Lecture 5
Fuzzy logic inference

Harry J. Whitlow
Department of Physics
University of Jyväskylä

Topics

- Fuzzy logic
  - Types of uncertainty
  - linguistic variable

- Fuzzy inference

- Hardware implementation
  - Combination with conventional logic and control.
  - Application to appliances and industrial control.

- Neural and learning approaches
  - Application in signal processors

Lost in translation......

- Fuzzy logic method first originated in the UK in 1970
- Japanese engineers in 80's developed the technique for industrial control
- In Japanese the word for ”***fuzzy logic***” *"juunan na shihai” or "juunan na sousa".* means flexible or expert control
- Sadly the word ”*fuzzy*” in English (The Japanese word is "*aimaina*”) means inprecise or diffuse giving rise to ”marketing” as *continious logic* or *multivalued logic*.
- The technique mimics an expert controling some plant
- **Fuzzy control is not inprecise** – it takes crisp inputs, **r**(*t*) and produces crisp outputs **c**(*t*).

Types of uncertainty: stochastic uncertainty

- Consider the statement:
  " Collectively, though not separately, these studies show appreciable hazards from residential radon, particularly for smokers and recent ex-smokers, and indicate that it is responsible for about 2% of all deaths from cancer in Europe." ***Radon in homes and risk of lung cancer: collaborative analysis of individual data from 13 European case-control studies*** S Darby, et al BMJ, doi:10.1136/bmj.38308.477650.63 (published 21 December 2004)

- This is a **stochastic uncertainty**.
  - The event: ***death*** is well defined
  - Even if a person has radon-induced lung cancer but gets killed in a traffic accident does not count to the 2%.
  - ***The uncertainty*** is if a particular person dies of radon-induced lung cancer or not.
  - **The stochastic probability** *"about 2%"* defines rather precisely (between 1 and 4%) how many people in a large population will die of lung cancer from radon exposure.

Types of uncertainty: linguistic uncertainty

- Consider the statement ”*We will probably have a succesful financial year.”*

- Subjective statement
  - If the company is doing well and stable it may increase its profits by 5%, 25%, 100%?

- No quantification of probablility
  - Probability is perceived property

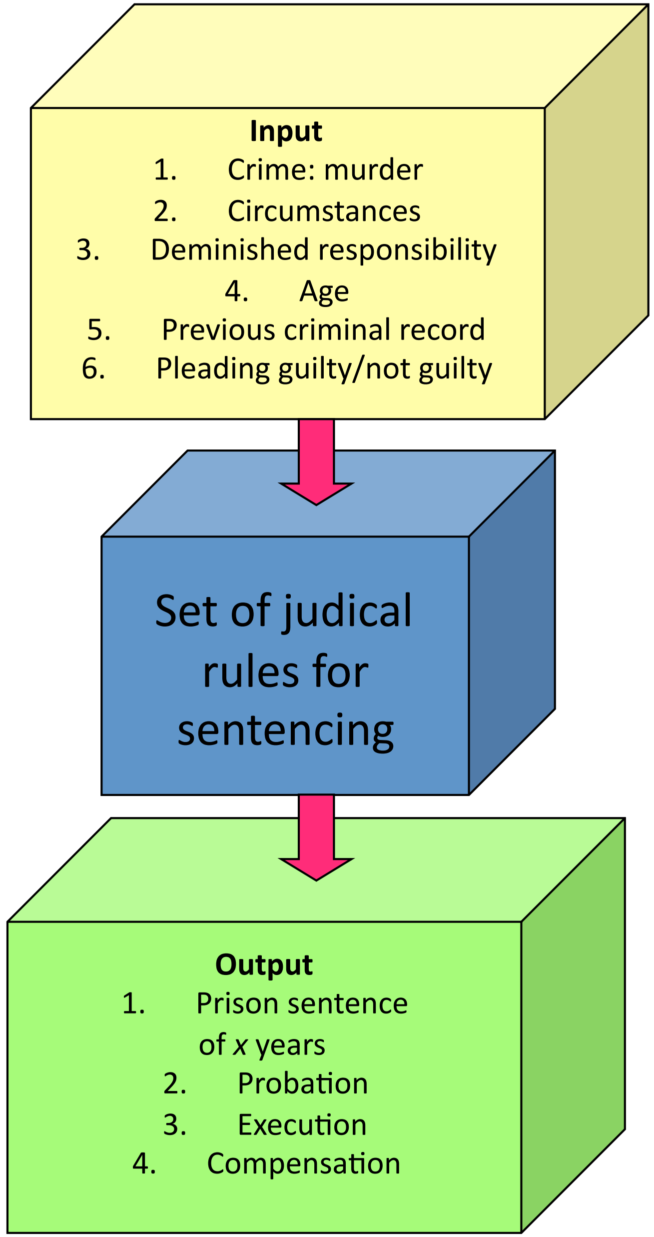
- Poorly defined event
  - If the company is on the brink of bankruptcy then it means they just might not become bankrupt this FY.

Linguistic uncertainty in human decison making

- Statements do not have quantitative connections yet *expert* humans use these quite *succesfully* for complex decision making.

- Based on a set of rules e.g. judicial system

Note: linguistic uncertainty



Fuzzy logic and probability theory

- Patients with hepetitus type-B show in 60% of cases a high fever, in 40% of cases yellowish skin and 30% of cases nausea.

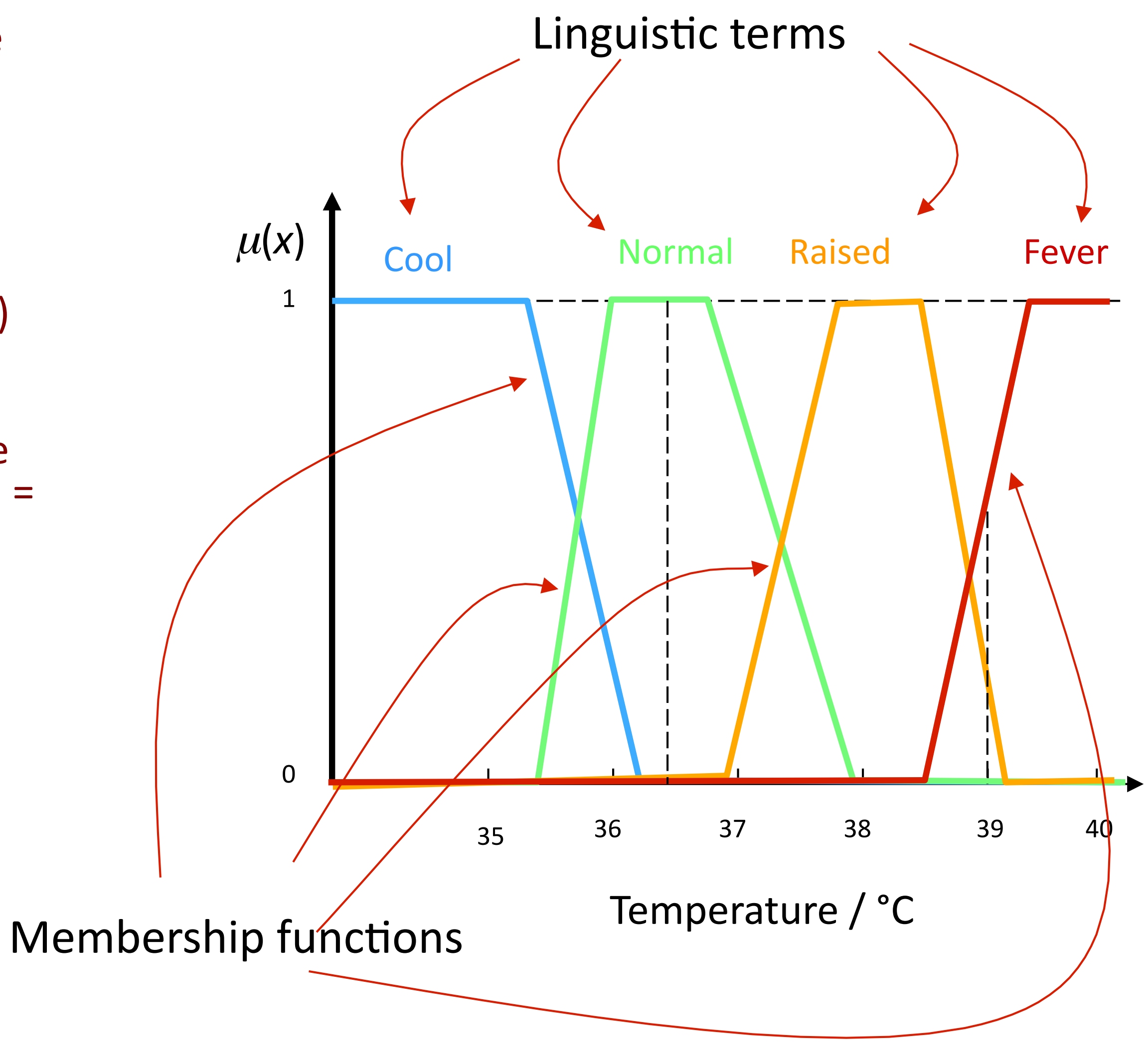
- Problem is defintion of high fever.
  - Typically this is taken to mean 39 °C
  - Patient with 38.9 may have a high fever while with 39.1 °C may not have high fever
  - Other factors may decide e.g. comparison with prototypes e.g. pale, sweating shivering or well-tempered patient

- Probability theory determines if an event will take place

- Lexical probability deals with uncertanty in the event itself.

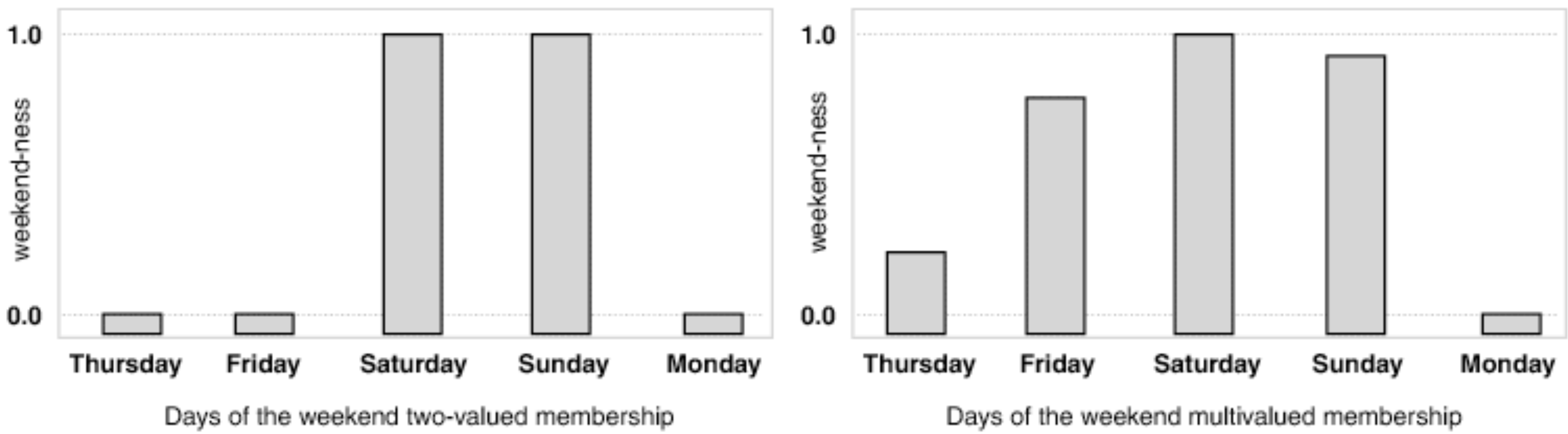
Linguistic variables

- The variable is assigned a degree of membership $\mu(x)$ to each set.
- $x \in$ {cool,normal,raised,fever}
- The membership of 36.4 °C to liguistic value "*normal*" is $\mu(36.4)$ =1
- 39 °C has membership to degree to "Fever" $\mu(39)$ =0.65 and $\mu(39)$ = 0.35 to "Raised"
- The *linguistic value* is for 39 °C then:
  Raised (0.35) Fever (0.65)



