Fuzzy Logic

Theory and Applications

n.h.reyes@massey.ac.nz

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
- Assignment #2

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

Motivation

Previously

Systematic exploration of alternatives

find a path or a plan of action

e.g. sequence of moves to solve a puzzle

Next set of problems

States finite and complete

e.g. arrangement of tiles in a puzzle

- States with continuous-valued and ill-defined inputs
- Dynamically-changing world

e.g. target is moving to the left and its speed is increasing fast

Moves deterministic and discrete

e.g. up, right, down, left

Continuous-valued actions
uncertainty in inputs

e.g. exact steering angle, speed, exact Force

back

menu

What is Fuzzy Logic?

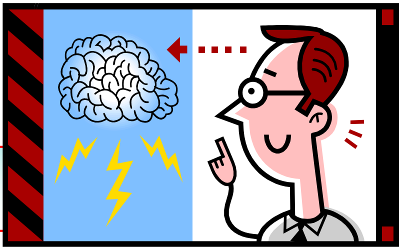
'Fuzzy' – a misnomer

n.h.reyes@massey.ac.nz

Fuzzy Logic

What is Fuzzy Logic?

A **computational paradigm that mimics 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 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" – a misnomer, has resulted in the mistaken suspicion that FL is somehow less exacting than traditional logic

e.g. *Why would you purchase a fuzzy-controlled auto-focusing camera?...*
Wouldn't it produce fuzzy images?

back

menu

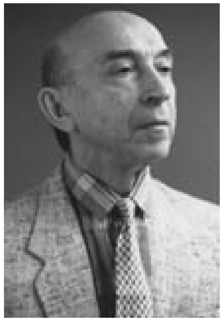
What is Fuzzy Logic?

History

n.h.reyes@massey.ac.nz

Fuzzy Logic

History of Fuzzy Logic



an American, mathematically oriented, computer scientist, electrical engineer of Iranian descent, born in Russia.

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.

This paper drew 90,000 Google Scholar citations (as of 2017).
It is the highest cited paper in the literature of Computer Science (Web of Science);

Source: August 30, 2004
(Computerworld)

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

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

Pioneering works

20 years later after its inception

- 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

back
menu

What is Fuzzy Logic?

Applications

n.h.reyes@massey.ac.nz

Fuzzy Logic

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

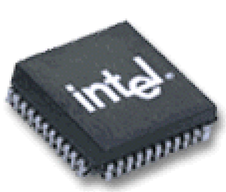
(*News excerpt from the 1990s*) **Fuzzy Logic** is one of the most talked-about technologies to hit the embedded control field in recent years. It has already transformed many product markets in Japan and Korea, and has begun to attract a widespread following In the United States. Industry watchers predict 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



MCS® 96

- Overview
- HSIO Family
- EPA Family
- Motor Control Family
- CAN Product Family (Express)

MCS® 296

- Overview
- Backgrounder
- Documentation

Motorola 68HC12 MCU

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

back

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

Segway Robotics Mobility Platform

- Algorithm: combination of Fuzzy Logic and Expert System
- Sensors: multiple gyroscopes and accelerometers
- Estimates the 3D position of the mobile robot
- The control system dynamically prevents the robot from falling over.

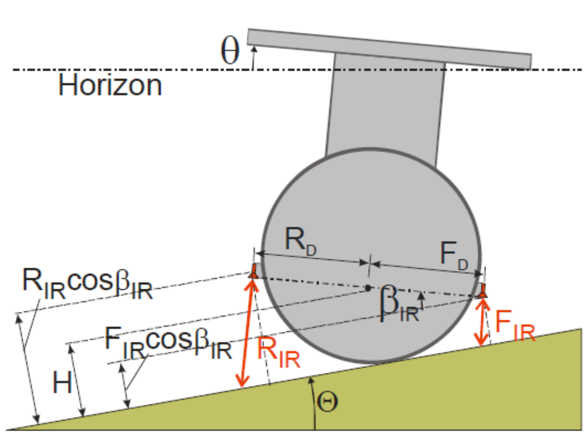
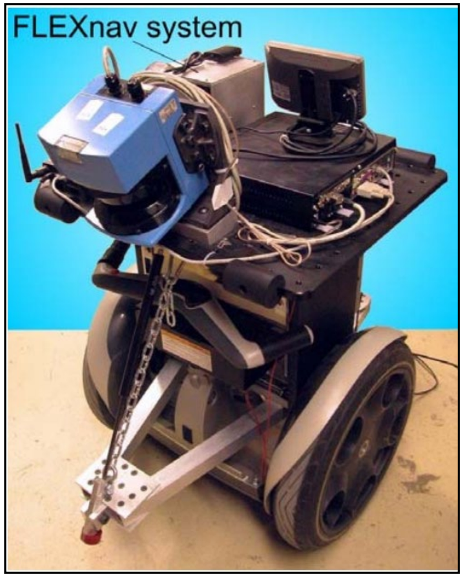


Figure 7: Range sensor location and corresponding angles.



2004, FLEXnav: A Fuzzy Logic Expert Dead-reckoning System for the Segway RMP
Lauro Ojeda, Mukunda Raju and Johann Borenstein
The University of Michigan, Advanced Technologies Lab

Sample Applications

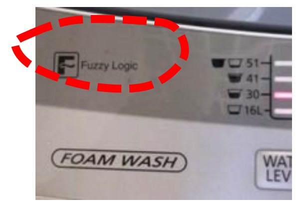
What about hoverboards? how to build its control system?

- self-balancing, two-wheeled scooters
- you just lean to move forward and put weight on whichever foot to go left or right.

Sample Applications

In high-end washing machines, dishwashers and refrigerators, fuzzy logic is now commonplace.

- These machines offer the advantages of performance, simplicity, productivity, and less cost.
- Sensors continually monitor varying conditions inside the machine and accordingly adjust operations for the best wash results.
- Typically, fuzzy logic controls the washing process, water intake, water temperature, wash time, rinse performance, and spin speed. This optimises the life span of the washing machine.
- (Neuro-Fuzzy) Some machines even learn from past experience, memorising programs and adjusting them to minimise running costs.



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.

- Fuzzy Logic: The Revolutionary Computer Technology That Is Changing Our World, by Daniel Mcneill, Paul Freiberger
- http://www.samsung.com/in/support/skp/faq/138486

Sample Applications

Fuzzy logic washing machines vs. PID washing machines

- Standard **PID controller** would require [**1,000-2,000**] rules to build the controller.
- **Fuzzy Logic** achieve the same results with **200** rules. (Productivity gain!)

Fuzzy logic dishwashers

- Using Fuzzy Logic the dishwasher can determine the dishwasher load. The duration and the water usage are then adapted accordingly, to ensure that no excess water is used. Rest assured that whether you wash a full load or a half load the dishwasher will adapt accordingly.

- http://www.electrolux.co.nz/support/usage-advice/dishwashers/

Meeting Lotfi in Germany

My Fuzzy Logic-based Research

- Navigation in unknown terrain (Hybrid Fuzzy-D*Lite)
- Robot soccer navigation
  - Real-time path-planning (Hybrid Fuzzy A*)
- Machine Vision
  - Real-time colour-object recognition
  - Fuzzy Colour Contrast Fusion

9th Fuzzy Days (2006), Dortmund, Germany

Meeting Prof. Yamakawa in Japan

ICONIP 2007, Kitakyushu, Japan

back

menu

Control Systems in General

n.h.reyes@massey.ac.nz

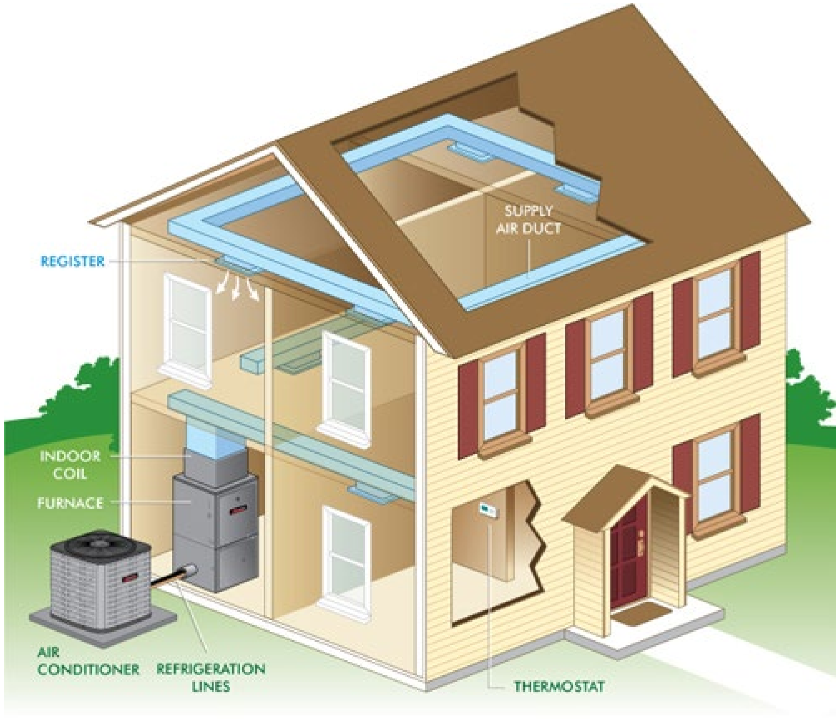
Control Systems in General

Objective

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

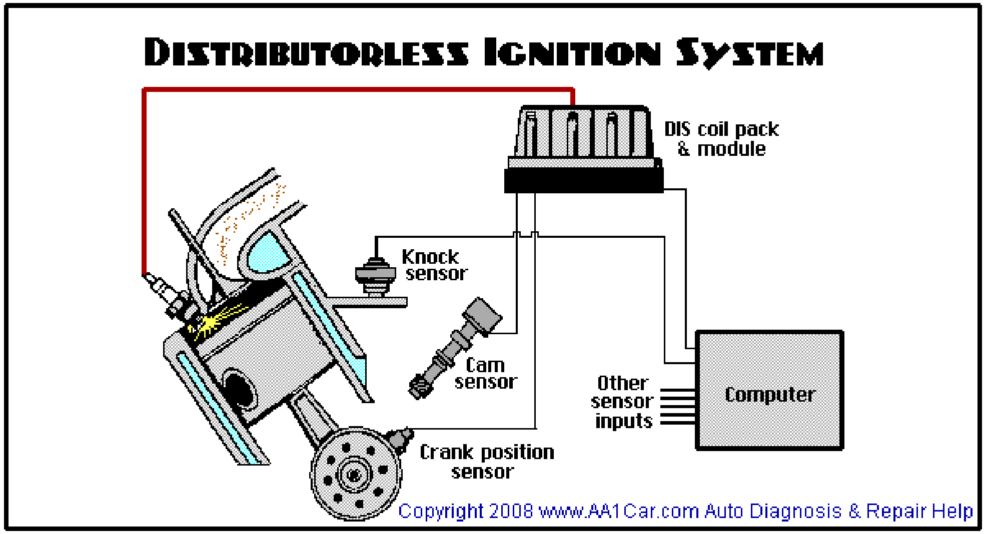
Example

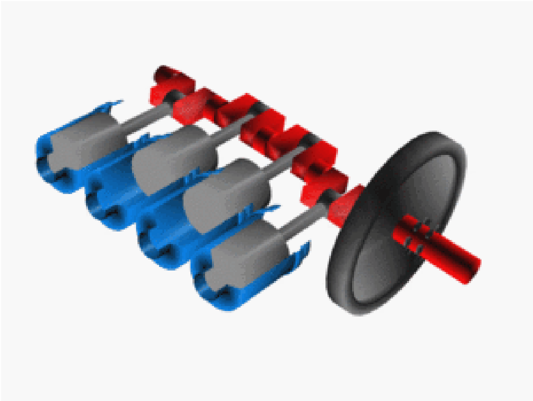
A household thermostat takes a temperature input and sends a control signal to a furnace.



Control Systems in General

Example





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.

Control Systems

Conventional Control vs. Fuzzy Control

n.h.reyes@massey.ac.nz

Conventional Control vs. Fuzzy

1. 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.

Conventional Control vs. Fuzzy

2. 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 of mathematical modeling

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

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

back

menu

Conventional Control vs. Fuzzy

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.

Troubleshooting a car (an analogy)

- A fuzzy system is analogous to a human expert who learned through experience.
- 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.

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.

back
menu

Fuzzy Set Theory

n.h.reyes@massey.ac.nz

Fuzzy Logic Explained

Fuzzy Set Theory

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.

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.

back

menu

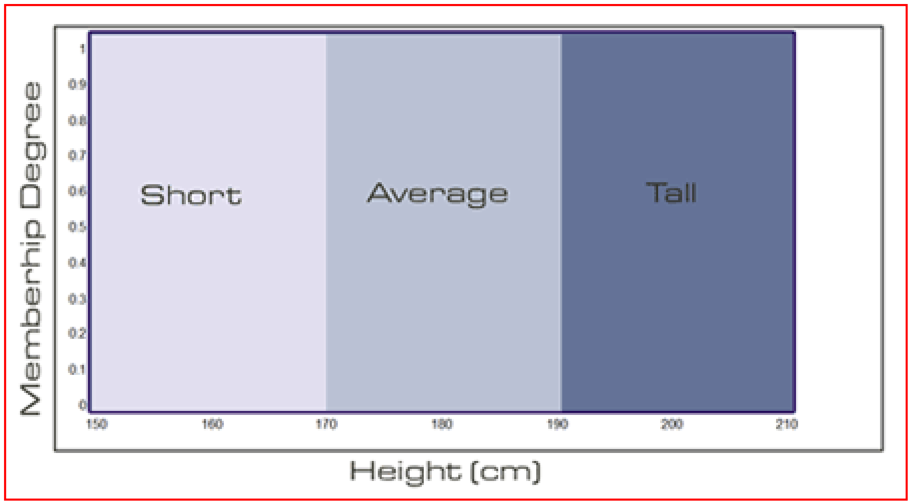
Fuzzy Logic Explained

Fuzzy Set Theory

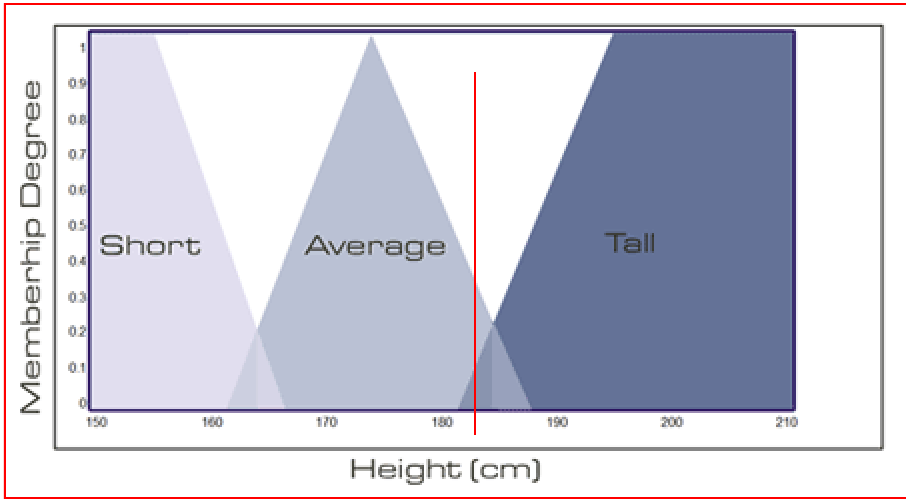
Is a man whose height is 5’ 11-1/2” average or tall?

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

Boolean representation

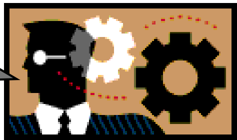


Fuzzy representation



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 Rules

Fuzzy rules may come in various forms

Examples:

(*a*) if pressure is high then volume is small — represents dependency
(*b*) if pressure is high and temperature is low then volume is very small — represents dependency, multiple antecedents
(*c*) if pressure is high then lower temperature slightly — command
(*d*) if pressure is high then volume is small unless temperature is high — represents dependency
(*e*) if pressure is high then usually volume is small. — represents dependency, dispositional

In these examples, pressure, volume and temperature are linguistic variables and small, low and high are their linguistic values.

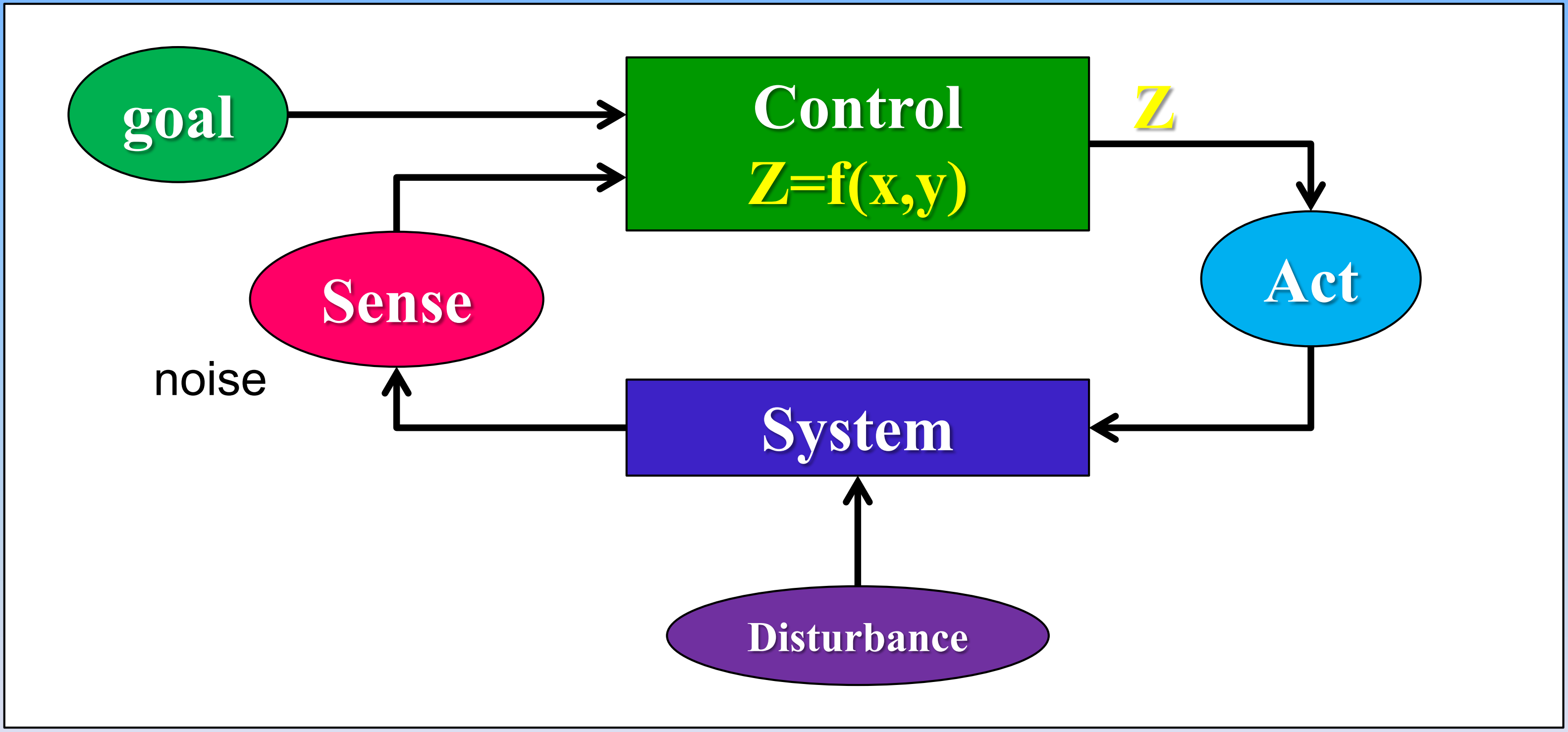
- All of the rules except (c) represent dependencies, with (c) representing a command.
- All of the rules except (d) and (e) are categorical.
- Rules (d) and (e) are qualified, with (d) qualified through an exception and (e) through usuality.
- (e) exemplifies what is referred to as a dispositional rule.
- All of the rules except for (b) involve a single variable in the antecedent

back

menu

Feedback Control/Closed Loop Control

- Closed-loop control allows for **uncertainty** in the model as well as **noise** and **disturbances** in the system under control



*__Controllers__ are used in the industry to **regulate** temperature, pressure, flow rate, chemical composition, speed and practically every other variable for which a measurement exists.

