Fuzzy Logic

Artificial Intelligence for
Interactive Media and Games

Professor Charles Rich
Computer Science Department
rich@wpi.edu

[Based on Buckland, Chapter 10]

Outline

- Background and Motivation
  - vagueness and discretization
  - application to weapon selection in Raven
  - fuzzy versus classical logic
  - DOM versus probabilities
- Theory and Algorithms
  - fuzzy set membership
  - linguistic variables
  - fuzzification and defuzzification
  - rule inference
- Implementation

Motivation

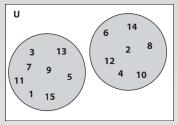
- Linguistic vagueness
  - "if the ball is far from the hole and the green is sloping gently downward from the left to the right, then hit the ball firmly and at an angle slightly to the left of the flag"
- Numerical discretization
  - Dumb: IQ < 90
  - Average: 90 ≤ IQ ≥ 110
  - Clever: 110 < IQ

Should you call a person with IQ 89 dumb, but with 90 average ?!

Motivation

- Examples in Raven weapon selection
  - if the target is far and you have lots of ammo, then the rocket launcher is a desirable choice
  - if target is at medium range and you have lots of ammo, then the rocket launcher is a very desirable choice

Classic ("Crisp") Sets



- Universe (of discourse)
- Characteristic (membership) function (predicate)
  - even: U ➔ boolean
  - odd: U ➔ boolean

  even(2) = T, even(3) = F, etc.

  singleDigit(2) = T, singleDigit(10) = F, etc.
- *Operators:* union, intersection, complement
  - and, or, not for characteristic predicates

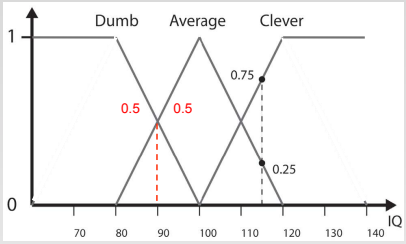
Fuzzy Set Membership

- Range of membership function for each set generalized from boolean to *real interval* (0,1)
  - dumb: IQ ➔ (0, 1)
  - average: IQ ➔ (0, 1)
  - clever: IQ ➔ (0, 1)
- A given value can be a member of *more* than one set with different *degrees*, e.g.,

  dumb(**20**) = 1.0, dumb(**89**) = 0.5, dumb(**90**) = 0.5

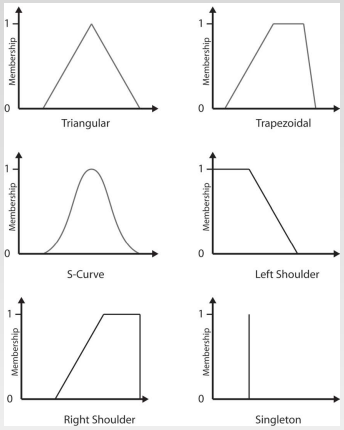
  average(**20**) = 0.0, average(**89**) = 0.5, average(**90**) = 0.5
  - *degree of membership (DOM)* shifts *gradually* as value changes

Fuzzy Set Membership



- membership functions can be any shape
- but for given value, degrees of membership in all sets (in "grouping", tbd) should sum to 1.0

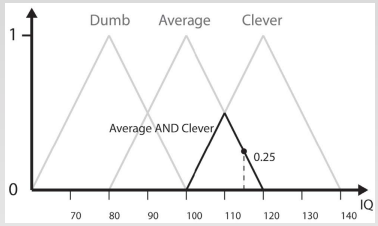
Membership Function Shapes



Membership versus Probability

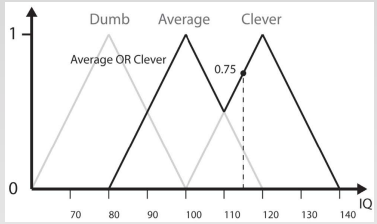
- Fuzzy logic closely related to probabilistic logics
  - both use real interval (0, 1)
- Probability reflects *uncertainty of outcome*
  - especially for repeated events (coin toss, etc.)
  - if I reach in a bag with 1 green ball and 3 red balls, there is a 25% chance I will pull out a green ball
- Fuzzy sets reflect *conceptual uncertainty*
  - e.g., is this weird-colored ball green or red?
  - "confidence value"

Fuzzy Set Operators



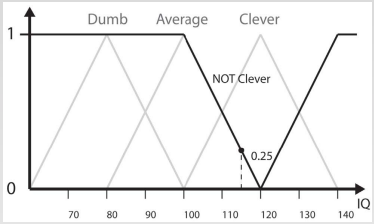
$$F_{A\cap B}(x) = \min\{F_A(x), F_B(x)\}$$

