Introduction to Robotics

Felipe P. Vista IV

Grading

- Attendance 5%

Name (Original Name)	User Email	Join Time	Leave Time	Duration (Minutes)
[redacted]		4/12/2021 9:12	4/12/2021 10:14	62
		4/12/2021 9:12	4/12/2021 9:14	3
		4/12/2021 9:12	4/12/2021 9:14	3
		4/12/2021 9:12	4/12/2021 9:14	3
		4/12/2021 9:12	4/12/2021 9:14	3
		4/12/2021 9:12	4/12/2021 9:14	3
		4/12/2021 9:13	4/12/2021 9:13	1
		4/12/2021 9:13	4/12/2021 9:14	2
		4/12/2021 9:14	4/12/2021 9:14	1
		4/12/2021 9:14	4/12/2021 9:14	1
		4/12/2021 9:14	4/12/2021 10:14	60

Bad ZOOM User Name (Absent)

- Iphone → Not your name
- SiAko 202100001 → Wrong order
- SiAko → Name only
- 202100001 → ID Num only

ZOOM User Name (Present)

- University ID Num_Name
- 202100001 SiAko → GOOD (Present)

Name (Original Name)	User Email	Total Duration (Minutes)
[redacted]		62
		63
		62
		62
		63
		62
		63

Student Responsibilities

- Download/Install ZOOM app for online lecture
  - *Zoom profile must be your* ***OASIS ID+name*** *similar to OASIS*
  - *Ex.:* ***202061234 YourName***
  - ***If you are asked, but no reply, then you'll be out of zoom & mark absent***
- Regularly login, check OLD IEILMS for updates, notifications
  - *https://ieilmsold.jbnu.ac.kr*
  - *Presentations & lecture videos will be uploaded after class*
- Regularly check Kakao Group Chat for class
  - *Everybody must have a Kakao talk account*
  - *Search & add account "****botjok****", introduce yourself and name of class ("Robotics"), then you will be added to the group chat*

Intro To Robotics

FUZZY LOGIC CONTROL

Intro

- Control algorithms in Section 6
  - *Exact math computations determine signals to control robot behaviour*
  - *Recall On-off, P, PI, PID Controllers*
- Alternate approach is use of **fuzzy logic**
  - *A control algorithm based on* **rules**

- For **Ex.:** Cruise control system might have rules of the form
  - *If car in front is far away or car in back is near, set the speed to fast.*
  - *If car in front is near, set the speed to slow.*
- Logic is "fuzzy" since rules expressed in linguistic variables
  - *Like* **speed** *whose values has no precise mathematical definition*
  - *But only imprecise linguistic specifications such as* **fast** *and* **slow**

Intro

(Expert)

- Fuzzy Logic controller has three phases run sequentially

1) Fuzzify
   - *Values of sensors converted into values of the linguistic variables*
     - Such as *far*, *closing*, *near*, called **premises**
   - *Each premise specify* **certainty** *w/c is prob of our belief that variable is true*
2) Apply Rules
   - *Set of rules expresses the control algorithm*
   - *Given set of* premises → *a* **consequent** *is inferred*
   - *Consequents also linguistic variables, ex: very fast, fast, cruise, slow, stop*
3) Defuzzify
   - *Consequents combined so as to produce* **crisp** *output*
     - A numerical value that controls robot aspects such as power to motor

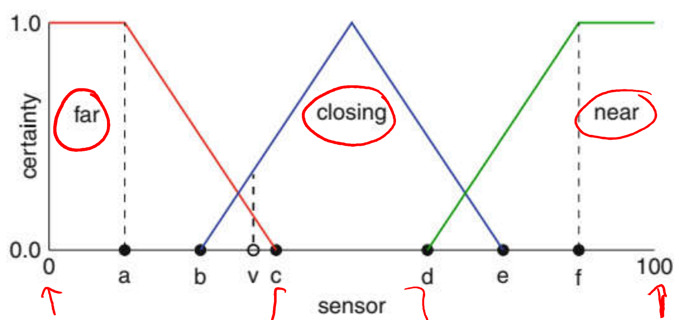
Fuzzy Logic Control

- Fuzzify
- Apply Rules
- Defuzzify

Fuzzify

Ex. Task : Robot approach an object & stop when very close to it

- When approaching an object
  - *Value read by horizontal proximity sensor increase from 0 → 100*
- Value returned by sensor is fuzzified
  - *By converting it to a value of a linguistic variable*
- Figure shows three graphs
  - *Convert sensor values into certainties of linguistic variables*

