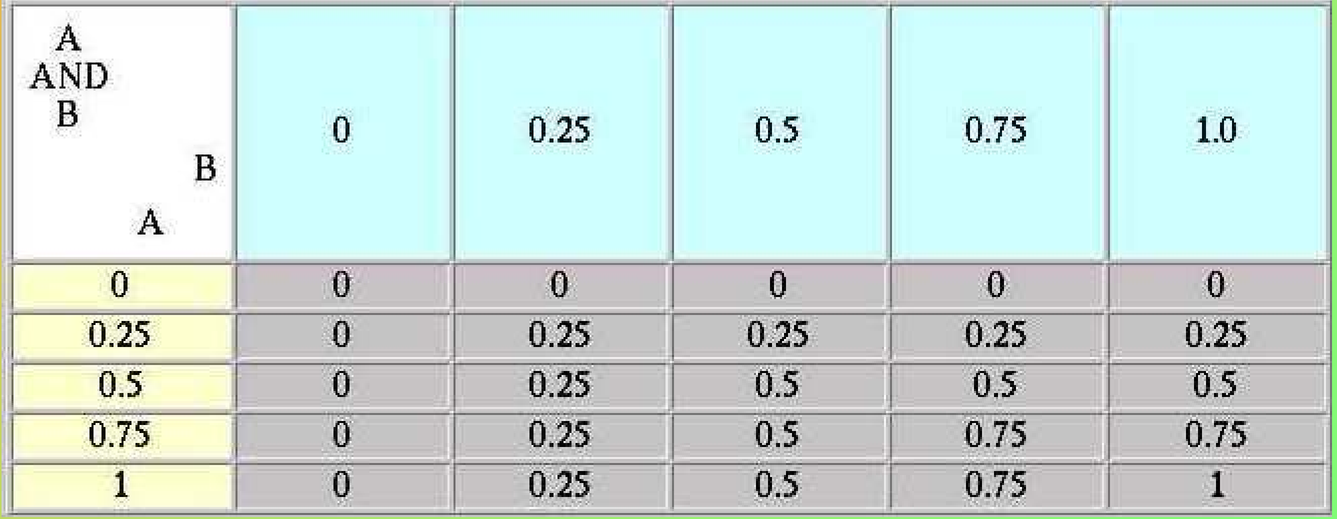
This technique computes the fuzzy “and” by multiplying the two membership values.

Fuzzy “and”

Both techniques have the following two properties:

T(0,0) = T(a,0) = T(0,a) = 0
T(a,1) = T(1,a) = a

One of the nice things about both definitions is that they also can be used to compute the Boolean “and”. The **fuzzy “and”** is an extension of the **Boolean “and”** to numbers that are not just 0 or 1, but between 0 and 1.



A AND B B A	0	0.25	0.5	0.75	1.0
0	0	0	0	0	0
0.25	0	0.25	0.25	0.25	0.25
0.5	0	0.25	0.5	0.5	0.5
0.75	0	0.25	0.5	0.75	0.75
1	0	0.25	0.5	0.75	1

menu

Fuzzy “or”

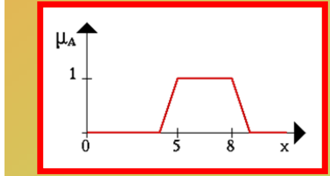
Fuzzy “or”

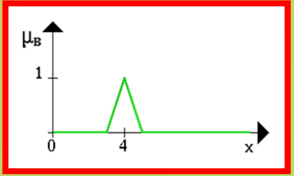
The fuzzy “or” is written as:

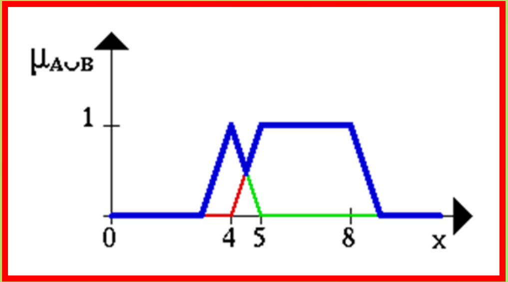
$$u_{A \cup B} = T(u_A(x), u_B(x))$$

where μ_A is read as “the membership in class A” and μ_B is read as “the membership in class B”.

Example:





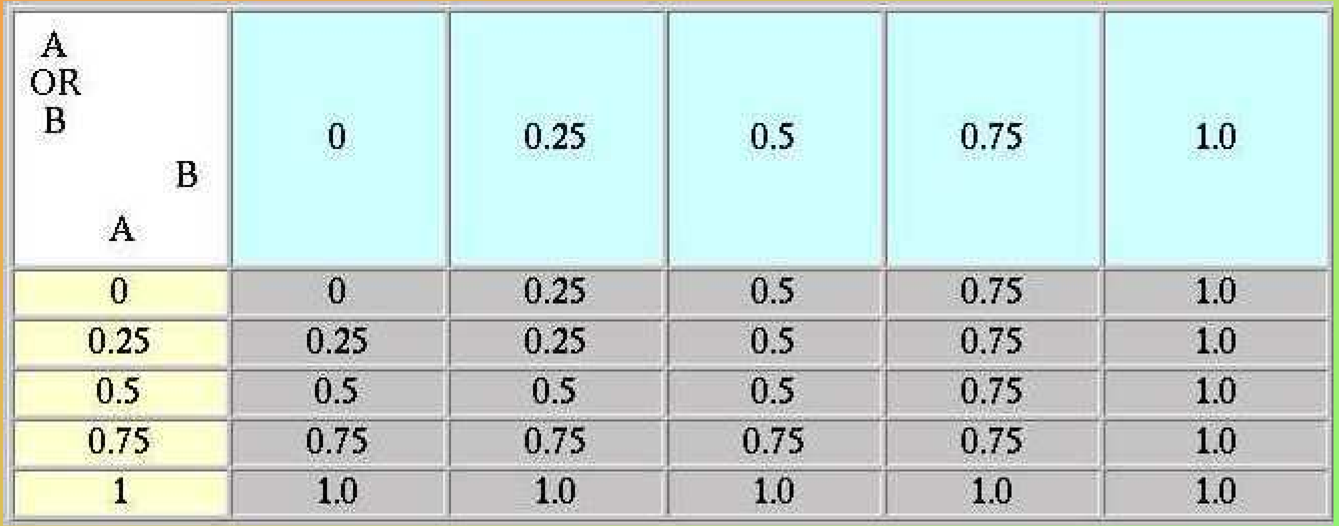


Union of **A** and **B**

menu

Fuzzy “or”

The fuzzy “or” is an extension of the Boolean “or” to numbers that are not just 0 or 1, but between 0 and 1.



A OR B B A	0	0.25	0.5	0.75	1.0
0	0	0.25	0.5	0.75	1.0
0.25	0.25	0.25	0.5	0.75	1.0
0.5	0.5	0.5	0.5	0.75	1.0
0.75	0.75	0.75	0.75	0.75	1.0
1	1.0	1.0	1.0	1.0	1.0

menu

Fuzzy "or"

There are many ways to compute "**or**". The two most common are:

$$\sigma(x, y) = \max(x, y)$$

Zadeh

$$\max(\mu_A\ (x), \mu_B\ (x))$$

it simply computes the "or" by taking the **maximum** of the two (or more) membership values. This is the most common definition of the fuzzy "or".

$$\sigma(x, y) = x + y - xy$$

Product

$$(\mu_A\ (x) + \mu_B\ (x)) - (\mu_A\ (x) * \mu_B\ (x))$$

This technique uses the difference between the sum of the two (or more) membership values and the product of their membership values.

Similar to the fuzzy "and", both definitions of the fuzzy "or" also can be used to compute the Boolean "or".

menu

Fuzzy “or”

Other ways to compute Fuzzy “**or**”:

$\sigma(x, y) = \min(1, x + y)$

Lukasiewicz Disjunction

min(1, *μA* (x) + *μB* (x))

$\sigma(x, y) = \frac{x + y - 2xy}{1 - xy}$

Hamacher Disjunction

$$\frac{\mu_A(x) + \mu_B(x) - 2\mu_A(x)\mu_B(x)}{1 - \mu_A(x)\mu_B(x)}$$

Fuzzy "**or**"

Other ways to compute Fuzzy "**or**":

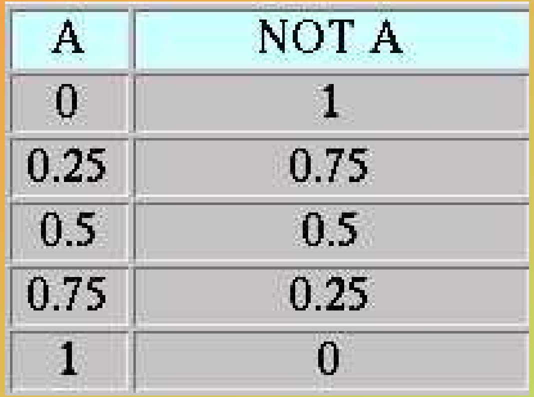
$$\sigma(x, y) = \frac{x + y}{1 + xy}$$

Einstein Disjunction

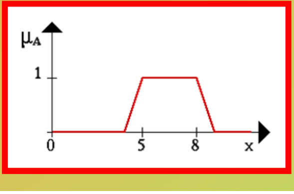
$$\frac{\mu_A(x) + \mu_B(x)}{1 + \mu_A(x)\mu_B(x)}$$

Fuzzy “not”

NOT (A) = 1 - A

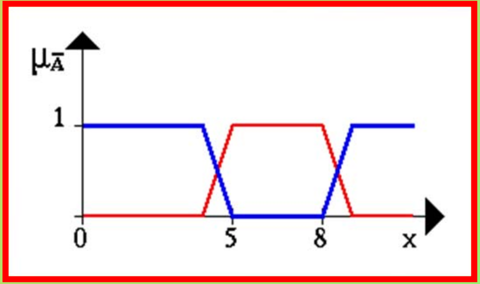


A	NOT A
0	1
0.25	0.75
0.5	0.5
0.75	0.25
1	0



Fuzzy set **A**

Example:



Negation of **A**

menu

Fuzzy Set operations

Fuzzy logic is a superset of conventional (Boolean) logic

All other operations on classical sets also hold for fuzzy sets, except for the excluded middle laws.