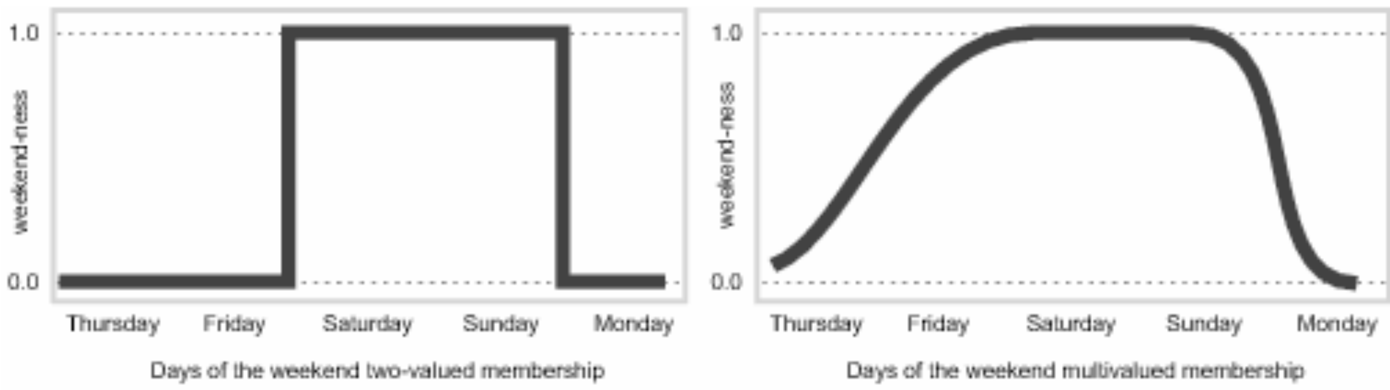
Liguistic variables

- Note that the liguistic variable is defined by:
  - Membership functions for each linguistic term
  - The degree of membership for each linguistic term
  - The lingusitic variable may either be transformed back to a precise input value or a range of input values.



From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

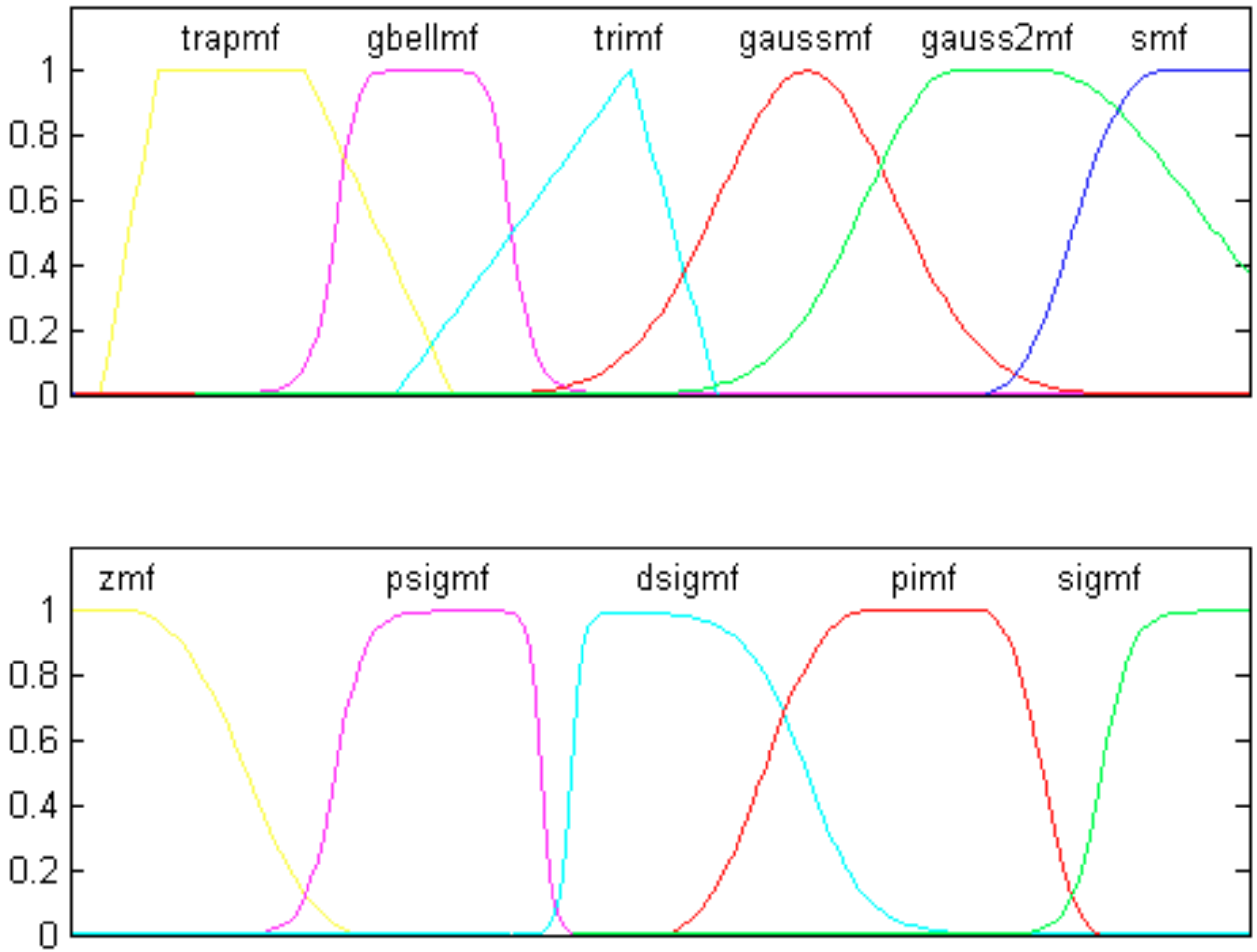
The linguistic variable: “weekendness”



From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

Membership functions

- Mer



Membership functions in Matlab

From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

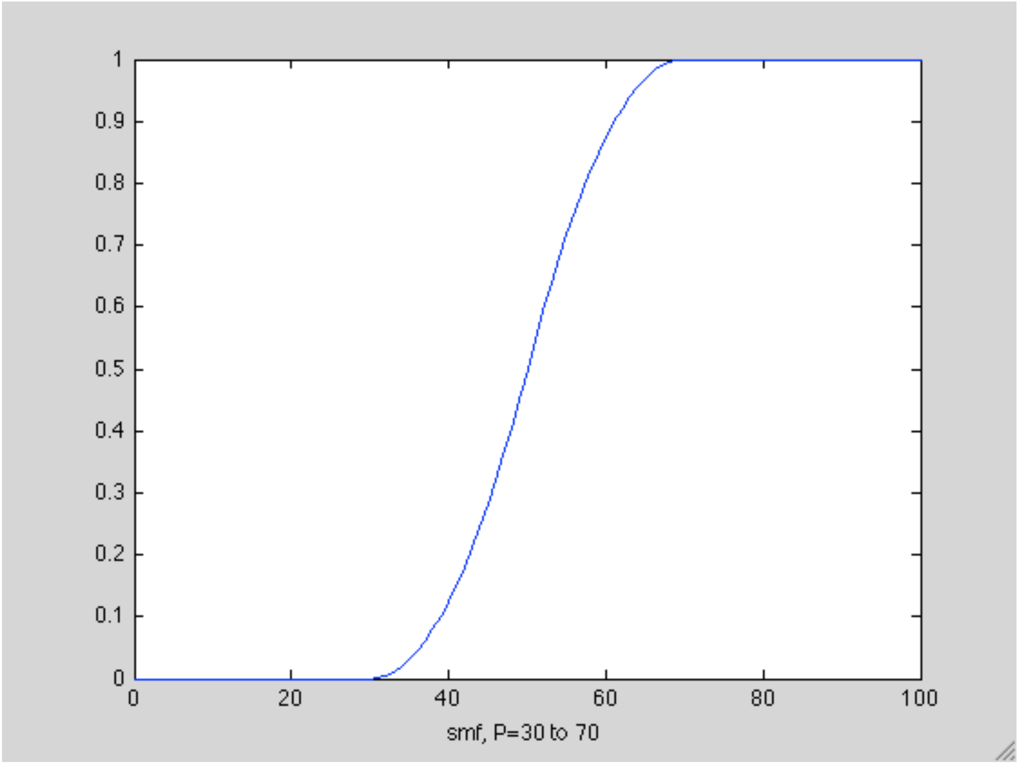
Membership functions in Matlab

- GUI interface provided in Matlab
- Command line (easy!)

- Exmaple “S-shaped function”
- Simple spline function:

$$f(x;a,b)=\begin{cases}0, & x\le a\\ 2\left(\frac{x-a}{b-a}\right)^2, & a\le x\le \frac{a+b}{2}\\ 1-2\left(\frac{x-b}{b-a}\right)^2, & \frac{a+b}{2}\le x\le b\\ 1, & x\ge b\end{cases}$$

```
>> x=0:1:100;
>> y=smf(x,[30 70]);
>> plot(x,y)
>> xlabel('smf, P=30 to 70');
```



From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

Linguistic control strategy

- Most fuzzy control uses rule-based fuzzy sets for control
- Consider the case of a container crane

Container crane control

- Containers are lifted and moved on the end of flexible wires from the crane head
- Crane head moves along a track
- Swaying of the container prevents a container from being released
- Operations should be as fast as possible
- Stratergies
  - Move so slowly no sway occurs
  - Position over the target and wait for sway to decay away

Control

- PID unsuitable as task is non-linear

- 5th degree differential equation for model based controller.
  - Crane motor is inherently non linear
  - Crane head moves with friction – slipping cannto be included
  - Disturbances (wind gusts) not included.

- Humans do it faster!

Operator actions

1. Start with medium power
2. Adjust motor power so container hangs a little behind the crane head
3. As one gets closer to the target reduce motor power so container is a little ahead of crane head
4. When very close increase motor power
5. Stop the motor when container is over the target and sway zero

I am very sorry – I did not mean to put that 32 ton shipping container expertly onto your foot.....

