Decision Making: Fuzzy Logic

2016-06-28

Questions (N-3)

1. How can we describe decision making?
2. What do the algorithms we’ve seen share?
3. What are the dimensions we tend to assess?
4. FSMs/Btrees: _____ :: Planning : ______
5. For the 2^{nd} blank, we need m______s.
6. When is reactive appropriate? Deliberative?
7. What is the ‘hot-potato’ passed around (KE)?
8. H_______ have helped in most approaches.
9. Which approach should you use?

Questions (N-2)

1. What are the 2 most “complex” decision making techniques we’ve seen?
2. What are their strengths? Weaknesses?
3. What is the key (insight) to their success?
4. What is typically necessary to support this insight (hint: used in Planning + RBS)?
5. What does Planning have that (forward chaining) RBS do not?
6. When do we need a communication mechanism?

Questions (N-1)

1. Cooperative problem solving / distributed expertise is using h___ to d__ problems into smaller parts.
2. R__ experts rarely communicate/collaborate.
3. Three types of communication are…
4. The three main parts of a Blackboard are…
5. An Arbiter can be used to…

enables a computer to reason about linguistic terms and rules in a way similar to humans

FUZZY LOGIC

1. Cut two slices of bread **medium thick**.
2. Turn the heat on the griddle on **high**.
3. Grill the slices on one side until **golden brown**.
4. Turn the slices over and add a **generous helping** of cheese.
5. Replace and grill until the top of the cheese is **slightly brown**.
6. Remove, sprinkle on a **small amount** of black pepper, and eat.

Motivation

- Fuzzy logic: truth degrees, vagueness, subjectivity
- E.g.: Cautious vs Confident
  - FSM w/ 2 states – switching looks unnatural
  - Cautious (range), sneak slowly (range)
  - Confident, walk normally
- Caveat emptor:
  - relatively popular in games industry
  - largely discredited in academic AI (?)

Example

- Have to make Golfing game, ask expert…
  - When putting: if **ball far from hole**, and **green is slightly downward** from left to right THEN **hit ball firmly** and at **angle slightly** to left of flag
  - When driving: if **wind is strong**, and **blowing right to left**, and **hole is far away** THEN **hit ball hard** and at **angle far to right** of flag
- Close: [0m,2m), Medium: [2m-5m), Far: [5m
- Ball at 4.99m? We want a gradual shift.

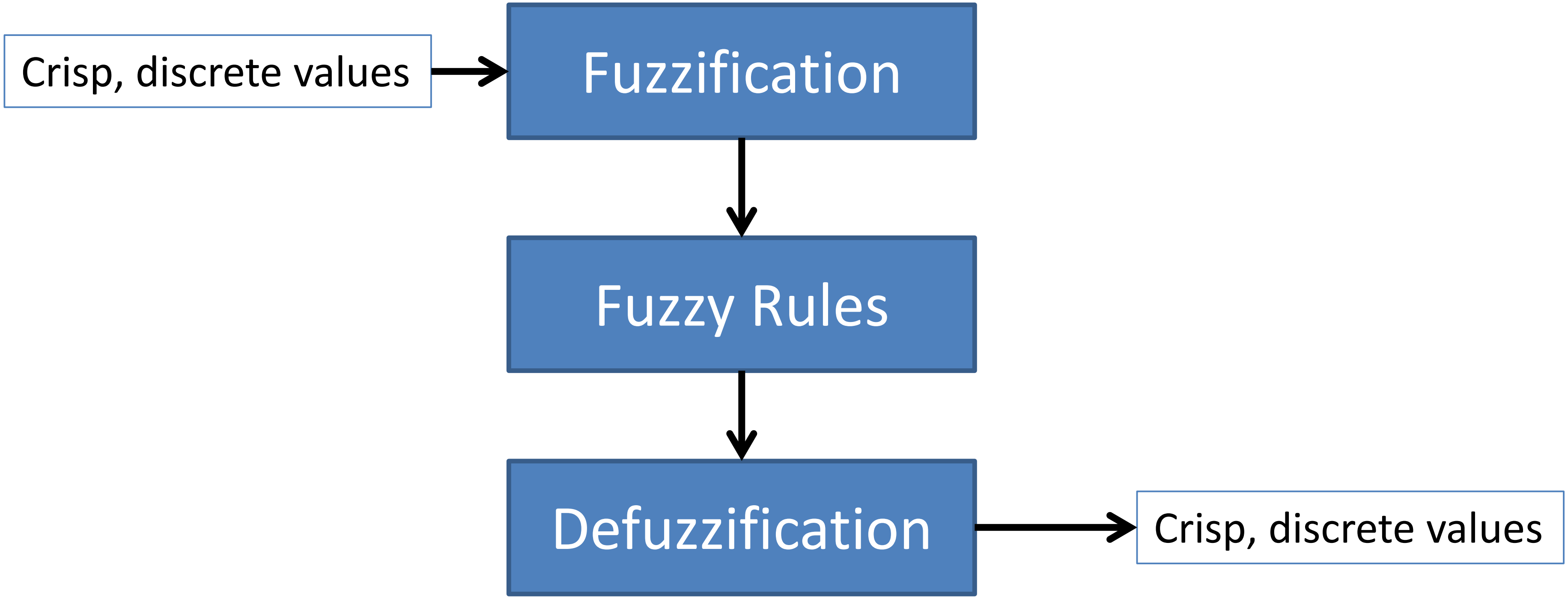
The Principle

- Traditional (binary set) logic: predicates (F.O.L)
  - Predicates: { hurt(*t*), hungry(*t*) }
  - Constants: { steve, sadie, brian }
  - hungry(steve), hurt(steve)
  - "Closed world assumption" (if not true, then false)
  - "classical sets" (either a member of set, or not)
- Fuzzy sets: "**degree of membership**" (DOM)
  - [0 to N] where N is completely in, 0 completely out
  - hungry(steve)[0.5], hurt(steve)[0.9]