$$A \cup \overline{A} \neq X$$

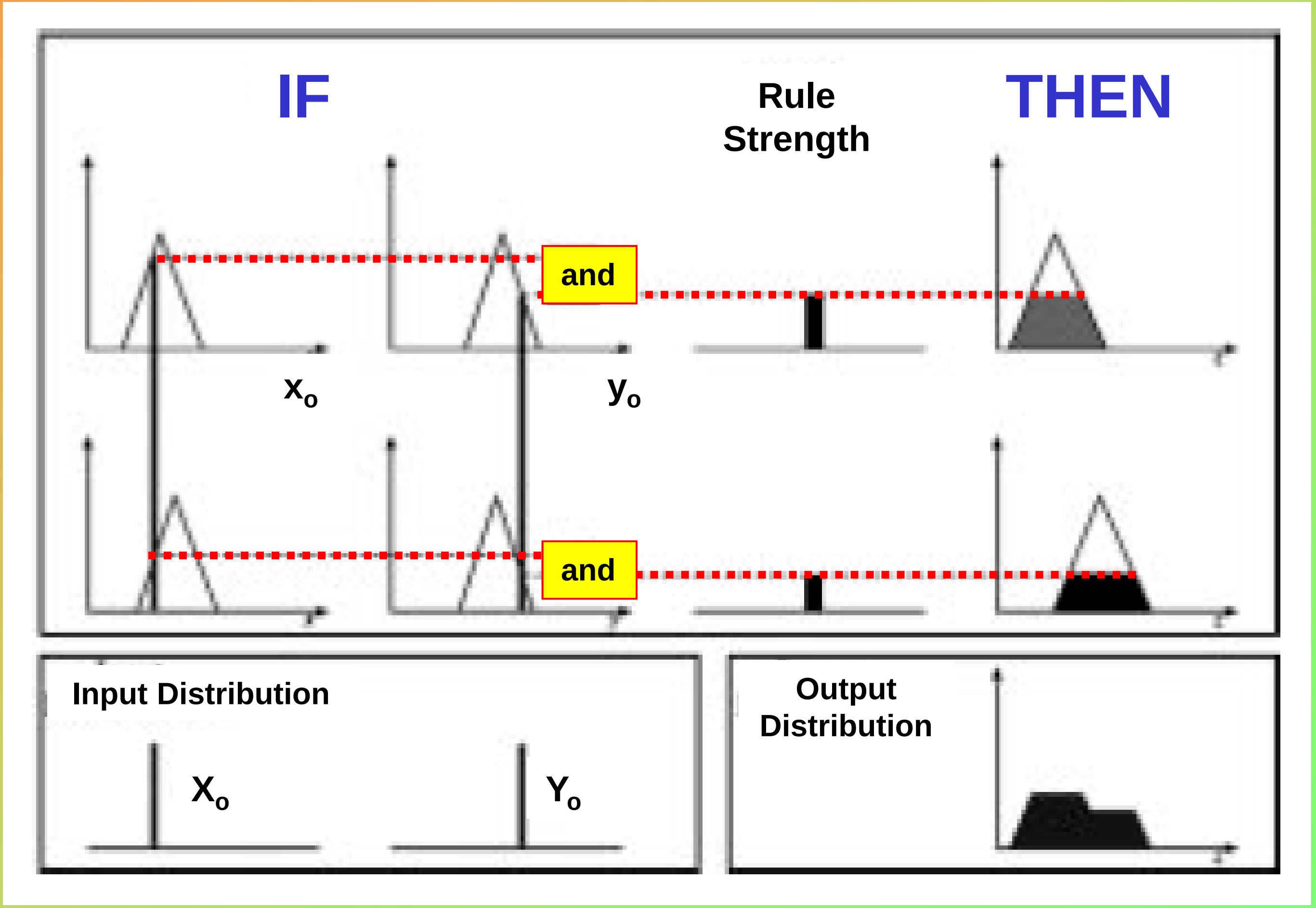
$$A \cap \overline{A} \neq 0$$

menu

Consequence

The consequence of a fuzzy rule is computed using two steps:

1 Computing the rule strength by combining the **fuzzified** inputs using the **fuzzy combination** process



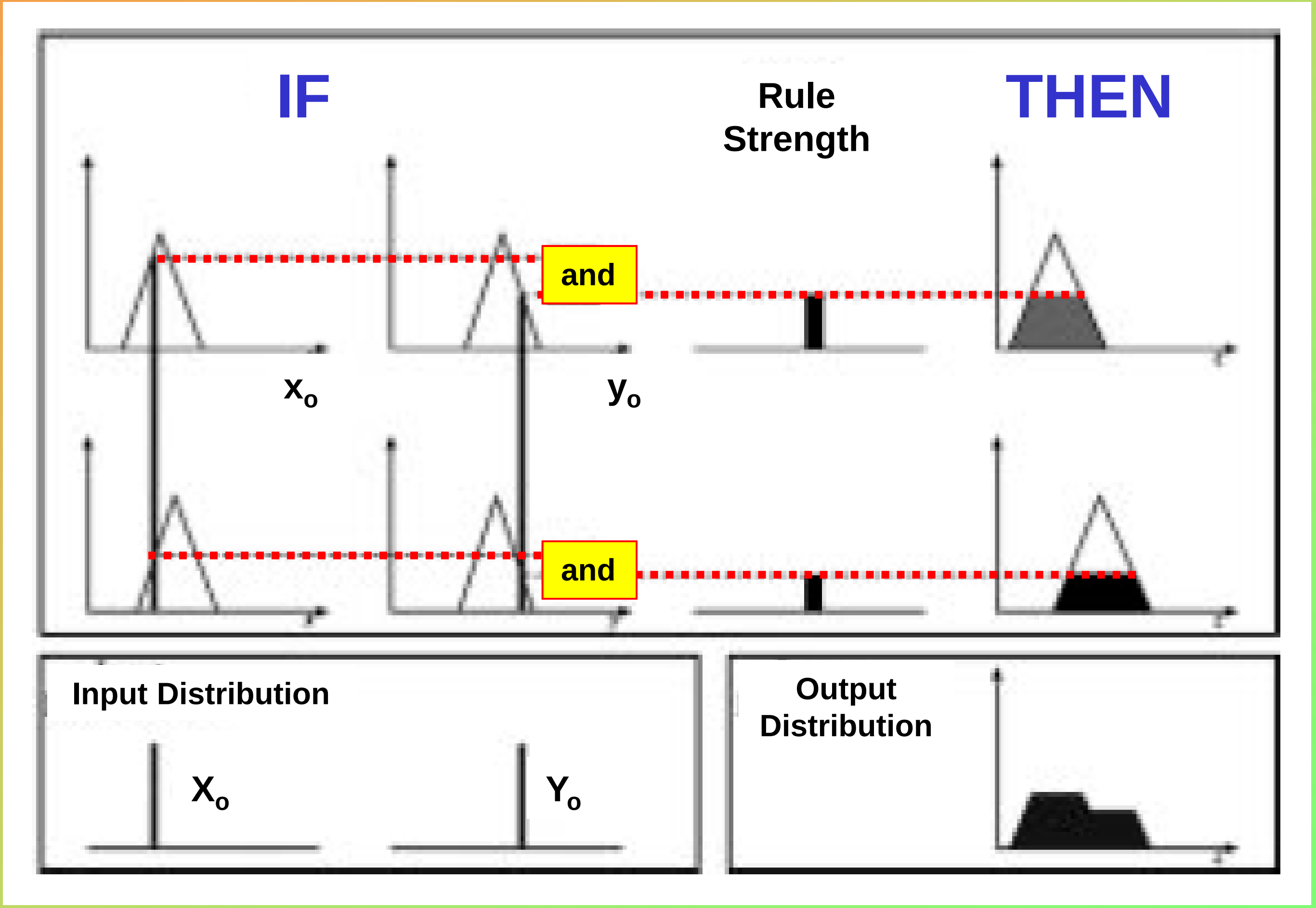
In this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

menu

Consequence

2

Clipping the output membership function at the rule strength.