Fuzzy Set Operators



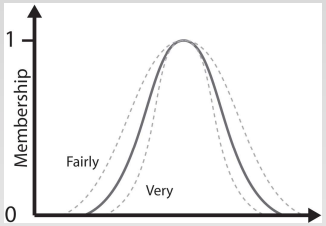
$$F_{A\cup B}(x) = \max\{F_A(x), F_B(x)\}$$

Fuzzy Set Operators



$$F_{\bar{A}}(x) = 1 - F_A(x)$$

Hedges

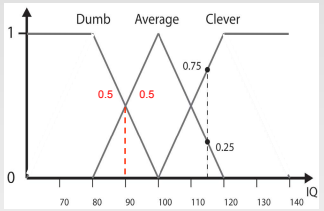


$$F_{Very(A)}(x) = [F_A(x)]^2$$

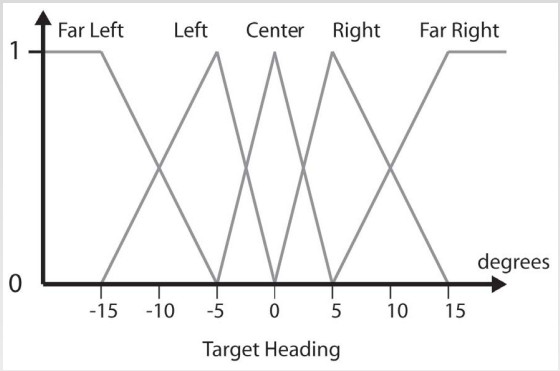
$$F_{Fairly(A)}(x) = \sqrt{F_A(x)}$$

Fuzzy Linguistic Variable

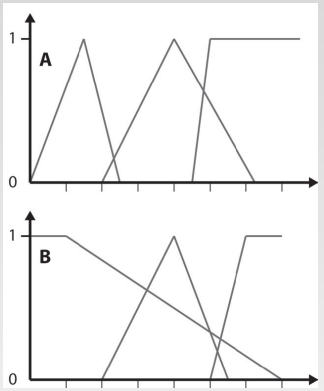
- conceptual grouping of several fuzzy sets (membership functions) with the same domain (universe)
  - IQ = { dumb, average, clever }



Target Heading Variable



Linguistic Variable Design Guidelines



BAD: values don't add to 1.0

BAD: values belong to more than two sets

Fuzzy Rules

IF ***antecedent*** **THEN** ***consequent***

- degree of *membership* of given value in the *antecedent* set determines the degree of *confidence* in the *consequent*
- antecedent and consequent may be primitive fuzzy sets or expressions composed with operators

Fuzzy Rules

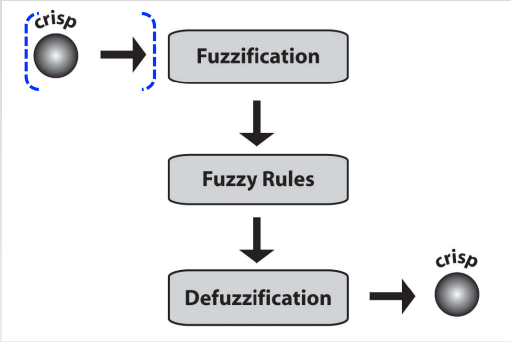
IF Target_isFarRight THEN Turn_QuicklyToRight

IF Very(Enemy_BadlyInjured) THEN Behavior_Aggressive

IF Ball_isCloseToHole AND Green_isLevel THEN HitBall_Gently AND HitBall_DirectlyAtHole

IF Target_Medium AND Ammo_Low THEN RocketLauncher_Desirable

Fuzzy Rule Inference



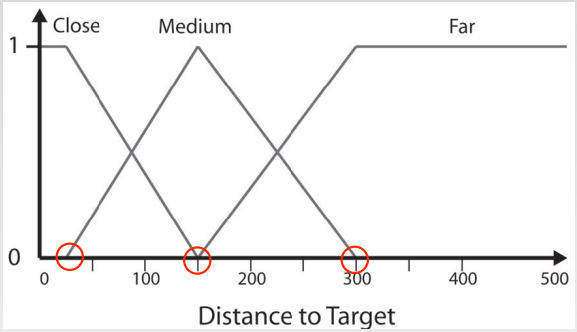
Raven Weapon Selection Example

1. Decide on antecedent and consequent linguistic variables
2. Design fuzzy membership functions for each variable
3. Define rules using variables