Fuzzy Sets

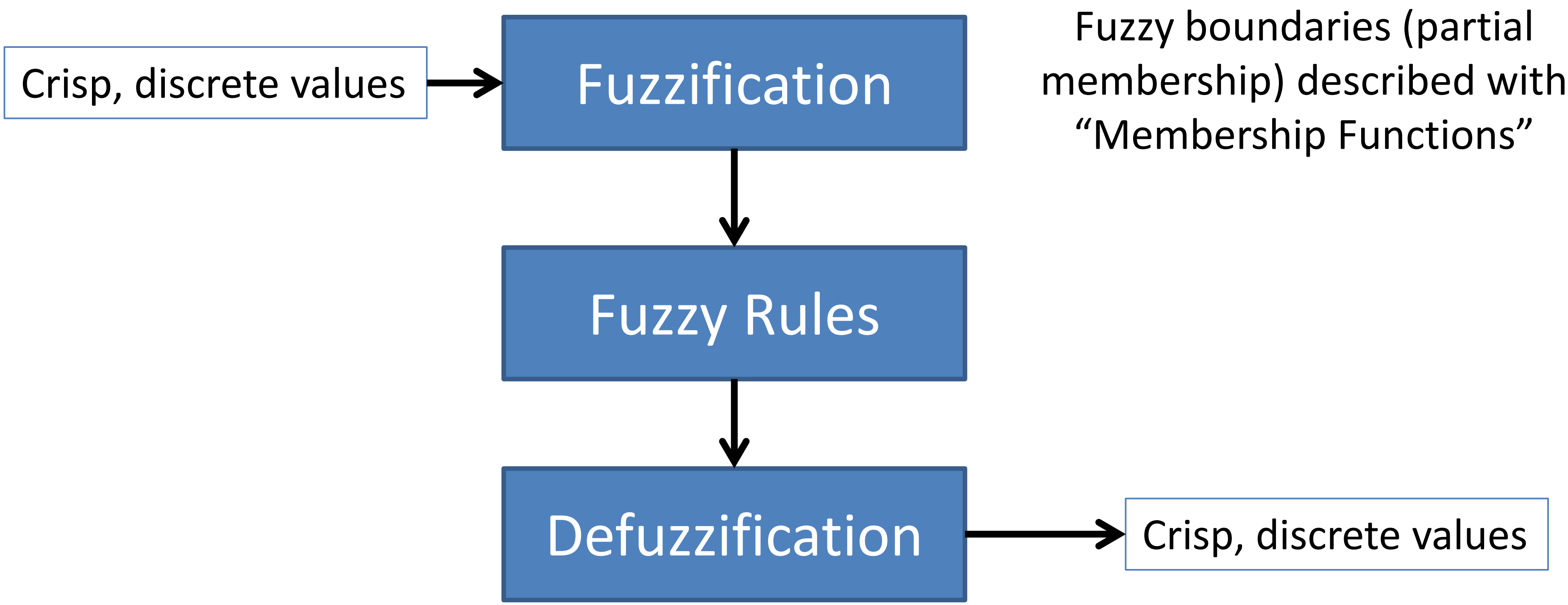
- DOM is usually [0 – 1] but…
  - Could use 0 – 255 (for int arithmetic speed)
  - Numbers are **NOT** probabilities **nor** percentages
- Fuzzy logic: truth degrees to model vagueness
- Probability theory: model non-determinism
- Mutual exclusion
  - Classical: some predicates are M.E. (hurt, healthy)
  - Fuzzy: can be a member of multiple sets
    - Can require DOM to sum to 1, but rare
    - Fuzzification to approximate; slightly off usually ok

