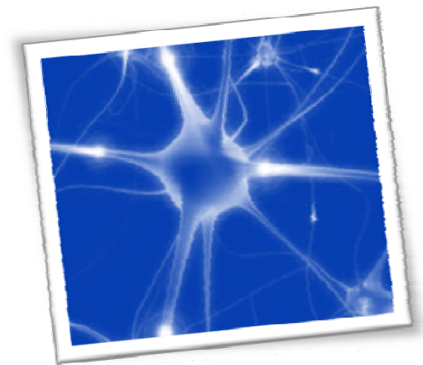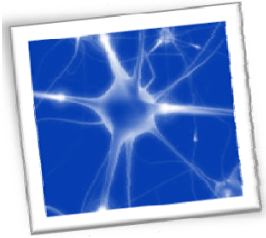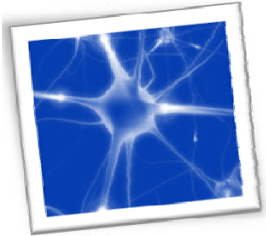# Lecture 6&7: Fuzzy Inference Systems

Outline

- Introduction to Fuzzy Inference Systems(FIS)
- Mamdani Fuzzy Models
- Sugeno Fuzzy Models
- Tsukamoto Fuzzy Models
- Other Considerations
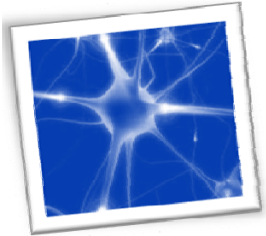 - Input Space Partitioning
  - Fuzzy Modeling

Introduction

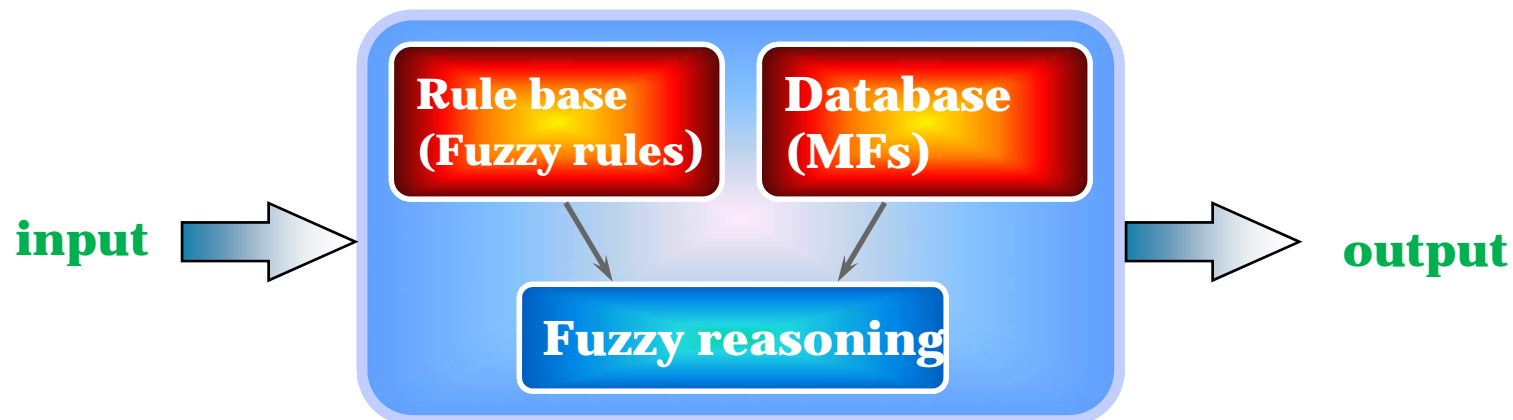- What is a fuzzy inference system (FIS)?

  A nonlinear mapping that derives its output based on fuzzy reasoning and a set of fuzzy if-then rules. The domain and range of the mapping could be fuzzy sets or points in a multidimensional spaces.
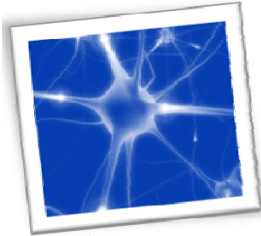
- Also known as
  - Fuzzy models
  - Fuzzy associate memory
  - Fuzzy-rule-based systems
  - Fuzzy expert systems
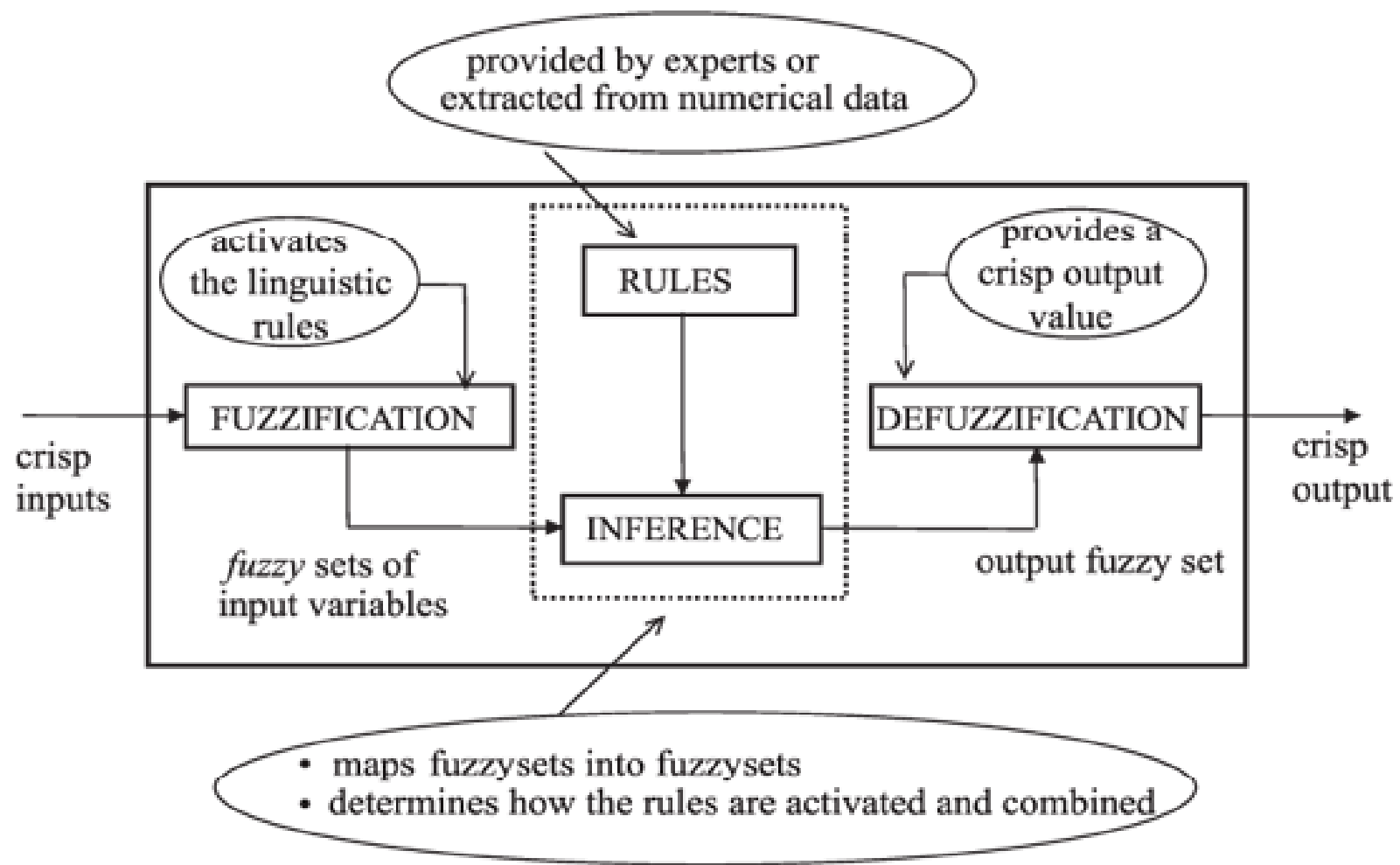 - Fuzzy Logic Controller
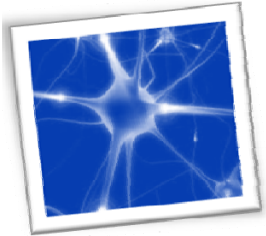
Structure of FIS

- The basic structure of a fuzzy inference system consists of three conceptual components:
  - A **rule base**, which contains a selection of fuzzy rules
  - A **database** (or **dictionary**), which defines the membership functions used in the fuzzy rules
  - And a **reasoning mechanism**, which performs the inference procedure upon the rules and given facts to derive a reasonable output or conclusion.
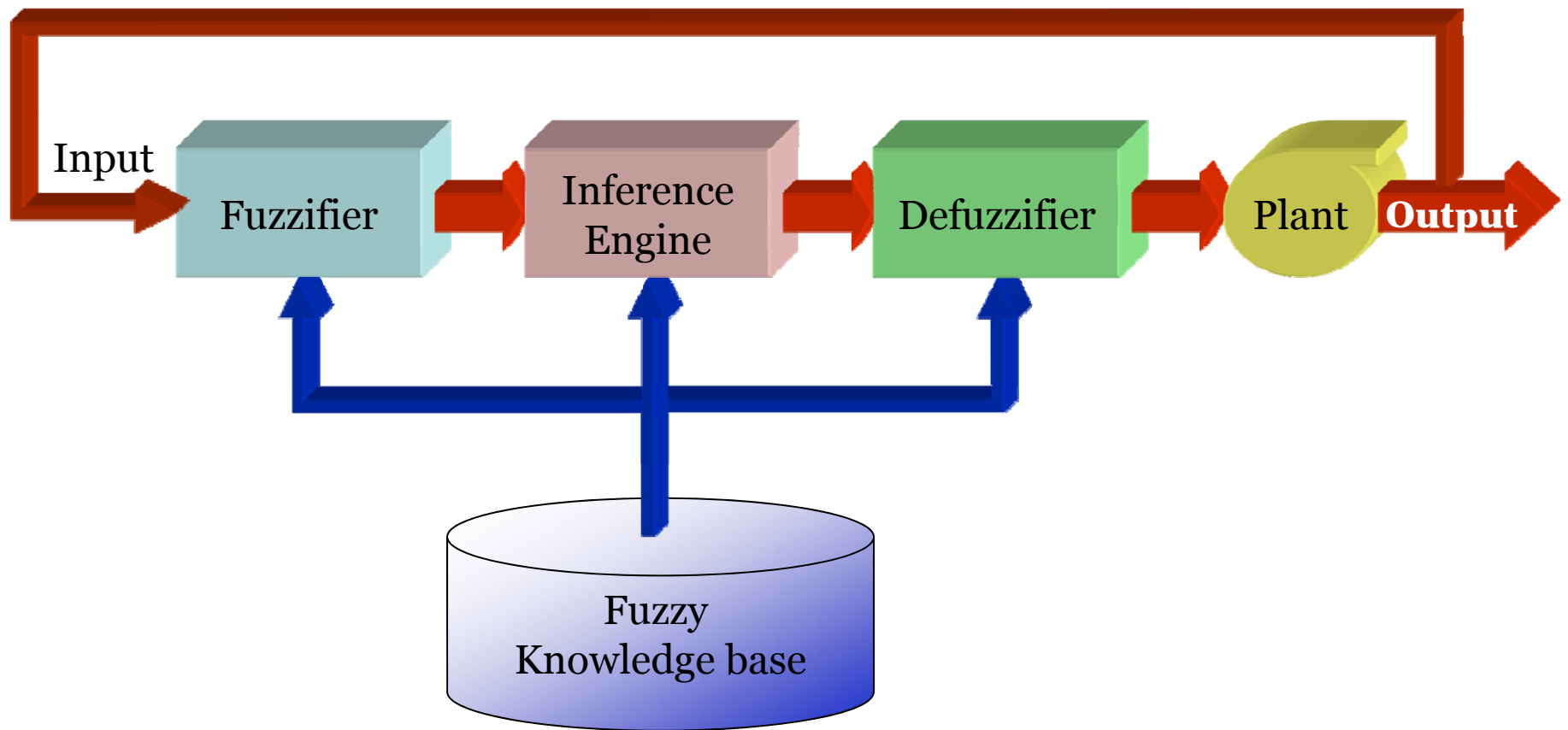
input → **Rule base (Fuzzy rules)** **Database (MFs)** → **Fuzzy reasoning** → output

Structure of FIS

provided by experts or extracted from numerical data

activates the linguistic rules

RULES

provides a crisp output value

FUZZIFICATION

DEFUZZIFICATION

crisp inputs

crisp output

fuzzy sets of input variables

INFERENCE

output fuzzy set

- maps fuzzysets into fuzzysets
- determines how the rules are activated and combined

Example: Fuzzy Control Systems

Input → **Fuzzifier** → **Inference Engine** → **Defuzzifier** → **Plant** → **Output**

Fuzzy Knowledge base
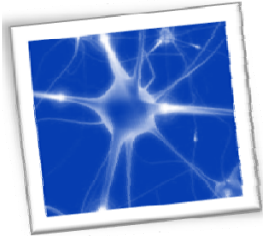
Inputs and Outputs of FIS

- The basic fuzzy inference system can take either fuzzy inputs or crisp inputs, but the outputs it produces are almost always fuzzy sets.

- Sometimes it is necessary to have a crisp output, especially in a situation where a fuzzy inference system is used as a controller.

- Therefore, we need a method of **defuzzification** to extract a crisp value to represent a fuzzy set.
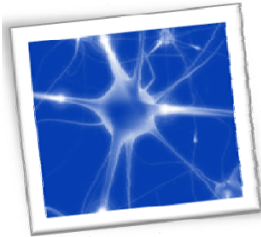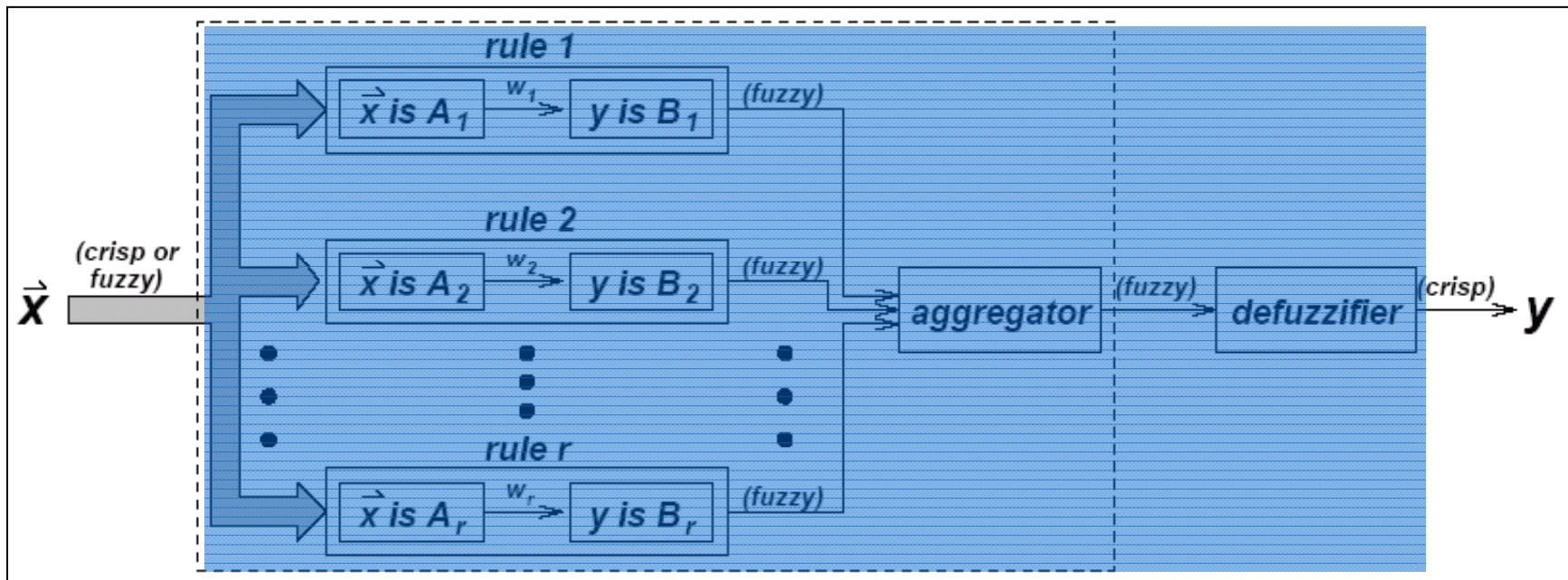
Block Diagram for a FIS with Crisp Output

- FIS implements a nonlinear mapping from its input space to output space. This mapping is accomplished by a number of fuzzy if-then rules.
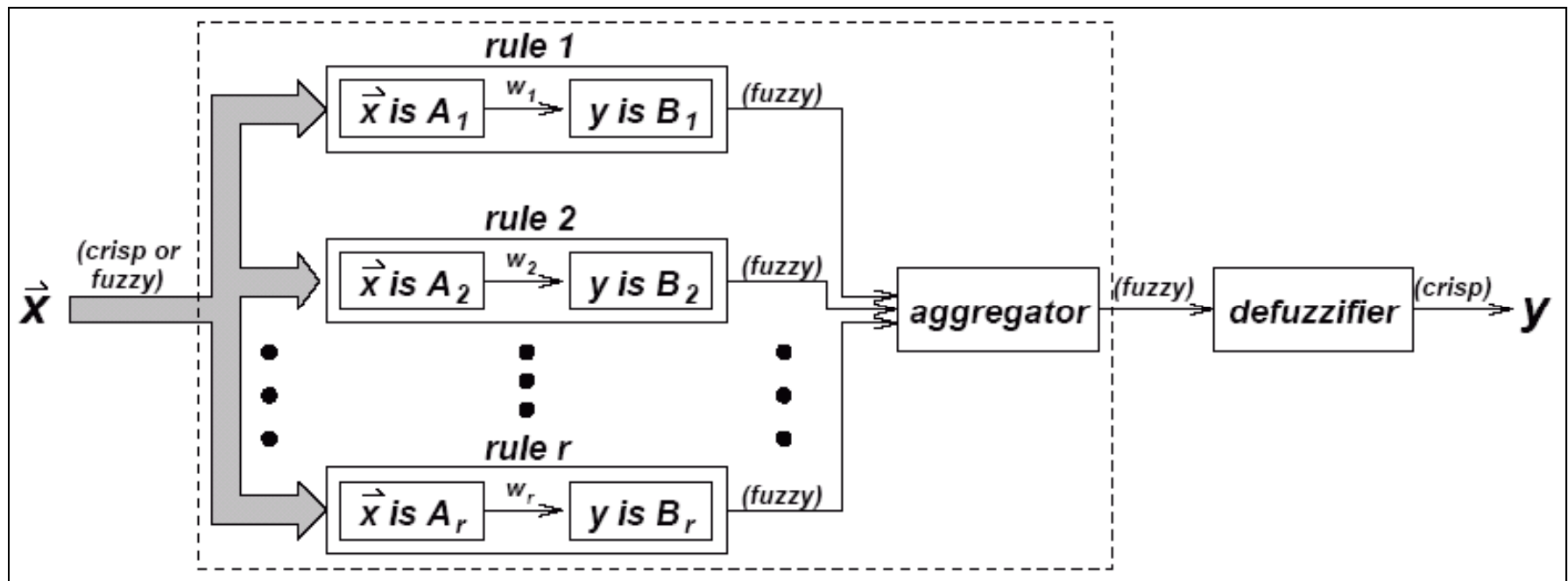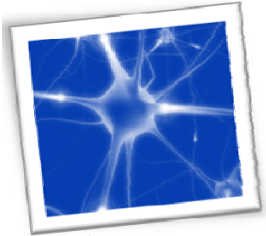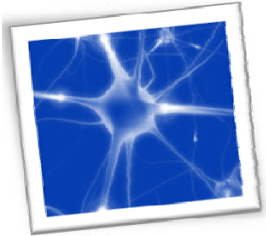
Nonlinearity

In the case of crisp inputs & outputs, a fuzzy inference system implements a nonlinear mapping from its input space to output space.
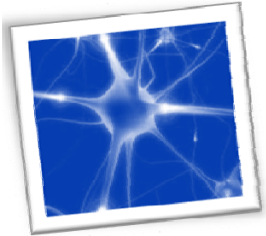
Three Popular Fuzzy Models (FIS)

- **Ebrahim Mamdani Fuzzy Models**
- **Sugeno Fuzzy Models**
- **Tsukamoto Fuzzy Models**