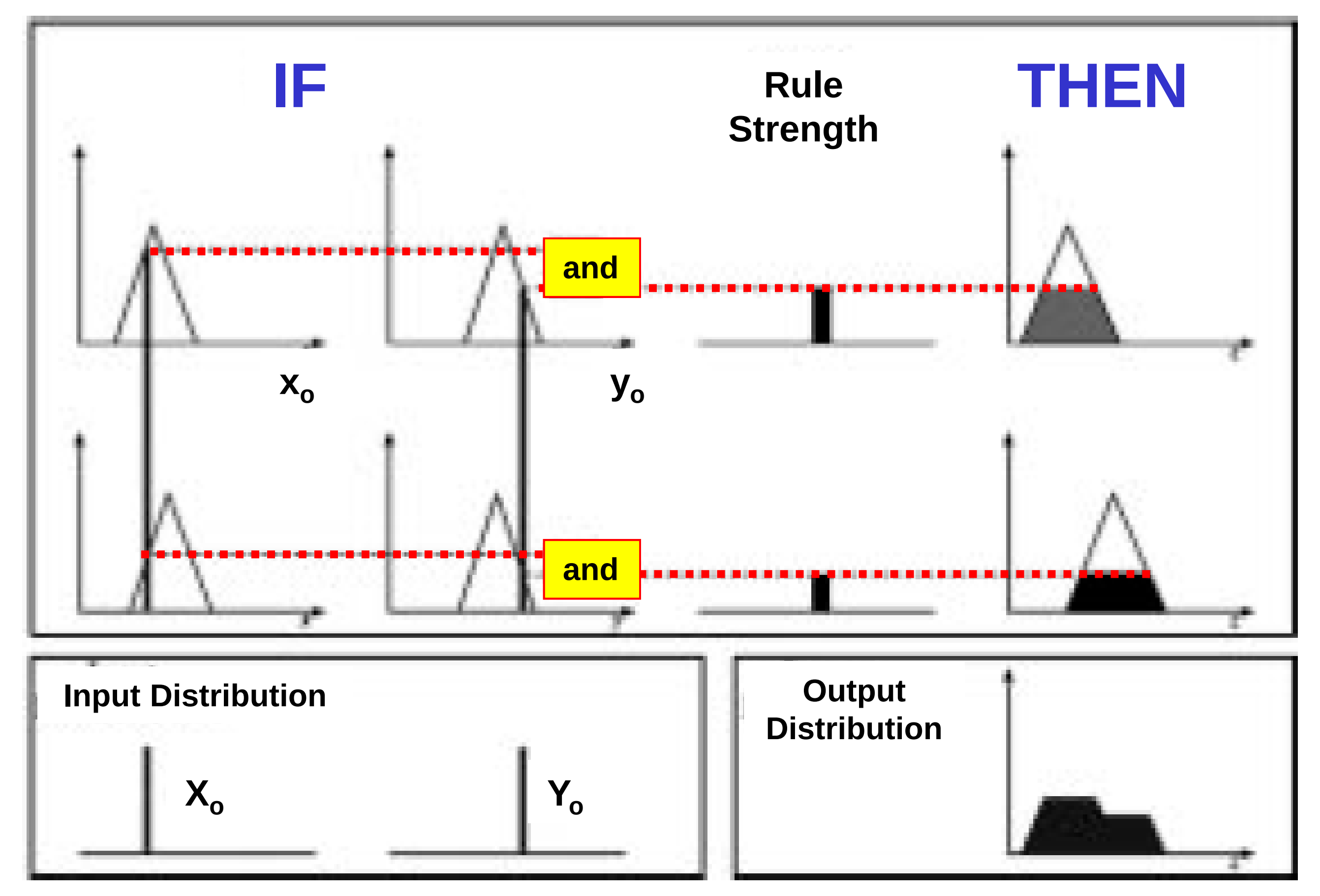


In this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

menu

Consequence

The outputs of all of the fuzzy rules must now be combined to obtain one fuzzy output distribution. This is usually, but not always, done by using the fuzzy “or”. The figure below shows an example of this.



The output membership functions on the right hand side of the figure are combined using the fuzzy “**or**” to obtain the output distribution shown on the lower right corner of the figure.

5 Defuzzification of Output Distribution

In many instances, it is desired to come up with a single crisp output from a FIS. For example, if one was trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIS would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained in a process known as defuzzification.

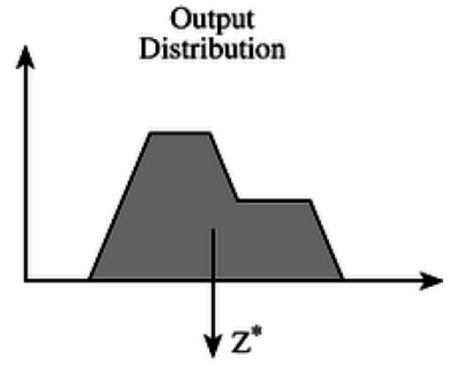
There are two common techniques for **defuzzifying**:

a) Center of mass

Center of mass - This technique takes the output distribution found in the previous slide and finds its center of mass to come up with one crisp number. This is computed as follows:

$$z = \frac{\sum_{j=1}^{q} Z_{\text{out}j} u_c(Z_j)}{\sum_{j=1}^{q} u_c(Z_j)}$$

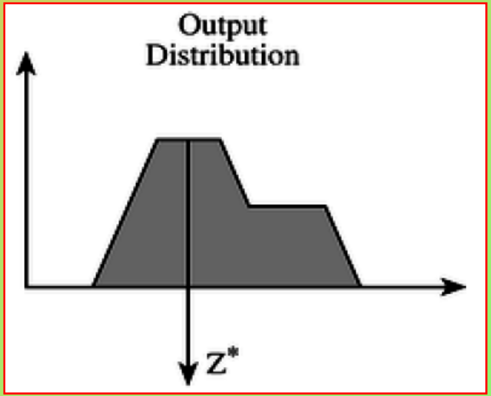
where z is the center of mass and μ_c is the membership in class c at value Z_j. An example outcome of this computation is shown in the figure at the right.



Defuzzification of Output Distribution

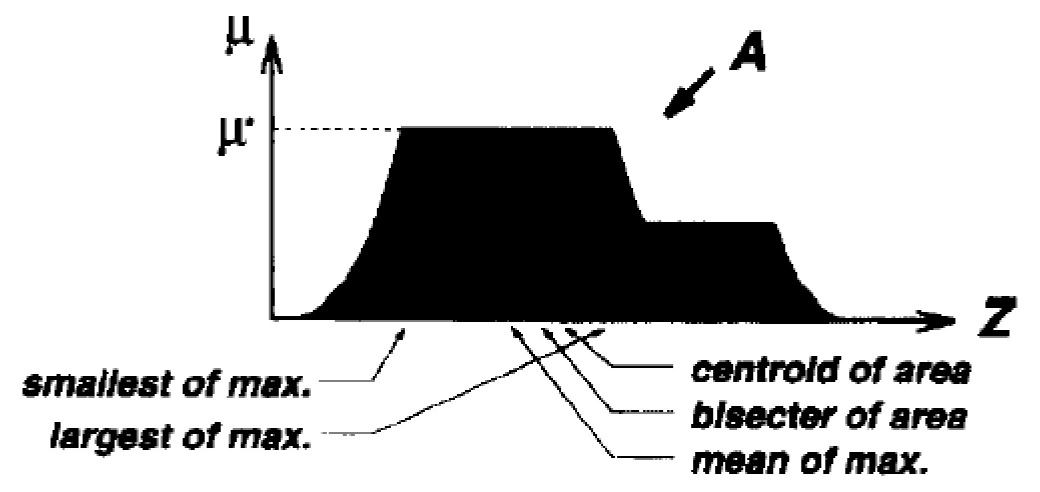
b) **Mean of maximum** - This technique takes the output distribution found in the previous section and finds its mean of maxima to come up with one crisp number. This is computed as follows:

$$z = \sum_{j=1}^{l} \frac{z_j}{l}$$



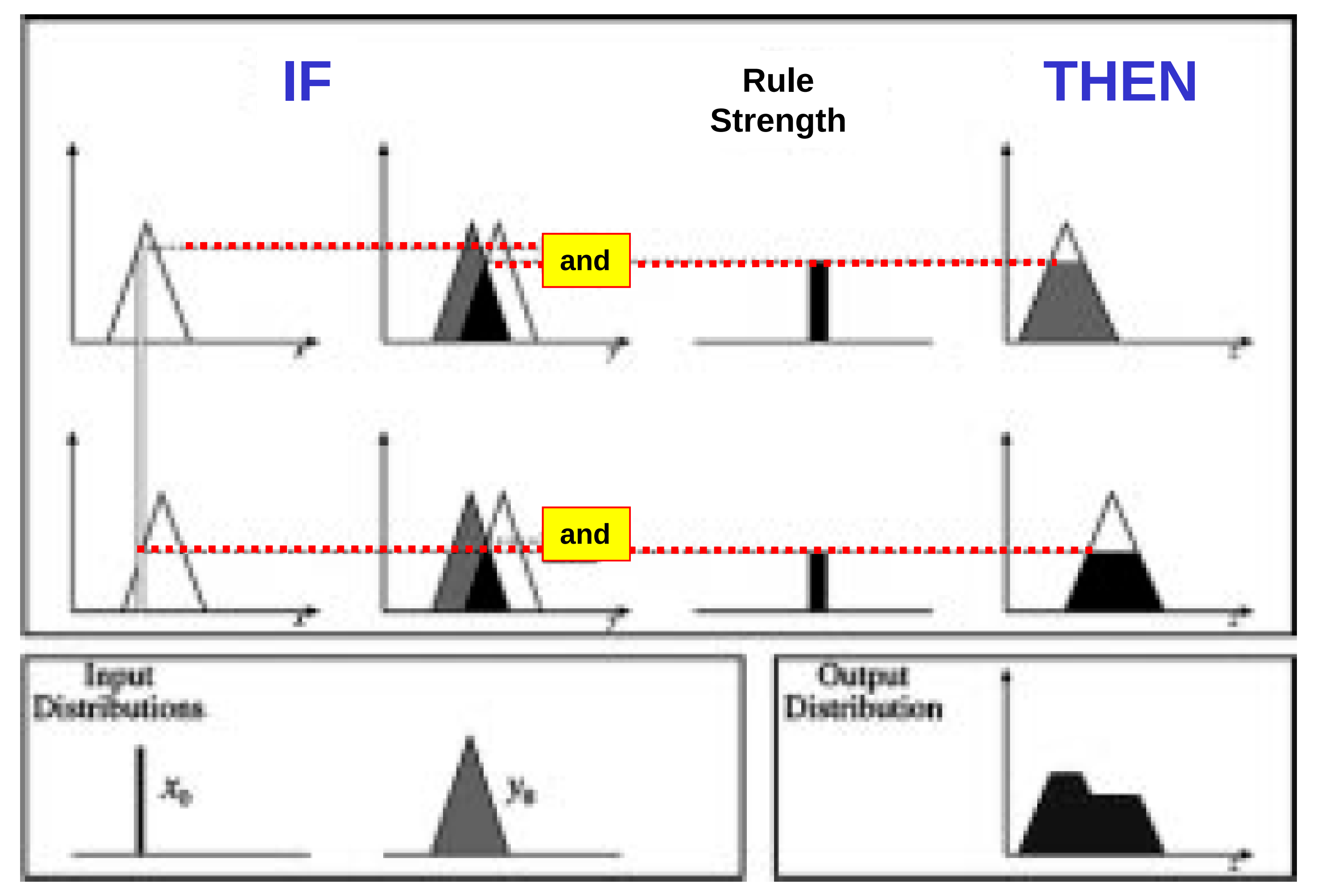
where **z** is the mean of maximum, **Z_j** is the point at which the membership function is maximum, and l is the number of times the output distribution reaches the maximum level. An example outcome of this computation is shown the figure at the right.

