EL2620 Nonlinear Control

Lecture 13

- Fuzzy logic and fuzzy control
- Artificial neural networks

Some slides copied from K.-E. Årzén and M. Johansson

Today's Goal

You should

- understand the basics of fuzzy logic and fuzzy controllers
- understand simple neural networks

Fuzzy Control

- Many plants are manually controlled by experienced operators
- Transfer process knowledge to control algorithm is difficult

Idea:

- Model operator's control actions (instead of the plant)
- Implement as rules (instead of as differential equations)

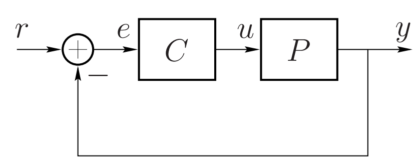
Example of a rule:

```
IF Speed is High AND Traffic is Heavy
   THEN Reduce Gas A Bit
```


Model Controller Instead of Plant

Conventional control design
Model plant $P \longrightarrow$ Analyze feedback $\longrightarrow$ Synthesize controller $C \longrightarrow$ Implement control algorithm

Fuzzy control design
Model manual control $\longrightarrow$ Implement control rules



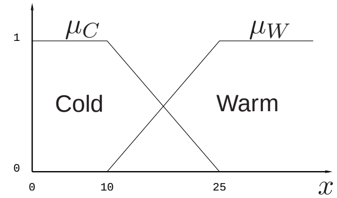
Fuzzy Set Theory

Specify how well an object satisfies a (vague) description

Conventional set theory: $x \in A$ or $x \notin A$

Fuzzy set theory: $x \in A$ to a certain degree $\mu_A(x)$

Membership function:
$\mu_A : \Omega \to [0, 1]$ expresses the degree x belongs to A

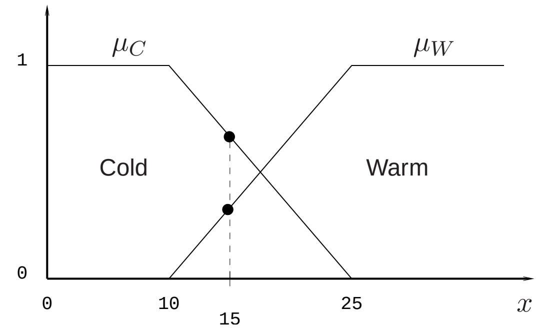


A **fuzzy set** is defined as (A, μ_A)

Example

Q1: Is the temperature $x = 15$ cold?
A1: It is quite cold since $\mu_C(15) = 2/3$.

Q2: Is $x = 15$ warm?
A2: It is not really warm since $\mu_W(15) = 1/3$.



Fuzzy Logic

How to calculate with fuzzy sets (A, μ_A)?

Conventional logic:
AND: $A \cap B$
OR: $A \cup B$
NOT: A'

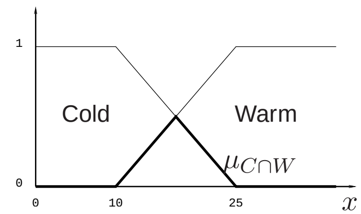
Fuzzy logic:
AND: $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$
OR: $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$
NOT: $\mu_{A'}(x) = 1 - \mu_A(x)$

Defines logic calculations as X AND Y OR Z

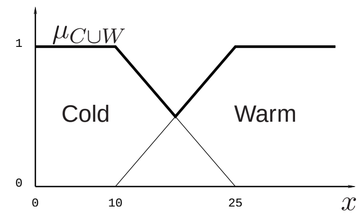
Mimic human linguistic (approximate) reasoning [Zadeh,1965]

Example

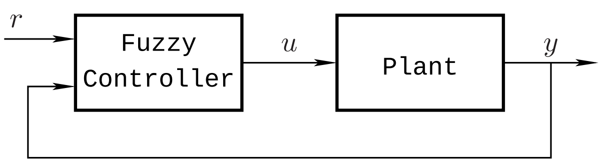
Q1: Is it cold AND warm?



Q2: Is it cold OR warm?



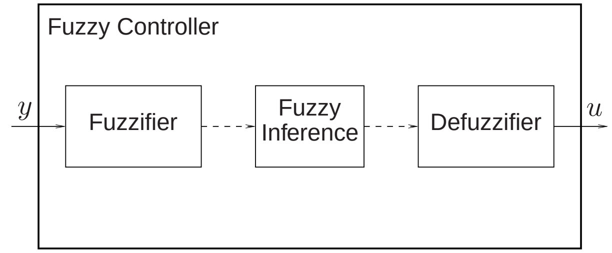
Fuzzy Control System



$r, y, u : [0, \infty) \mapsto \mathbb{R}$ are conventional signals

Fuzzy controller is a nonlinear mapping from y (and r) to u

Fuzzy Controller



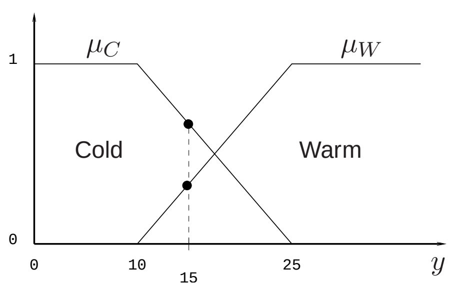
Fuzzifier: Fuzzy set evaluation of y (and r)
Fuzzy Inference: Fuzzy set calculations
Defuzzifier: Map fuzzy set to u