Implementation

1. Start with medium power
2. Adjust motor power so container hangs a little behind the crane head
3. As one gets loser to the target reduce motor power so container is a little ahead of crane head
4. When very close increase motor power
5. Stop the motor when container is over the target and sway zero

1. IF *Distance* = far and *Angle* = zero
   THEN *Power* = pos_medium
2.
   1. IF *Distance* = far and *Angle* = neg_small
      THEN *Power* = pos_big
   2. IF *Distance* = far and *Angle* = neg_big
      THEN *Power* = pos_medium
3. IF *Distance* = medium and *Angle* = neg_small
   THEN *Power* = neg_medium
4. IF *Distance* = close and *Angle* = pos_small
   THEN *Power* = neg_medium
5. IF *Distance* = zero and *Angle* = zero
   THEN *Power* = zero

Fuzzy IF-THEN inference

- Aggregation
  - IF part determines if rule is valid or not
  - Boolean logic
    - Cannot handle sutuations where conditions are more-or-less true
  - Fuzzy logic operators

AND	$\mu_{A \wedge B}$	$\min\{\mu_A, \mu_b\}$
OR	$\mu_{A \vee B}$	$\max\{\mu_A, \mu_b\}$
NOT	$\mu_{\neg A}$	$1-\mu_A$

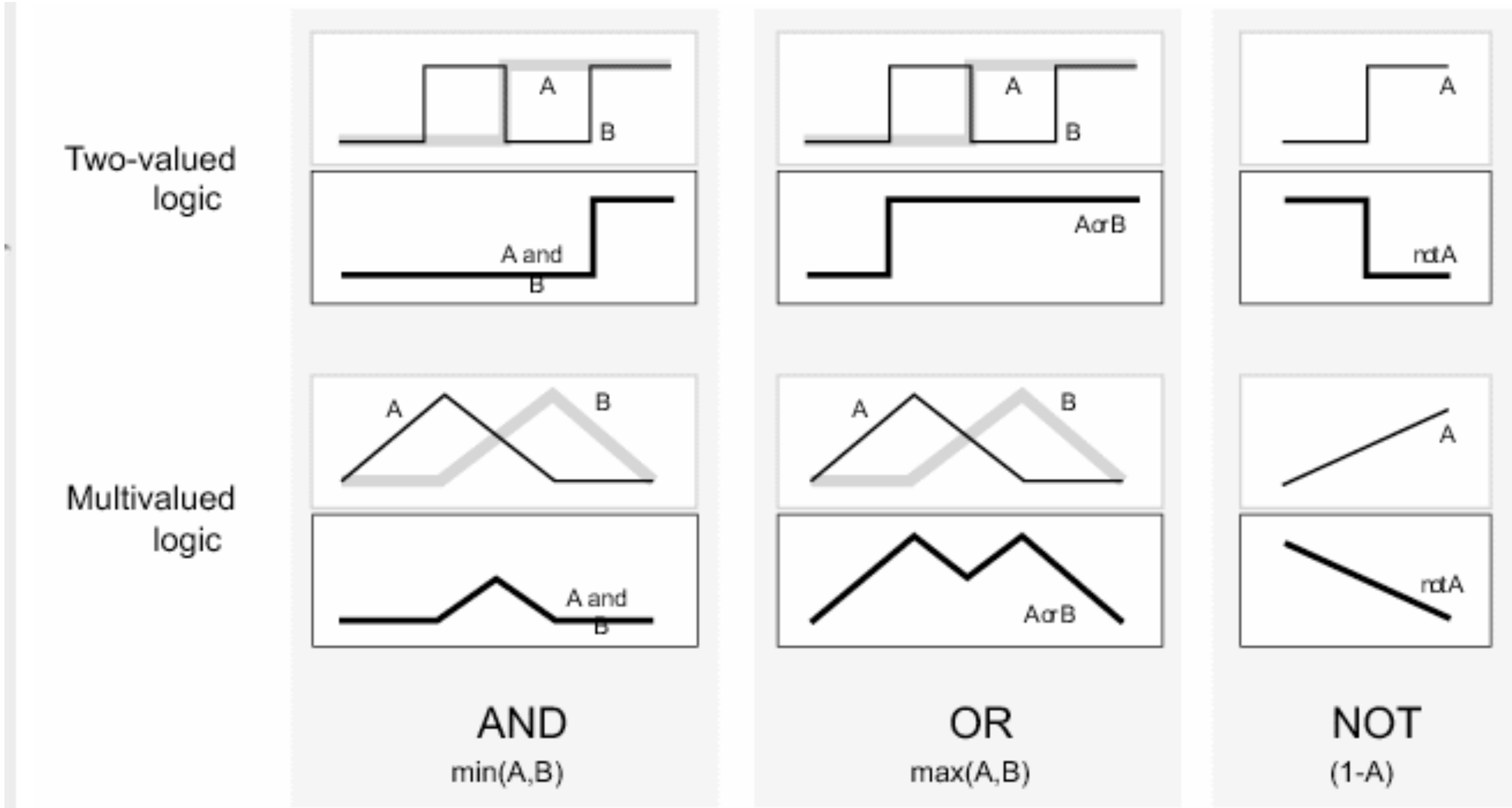
- Composition
  - THEN part defines the action to be taken
  - The degree to which each rule is valid (membership degree) then determines how the action associated with each rule decides the actuating signal after the defuzzification step
  - E.g. If rules 1through 4 corresponding to THENs of high,medium, and zero power are satisified to the degree 0.7, 0.0, 0.3, 0.0 repectively the result will be high power to degree 0.7 + low power to degree 0.3

Fuzzy logic elements in Matlab

- Three functions – that are essentially multivalue AND, OR and NOT operations

max(A,B)
min(A,B)
not((A,B)

- These are sufficient to implement all fuzzy logical inference.

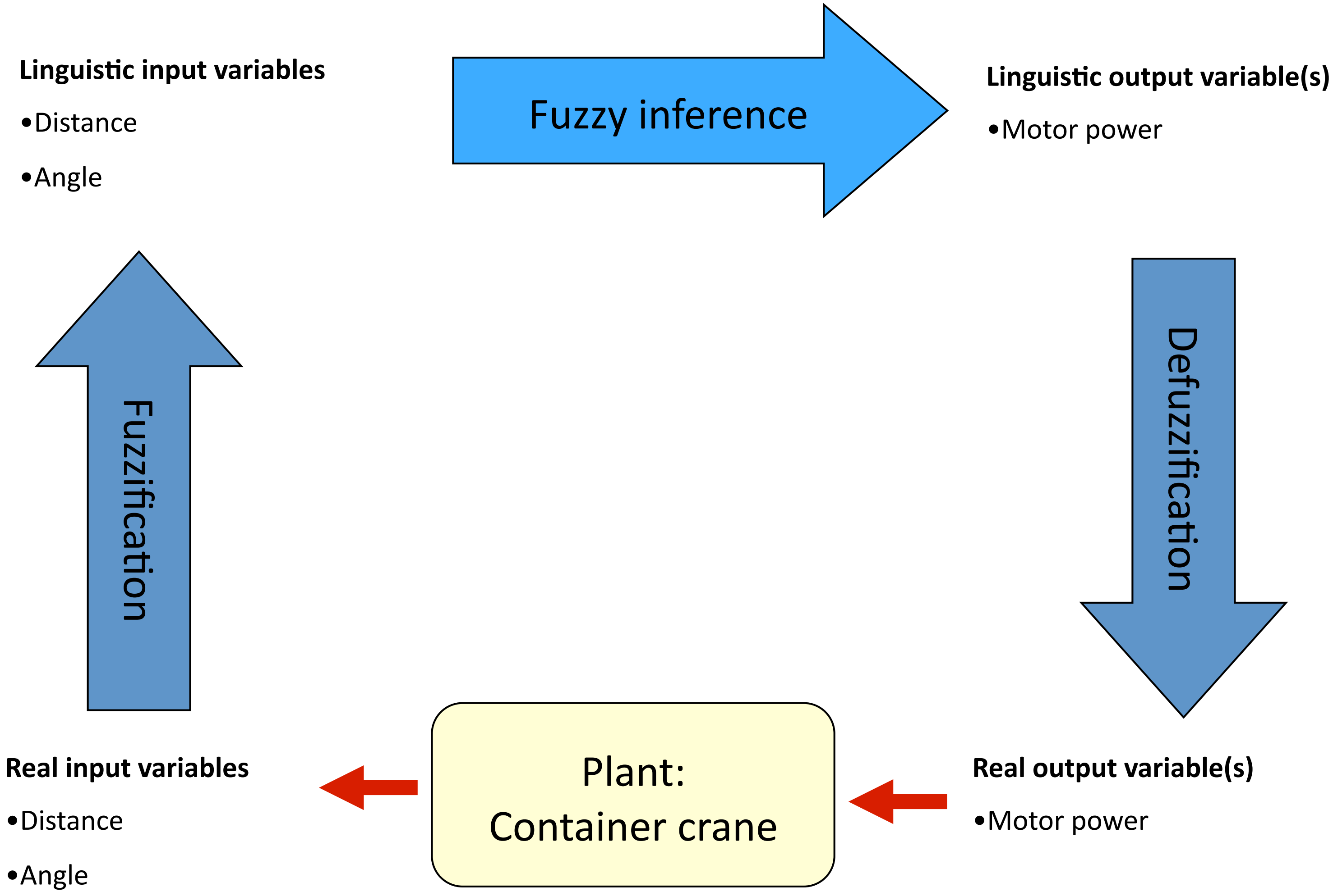


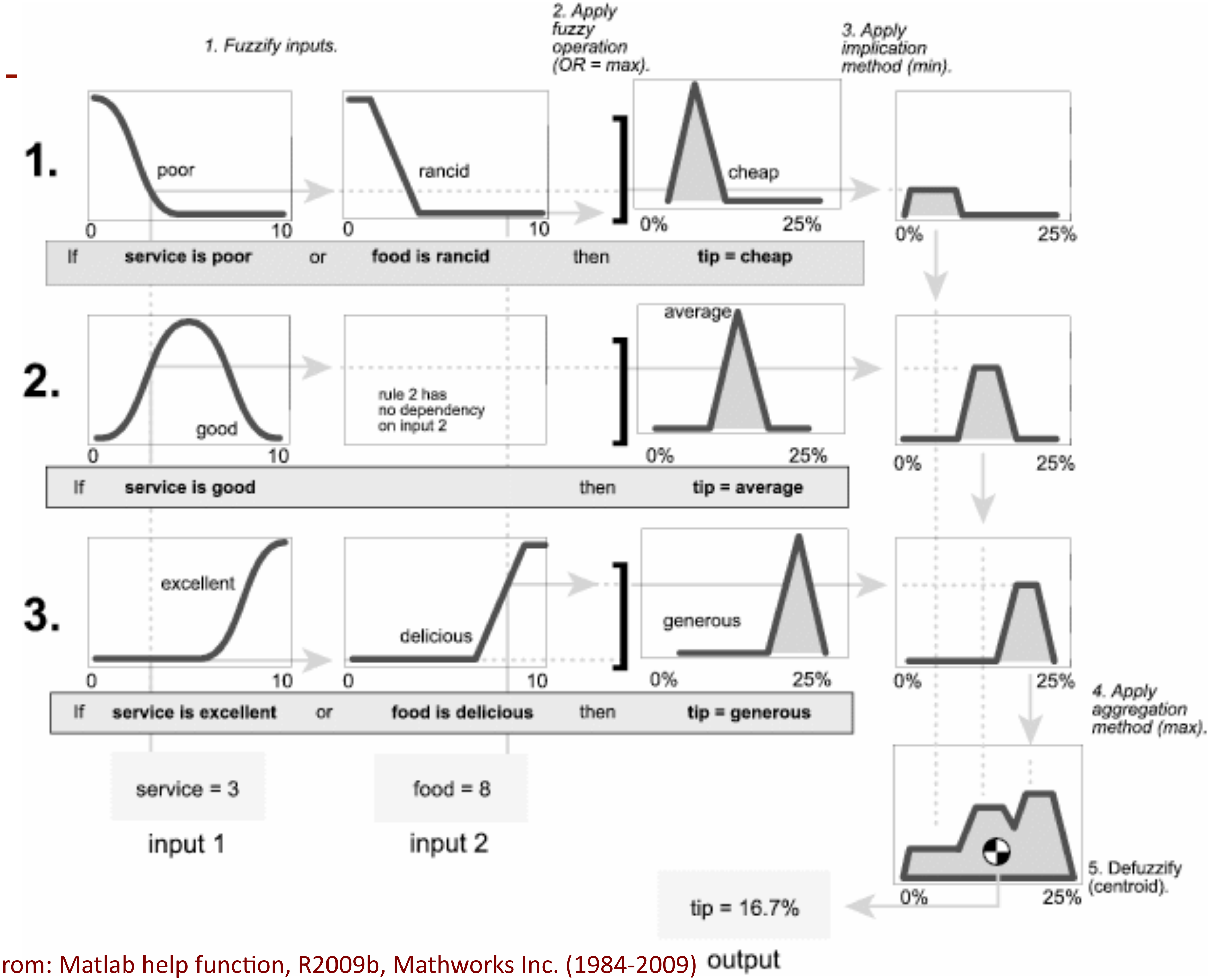
From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

- In Matlab the inference functions operate on column-wise matrices.