Fuzzy Rule-based Inference



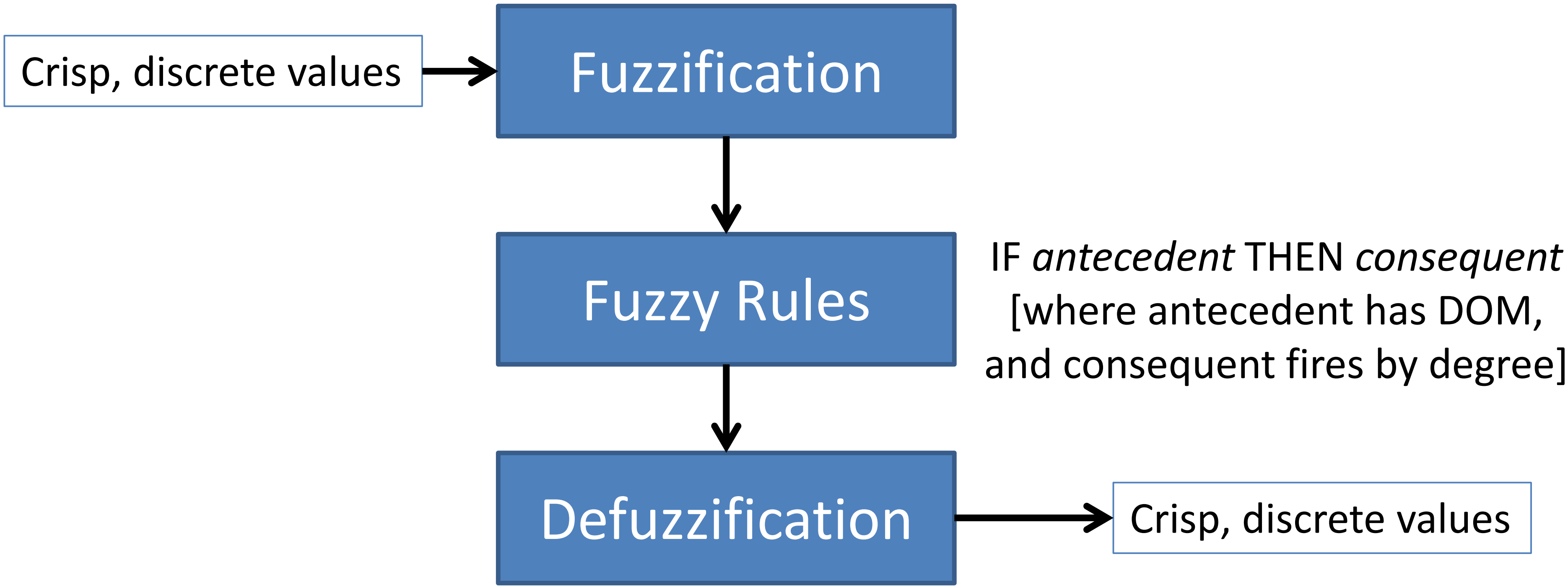
Buckland 10.1

Fuzzy Rule-based Inference



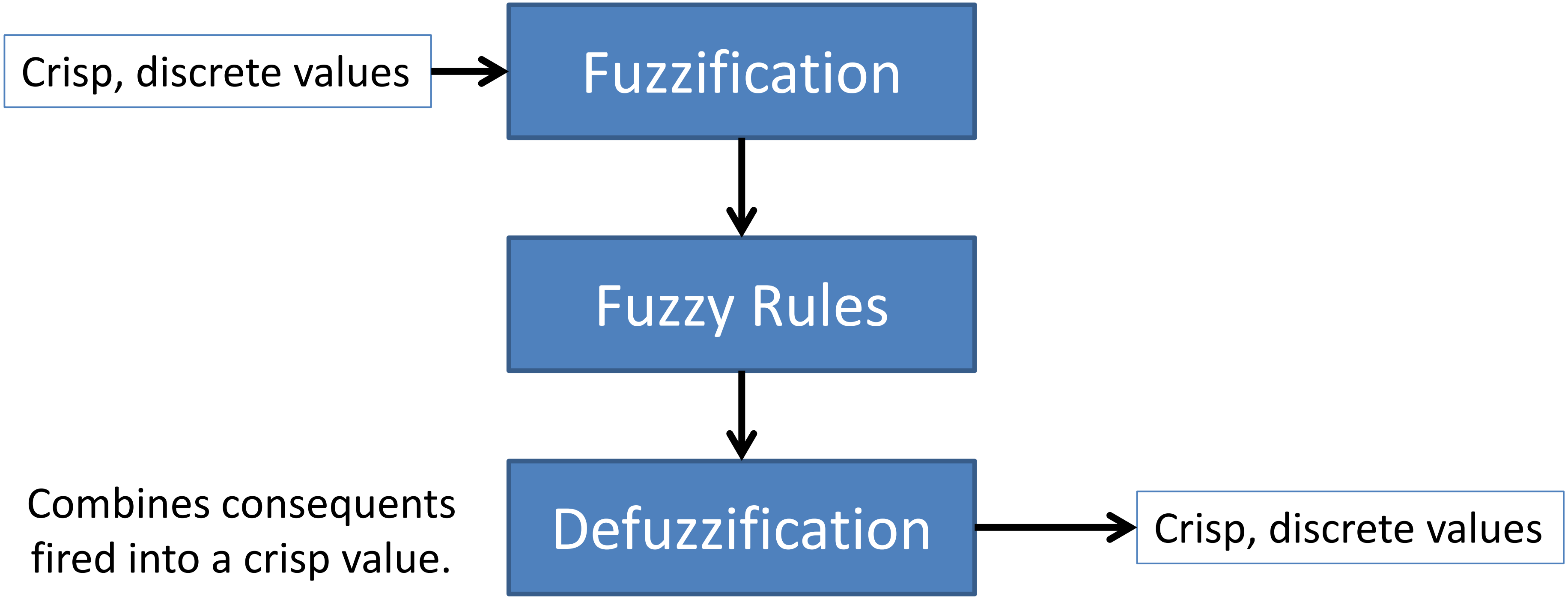
Buckland 10.1

Fuzzy Rule-based Inference



Buckland 10.1

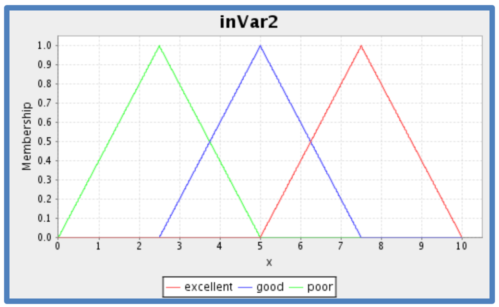
Fuzzy Rule-based Inference

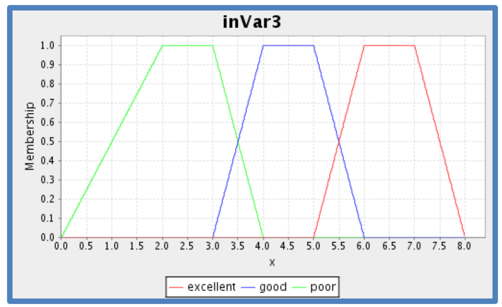


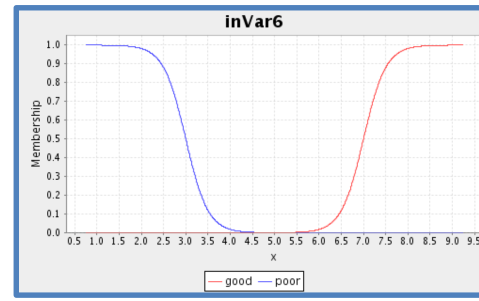
Buckland 10.1

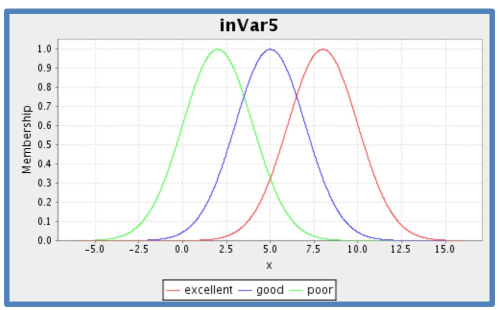
Fuzzy Inference

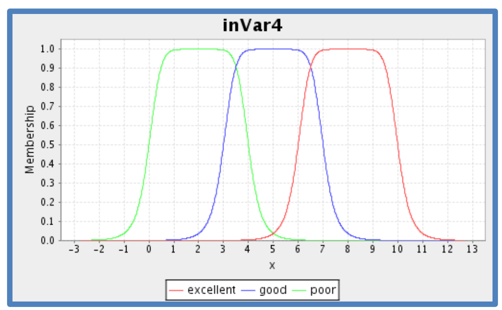
- For each rule,
  - For each antecedent, calculate the degree of membership of the input data.
  - Calculate the rule's inferred conclusion based upon the values in previous step
- Combine all the inferred conclusions into a single conclusion (a fuzzy set)
- For crisp values, the conclusion from 2 must be defuzzified

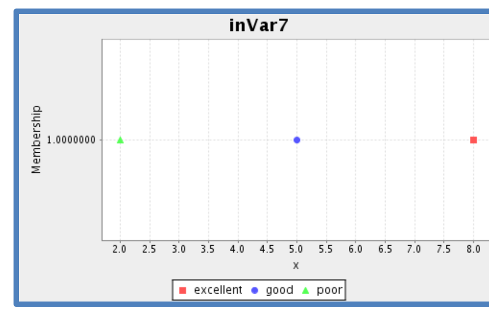

inVar2
Membership
x
excellent
good
poor


inVar3
Membership