Fuzzifier and defuzzifier act as interfaces to the crisp signals

Fuzzifier

Fuzzy set evaluation of input y

Example
$y = 15$: $\mu_C(15) = 2/3$ and $\mu_W(15) = 1/3$



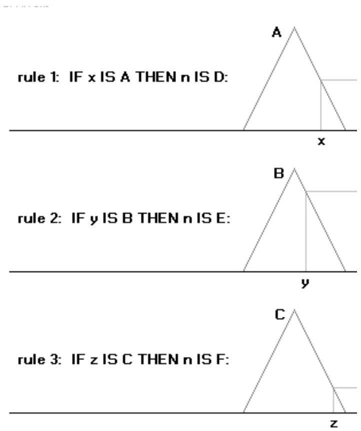
Fuzzy Inference

Fuzzy Inference:

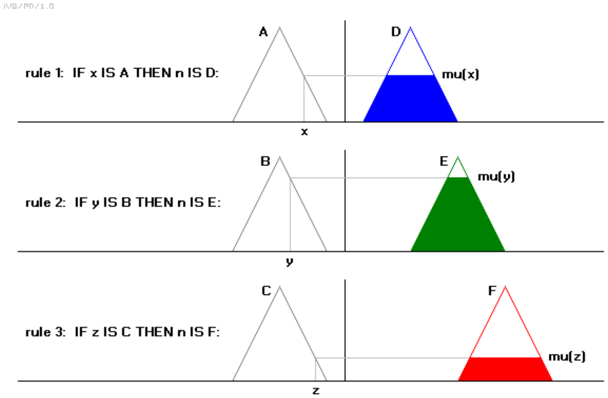
1. Calculate degree of fulfillment for each rule
2. Calculate fuzzy output of each rule
3. Aggregate rule outputs

Examples of fuzzy rules:
Rule 1: IF $\underbrace{y \text{ is Cold}}_{1.}$ THEN $\underbrace{u \text{ is High}}_{2.}$
Rule 2: IF $\underbrace{y \text{ is Warm}}_{1.}$ THEN $\underbrace{u \text{ is Low}}_{2.}$

1. Calculate degree of fulfillment for rules

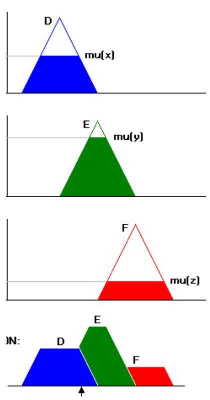


2. Calculate fuzzy output of each rule

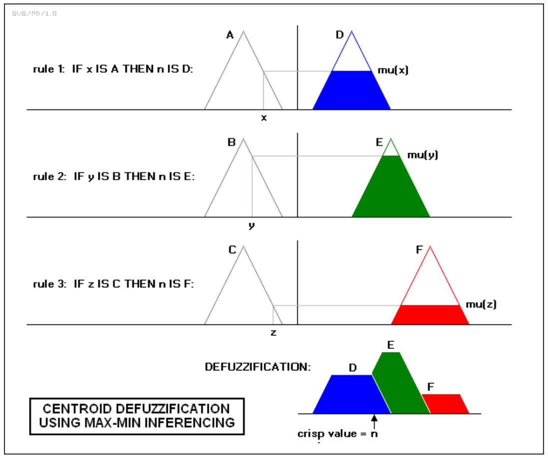


Note that "mu" is standard fuzzy-logic nomenclature for "truth value":

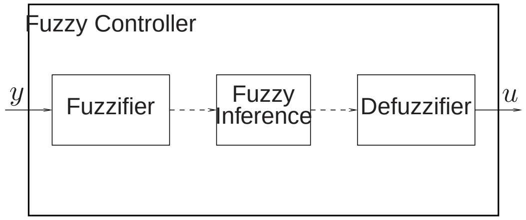
3. Aggregate rule outputs



Defuzzifier



Fuzzy Controller—Summary



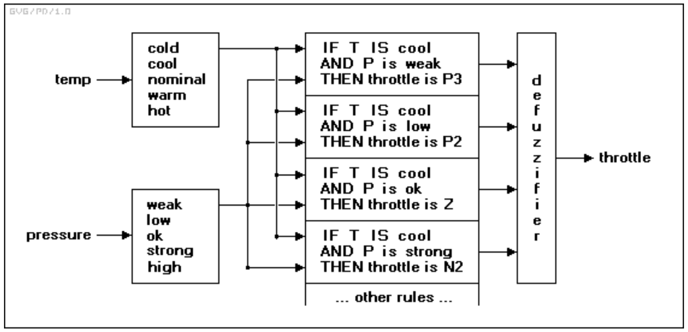
Fuzzifier: Fuzzy set evaluation of y (and r)