- The differences between these three FISs lie in the consequents of their fuzzy rules, and thus their aggregation and defuzzification procedures differ accordingly.
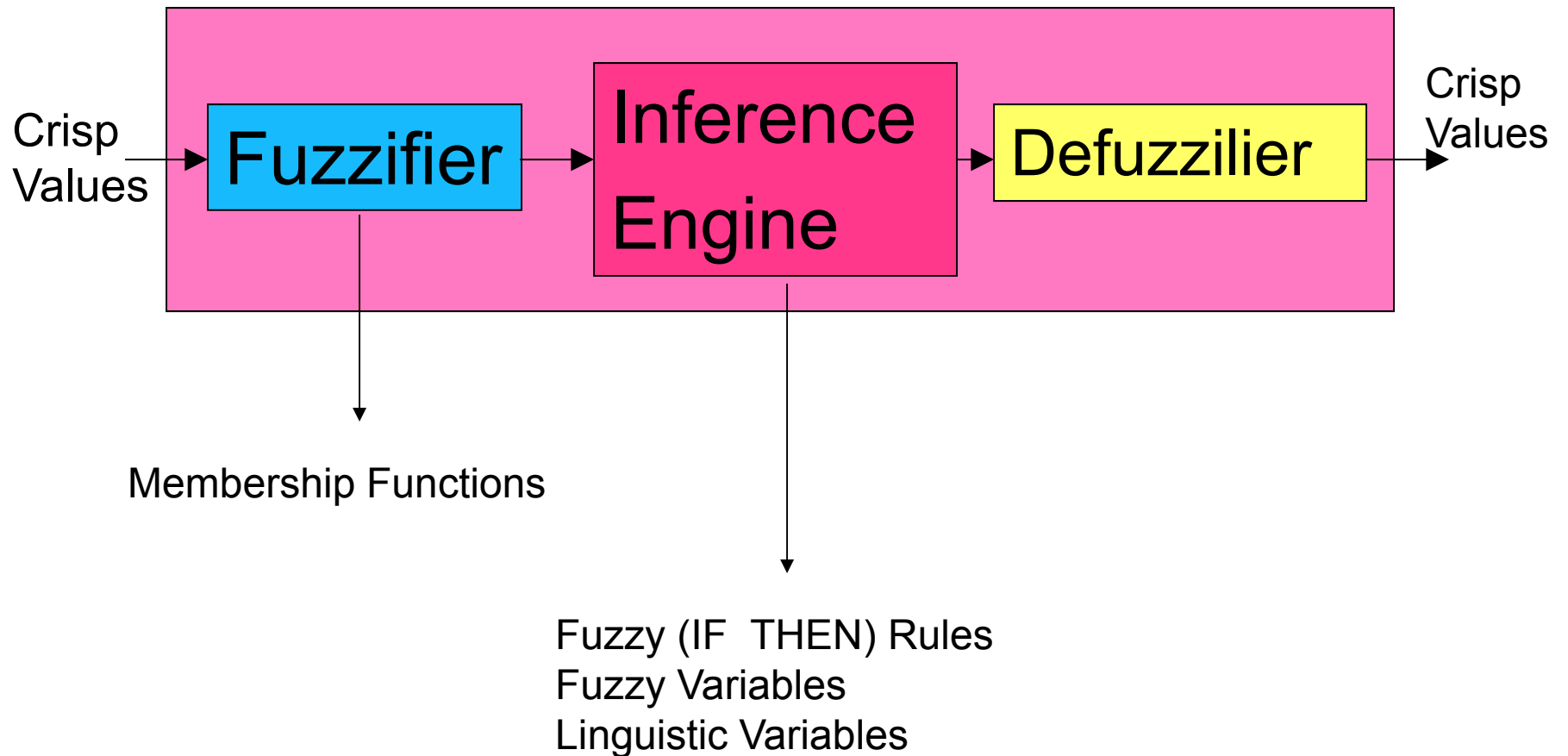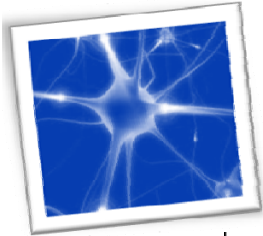
Mamdani Fuzzy Models

- The most commonly used fuzzy inference technique is the so-called **Mamdani** method.
- In 1975, Professor Ebrahim Mamdani of London University built one of the first fuzzy systems to control a steam engine and boiler combination. He applied a set of fuzzy rules supplied by experienced human operators.
- The Mamdani-style fuzzy inference process is performed in four steps:

  1. Fuzzification of the input variables

  2. Rule evaluation (inference)

  3. Aggregation of the rule outputs (composition)

  4. Defuzzification

Example: fuzzy control

Crisp Values → **Fuzzifier** → **Inference Engine** → **Defuzzilier** → Crisp Values

Membership Functions

Fuzzy (IF THEN) Rules
Fuzzy Variables
Linguistic Variables
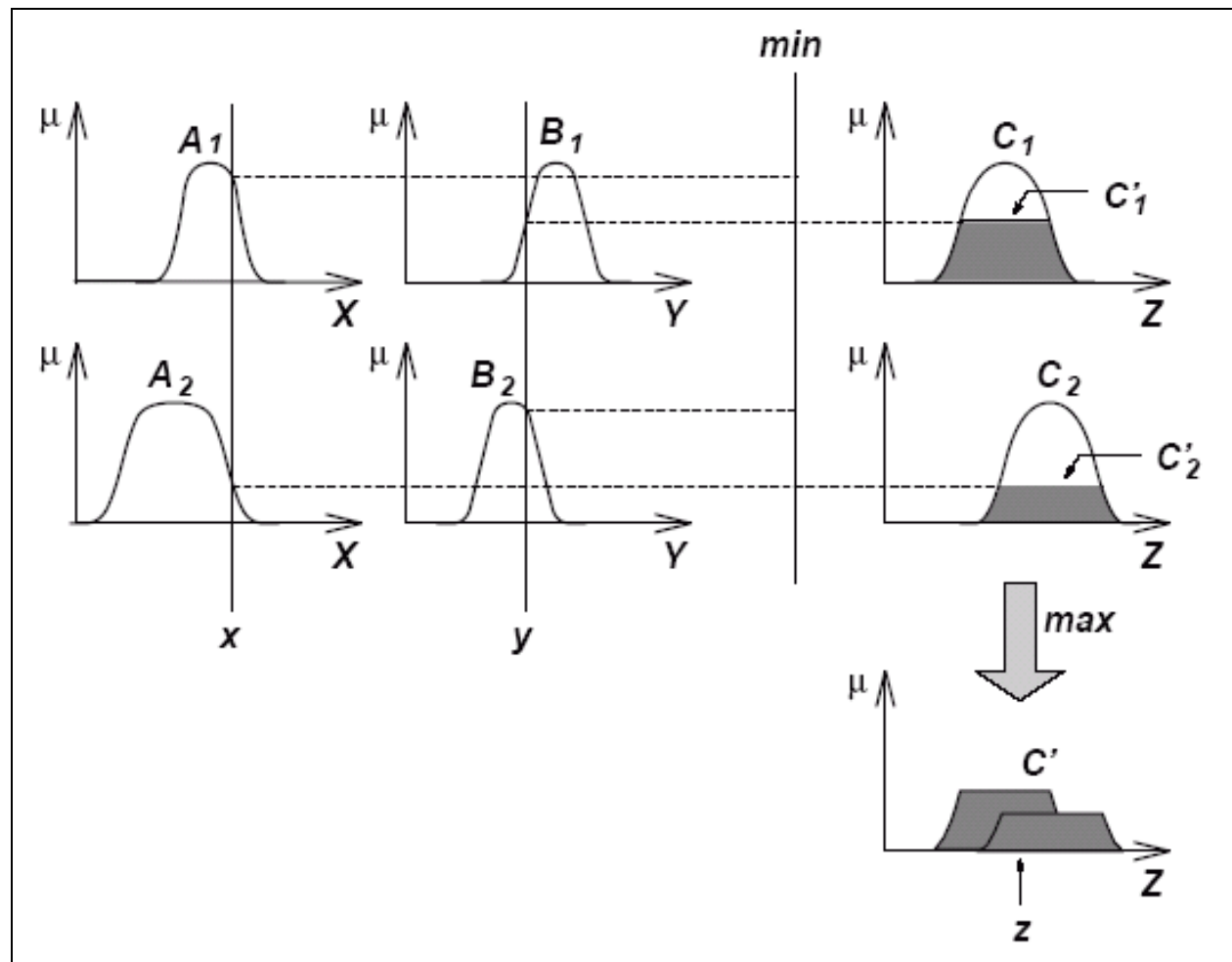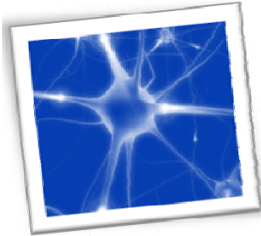
Mamdani Fuzzy Models
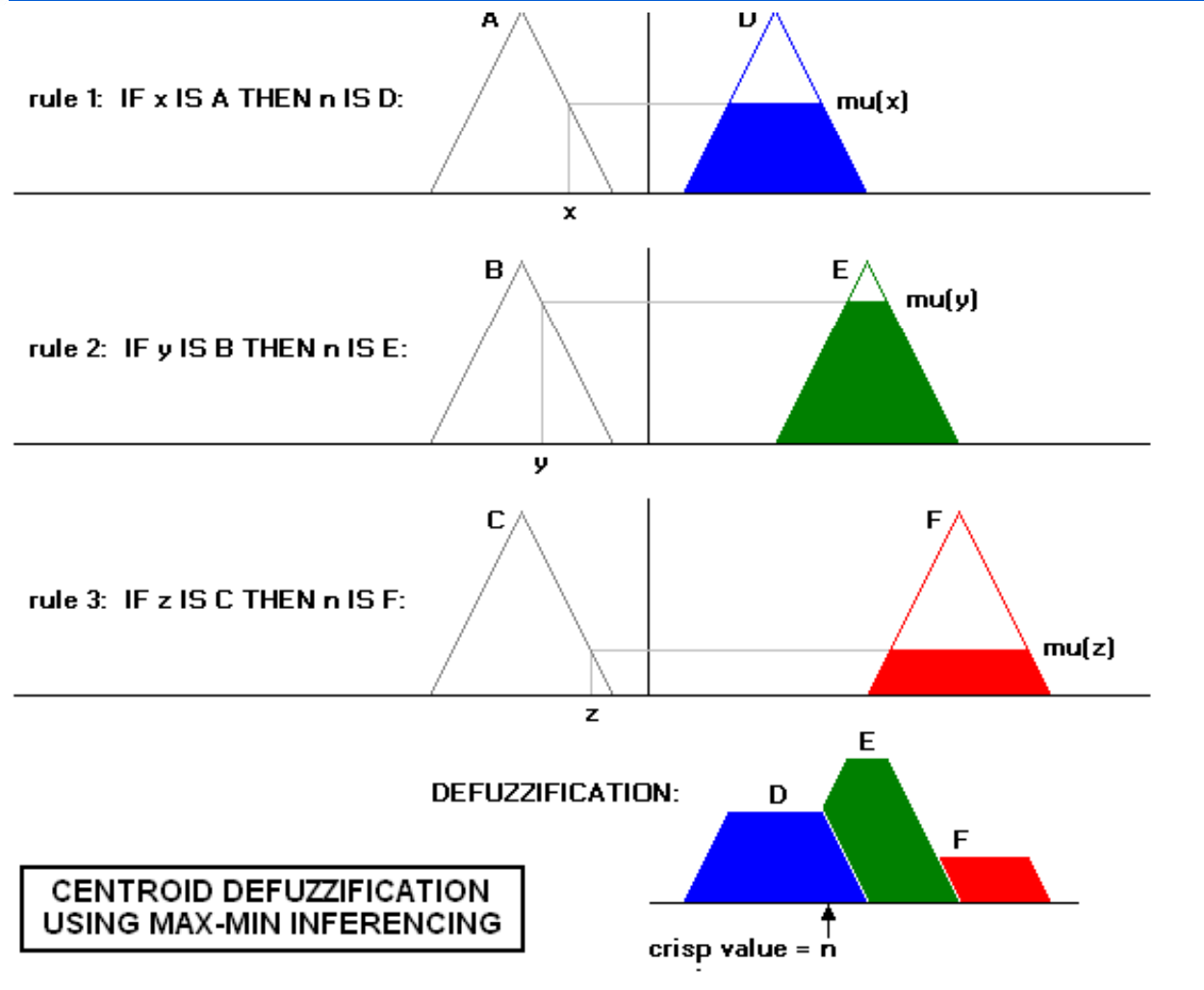
Mamdani composition of three SISO fuzzy outputs
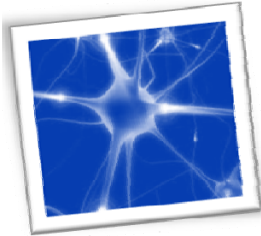http://en.wikipedia.org/wiki/Fuzzy_control_system

Two-rule Mamdani with min and max operators

The mamdani FIS using **min** and **max** for **T-norm** and **S-norm**, and subject to two crisp inputs x and y

Two-rule Mamdani FIS with max and product operators
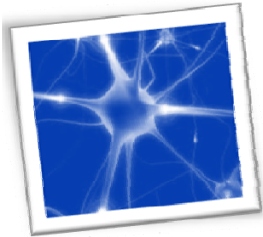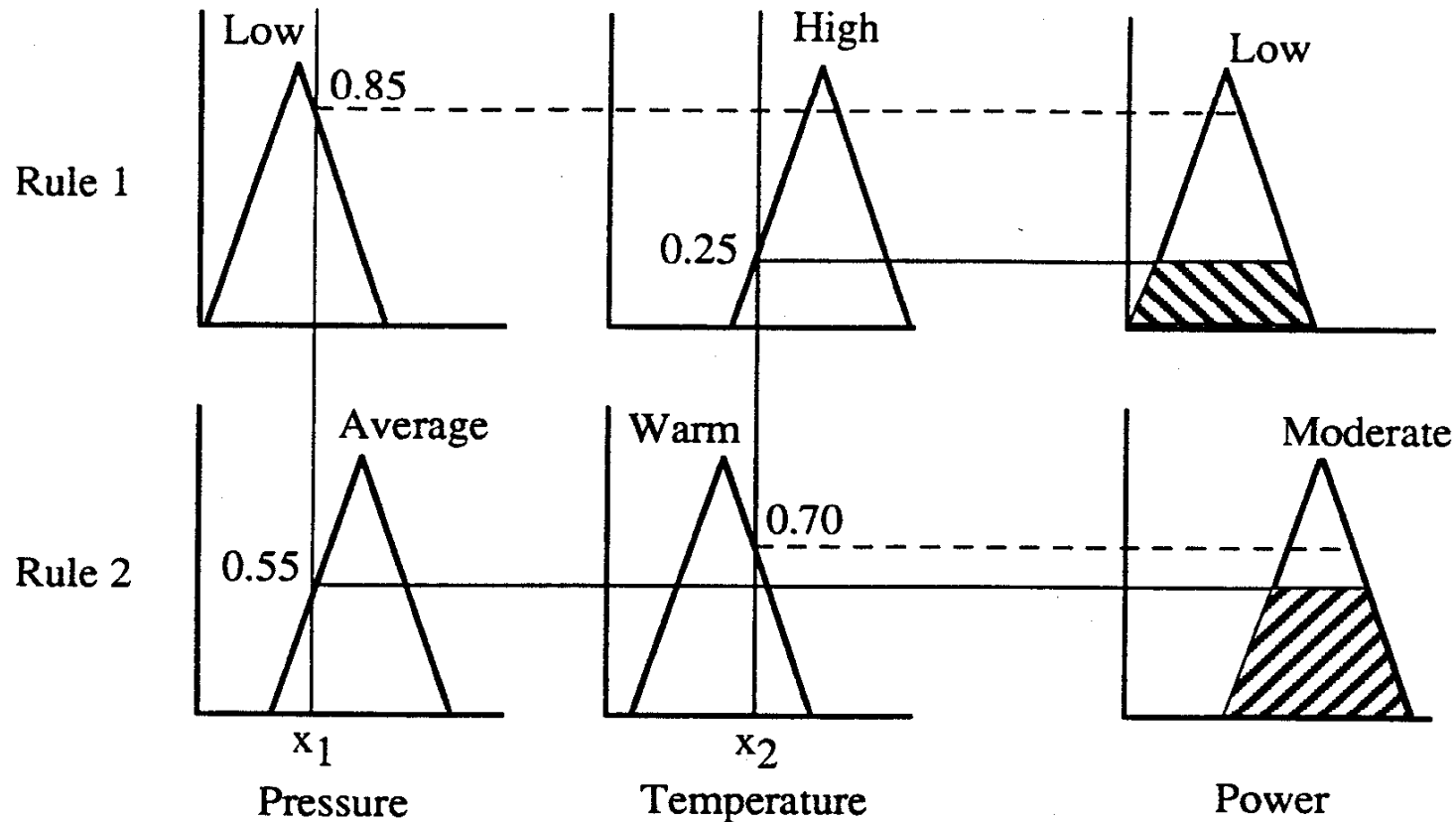
The mamdani FIS using **product** and **max** for **T-norm** and **S-norm**, and subject to two crisp inputs x and y
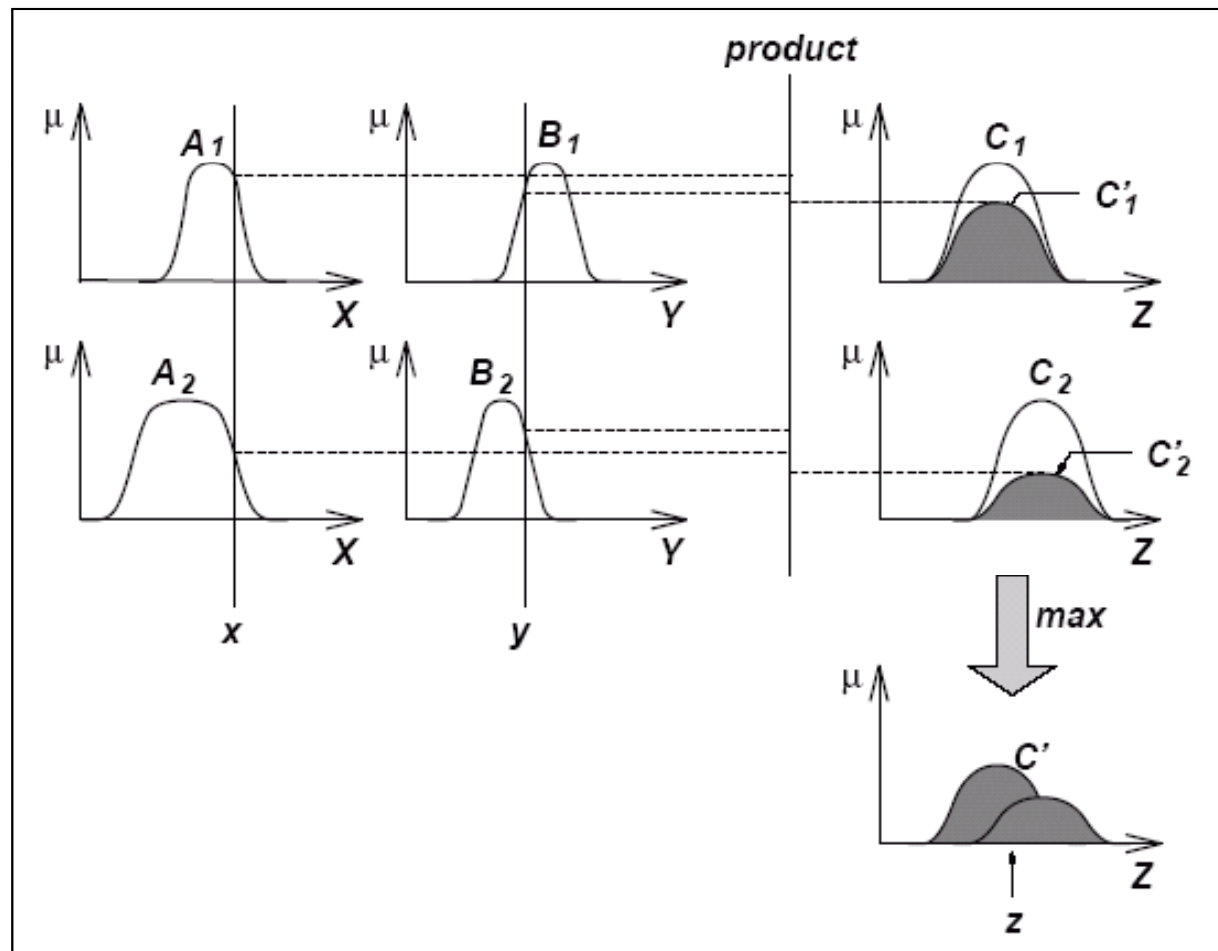
Example

Rule 1: **If pressure is low and temperature is high then power is low**
Rule 2: **If pressure is average and temperature is warm**
            then power is moderate

Mamdani Fuzzy Models

Two-input, one-ouput example:

If x is A_i and y is B_k then z is $C_{m(i,k)}$

y

	B_1	B_2	B_3	B_4
A_1	\	C_2	C_3	C_4
A_2	C_5	C_6	C_7	C_8
A_3	C_9	C_{10}	C_{11}	C_{12}

x

z

Y

	L	M	H	VH
L	Z	L	M	H
M	L	M	H	VH
H	M	H	VH	VH

x

Playing time
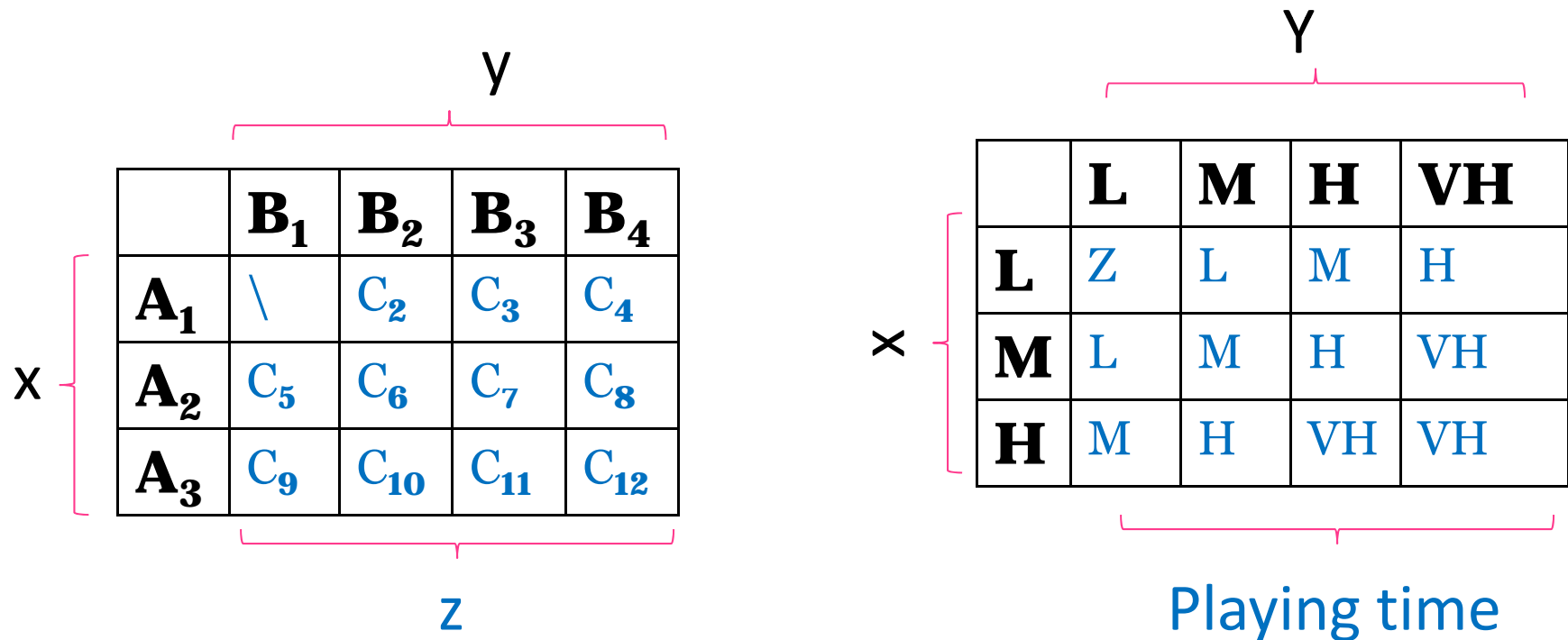
Why Defuzzification Is Needed?

