The extension principle, membership functions and fuzzy control systems

Shabana K M
PhD Research Scholar
Department of Computer Science and Engineering

IIT Palakkad

17 June 2021

INDIAN INSTITUTE OF TECHNOLOGY PALAKKAD

The Extension Principle

- one of the most basic ideas in fuzzy set theory

The Extension Principle

- one of the most basic ideas in fuzzy set theory
- used to perform algebraic operations on fuzzy numbers

The Extension Principle

- one of the most basic ideas in fuzzy set theory
- used to perform algebraic operations on fuzzy numbers
- extends the standard arithmetic and algebraic operations based on crisp sets to fuzzy arithmetic and fuzzy algebraic operations

The Extension Principle

- one of the most basic ideas in fuzzy set theory
- used to perform algebraic operations on fuzzy numbers
- extends the standard arithmetic and algebraic operations based on crisp sets to fuzzy arithmetic and fuzzy algebraic operations
- Consider a function/operation defined on the universe of discourse $\mathcal{X}$ given by $f : y = f(x)$

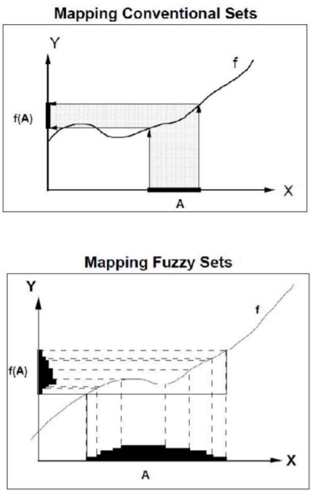
The Extension Principle

- one of the most basic ideas in fuzzy set theory
- used to perform algebraic operations on fuzzy numbers
- extends the standard arithmetic and algebraic operations based on crisp sets to fuzzy arithmetic and fuzzy algebraic operations
- Consider a function/operation defined on the universe of discourse $\mathcal{X}$ given by $f : y = f(x)$
  - when f acts on elements of a crisp subset of $\mathcal{X}$, say x', the mapped elements y' also form a crisp set

The Extension Principle

- one of the most basic ideas in fuzzy set theory
- used to perform algebraic operations on fuzzy numbers
- extends the standard arithmetic and algebraic operations based on crisp sets to fuzzy arithmetic and fuzzy algebraic operations
- Consider a function/operation defined on the universe of discourse $\mathcal{X}$ given by $f : y = f(x)$
  - when f acts on elements of a crisp subset of $\mathcal{X}$, say x', the mapped elements y' also form a crisp set
  - what happens when f acts on fuzzy subsets of $\mathcal{X}$?

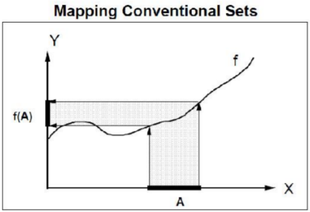
Extension Principle - One to one function $f : \mathcal{X} \to \mathcal{Y}$

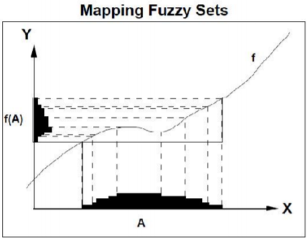


Case 1: One to one function $f : \mathcal{X} \to \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X}) \to \mathbb{F}(\mathcal{Y})$ such that for $A \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with $B = \hat{f}(A)$ is given by:

Extension Principle - One to one function $f : \mathcal{X} \to \mathcal{Y}$



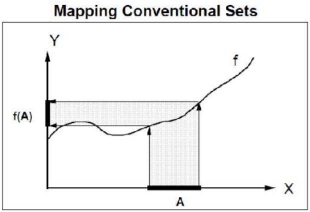


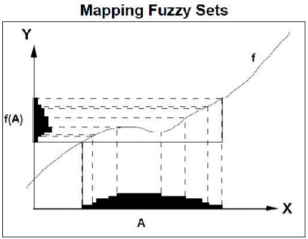
Case 1: One to one function $f : \mathcal{X} \to \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X}) \to \mathbb{F}(\mathcal{Y})$ such that for $A \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with $B = \hat{f}(A)$ is given by:

$\mu_B(y) = \{\mu_A(x) \mid f(x) = y, x \in \mathcal{X}\}$ for all $y \in \mathcal{Y}$

Extension Principle - One to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$



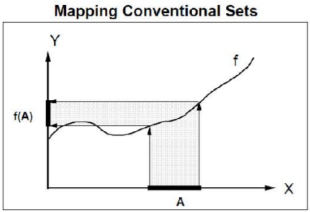


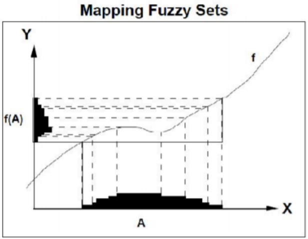
Case 1: One to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$

Example: Let $A = (\mathcal{X}, \mu_A) = \mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \ldots + \mu_A(x_n)/x_n$

$f : \mathcal{X} \rightarrow \mathcal{Y}$ is one to one

Extension Principle - One to one function $f : \mathcal{X} \to \mathcal{Y}$





Case 1: One to one function $f : \mathcal{X} \to \mathcal{Y}$

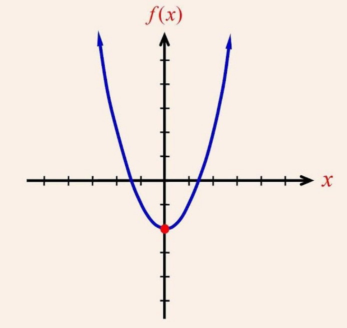
Example: Let $A = (\mathcal{X}, \mu_A) = \mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \ldots + \mu_A(x_n)/x_n$

$f : \mathcal{X} \to \mathcal{Y}$ is one to one

$$\begin{aligned} B &= \hat{f}(A) \\ &= \mu_A(x_1)/y_1 + \ldots + \mu_A(x_n)/y_n \end{aligned}$$

where $y_i = f(x_i)$

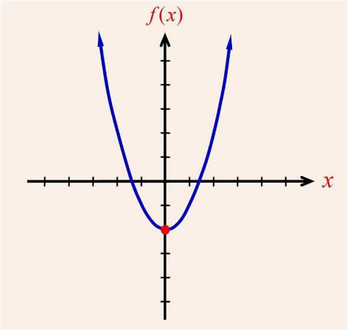
Extension Principle - Many to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$



Case 2: Many to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X}) \rightarrow \mathbb{F}(\mathcal{Y})$ such that for $A \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with $B = \hat{f}(A)$ is given by:

Extension Principle - Many to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$



Case 2: Many to one function $f : \mathcal{X} \rightarrow \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X}) \rightarrow \mathbb{F}(\mathcal{Y})$ such that for $A \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with $B = \hat{f}(A)$ is given by:

$$\mu_B(y) = \max_{\substack{x \in \mathcal{X} \\ f(x) = y}} \mu_A(x) \text{ for all } y \in \mathcal{Y}$$

Extension Principle - One to one function $f : \mathcal{X}^n \rightarrow \mathcal{Y}$

Case 3: One to one function $f : \mathcal{X}^n \rightarrow \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X})^n \rightarrow \mathbb{F}(\mathcal{Y})$ such that for $A_1, A_2, ..., A_n \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with

$$B = \hat{f}(A_1, A_2, ..., A_n) \text{ is given by:}$$

Extension Principle - One to one function $f : \mathcal{X}^n \to \mathcal{Y}$

Case 3: One to one function $f : \mathcal{X}^n \to \mathcal{Y}$

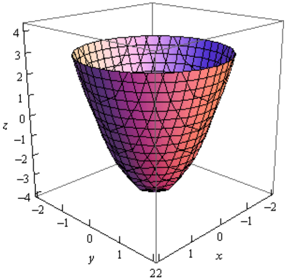
The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X})^n \to \mathbb{F}(\mathcal{Y})$ such that for $A_1, A_2, ..., A_n \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with

$$B = \hat{f}(A_1, A_2, ..., A_n) \text{ is given by:}$$

$$\mu_B(y) = \min_{\substack{x_1, x_2, \ldots, x_n \in \mathcal{X} \\ f(x_1, x_2, \ldots, x_n) = y}} \{\mu_{A_1}(x_1), \mu_{A_2}(x_2), ..., \mu_{A_n}(x_n)\}$$