Fuzzy Inference Process

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

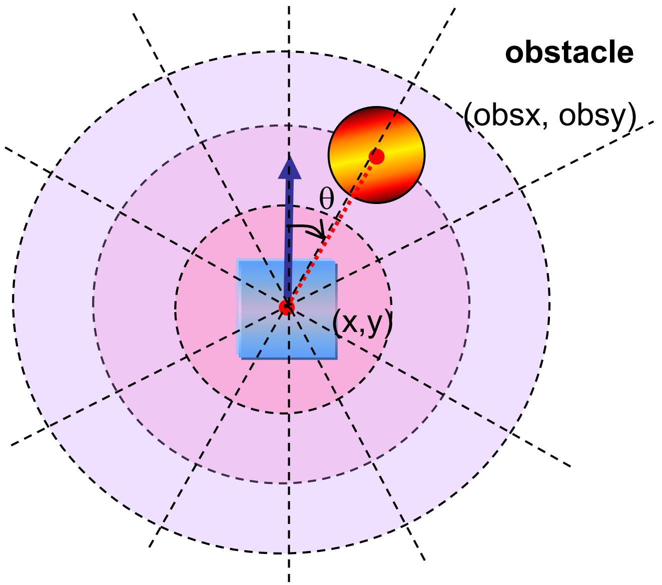
Fuzzy Inference Process

Fuzzy Inference Process

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?

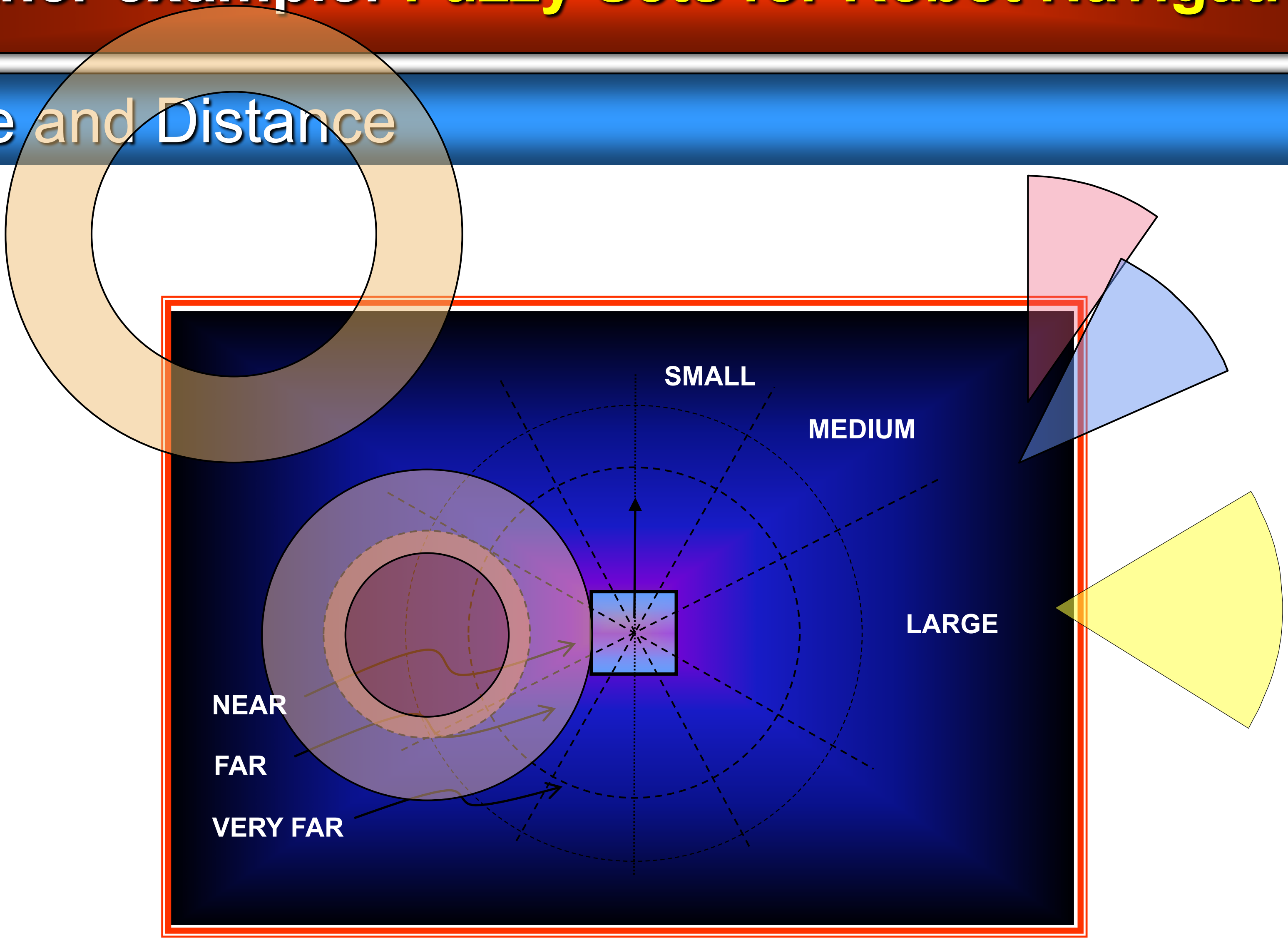
Obstacle Avoidance & Target Pursuit

Demonstration

back

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.*

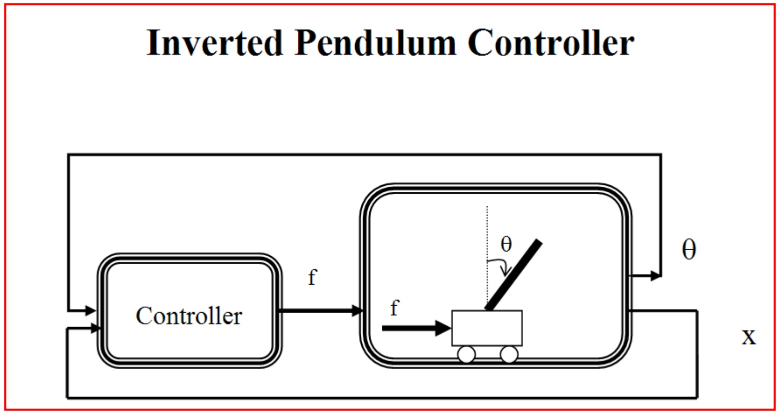
Speed

EXAMPLE

Inverted Pendulum Problem

Reference: Neural Network and Fuzzy Logic Applications in C/C++ (Wiley Professional Computing) by Stephen Welstead

Inverted Pendulum Problem



A Classic test case in embedded control

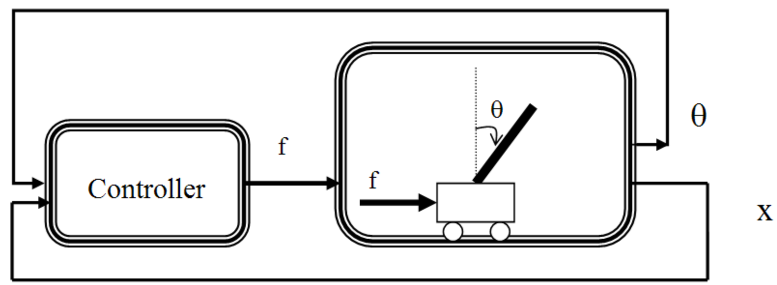
A pole with a weight on top is mounted on a motor-driven cart. The pole can swing freely, and the cart must move back and forth to keep it vertical.

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).

Inverted Pendulum Problem

Inverted Pendulum Controller



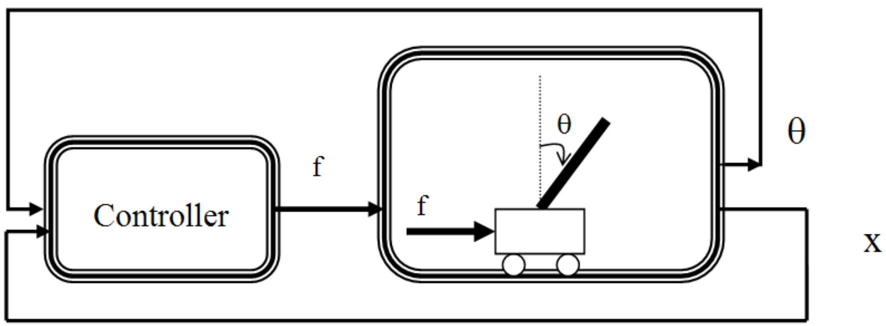
Conventional mathematical solution

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

Inverted Pendulum Controller



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

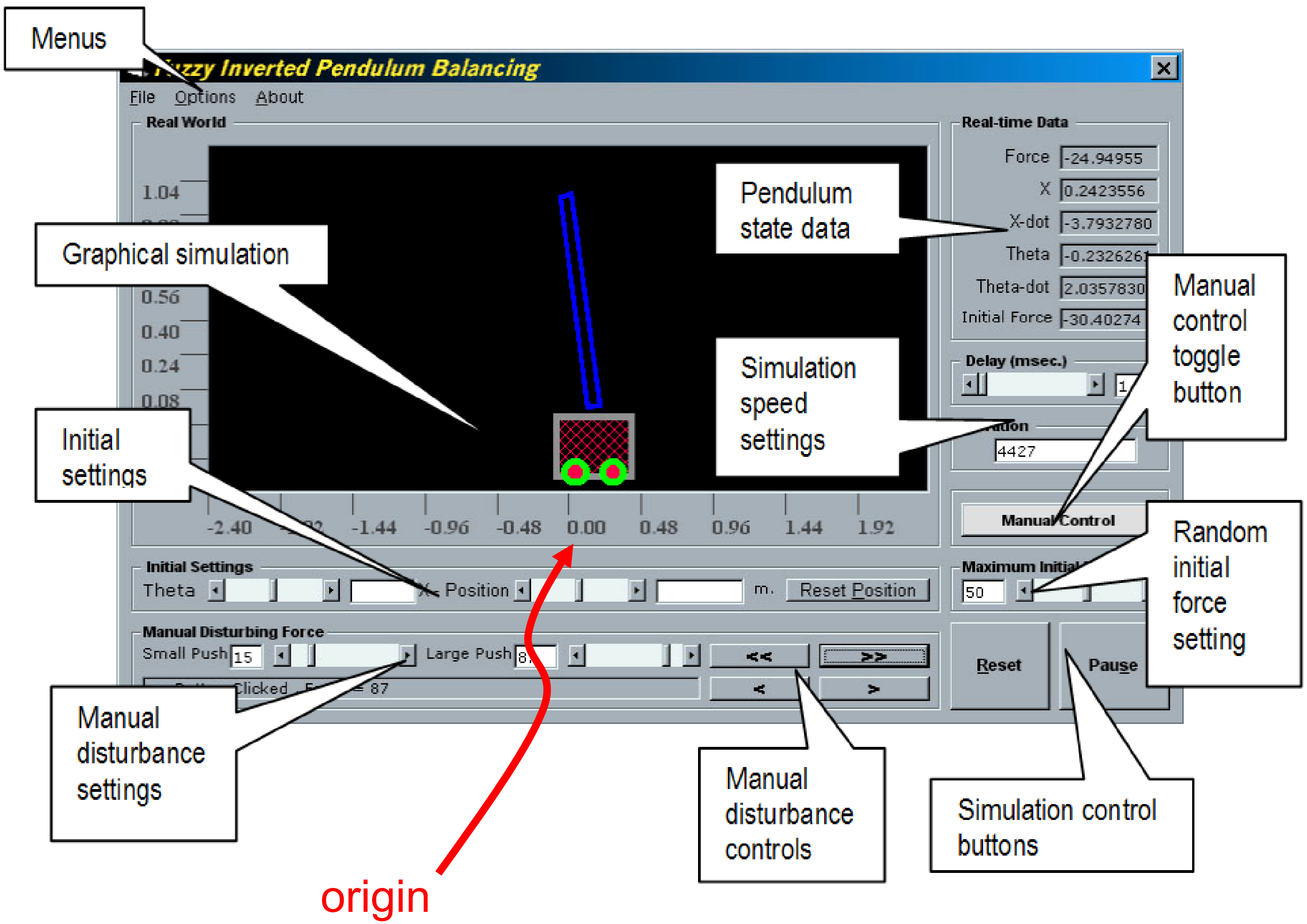
Derived Input Values

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



back

menu

Parameters for a Fuzzy System

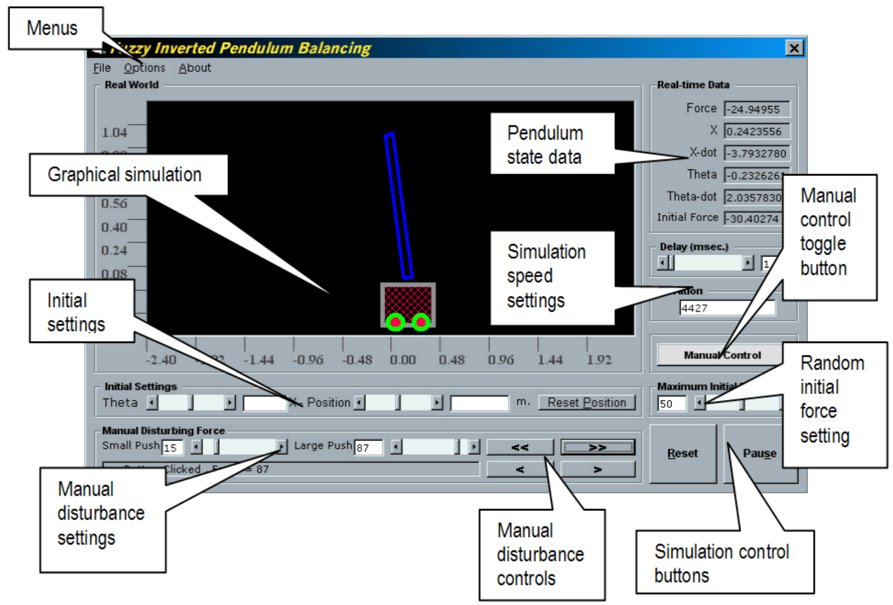
Once you have determined the appropriate inputs and outputs for your application, there are three steps to designing the parameters for a fuzzy system:

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**.

back

menu

Fuzzy Sets



We can 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:

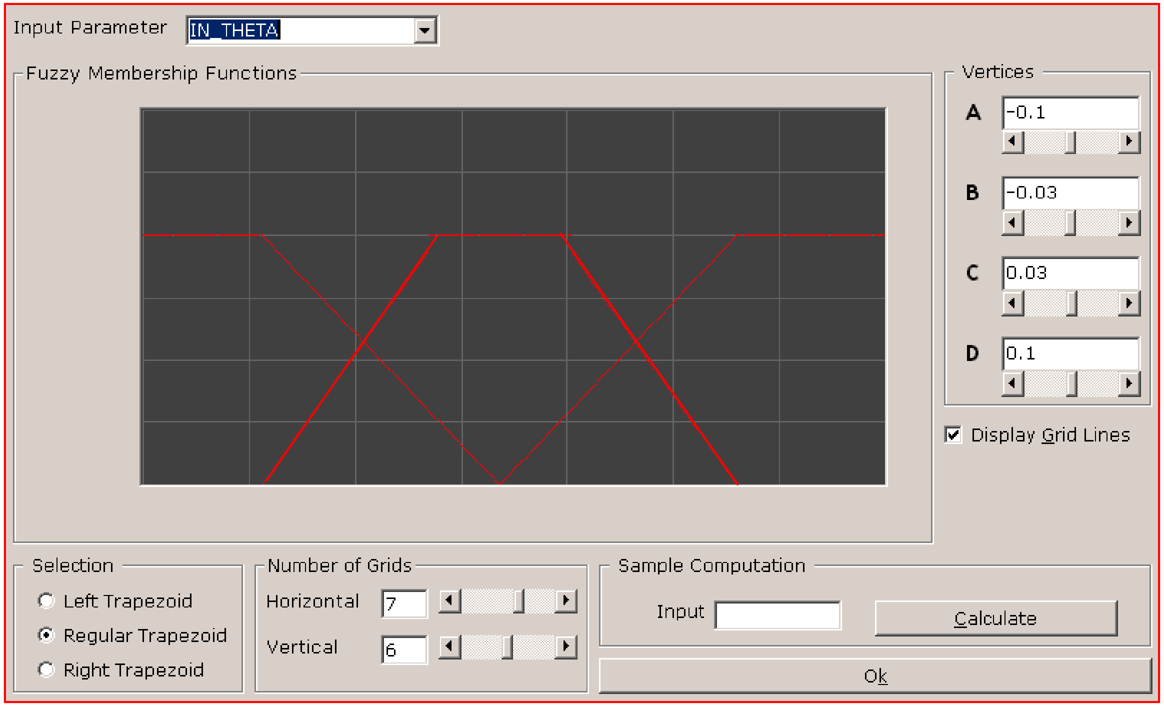
1. Inclined to the Left (N).
2. Vertical (Zero).
3. Inclined to the Right (P).

back

menu

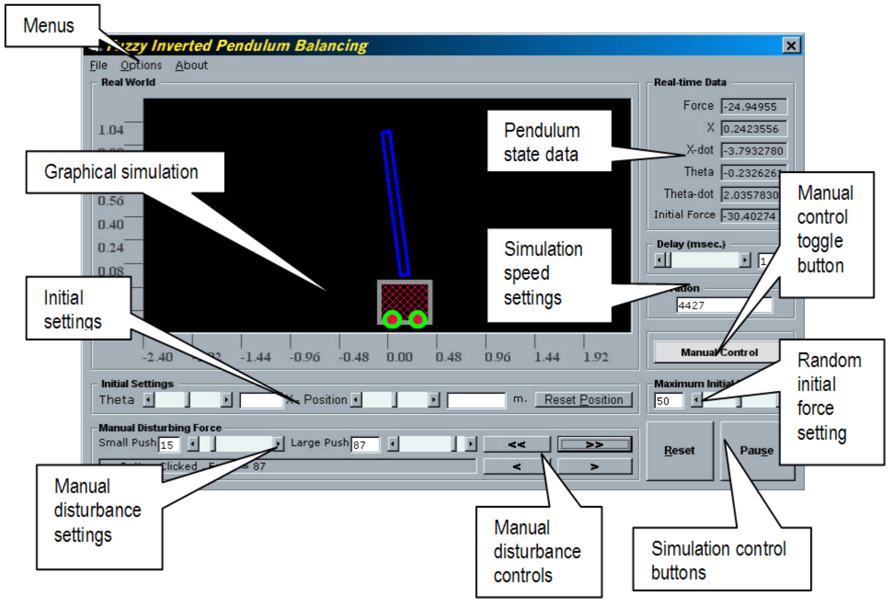
Membership Functions

Membership Functions for the Pole Angle



Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -0.1 rad. = -5.73 deg.	A = -0.1 rad = -5.73 deg.	A = 0
B = 0	B = -0.03 = -1.72 deg.	B = 0.1 rad = 5.73 deg.
C = 0	C = 0.03 = 1.72 deg.	C = 0
D = 0	D = 0.1 rad = 5.73 deg.	D = 0