- In many applications we have to use crisp values as inputs for controlling of machines and systems.

- So, we have to use a defuzzifier to convert a fuzzy set to a crisp value.

Defuzzification

- Defuzzification refers to the way a crisp value is extracted from a fuzzy set as a representative value.
- Defuzzification Methods:
  - Centroid of Area
  - Bisector of Area
  - Mean of Max
  - Smallest of Max
  - Largest of Max

Centroid of Area (z_{COA})

$$z_{COA} = \frac{\int_Z \mu_A(z) z \, dz}{\int_Z \mu_A(z) \, dz},$$

- where μ_A is aggregated output MF.
- This is the most widely adopted defuzzification strategy, which is reminiscent of the calculation of expected values of probability distributions.

Bisector of Area (z_{BOA})

- z_{BOA} satisfies

$$\int_{\alpha}^{z_{BOA}} \mu_A(z)\, dz = \int_{z_{BOA}}^{\beta} \mu_A(z)\, dz,$$

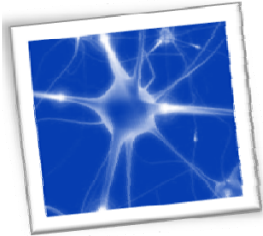$$\alpha = min\{z|z \in Z\} \qquad \beta = max\{z|z \in Z\}$$

- That is, the vertical line z = z_{BOA} partitions the region between z = α, z = β, y = 0 and y = $\mu_A(z)$ into two regions with the same area.

Mean of Maximum (z_{MOM})

- z_{MOM} is the mean of maximizing z at which the MF reaches maximum μ^*. In Symbols,

$$z_{MOM} = \frac{\int_{Z'} z \, dz}{\int_{Z'} dz},$$
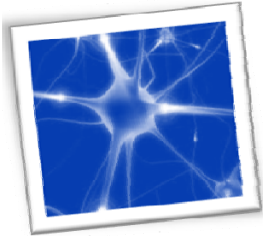
$$Z' = \{z | \mu_A(z) \in \mu^*\}$$

- In particular, if $\mu_A(z)$ has a single maximum at $z = z^*$, then the $z_{MOM} = z^*$.
- Moreover, if $\mu_A(z)$ reaches its maximum whenever

$$z \in \left[z_{left}, z_{right} \right]$$

then

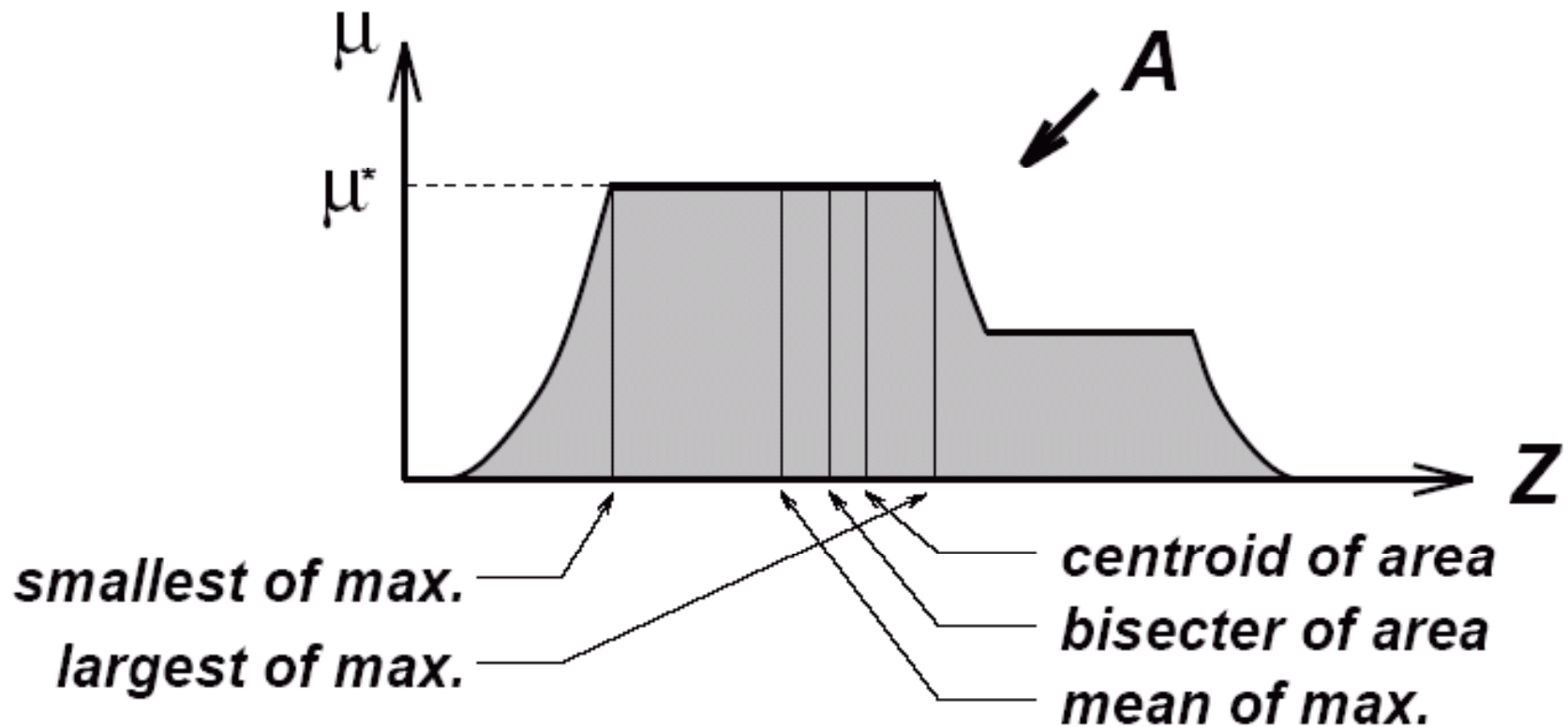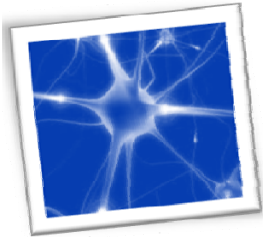$$z_{MOM} = (z_{left} + z_{right})/2$$

Smallest of Maximum (z_{SOM}) and Largest of Maximum (z_{LOM})

- z_{SOM} is the minimum (in terms of magnitude) of the maximizing z.
- z_{LOM} is the maximum (in terms of magnitude) of the maximizing z.

- Because of their obvious bias, z_{SOM} and z_{LOM} are not used as often as the other three defuzzification methods.

Defuzzification
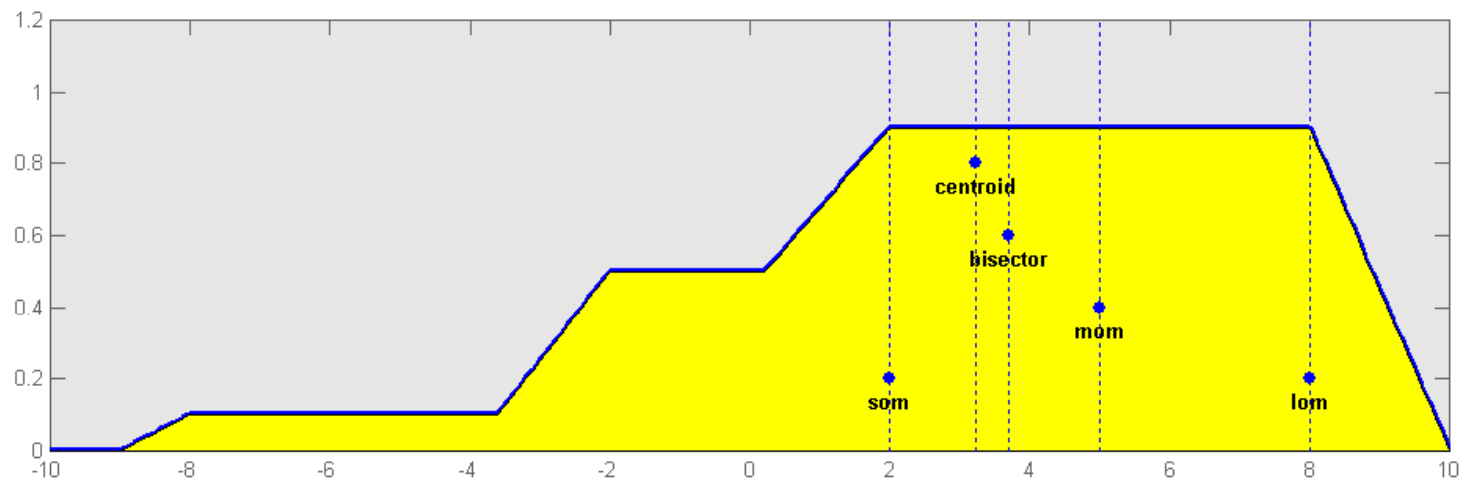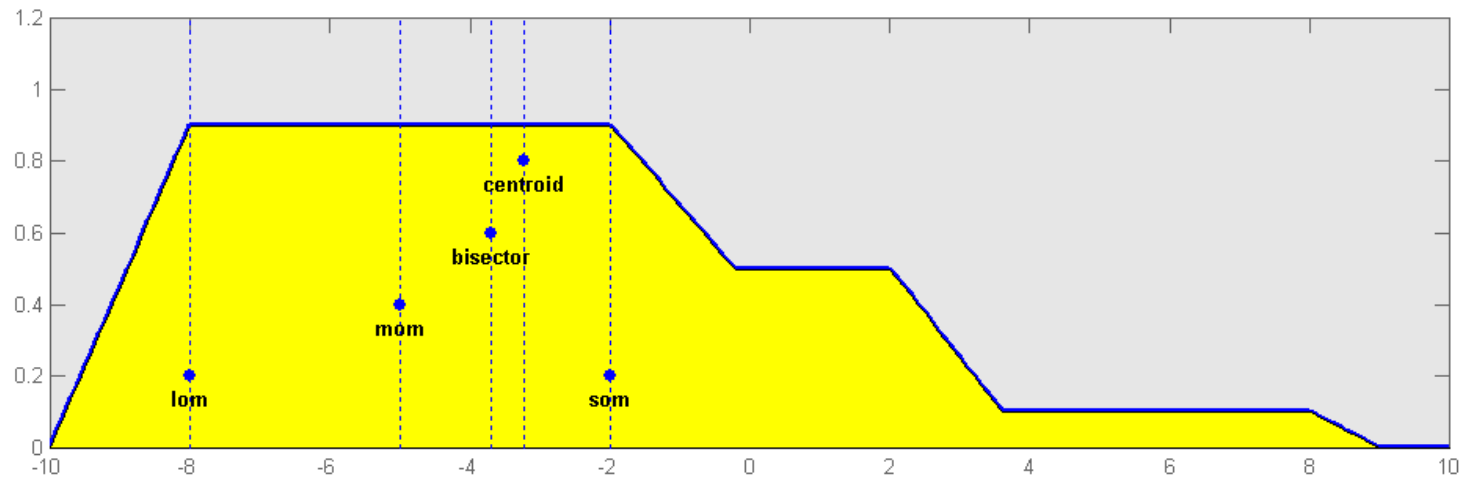
Example

Example: Single-input Single-output Mamdani Fuzzy Model

- An example of a single-input single-output Mamdani fuzzy model with three rules can be expressed as
  - If X is small then Y is small.
  - If X is medium then Y is medium.
  - If X is large then Y is large.
- Figure (next slide) plots the membership functions of input X and output Y.
- With max-min composition and centroid defuzzification, we can find the overall input-output curve, as shown in the Figure.

Single-input single-output Mamdani Fuzzy Model

a) MFs of the input and output
b) Overall input-output curve

Example: Two-input Single-output Mamdani Fuzzy Model
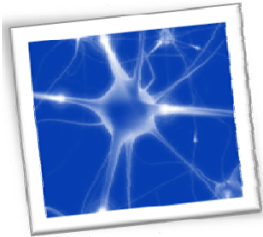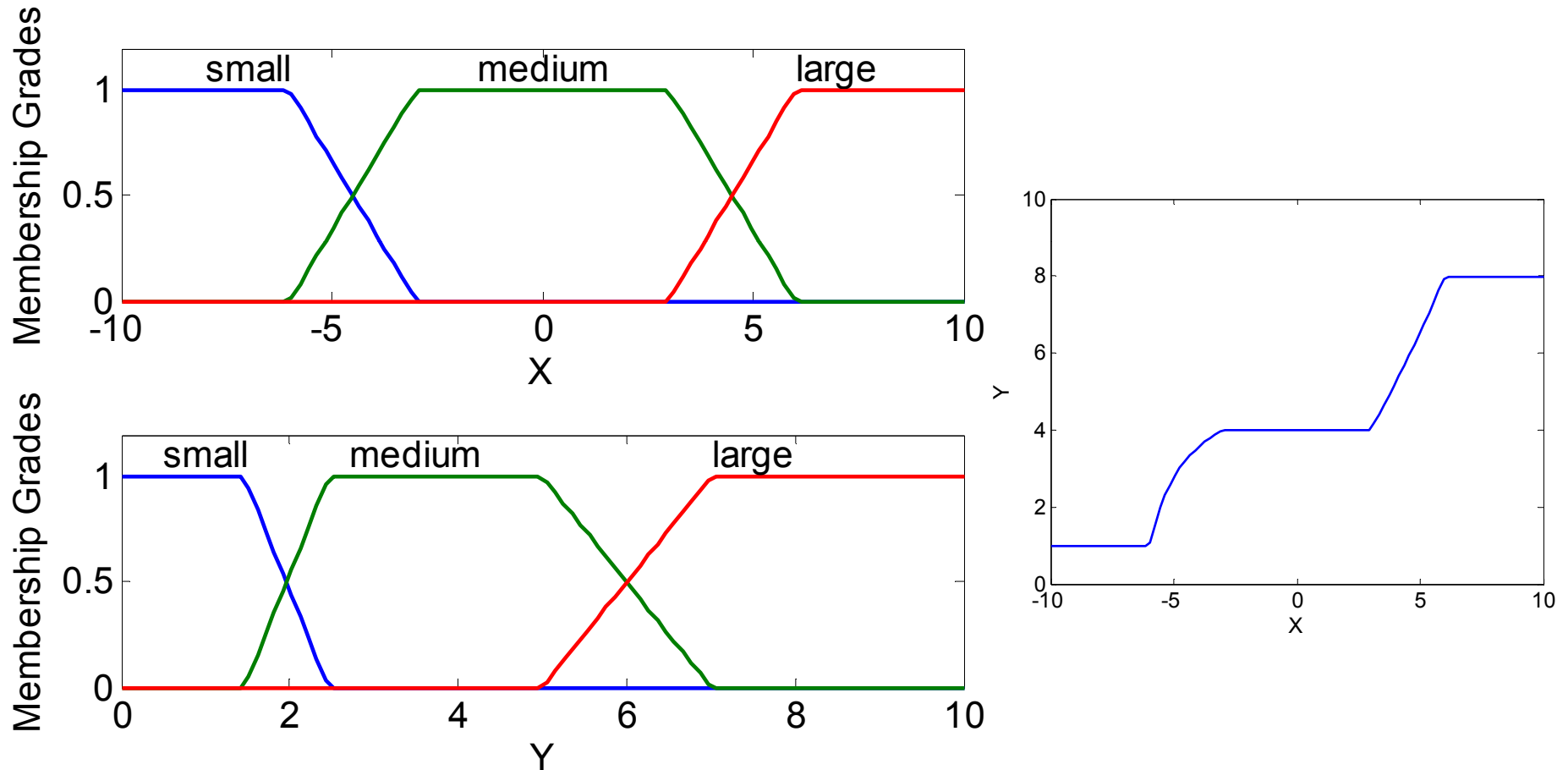
- An example of a two-input single-output Mamdani fuzzy model with four rules can be expressed as
  - If X is small and Y is small then Z is negative large.
  - If X is small and Y is large then Z is negative small.
  - If X is large and Y is small then Z is positive small.
  - If X is large and Y is large then Z is positive large.
- Figure (next slide) plots the membership functions of input X and Y and output Z.
- With max-min composition and centroid defuzzification, we can find the overall input-output surface, as shown in the Figure.

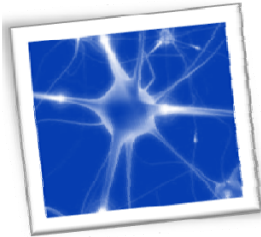# Two-input Single-output Mamdani Fuzzy Model

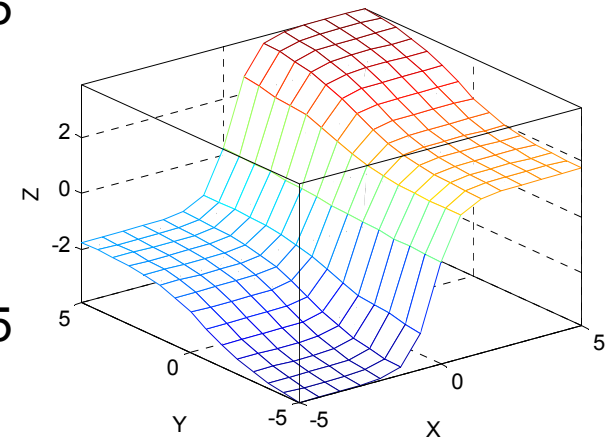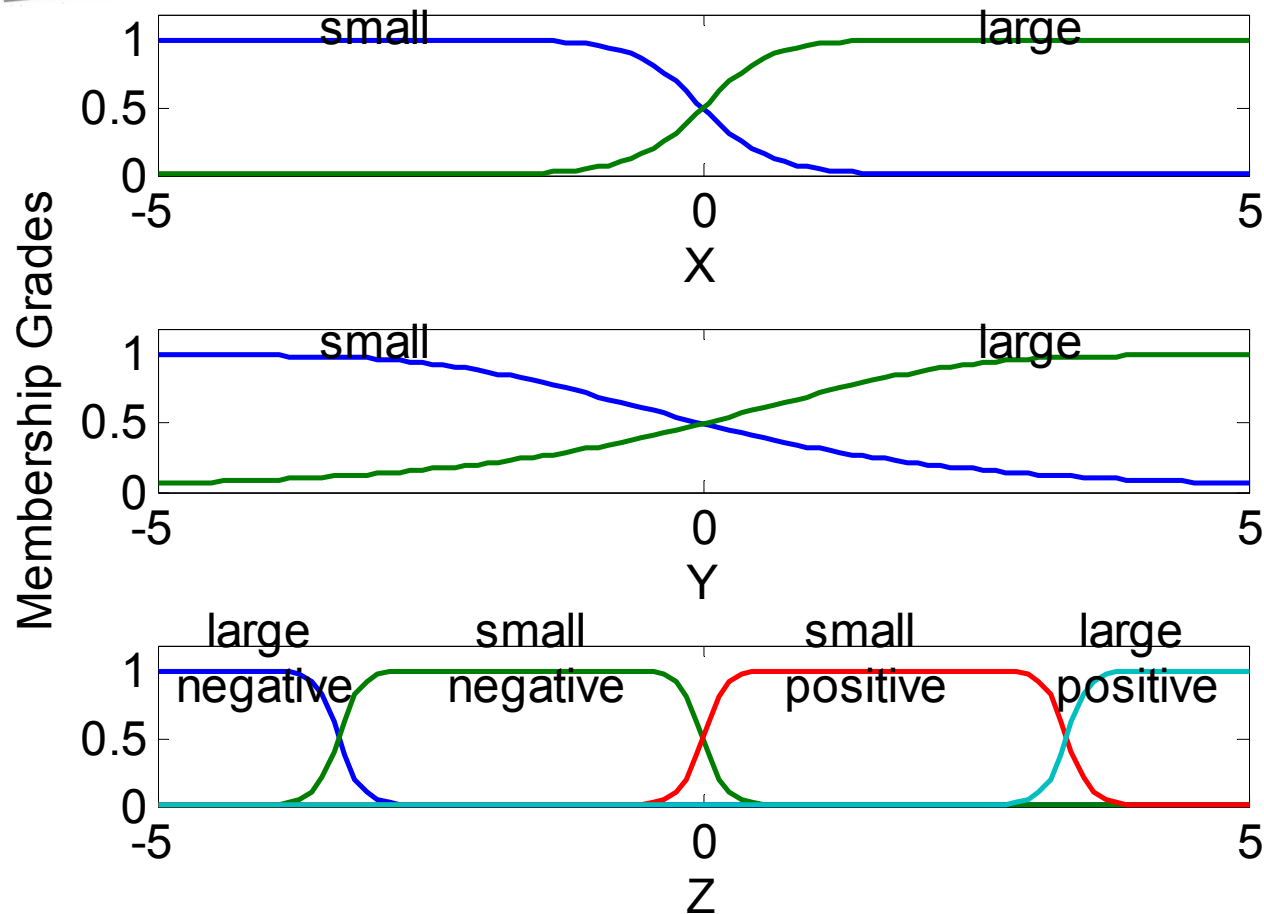a) MFs of the inputs and output
b) Overall input-output curve

Operations in Mamdani Fuzzy Model

- To completely specify the operation of a Mamdani fuzzy inference system, we need to assign a function for each of the following operators:
  - **AND operator** (usually T-norm) for calculating the firing strength of a rule with AND'ed antecedents.
  - **OR operator** (usually T-conorm) for calculating the firing strength of a rule with OR'ed antecedents.
  - **Implication operator** (usually T-norm) for calculating qualified consequent MFs based on given firing strength.
  - **Aggregate operator** (usually T-conorm) for aggregating qualified consequent MFs to generate an overall output MF.
  - **Defuzzification** operator for transforming an output MF to a crisp single output value.

Yet more on Mamdani model

- One such example is to use product for the implication operator and Point-wise summation (sum) for the aggregation operator (Note that sum is not even a T-conorm operator !).

- Advantage:
  - The Final crisp output via centroid defuzzification is equal to the weighted average of the centroids of consequent MFs.
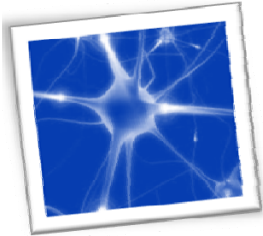
Theorem:
Computation shortcut for Mamdani FIS
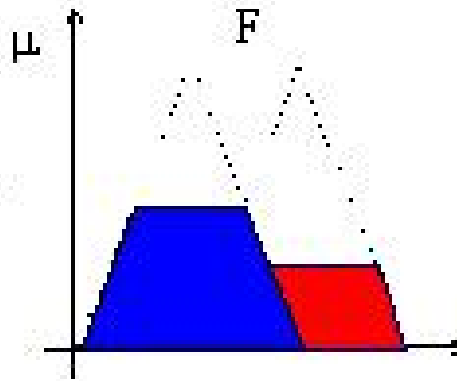
Defuzzification can be computationally expensive. What is the center of area of this fuzzy set?