Fuzzy Inference: Fuzzy set calculations

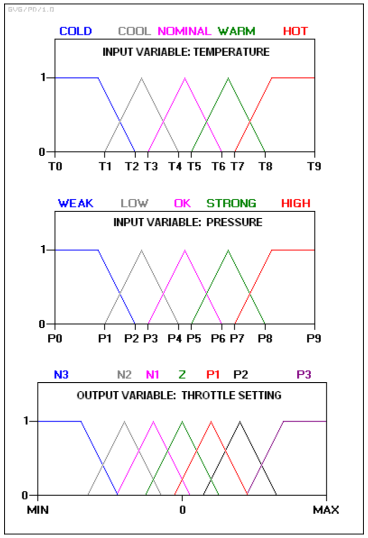
1. Calculate degree of fulfillment for each rule
2. Calculate fuzzy output of each rule
3. Aggregate rule outputs

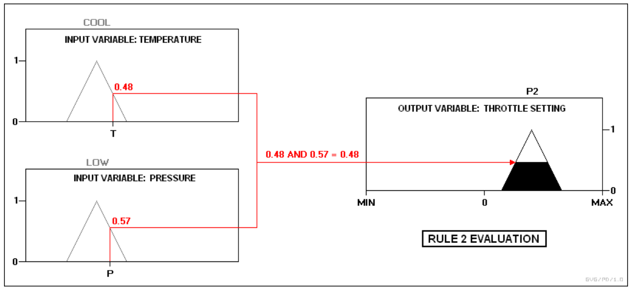
Defuzzifier: Map fuzzy set to u

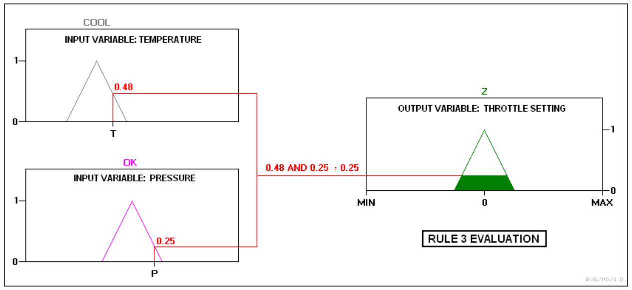
Example—Fuzzy Control of Steam Engine

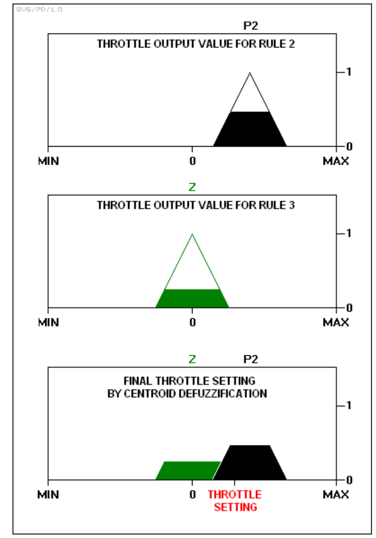


http://isc.faqs.org/docs/air/ttfuzzy.html

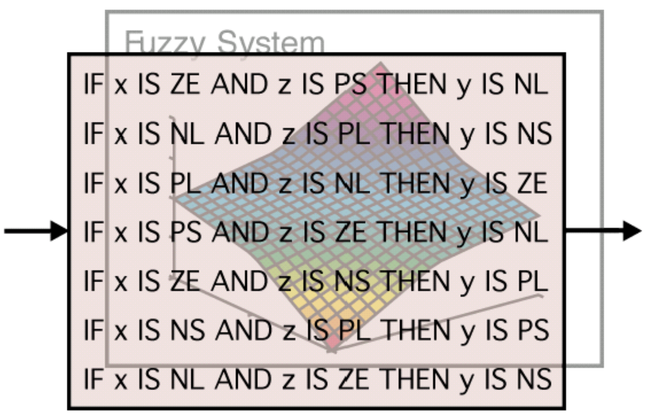




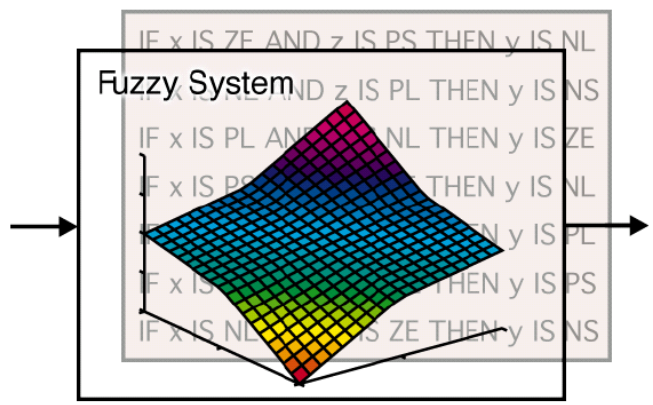




Rule-Based View of Fuzzy Control



Nonlinear View of Fuzzy Control



Pros and Cons of Fuzzy Control

Advantages

- User-friendly way to design nonlinear controllers
- Explicit representation of operator (process) knowledge
- Intuitive for non-experts in conventional control

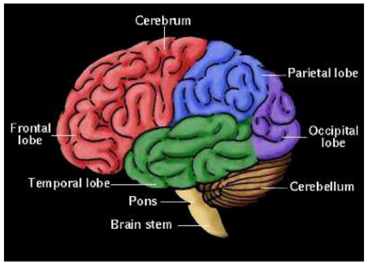
Disadvantages

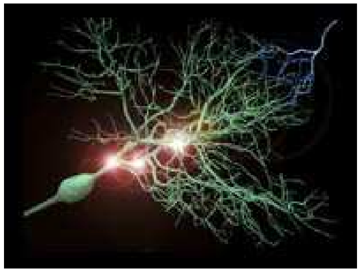
- Limited analysis and synthesis
- Sometimes hard to combine with classical control
- Not obvious how to include dynamics in controller

Fuzzy control is a way to obtain a class of nonlinear controllers

Neural Networks

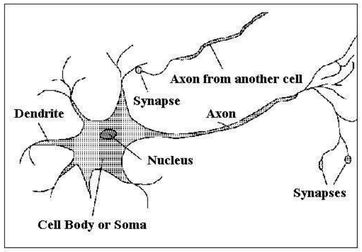
- How does the brain work?
- A network of computing components (neurons)

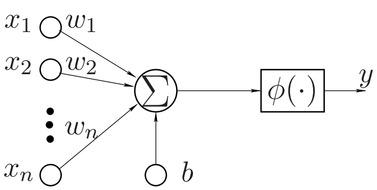




Neurons