Fuzzy Sets



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

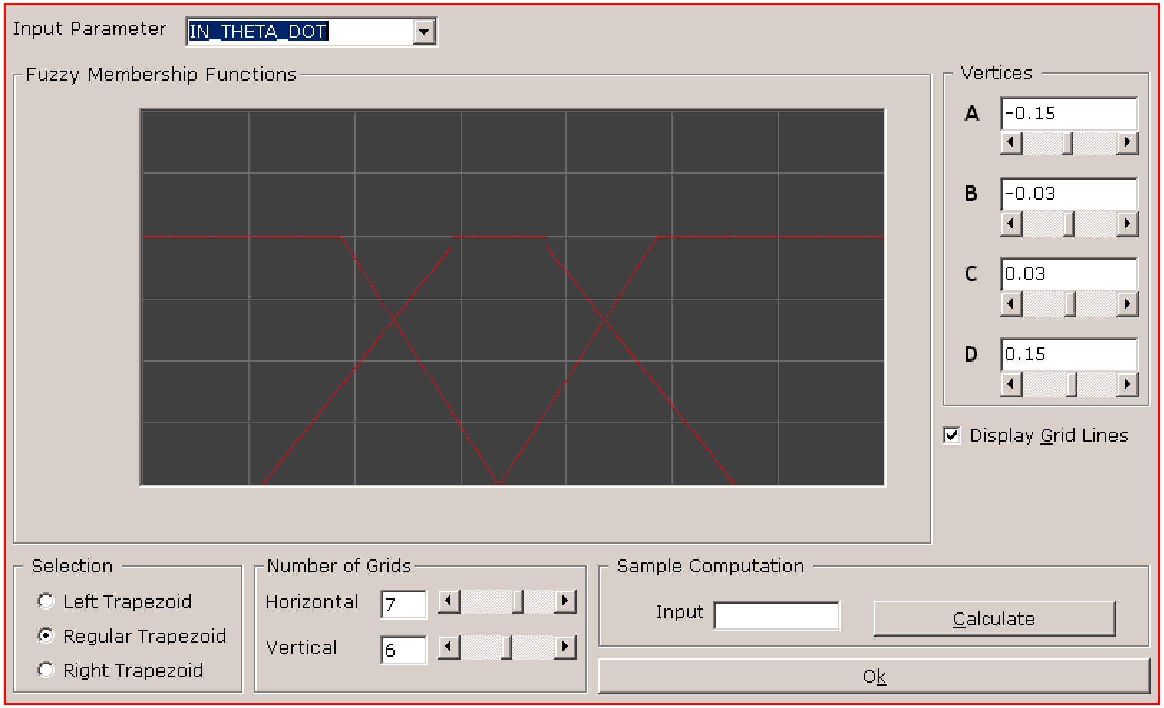
1. Falling to the Left (N).
2. Still (Zero).
3. Falling to the Right (P).

back

menu

Membership Functions

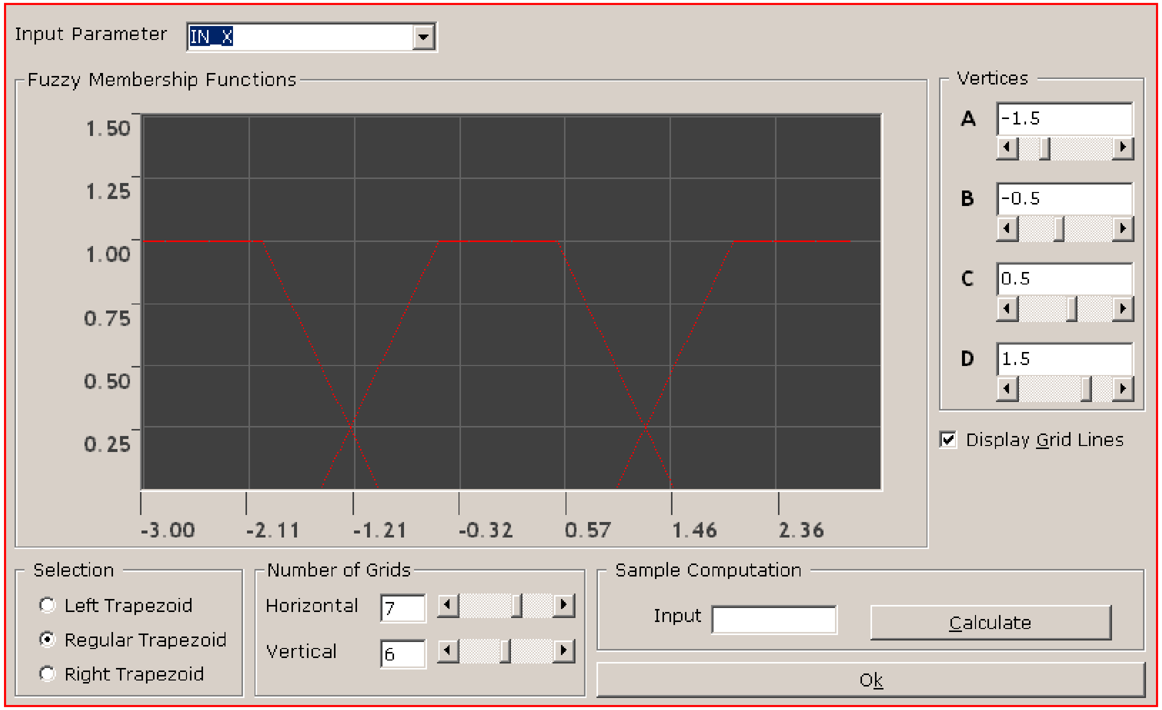
Membership Functions for the Broom Angular Velocity



Trapezoid Vertices		
Left Trapezoid	**Regular**	**Right**
A = -0.1=-5.73 deg/s.	A = -0.15 rad/s = -8.59 deg/s	A = 0
B = 0	B = -0.03 rad/s =-1.72 deg/s	B = 0.1 rad/s = 5.73 deg/s
C = 0	C = 0.03=1.72 deg/s	C = 0
D = 0	D = 0.15= 8.59 deg/s	D = 0

Membership Functions

Membership Functions for the Cart Position x

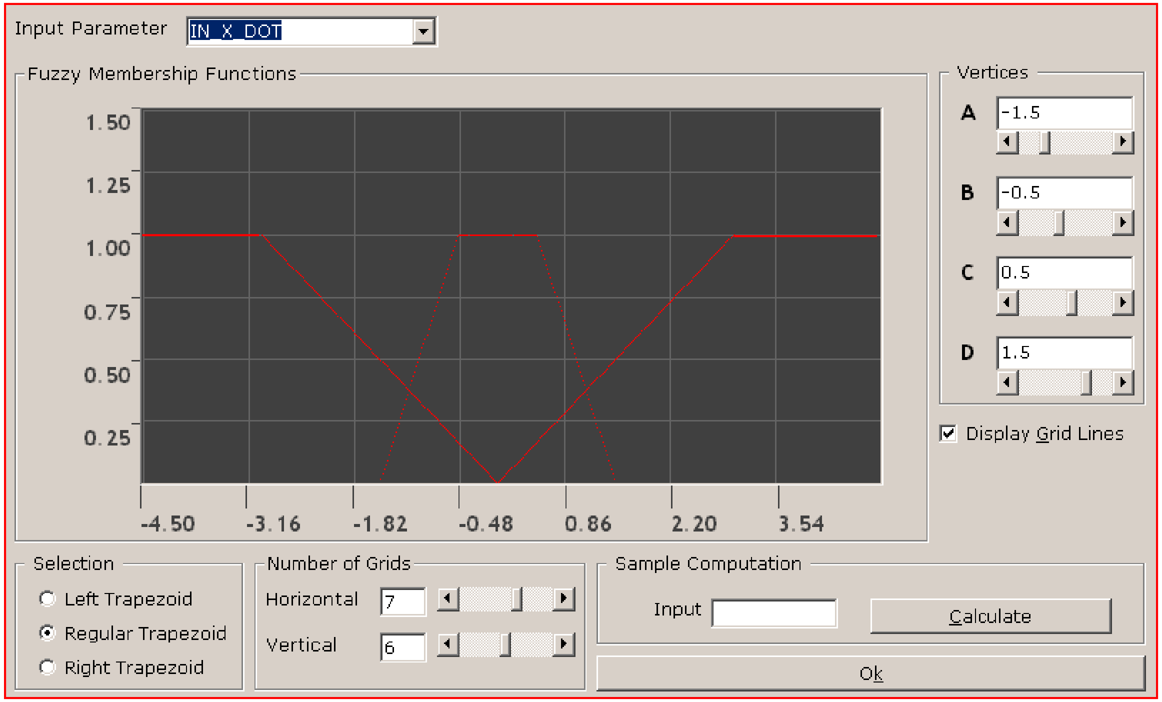


Take note of the position of the origin.

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

Membership Functions

Membership Functions for the Cart Velocity



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

FUZZY LOGIC SYSTEM

FAMM2: θ vs. θ'

n.h.reyes@massey.ac.nz

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

……..

Input θ: {N, ZE, P}={leaning to the left, centre, leaning to the right}

Input θ': {N, ZE, P}={moving to the left, still, moving to the right}

F: {PL, PS, ZE, NS, NL}={Large push to the right, small push to the right, don't push, small push to the left, Large push to the left}

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.

3 x 3 FAMM

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

Input θ: {N, ZE, P}={leaning to the left, centre, leaning to the right}

Input θ': {N, ZE, P}={moving to the left, still, moving to the right}

F: {PL, PS, ZE, NS, NL}={Large push to the right, small push to the right, don't push, small push to the left, Large push to the left}

back
menu

Fuzzy Rule

If (θ is NEGATIVE **and** θ' is **NEGATIVE**)

Then NEGATIVELY_LARGE_FORCE.

$F_{SMALL}(\theta')$ = degree of membership of the given **ANGULAR VELOCITY** in the Fuzzy Set NEGATIVE

Could be a constant or another Membership Function

$F_{NEGATIVE}(\theta)$ = degree of membership of the given **ANGLE** in the Fuzzy Set NEGATIVE

Input θ: {N, ZE, P}={leaning to the left, centre, leaning to the right}

Input θ': {N, ZE, P}={dropping to the left, still, dropping to the right}

F: {PL, PS, ZE, NS, NL}={Large push to the right, small push to the right, don't push, small push to the left, Large push to the left**}**

FUZZY LOGIC SYSTEM

FAMM1: x vs. x'

n.h.reyes@massey.ac.nz

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

Input X: {N, ZE, P} = {left, centre, right}

Input X’: {N, ZE, P} = {going to the left, still, going to the right}

F: {PL, PS, ZE, NS, NL}={Large push to the right, small push to the right, don’t push, small push to the left, Large push to the left**}**

back

menu

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.

3 x 3 FAMM

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

Input X: {N, ZE, P} = {left, centre, right}

Input X’: {N, ZE, P} = {going to the left, still, going to the right}

F: {PL, PS, ZE, NS, NL}={Large push to the right, small push to the right, don’t push, small push to the left, Large push to the left}

(FAMM 1) and (FAMM 2)

The final output can be calculated by combining all the rule outputs from the two FAMMS, into one big centre of mass formula.

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

FAMM 1

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

FAMM 2

Final output

FUZZY LOGIC SYSTEM

Alternate solution
(Yamakawa's approach)

(Yamakawa's approach)

Fuzzy Rules (FAMM 1)

We can reduce the number of rules further by combining related inputs together in a linear equation.

$$X = (A * x) + (B * x\_dot)$$

$$Y = (C * theta) + (D * theta\_dot)$$

Note: A, B, C & D are determined empirically.

This approach requires only a single FAMM to solve the control problem.

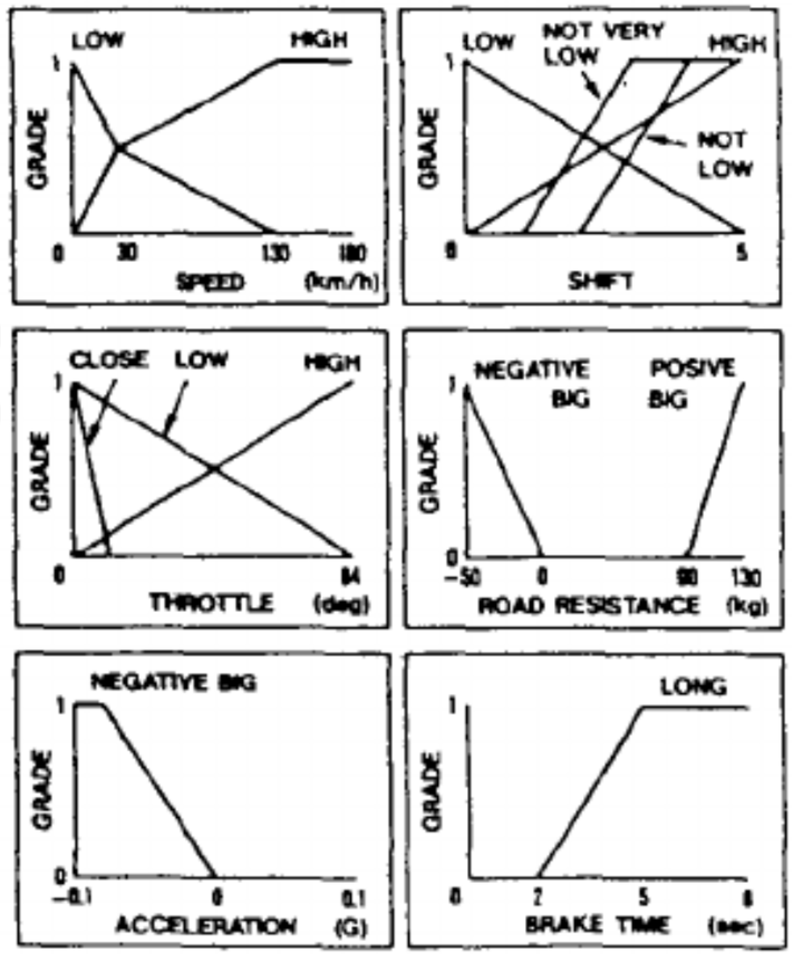
menu

Honda's fuzzy logic transmission

CONTROL RULES

1. IF (SPEED IS LOW) AND (SHIFT IS HIGH)
   THEN (–3)
2. IF (SPEED IS HIGH) AND (SHIFT IS LOW)
   THEN (+3)
3. IF (THROT IS LOW) AND (SPEED IS HIGH)
   THEN (+3)
4. IF (THROT IS LOW) AND (SPEED IS LOW)
   THEN (+1)
5. IF (THROT IS HIGH) AND (SPEED IS HIGH)
   THEN (–1)
6. IF (THROT IS HIGH) AND (SPEED IS LOW)
   THEN (–3)
7. IF (RESIST IS POSITIVE BIG) AND
   (THROT IS LOW) AND
   (SHIFT IS NOT VERY LOW)
   THEN (–1.5)
8. IF (RESIST IS NEGATIVE BIG) AND
   (THROT IS CLOSE) AND
   (SHIFT IS NOT LOW)
   THEN (–2)
9. IF (BRAKE TIME IS LONG) AND
   (ALF IS NEGATIVE BIG) AND
   (SHIFT IS NOT LOW)
   THEN (–2)

FUZZY SET



Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers By Lotfi Asker Zadeh, George J. Klir, Bo Yuan, page 798

The role of fuzzy logic in modeling, identification and control (1994)

Other applications

1. Quantitative Fuzzy Semantics - Zadeh (page 159, world scientific)
2. A Rationale for fuzzy control – Zadeh (page 123, world scientific)

menu

Hand-Simulation

Fuzzy Control

Different stages of Fuzzy control

1. Fuzzification

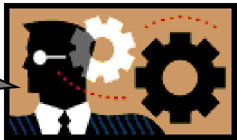
Input variables are assigned degrees of membership in various classes

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.

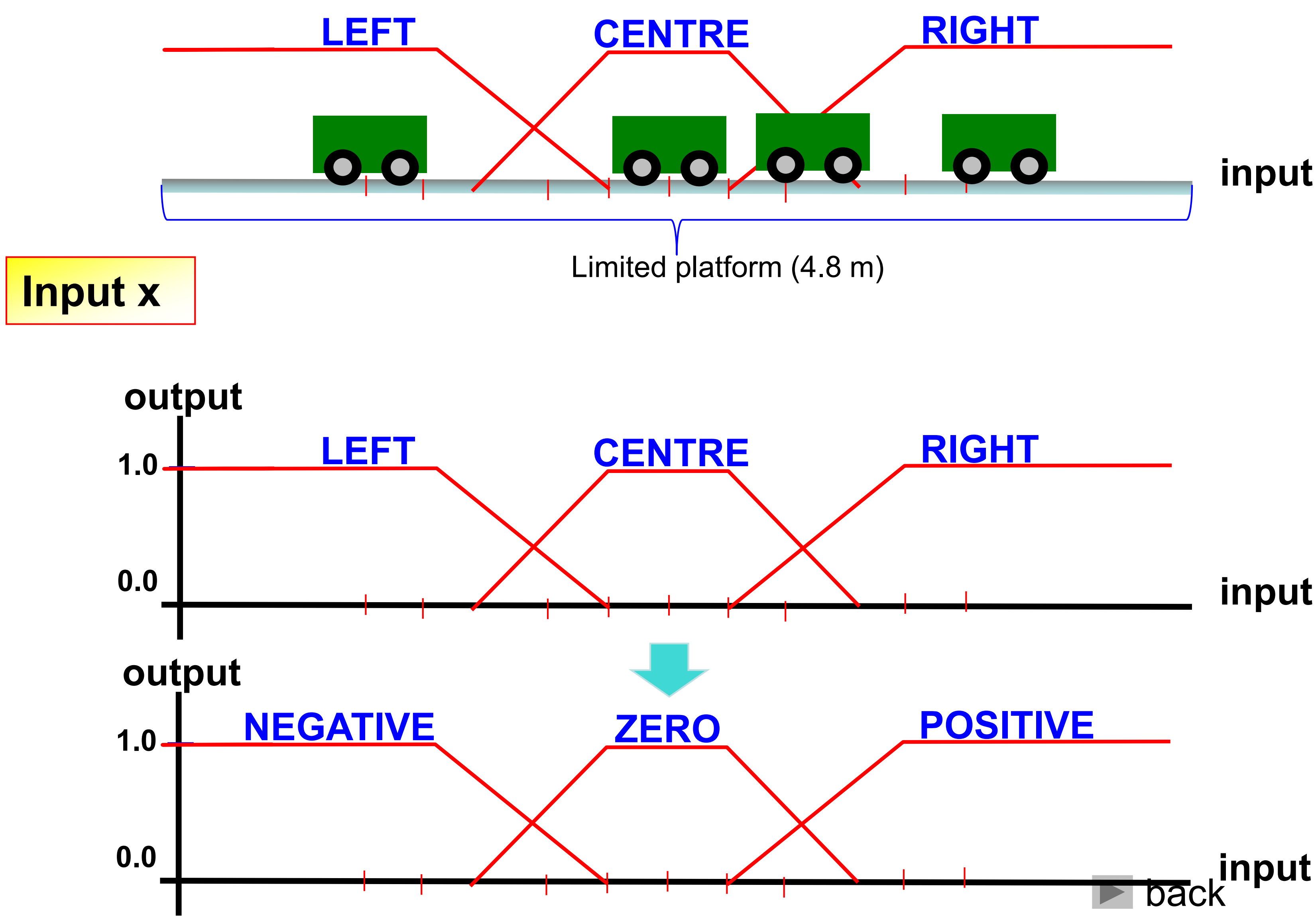
Membership Function: $\mathbf{F_{Set}(x)} = [0, 1]$ = *assigns a degree of membership of x, in the fuzzy set defined.*

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



Fuzzification

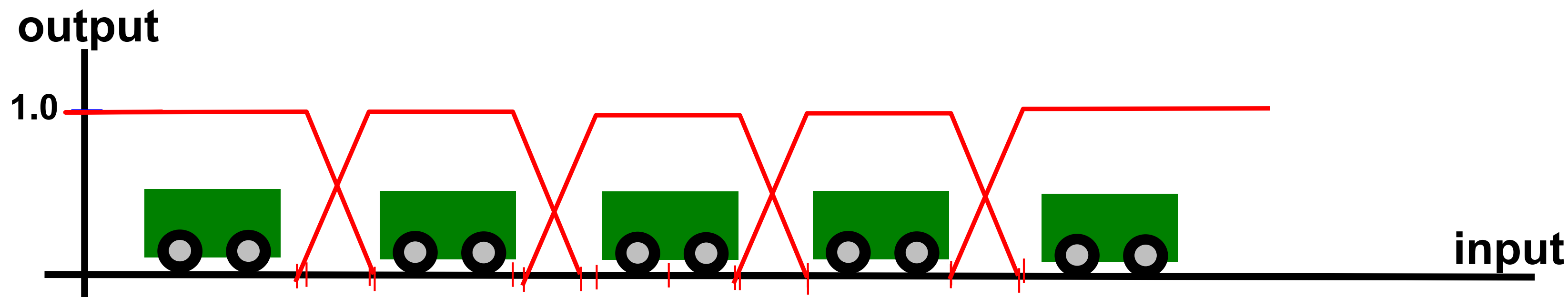
Let's consider one of the inputs, x (position of cart):



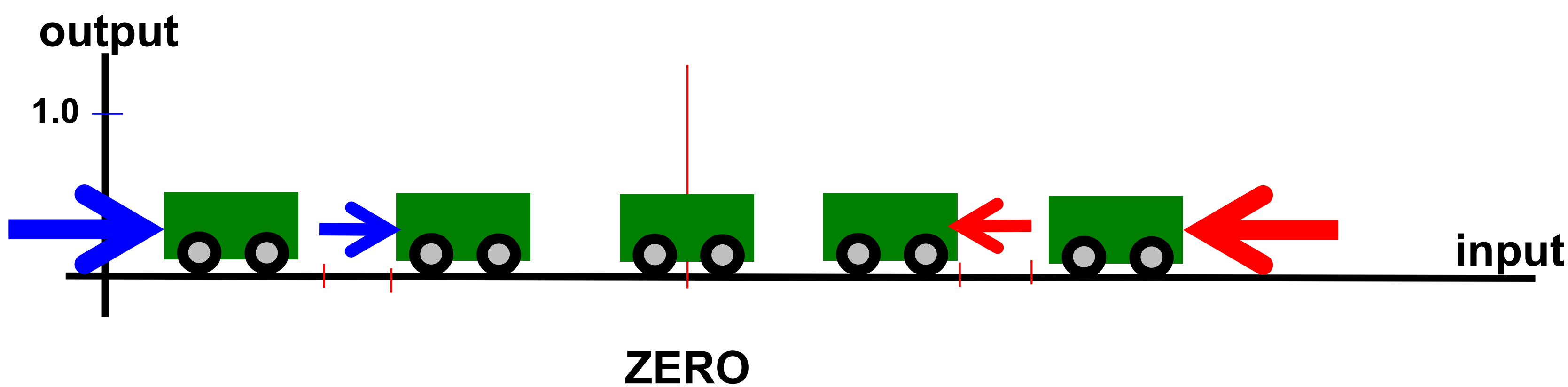
back

menu

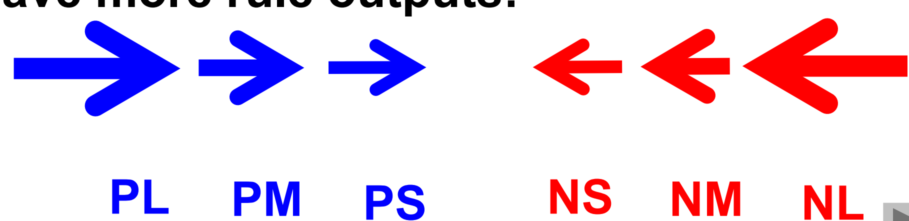
Fuzzification



By defining more fuzzy sets, more accurate control rules can be defined.