Theorem 4.1: If we use the product T-norm, and the summation S-norm (which is not really an S-norm), and centroid defuzzification, then the crisp output z is:

$$z = \frac{\sum w_i a_i z_i}{\sum w_i a_i}$$

w_i = firing strength (input MF value)

a_i = consequent MF area

z_i = consequent MF centroid

a_i and z_i can be calculated ahead of time!

Example

We examine a simple two-input one-output problem that includes three rules:

Rule: 1	IF x is A3		OR	y is B1	THEN	z is C1
Rule: 2	IF x is A2		AND	y is B2	THEN	z is C2
Rule: 3	IF x is A1				THEN	z is C3

Real-life example for these kinds of rules:

Rule: 1   IF project_funding is adequate OR project_staffing is small THEN
          risk is low

Rule: 2   IF project_funding is marginal AND project_staffing is large THEN
          risk is normal

Rule: 3   IF project_funding is inadequate   THEN
          risk is high
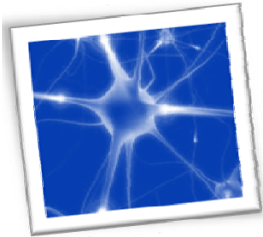
Step 1: Fuzzification

- The first step is to take the crisp inputs, x1 and y1 (*project funding* and *project staffing*), and determine the degree to which these inputs belong to each of the appropriate fuzzy sets.

$$\mu_{(x=A1)} = 0.5$$
$$\mu_{(x=A2)} = 0.2$$
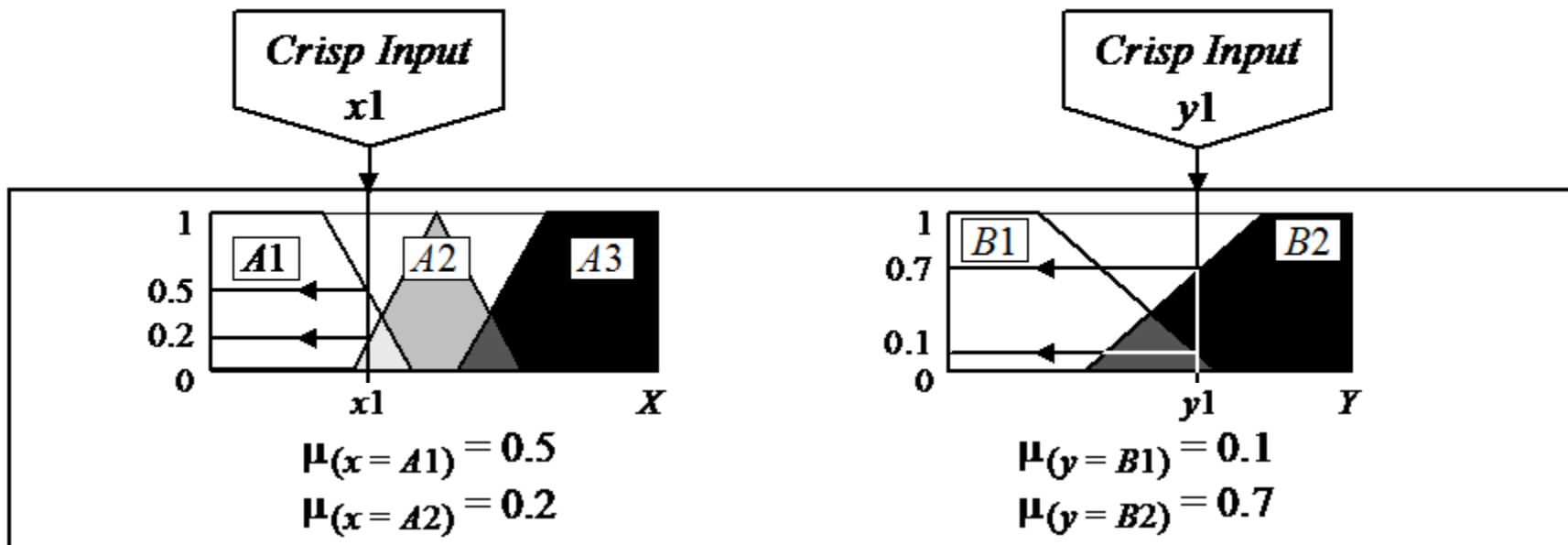
$$\mu_{(y=B1)} = 0.1$$
$$\mu_{(y=B2)} = 0.7$$

Step 2: Rule Evaluation

- The second step is to take the fuzzified inputs, $\mu_{(x=A1)} = 0.5$, $\mu_{(x=A2)} = 0.2$, $\mu_{(y=B1)} = 0.1$ and $\mu_{(y=B2)} = 0.7$, and apply them to the antecedents of the fuzzy rules.
- If a given fuzzy rule has multiple antecedents, the fuzzy operator (AND or OR) is used to obtain a single number that represents the result of the antecedent evaluation.

RECALL: To evaluate the disjunction of the rule antecedents, we use the **OR** fuzzy operation. Typically, fuzzy expert systems make use of the classical fuzzy operation union:

$$\mu_{A \cup B}(x) = \max\left[\mu_A(x),\ \mu_B(x)\right]$$

Similarly, in order to evaluate the conjunction of the rule antecedents, we apply the **AND** fuzzy operation intersection:

$$\mu_{A \cap B}(x) = \min\left[\mu_A(x),\ \mu_B(x)\right]$$

Rule 1: IF x is A3 (0.0) OR y is B1 (0.1) THEN z is C1 (0.1)

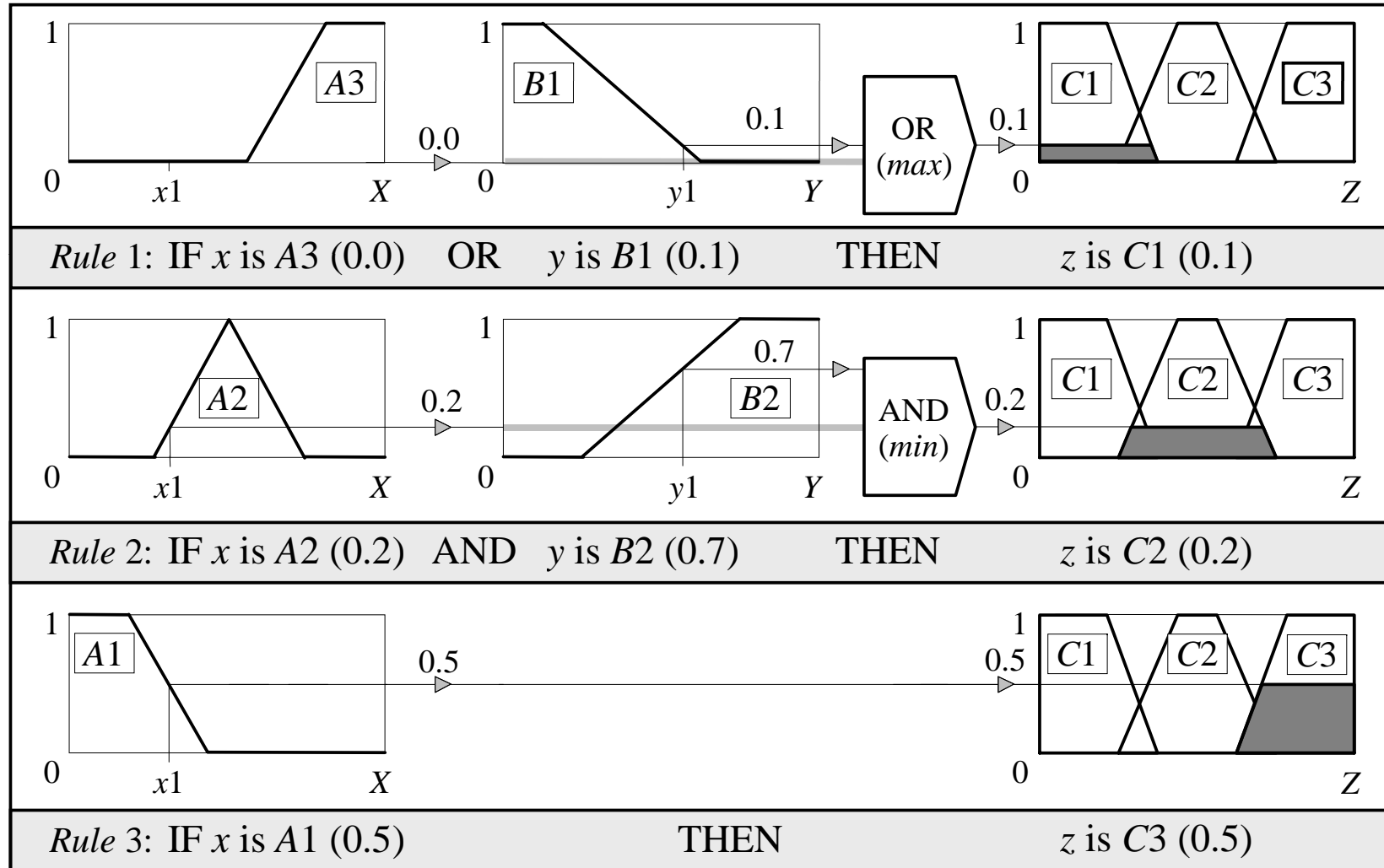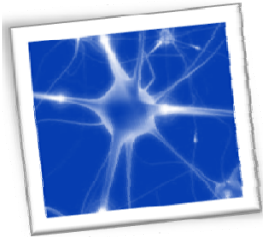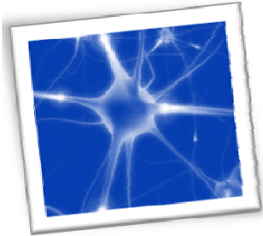Rule 2: IF x is A2 (0.2) AND y is B2 (0.7) THEN z is C2 (0.2)

Rule 3: IF x is A1 (0.5) THEN z is C3 (0.5)

Step 2: Rule Evaluation

- Now the result of the antecedent evaluation can be applied to the membership function of the consequent.
- The most common method is to cut the consequent membership function at the level of the antecedent truth. This method is called **clipping** (alpha-cut).
  - Since the top of the membership function is sliced, the clipped fuzzy set loses some information.
  - However, clipping is still often preferred because it involves less complex and faster mathematics, and generates an aggregated output surface that is easier to defuzzify.
- While clipping is a frequently used method, **scaling** offers a better approach for preserving the original shape of the fuzzy set.
  - The original membership function of the rule consequent is adjusted by multiplying all its membership degrees by the truth value of the rule antecedent.
  - This method, which generally loses less information, can be very useful in fuzzy expert systems.

Degree of Membership

1.0

C2

0.2

0.0

z

clipping

Degree of Membership
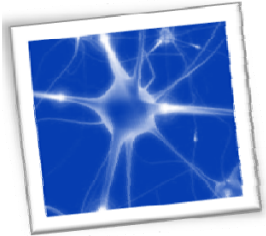
1.0

C2

0.2

0.0

z

scaling
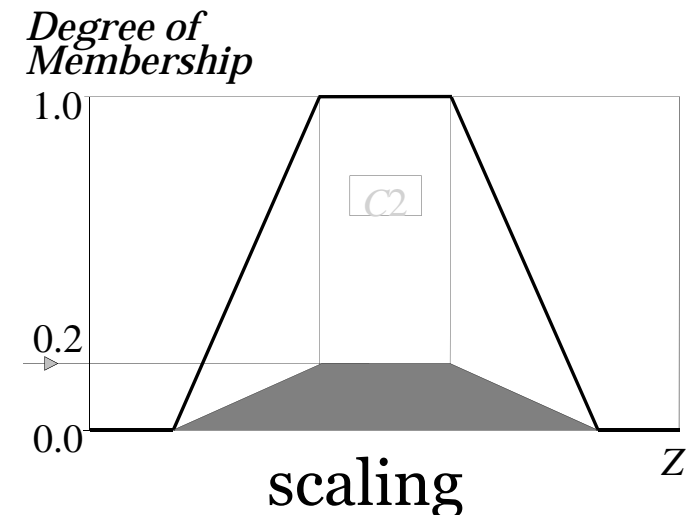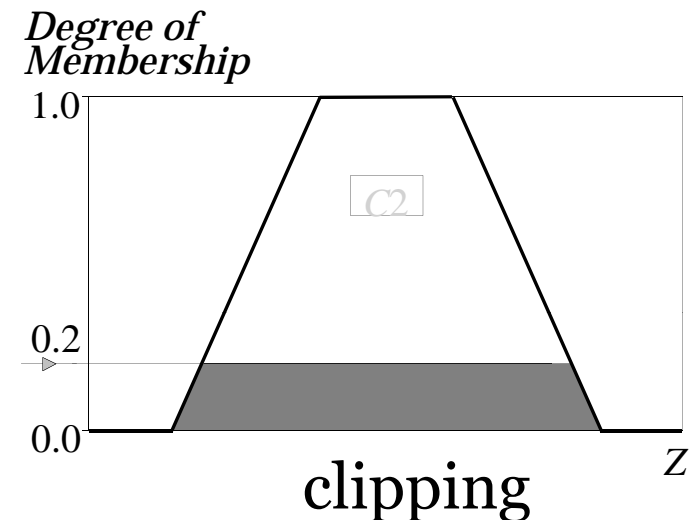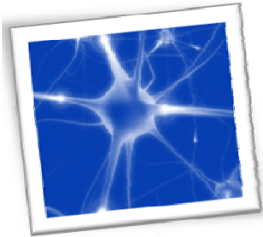
Step 3: Aggregation of the Rule Outputs

- Aggregation is the process of unification of the outputs of all rules.
- We take the membership functions of all rule consequents previously clipped or scaled and combine them into a single fuzzy set.
- The input of the aggregation process is the list of clipped or scaled consequent membership functions, and the output is one fuzzy set for each output variable.

|			
z is C1 (0.1)	z is C2 (0.2)	z is C3 (0.5)	Σ
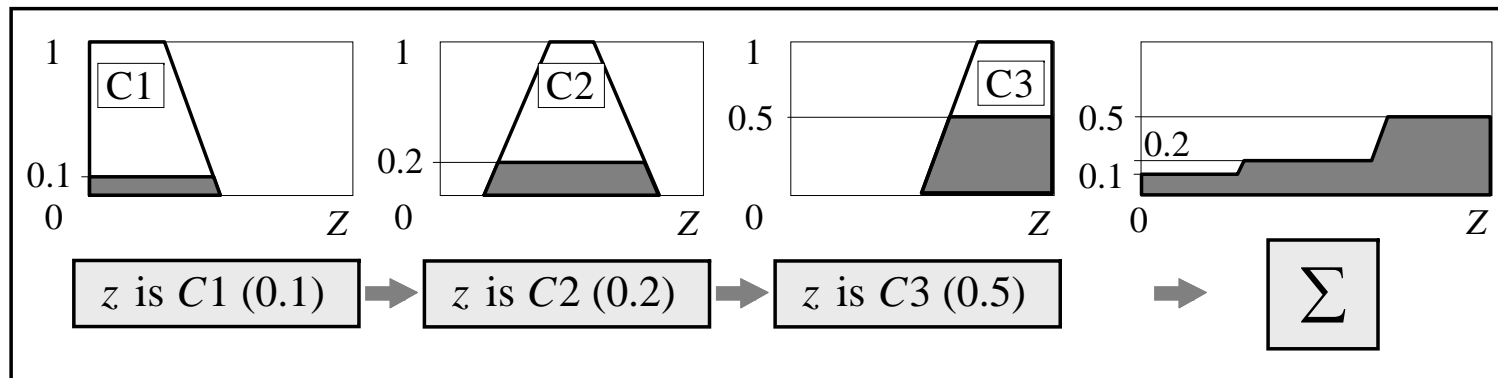
Step 4: Defuzzification

- The last step in the fuzzy inference process is defuzzification.

- Fuzziness helps us to evaluate the rules, but the final output of a fuzzy system has to be a crisp number.

- The input for the defuzzification process is the aggregate output fuzzy set and the output is a single number.

- There are several defuzzification methods, but probably the most popular one is the **centroid technique**. It finds the point where a vertical line would slice the aggregate set into two equal masses. Mathematically this **centre of gravity** (**COG**) can be expressed as:

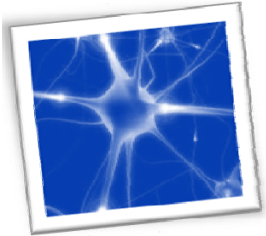$$COG = \frac{\int_a^b \mu_A(x)\, x\, dx}{\int_a^b \mu_A(x)\, dx}$$
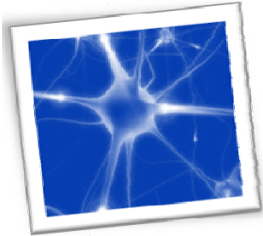
Step 4: Defuzzification

- Centroid defuzzification method finds a point representing the centre of gravity of the aggregated fuzzy set *A*, on the interval [*a, b*].
- A reasonable estimate can be obtained by calculating it over a sample of points.
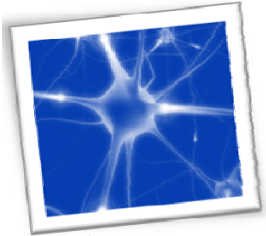
Degree of
Membership

67.4

z
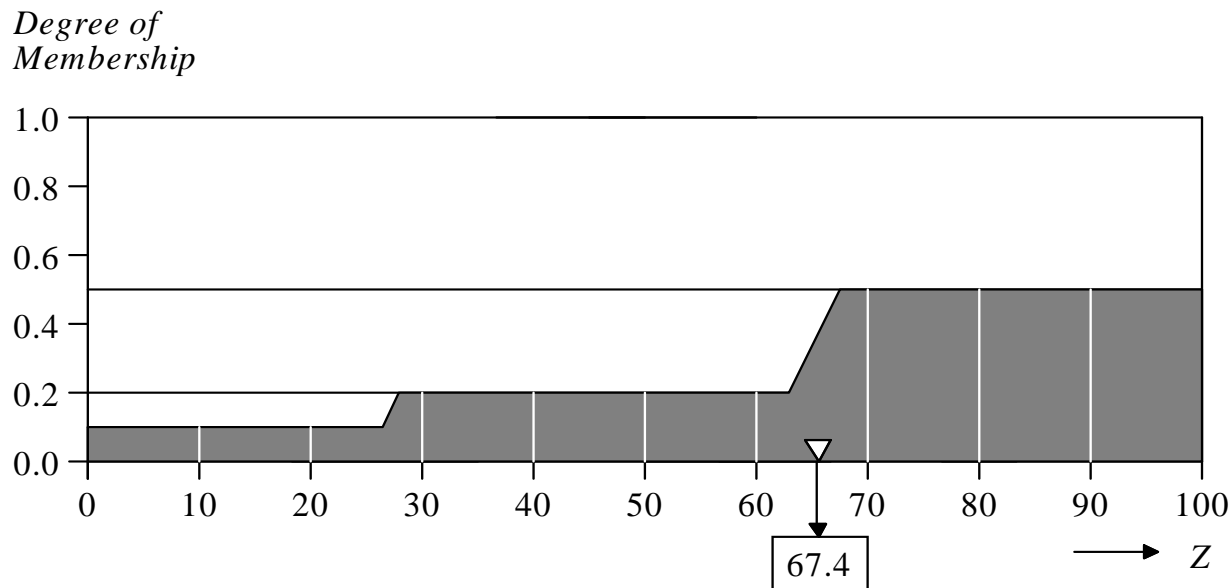
$$COG = \frac{(0+10+20)\times 0.1 + (30+40+50+60)\times 0.2 + (70+80+90+100)\times 0.5}{0.1+0.1+0.1+0.2+0.2+0.2+0.2+0.5+0.5+0.5+0.5} = 67.4$$
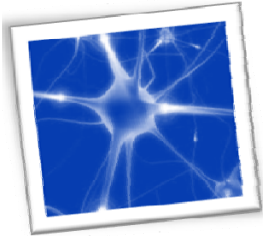
Sugeno Fuzzy Inference

- Mamdani-style inference, as we have just seen, requires us to find the centroid of a two-dimensional shape by integrating across a continuously varying function. In general, this process is not computationally efficient.

- Michio Sugeno suggested to use a single spike, a singleton, as the membership function of the rule consequent.

- A singleton, or more precisely a fuzzy singleton, is a fuzzy set with a membership function that is unity at a single particular point on the universe of discourse and zero everywhere else.

Sugeno Fuzzy Model

- Also known as the TSK fuzzy model (proposed by Takagi, Sugeno, and Kang)
- For developing a systematic approach to generating fuzzy rules from a given input-output data set
- A typical fuzzy rule in a Sugeno fuzzy model:

  if *x is A and y is B then z = f (x, y)*

- *A* and *B:* fuzzy sets
- *z = f(x, y)*: a crisp function (usually polynomial in the input variables *x and y*)

Sugeno Fuzzy Model

- Sugeno-style fuzzy inference is very similar to the Mamdani method.
- Sugeno changed only a rule consequent: instead of a fuzzy set, he used a mathematical function of the input variable.
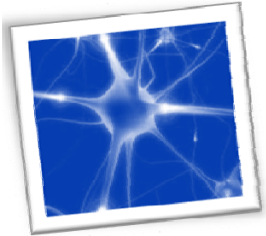- The format of the **Sugeno-style fuzzy rule** is

  IF x **is** A **AND** y **is** B **THEN** z **is** $f(x, y)$

  where:
    - x, y and z are linguistic variables;
    - A and B are fuzzy sets on universe of discourses X and Y, respectively;
    - $f(x, y)$ is a mathematical function.