Brain neuron **Artificial neuron**

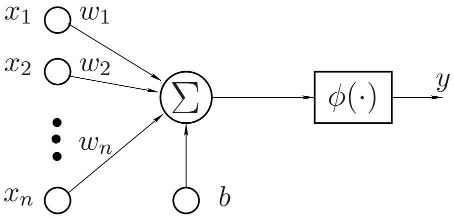




Model of a Neuron

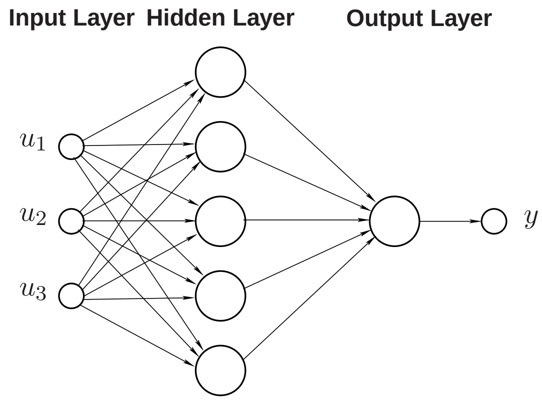
Inputs: $x_1, x_2, \ldots, x_n$
Weights: $w_1, w_2, \ldots, w_n$
Bias: b
Nonlinearity: $\phi(\cdot)$
Output: y

$$y = \phi\big(b + \sum_{i=1}^{n} w_i x_i\big)$$



A Simple Neural Network

Neural network consisting of six neurons:



Represents a nonlinear mapping from inputs to outputs

Neural Network Design

1. How many hidden layers?
2. How many neurons in each layer?
3. How to choose the weights?

The choice of weights are often done adaptively through **learning**

Success Stories

Fuzzy controls:

- Zadeh (1965)
- Complex problems but with possible linguistic controls
- Applications took off in mid 70's
  - Cement kilns, washing machines, vacuum cleaners

Artificial neural networks:

- McCulloch & Pitts (1943), Minsky (1951)
- Complex problems with unknown and highly nonlinear structure
- Applications took off in mid 80's
  - Pattern recognition (e.g., speech, vision), data classification

Today's Goal

You should

- understand the basics of fuzzy logic and fuzzy controllers
- understand simple neural networks

Next Lecture

- EL2620 Nonlinear Control revisited
- Spring courses in control
- Master thesis projects
- PhD thesis projects