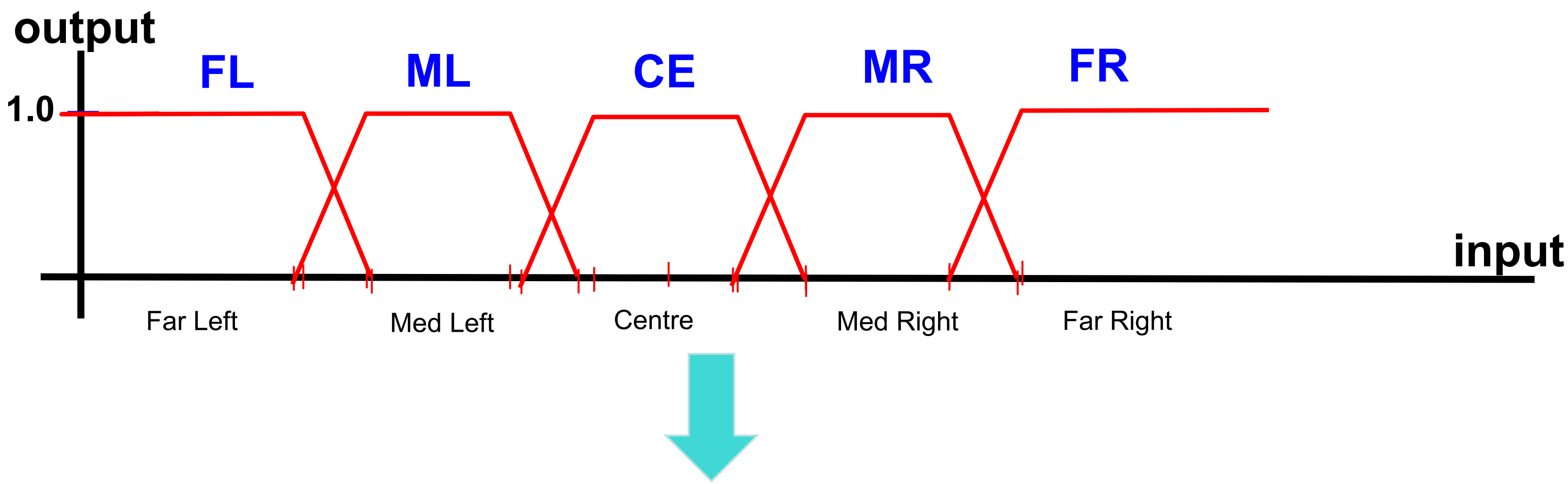


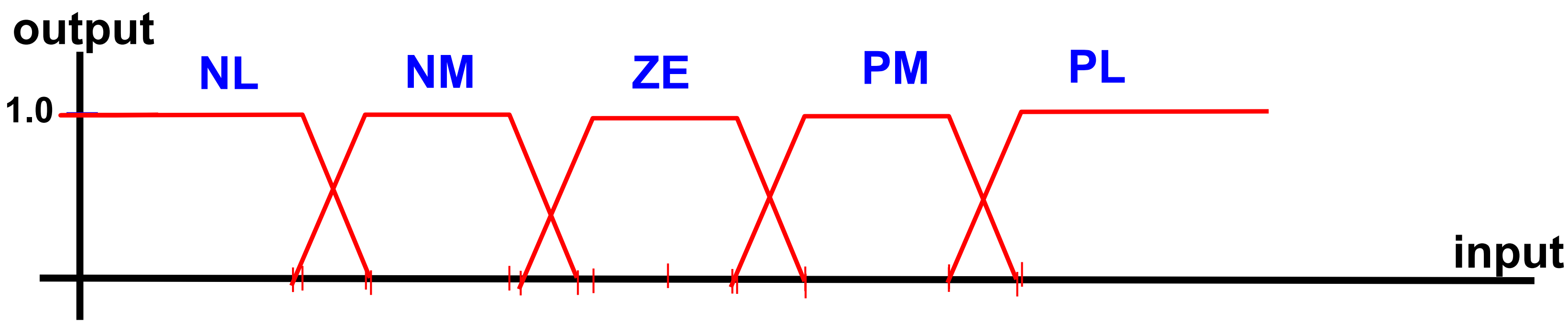
As desired, we can have more rule outputs:



back
menu

We can describe the position of the cart using the following 5 fuzzy sets.

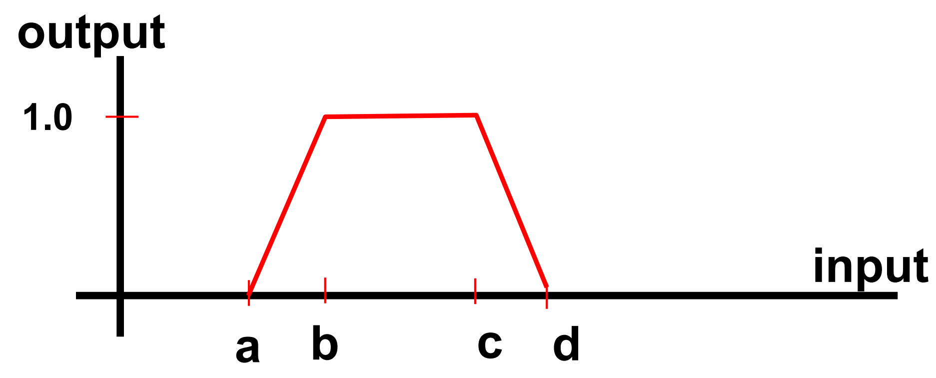




These terms are more commonly used in the literature.

Trapezoidal Membership Function

Crisp Input: x
Membership Function: F(x)



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

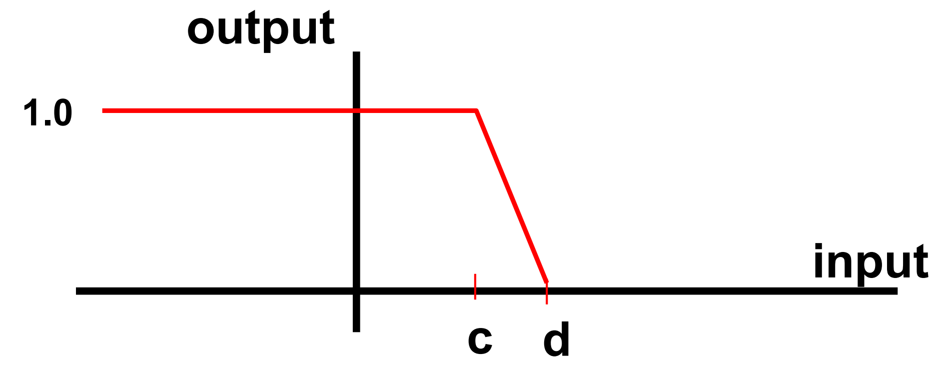
Given input

back

menu

Trapezoidal Membership Function

Crisp Input: x
Membership Function: F(x)

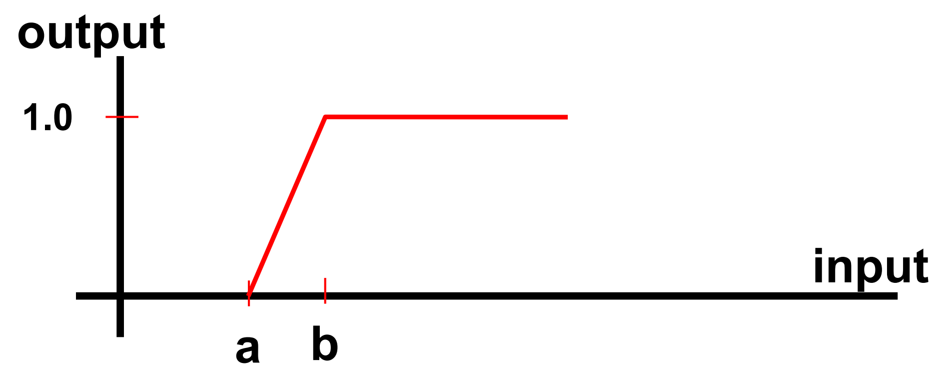


$$F_{left\_trapezoid}(x) = \max\left(\min\left(1, \frac{d-x}{d-c} \right), 0 \right)$$

Given input

Trapezoidal Membership Function

Crisp Input: x
Membership Function: F(x)



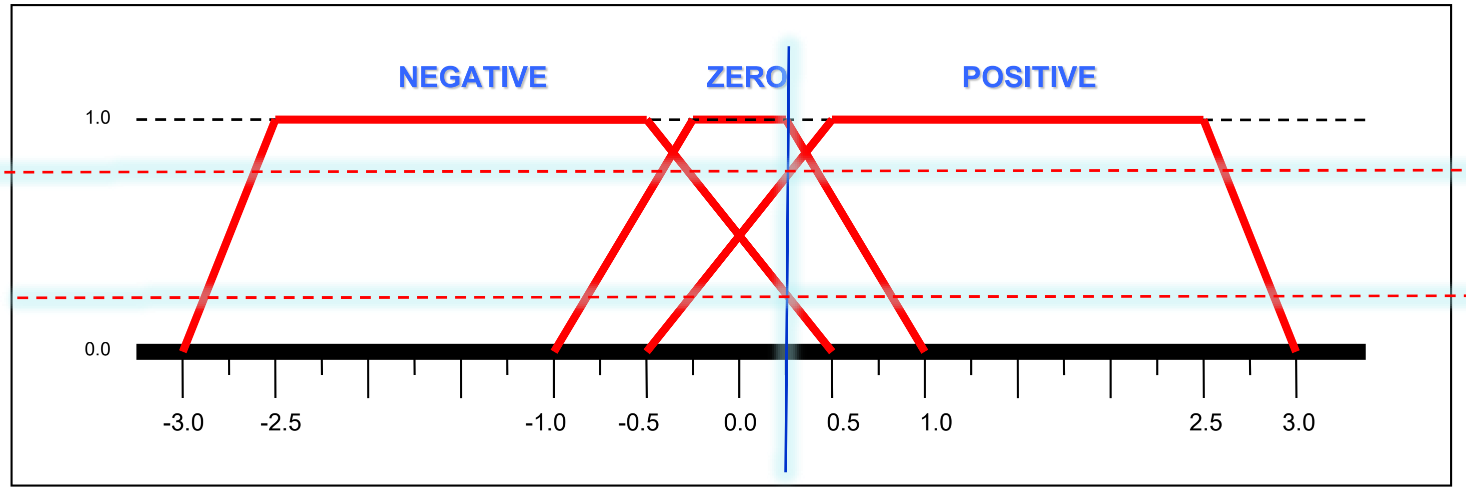
$$F_{right\_trapezoid}(x) = \max\left(\min\left(1, \frac{x-a}{b-a}\right), 0\right)$$

Given input

Fuzzification

Fuzzification Example

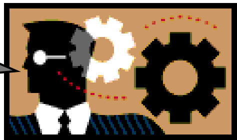
Fuzzy Sets = { Negative, Zero, Positive }



Assuming that we are using the same trapezoidal membership functions for both input variables, x and y.

Crisp Inputs: **x = 0.25, y = -0.25**

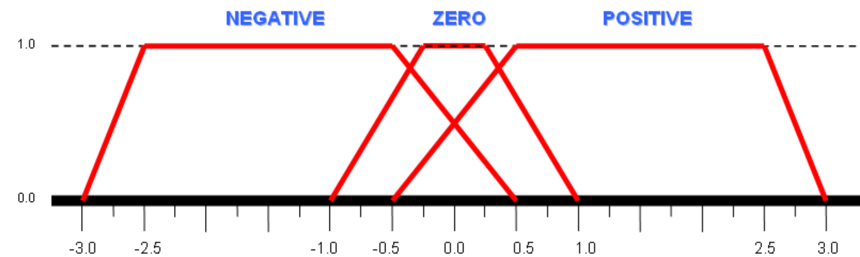
What is the degree of membership of x and y in each of the Fuzzy Sets?



back

Sample Calculations

Crisp Input: $x = 0.25$



$F_{zero}(0.25)$

F_{zero}: a=-1,b=-0.25, c=0.25, d=1.0

Given input

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

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

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

$$= 1$$

$F_{positive}(0.25)$

$F_{positive}$: a=-0.5,b=0.5, c=2.5, d=3.0

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

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

$$= 0.75$$

$F_{negative}(0.25)$

$F_{negative}$: a=-3.0,b=-2.5, c=-0.5, d=0.5

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

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

$$= 0.25$$

back

menu

Sample Calculations

Crisp Input: $y = -0.25$

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

$F_{zero}(-0.25)$

F_{zero}: a=-1,b=-0.25, c=0.25, d=1.0

Given input

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

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

$$= 1$$

$F_{positive}(-0.25)$

$$F_p(-0.25) = \max\left(\min\left(\frac{-0.25-(-0.5)}{0.5-(-0.5)},1,\frac{3.0-(-0.25)}{3.0-2.5}\right),0\right)$$

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

$$= 0.25$$

$F_{positive}$: a=-0.5,b=0.5, c=2.5, d=3.0

$F_{negative}(-0.25)$

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

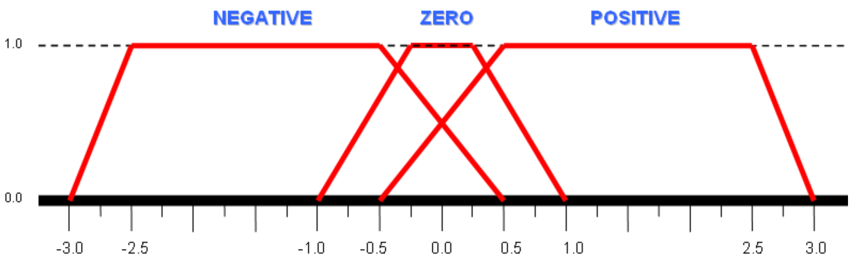
$$= \max\left(\min\left(5.5,1,0.75\right),0\right)$$

$$= 0.75$$

$F_{negative}$: a=-3.0,b=-2.5, c=-0.5, d=0.5

back
menu

Fuzzification results



Crisp Input: $x = 0.25$

$F_{zero}(0.25)=1$

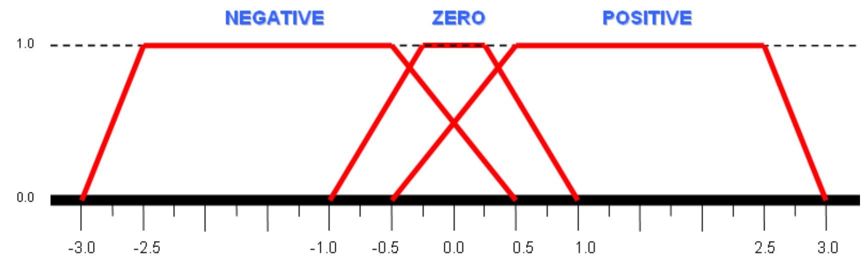
F_{zero}: a=-1,b=-0.25, c=0.25, d=1.0

$F_{positive}(0.25)=0.75$

$F_{positive}$: a=-0.5,b=0.5, c=2.5, d=3.0

$F_{negative}(0.25) = 0.25$

$F_{negative}$: a=-3.0,b=-2.5, c=-0.5, d=0.5



Crisp Input: $y = -0.25$

$F_{zero}(-0.25) = 1$

F_{zero}: a=-1,b=-0.25, c=0.25, d=1.0

$F_{positive}(-0.25) = 0.25$

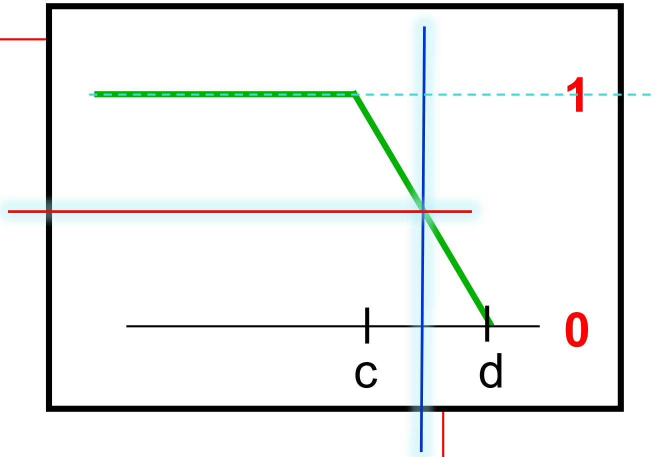
$F_{positive}$: a=-0.5,b=0.5, c=2.5, d=3.0

$F_{negative}(-0.25) = 0.75$

$F_{negative}$: a=-3.0,b=-2.5, c=-0.5, d=0.5

Trapezoidal Membership Functions

Alternative Implementation



LeftTrapezoid

Left_Slope = 0
Right_Slope = 1 / (c - d)

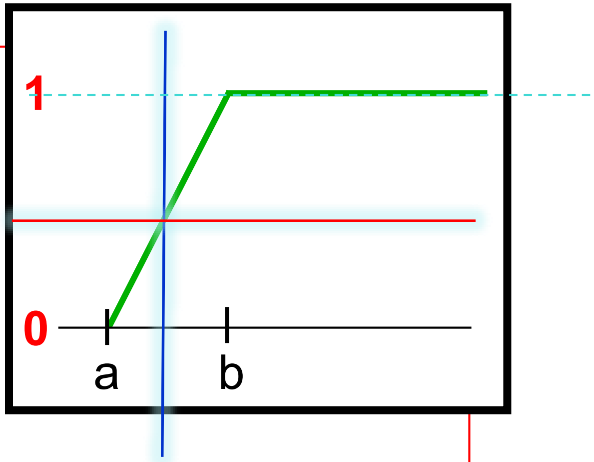
CASE 1: X <= c
Membership Value = 1

CASE 2: X >= d
Membership Value = 0

CASE 3: c < x < d
Membership Value = Right_Slope * (X - d)

Trapezoidal Membership Functions

Alternative Implementation



RightTrapezoid

Left_Slope = 1 / (b - a)
Right_Slope = 0

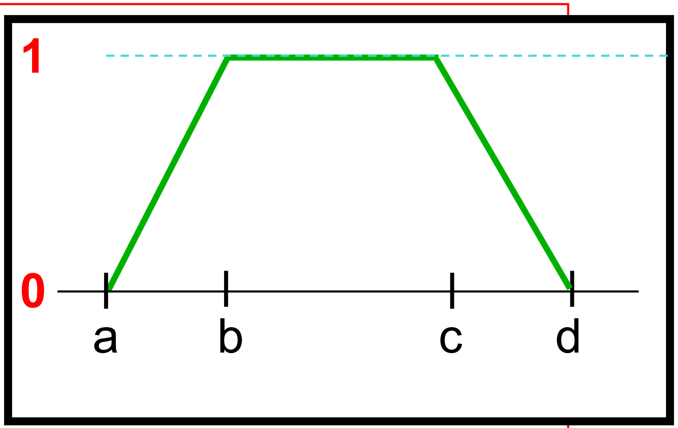
CASE 1: X <= a
Membership Value = 0

CASE 2: X >= b
Membership Value = 1

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

Trapezoidal Membership Functions

Alternative Implementation



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)

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.*

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

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

Rule Strength

W1	W4	W7
W2	W5	W8
W3	W6	W9

Fuzzy Control

Rule Evaluation Example

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

FAMM

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

Rule Strength

W1	W4	W7
W2	W5	W8
W3	W6	W9

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

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

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

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

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

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

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

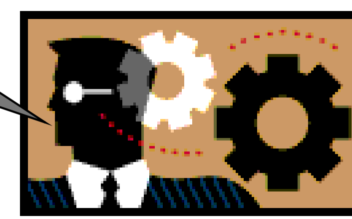
$$W_8 = \min[F_P(0.25), F_{ZE}(-0.25)] = \min[0.75, 1] = 0.75$$

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

Fuzzified inputs
back
menu

Does a FAMM need to be a square?