x
excellent
good
poor


inVar6
Membership
x
good
poor


inVar5
Membership
x
excellent
good
poor

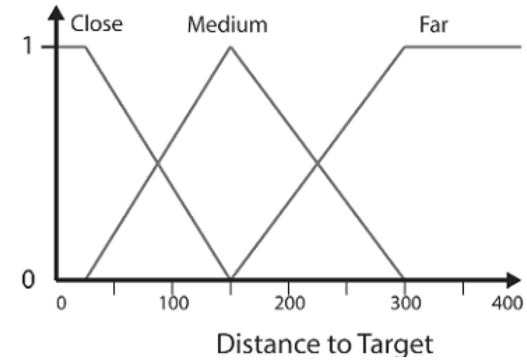

inVar4
Membership
x
excellent
good
poor

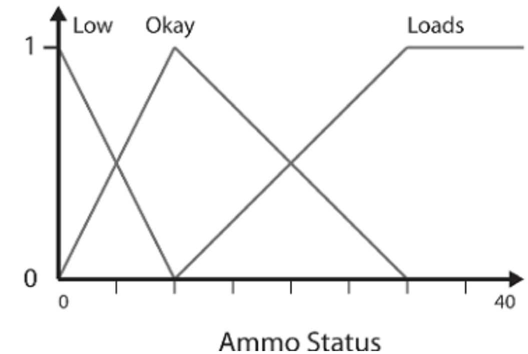

inVar7
Membership
1.0000000
x
excellent
good
poor

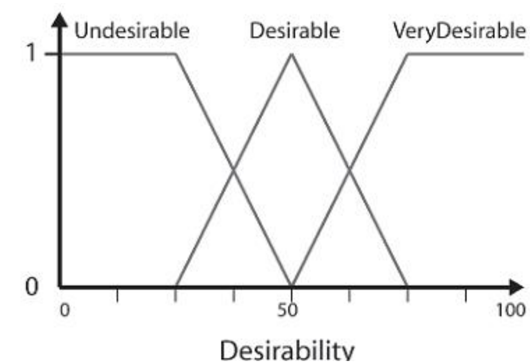
See Buckland CH10

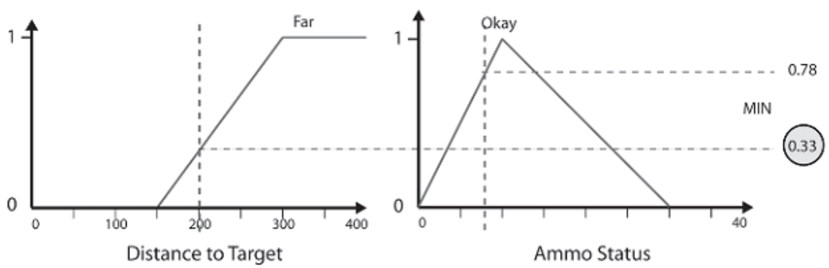
Example: Target 200px, Ammo=8

Rule 1. IF Target_Far AND Ammo_Loads THEN Desirable
Rule 2. IF Target_Far AND Ammo_Okay THEN Undesirable
Rule 3. IF Target_Far AND Ammo_Low THEN Undesirable
Rule 4. IF Target_Medium AND Ammo_Loads THEN VeryDesirable
Rule 5. IF Target_Medium AND Ammo_Okay THEN VeryDesirable
Rule 6. IF Target_Medium AND Ammo_Low THEN Desirable
Rule 7. IF Target_Close AND Ammo_Loads THEN Undesirable
Rule 8. IF Target_Close AND Ammo_Okay THEN Undesirable
Rule 9. IF Target_Close AND Ammo_Low THEN Undesirable