for all $y \in \mathcal{Y}$

Extension Principle - Many to one function $f : \mathcal{X}^n \to \mathcal{Y}$

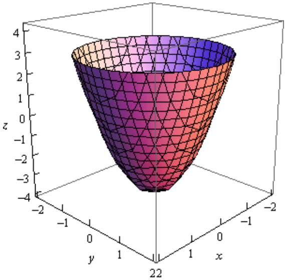


Case 4: Many to one function $f : \mathcal{X}^n \to \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X})^n \to \mathbb{F}(\mathcal{Y})$ such that for $A_1, A_2, ..., A_n \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with

$$B = \hat{f}(A_1, A_2, ..., A_n) \text{ is given by:}$$

Extension Principle - Many to one function $f : \mathcal{X}^n \to \mathcal{Y}$



Case 4: Many to one function $f : \mathcal{X}^n \to \mathcal{Y}$

The function f is extended to a function $\hat{f} : \mathbb{F}(\mathcal{X})^n \to \mathbb{F}(\mathcal{Y})$ such that for $A_1, A_2, ..., A_n \in \mathbb{F}(\mathcal{X})$, the set $B \in \mathbb{F}(\mathcal{Y})$ with

$B = \hat{f}(A_1, A_2, ..., A_n)$ is given by:

$$\mu_B(y) = \sup_{\substack{x_1, x_2, \ldots, x_n \in \mathcal{X} \\ f(x_1, x_2, \ldots, x_n) = y}} \min\{\mu_{A_1}(x_1), ..., \mu_{A_n}(x_n)\}$$

for all $y \in \mathcal{Y}$

Extension Principle - Example

Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \to \mathcal{X}$

Extension Principle - Example

Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \to \mathcal{X}$

$A = 0.3/1 + 0.6/2 + 1/3 + 0.6/4 + 0.3/5$
$B = 0.5/10 + 1/11 + 0.5/12$
$C = \hat{f}(A, B)$

Extension Principle - Example

Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \rightarrow \mathcal{X}$

$A = 0.3/1 + 0.6/2 + 1/3 + 0.6/4 + 0.3/5$
$B = 0.5/10 + 1/11 + 0.5/12$
$C = \hat{f}(A, B)$

$\mu_C(11) = \max\{\min\{0.3, 0.5\}\} = 0.3$

Extension Principle - Example

Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \rightarrow \mathcal{X}$

$A = 0.3/1 + 0.6/2 + 1/3 + 0.6/4 + 0.3/5$
$B = 0.5/10 + 1/11 + 0.5/12$
$C = \hat{f}(A, B)$

$\mu_C(11) = \max\{\min\{0.3, 0.5\}\} = 0.3$

$\mu_C(12) = \max\{\min\{0.5, 0.6\}, \min\{1, 0.3\}\} = \max\{0.5, 0.3\} = 0.5$

Extension Principle - Example

Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \to \mathcal{X}$

$$A = 0.3/1 + 0.6/2 + 1/3 + 0.6/4 + 0.3/5$$
$$B = 0.5/10 + 1/11 + 0.5/12$$
$$C = \hat{f}(A, B)$$

$$\mu_C(11) = \max\{\min\{0.3, 0.5\}\} = 0.3$$

$$\mu_C(12) = \max\{\min\{0.5, 0.6\}, \min\{1, 0.3\}\} = \max\{0.5, 0.3\} = 0.5$$

$$\begin{aligned}\mu_C(13) &= \max\{\min\{0.5, 1\}, \min\{1, 0.6\}, \min\{0.5, 0.3\}\} \\ &= \max\{0.5, 0.6, 0.3\} = 0.6\end{aligned}$$