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



back

menu

Fuzzy Control

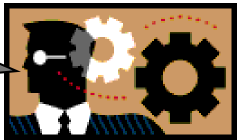
Different stages of Fuzzy control

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.

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

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

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

FAMM

<u>Outputs</u>
NL=-5
NS=-2.5
ZE=0
PS=2.5
PL=5.0

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

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

Fuzzified inputs   back   menu

Summary of Steps

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

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

More details…

- Fuzzy Inference Systems (FIS)
- Fuzzy Sets
- Fuzzy Combination Operators
- Membership functions
- Implication and Aggregation Operators
- Defuzzification Methods

Fuzzy Inference

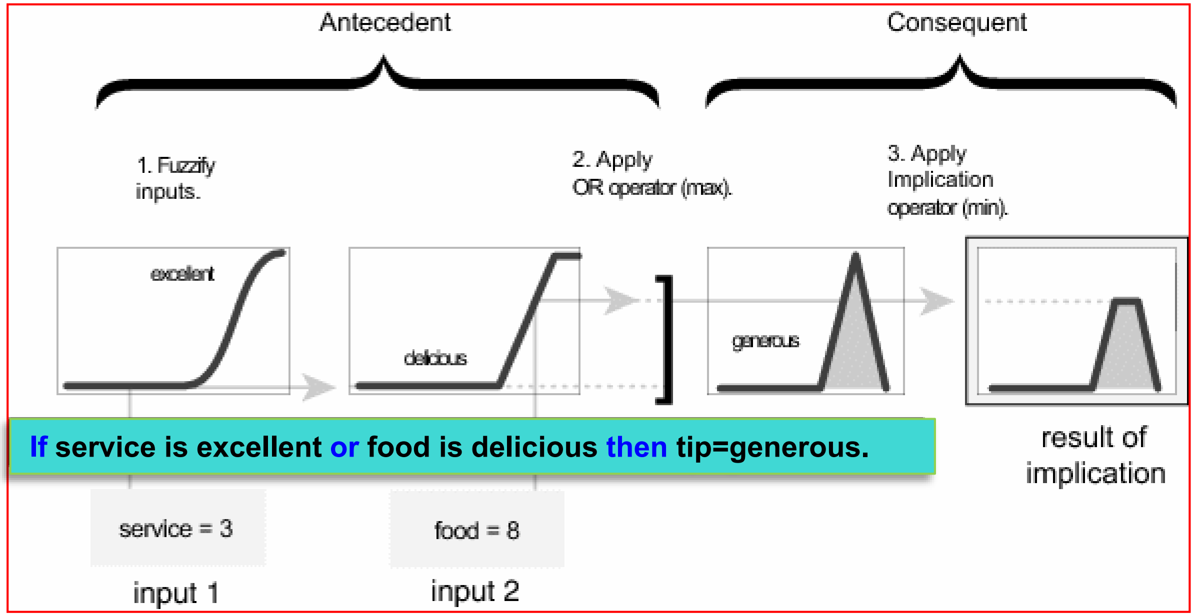
Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis from which decisions can be made, or patterns discerned. The process of fuzzy inference involves all of the pieces that were described in the previous sections: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

http://www.mathworks.com/help/fuzzy/fuzzy-inference-process.html

Mamdani Fuzzy Inference System

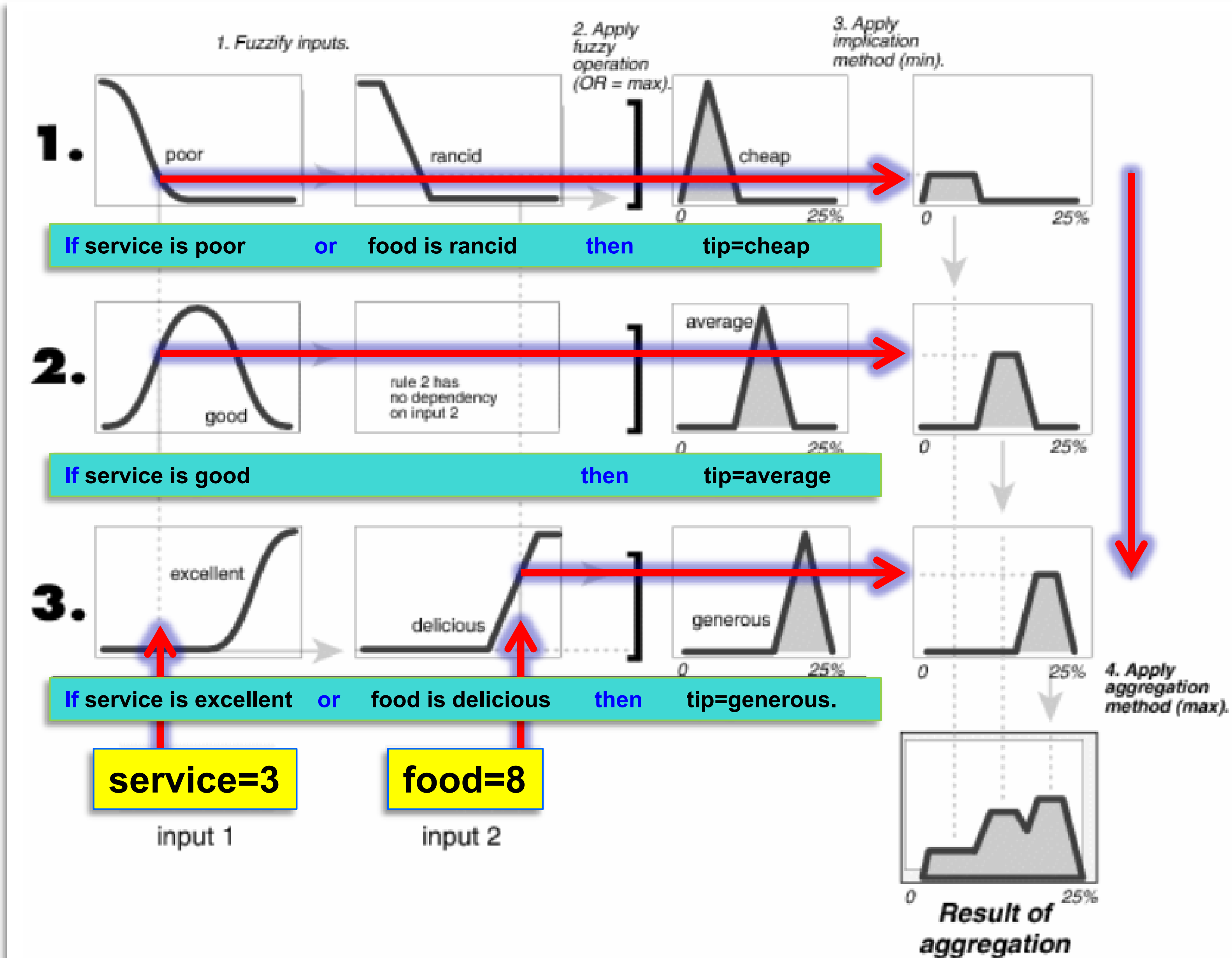
Mamdani FIS

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.



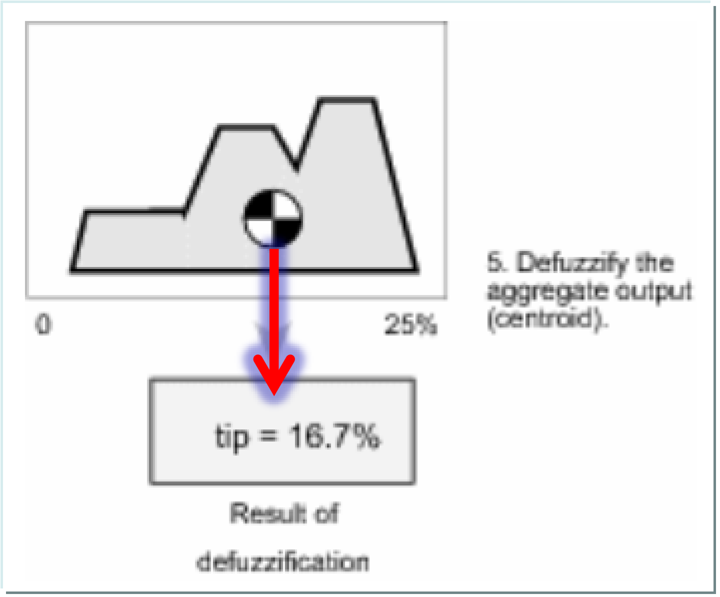
http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/index.html?/access/helpdesk/help/toolbox/fuzzy/bp78l6_-1.html&http://www.mathworks.com/cgi-bin/texis/webinator/search/?db=MSS&prox=page&rorder=750&rprox=750&rdfreq=500&rwfreq=500&rlead=250&sufs=0&order=r&is_summary_on=1&ResultCount=10&query=fuzzy+logic&submitButtonName=Search

Mamdani FIS



menu

Mamdani FIS



back

menu

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**.

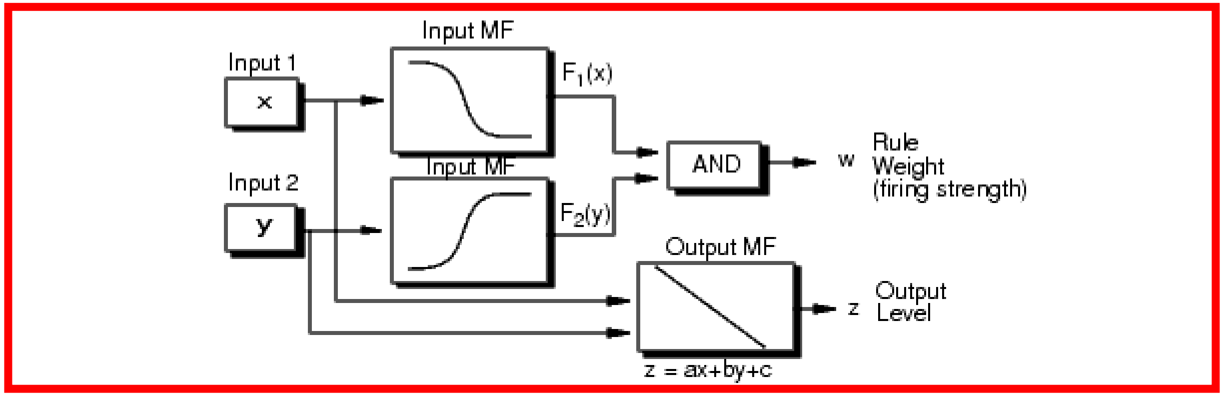
- It enhances the efficiency of the defuzzification process because it **greatly simplifies the computation** required by the more general Mamdani method, which finds the **centroid of a 2-D function**.

- Rather than integrating across the two-dimensional function to find the centroid, you use the **weighted average of a few data points**.

Sugeno Fuzzy Inference System

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:

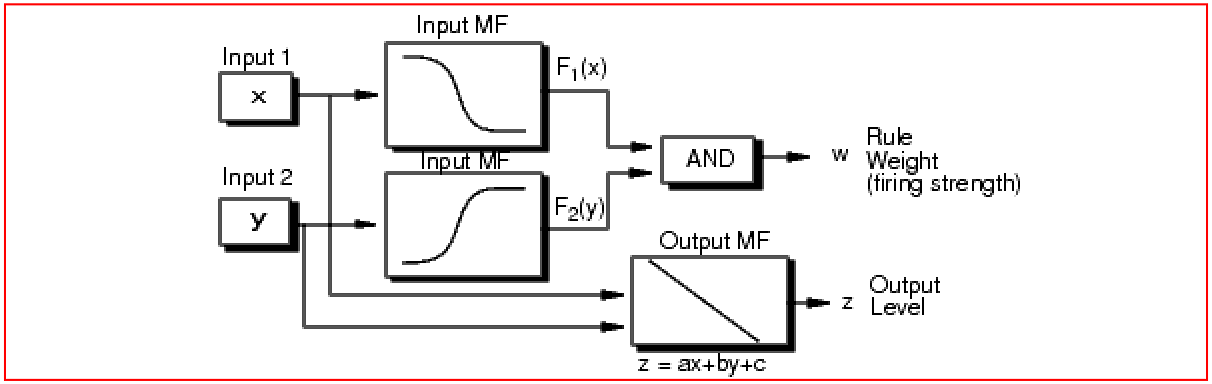
If (Input 1 = x and Input 2 = y) **then** Output is **z = ax + by + c**

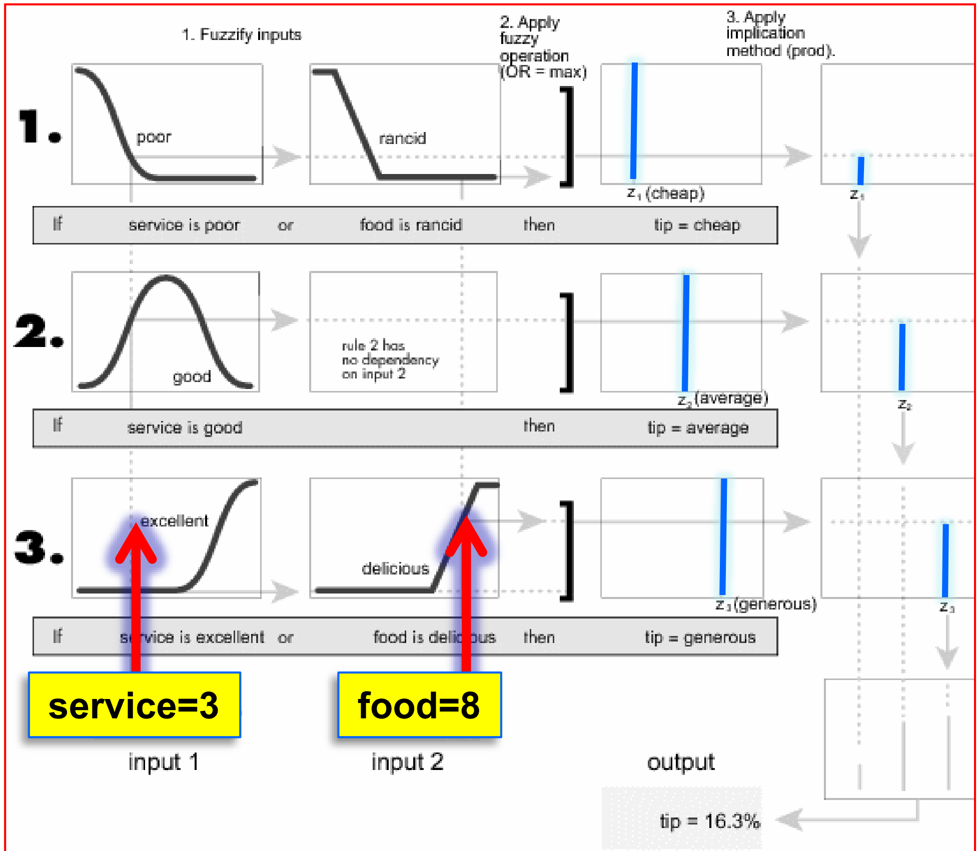
For a **zero-order Sugeno model**, the output **z** is a **constant** (where a=b=0).

back

menu

Sugeno FIS





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

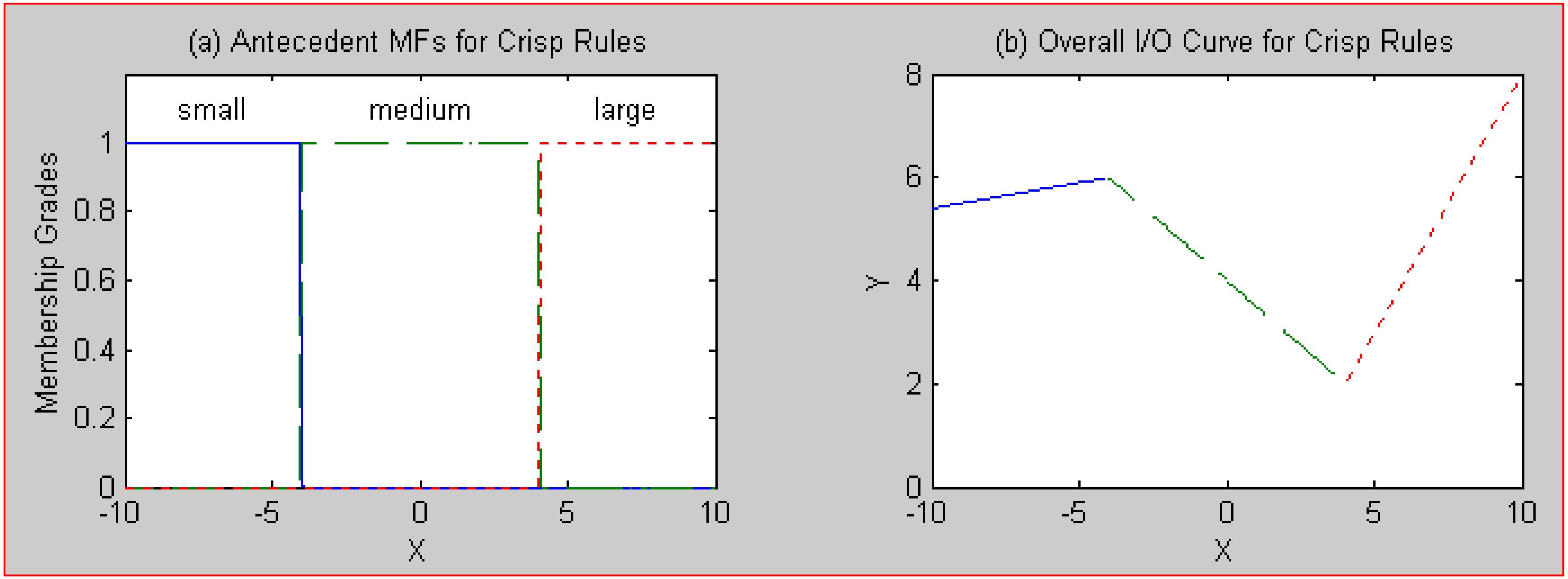
back

menu

Sugeno FIS Example

X = input $\in [-10, 10]$

R1: If X **is small then** $Y = 0.1X + 6.4$
R2: If X **is medium then** $Y = -0.5X + 4$
R3: If X **is large then** $Y = X - 2$

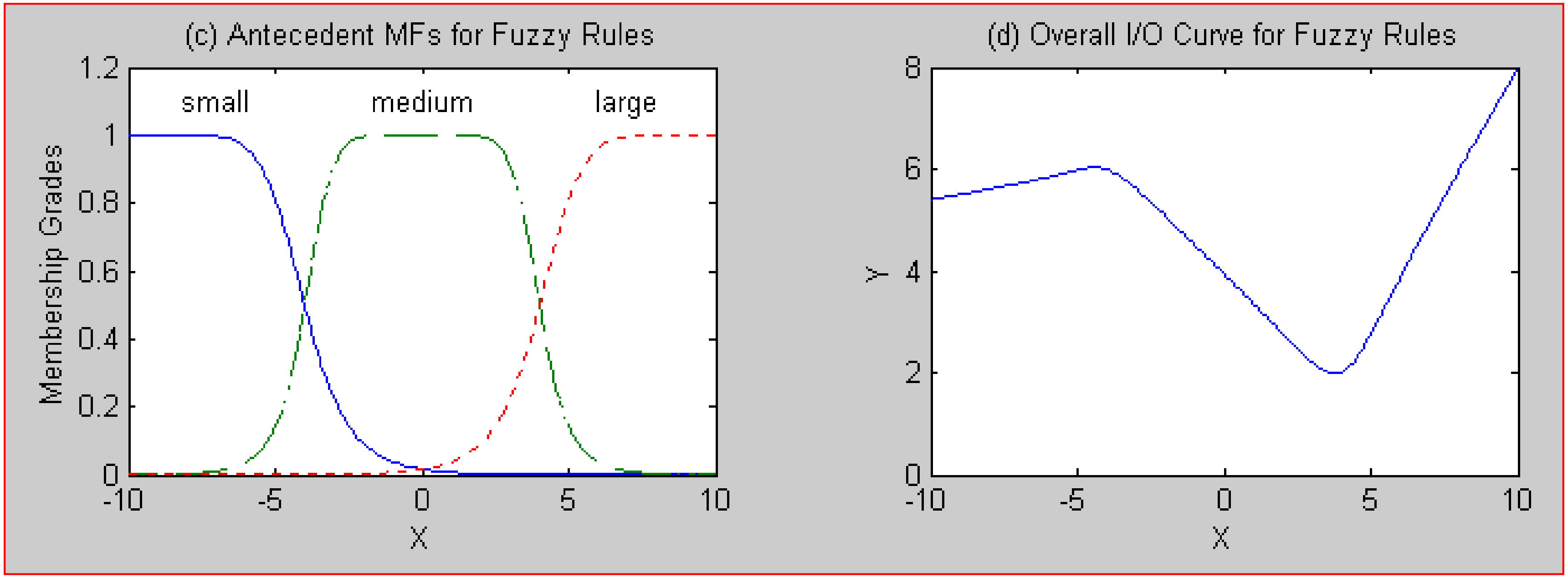


Using **jagged** membership functions, the overall input-output curve produced contains sharp edges.