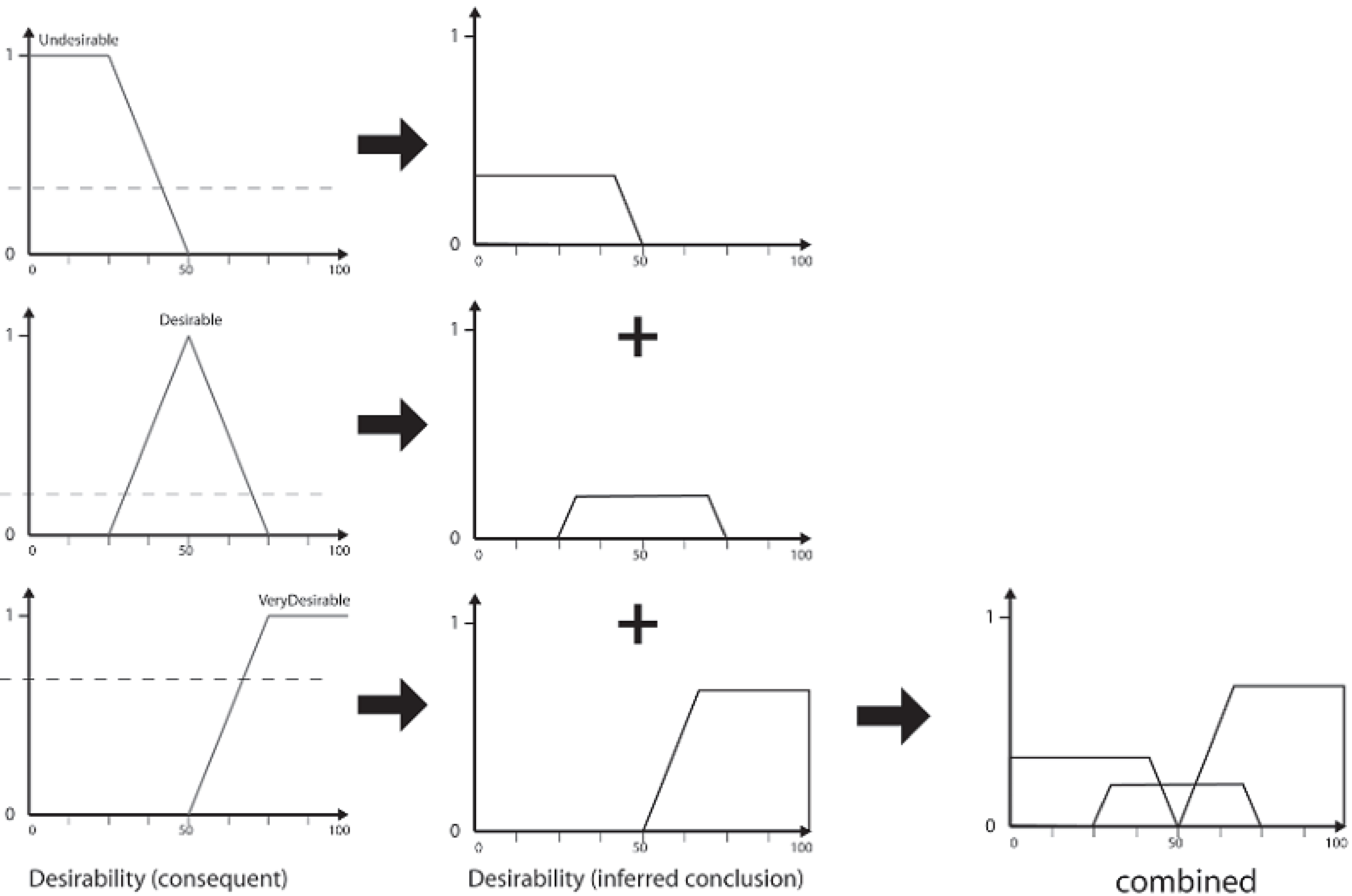




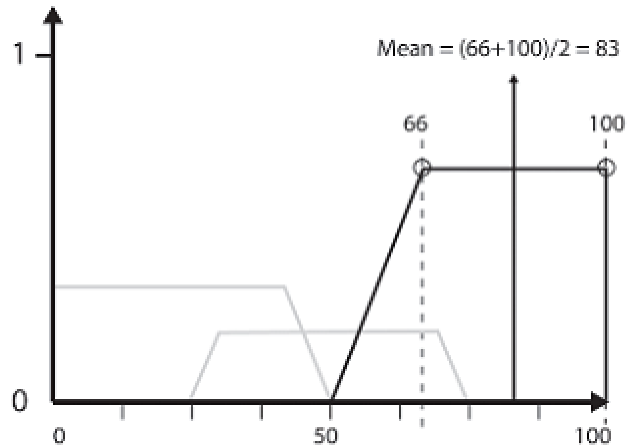
	Target_Close	Target_Medium	Target_Far
Ammo_Low	Undesirable 0	Desirable 0.2	Undesirable 0.2
Ammo_Okay	Undesirable 0	VeryDesirable 0.67	Undesirable 0.33
Ammo_Loads	Undesirable 0	VeryDesirable 0	Desirable 0