Extension Principle - Example

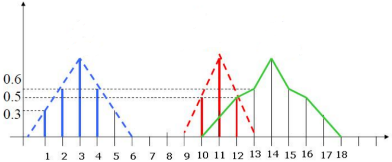
Consider the function $f(x_1, x_2) = x_1 + x_2$, defined from $\mathcal{X}^2 \to \mathcal{X}$

$$A = 0.3/1 + 0.6/2 + 1/3 + 0.6/4 + 0.3/5$$
$$B = 0.5/10 + 1/11 + 0.5/12$$
$$C = \hat{f}(A, B)$$

$$\mu_C(11) = \max\{\min\{0.3, 0.5\}\} = 0.3$$

$$\mu_C(12) = \max\{\min\{0.5, 0.6\}, \min\{1, 0.3\}\} = \max\{0.5, 0.3\} = 0.5$$

$$\mu_C(13) = \max\{\min\{0.5, 1\}, \min\{1, 0.6\}, \min\{0.5, 0.3\}\} = \max\{0.5, 0.6, 0.3\} = 0.6$$



Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set
- Hence determining a good membership function is a key issue

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set
- Hence determining a good membership function is a key issue
- Membership functions can be:
  - either be chosen by the user arbitrarily, based on experience

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set
- Hence determining a good membership function is a key issue
- Membership functions can be:
  - either be chosen by the user arbitrarily, based on experience
  - or be designed using machine learning methods

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set
- Hence determining a good membership function is a key issue
- Membership functions can be:
  - either be chosen by the user arbitrarily, based on experience
  - or be designed using machine learning methods
- Membership functions can take any form, but there are some common examples that appear in real applications

Fuzzy membership functions

- A fuzzy set is completely characterized by its membership function μ
- A membership function (MF) provides a measure of the degree of similarity of an element to a fuzzy set
- Hence determining a good membership function is a key issue
- Membership functions can be:
  - either be chosen by the user arbitrarily, based on experience
  - or be designed using machine learning methods
- Membership functions can take any form, but there are some common examples that appear in real applications
  - triangular, trapezoidal, piecewise-linear, Gaussian, bell-shaped, etc.

Fuzzy membership functions

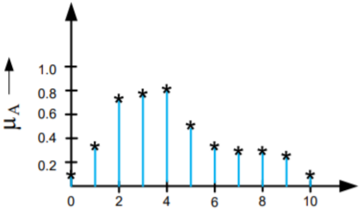
A membership function can be defined on:

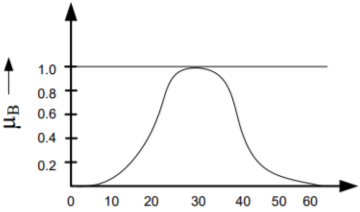
- a discrete universe of discourse, and
- a continuous universe of discourse

Fuzzy membership functions

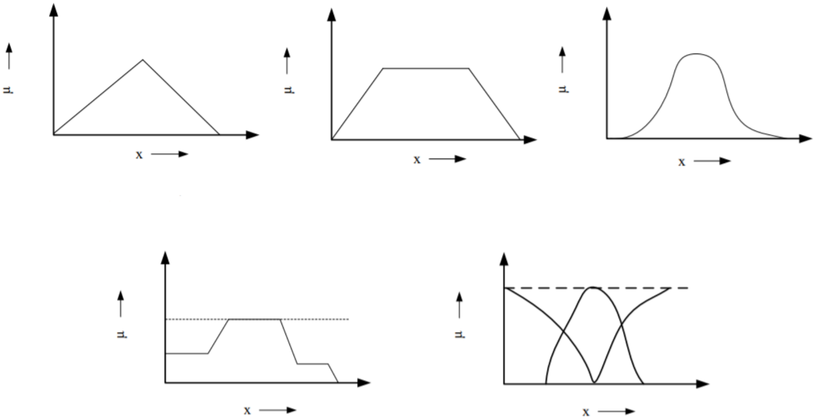
A membership function can be defined on:

- a discrete universe of discourse, and
- a continuous universe of discourse





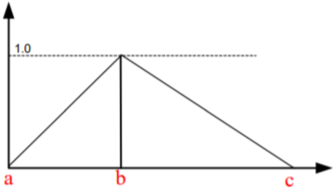
Fuzzy membership functions



Fuzzy membership functions - Triangular MF

A **triangular membership function** is specified by three parameters $\{a, b, c\}$ and can be defined as follows:

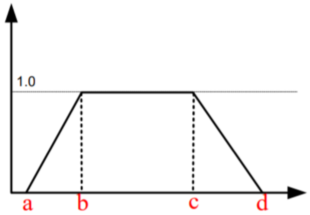
$$triangle(x; a, b, c) = \begin{cases} 0 \text{ if } x \leq a \\ \frac{x-a}{b-a} \text{ if } a \leq x \leq b \\ \frac{c-x}{c-b} \text{ if } b \leq x \leq c \\ 0 \text{ if } x \geq c \end{cases}$$



Fuzzy membership functions - Trapezoidal MF

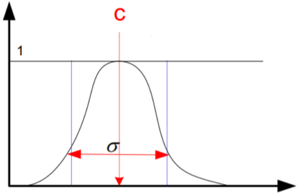
A **trapezoidal membership function** is specified by four parameters $\{a, b, c, d\}$ and can be defined as follows:

$$trapezoid(x; a, b, c, d) = \begin{cases} 0 \text{ if } x \leq a \\ \frac{x-a}{b-a} \text{ if } a \leq x \leq b \\ 1 \text{ if } b \leq x \leq c \\ \frac{c-x}{c-b} \text{ if } c \leq x \leq d \\ 0 \text{ if } x \geq d \end{cases}$$



Fuzzy membership functions - Gaussian MF

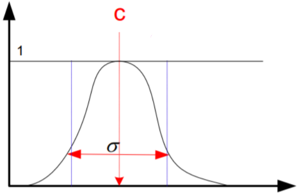
A **Gaussian membership function** is specified by two parameters $\{c, \sigma\}$ and can be defined as follows:



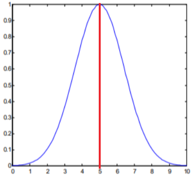
$$gaussian(x; c, \sigma) = e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2}$$

Fuzzy membership functions - Gaussian MF

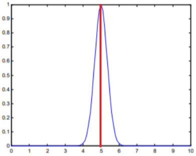
A **Gaussian membership function** is specified by two parameters $\{c, \sigma\}$ and can be defined as follows:



$$gaussian(x; c, \sigma) = e^{-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2}$$



c=5, s=0.5

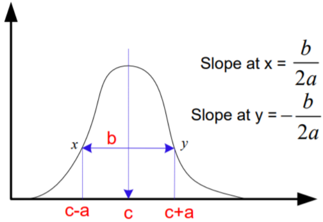


c=5, s=0.2

c=5, s=5

Fuzzy membership functions - Generalized Bell

A **Generalized Bell membership function** is specified by three parameters $\{a, b, c\}$ and is defined as follows:

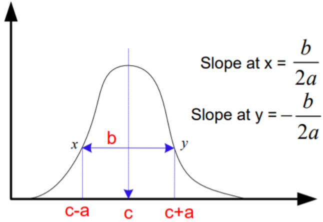


$$bell(x; a, b, c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}$$

- $a \rightarrow$ controls width
- $b \rightarrow$ controls slope
- $c \rightarrow$ represents center

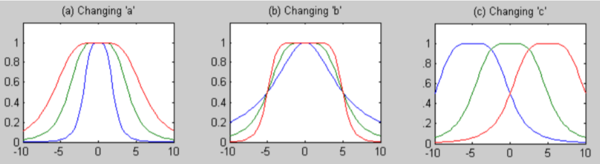
Fuzzy membership functions - Generalized Bell

A **Generalized Bell membership function** is specified by three parameters $\{a, b, c\}$ and is defined as follows:



$$bell(x; a, b, c) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}}$$

- $a \rightarrow$ controls width
- $b \rightarrow$ controls slope
- $c \rightarrow$ represents center



Control systems

- manages, commands, directs, or regulates the behavior of other devices or systems to achieve a desired result

Control systems

- manages, commands, directs, or regulates the behavior of other devices or systems to achieve a desired result
- system that controls other devices or systems

Control systems

- manages, commands, directs, or regulates the behavior of other devices or systems to achieve a desired result
- system that controls other devices or systems
- compares the value or status of the process variable (PV) being controlled with the desired value or setpoint (SP)

Control systems

- manages, commands, directs, or regulates the behavior of other devices or systems to achieve a desired result
- system that controls other devices or systems
- compares the value or status of the process variable (PV) being controlled with the desired value or setpoint (SP)
- applies the difference as a control signal to bring the process variable output to the same value as the setpoint

Control systems

- manages, commands, directs, or regulates the behavior of other devices or systems to achieve a desired result
- system that controls other devices or systems
- compares the value or status of the process variable (PV) being controlled with the desired value or setpoint (SP)
- applies the difference as a control signal to bring the process variable output to the same value as the setpoint
- find applications in air conditioners, refrigerators, washing machines etc.