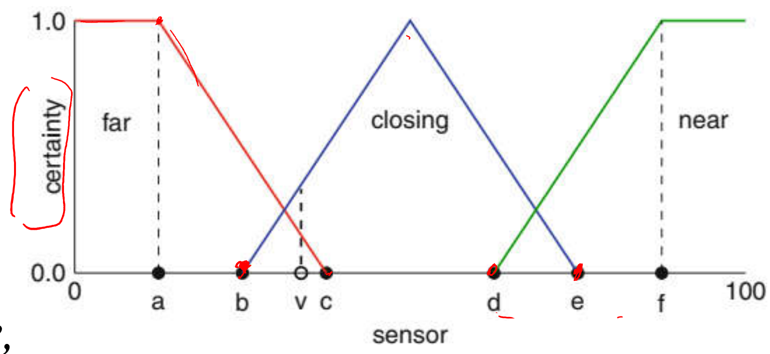


(a) Fuzzify the value of the horizontal proximity sensor

Fuzzify

Ex. Task : Robot approach an object & stop when very close to it

- Figure (a) shows three graphs
  - *Convert sensor values into certainties of linguistic variables* far, closing *&* near
  - X-axis : *Value returned by sensor*
  - Y-axis : *Gives the premise for each variable, the certainty that the linguistic variable is true*
- The points in x-axis refer to thresholds
  - *(a)* far_low, *(b)* closing_low, *(c)* far_high, *(d)* near_low, *(e)* closing_high *(f)* near_high

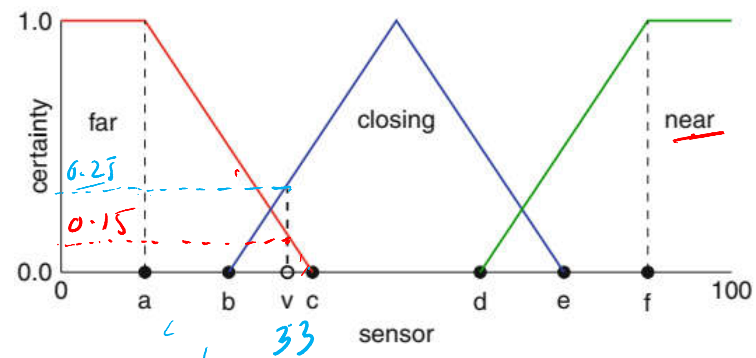


(a) Fuzzify the value of the horizontal proximity sensor

Fuzzify

Ex. Task : Robot approach an object & stop when very close to it

- The points in x-axis refer to thresholds
  - *(a)* far_low, *(b)* closing_low, *(c)* far_high, *(d)* near_low, *(e)* closing_high *(f)* near_high
  - *If (sensor value < far_low) → completely certain object far away w/ certainty of 1*
  - *If (closing_low < value < far_high) → somewhat certain object is far away, but also somewhat certain that it is closing.*



(a) Fuzzify the value of the horizontal proximity sensor

- Fuzziness results from the overlapping ranges
  - *When value is between points (b) and (c) → can't say w/ complete certainty if object is far away or closing*
  - **Ex.** *v = 33* → far *(certainty = 0.15)* & closing *(certainty = 0.25)*

Fuzzy Logic Control

- Fuzzify
- Apply Rules
- Defuzzify

Apply Rules

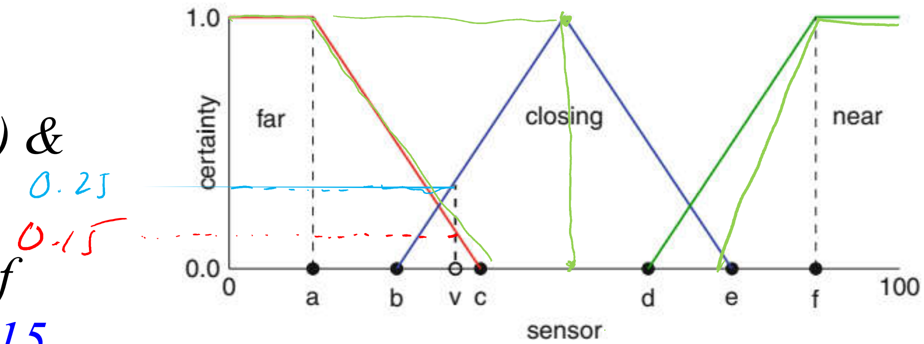
- The three premises, certainties of **far**, **closing** & **near**
  - *For computing five consequents using following rules*