Defuzzification of Output Distribution



Mamdani FIS with a Fuzzy Input

A two Input, two rule Mamdani FIS with a fuzzy input



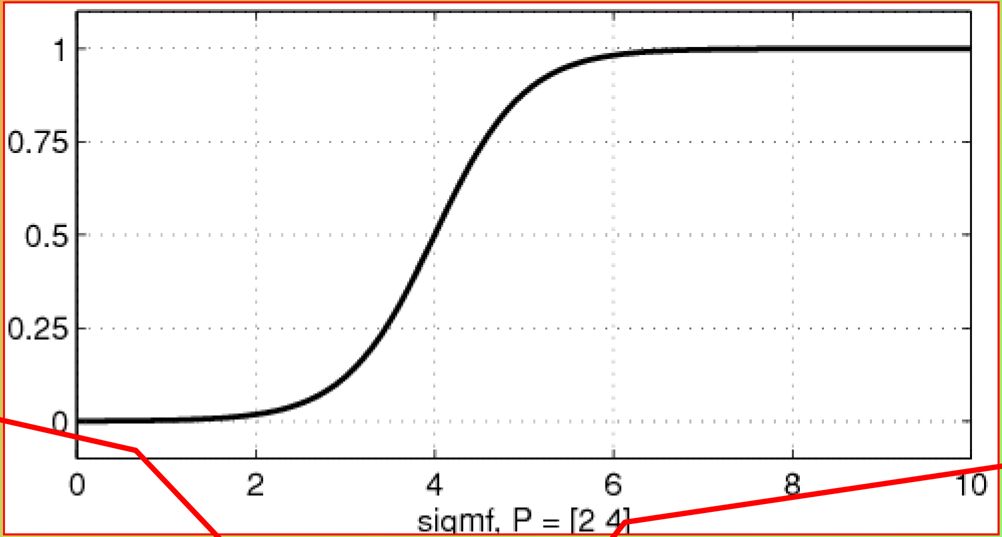
shows a modification of the Mamdani FIS where the input y0 is fuzzy, not crisp.
This can be used to model inaccuracies in the measurement.
For example, we may be measuring the output of a pressure sensor. Even with the exact same pressure applied, the sensor is measured to have slightly different voltages.
The fuzzy input membership function models this uncertainty.

The input fuzzy function is combined with the rule input membership function by using the fuzzy “**and**”

Membership Functions

The Sigmoidal function

sigmf(x,[a c]), as given in the following equation by f(x,a,c) is a mapping on a vector x, and depends on two parameters a and c.



Slope

crossover point

$$f\left(x,a,c\right)=\frac{1}{1+e^{-a\left(x-c\right)}}$$

menu

Membership Functions

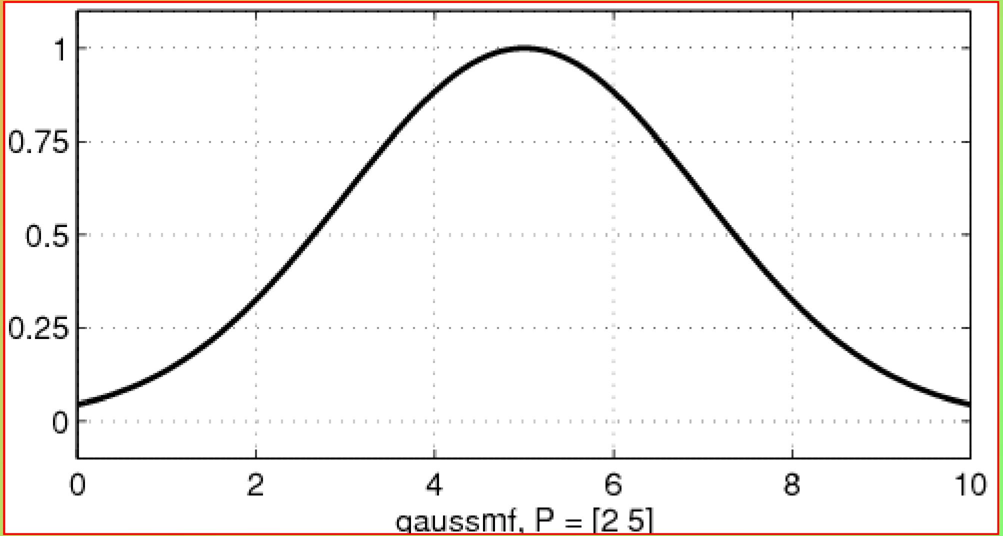
The Gaussian function

The symmetric Gaussian function depends on two parameters σ and c as given by

$$f(x;\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}}$$

center

width



menu

Membership Functions

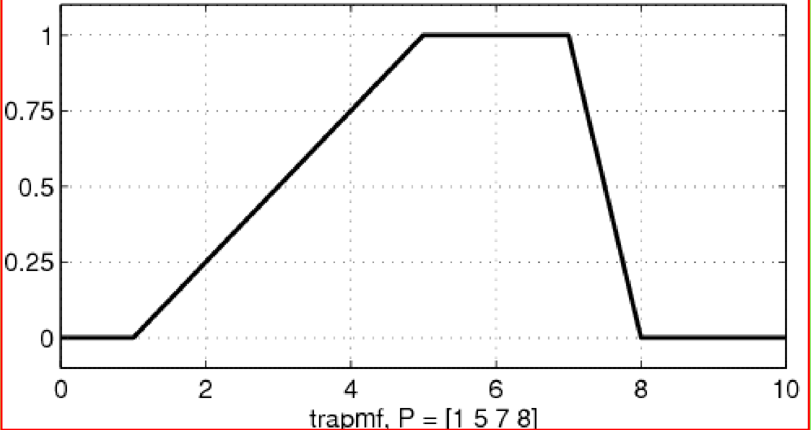
The Trapezoidal function

The trapezoidal curve is a function of a vector, x, and depends on four scalar parameters a, b, c, and d, as given by

$$f(x;a,b,c,d)=\begin{Bmatrix} 0, & x \le a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{Bmatrix}$$

or

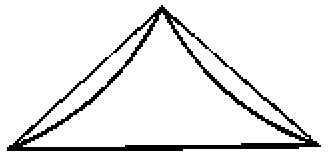
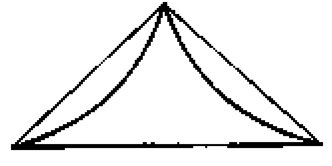
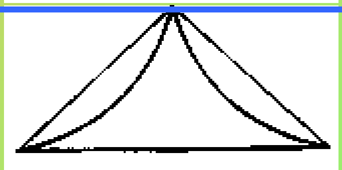
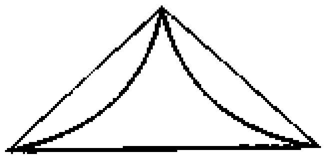
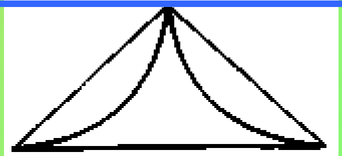
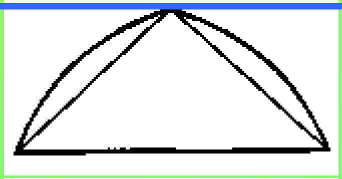
$$f(x;a,b,c,d)=\max\left(\min\left(\frac{x-a}{b-a},1,\frac{d-x}{d-c}\right),o\right)$$



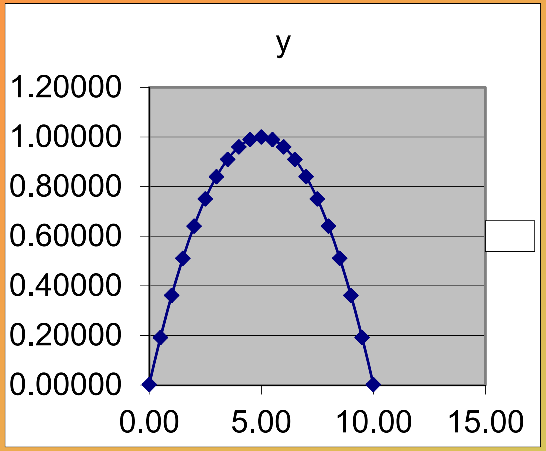
The parameters a and d locate the "feet" of the trapezoid and the parameters b and c locate the "shoulders."

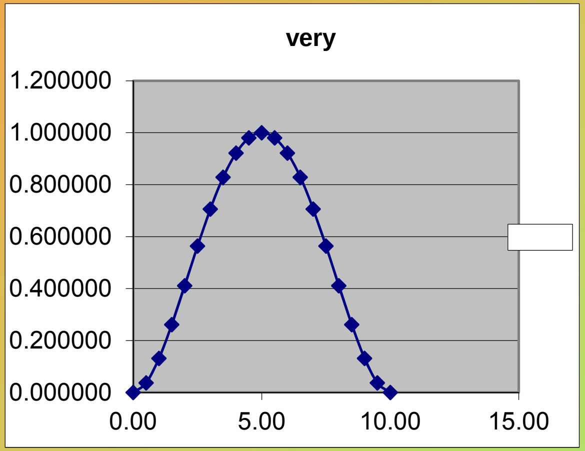
Hedges

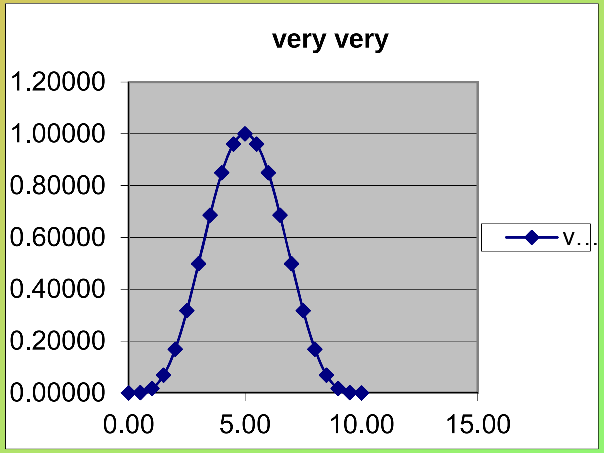
http://blog.peltarion.com/2006/10/25/fuzzy-math-part-1-the-theory/

Hedge	Operator	Effect
A little	$\mu_A(x)^{1.3}$	
Slightly	$\mu_A(x)^{1.7}$	
Very	$\mu_A(x)^2$	
Extremely	$\mu_A(x)^3$	
Very very	$\mu_A(x)^4$	
Somewhat	$\mu_A(x)^{\frac{1}{2}}$	
Indeed	$2\mu_A(x)^2 \quad \text{if } 0 \le \mu_A(x) \le 0.5$ $1-2(1-\mu_A(x))^2 \quad \text{if } 0.5 < \mu_A(x) \le 1$	

Samples







Sigma	3.54
c	5

Fuzzy Logic Applications

- **Robot Navigation**
  - Derivation of equations for motion
  - Variations of settings:
    - FAMM
    - Shape of membership functions
    - Fuzzy Output values settings
  - Start-up codes

menu

Fuzzy Logic Applications

- **Inverted Pendulum**
  - – Analyze the simulation (see implementation)
  - – Variations of settings:
    - (3x3, 5x5 FAMMS)
    - Shape of membership functions
    - Fuzzy Output values settings, time delay
  - – Utilisation of Euler's Method in the simulation (see pdf)

menu

References

- Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann
- Neural Network and Fuzzy Logic Applications in C/C++ (Wiley Professional Computing) by Stephen Welstead
- Fuzzy Logic with Engineering Applications by Timothy Ross
- Fuzzy Sets and Pattern Recognition by Benjamin Knapp

menu

The End.