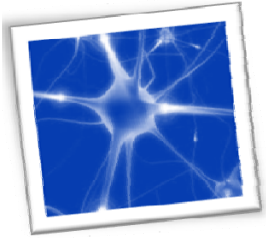- The most commonly used **zero-order Sugeno fuzzy model** applies fuzzy rules in the following form:

  IF x **is** A **AND** y **is** B **THEN** z **is** k

- where k is a constant.
- In this case, the output of each fuzzy rule is constant and all consequent **membership functions** are represented by **singleton spikes.**

So …

- **First-order Sugeno fuzzy model**: $f(x, y)$ *is a first-order polynomial*

- **zero-order Sugeno fuzzy model**: f is a constant

  - a special case of the <u>Mamdani fuzzy inference system</u>, in which each rule's consequent is specified by a fuzzy singleton;

  - or a special case of the <u>Tsukamoto fuzzy model</u> (to be introduced next) in which each rule's consequent is specified by an MF of a step function center at the constant

Sugeno Fuzzy Models
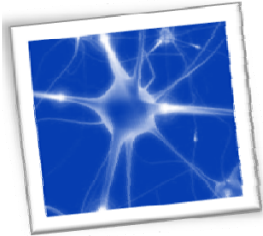
The output is a weighted average:

$$z = \frac{\sum \mu_{A_i, B_k}(x, y) f_{m(i,k)}(x, y)}{\sum \mu_{A_i, B_k}(x, y)}$$

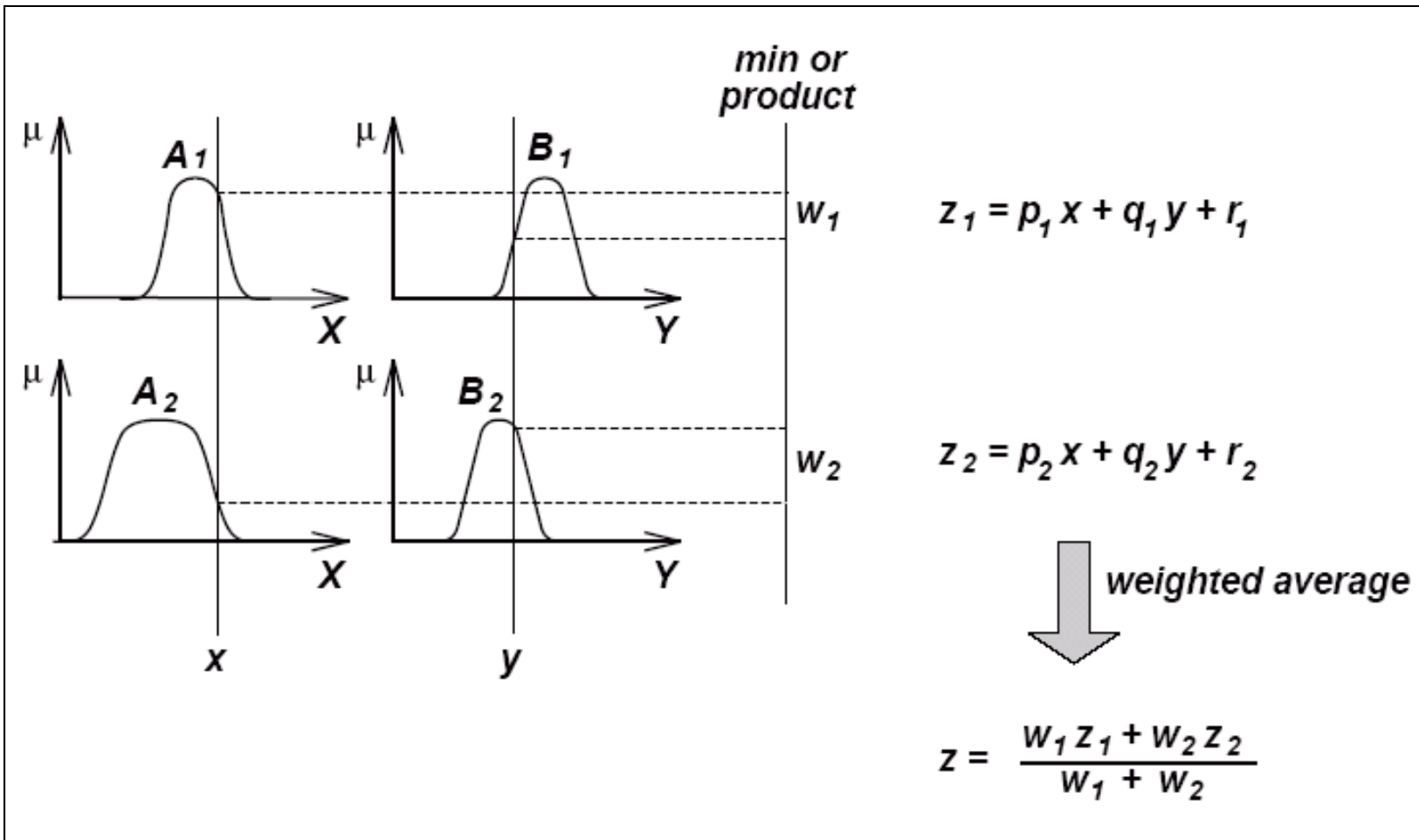Double summation over all i (x MFs) and all k (y MFs)
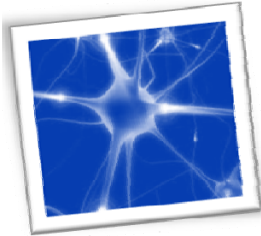
$$= \frac{\sum w_i f_i(x, y)}{\sum w_i}$$

Summation over all i (fuzzy rules)

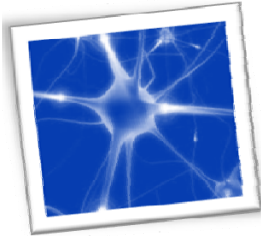where w_i is the firing strength of the i-th output

Fuzzy Reasoning Procedure For a First-Order Sugeno Fuzzy Model
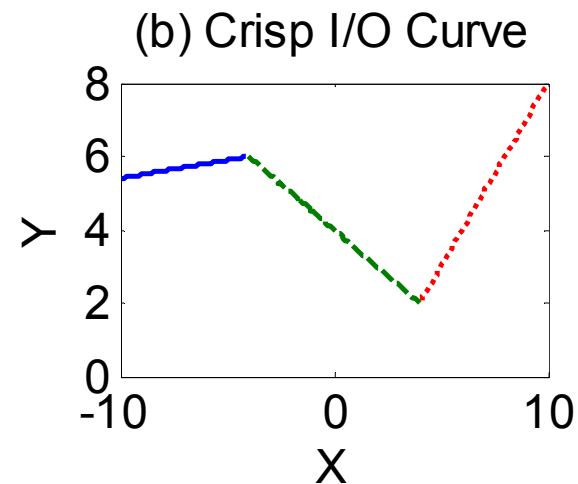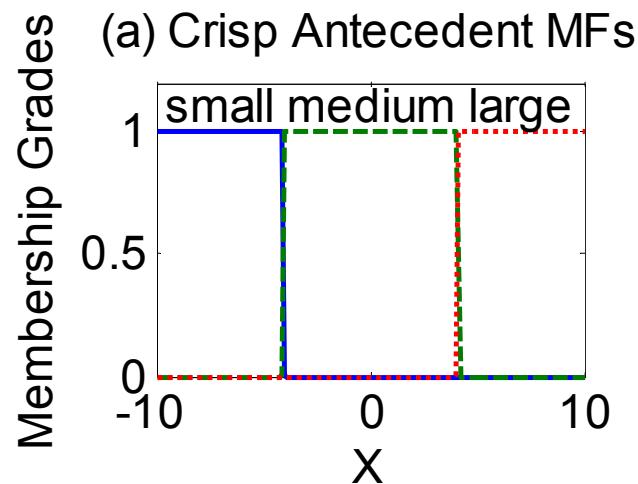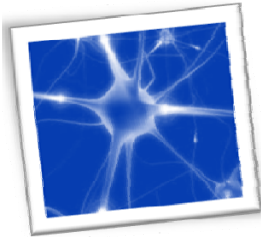
Example: Fuzzy and nonfuzzy rule sets

- An example of a single-input Sugeno fuzzy model:
  - If *X is small then Y = 0.1X + 6.4.*
  - If *X is medium then Y = -0.5X + 4.*
  - If *X is large then Y = X - 2.*

# Example: Fuzzy and nonfuzzy rule set-a comparison

- If "small," "medium," and "large" are nonfuzzy sets with membership functions shown in figure (a), then the overall input-output curve is piecewise linear, as shown in figure (b):

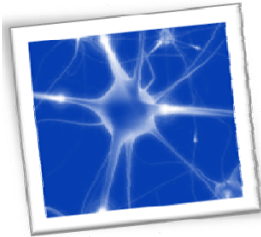(a) Crisp Antecedent MFs

(b) Crisp I/O Curve

Example: Fuzzy and nonfuzzy rule sets

- If we have smooth membership functions [figure (c)] instead, the overall input-output curve [figure (d)] becomes a smoother one:
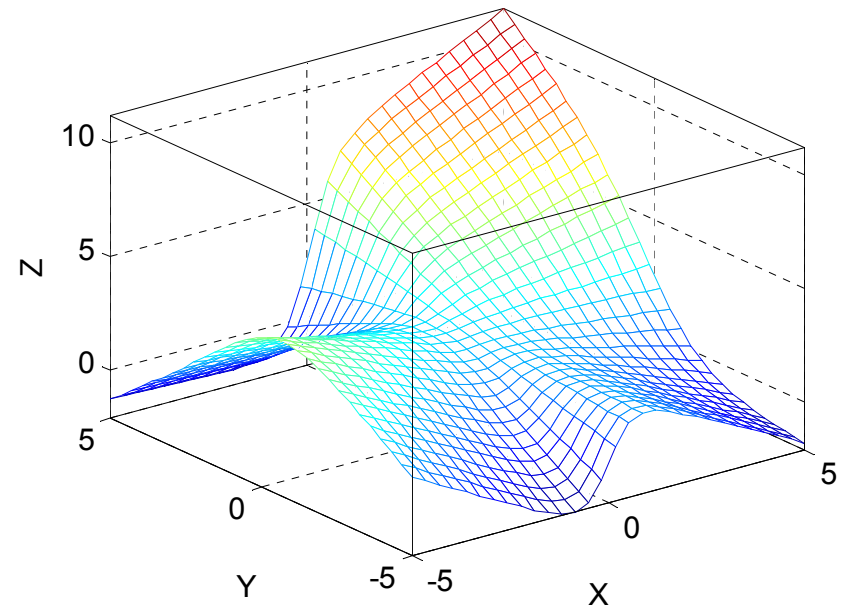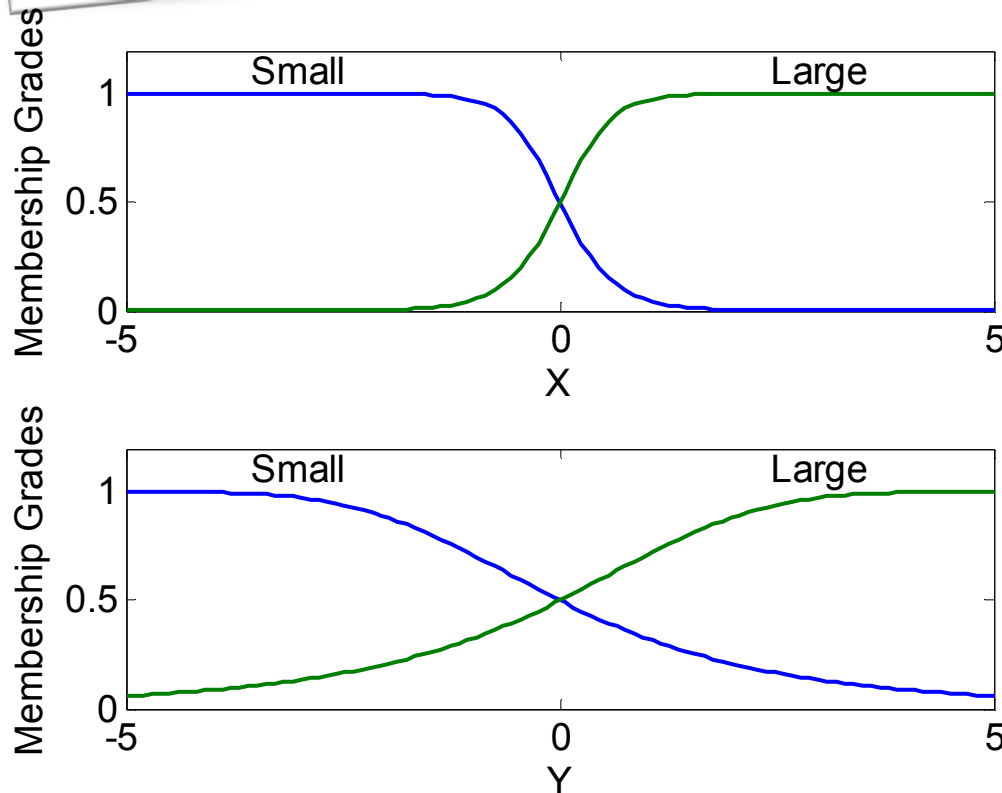

(c) Fuzzy Antecedent MFs

(d) Fuzzy I/O Curve
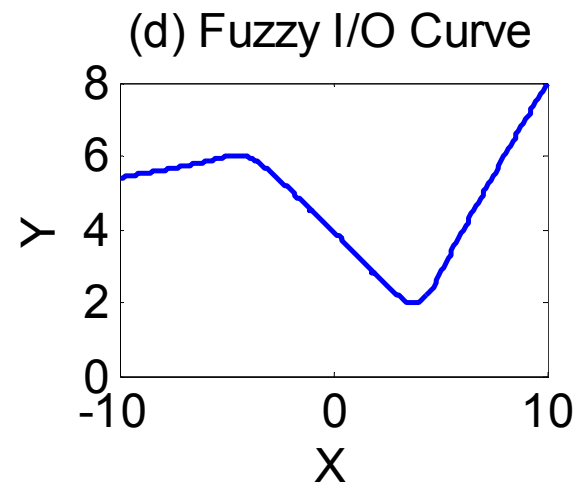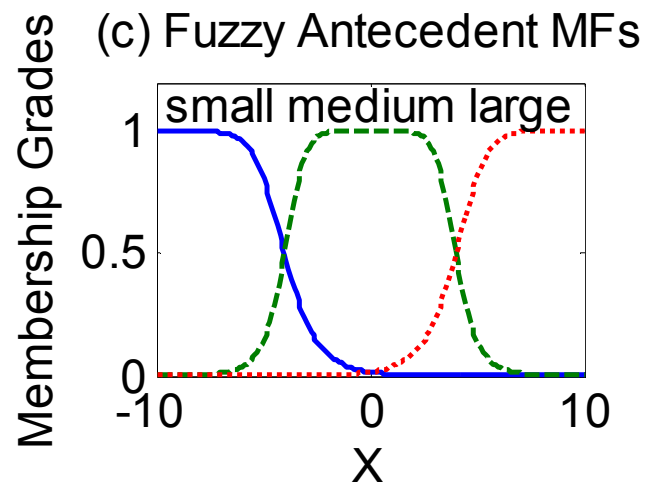
Example: Two-input single-output Sugeno fuzzy model

- An example of a two-input single-output Sugeno fuzzy model with four rules:
  - If *X is small and Y is small then z = -x + y + 1.*
  - If *X is small and Y is large then z = -y + 3.*
  - If *X is large and Y is small then z = -x + 3.*
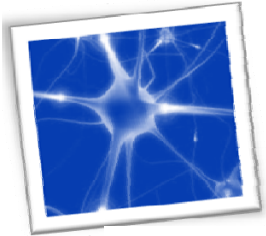 - If *X is large and Y is large then z = x + y + 2.*
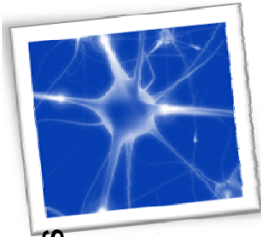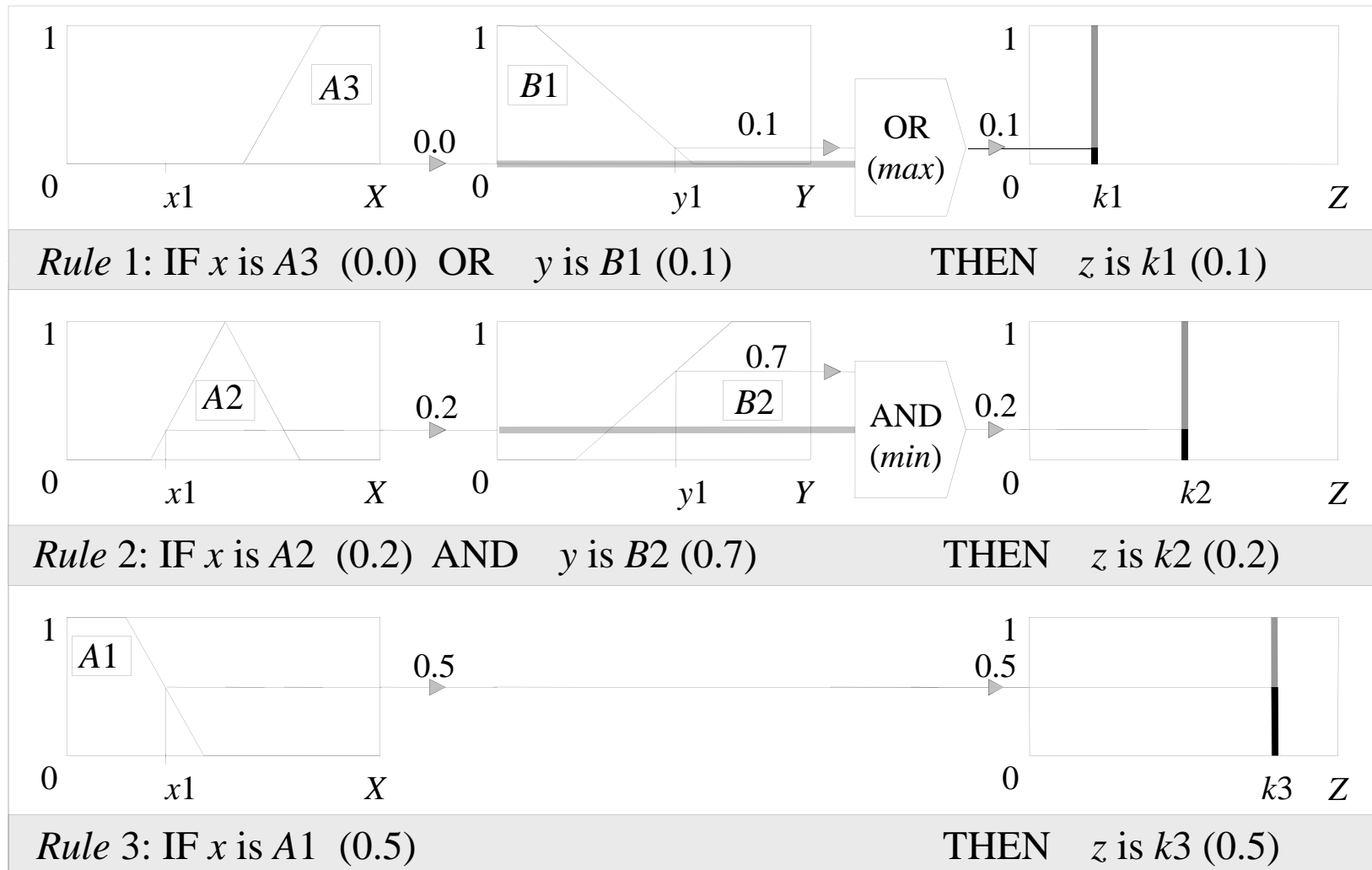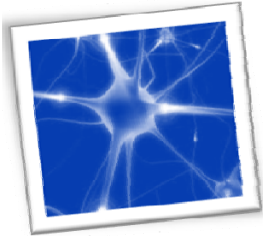
Example: Two-input single-output Sugeno fuzzy model

a) MFs of the inputs and output

b) Overall input-output curve

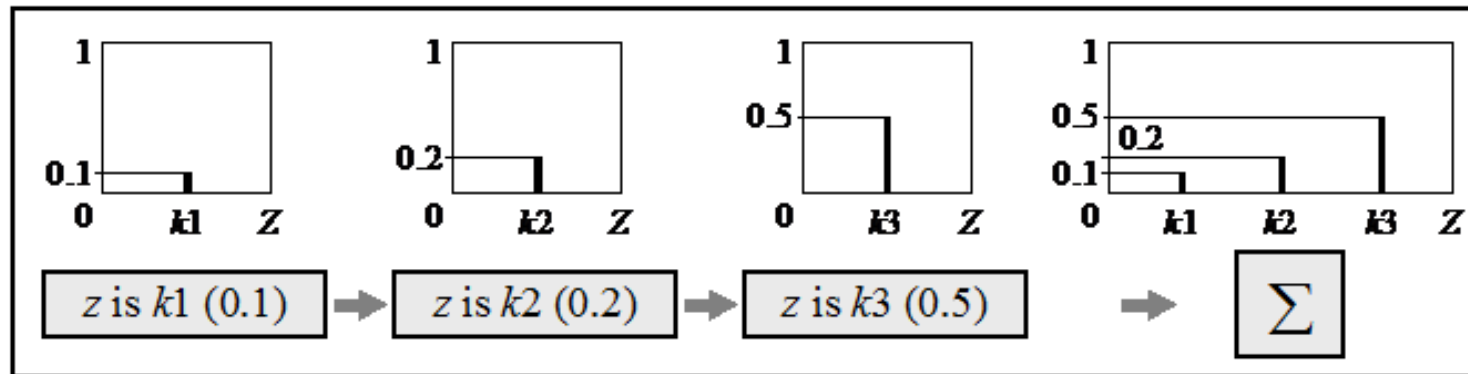- The surface is composed of four planes, each of which is specified by the output equation of a fuzzy rule.