- Best implimented usign the graphic user interface (GUI)

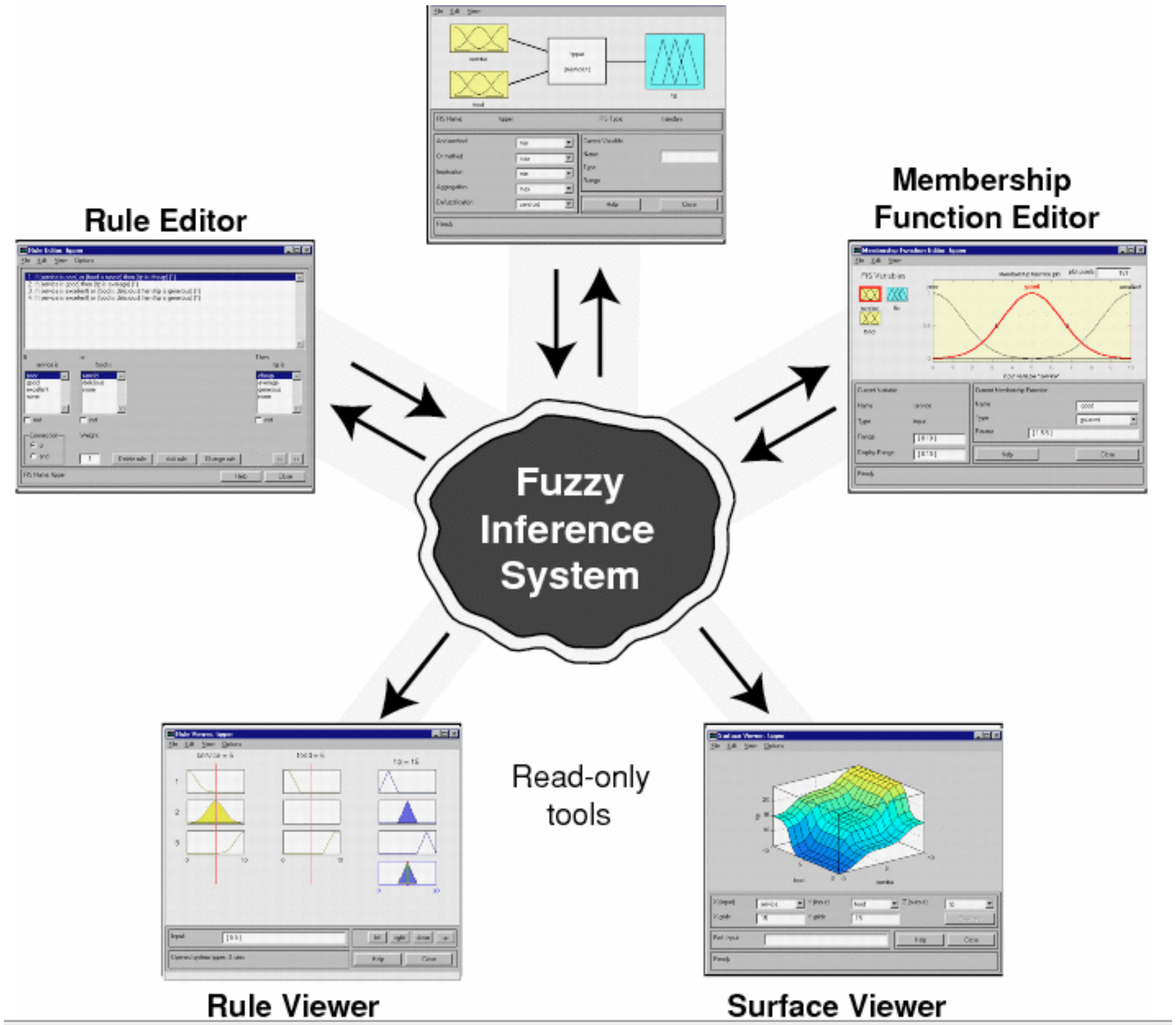
Fuzzy logic crane controller





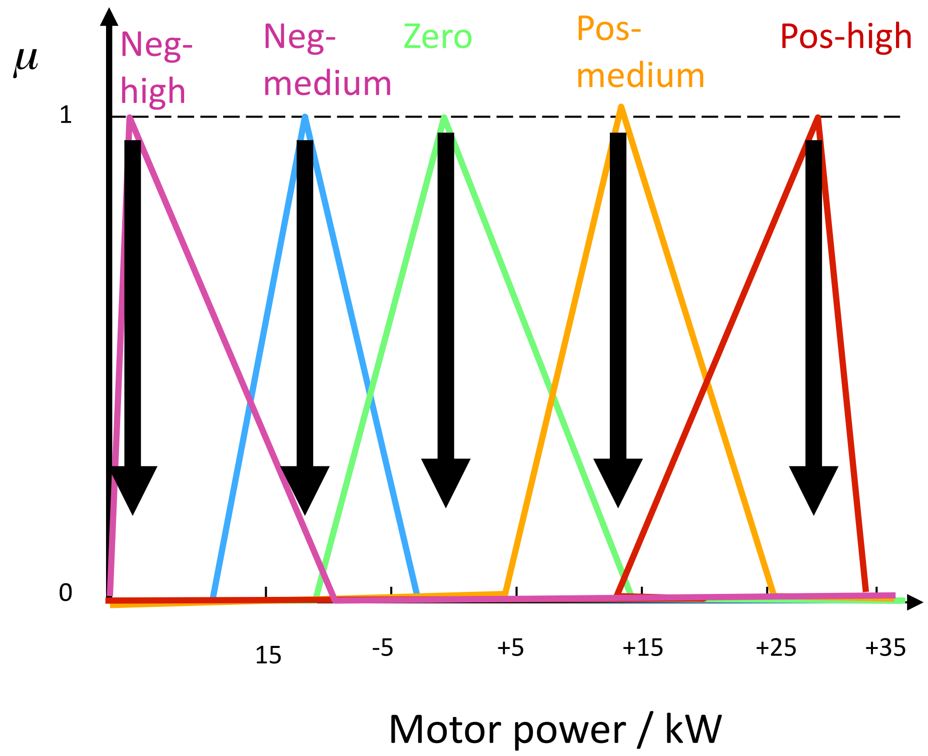
From: Matlab help function, R2009b, Mathworks Inc. (1984-2009)

FIS system in Matlab



Defuzzyfication

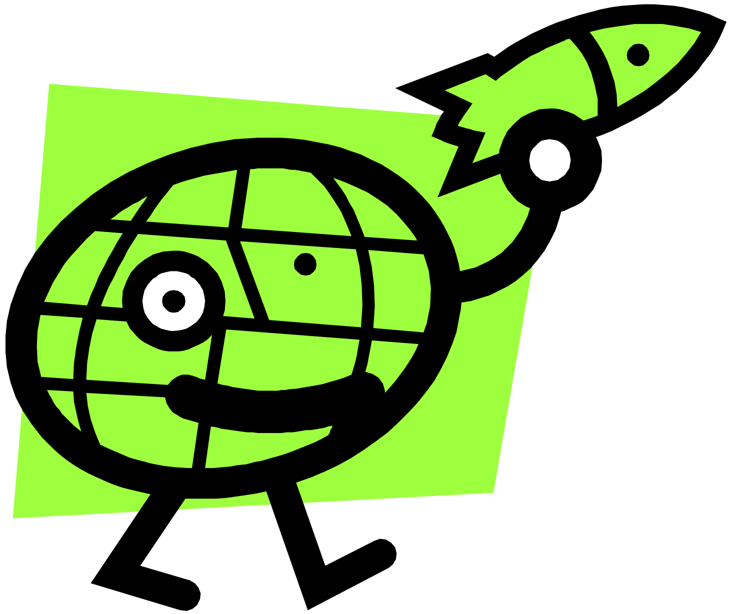
- Transforms linguistic variable to a ”sharp” real value
- Takes some defined weighted mean of membership function
- Positive-high 0.7 and pos-medium 0.3 gives: *P*=0.3*14+0.7*29 = + 24.5 kW
- Approach is to find best compromise
- Inherently unlinear



Defuzzification

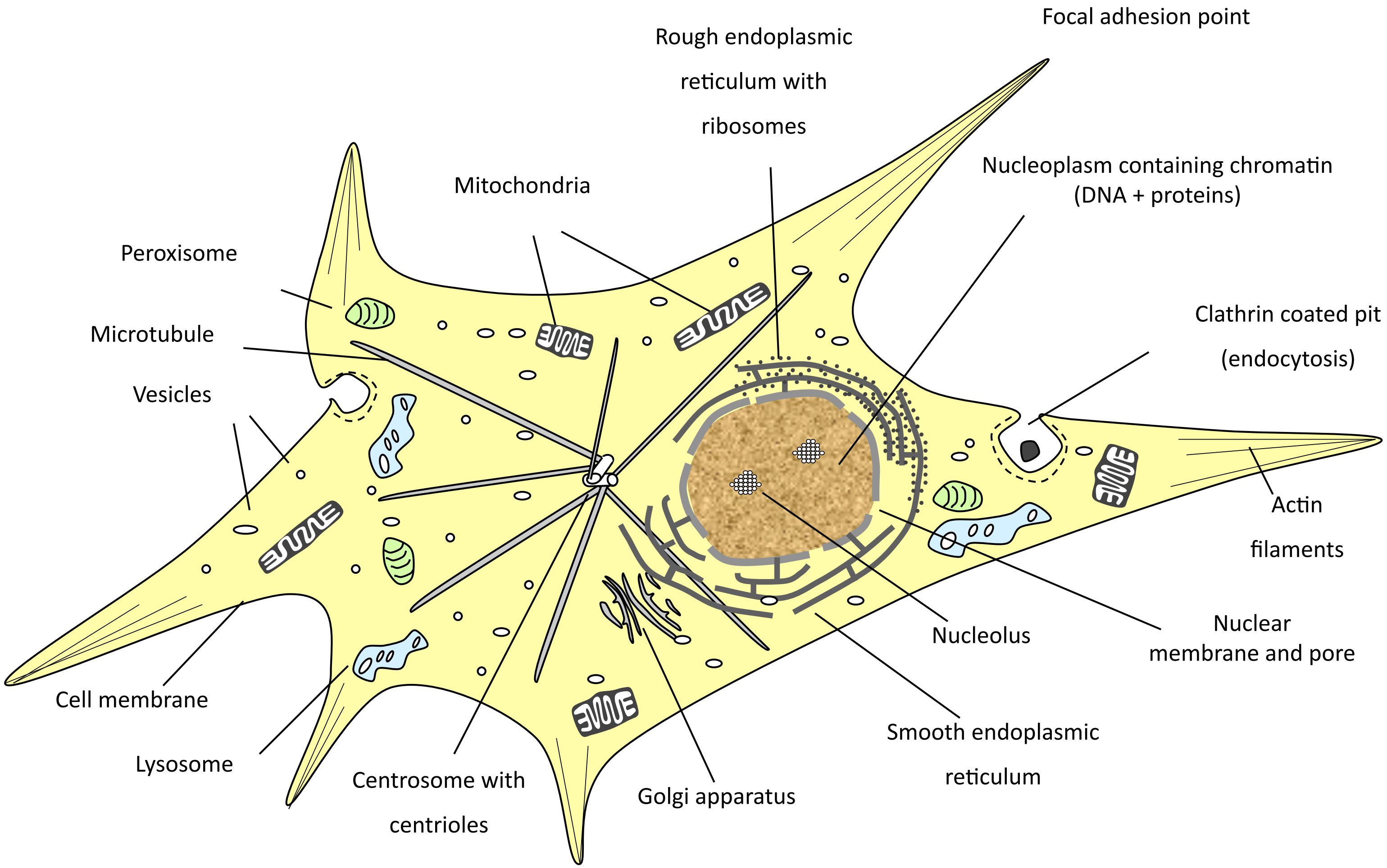
- Different approaches can be selected for defuzification
- Mean-of-maximum (MoM)
  - Fast needs only comparisons
- Centre of Area (CoA)
  - Slow as needs computation of area under membership function
- Center of Maximum
  - Relatively fast – only needs division
- CoA and CoM are same as MoM for singleton functions