1) If far then very fast
2) If far and closing then fast
3) If closing then cruise
4) If closing and near then slow
5) If near then stop

- Certainties of consequents
  - *Are same as certainties of corresponding premises*
    - When resulting from *Rules 1, 3, & 5*
  - *Are computed from the minimum of the certainties of the premises*
    - Such as when there are two premises (*Rules 2 & 4*)
    - Since **both** premises *must apply* → We can't be **more certain** of the consequent than we are of the smaller of the premises

Apply Rules

- For **value *v*** in the figure
  - **Ex.** *v = 33* → *far (certainty = 0.15) & closing (certainty = 0.25)*
  - *Rule 2* applies → *hence certainty of consequent is min(0.15, 0.25) = 0.15*



1) If far then very fast
2) If far and closing then fast
3) If closing then cruise
4) If closing and near then slow
5) If near then stop

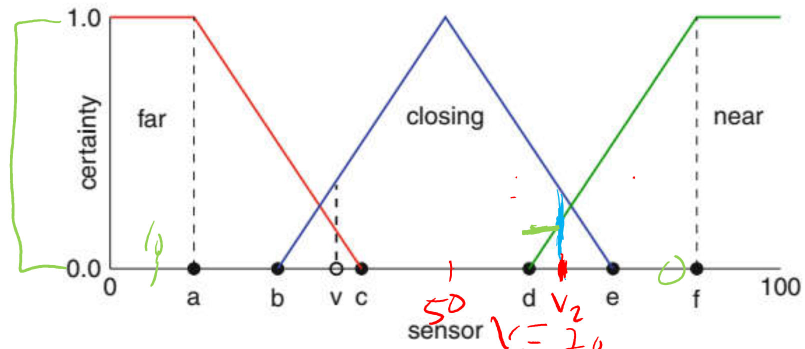
- Another way of combining premises
  - *Is to take their joint probability:*

$$p(A \cap B) = P(A) \cdot P(B)$$

  - **Ex.** *v = 33* → *far (certainty = 0.15) & closing (certainty = 0.25)*
  - *Hence, certainty of the consequent is ? ? ? ? ?, much less than certainty obtained from the minimum function.*

Apply Rules

- For **value *v*** in the figure
  - **Ex.** *v = 33* → *far (certainty = 0.15) & closing (certainty = 0.25)*
  - *Rule 2 applies* → *hence certainty of consequent is min(0.15, 0.25) = 0.15*



1) If far then very fast
2) If far and closing then fast
3) If closing then cruise
4) If closing and near then slow
5) If near then stop

- Another way of combining premises
  - *Is to take their joint probability:*

$$p(A \cap B) = P(A) \cdot P(B)$$

  - **Ex.** *v = 33* → *far (certainty = 0.15) & closing (certainty = 0.25)*
  - *Hence, certainty of the consequent is (0.15)(0.25) = 0.0375, much less than certainty obtained from the minimum function.*

Fuzzy Logic Control

- Fuzzify

Sensor value (Fuzzify) → Linguistic

(D.Nyu) ~ 0.33

Far - 0.75

Closing - 0.25

Near - 0

- Apply Rules

Far

closing

Near

stop

slow

cruise - 16

Fast - 9

Very Fast

m.l = 50

m.r = 76

- Defuzzify

far closing near

stop

slow

cruise

Fast

Very Fast

Motor Value = 59

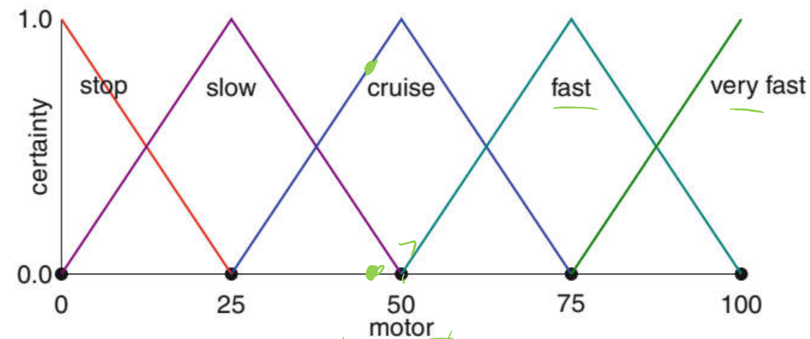
(D. Nyu)