Another Example

1   *A3*   1   *B1*   0.1   OR   0.1   1

0.0

(*max*)

0   0   0
$x1$ X $y1$ Y $k1$ Z

Rule 1: IF x is $A3$ (0.0) OR y is $B1$ (0.1)   THEN z is $k1$ (0.1)

1   1   0.7   1

A2   0.2   *B2*   AND 0.2

(*min*)

0   0   0
$x1$ X $y1$ Y $k2$ Z

Rule 2: IF x is $A2$ (0.2) AND y is $B2$ (0.7)   THEN z is $k2$ (0.2)

1 *A1*   0.5   1
                                                          0.5

0   0
$x1$ X $k3$ Z

Rule 3: IF x is $A1$ (0.5)   THEN z is $k3$ (0.5)
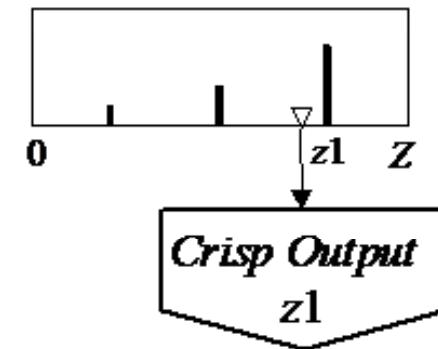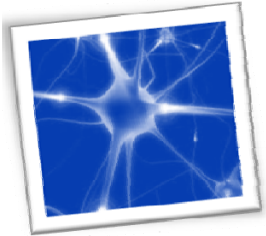
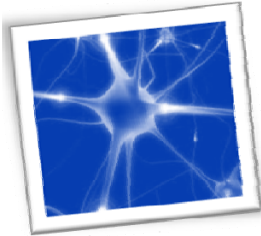# Another Example

COG becomes Weighted Average (WA)

$$WA = \frac{\mu(k1) \times k1 + \mu(k2) \times k2 + \mu(k3) \times k3}{\mu(k1) + \mu(k2) + \mu(k3)} = \frac{0.1 \times 20 + 0.2 \times 50 + 0.5 \times 80}{0.1 + 0.2 + 0.5} = 65$$
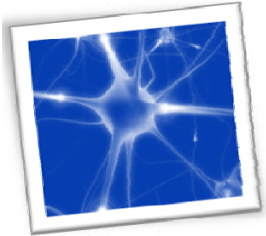
More on Sugeno model

- Unlike the Mamdani fuzzy model, the Sugeno fuzzy model cannot follow the compositional rule of inference strictly in its fuzzy reasoning mechanism
- Without the time-consuming and mathematically intractable defuzzification operation, the Sugeno fuzzy model is by far the most popular candidate for sample data-based fuzzy modeling (we will see an application in ANFIS)

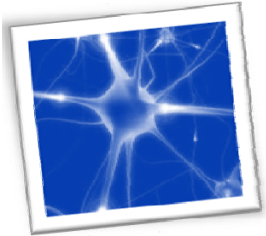# How to make a decision on which method to apply – Mamdani or Sugeno?

■ Mamdani method is widely accepted for capturing expert knowledge. It allows us to describe the expertise in more intuitive, more human-like manner. However, Mamdani-type fuzzy inference entails a substantial computational burden.

■ On the other hand, Sugeno method is computationally effective and works well with optimisation and adaptive techniques, which makes it very attractive in adaptive problems, particularly for dynamic nonlinear systems.
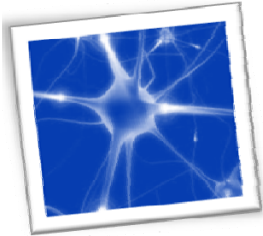
Tsukamoto Fuzzy Model

- In the Tsukamoto fuzzy model, the consequent of each fuzzy if-then rule is represented by a fuzzy set with a **monotonical** MF, as shown in the following figure:


$$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2}$$
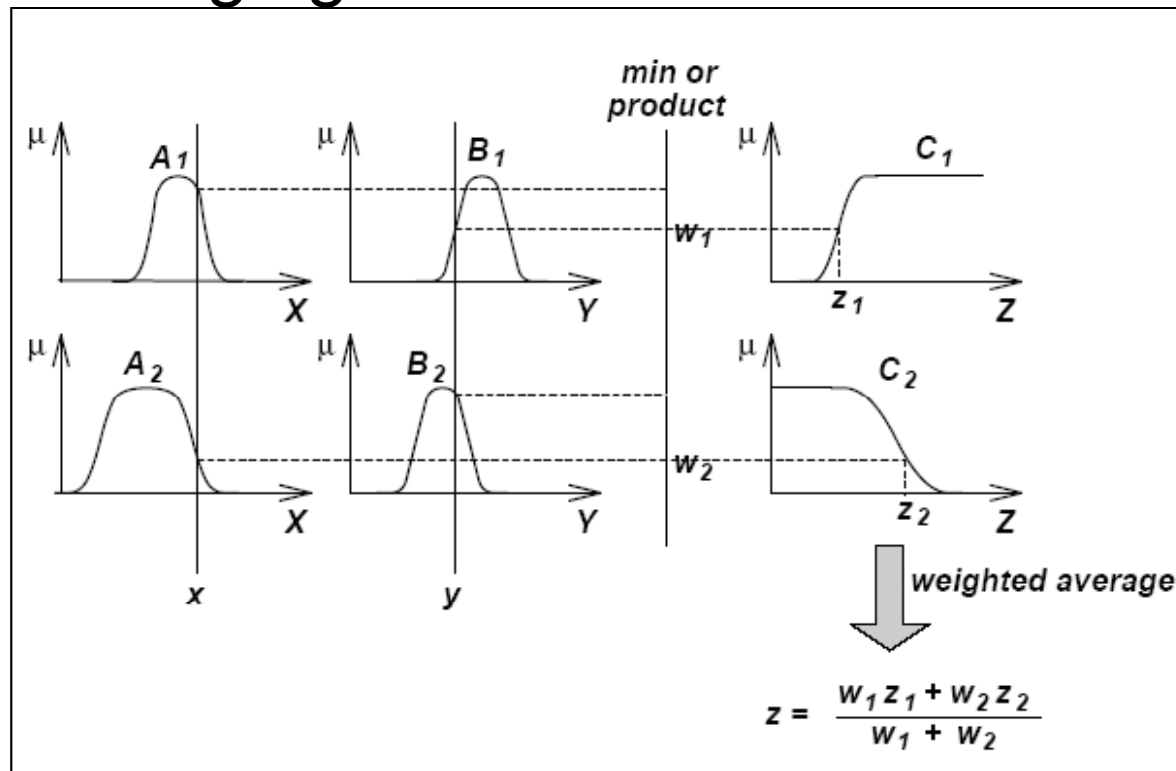
Tsukamoto Fuzzy Model

- The inferred output of each rule is defined as a crisp value induced by the rule's firing strength
- The overall output is taken as the weighted average of each rule's output
  - Thus avoids the time-consuming process of defuzzification
- This model is not used often since it is not as transparent as either the Mamdani or Sugeno fuzzy models
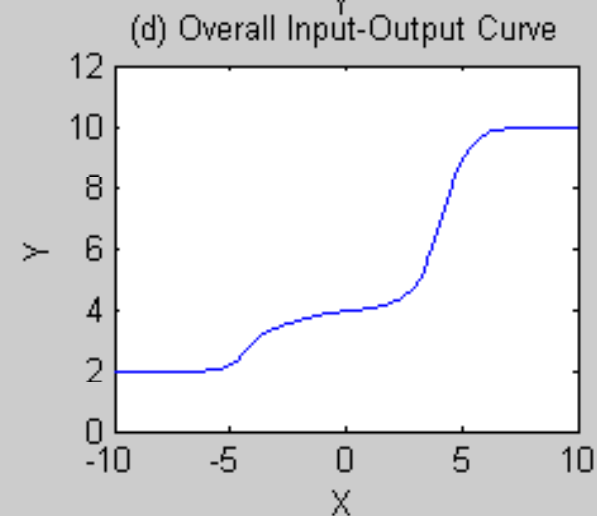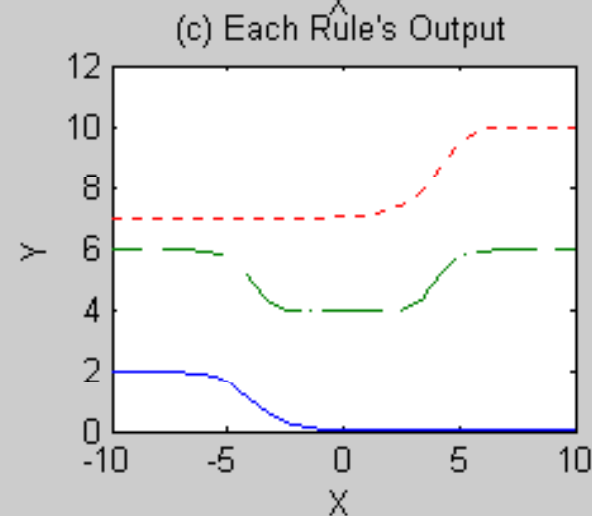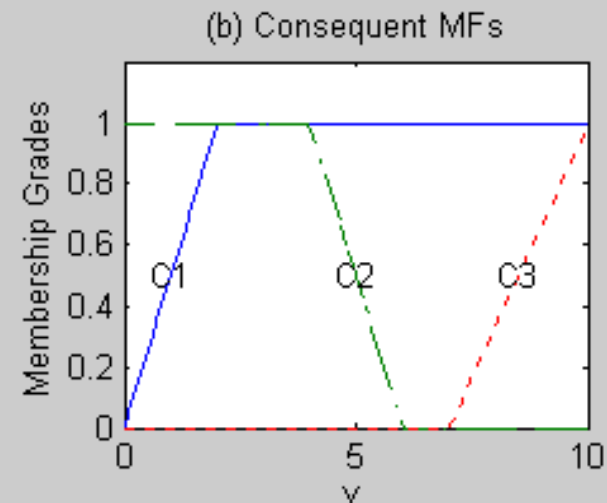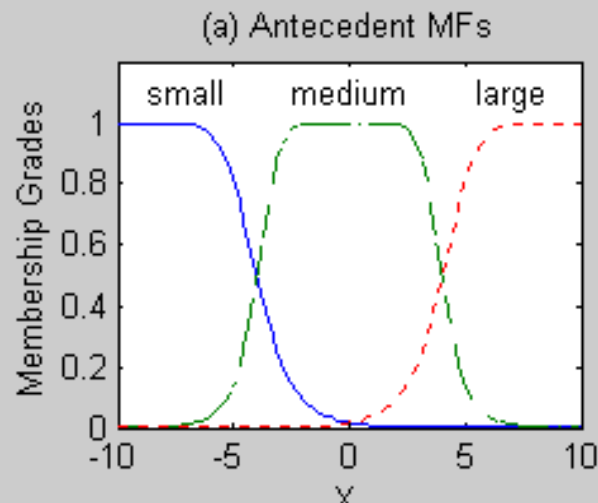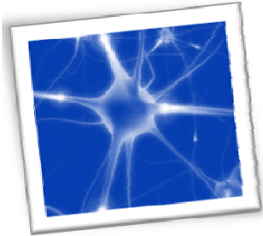
Example: Single-input Tsukamoto fuzzy model

- An example of a single-input Tsukamoto fuzzy model:
    - IT X is small then *Y is C$_1$*
    - IT X is medium then *Y is C$_2$*
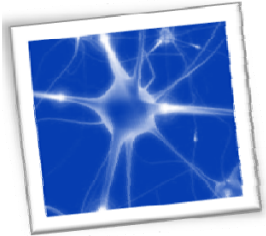   - IT X is large then *Y is C$_3$*,
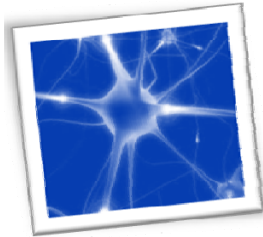
Example: Single-input Tsukamoto fuzzy model

Example: Single-input Tsukamoto fuzzy model

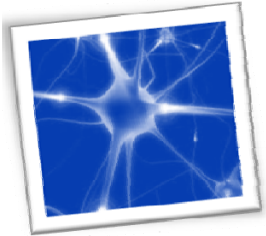- The overall input-output curve is equal to:

$$\left(\sum_{i=1}^{3} w_i f_i\right) / \left(\sum_{i=1}^{3} w_i\right),$$

  - Where f_i is the output of each rule induced by the firing strength w_i and MF for C_i

- If we plot each rule's output f_i as a function of x, we obtain figure (c), which is not quite obvious from the original rule-base and MF plots.

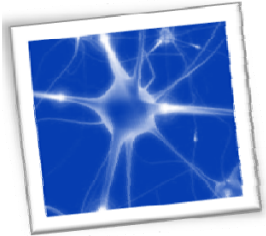- The output is always crisp even when the inputs are fuzzy.

FIS Optimisation

- Why to optimise?
    - trial and error tuning is laborious
    - it can be impossibly complicated if number of input parameters are large (100's or more)
    - far too many parameters to tune: number of rules, membership functions, rule consequents
    - arbitrariness of FIS is eliminated
   - if not optimised, the FIS performance is not optimum

FIS Optimisation

- Optimisation methods:
  - Adaptive Neuro-Fuzzy systems
    multilayer perceptron neural networks (ANFIS, FuNeI)
    RBFN → we will see these systems later on
  - evolutionary techniques (genetic algorithms)
  - clustering methods (cluster analysis, Hard C-means, Fuzzy C-means)
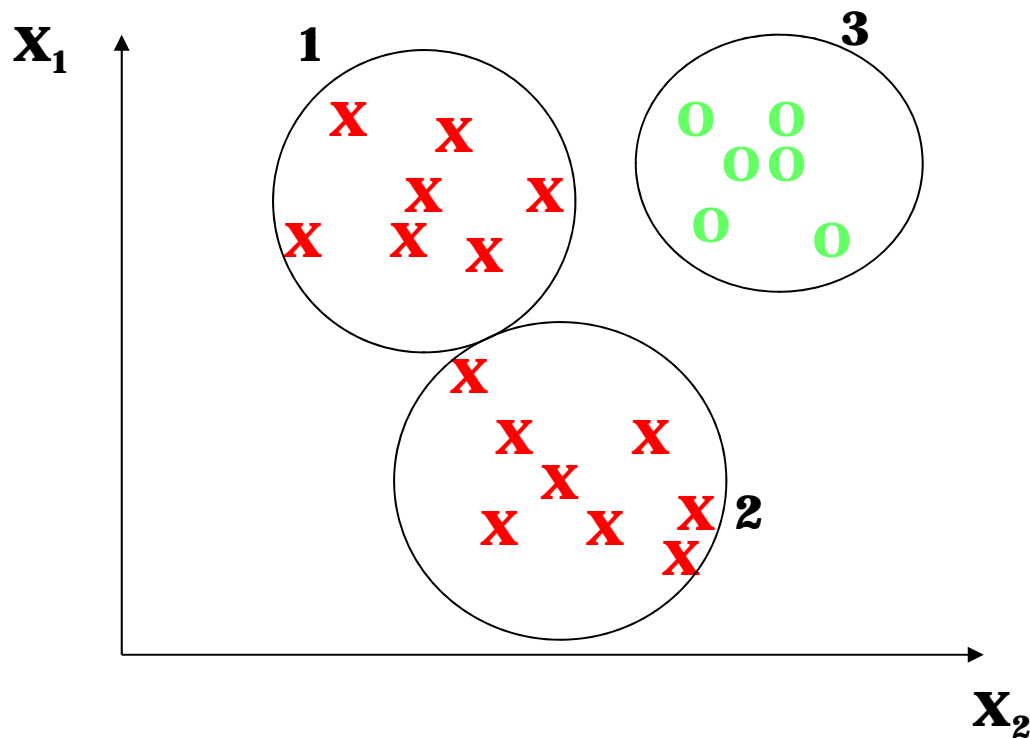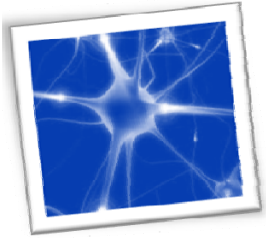 - ...

FIS Optimisation

- Optimisation can be viewed as knowledge discovery
- Optimisation = Learning
- Resulting model describes input-output relationships qualitatively and quantitatively.
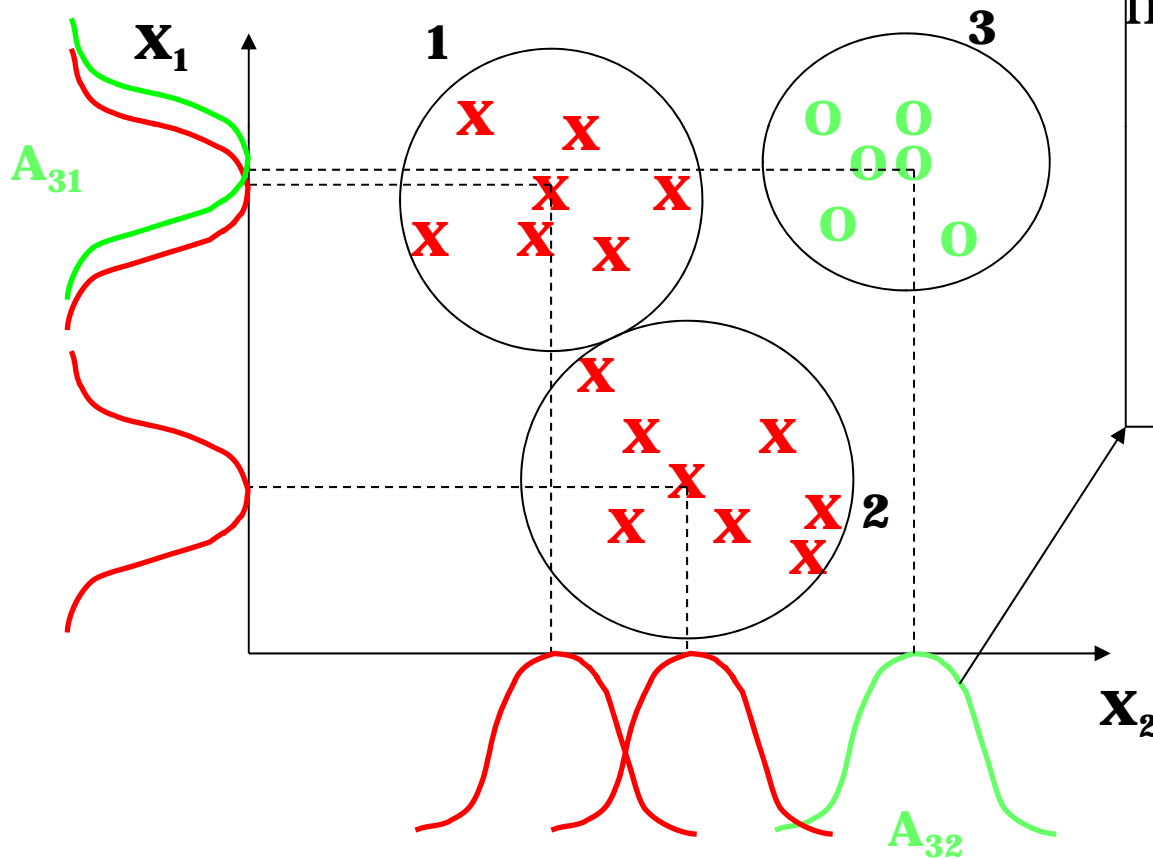- Easy validation and verification of FIS
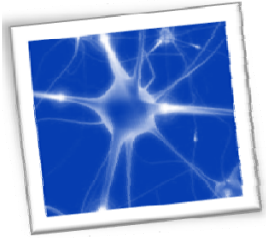
Example: fuzzy clustering

- Clustering the data of each class separately (*substractive clustering*)
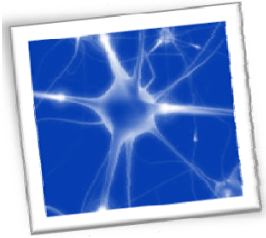
Example: fuzzy clustering

- fuzzy rules generation
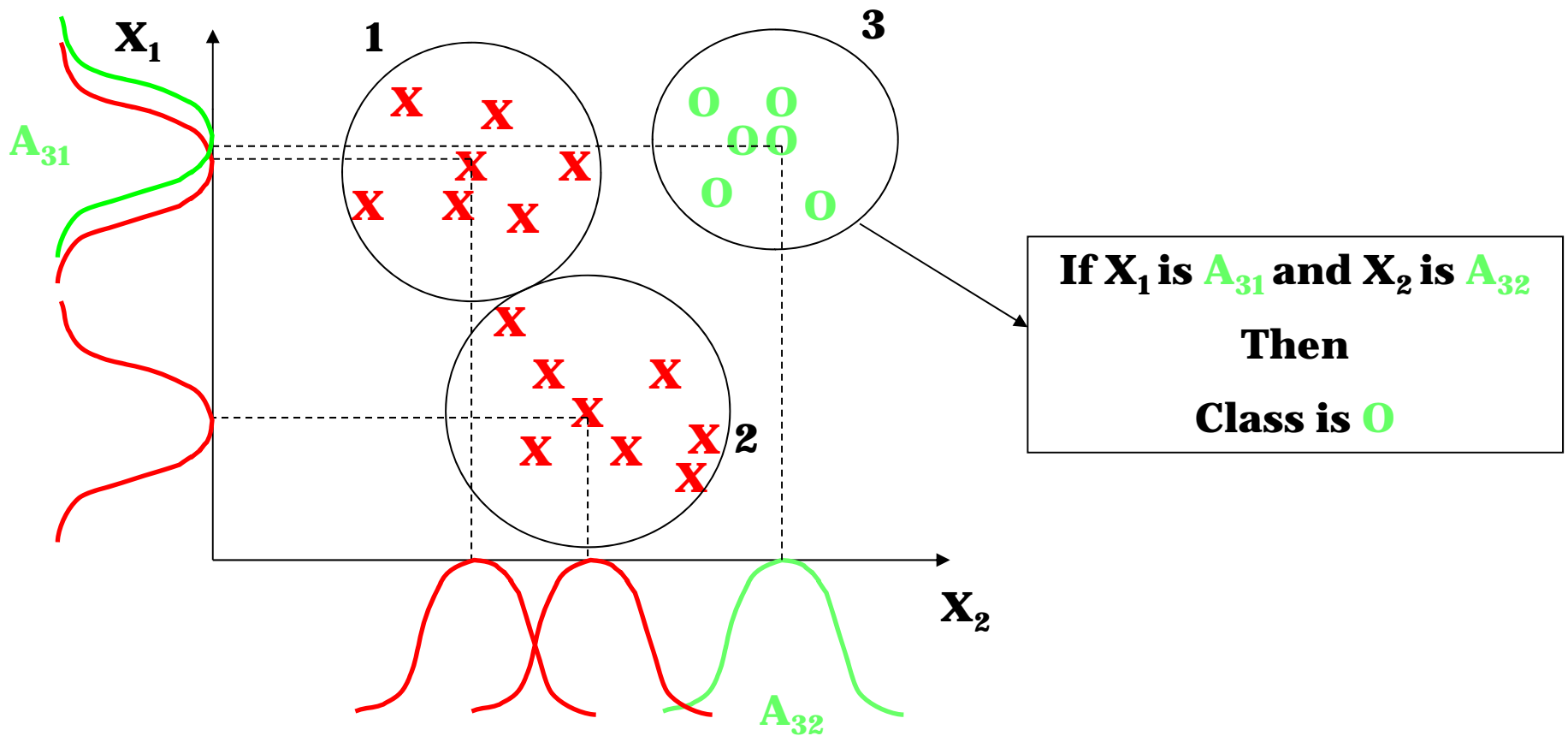
Gaussian fuzzy membership function
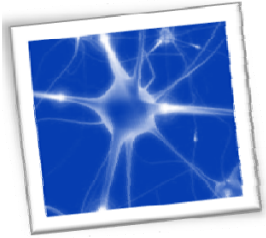
$$A_{32}(X_2) = e^{-\frac{1}{2}\left(\frac{X_2 - x_{32}}{\sigma_{32}}\right)^2}$$
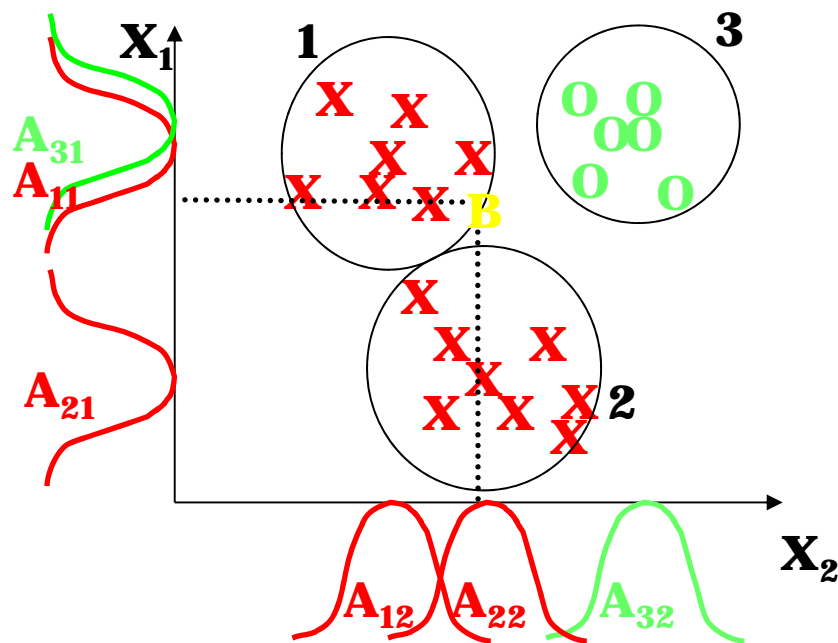
Example: fuzzy clustering