Consequent	Confidence
Undesirable	0.33
Desirable	0.2
Very Desirable	0.67

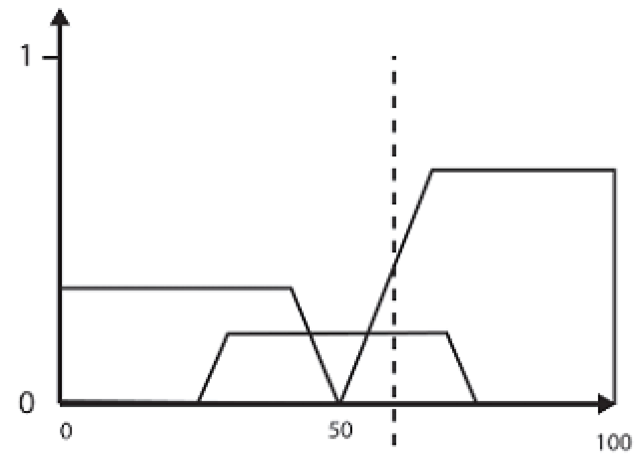
See Buckland CH10



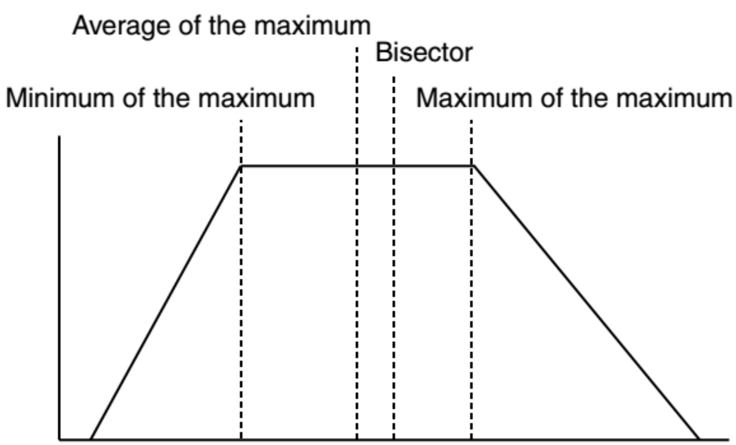
See Buckland CH10



Mean of Maximum



Centroid



Representative Value

Set	Representative Value	Confidence
Undesirable	12.5	0.33
Desirable	50	0.2
VeryDesirable	87.5	0.67

$$Desirability = \frac{12.5 \times 0.33 + 50 \times 0.2 + 87.5 \times 0.67}{0.33 + 0.2 + 0.67}$$

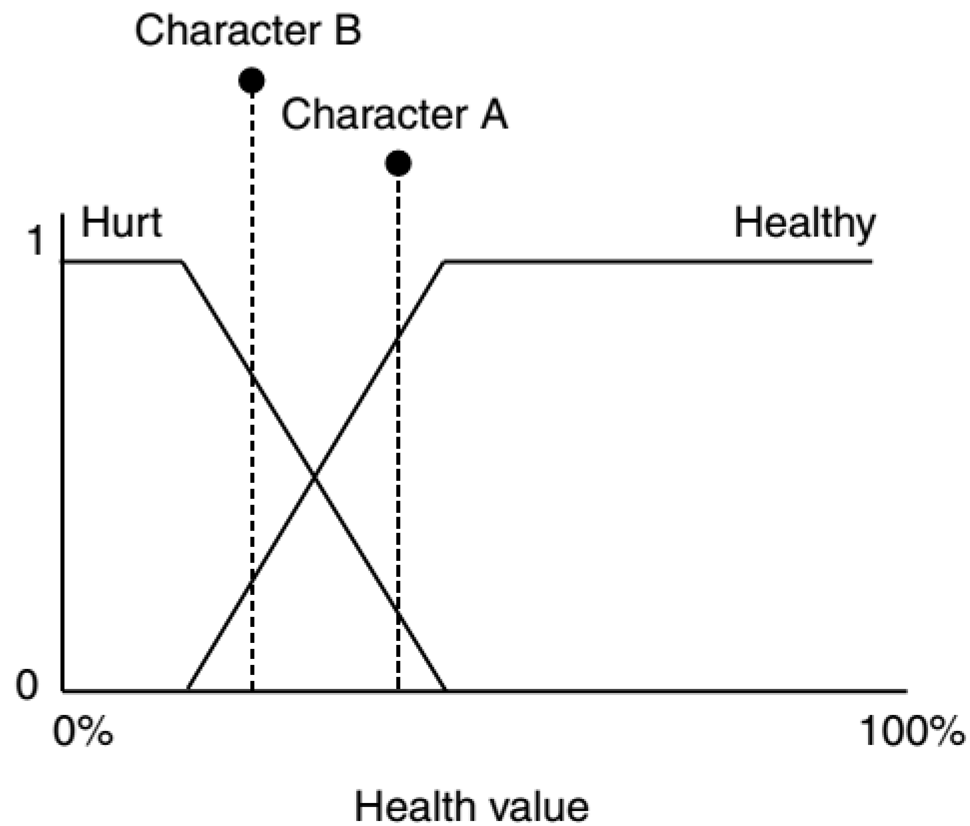
$$= \frac{72.75}{1.2}$$

$$Desirability = 60.625$$

“Fuzzification”

- Fuzzification:
  Game state → DOM
  “Membership Function”
  - Triangular
  - Trapezoidal
  - S-Curve
  - Left/Right Shoulder
  - Singleton
  - **Note**: Vert. lines should sum to 1
- De-fuzz:
  DOM → Game state

- Numeric Fuzzification:
  f(numeric) → DOM



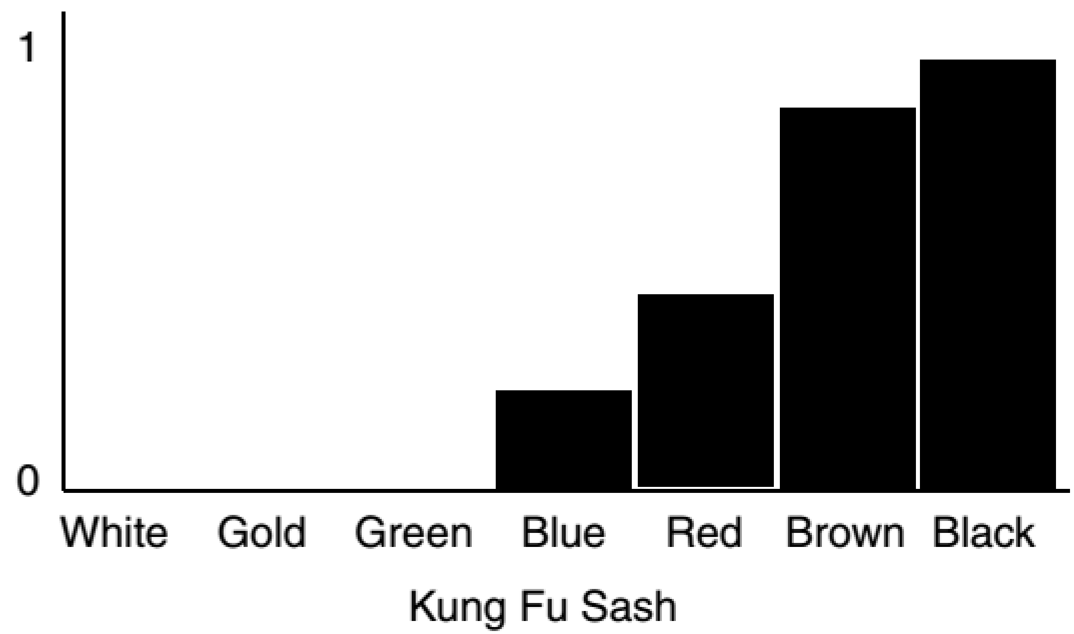
Millington 5.38

F(health val) → DOM(healthy)
F(health val) → DOM(hurt)