Pause

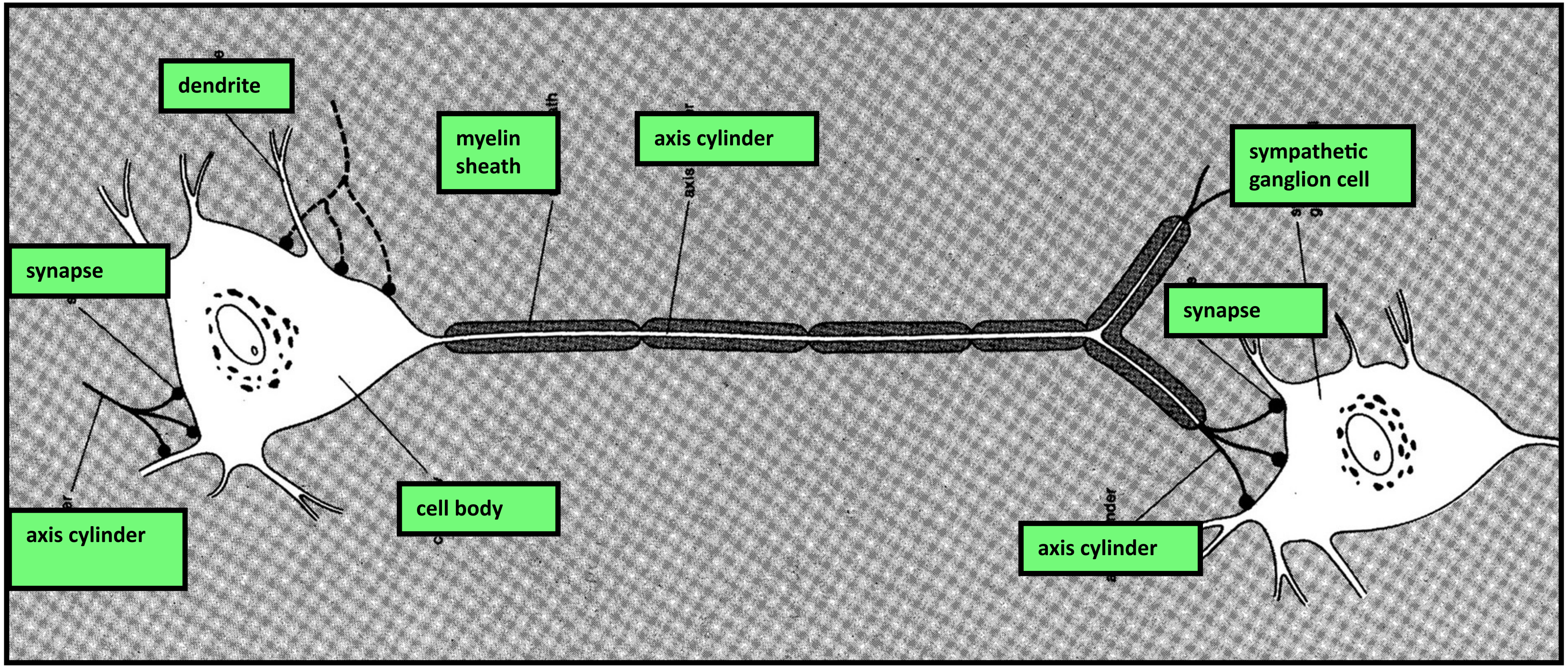


Neuro Fuzzy

Generic animal cell



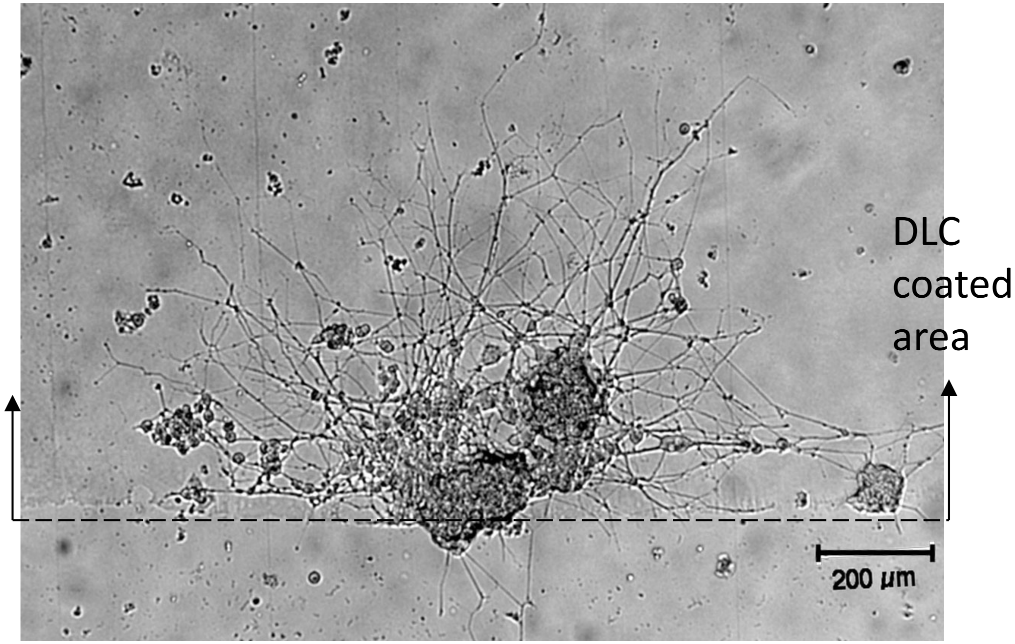
From: H. J. Whitlow and Dan J. White; *Ion beam techniques in nanotechnology* (eds.) Y. Zhang, R. Hellborg and H.J. Whitlow (In preperation) neuron — Dendrites & Axons — synapses — Neural networks



From: Somjai Sangyueyongpipat

The human brain

- First realised to be more than a consistency of curds by Willis, Boyle, Wren in Oxford in the period of English civil war
- 10^{11} nerve cells or neurons
- 10^{14} synapses
- Each cell has an activation level between a minimum and maximum
- Synapses transfer information from one neuron to another
- Unlike Boolean logic more than 2 values

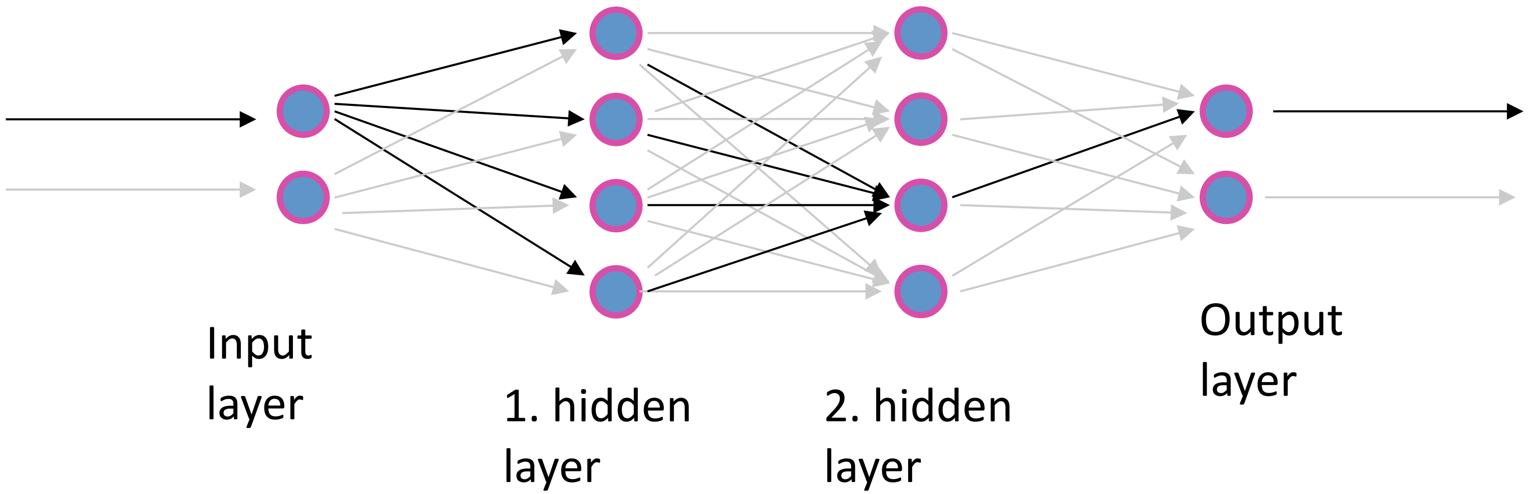


Neurons growing on a diamond like carbon film coated with collagen

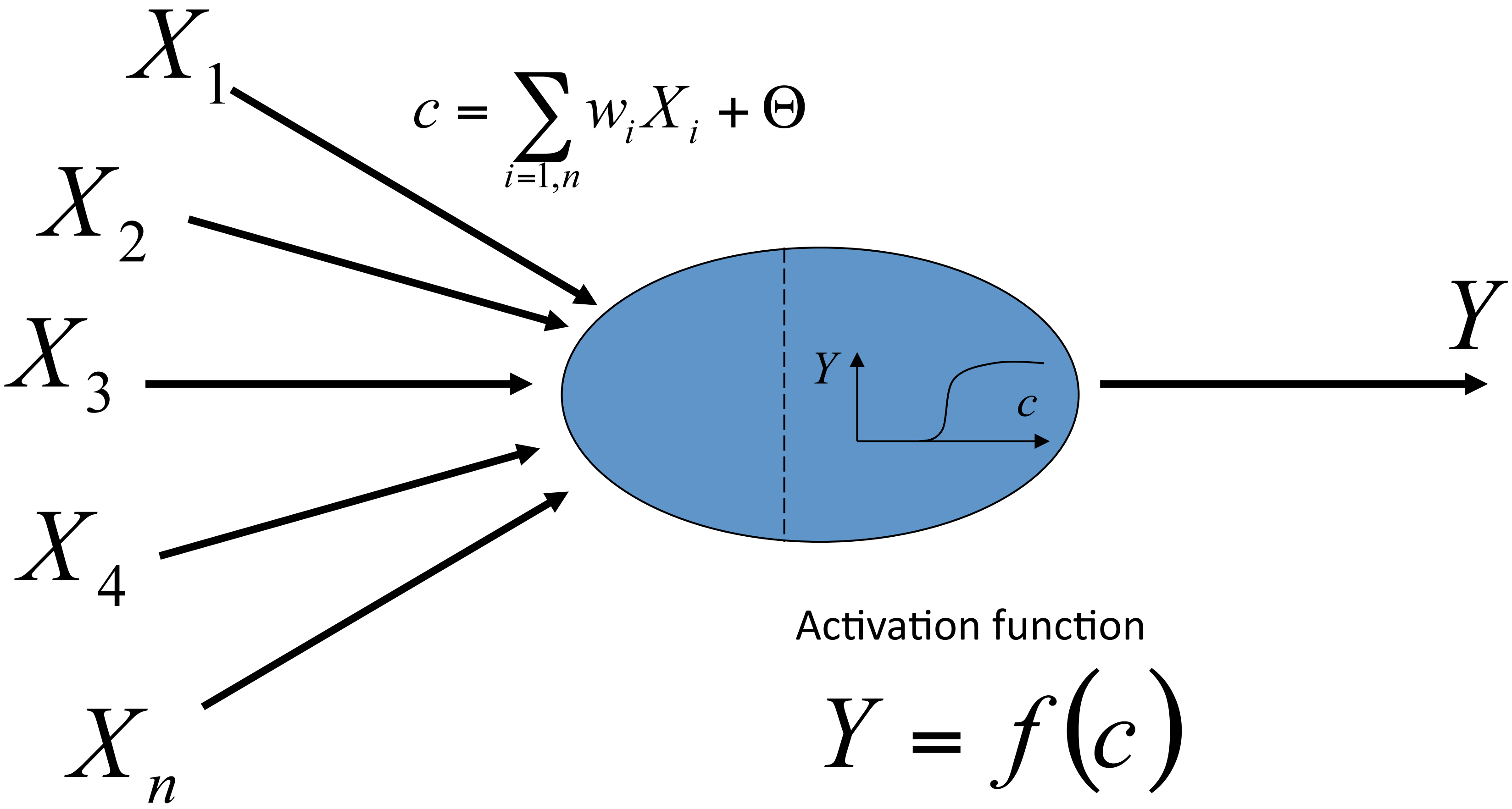
From: Dr. Somjai Sangyuenyongpipat (PhD thesis, Chiang Mai University, 2006)