Sugeno FIS Example

X = input $\in [-10, 10]$

R1: If X **is small then** $Y = 0.1X + 6.4$
R2: If X **is medium then** $Y = -0.5X + 4$
R3: If X **is large then** $Y = X - 2$

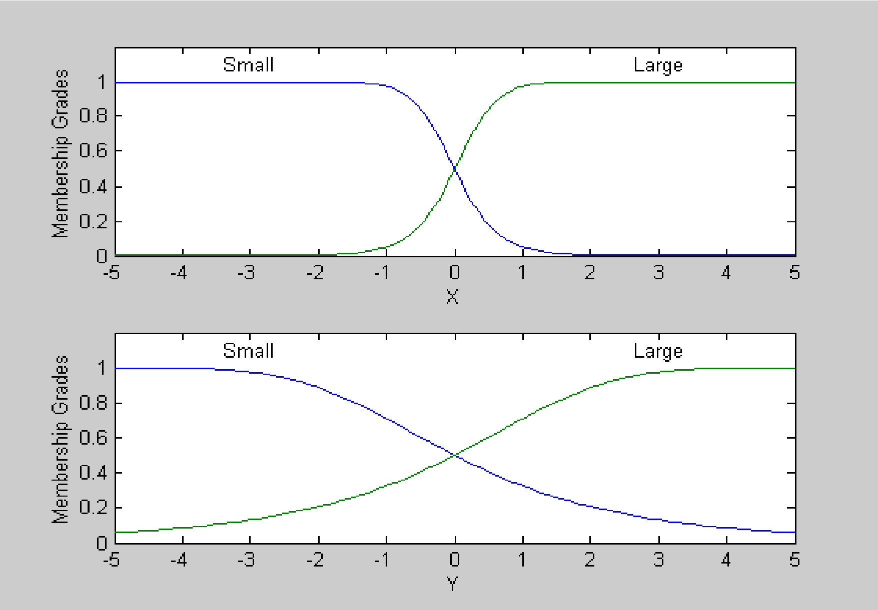


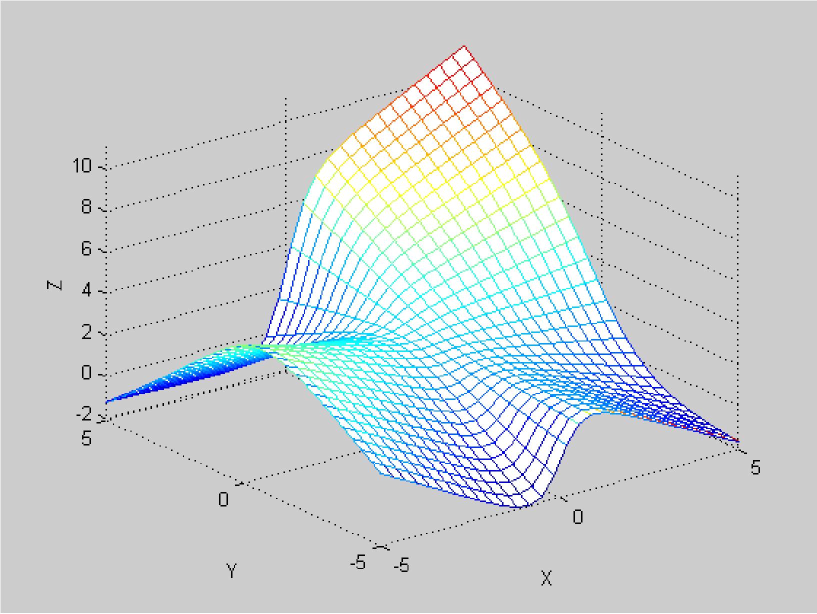
Using **smooth** membership functions, the overall input-output curve produced becomes smoother.

Sugeno FIS Example

R1: if X is small and Y is small then $z = -x + y + 1$
R2: if X is small and Y is large then $z = -y + 3$
R3: if X is large and Y is small then $z = -x + 3$
R4: if X is large and Y is large then $z = x + y + 2$

$X, Y \in [-5, 5]$





Highly non-linear problems could be solved.

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.

back

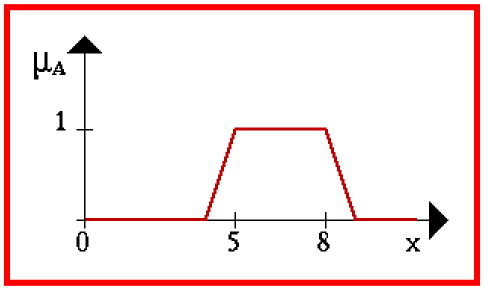
menu

Properties of Fuzzy Sets

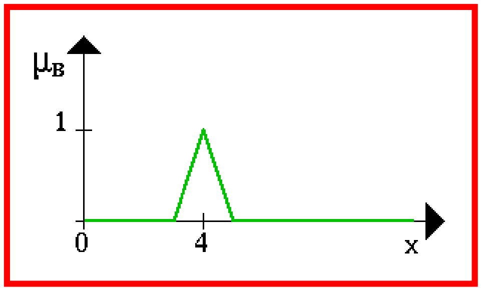
Fuzzy Sets

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

Examples:



Fuzzy Set **A**



Fuzzy Set **B**

back

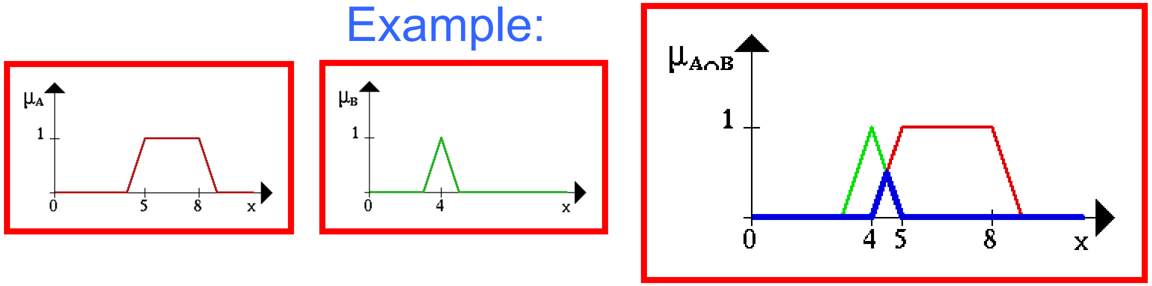
menu

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 intersections** 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 And - min($\mu_A(x)$, $\mu_B(x)$)

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

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

back

menu

Fuzzy “or”

Fuzzy “or”

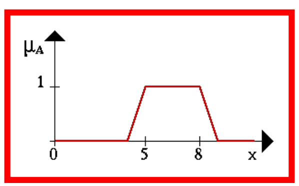
The fuzzy “or” is written as:

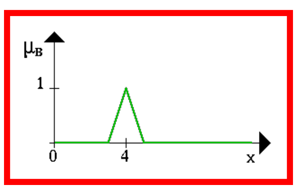
$$u_{A\cup B} = T(u_A(x), u_B(x))$$

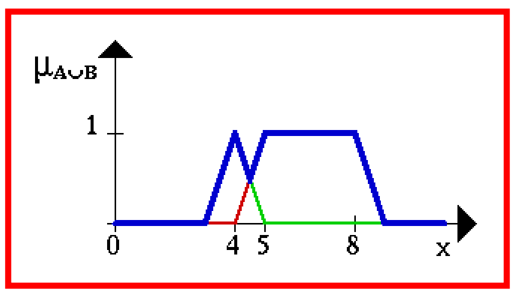
where μ_A is read as “the membership in class A” and μ_B is read as “the membership in class B”.

Fuzzy unions are also referred to as “**T-conorms**” or “**S-norms**”.

Example:







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

back
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

Fuzzy "or"

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

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

Zadeh OR

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

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".

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**":

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

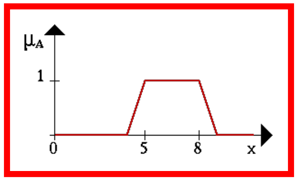
Einstein Disjunction

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

Fuzzy "not"

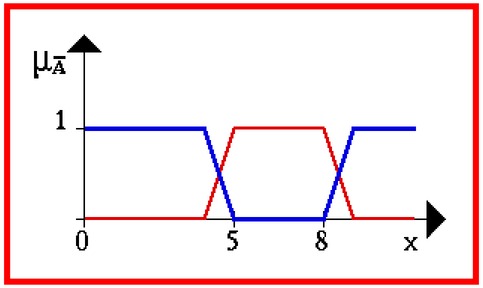
NOT (A) = 1 - A

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**

Frequently used properties

Define three fuzzy sets $\underset{\sim}{A}$, $\underset{\sim}{B}$, and $\underset{\sim}{C}$ on the universe X.

- *commutativity*

$$\underset{\sim}{A} \cup \underset{\sim}{B} = \underset{\sim}{B} \cup \underset{\sim}{A}$$

$$\underset{\sim}{A} \cap \underset{\sim}{B} = \underset{\sim}{B} \cap \underset{\sim}{A}$$

- *associativity*:

$$\underset{\sim}{A} \cup \left(\underset{\sim}{B} \cup \underset{\sim}{C}\right) = (\underset{\sim}{A} \cup \underset{\sim}{B}) \cup \underset{\sim}{C}$$

$$\underset{\sim}{A} \cap \left(\underset{\sim}{B} \cap \underset{\sim}{C}\right) = (\underset{\sim}{A} \cap \underset{\sim}{B}) \cap \underset{\sim}{C}$$

- *distributivity*:

$$\underset{\sim}{A} \cup \left(\underset{\sim}{B} \cap \underset{\sim}{C}\right) = (\underset{\sim}{A} \cup \underset{\sim}{B}) \cap \left(\underset{\sim}{A} \cup \underset{\sim}{C}\right)$$

$$\underset{\sim}{A} \cap \left(\underset{\sim}{B} \cup \underset{\sim}{C}\right) = (\underset{\sim}{A} \cap \underset{\sim}{B}) \cup \left(\underset{\sim}{A} \cap \underset{\sim}{C}\right)$$

- *idempotency*:

$$\underset{\sim}{A} \cup \underset{\sim}{A} = \underset{\sim}{A} \quad \text{and} \quad \underset{\sim}{A} \cap \underset{\sim}{A} = \underset{\sim}{A}$$

back

menu

Frequently used properties

Define three fuzzy sets $\underset{\sim}{A}$, $\underset{\sim}{B}$, and $\underset{\sim}{C}$ on the universe X.

- *identity*

$$\underset{\sim}{A} \cup \emptyset = \underset{\sim}{A} \quad \text{and} \quad \underset{\sim}{A} \cap X = \underset{\sim}{A}$$

$$\underset{\sim}{A} \cap \emptyset = \emptyset \quad \text{and} \quad \underset{\sim}{A} \cup X = X$$

- *transitivity*:

$$\text{If } \underset{\sim}{A} \subseteq \underset{\sim}{B} \text{ and } \underset{\sim}{B} \subseteq \underset{\sim}{C}, \text{ then } \underset{\sim}{A} \subseteq \underset{\sim}{C}$$

$A \subset B$	$\Rightarrow$	A is fully contained in B (if $x \in A$, then $x \in B$)
$A \subseteq B$	$\Rightarrow$	A is contained in or is equivalent to B
$(A \leftrightarrow B)$	$\Rightarrow$	$A \subseteq B$ and $B \subseteq A$ (A is equivalent to B)

- *involution*:

$$\overline{\overline{\underset{\sim}{A}}} = \underset{\sim}{A}$$

back

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**.

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

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

for any proposition, either that proposition is true or its negation is true

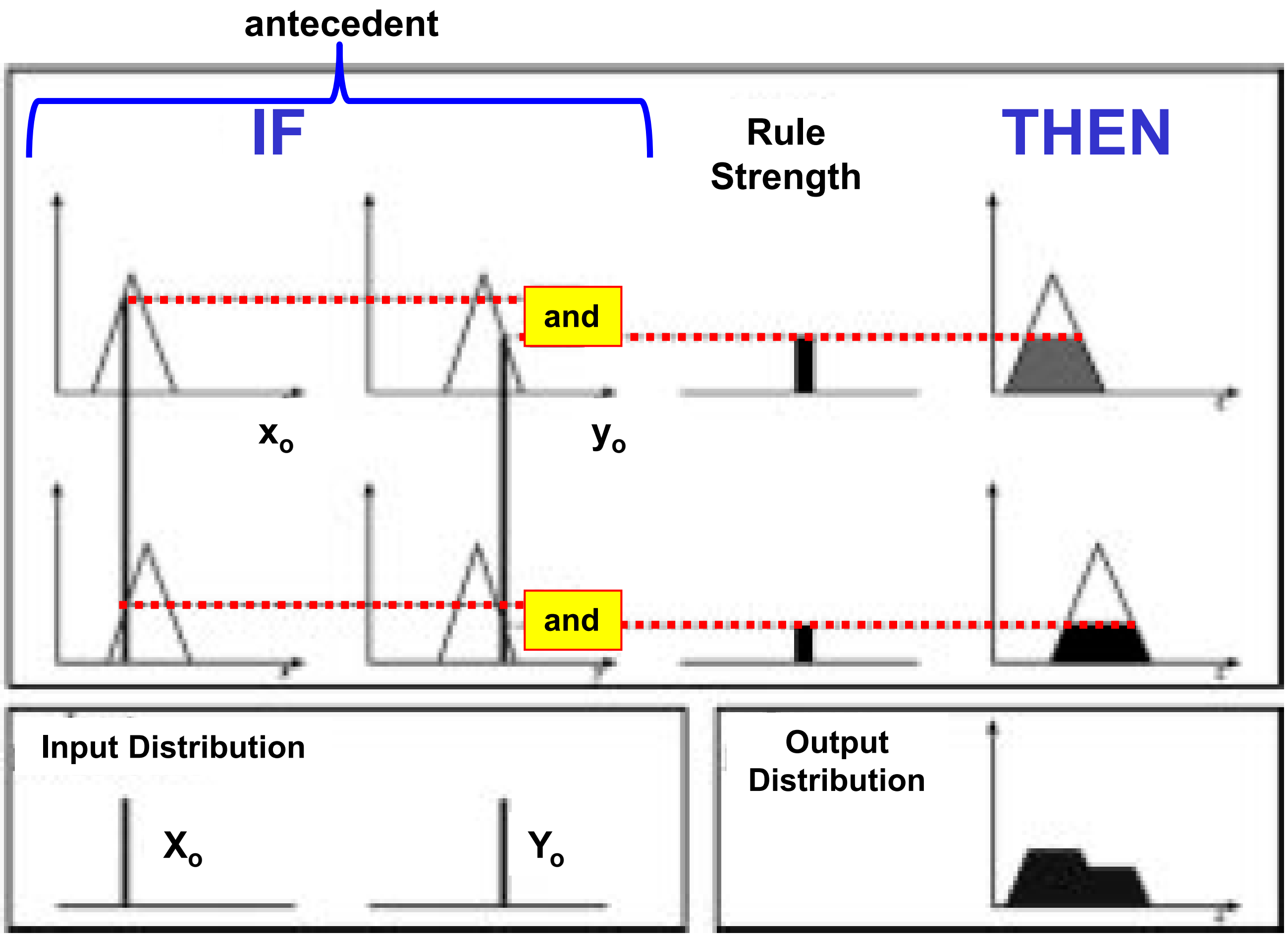
The **whole set X** is the set of all elements in the universe

Variations in computing for the rule consequence

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

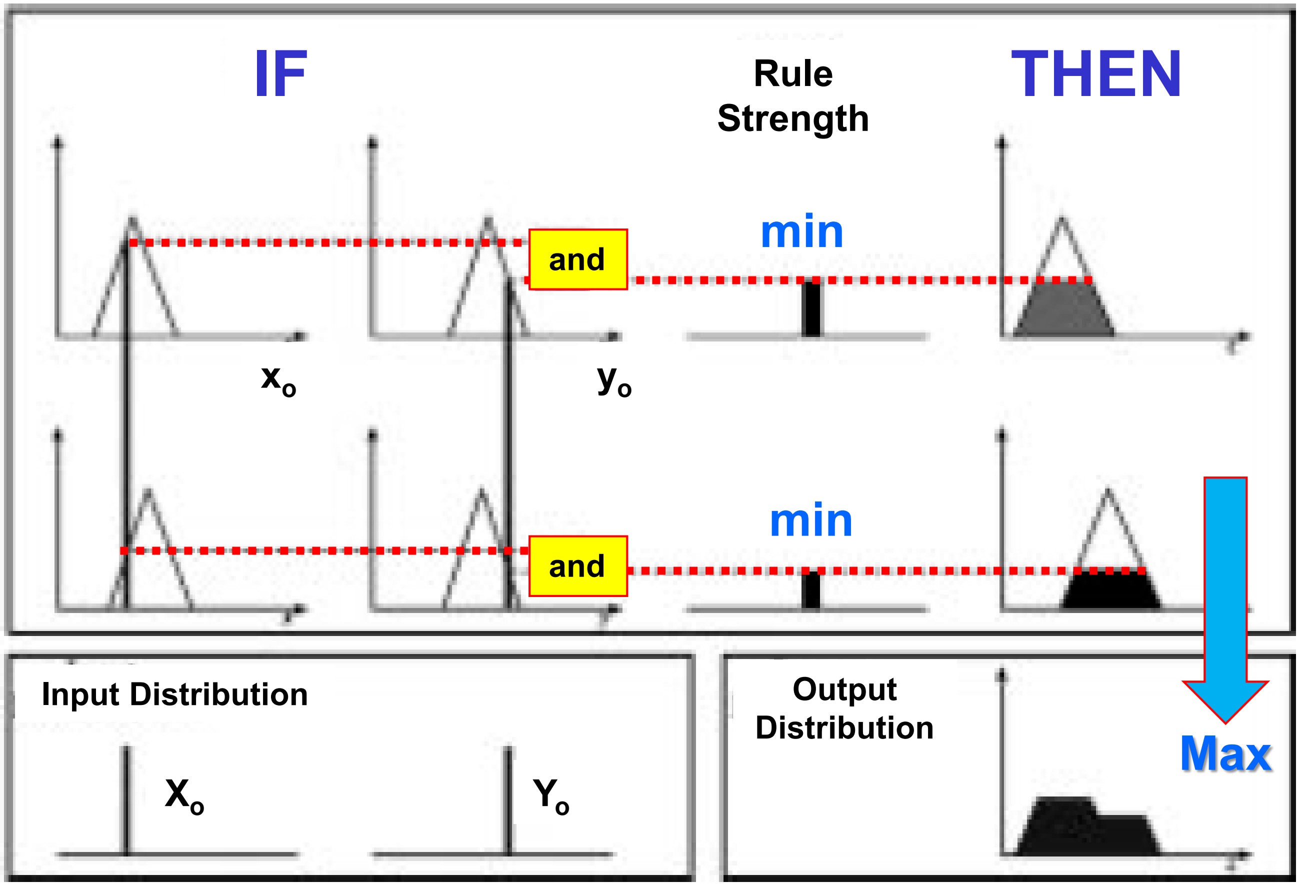


In this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

Consequence

2a Max-Min Composition

Clipping the output membership function at the rule strength.