Raven Weapon Selection Example

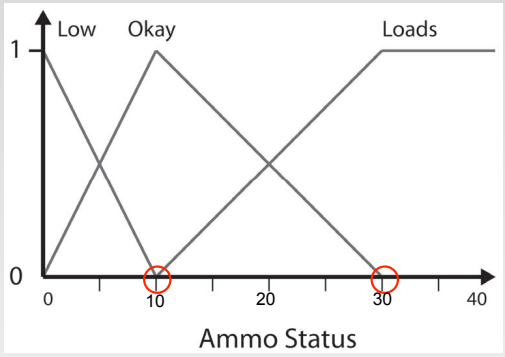
- Weapon selection depends on (antecedents):
  - distance to target
  - ammo status
- Conclusion (consequent) is
  - desirability of weapon
- Separate sets of rules for each weapon

Designing Membership Functions



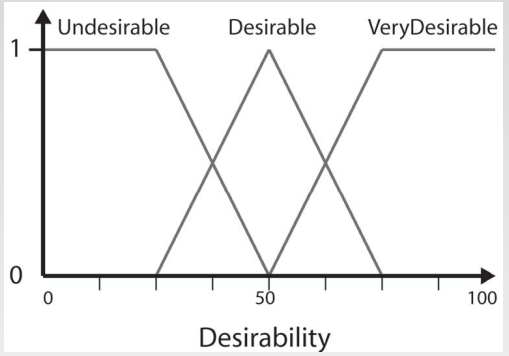
Rocket Launcher

Designing Membership Functions

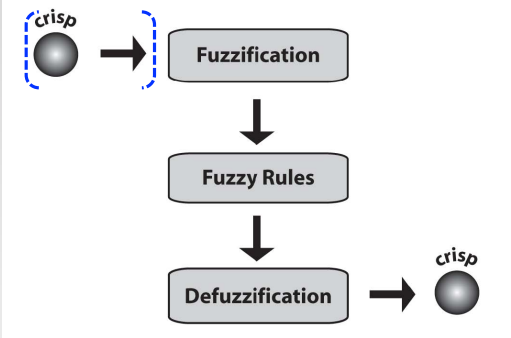


Rocket Launcher

Designing Membership Functions



Fuzzy Rule Inference



Rocket Launcher Selection Rules

(1) IF Target_Far AND Ammo_Loads THEN Desirable
(2) IF Target_Far AND Ammo_Okay THEN Undesirable
(3) IF Target_Far AND Ammo_Low THEN Undesirable

(4) IF Target_Medium AND Ammo_Loads THEN VeryDesirable
(5) IF Target_Medium AND Ammo_Okay THEN VeryDesirable
(6) IF Target_Medium AND Ammo_Low THEN Desirable

(7) IF Target_Close AND Ammo_Loads THEN Undesirable
(8) IF Target_Close AND Ammo_Okay THEN Undesirable
(9) IF Target_Close AND Ammo_Low THEN Undesirable

** can reduce to 6 rules by Comb's Method*

Fuzzy Inference – Running the Rules

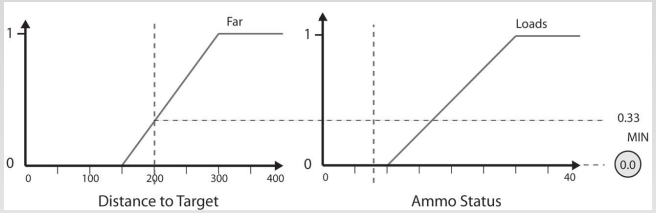
1. Given an input value for each linguistic variable used in the rule antecedents
2. For each rule
   - calculate degree of confidence in conclusion from degree of membership of input value in antecedent
3. Combine all the inferred conclusions into a single fuzzy variable
4. Defuzzify the conclusion to single (crisp) output value

Running the Rules

- Input values:
  - Distance to Target = 200 pixels
  - Ammo = 8 rockets
- Rule (1)

  IF Target_Far AND Ammo_Loads THEN Desirable
  - Target_Far(200) = 0.33
  - Ammo_Loads(8) = 0.0
  - Desirable = min(0.33, 0.0) = 0.0

Running the Rules

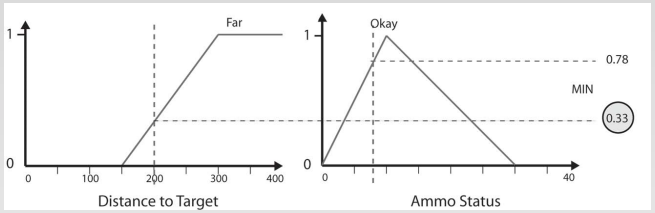


- Rule (1)