Neural-net topography

- Mimics network of neurons



Mathematical model of a neuron



Activation function

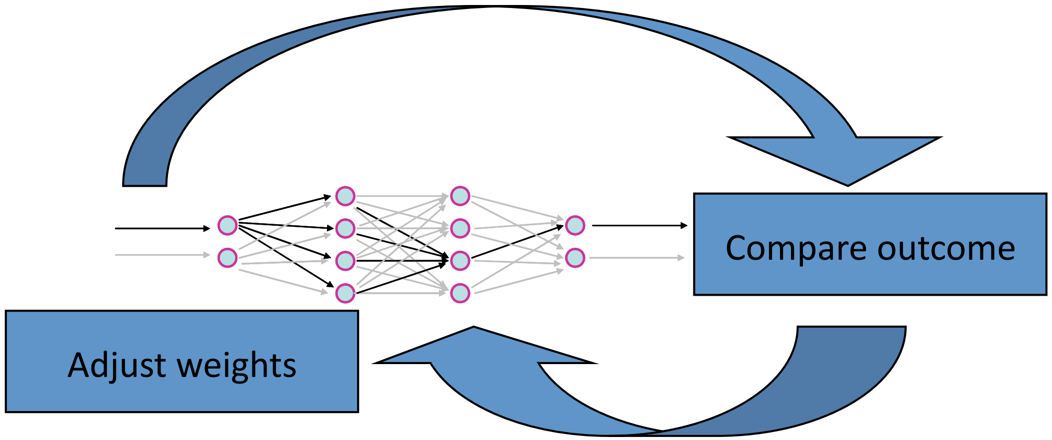
$$Y = f(c)$$

Learning and working phases

- Learning phase
  - Initially the neural net cannot function becasue it has no prior knowledge of how to reposnd to given stimulii
  - Learning is the process of defining the weights w_i assoicated with each neuron input.
  - A teacher is a mathematical process or person that applies data sets and rates the quality of the neural net performance.
  - Uses real or constructed data sets can be used.
- Working phase
  - In working phase the neural net outputs values similar to the corresponding training values.
  - Generally in the working phase the neural net does not learn.
  - This avoids risk for drifting into output of hazardous data.

Error back algorithum

- Hebb deduced the following learning rule:
  - Increase weight to active input neuron if output from the neuron should be active
  - Decrease the weight if the output from the neuron should be inactive
  - Known as the **Hebbian Rule**
- Feedback used iteratively to train loop
- Called the **Error Back Algorithm**
- Error is not normaly reduced to zero
- High training performance



Comparisson of neuro- and fuzzy logic

- **Fuzzy**
  - Inference requires large number of if-then relations
  - Need to be manually defined
  - Explicite verification easy and efficent
  - Cannot be trained
  - Interpretation easy (important for fail safe applications

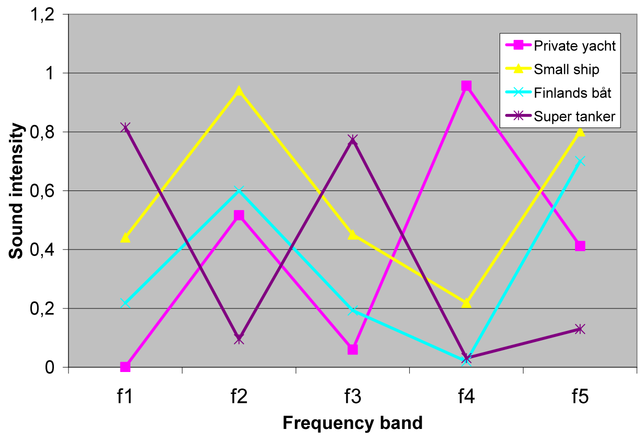
- **Neuro**
  - Implicit
  - No easy interpretation or modification (objectionable for fail-safe scenarios)
  - Trains itself by learning of data sets
  - Black box
  - Setting up the training algorithm is a black art

Combine to get best of both situations

Neurofuzzy

Development of neurofuzzy systems

1. Obtain training data
   1. Example: Ship size identification from underwater sonic emission



2. Create initial system
   based on idears of membership functions
   e.g.
   IF F2 high AND F4 low AND F5 low THEN
   Vesseltype=3

   This enhances the speed and
   performance of learning process

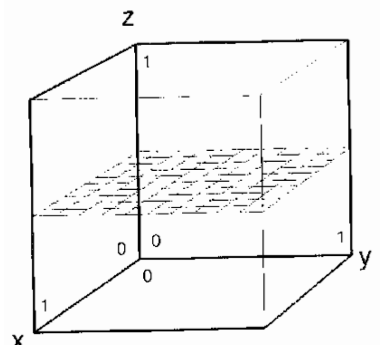
- Adjust learning process
  – Learn rate for rules
  – Learn rate for membership functions
  – Use lower learn rates for complex systems and data sets to to optimise convergence

- Optimise and verify
  – Remove redundant rules
  – Check for stable operation by simulation

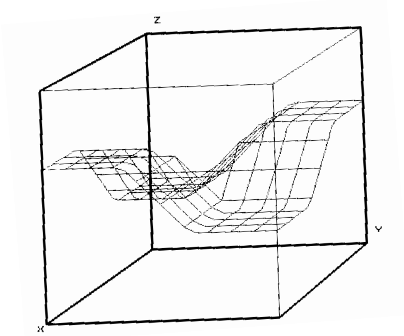
Training of XOR function

X	Y	Z=XOR(X,Y)
0	0	0
1	0	1
0	1	1
1	1	0

Initial state $Z = 0.5$

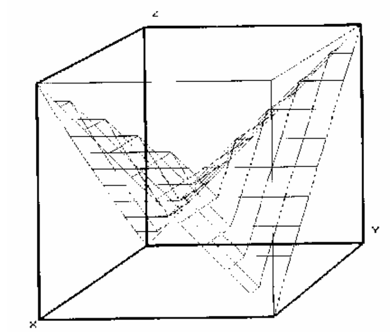


Training:
folding of the transfer surfaace



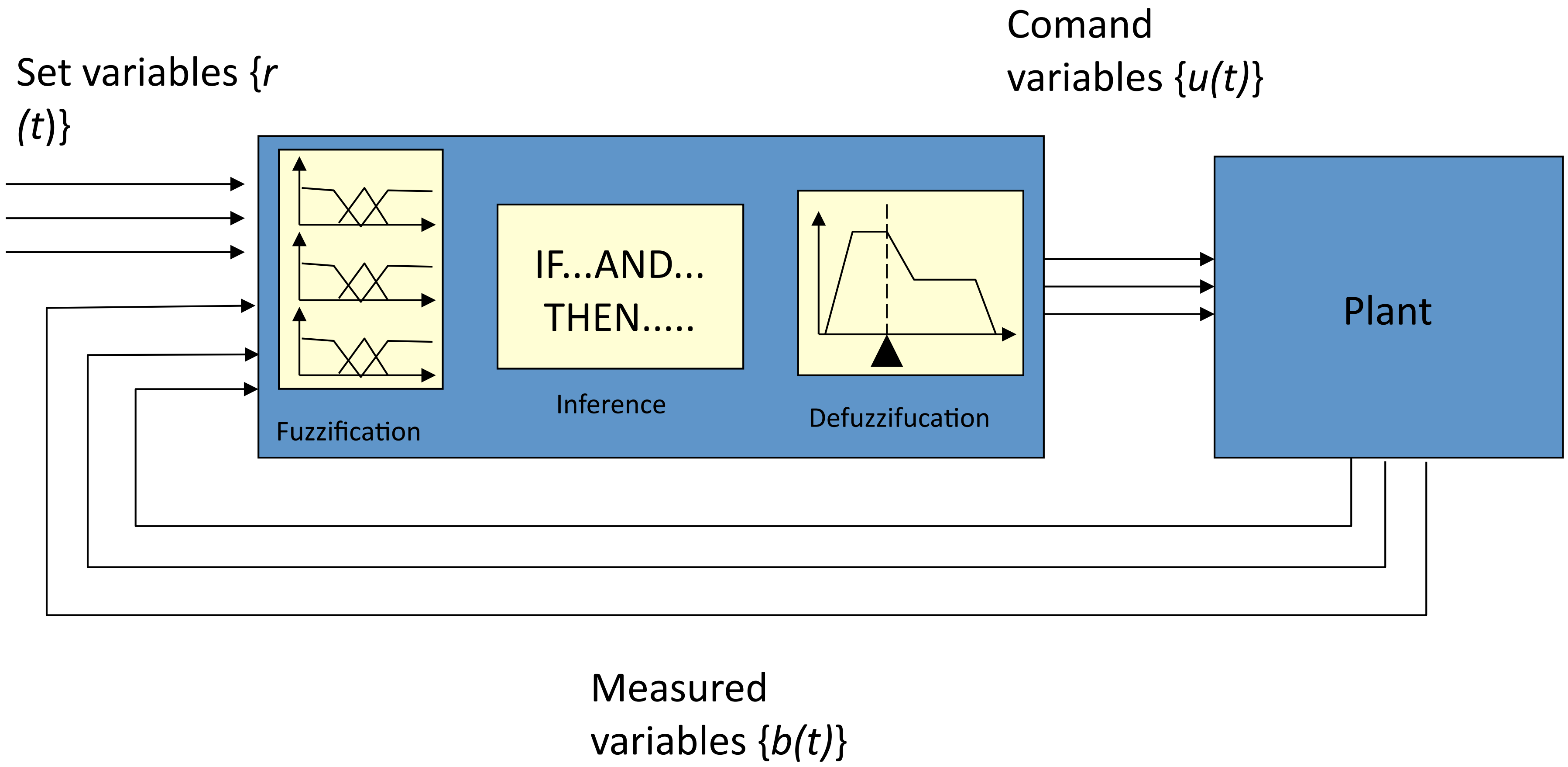
Training time increases rapidly as number of input variables and training samples increase.