Fuzzy control system

- a control system based on fuzzy logic

Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage

Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions

Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions **(fuzzification)**

Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions **(fuzzification)**
- the processing stage invokes a set of logic rules and combines the results of the rules into a set of fuzzy outputs

Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions **(fuzzification)**
- the processing stage invokes a set of logic rules and combines the results of the rules into a set of fuzzy outputs
- the output stage combines the fuzzy outputs into discrete values needed to drive the control mechanism

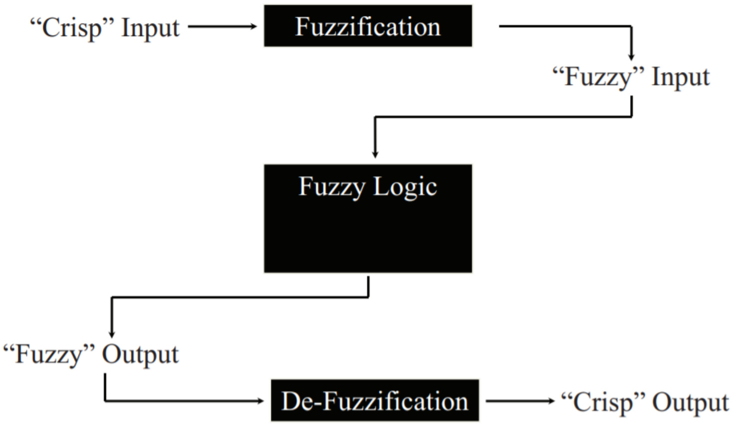
Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions **(fuzzification)**
- the processing stage invokes a set of logic rules and combines the results of the rules into a set of fuzzy outputs
- the output stage combines the fuzzy outputs into discrete values needed to drive the control mechanism **(defuzzification)**

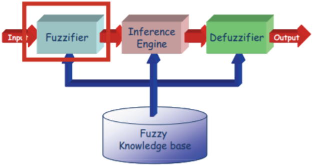
Fuzzy control system

- a control system based on fuzzy logic
- consist of an input stage, a processing stage, and an output stage
- the input stage maps the inputs from a set of sensors to values between 0 and 1 using a set of membership functions **(fuzzification)**
- the processing stage invokes a set of logic rules and combines the results of the rules into a set of fuzzy outputs
- the output stage combines the fuzzy outputs into discrete values needed to drive the control mechanism **(defuzzification)**
- fuzzy systems have the advantage of casting the solution to a problem in terms that humans can understand

Fuzzy control systems

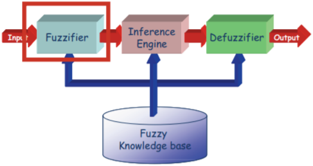


Fuzzification



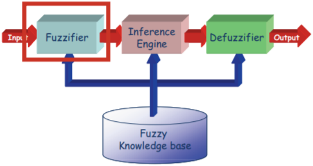
- process of making a crisp quantity fuzzy using membership functions

Fuzzification



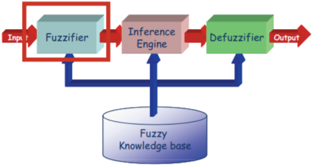
- process of making a crisp quantity fuzzy using membership functions
- hardware such as a digital voltmeter generates crisp data, but are subject to experimental error

Fuzzification



- process of making a crisp quantity fuzzy using membership functions
- hardware such as a digital voltmeter generates crisp data, but are subject to experimental error
- fuzzification helps deal with such ambiguity in data