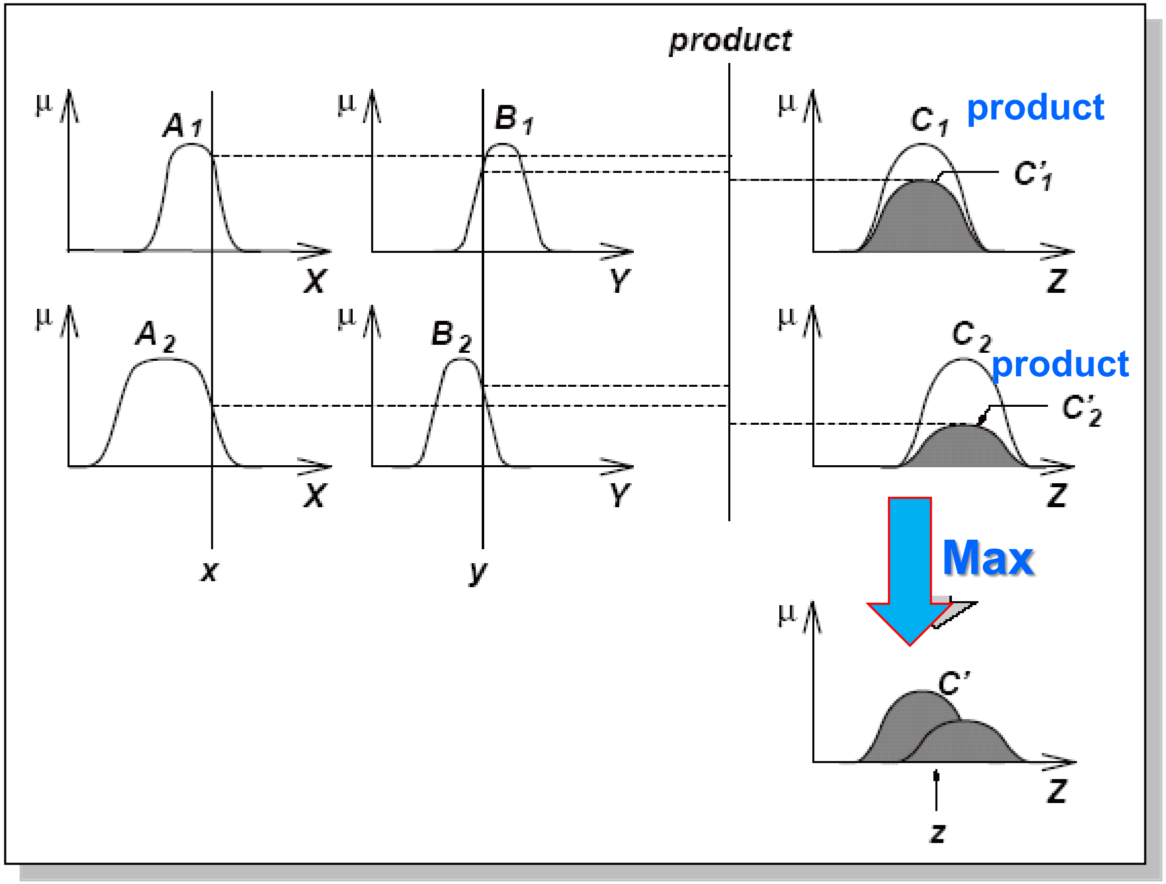


In this example, the fuzzy "and" is used to combine the membership functions to compute the rule strength.

Consequence

2b **(Alternatively) Max-Product Composition**

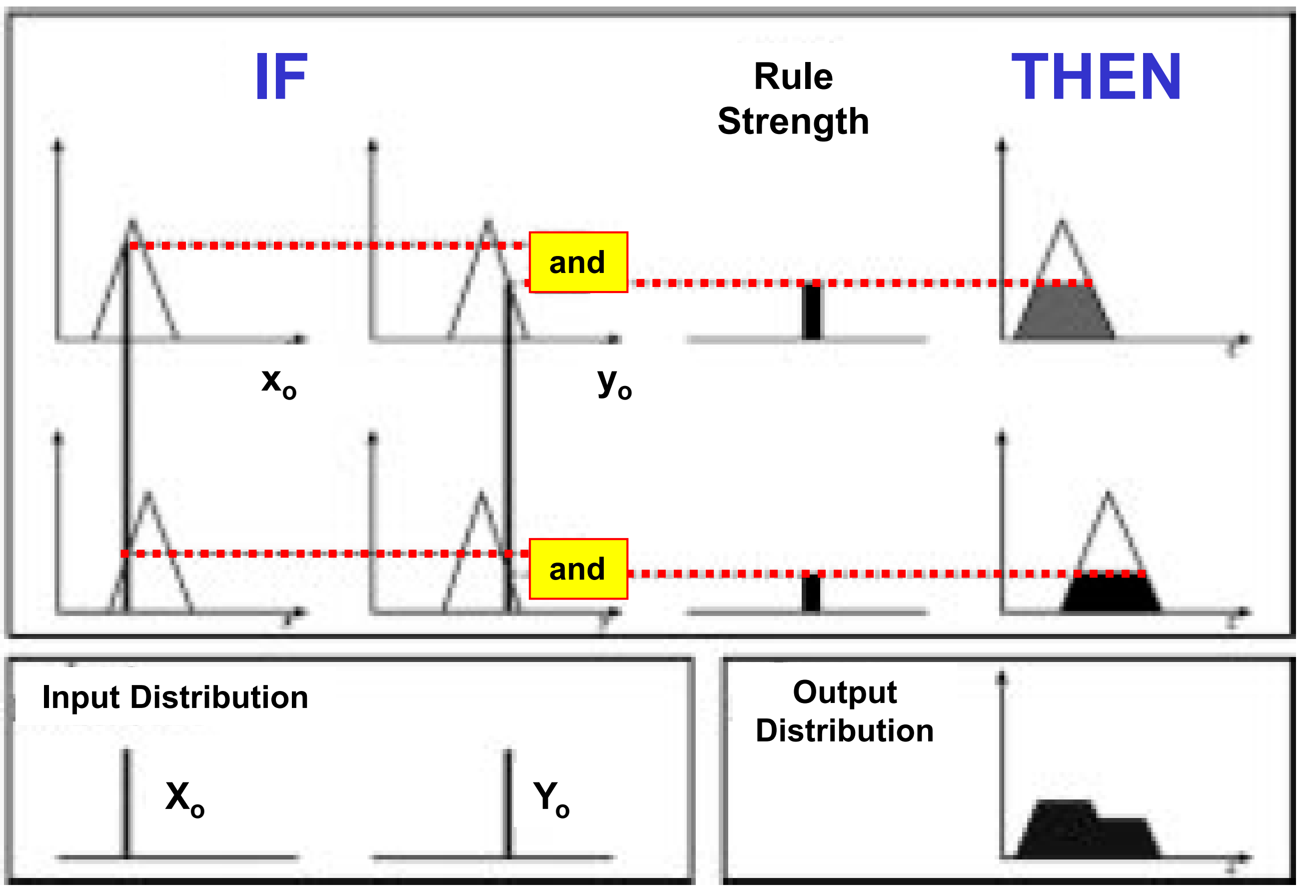
Multiplying the output membership function by the rule strength.



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

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.



Aggregation Method

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.

Defuzzification techniques

back

menu

3 Defuzzification of Output Distribution

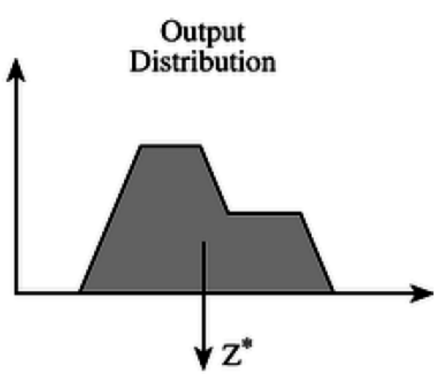
In many instances, it is desired to come up with a single crisp output from a FIS. For example, if one was trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIS would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained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

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

a) Center of mass - 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:

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

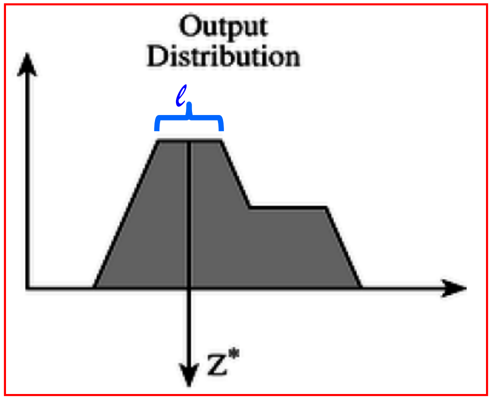
where z is the center of mass and μ_c is the membership in class c at value Z_j. An example outcome of this computation is shown in the figure at the right.



Defuzzification of Output Distribution

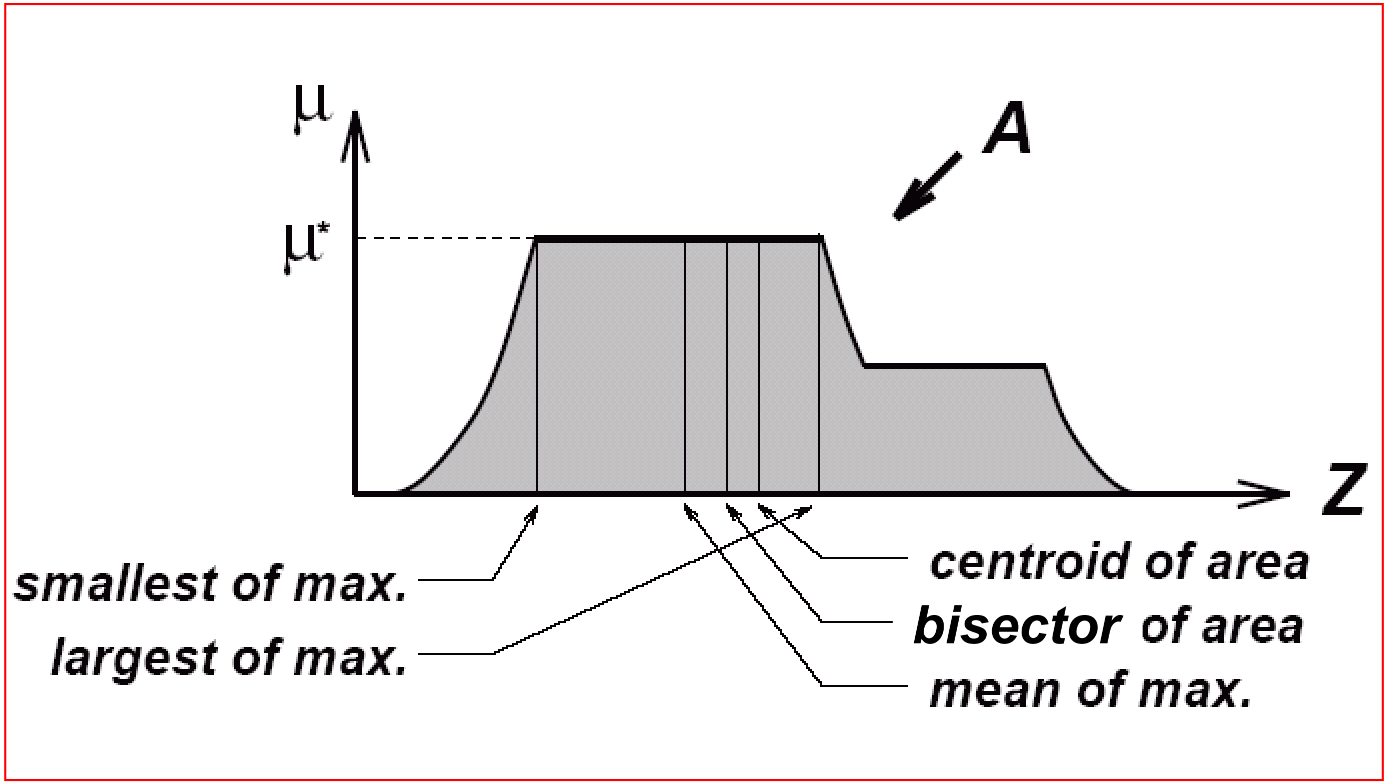
b) **Mean of maximum** - This technique takes the output distribution found in the previous section and finds its mean of maxima to come up with one crisp number. This is computed as follows:

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



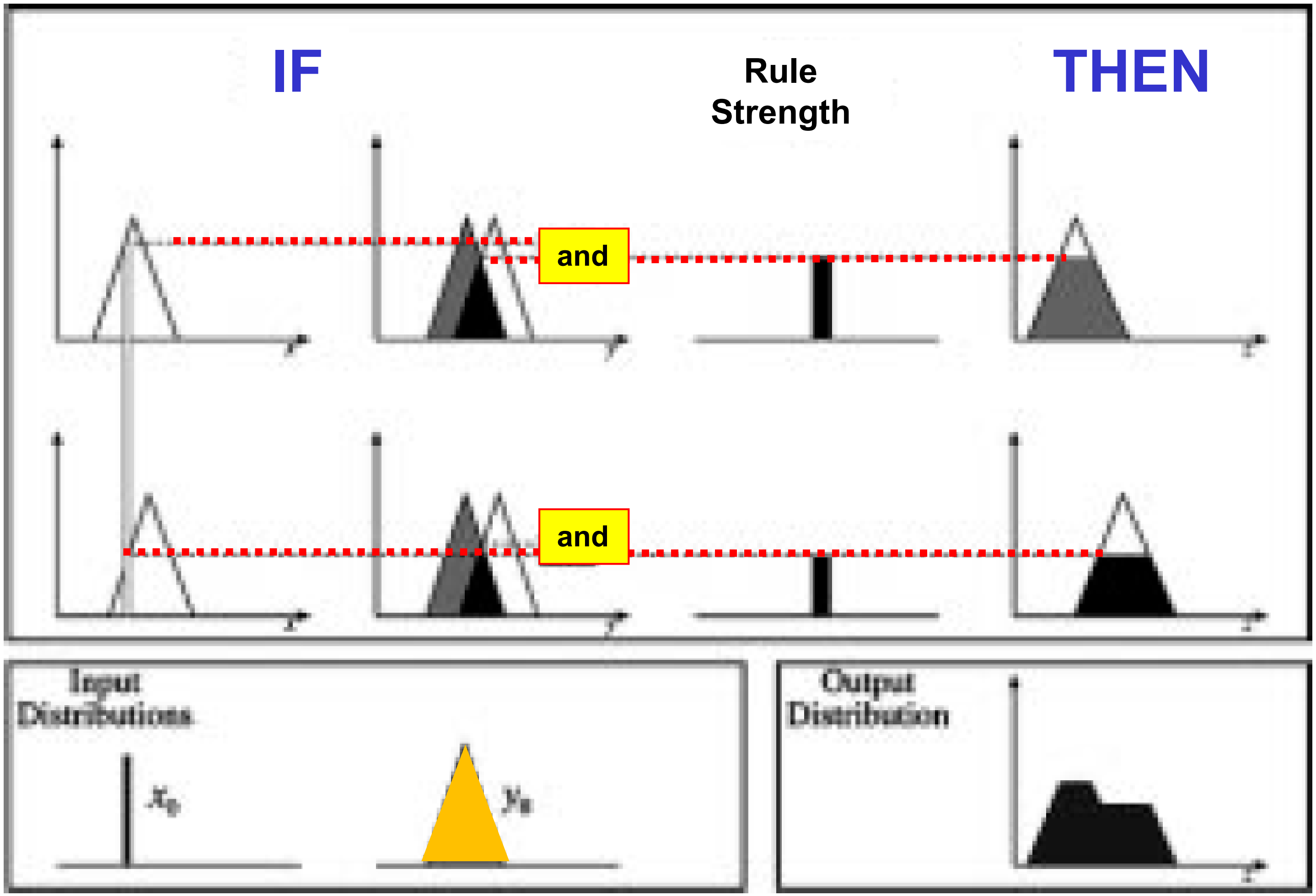
where **z** is the mean of maximum, Z_j is the point at which the membership function is maximum, and ℓ is the number of times the output distribution reaches the maximum level.
An example outcome of this computation is shown on the figure at the right.