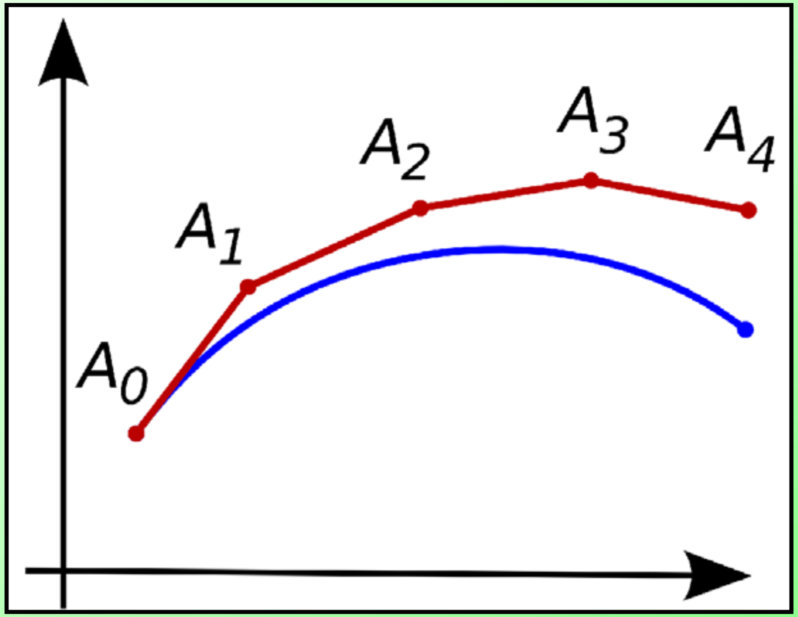
menu

This is NOT going to be part of the exam anymore.

OPTIONAL: Euler Method

Euler Method

- In mathematics and computational science, the **Euler method**, named after Leonhard Euler, is a numerical procedure for solving ordinary differential equations (ODEs) with a given initial value. It is the most basic kind of explicit method for numerical integration for ordinary differential equations.

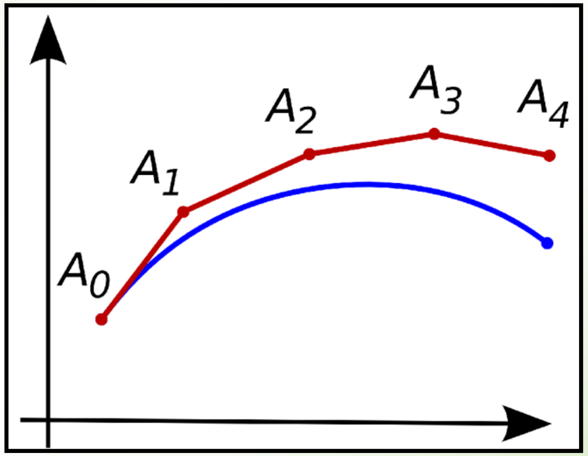


Source: http://www.answers.com/topic/euler-method-1?cat=technology

menu

Euler Method

Informal geometrical description



- Consider the problem of calculating the shape of an unknown curve which starts at a given point and satisfies a given differential equation. Here, a differential equation can be thought of as a formula by which the slope of the tangent line to the curve can be computed at any point on the curve, once the position of that point has been calculated.

- The idea is that while the curve is initially unknown, its starting point, which we denote by *A*0, is known (see the picture on top right). Then, from the differential equation, the slope to the curve at *A*0 can be computed, and so, the tangent line.

- Take a small step along that tangent line up to a point *A*1. If we pretend that *A*1 is still on the curve, the same reasoning as for the point *A*0 above can be used. After several steps, a polygonal curve *A*0*A*1*A*2*A*3... is computed. In general, this curve does not diverge too far from the original unknown curve, and the error between the two curves can be made small if the step size is small enough and the interval of computation is finite (although things are more complicated for stiff equations, as discussed below).

Source: http://www.answers.com/topic/euler-method-1?cat=technology

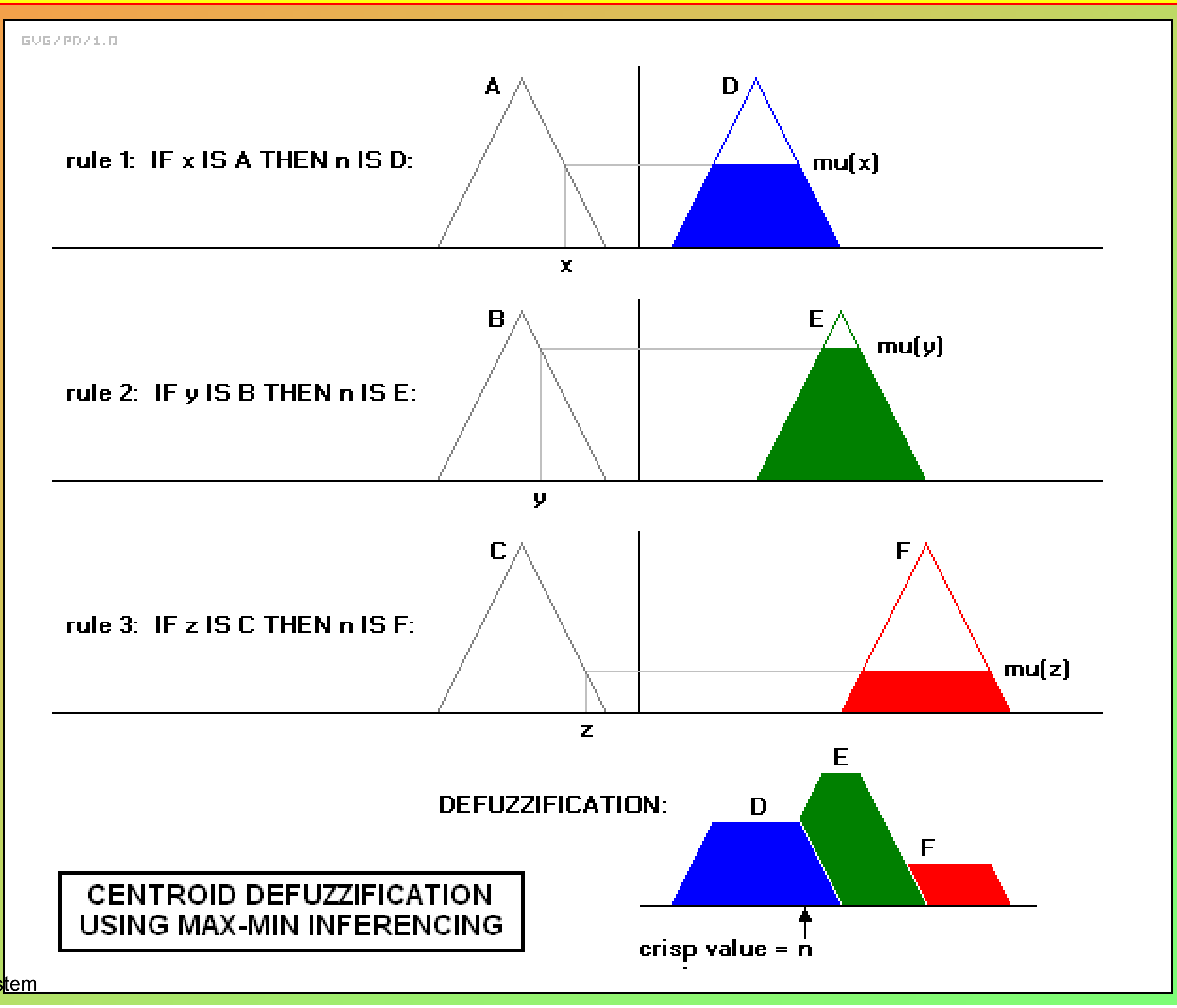
menu

This is NOT going to be part of the exam anymore.

OPTIONAL: More on Mamdani FIS

Mamdani FIS

Max-min inferencing and centroid defuzzification for a system with input variables "x", "y", and "z" and an output variable "n". Note that "mu" is standard fuzzy-logic nomenclature for "truth value":



http://en.wikipedia.org/wiki/Fuzzy_system

menu

Sample: Mamdani

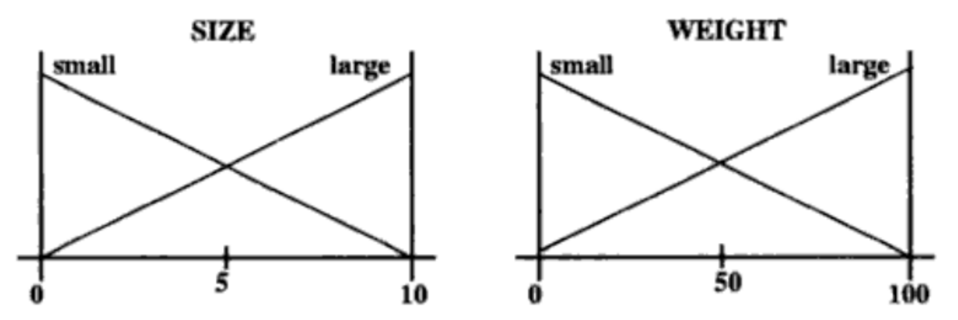
Classical Inference Engine: max-min-CG

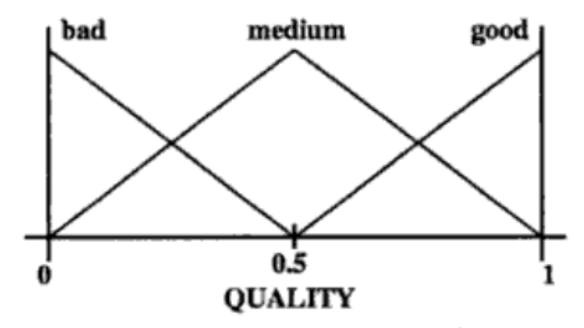
Consider a problem with 2 input variables, **size** and **weight**, and one output variable, **quality,** with the following **lingustic term sets** associated**:**

$$D_{size} = \{small, large\} \qquad D_{weight} = \{small, large\}$$

$$D_{quality} = \{bad, medium, good\}$$

The **semantics** of these linguistic terms are defined **by triangular-shaped membership functions:**