Final state Z = XOR (X,Y)

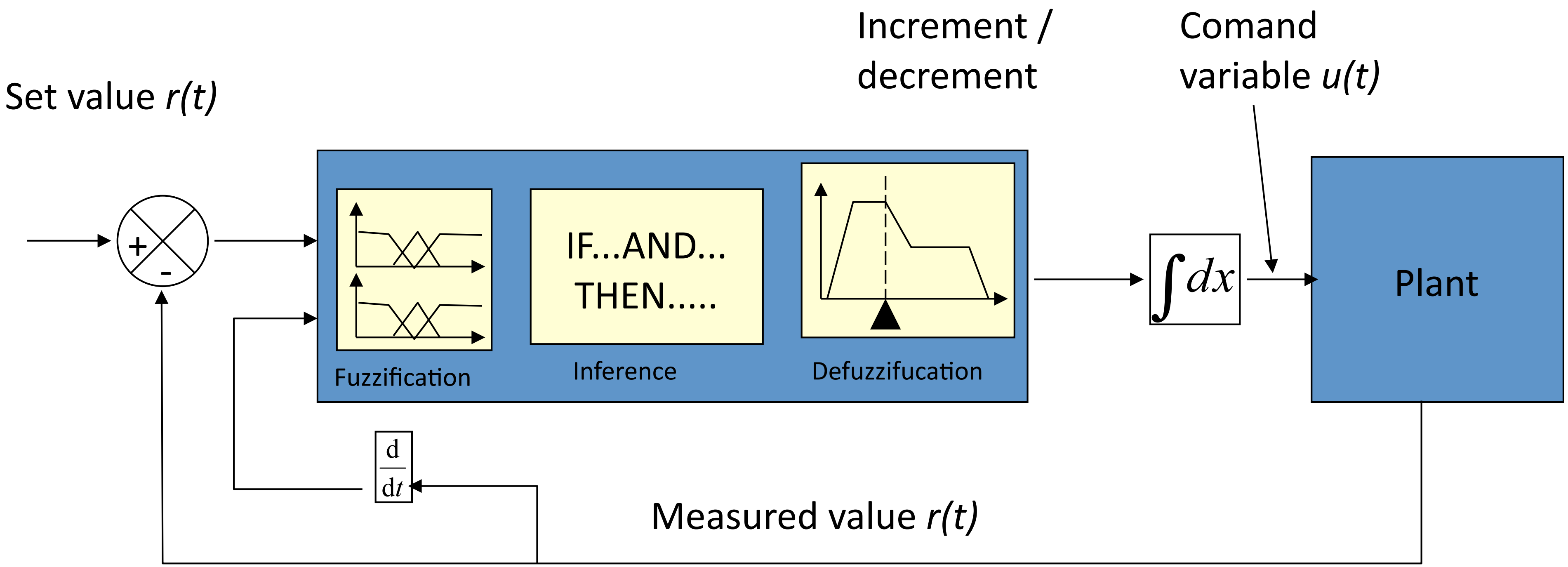


From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.

Fuzzy logic closed loop controller



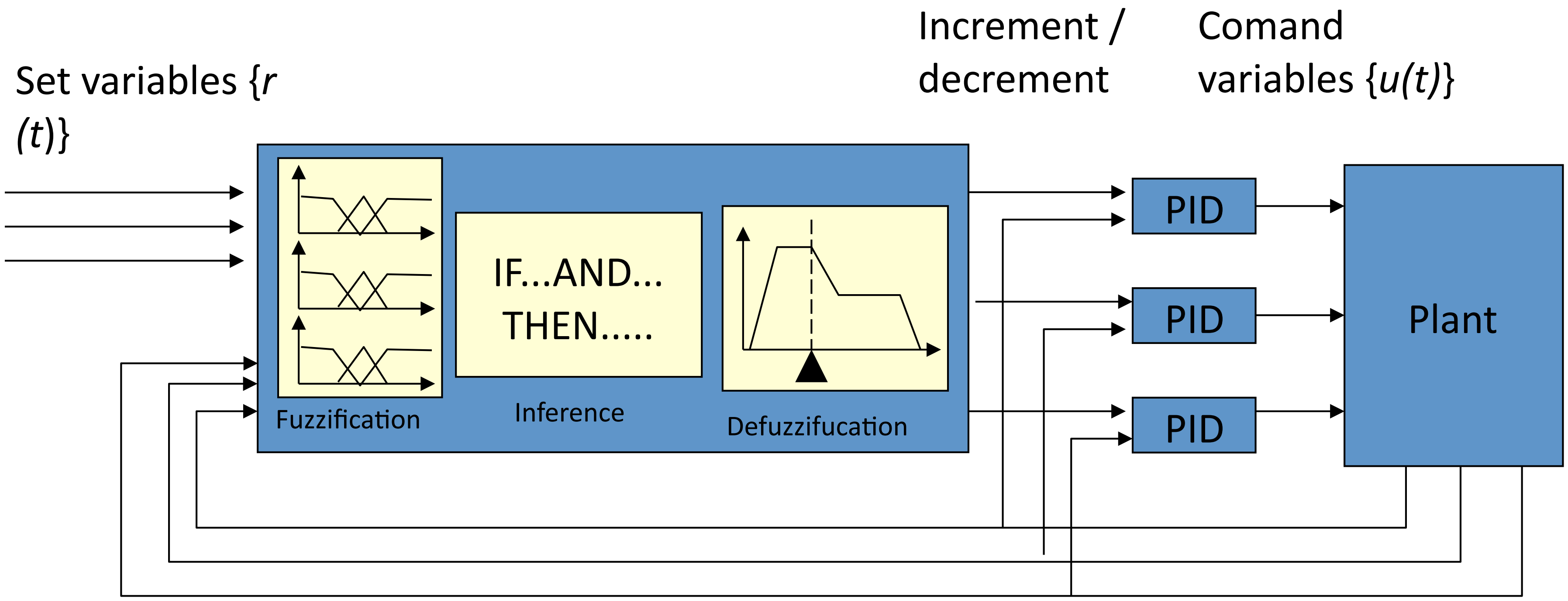
Fuzzy PID controller



- Fuzzy PID can impliment non-linear stratergies
- Fuzzy PID also can impliment liguistic rules
- In linear ssytem the control variable can be cast in terms of the usual PID transfer equation

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)\,\mathrm{d}t + K_p T_d \frac{\mathrm{d}e(t)}{\mathrm{d}t}$$

Combination of fuzzy and conventional PID control



- Exploits fact that systems are (generally) linear around their operating points and conventional PID work well
- Fuzzy logic needed for the non-linear control allowing use of linguistic rules

APPLIANCE AND INDUSTRIAL APPLCIATIONS

Other combinations of fuzzy and conventional control logic

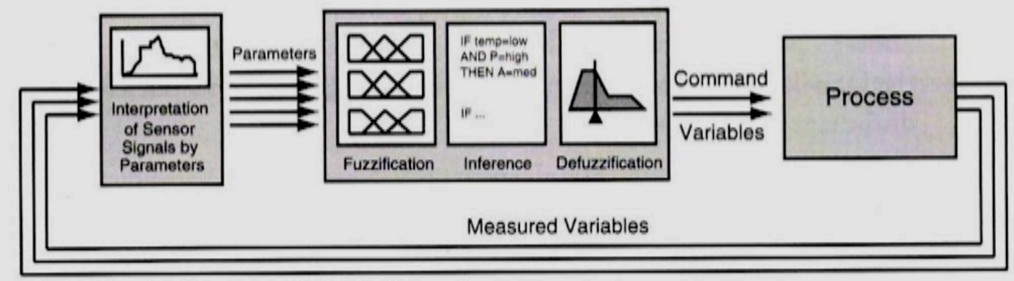


Figure 5.4 Some applications require preprocessing of the process signals.

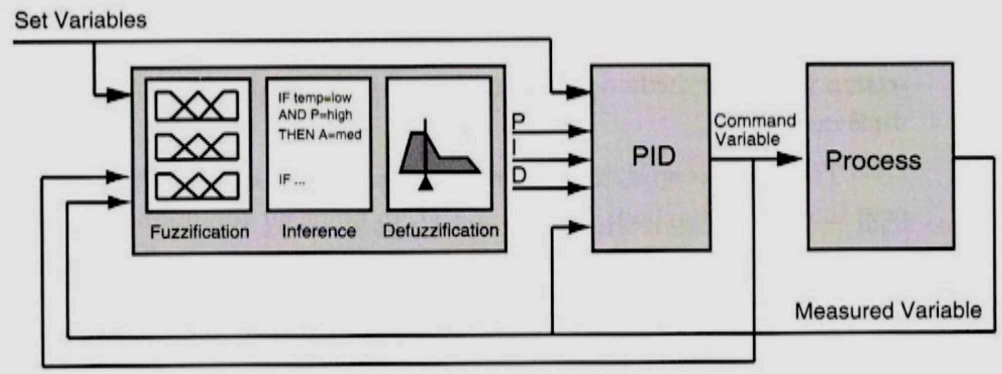


Figure 5.5 Adaptation of P, I, and D parameters by a fuzzy logic system.

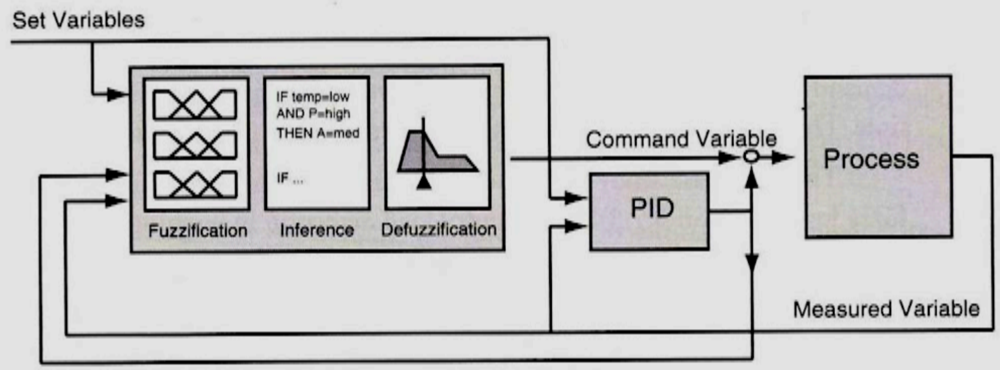
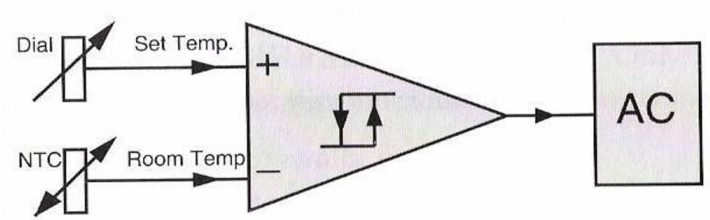


Figure 5.6 Intervention of a fuzzy logic system to a PID controller output.