Other Defuzzification Methods



Mamdani FIS with a Fuzzy Input

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



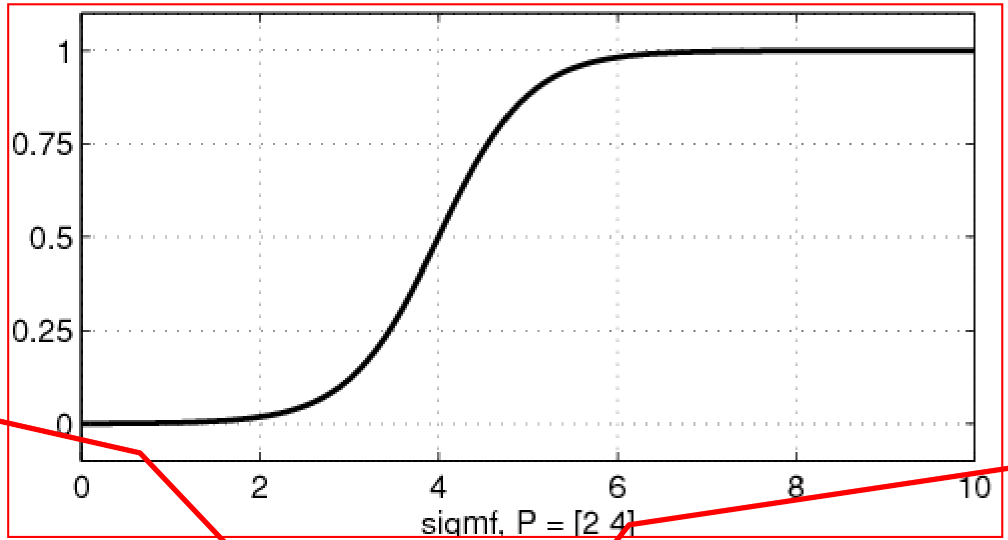
shows a modification of the Mamdani FIS where the input **y0** is **fuzzy**, not crisp. This can be used to model inaccuracies in the measurement. For example, we may be measuring the output of a pressure sensor. Even with the exact same pressure applied, the sensor is measured to have slightly different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

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

back

menu

Membership Functions

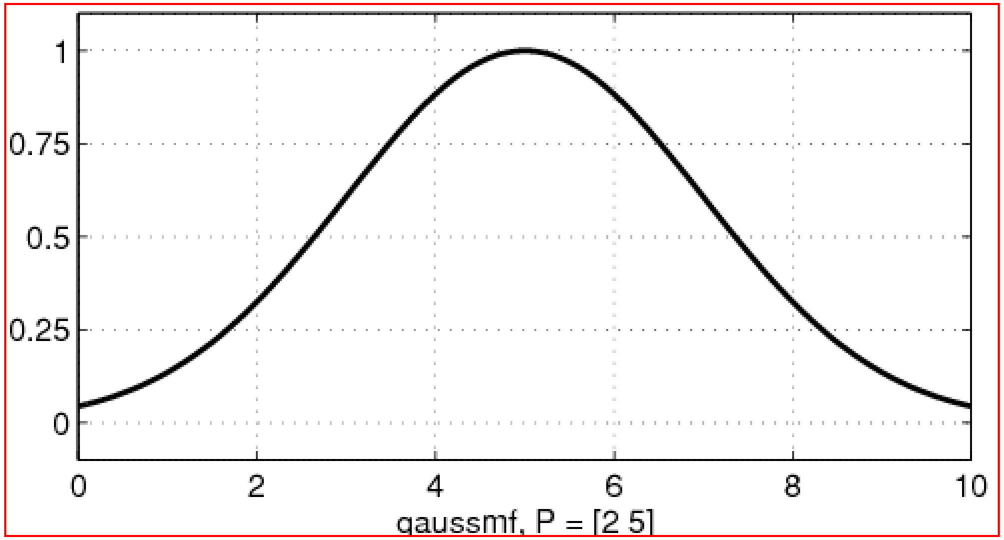
The Gaussian function

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

centre

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

width



back

menu

Membership Functions

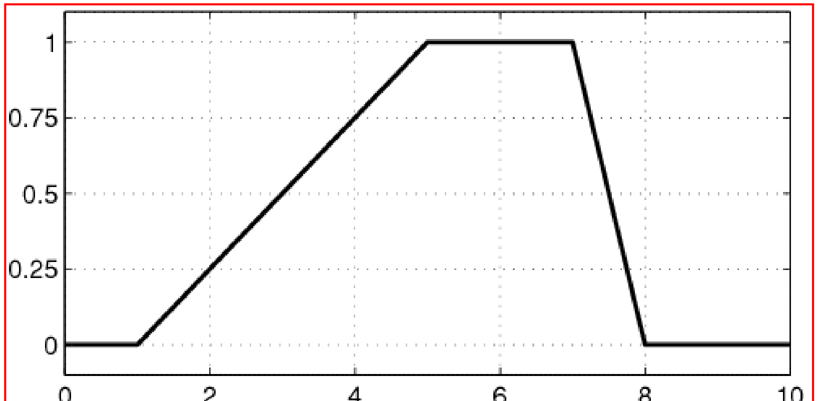
The Trapezoidal function

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

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

or

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

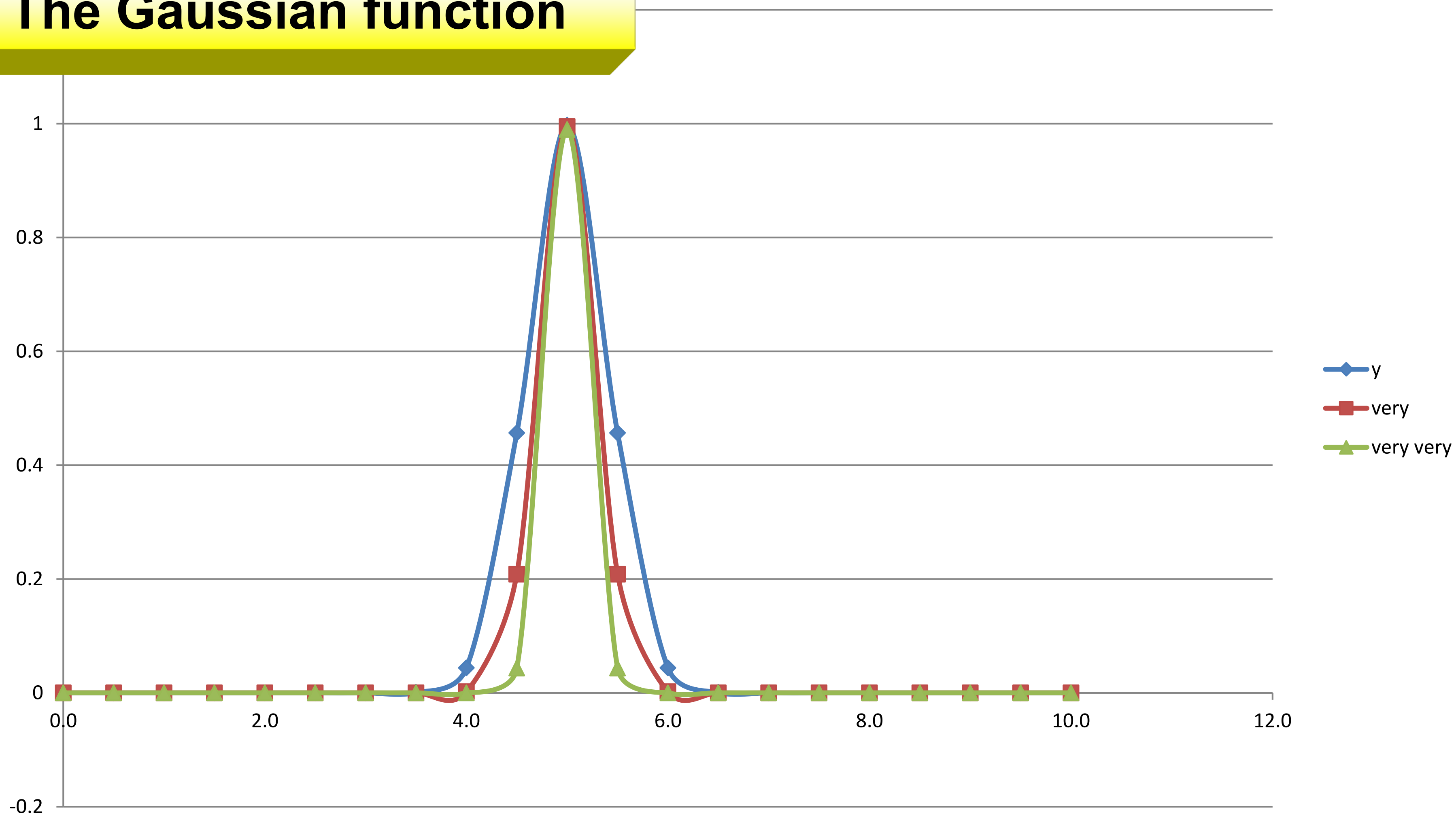


trapmf, P = [1 5 7 8]

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

Fuzzy Hedges

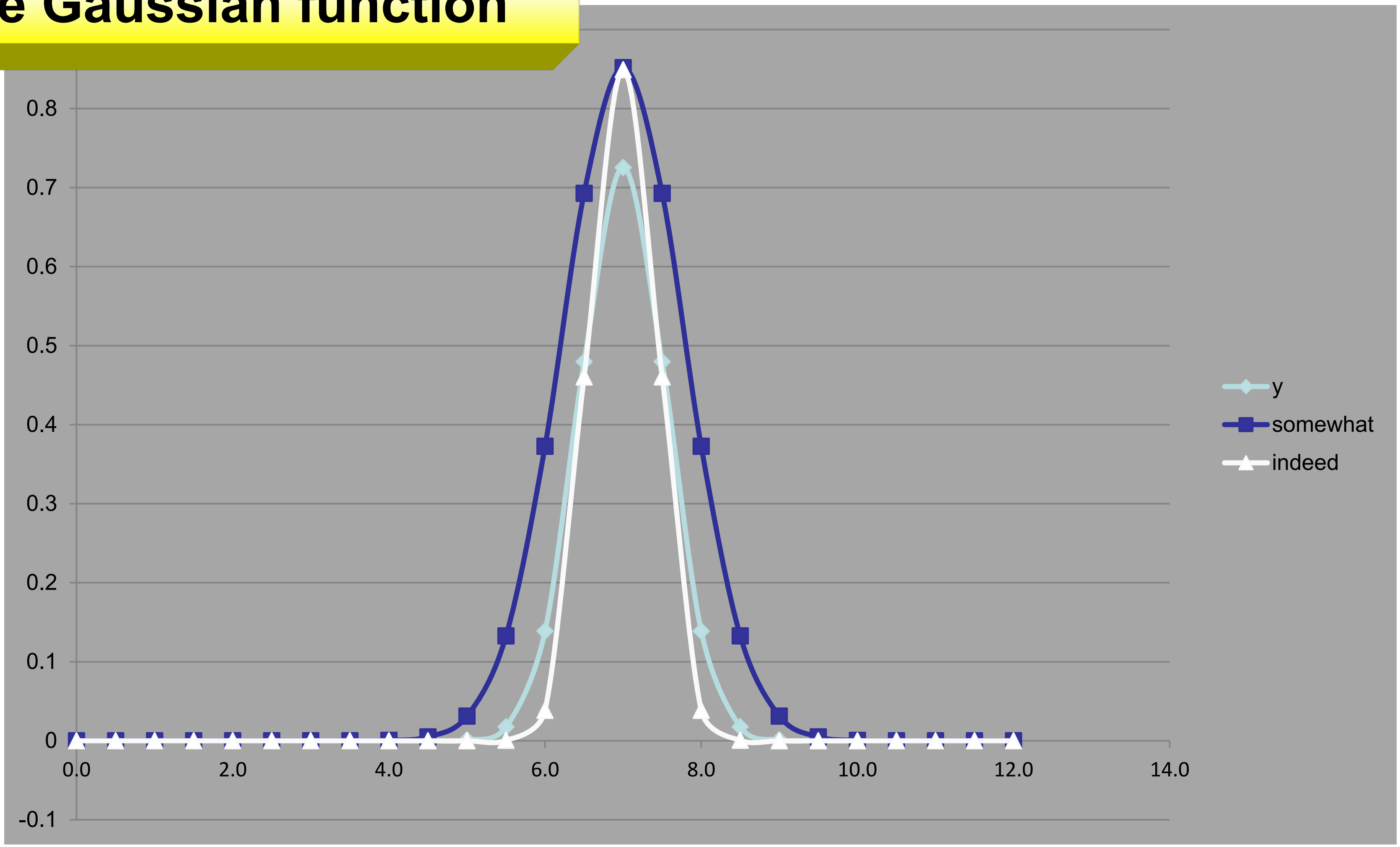
The Gaussian function



sigma	0.4
c	5

Fuzzy Hedges

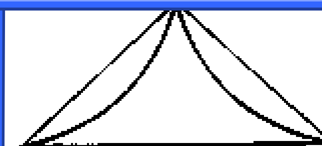
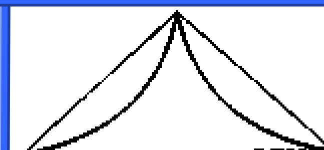
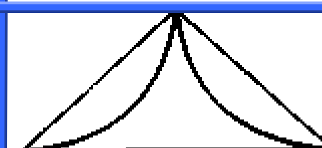
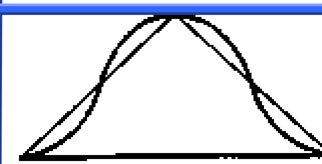
The Gaussian function



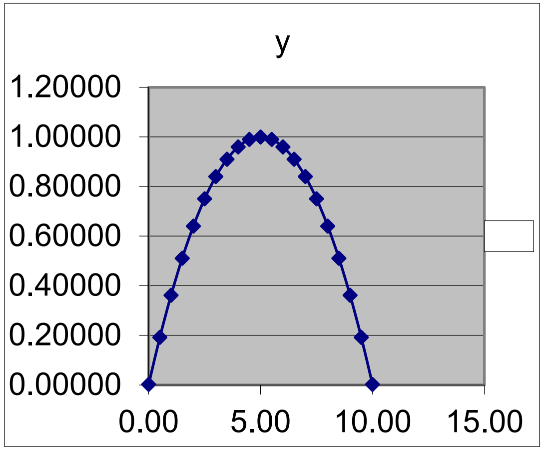
sigma	0.55
c	7

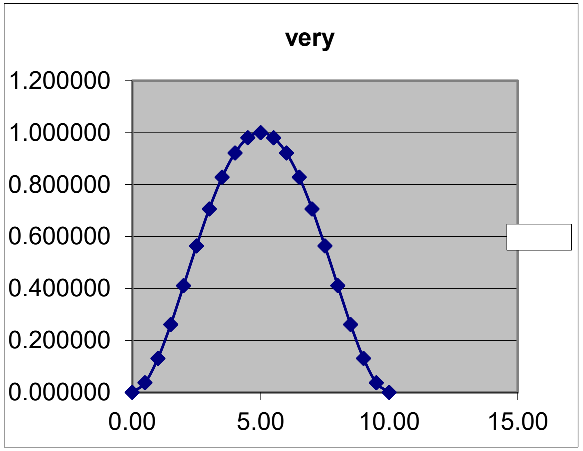
Fuzzy Hedges

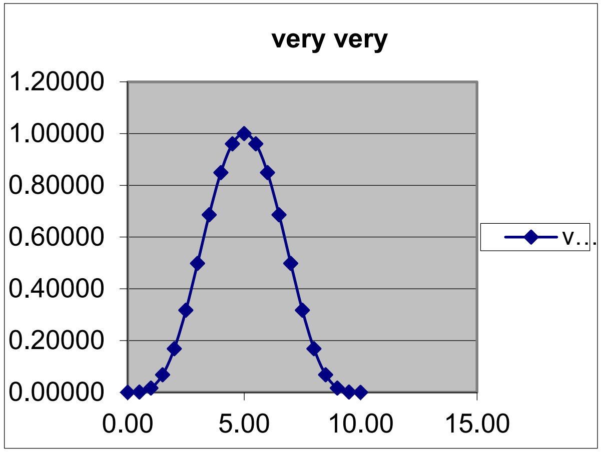
http://blog.peltarion.com/2006/10/25/fuzzy-math-part-1-the-theory/

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	$2\mu_A(x)^2 \quad \text{if } 0 \le \mu_A(x) \le 0.5$ $1-2(1-\mu_A(x))^2 \quad \text{if } 0.5 < \mu_A(x) \le 1$	

More Examples







Sigma	3.54
c	5

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

The End.

back

menu

U.S. Patent 2015

(h) automobile (see FIG. **101**, e.g. measuring environmental parameters, to adjust braking system in different driving conditions),

p.235, Fuzzy logic applications

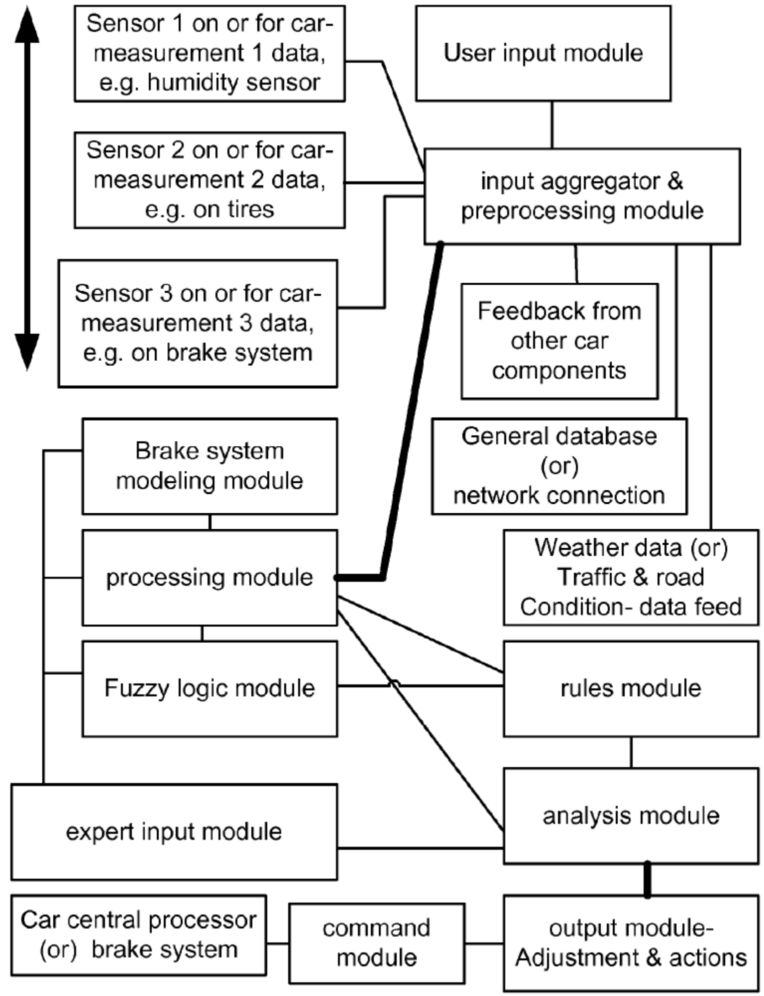


Fig. 101

U.S. Patent 2015

(i) control systems and autonomous systems (see FIG. **102**, e.g. for driving a car autonomously, without a driver),

p.235, Fuzzy logic applications

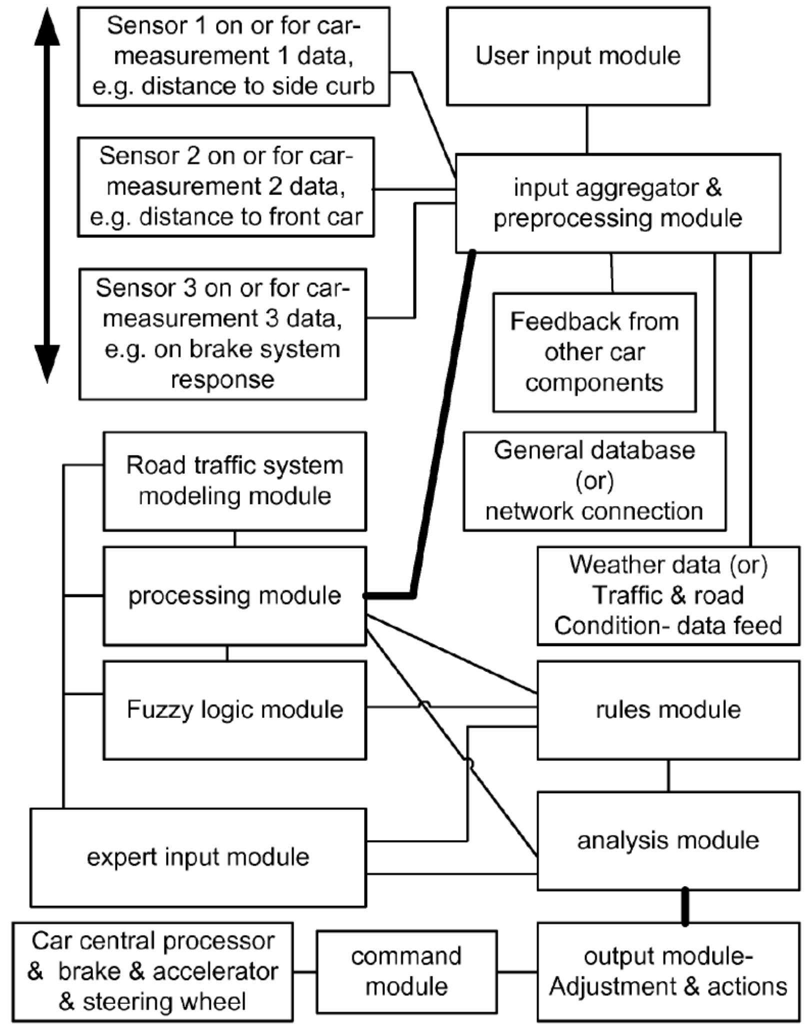


Fig. 102

U.S. Patent 2015

p.197

In one embodiment, e.g., in a car engine diagnosis, the following natural language rule "Usually, when engine makes rattling slapping sound, and it gets significantly louder or faster when revving the engine, the timing chain is loose." is converted to a protoform, such as:

$$\text{IF}\left(\begin{array}{c}\text{type(sound(engine)) is } \mathit{RattlingSlapping}\\ \text{AND}\\ \left(\begin{array}{c}\text{(level(sound(revved.engine)), level(sound(engine)))}\\ \text{is significantly.louder}\\ \text{OR}\\ \text{(rhythm(sound(revved.engine)), rhythm(sound(engine)))}\\ \text{is significantly.faster}\end{array}\right)\end{array}\right)$$

THEN

(Prob{(tension(*TimingChain*) is loose)} is usually).

back

menu