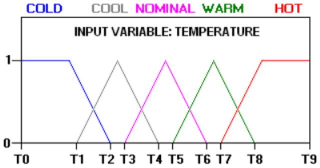
Fuzzification

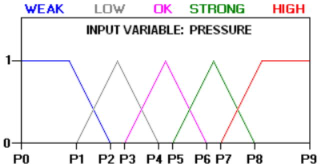


- process of making a crisp quantity fuzzy using membership functions
- hardware such as a digital voltmeter generates crisp data, but are subject to experimental error
- fuzzification helps deal with such ambiguity in data
- input variables are assigned degrees of membership to various classes

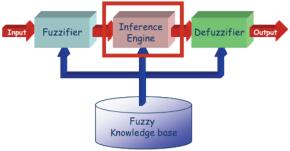
Fuzzification

For instance, the input variable temperature can be assigned membership to a number of fuzzy sets such as `cold`, `cool`, `nominal`, `warm`, `hot`, etc.



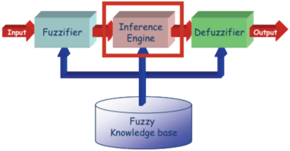


Inference Engine



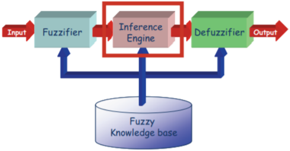
- converts fuzzy input into fuzzy output based on a collection of `IF-THEN` type logic rules
  - `IF` part is called the **antecedent**
  - `THEN` part is called the **consequent**

Inference Engine



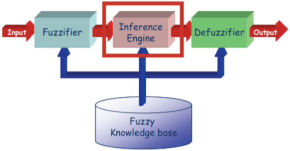
- converts fuzzy input into fuzzy output based on a collection of `IF-THEN` type logic rules
  - `IF` part is called the **antecedent**
  - `THEN` part is called the **consequent**
- example: "`IF (temperature is "cold") THEN turn (heater is "high")`"

Inference Engine



- converts fuzzy input into fuzzy output based on a collection of `IF-THEN` type logic rules
  - `IF` part is called the **antecedent**
  - `THEN` part is called the **consequent**
- example: "`IF (temperature is "cold") THEN turn (heater is "high")`"
- the truth value of the rules inferred using `MAX-MIN` method

Inference Engine



- converts fuzzy input into fuzzy output based on a collection of `IF-THEN` type logic rules
  - `IF` part is called the **antecedent**
  - `THEN` part is called the **consequent**
- example: "`IF (temperature is "cold") THEN turn (heater is "high")`"
- the truth value of the rules inferred using `MAX-MIN` method
- this method sets the fuzzy value of the antecedent to the output membership value

Inference Engine

Example

Rule 1: `IF` x_1 is **Small** `AND` x_2 is **High** `THEN` y is **Medium**

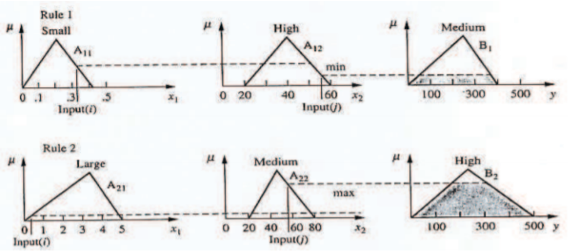
Rule 2: `IF` x_1 is **Large** `OR` x_2 is **Medium** `THEN` y is **High**

Inference Engine

Example

Rule 1: IF x_1 is **Small** AND x_2 is **High** THEN y is **Medium**

Rule 2: IF x_1 is **Large** OR x_2 is **Medium** THEN y is **High**



References

1. https://ekoharsono.files.wordpress.com/2013/01/5-fuzzy-numbers-arithmetic-and-the-extension-principle.pdf
2. http://web.fsktm.um.edu.my/~cschan/doc/IJCAI2.pdf
3. https://cse.iitkgp.ac.in/~dsamanta/courses/sca/resources/slides/FL-01%20Introduction.pdf
4. http://web.fsktm.um.edu.my/~cschan/doc/IJCAI2.pdf
5. https://cse.iitkgp.ac.in/~dsamanta/courses/archive/sca/Archives/Chapter%203%20Fuzzy%20Membership%20Functions.pdf
6. https://uomustansiriyah.edu.iq/media/lectures/5/5_2018_12_17!10_42_05_AM.pdf
7. https://en.wikipedia.org/wiki/Fuzzy_control_system