motor.left ← 59

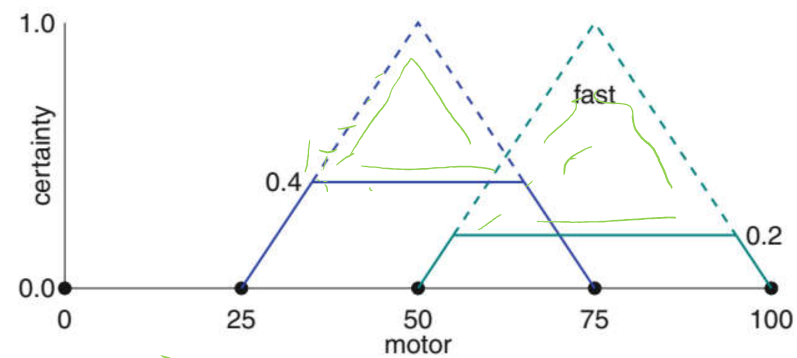
motor.right ← 59

Defuzzify

- Next step combining the consequents
  - *Taking into account their certainties*
- Figure given in (a)
  - *motor powers for each five consequents*
  - Ex.: *If completely certain output is* **cruise** → *motor power should be set to* **50**; *If less certain* → *motor power should be less or more*
- Suppose certainty of consequent of **cruise** computed as **0.4**
  - *Center triangle in (a) no longer relevant*
    - Since certainty can never be more than 0.4 displayed as a trapezoid in (b)



(a) Defuzzify to obtain crisp motor setting

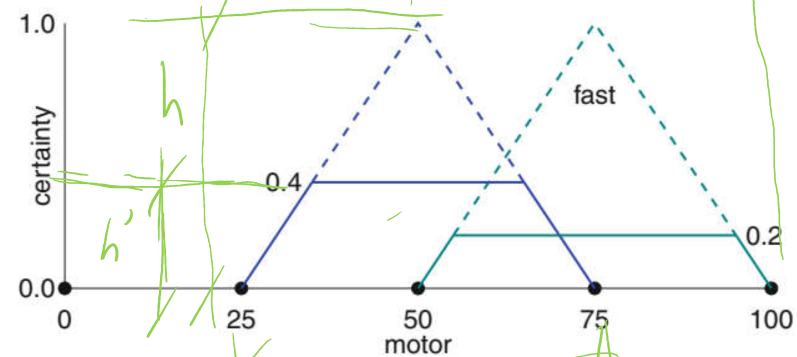


(b) Areas defined by certainties of the consequents

Defuzzify

100 − 50 (50)

- Let ***w*** & ***h*** be width & height of triangle
  - *The area of trapezoid bounded by line at height h' is:* $wh'\left(1-\dfrac{h'}{2h}\right)$
- Possible more than one consequent
  - *Can have positive values*
  - *Trapezoids for consequent* **cruise** (certainty=0.4) & **fast** (certainty=0.2) *given in figure.*
  - *For* $w=50, h=1, h'_c=0.4(cruise), h'_f=0.2(fast)$, *areas of the trapezoids* $a_c(cruise)$ and $a_f(fast)$ *are:*



Areas defined by certainties of the consequents

75 − 25 (50)