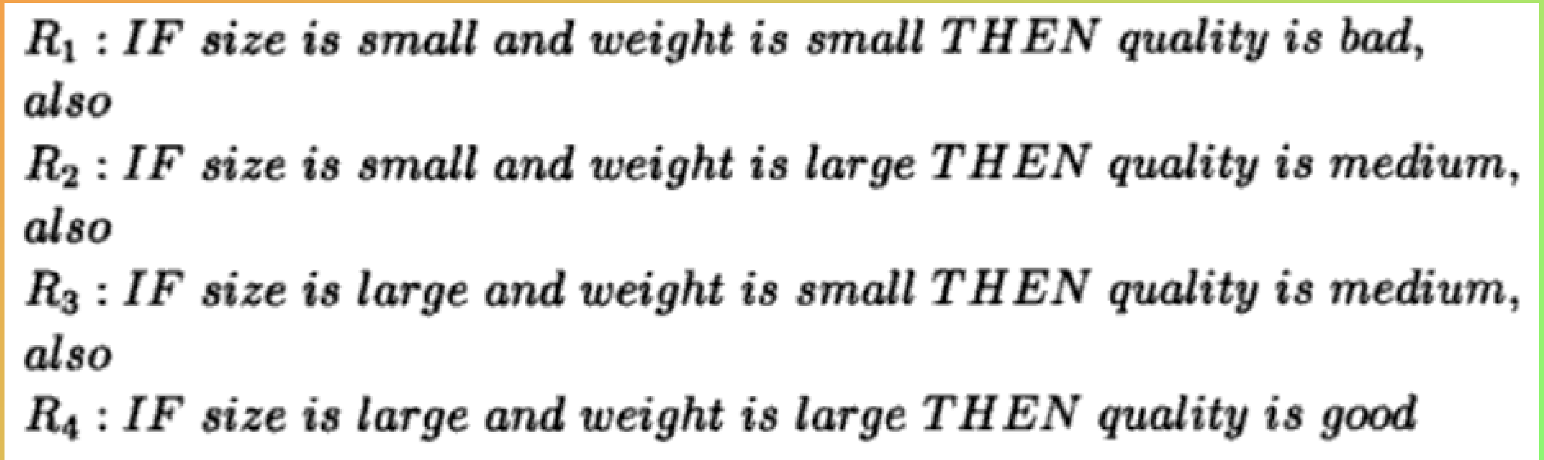
Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

Classical Inference Engine: max-min-CG

Rule Base (RB):



R_1 : *IF size is small and weight is small THEN quality is bad,*
also
R_2 : *IF size is small and weight is large THEN quality is medium,*
also
R_3 : *IF size is large and weight is small THEN quality is medium,*
also
R_4 : *IF size is large and weight is large THEN quality is good*

Let us consider a sample input to the system, **X_o = {2, 25} = {size, weight}**

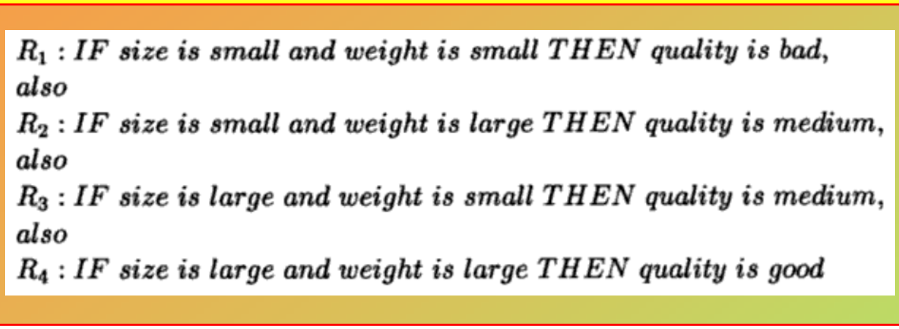
Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

Classical Inference Engine: max-min-CG

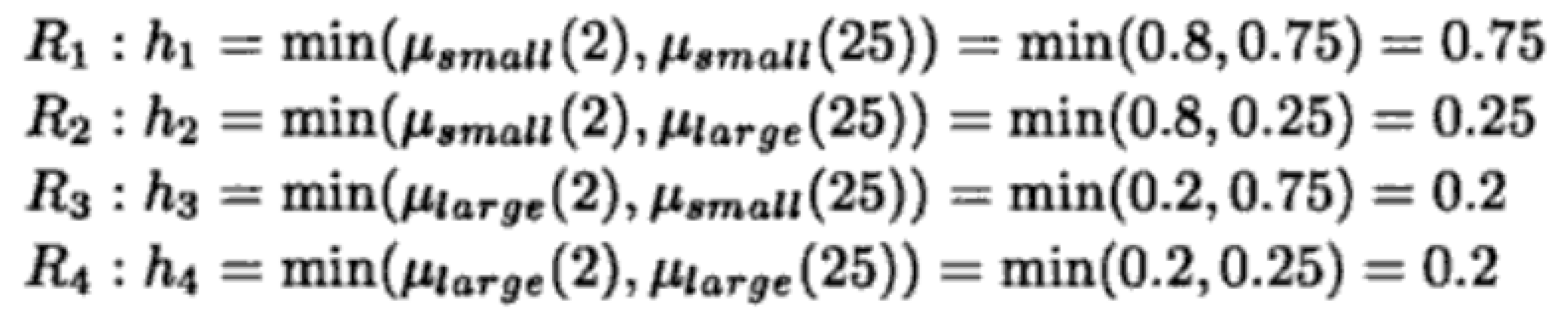
Rule Base:



R_1 : *IF size is small and weight is small THEN quality is bad,*
also
R_2 : *IF size is small and weight is large THEN quality is medium,*
also
R_3 : *IF size is large and weight is small THEN quality is medium,*
also
R_4 : *IF size is large and weight is large THEN quality is good*

Let us consider a sample input to the system, **X_o = {2, 25}.**
This input is matched against the rule antecedents in order to determine the rule-firing strength *hi* of each rule R*i* in the Rule Base.

Using a minimum T-norm as conjunctive, the following results are obtained:



$$R_1 : h_1 = \min(\mu_{small}(2), \mu_{small}(25)) = \min(0.8, 0.75) = 0.75$$
$$R_2 : h_2 = \min(\mu_{small}(2), \mu_{large}(25)) = \min(0.8, 0.25) = 0.25$$
$$R_3 : h_3 = \min(\mu_{large}(2), \mu_{small}(25)) = \min(0.2, 0.75) = 0.2$$
$$R_4 : h_4 = \min(\mu_{large}(2), \mu_{large}(25)) = \min(0.2, 0.25) = 0.2$$

Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

Classical Inference Engine: max-min-CG

Rule-firing strength h*i* of each rule R*i* in the Rule Base:

$$
\begin{aligned}
R_1 &: h_1 = \min(\mu_{small}(2), \mu_{small}(25)) = \min(0.8, 0.75) = 0.75\\
R_2 &: h_2 = \min(\mu_{small}(2), \mu_{large}(25)) = \min(0.8, 0.25) = 0.25\\
R_3 &: h_3 = \min(\mu_{large}(2), \mu_{small}(25)) = \min(0.2, 0.75) = 0.2\\
R_4 &: h_4 = \min(\mu_{large}(2), \mu_{large}(25)) = \min(0.2, 0.25) = 0.2
\end{aligned}
$$

The inference system applies the **compositional rule of inference** to obtain the **inferred fuzzy sets B*i*'**

$$
\begin{aligned}
R_1 &: \mu_{B_1'}(y) = \min(h_1, \mu_{B_1}(y)) = \min(0.75, \mu_{bad}(y))\\
R_2 &: \mu_{B_2'}(y) = \min(h_2, \mu_{B_2}(y)) = \min(0.25, \mu_{medium}(y))\\
R_3 &: \mu_{B_3'}(y) = \min(h_3, \mu_{B_3}(y)) = \min(0.2, \mu_{medium}(y))\\
R_4 &: \mu_{B_4'}(y) = \min(h_4, \mu_{B_4}(y)) = \min(0.2, \mu_{good}(y))
\end{aligned}
$$

Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

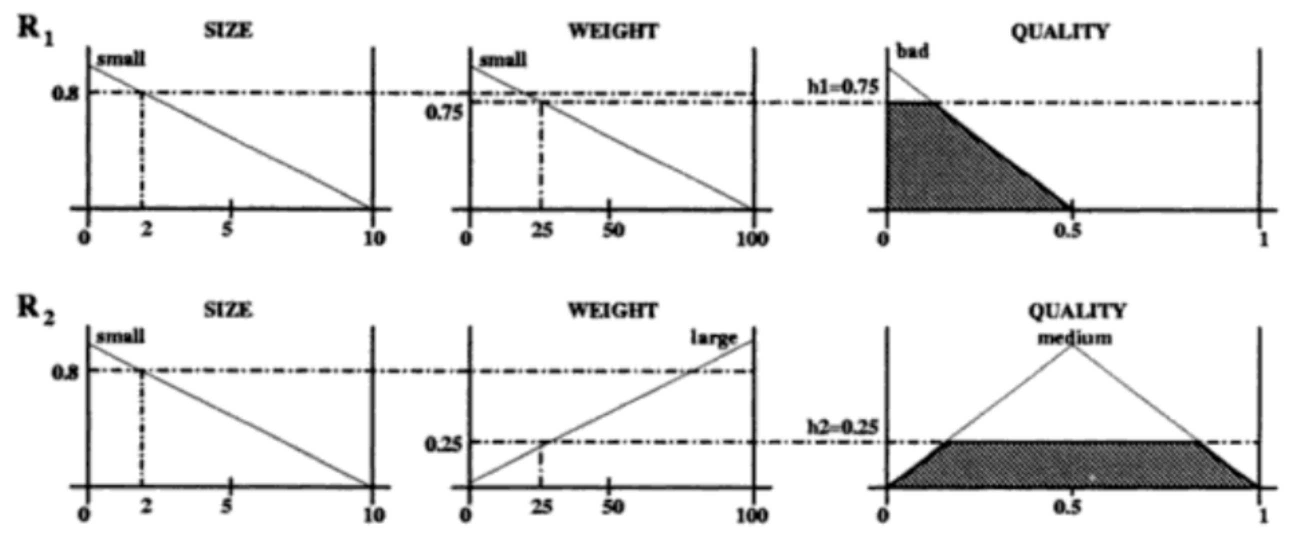
Rule-firing strength h*i* of each rule R*i* in the Rule Base:

$$R_1 : h_1 = \min(\mu_{small}(2), \mu_{small}(25)) = \min(0.8, 0.75) = 0.75$$
$$R_2 : h_2 = \min(\mu_{small}(2), \mu_{large}(25)) = \min(0.8, 0.25) = 0.25$$
$$R_3 : h_3 = \min(\mu_{large}(2), \mu_{small}(25)) = \min(0.2, 0.75) = 0.2$$
$$R_4 : h_4 = \min(\mu_{large}(2), \mu_{large}(25)) = \min(0.2, 0.25) = 0.2$$