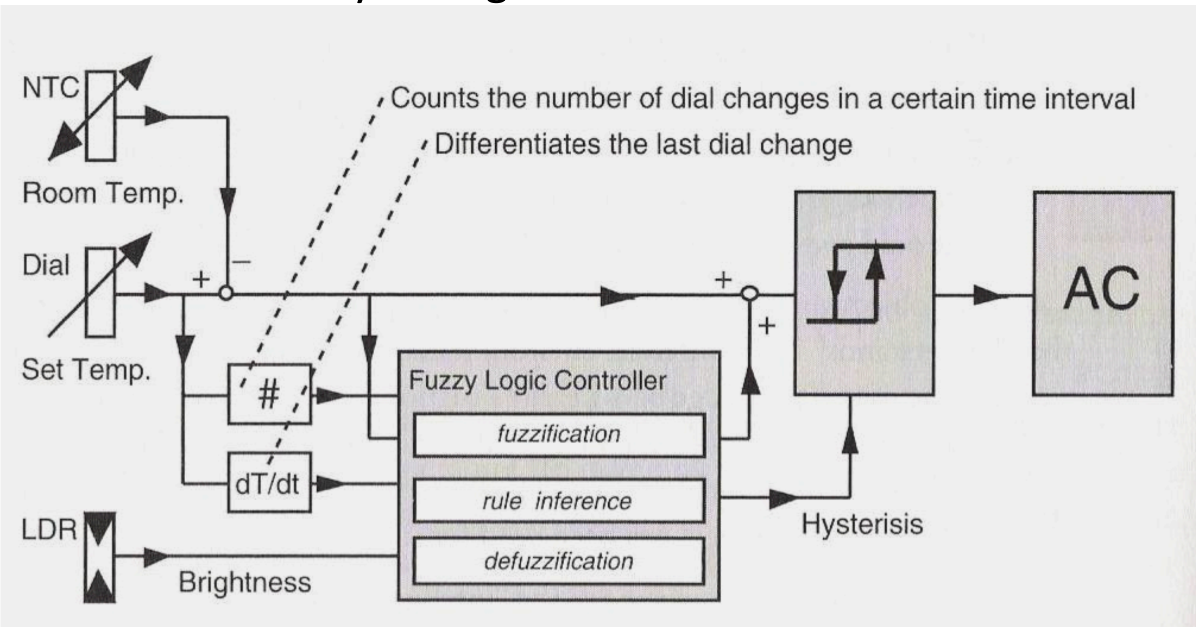
From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.

Application: air-con



Conventional bang-bang AC controller

Fuzzy inteligent AC controller



From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.

Fuzzy input variables

1. Temp error between set and room temperature
2. Set temp signal differentiated with a time constant
3. Number of set temp changes per hour
4. Brightness sensor used to sense night and day

Application: ABS brake system

Slack value: *s*

$$s = \frac{v_{car} - v_{wheel}}{v_{car}}$$

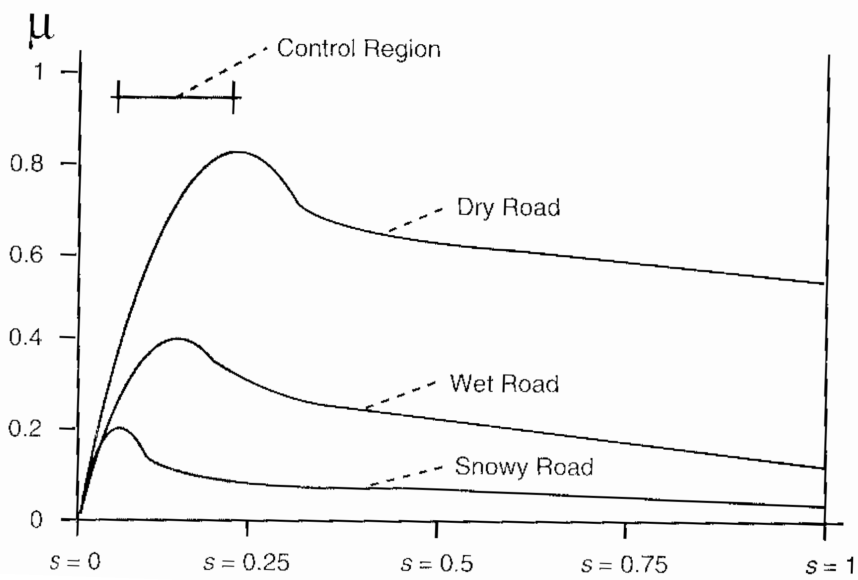


Table 5.1 Slack Value for Maximum Brake Effect Depending on Type of Road

Road Condition	Optimum Slack
Dry	0.2
Slippery or wet asphalt	0.12
Ice or snow	0.05

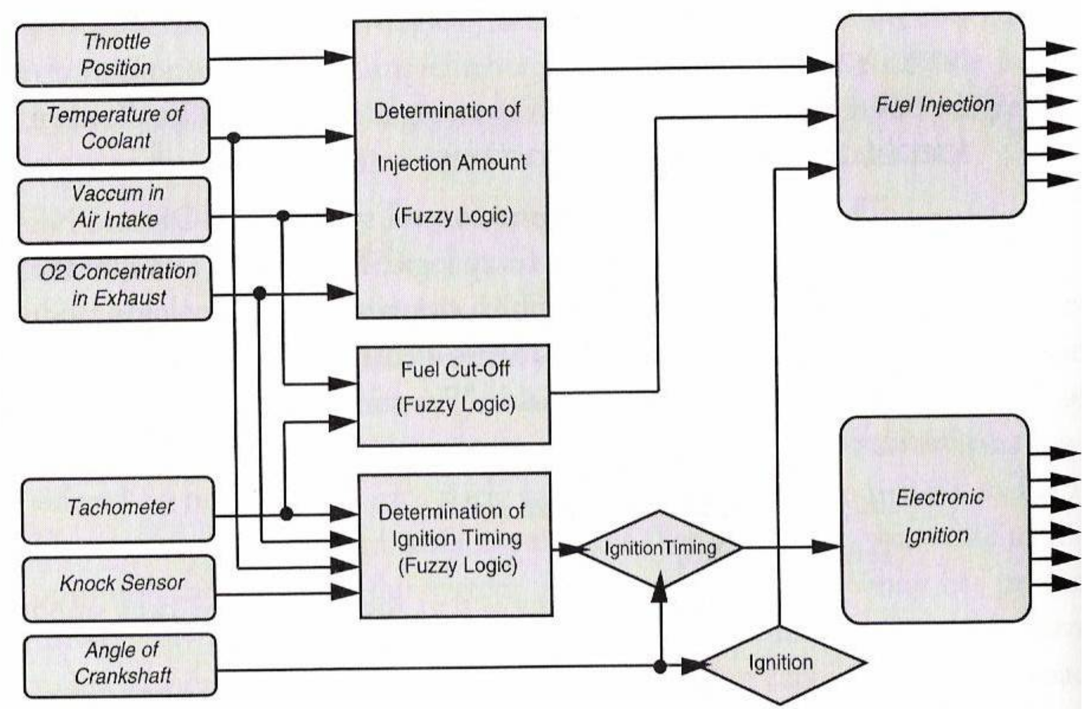
For best effect knowledge of the value of the slack is need for the actual road surface

Use fuzzy logic:

1. With first blocking of wheel adjust brake presure so *s*= 0.1
2. Evaluate reaction to braking use fuzzy logic to estimate road surface
3. Correct value of slack to achieve maximum braking

From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.

Application: Fuzzy engine control



From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.

- Linguistic variable "*situation*"

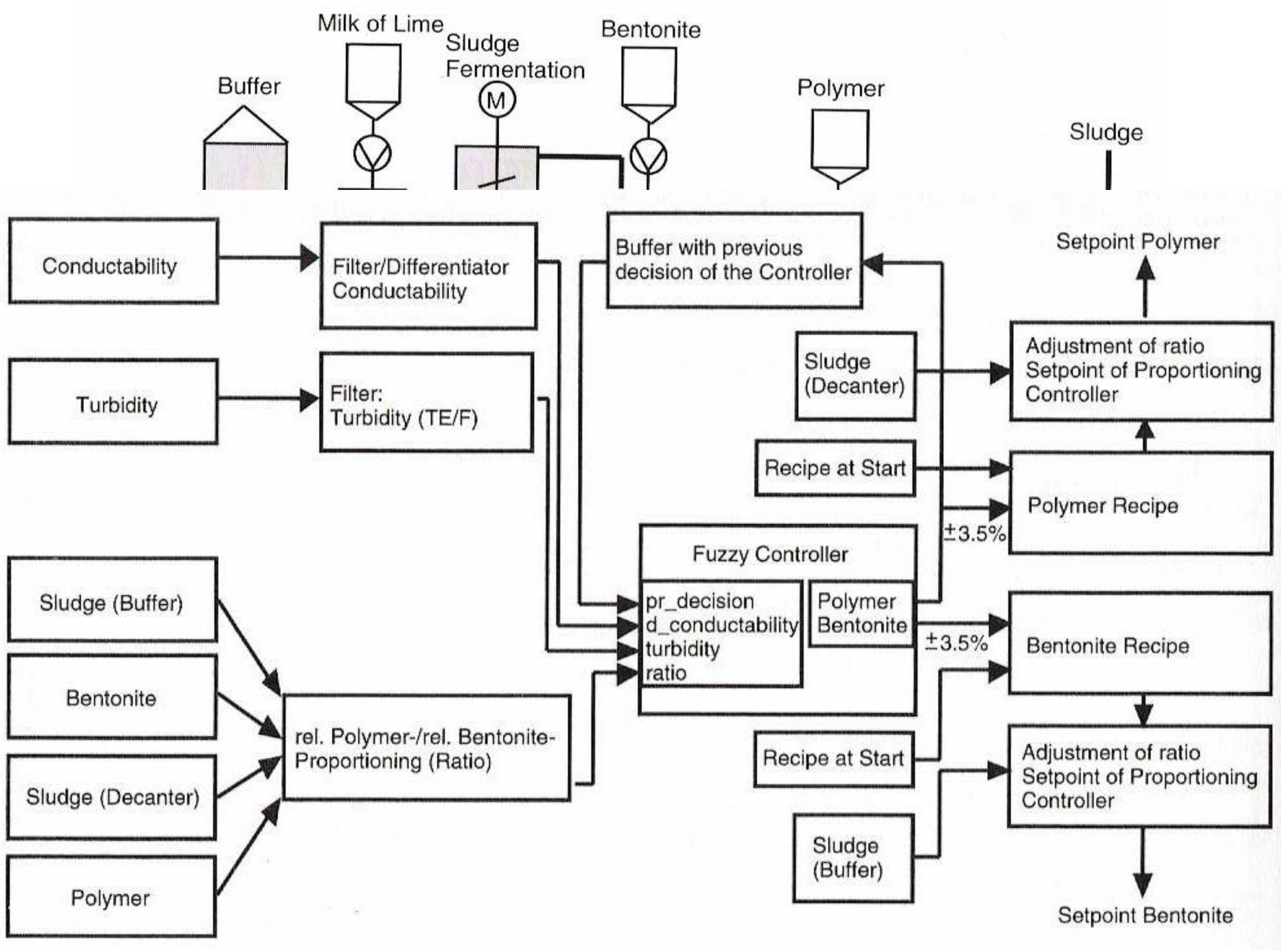
1. **Start** running smooth
   early ignition and fat mix
2. **Idle**
   timing and mixture set by engine temp
3. **Normal** low medium load Meagre mix and timing set by knocking
4. **Normal** high load
   Fat Mix early ignition
5. **Coasting**
   Fuel cut-off depending on situation
6. **Acceleration**
   Depending on load fat mix

Typical operating could be:
[0.1; 0; 1; 0; 0; 0.4]
Starting:
[1.;0.;0.1;0.4;0.1,0.5]

Application: industrial sludge fermetation

- Control stratergy
  1. Minimise amount of precipitants (expensive)
  2. Minimise water content (evaporation energy intensive)
  3. Biological content 0.7 g/l (too high gives breakdown)

- Uses 3 input signals;
  - Conductance,
  - Turgidity
  - Previous recipe



From: C.A. Von Altrock, *Fuzzy logic and Neurofuzzy applications explained* (Prentice Hall, N-J)1995.


Thats all for today