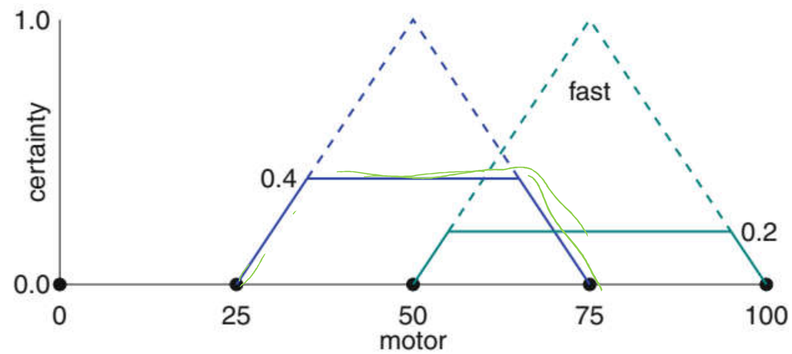
$a_c = ?????$, $a_f = ?????$

$a_c = 50(0.4)\left(1-\frac{0.4}{2(1)}\right) = 20(1-0.2) = 20(0.8) = 16$

$= 50(0.2)\left(1-\frac{0.2}{2}\right) = 10(1-0.1) = 10(0.9) = 9.0$

Defuzzify

- Let ***w*** & ***h*** be width & height of triangle
  - *The area of trapezoid bounded by line at height h' is:* $wh'\left(1-\frac{h'}{2h}\right)$
- Possible more than one consequent
  - *Can have positive values*
  - *Trapezoids for consequent* **cruise** (certainty=0.4) & **fast** *(*certainty=0.2*) given in figure.*
  - *For* $w=50, h=1, h'_c=0.4(cruise), h'_f=0.2(fast)$, *areas of the trapezoids* $a_c(cruise)$ and $a_f(fast)$ *are:*



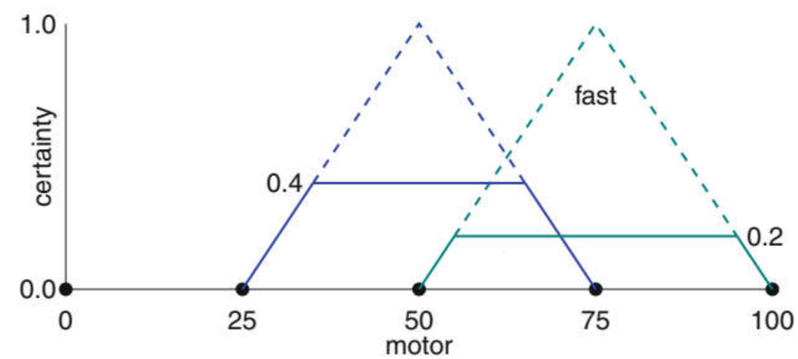
Areas defined by certainties of the consequents

$$a_c=(50)(0.4)\left(1-\frac{0.4}{2}\right)=16,\quad a_f=(50)(0.2)\left(1-\frac{0.2}{2}\right)=9$$

Defuzzify

- To obtain a crisp value
  - *The* **center of gravity** *is computed*
  - *It is sum of the areas of the trapezoids, weighed by value at center of the base of each trapezoid, divided by sum of the areas. i.e.:*

$$\frac{a_c c_c + a_f c_f}{a_c + a_f} = ?????$$

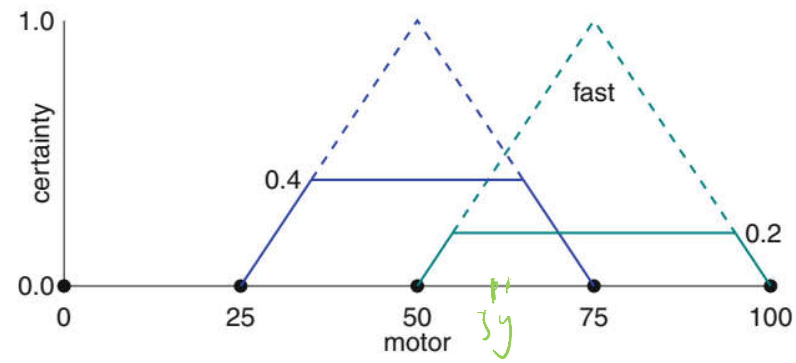


- The value is closer to
  - *Value associated with* **cruise** *rather than value associated with* **fast**
  - *Not surprising since the certainty of* **cruise** *is greater than the certainty of* **fast**

Defuzzify

- To obtain a crisp value
  - *The* **center of gravity** *is computed*
  - *It is sum of the areas of the trapezoids, weighed by value at center of the base of each trapezoid, divided by sum of the areas. i.e.:*

$$\frac{a_c c_c + a_f c_f}{a_c + a_f} = \frac{(16)(50)+(9)(75)}{16+9} = 59$$



- The value is closer to
  - *Value associated with* **cruise** *rather than value associated with* **fast**
  - *Not surprising since the certainty of* **cruise** *is greater than the certainty of* **fast**

Summary

- Fuzzy Logic Control
  - *An alternative to classical control algorithms (On-Off, P, PI, PID)*
- Advantage
  - Does not demand precise mathematical specifications of robot's behaviour
    - Which might be difficult to define
- Disadvantage
  - *Behaviour of is not transparent as that of classical control algorithms*
- We studied an example for fuzzy definitions of speed
- Other examples can be
  - *color : when does shade of range becomes orange*
  - *temperature : when does a warm room become hot*

Thank you.