Fuzzification of Small Sets

Store pre-determined membership values

- Boolean var
  - hasPwrflArtifact
- Enum var
  - fearsmFighter from one of set of sashes



Millington 5.39

Set Operations (And, Or, Not)

- Boolean logic: True, False
- Fuzzy: DOM of a fuzzy set
  – Little rain (0.3) AND very cold (0.8)

A	B	A && B	A OR B
F	F	F	F
F	T	F	T
T	F	F	T
T	T	T	T

Fuzzy Logic

$m_{(A\ \&\&\ B)} = \min(m_A, m_B)$

$m_{(A\ OR\ B)} = \max(m_A, m_B)$

$m_{(NOT\ A)} = 1 - m_A$

Hedge: VERY = $(m_A)^2$

Hedge: FAIRLY = $(m_A)^{0.5}$

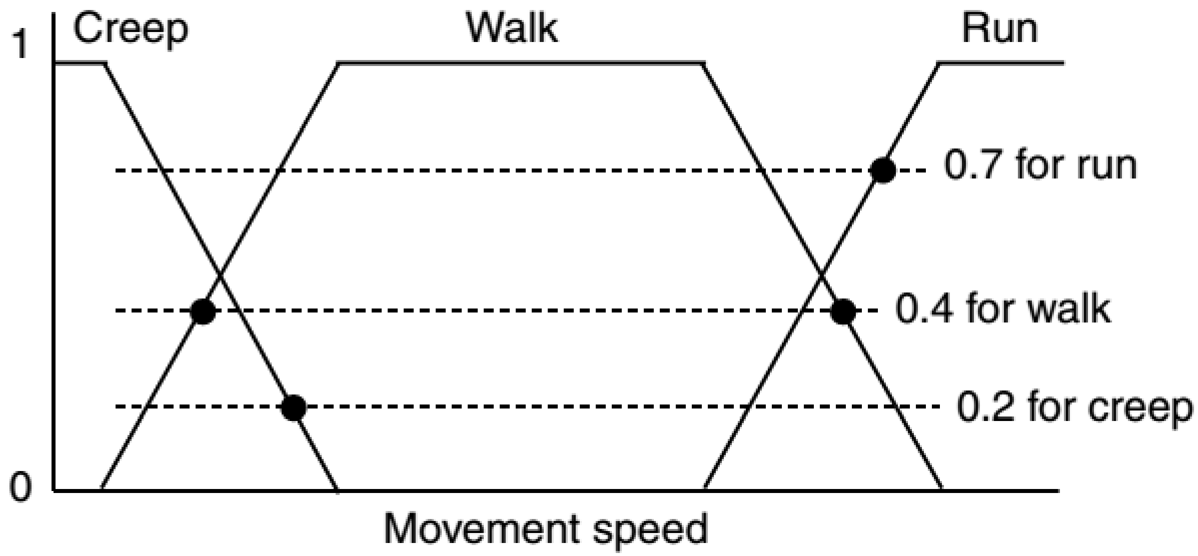
Fuzzy Rules

- Relate the known membership of certain fuzzy sets to generate new DOM values for other fuzzy sets
  - Vital: must create rule for each possible combination of antecedent sets
- e.g. “If I am close to the corner AND I am traveling fast, then I should brake”
- $m_{(\text{should brake})} = \min(m_{(\text{close to corner})}, m_{(\text{traveling quickly})})$
- Membership of should break with “close to corner” 0.6 and “traveling fast” 0.9?

Defuzzification

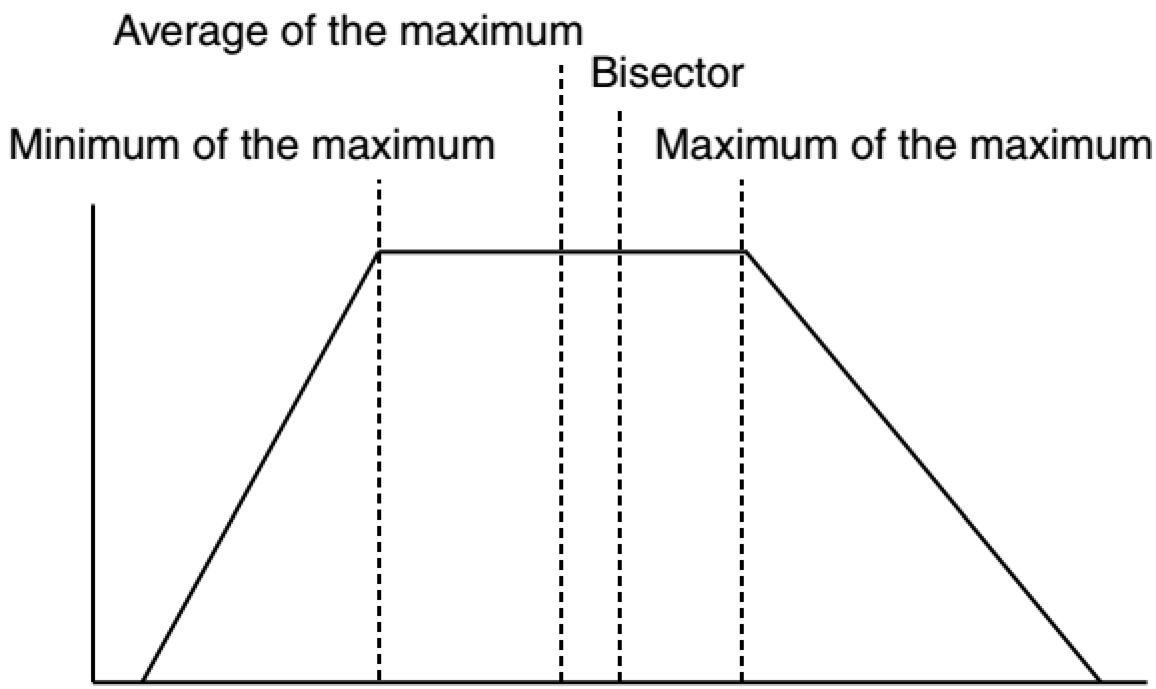
- Need to translate data back after applying whatever logic was needed
- Multiple approaches
  - Mean of maximum
  - Centroid
  - Average of Maxima
  - …
- Problem: Turn a set of membership values into a (typically) single number