- fuzzy rules generation

If X_1 is A_{31} and X_2 is A_{32}

Then

Class is O

Example: fuzzy clustering

- Classification of an unseen vector B
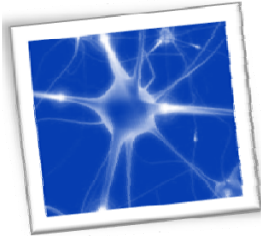

If X_1 is A_{11} and X_2 is A_{12} Then Class is X
$A_{11}(X_1) * A_{12}(X_2) = 0.8 * 0.2 = \boxed{0.16}$

If X_1 is A_{21} and X_2 is A_{22} Then Class is X
$A_{21}(X_1) * A_{22}(X_2) = 0.01 * 0.9 = 0.009$

If X_1 is A_{31} and X_2 is A_{32} Then Class is O
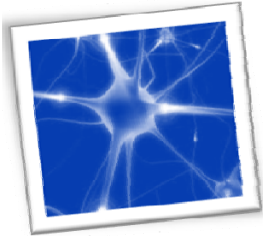$A_{31}(X_1) * A_{32}(X_2) = 0.1 * 0.01 = 0.001$

B belongs to X

Building a fuzzy expert system: case study

- A service centre keeps spare parts and repairs parts.

- A customer brings a failed item and receives a spare of the same type.

- Failed parts are repaired by **servers**, placed on the shelf, and thus become spares.

- The objective here is to advise a manager of the service centre on certain decision policies to keep the customers satisfied.

- Advise on the initial number of spares to keep **delay** reasonable
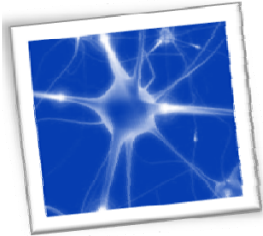
  From: http://www2.cs.siu.edu/~rahimi

Step 1: Specify the problem and define linguistic variables

There are four main linguistic variables: average waiting time (mean delay) *m*, repair utilisation factor of the service centre ρ, number of servers *s*, and initial number of spare parts *n*.
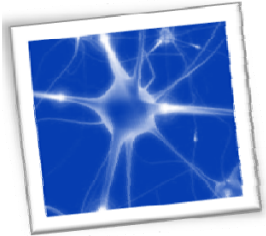
$$\rho = \frac{CustomerArrivalRate}{CustemerDepartureRate}$$

The system must advise management on the number of spares to keep as well as the number of servers. Increasing either will increase cost and decrease waiting time in some proportion.
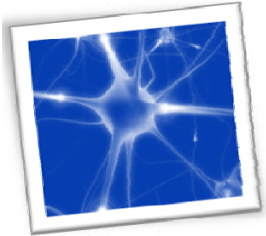
Linguistic variables and their ranges

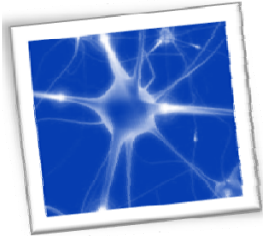| Linguistic Variable: *Mean Delay*, *m*		
Linguistic Value	Notation	Numerical Range (normalised)
Very Short	VS	[0, 0.3]
Short	S	[0.1, 0.5]
Medium	M	[0.4, 0.7]
Linguistic Variable: *Number of Servers*, *s*		
Linguistic Value	Notation	Numerical Range (normalised)
Small	S	[0, 0.35]
Medium	M	[0.30, 0.70]
Large	L	[0.60, 1]
Linguistic Variable: *Repair Utilisation Factor*, ρ		
Linguistic Value	Notation	Numerical Range
Low	L	[0, 0.6]
Medium	M	[0.4, 0.8]
High	H	[0.6, 1]
Linguistic Variable: *Number of Spares*, *n*		
Linguistic Value	Notation	Numerical Range (normalised)
Very Small	VS	[0, 0.30]
Small	S	[0, 0.40]
Rather Small	RS	[0.25, 0.45]
Medium	M	[0.30, 0.70]
Rather Large	RL	[0.55, 0.75]
Large	L	[0.60, 1]
Very Large	VL	[0.70, 1]

Step 2: Determine fuzzy sets

Fuzzy sets can have a variety of shapes. However, a triangle or a trapezoid can often provide an adequate representation of the expert knowledge, and at the same time, significantly simplifies the process of computation.
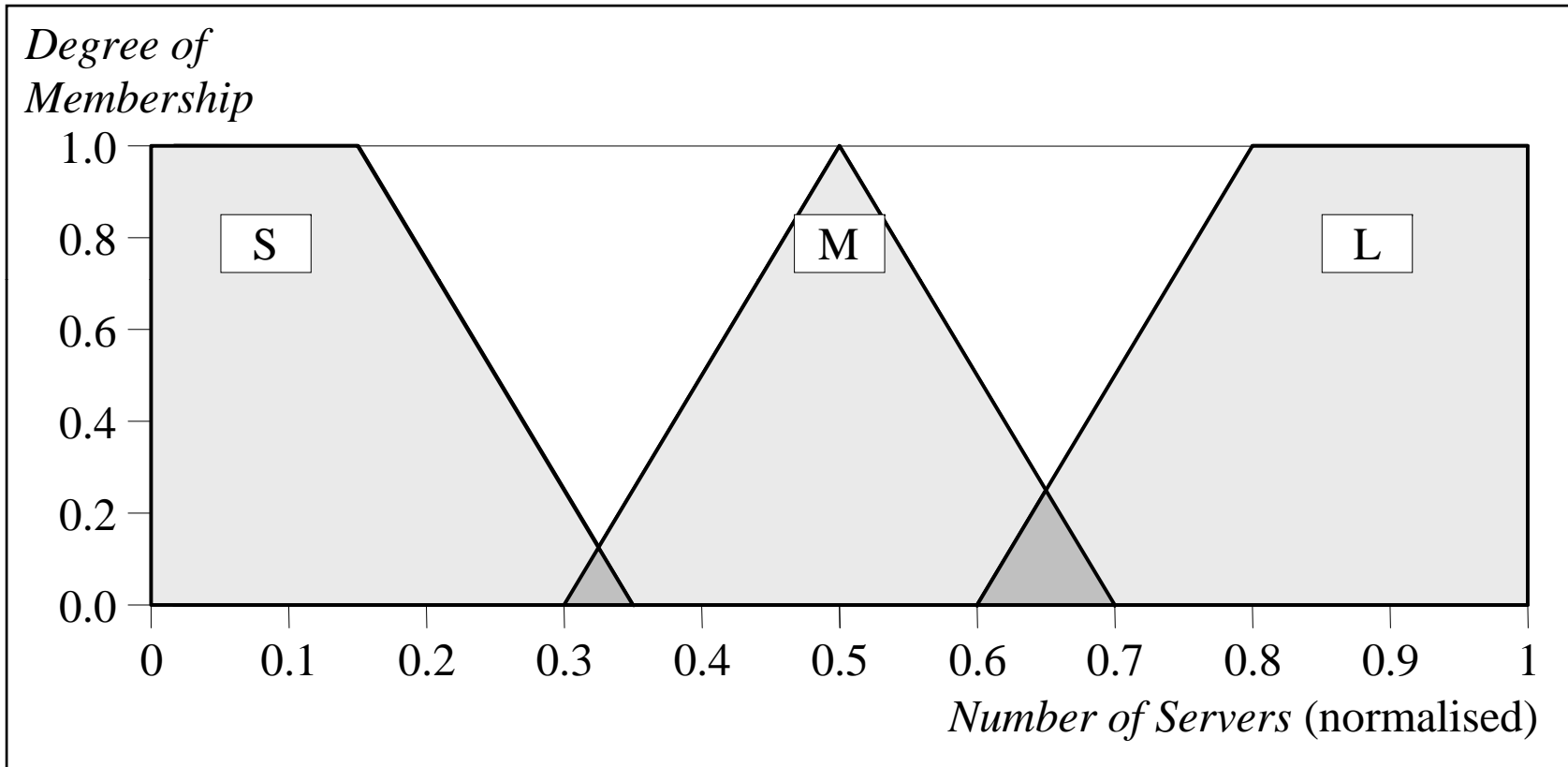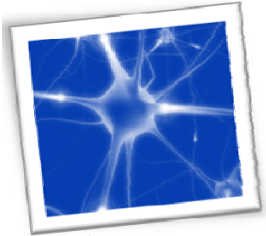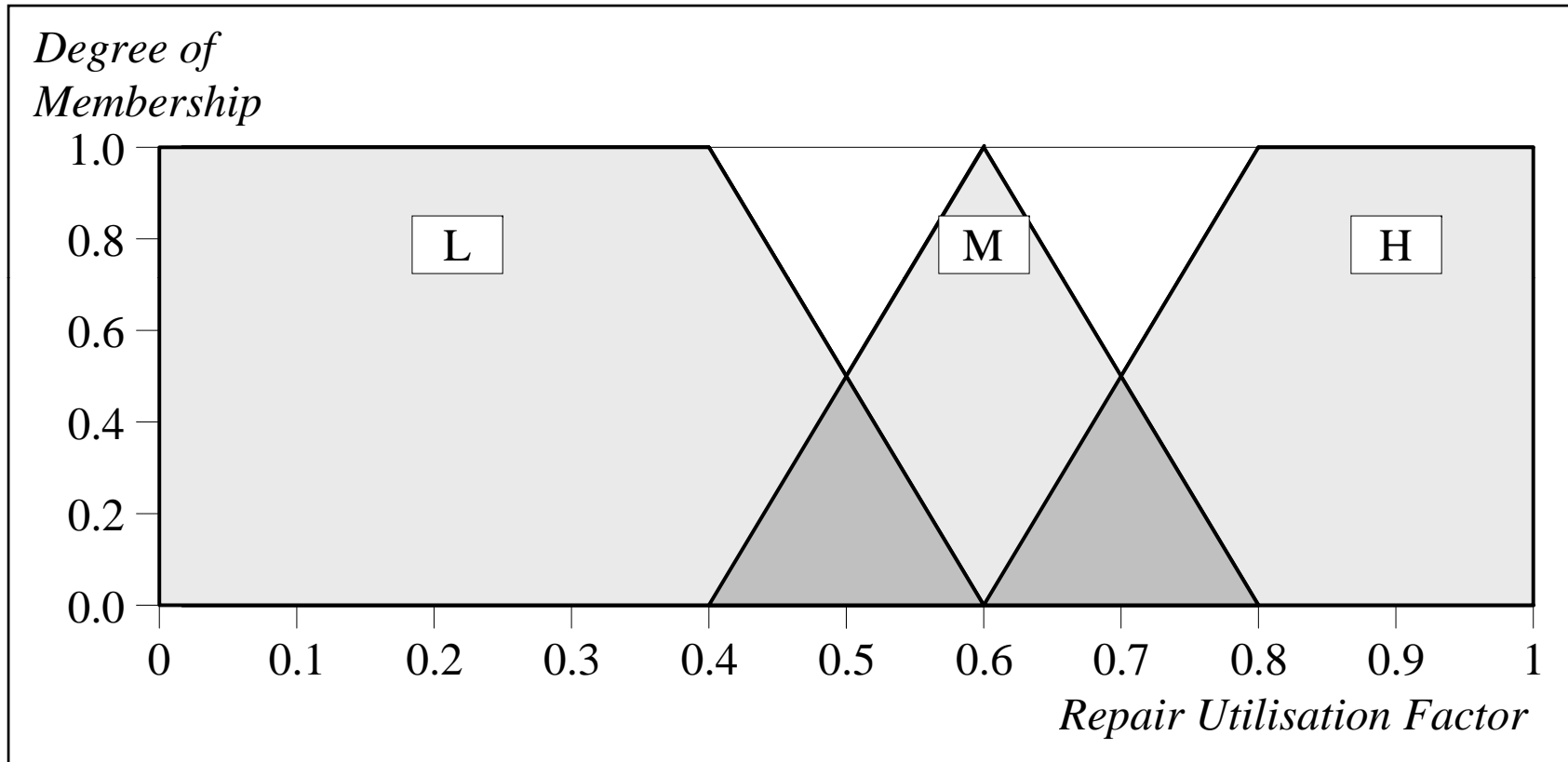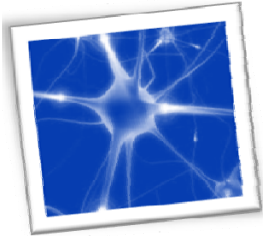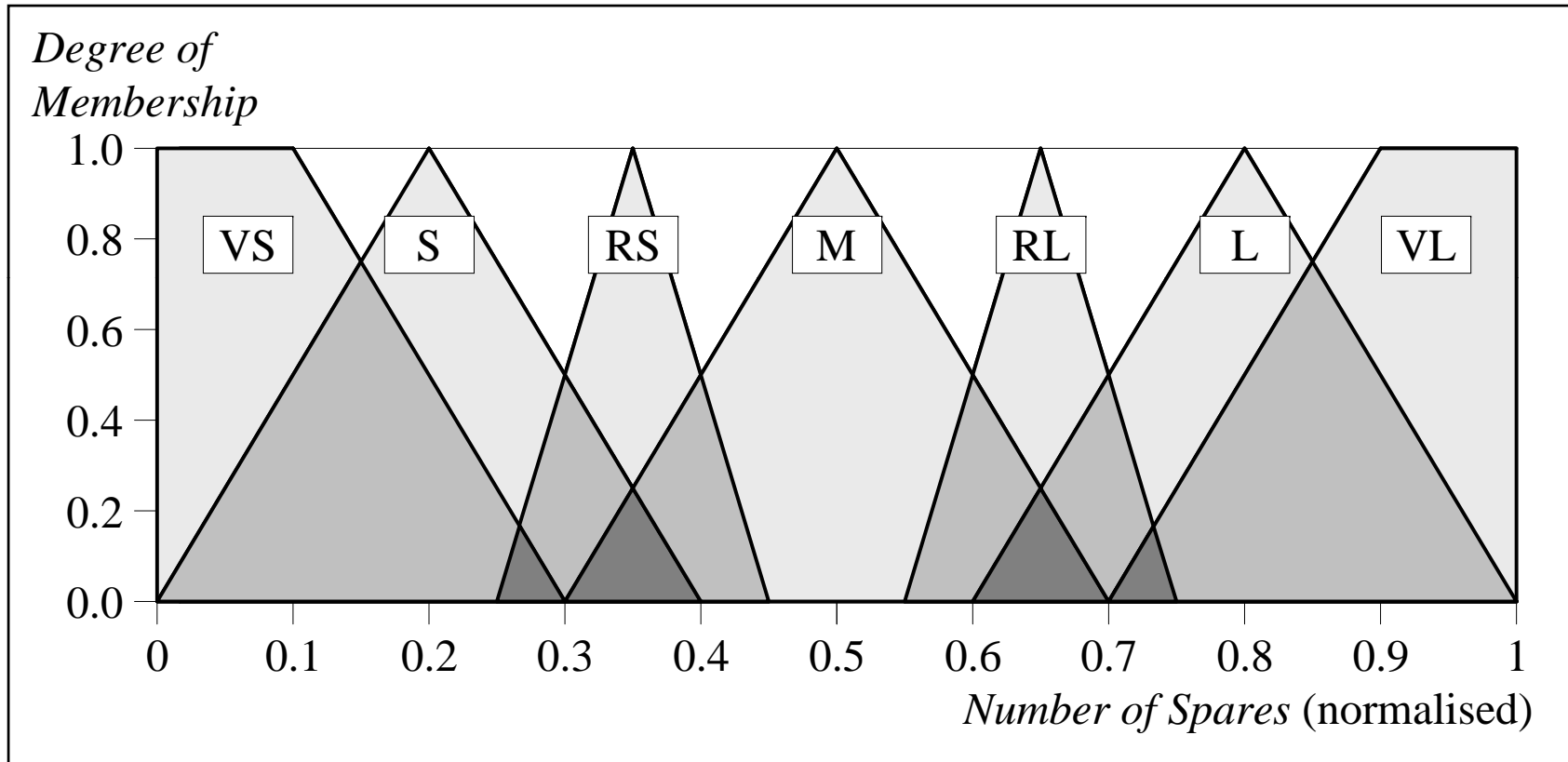
Fuzzy sets of Mean Delay m
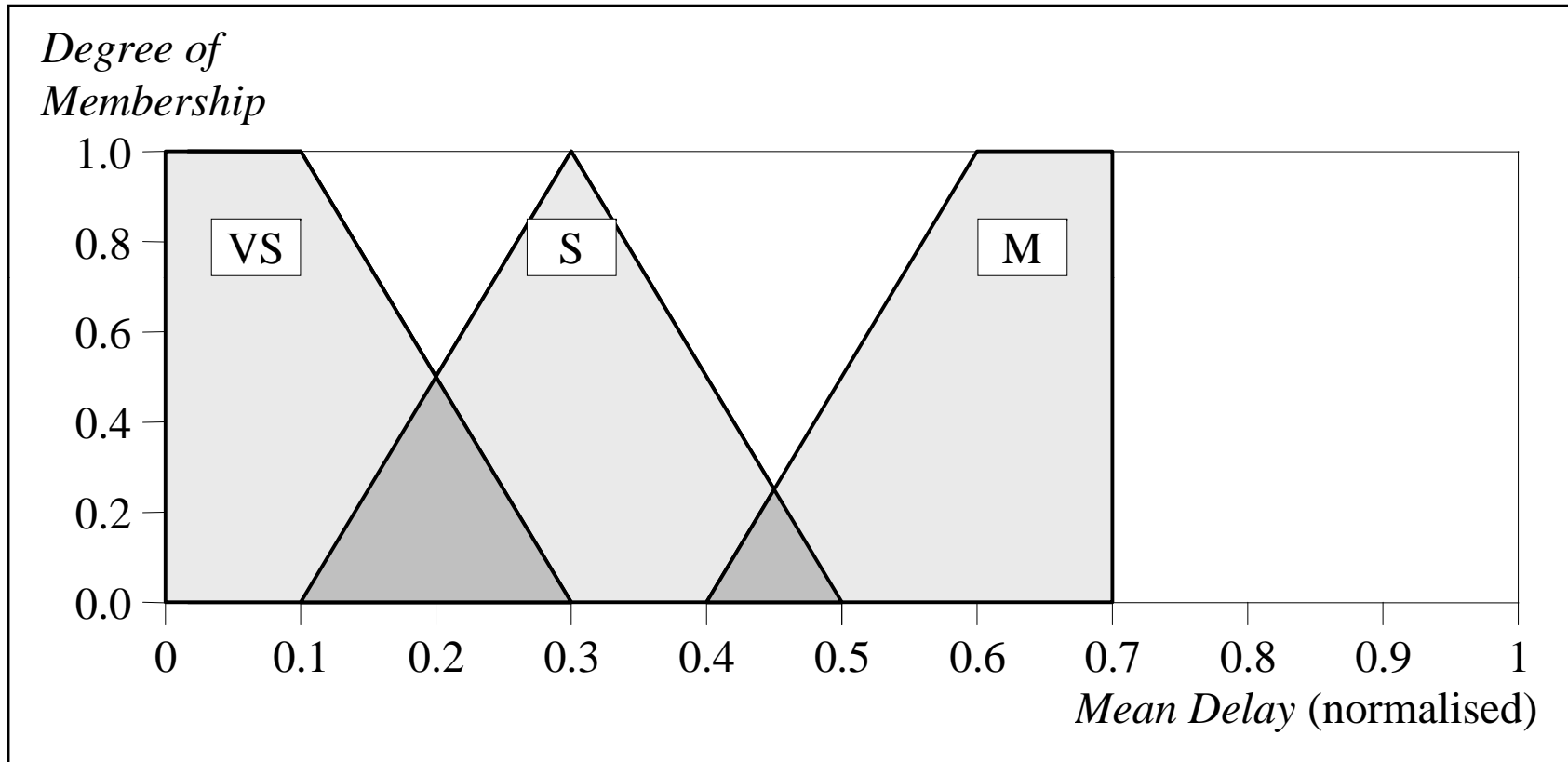
Fuzzy sets of Number of Servers s

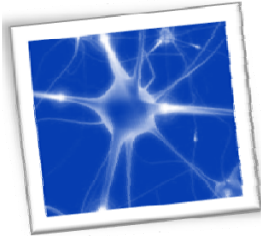# Fuzzy sets of Repair Utilisation Factor ρ

Fuzzy sets of Number of Spares n

Step 3: Elicit and construct fuzzy rules

To accomplish this task, we might ask the expert to describe how the problem can be solved using the fuzzy linguistic variables defined previously.