  IF Target_Far AND Ammo_Loads THEN Desirable

  - Target_Far(200) = 0.33
  - Ammo_Loads(8) = 0.0
  - Desirable = min(0.33, 0.0) = 0.0

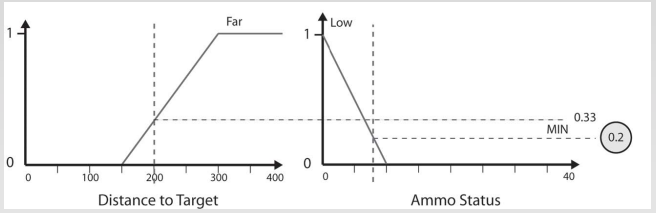
Running the Rules



- Rule (2)

  IF Target_Far AND Ammo_Okay THEN Undesirable

Running the Rules



- Rule (3)

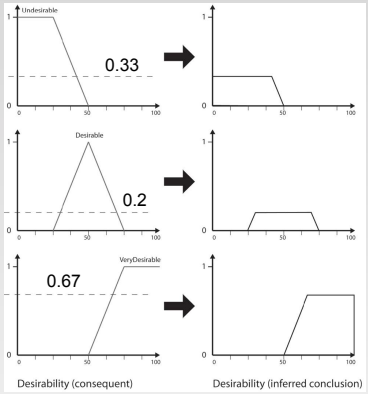
  IF Target_Far AND Ammo_Low THEN Undesirable

Fuzzy Associative Matrix

	Target_Close	Target_Medium	Target_Far
Ammo_Low	Undesirable 0	Desirable 0.2	Undesirable 0.2
Ammo_Okay	Undesirable 0	VeryDesirable 0.67	Undesirable 0.33
Ammo_Loads	Undesirable 0	VeryDesirable 0	Desirable 0

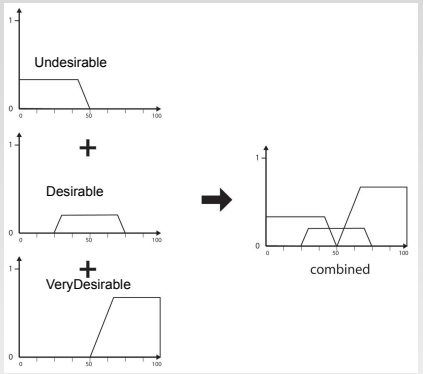
- Shaded cells from fired rules
- What should confidence level be for Undesirable?
- How to <u>combine</u> output values?
- Use max (0.33)

Inferred Consequent Sets



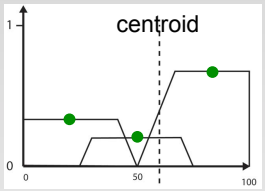
- results of each rule OR'ed together with underlying membership function
- "clips" each shape

Combined Output Variable



- output variable could feed into other rules
- or if the end of the line, extract a single value ("defuzz")

Defuzzification – Extracting a single value



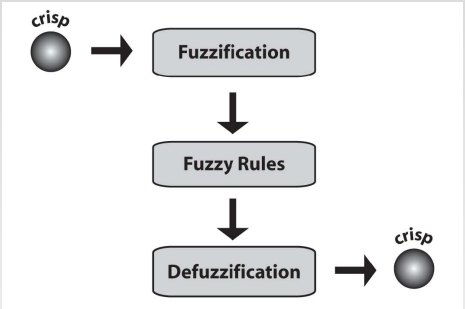
Rocket Launcher
Desirability = 60.625

- Average of Maxima (MaxAv)
  - good approximation to centroid
  - weighted sum of representative values

$$\frac{\sum(representativeValue \times confidence)}{\sum confidence}$$

Algorithm Summary

Distance to target = 200 pixels
Ammo = 8 rockets

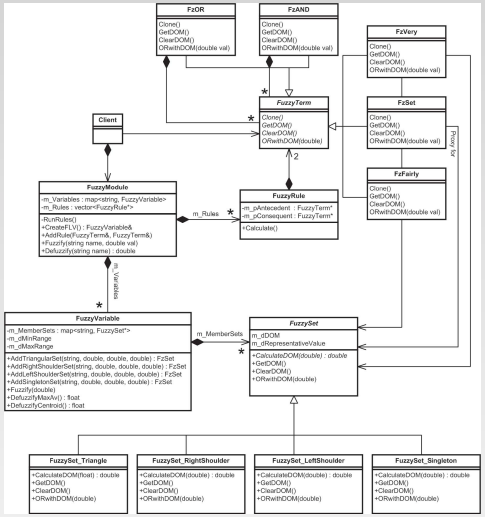


Rocket Launcher Desirability = 60.625

Implementation Classes

- FuzzyModule
- FuzzySet
- FuzzyVariable
- FuzzyTerm
- Fuzzy Operator
- Fuzzy Rule