Application of **compositional rule of inference** to obtain the **inferred fuzzy sets B*i*'**

$$R_1 : \mu_{B_1'}(y) = \min(h_1, \mu_{B_1}(y)) = \min(0.75, \mu_{bad}(y))$$
$$R_2 : \mu_{B_2'}(y) = \min(h_2, \mu_{B_2}(y)) = \min(0.25, \mu_{medium}(y))$$
$$R_3 : \mu_{B_3'}(y) = \min(h_3, \mu_{B_3}(y)) = \min(0.2, \mu_{medium}(y))$$
$$R_4 : \mu_{B_4'}(y) = \min(h_4, \mu_{B_4}(y)) = \min(0.2, \mu_{good}(y))$$

Graphical representation:



menu

Sample: Mamdani

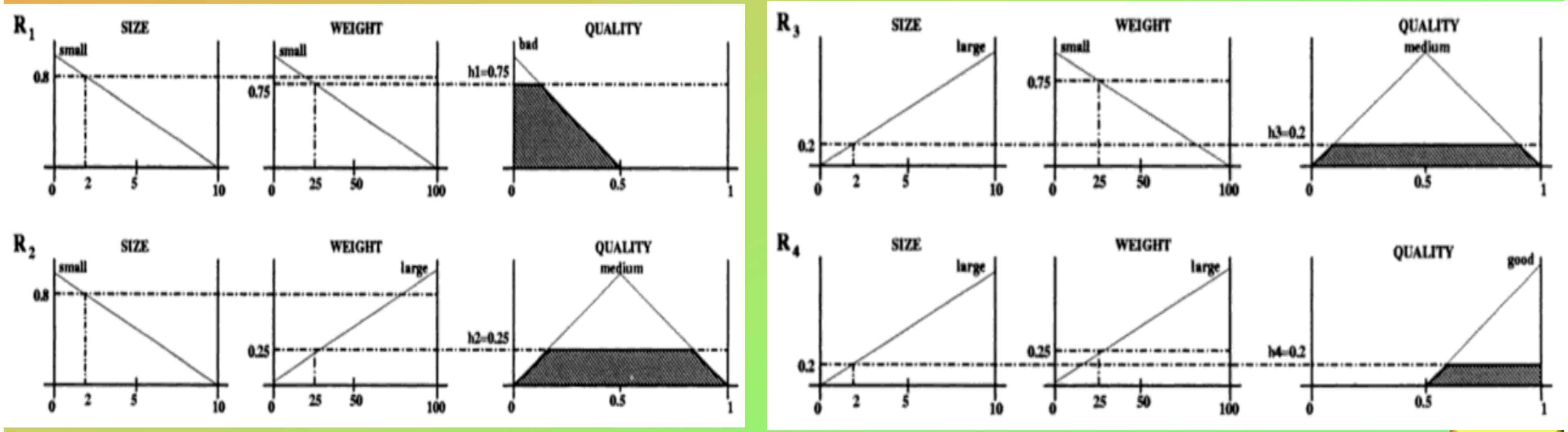
Rule-firing strength h*i* of each rule R*i* in the Rule Base:

$$R_1 : h_1 = \min(\mu_{small}(2), \mu_{small}(25)) = \min(0.8, 0.75) = 0.75$$
$$R_2 : h_2 = \min(\mu_{small}(2), \mu_{large}(25)) = \min(0.8, 0.25) = 0.25$$
$$R_3 : h_3 = \min(\mu_{large}(2), \mu_{small}(25)) = \min(0.2, 0.75) = 0.2$$
$$R_4 : h_4 = \min(\mu_{large}(2), \mu_{large}(25)) = \min(0.2, 0.25) = 0.2$$

Application of **compositional rule of inference** to obtain the **inferred fuzzy sets B*i*'**

$$R_1 : \mu_{B_1'}(y) = \min(h_1, \mu_{B_1}(y)) = \min(0.75, \mu_{bad}(y))$$
$$R_2 : \mu_{B_2'}(y) = \min(h_2, \mu_{B_2}(y)) = \min(0.25, \mu_{medium}(y))$$
$$R_3 : \mu_{B_3'}(y) = \min(h_3, \mu_{B_3}(y)) = \min(0.2, \mu_{medium}(y))$$
$$R_4 : \mu_{B_4'}(y) = \min(h_4, \mu_{B_4}(y)) = \min(0.2, \mu_{good}(y))$$

Graphical representation:

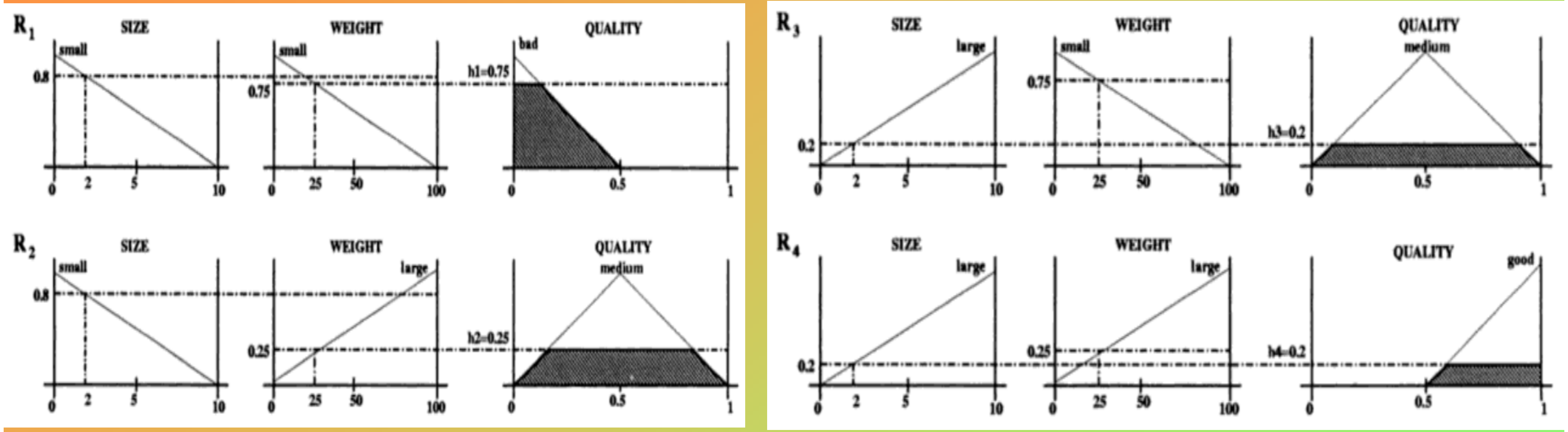


Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

Graphical representation:



Aggregation of the 4 individual output fuzzy sets by means of the **maximum t-conorm**

$$\mu_{B'}(y) = \max\left\{\mu_{B'_1}(y), \mu_{B'_2}(y), \mu_{B'_3}(y), \mu_{B'_4}(y)\right\}$$

The final output is calculated by defuzzification using **Centre of Gravity (CG)**

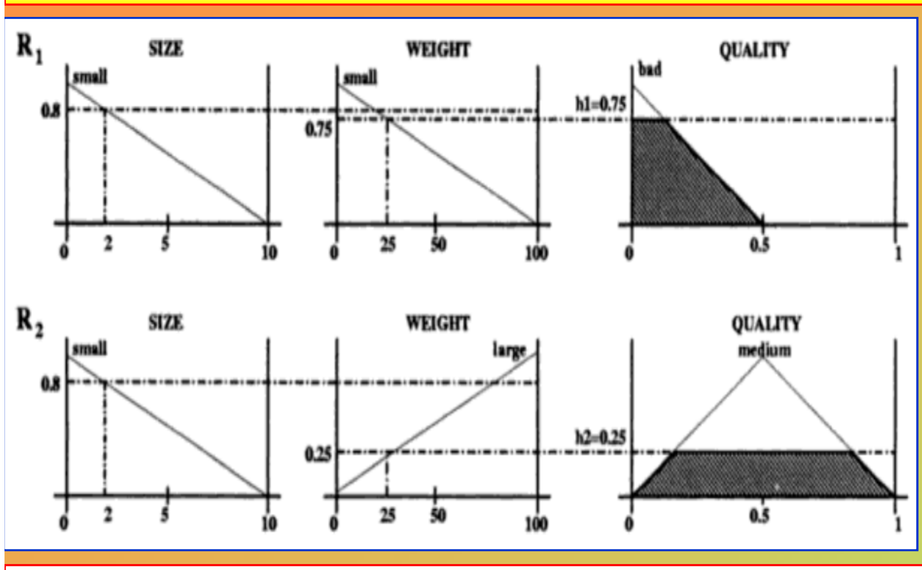
$$y_0 = \frac{\int_Y y \cdot \mu_{B'}(y)dy}{\int_Y \mu_{B'}(y)dy}$$