Defuzzification



Millington 5.40

Highest Membership



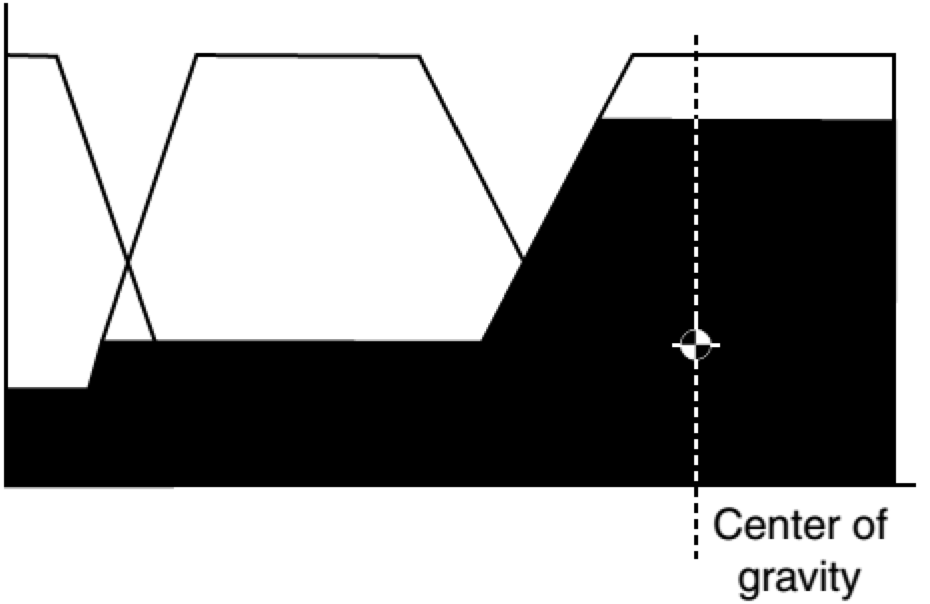
Millington 5.41

Blending Based on Membership

- Use DOM as weights
  - 0.33 creep, 0.33 walk, 0.34 run
  - 0.33 * characteristic creep speed + 0.33 * characteristic walk speed + 0.34 * characteristic run speed
  - Normalize values
- Can use minimum values (Smallest of Maximum method or Left of Maximum, LM)

Center of Gravity

- Crop membership function at DOM value
- Integrate each in turn to find center of gravity
- Method often used, but is expensive
- Blending works about as well and is cheap



Millington 5.40

Fuzzy in Decision Making

- Can use in any system we'd use boolean logic
- Determine FSM transitions
- Define rules for RBS

Fuzzy state machines

- Multiple interpretations
  - Any state machine with some element of fuzzy
- Example: crisp triggers, fuzzy states
  - Can be in any or all states with DOM
  - At each iteration, transitions belonging to all active states are given chance to trigger; fire transitions belonging to each state in decreasing DOM order
  - DOM of target is given by DOM of current state ANDed with degree of transition

Scalability

- Weakness of this approach: combinatorial explosion
  - rule for each possible combination of antecedent sets
  - 10 input variables and 5 states ---> approx. 10 million rules
- (William) Combs Method; Boeing 1997
  - IF target_far AND ammo_loads THEN Desirable
  - IF target_far THEN Desirable
    OR
    IF ammo_loads THEN Desirable
- **See Buckland CH 10**

Pros and Cons

- **Pro**
  - Easy to understand; supports explanation
  - Efficient way to represent **linguistic** and **subjective** attributes of the real world in computing.
  - Supports smooth transitions between behaviors
  - Generally easier to create versus a neural network
- **Cons**
  - Defining set membership functions can be difficult
  - Debugging knowledge can be difficult
  - De-fuzzify step can have surprising subtleties

Current Real-world Applications

- Industrial
  - Anti-sway control of cranes, climate control, positioning systems, coal power plant automated adaptations to coal quality, supervisory systems, humidity control, quality assurance, water purification, cement kiln controls
- Military Systems
  - Classification of DEMON spectra, automatic target tracking, airborne defense
- Appliances
  - Rice cooker, washing machine, climate control, vacuum, canon auto focus
- Automotive
  - Anti-lock braking system, traffic control, truck engine, transmissions (improving efficiency), governors, shift scheduling
- Aerospace
  - Altitude control of spacecraft & satellites, flow and mixture regulation deicing vehicles
- Trains
  - Monorail, high speed train Sendai
- Image Processing
  - Monitoring glaucoma, edge detection, image stabilization
- Video Games & FX

See Also

- http://videolectures.net/acai05_berthold_fl/
- Buckland 10, Millington 5.5
- Tools
  - Matlab
  - R (CRAN) packages (e.g. frbs)
  - jFuzzy logic
  - Fuzzy Control Language (FCL)
  - Octave & Fuzzy Logic Toolkit