...for weapon selection in Raven



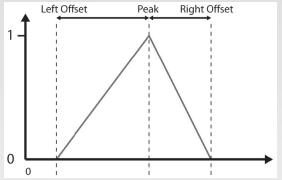
FuzzyModule

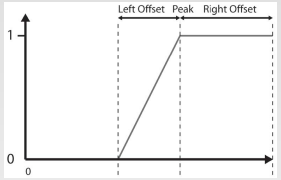
- Main members
  - linguistic variables
    - DistToTarget, Desirability, etc.
  - rule base
    - IF Target_Close AND Ammo_Low THEN Undesirable
    - etc.
- Instance for each “client”
  - Raven_Weapon

[see code]

FuzzySet

- Base class for different “shapes”
  - Triangle, Shoulders, etc.
  - FzSet proxy (wrapper) class

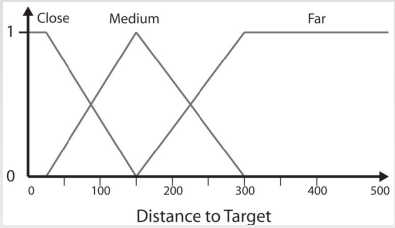




[see code]

FuzzyVariable

- holds collection of fuzzy sets
  - Close, Medium, Far, etc.
- only supports number (double) universe



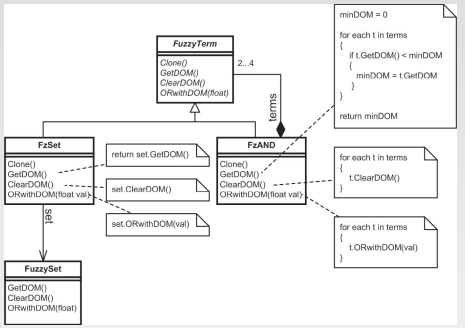
[see code]

FuzzyTerm and FuzzyOperator

Very(A) AND (B OR C)

- Composite design pattern

[see code]



FuzzyRule

IF Very(A) AND (B OR C) THEN D

fm.AddRule(FzAND(FzVery(A), FzOR(B,C)), D);

[see code]

Raven Weapon Selection

- each weapon instance contains a FuzzyModule instance
  - for tournament play, each bot could keep private rule base for each type of weapon
  - override WeaponSystem::SelectWeapon in bot-specific code
- highest desirability weapon chosen

[see code]

Combs Method

- avoids combinatorial explosion in rules
  - as number of variables increases

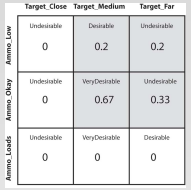
Variables	Traditional Rules	Combs Rules
2	25	10
3	125	15
4	625	20
5	3,125	25
6	15,625	30
7	78,125	35
8	390,625	40

Combs Method

(1) IF Target_Far AND Ammo_Loads THEN Desirable
(2) IF Target_Far AND Ammo_Okay THEN Undesirable
(3) IF Target_Far AND Ammo_Low THEN Undesirable

(4) IF Target_Medium AND Ammo_Loads THEN VeryDesirable
(5) IF Target_Medium AND Ammo_Okay THEN VeryDesirable
(6) IF Target_Medium AND Ammo_Low THEN Desirable

(7) IF Target_Close AND Ammo_Loads THEN Undesirable
(8) IF Target_Close AND Ammo_Okay THEN Undesirable
(9) IF Target_Close AND Ammo_Low THEN Undesirable



(1) IF Target_Close THEN Undesirable
(2) IF Target_Medium THEN VeryDesirable
(3) IF Target_Far THEN Undesirable

(4) IF Ammo_Low THEN Undesirable
(5) IF Ammo_Okay THEN Desirable
(6) IF Ammo_Loads THEN VeryDesirable

Combs Method

- Based on logical equivalence

IF (A AND B) THEN C = (IF A THEN C) OR (IF B THEN C)

- *Arbitrary* set of traditional rules *cannot* be written in Combs
- But many fuzzy associative matrixes commonly can
- Easier to *start* writing in restricted format
- For more details see Millington, Sec. 5.4
- Or original Combs paper