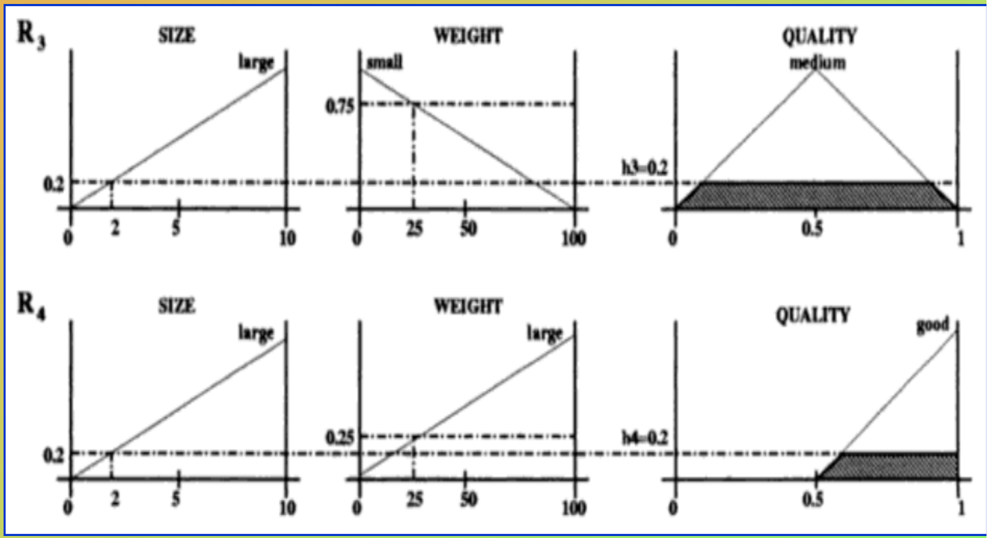
Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

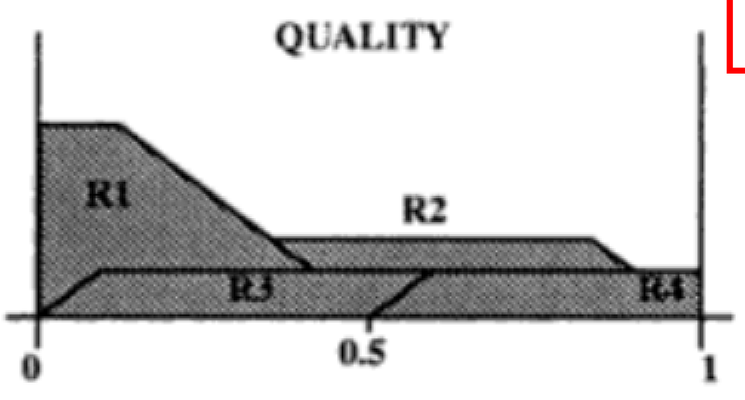
Graphical representation:

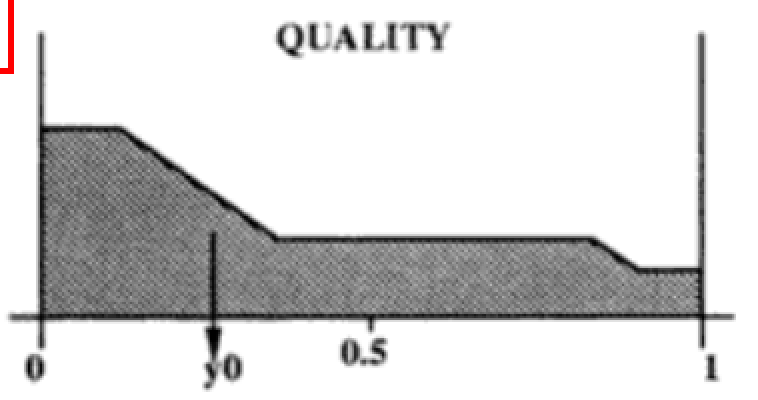




The final output is calculated by defuzzification using **Centre of Gravity (CG)**

$$y_0 = \frac{\int_Y y \cdot \mu_{B'}(y)dy}{\int_Y \mu_{B'}(y)dy}$$





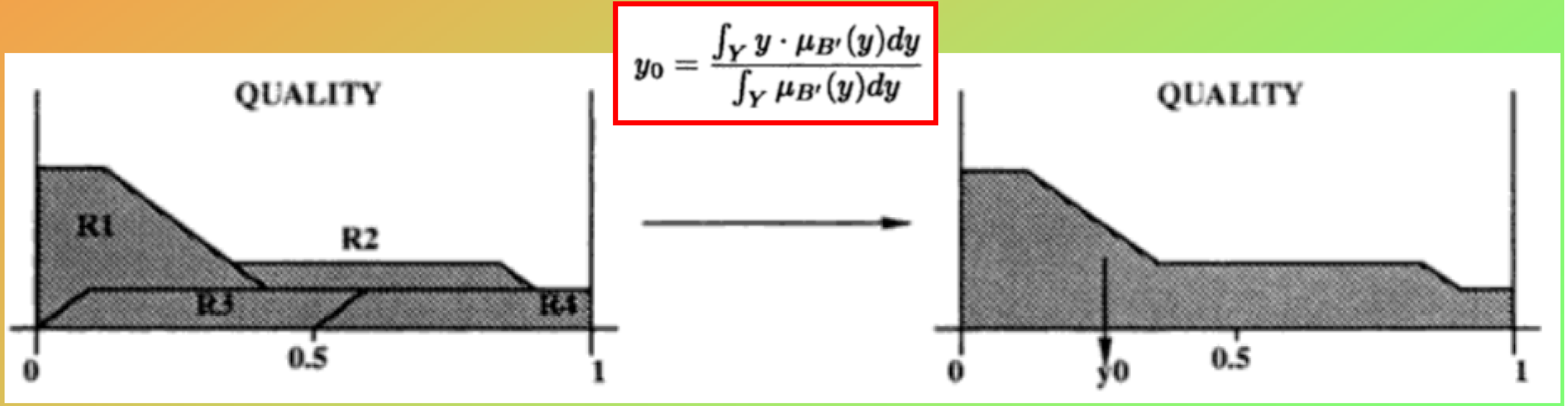
Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

menu

Sample: Mamdani

Mode A-FATI with Max - Centre of Gravity (CG) strategy

“First Aggregate, Then Infer”



$$y_0 = \frac{\int_Y y \cdot \mu_{B'}(y)dy}{\int_Y \mu_{B'}(y)dy}$$

Reference: **Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann**

The final crisp output is *Yo* = 0.3698

menu

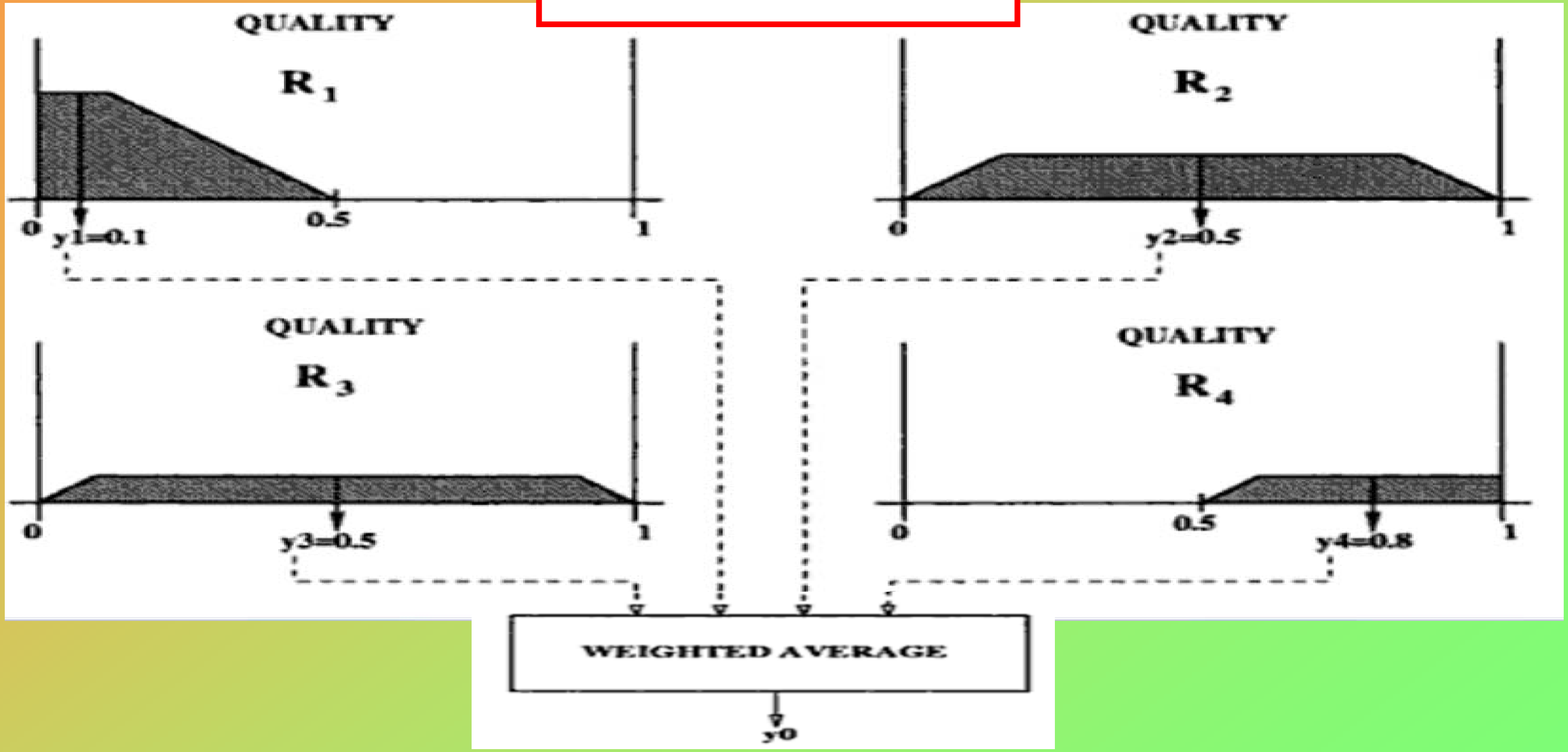
Sample: Mamdani

Reference: Genetic fuzzy systems by Oscar Cordón, Francisco Herrera, Frank Hoffmann

Alternatively, ("First Infer, Then Aggregate")

Mode B-FITA with Maximum Value weighted by the matching strategy

$$y_0 = \frac{\sum_{i=1}^{m} h_i \cdot MV_i}{\sum_{i=1}^{m} h_i}$$



$$y_0 = \frac{0.75 \cdot 0.1 + 0.25 \cdot 0.5 + 0.2 \cdot 0.5 + 0.2 \cdot 0.8}{1.4} = \frac{0.46}{1.4} = 0.3286$$

menu