Required knowledge also can be collected from other sources such as books, computer databases, flow diagrams and observed human behaviour.

Rules about utilization and spares

1. If (utilisation_factor is L) then (number_of_spares is S)
2. If (utilisation_factor is M) then (number_of_spares is M)
3. If (utilisation_factor is H) then (number_of_spares is L)

Rules about delay, servers and spares

4. If (mean_delay is VS) and (number_of_servers is S) then (number_of_spares is VL)
5. If (mean_delay is S) and (number_of_servers is S) then (number_of_spares is L)
6. If (mean_delay is M) and (number_of_servers is S) then (number_of_spares is M)
7. If (mean_delay is VS) and (number_of_servers is M) then (number_of_spares is RL)
8. If (mean_delay is S) and (number_of_servers is M) then (number_of_spares is RS)
9. If (mean_delay is M) and (number_of_servers is M) then (number_of_spares is S)
10. If (mean_delay is VS) and (number_of_servers is L) then (number_of_spares is M)
11. If (mean_delay is S) and (number_of_servers is L) then (number_of_spares is S)
12. If (mean_delay is M) and (number_of_servers is L) then (number_of_spares is VS)
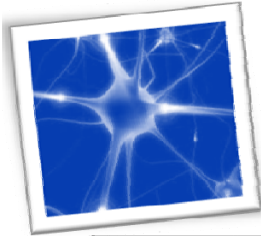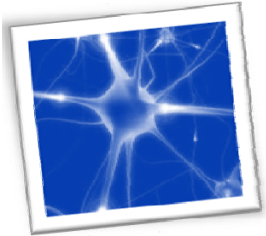
The larger rule base for three combinations

Rule	m	s	ρ	n	Rule	m	s	ρ	n	Rule	m	s	ρ	n
1	VS	S	L	VS	10	VS	S	M	S	19	VS	S	H	VL
2	S	S	L	VS	11	S	S	M	VS	20	S	S	H	L
3	M	S	L	VS	12	M	S	M	VS	21	M	S	H	M
4	VS	M	L	VS	13	VS	M	M	RS	22	VS	M	H	M
5	S	M	L	VS	14	S	M	M	S	23	S	M	H	M
6	M	M	L	VS	15	M	M	M	VS	24	M	M	H	S
7	VS	L	L	S	16	VS	L	M	M	25	VS	L	H	RL
8	S	L	L	S	17	S	L	M	RS	26	S	L	H	M
9	M	L	L	VS	18	M	L	M	S	27	M	L	H	RS

if mean_delay is VS
 and number_servers is S
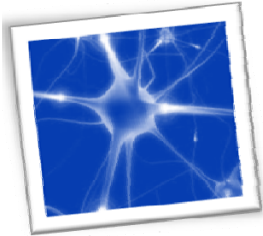 and utilization is Low
then spares is VS
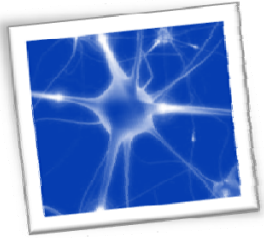
Step 4: Encode the fuzzy sets, fuzzy rules and procedures to perform fuzzy inference into the expert system
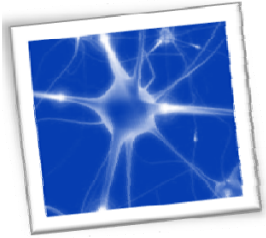
Step 5: Evaluate and tune the system

- The last and the most laborious task is to evaluate and tune the system. We want to see whether our fuzzy system meets the requirements specified at the beginning.
- Several test situations depend on the mean delay, number of servers and repair utilisation factor.
- The MatLab's Fuzzy Logic Toolbox can generate surface to help us analyse the system's performance.
- However, the expert might not be satisfied with the system performance.
- To improve the system performance, we may use additional sets – *Rather Small* and *Rather Large* – on the universe of discourse *Number of Servers*, and then extend the rule base.
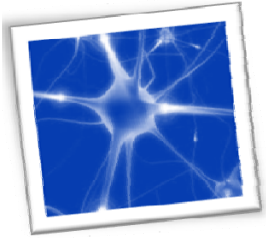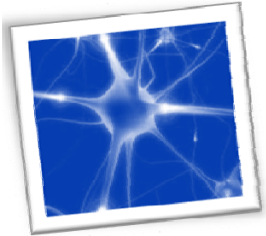
Modified fuzzy sets of Number of Servers
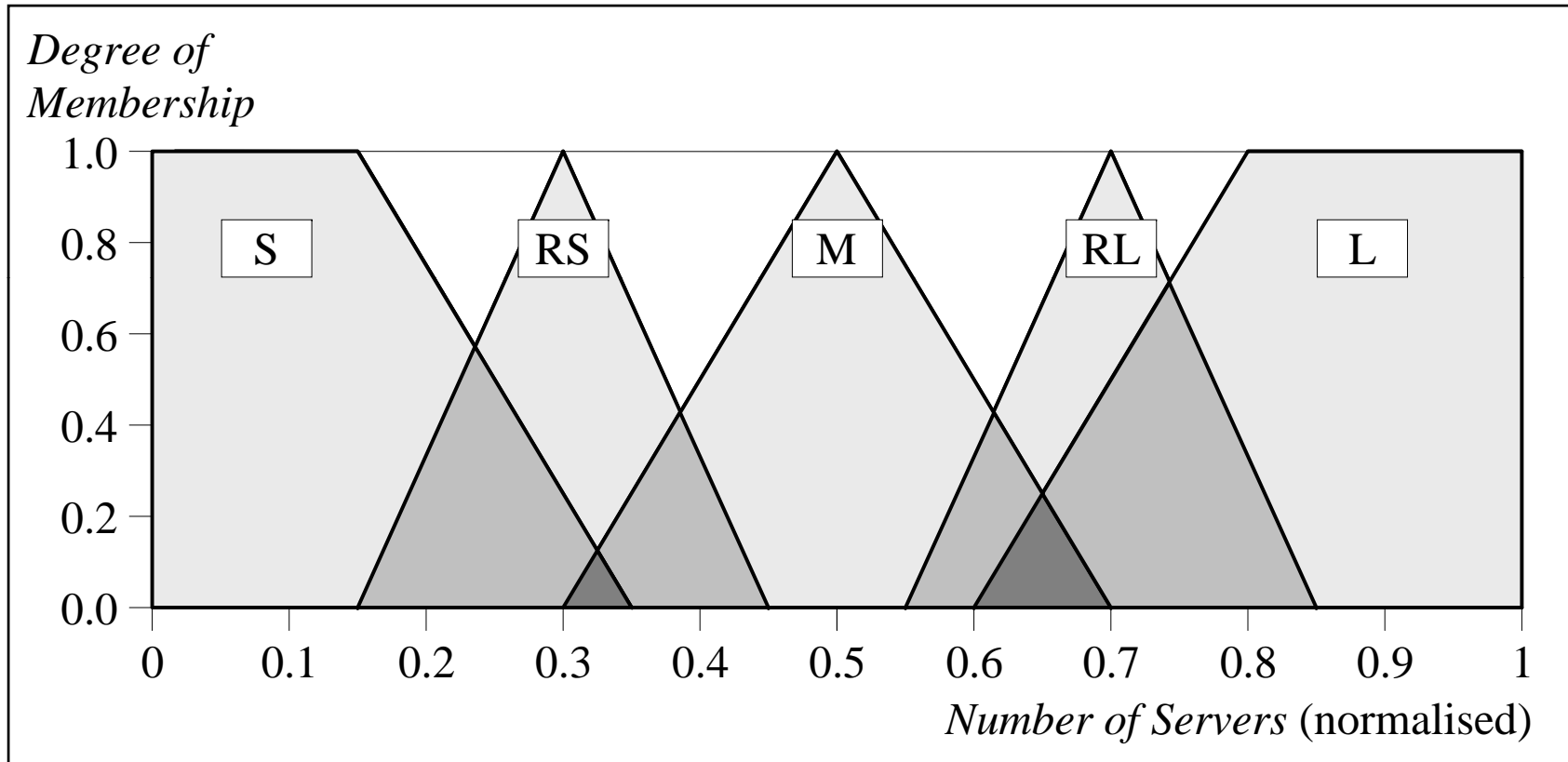
Tuning fuzzy systems

1. Review model input and output variables, and if required redefine their ranges.
2. Review the fuzzy sets, and if required define additional sets on the universe of discourse.
3. Provide sufficient overlap between neighbouring sets. It is suggested that triangle-to-triangle and trapezoid-to-triangle fuzzy sets should overlap between 25% to 50% of their bases.
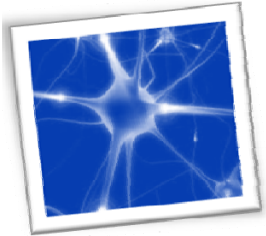
Tuning fuzzy systems

4. Review the existing rules, and if required add new rules to the rule base.
5. Examine the rule-base for opportunities to write hedge rules to capture the pathological behaviour of the system.
6. Adjust the rule execution weights. Most fuzzy logic tools allow control of the importance of rules by changing a weight multiplier **(we did not see this one in this course !)**.
7. Revise shapes of the fuzzy sets. In most cases, fuzzy systems are highly tolerant of a shape approximation.
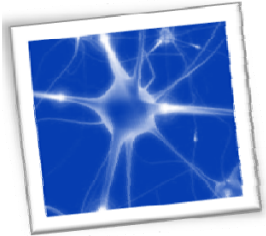
Other Considerations

- certain common issues concerning all these three fuzzy inference systems
    - how to partition an input space
   - how to construct a fuzzy inference system for a particular application

Input Space Partitioning

- The antecedent of a fuzzy rule defines a local fuzzy region,
- The consequent describes the behavior within the region via various constituents
  - constituent can be a consequent MF (Mamdani and Tsukamoto fuzzy models),
  - a constant value (zero-order Sugeno model),
  - or a linear equation (first-order Sugeno model).
- Different consequent constituents result in different fuzzy inference systems,
- but their antecedents are always the same. Therefore, methods of partitioning input spaces to form the antecedents of fuzzy rules is applicable to all three types of fuzzy inference systems
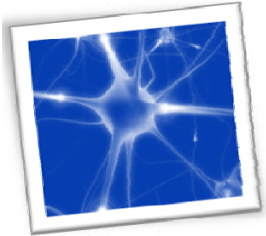
Grid partition
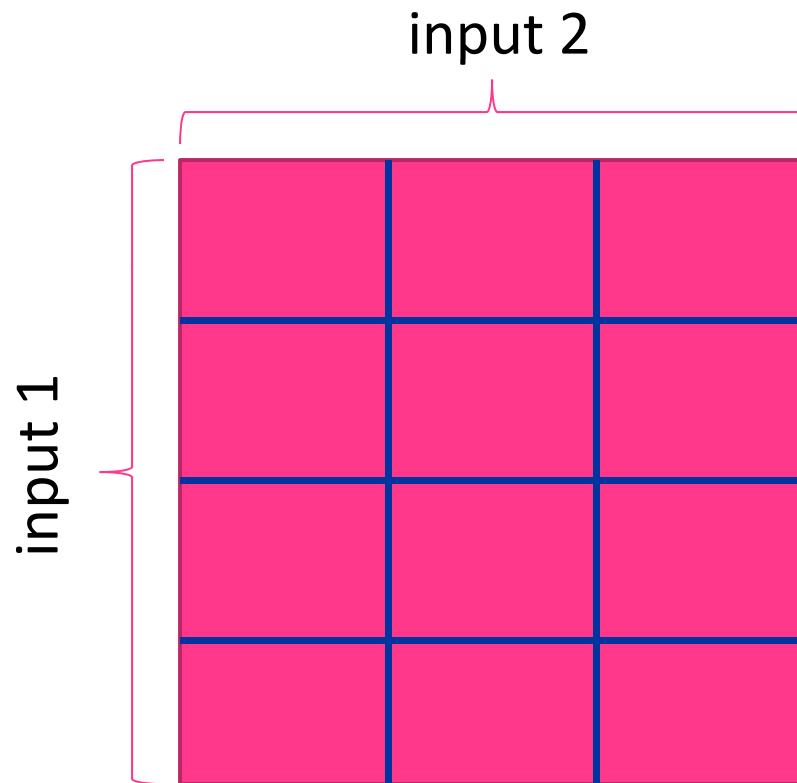
input 2
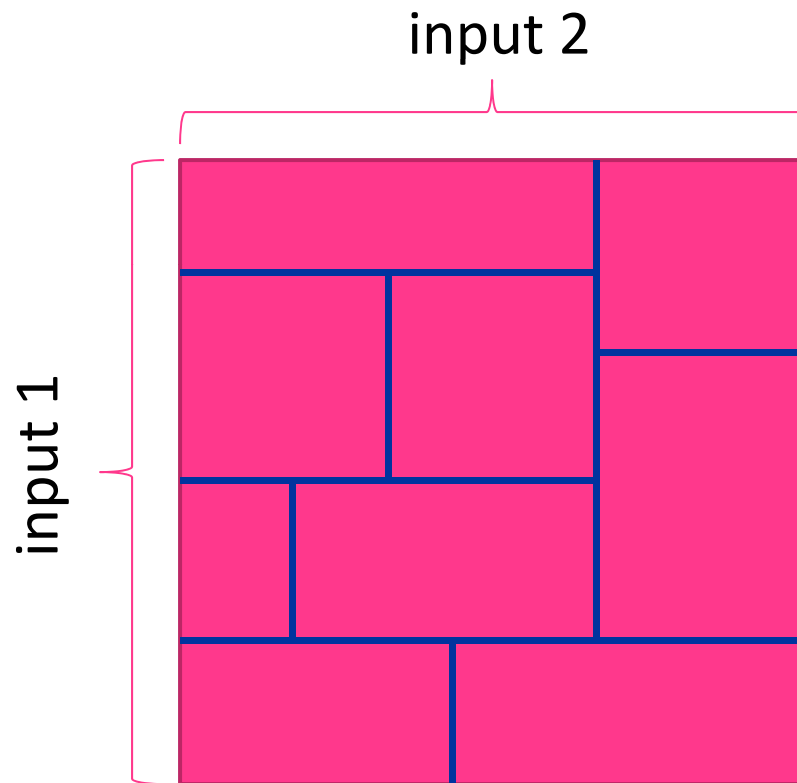
input 1

Grid Partition

<u>The curse of dimensionality:</u>

If we have n inputs and m FMs per input, then we have m^n if-then rules.

For example, 6 inputs and 5 memberships per input $\rightarrow 5^6 = 15,625$ rules!
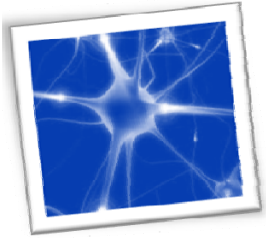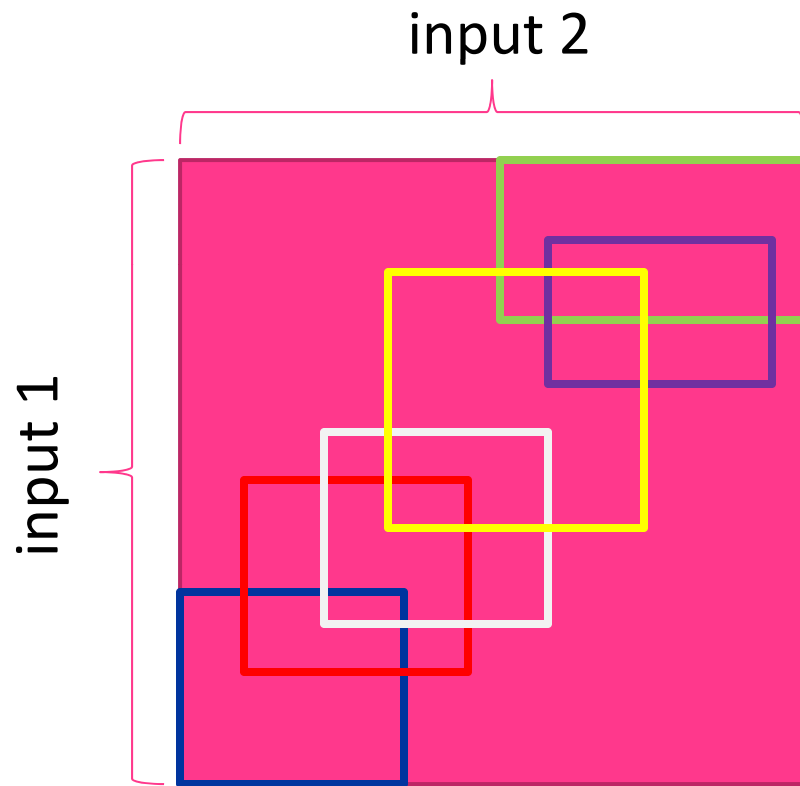
Tree partition

input 2

input 1

Tree Partition

- Reduces the number of rules
- Requires more MFs per input
- MFs do not have clear linguistic meanings

Scatter partition

input 2

input 1

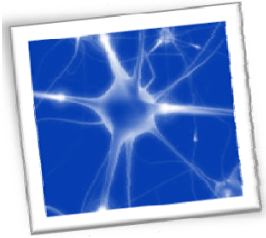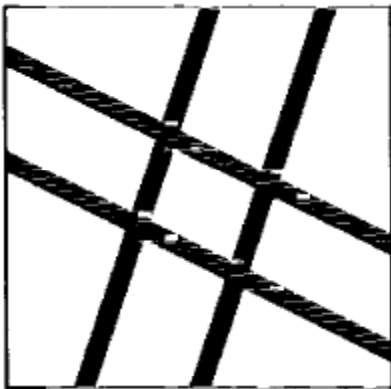## Scatter Partition

- In many systems, extremes occur rarely
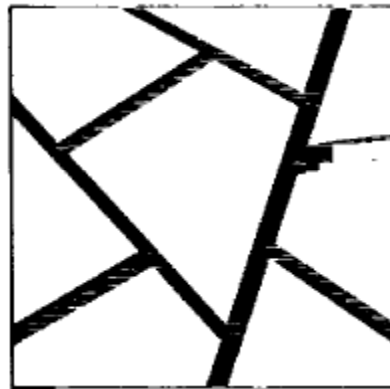- Number of active rules depends on input values

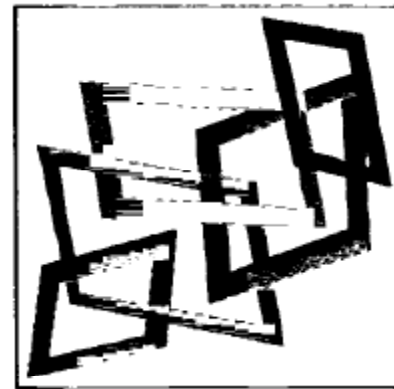# Input Space Partitioning

- A more flexible partition style, when MFs are defined on certain transformations of the input Variables:
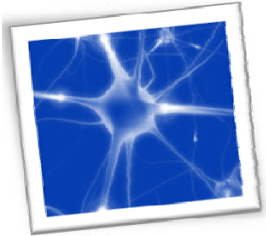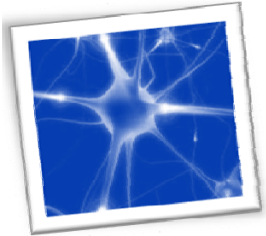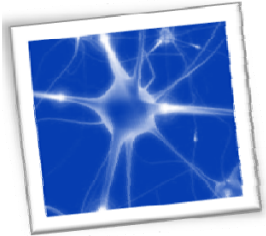

(a)   (b)   (c)

Fuzzy Modeling

- In general, we design a fuzzy inference system based on the past known behavior of a target system.

- now consider how we might construct a fuzzy inference system for a specific application

- **Fuzzy Modeling:** The standard method for constructing a fuzzy inference system
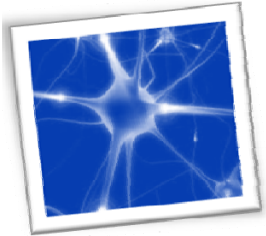
Fuzzy Modeling

- **fuzzy modeling** has the following features:
  - Takes advantage of **domain knowledge**
    - Making it easy to incorporate human expertise about the target system directly into the modeling process
  - The use of **numerical data** also plays an important role in fuzzy modeling
    - When the input-output data of a target system is available, conventional system identification techniques can be used for fuzzy modeling
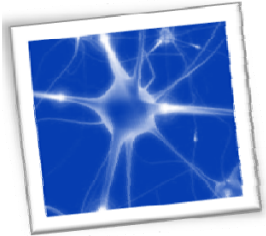
Fuzzy Modeling

- fuzzy modeling can be pursued in two stages, which are not totally disjoint:
    1. the identification of the **surface structure**
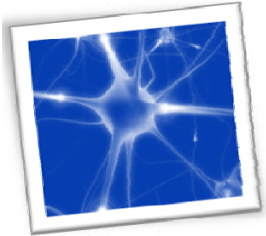   2. the identification of **deep structure**

Fuzzy Modeling

- The identification of the surface structure includes the following tasks:
  1. Select relevant input and output variables
  2. Choose a specific type of fuzzy inference system.
  3. Determine the number of linguistic terms associated with each input and output variables. (For a Sugeno model, determine the order of consequent equations.)
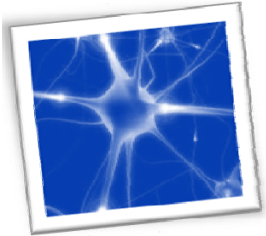 4. Design a collection of fuzzy if-then rules.

Fuzzy Modeling

- After the first stage of fuzzy modeling, we obtain a rule-base that can describe the behavior of the target system by means of linguistic terms.
- The meaning of these linguistic terms is determined in the second stage, the identification of **deep structure**, which determines the MFs of each linguistic term (and the coefficients of each rule's output polynomial if a Sugeno fuzzy model is used).

Fuzzy Modeling

- The identification of deep structure includes the following tasks:
    1. Choose an appropriate family of parameterized MFs
    2. Interview human experts familiar with the target systems to determine the parameters of the MFs used in the rule-base.
    3. Refine the parameters of the MFs using regression and optimization techniques.

Reading

- J-S R Jang and C-T Sun, Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997 (Chapter 4).