Fuzzy Set Membership Functions

We will be concerned here with continuous fuzzy set membership functions, although discrete ones are used for cerain situations.

Definition: We take a *fuzzy set membership function* (FSMF) to be a unimodal (one hump shaped) continuous nonnegative function that monotonically decreases moving away from its maximum value of 1.

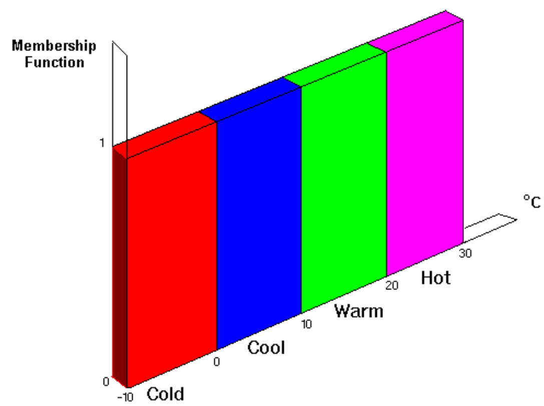


Fig. 1 : Bivalent Sets to Characterize the Temp. of a room.

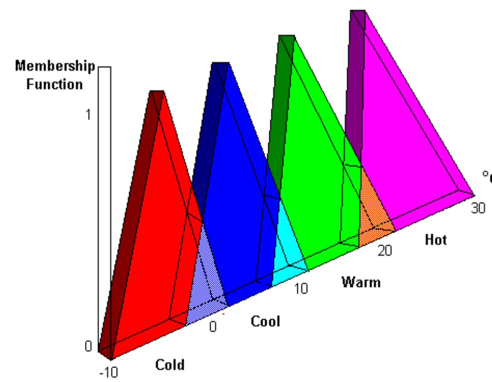


Fig. 2 - Fuzzy Sets to characterize the Temp. of a room.

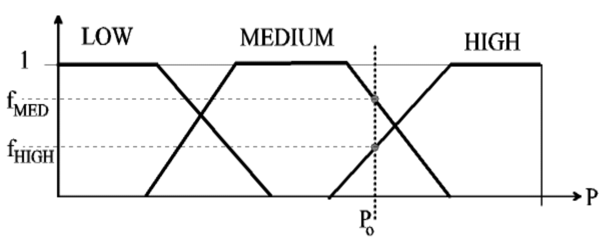


Fig. 3. Traditional FSMFs.

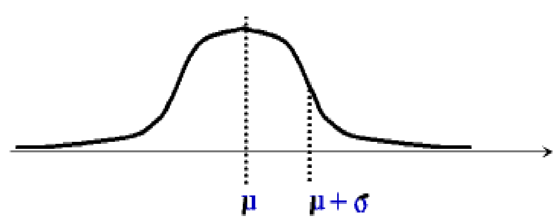


Fig. 4. Gausian FSMFs.

Traditionally, triangular shaped and trapezoidal shaped functions have been used. Nowadays, bell shaped functions are also used. Especially useful are Gaussians that take a center vector μ and a spread parameter σ. For vectors with a center vector **c**, and spread parameter σ, the format is

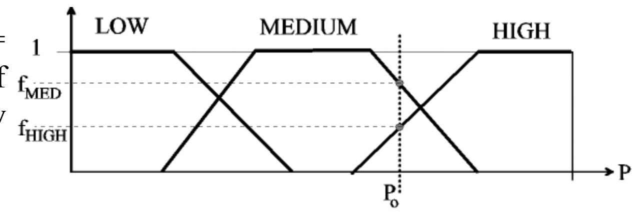
$$f(\mathbf{x}) = exp[-||\mathbf{x} - \mathbf{c}||^2/(2\sigma^2)]$$

Because the values of a FSMF are between *0* and *1*, we say they are truth values that the vector ***x*** belongs to a set represented by the function. The description of the set the function represents is a *linguistic variable*.

As an example, consider the linguistic variables: 1) the pressure is LOW; 2) the pressure is MEDIUM; and 3) the pressure is HIGH. For each such linguistic variable we use a FSMF to get the fuzzy truth that a value for variable P (pressure) is one of the FSMFs LOW, MEDUM, or HIGH.

In Fig. 2 the variable P (pressure) has 3 FSMFs defined on its range for a particular application. The FSMFs represent the linguistic conditions (P is LOW), (P is MEDIUM), and (P is HIGH).

Given a particular value of the pressure, $P = P_o$, we fuzzify P_o by passing it through each of the FSMFs for P. In this case we obtain the fuzzy values



f_{MED} and f_{HIGH}

Thus we have the fuzzification

(P is MED [f_{MED}]), (P is HIGH [f_{HIGH}])

Such fuzzified conditions are used in fuzzy rules. For example, if we have a set of rules, of which one is the rule

(P is MED [f_{MED}]) AND (P is HIGH [f_{HIGH}]) =>

(VALVE-ADJUSTMENT is $-\Delta$ [$f_{VALVE\text{-}ADJUST}$])

then this tells us that when we have the *antecedents (P is MED [f_{MED}])* and *(P is HIGH [f_{HIGH}])* being true, then the consequent is *(VALVE-ADJUSTMENT is $-\Delta$)*. The negative increment means that we should turn down the valve on the fuel supply to prevent the pressure from being too high. But the fuzzy truths provide a way to also adjust the increment $-\Delta$.

We need a fuzzy truth for the consequent so we can take the size of the valve turn increment to be a proportion of $-\Delta$. Thus we use a fuzzy weighting such as

$f_{VALVE\text{-}ADJUST1} = min\{f_{MED}, f_{HIGH}\}$, so the actual adjustment is $(f_{VALVE\text{-}ADJUST1})(-\Delta)$

If another rule (say with temperature T) also affects the valve adjustment, such as

(P is HIGH [f_{HIGH}]) AND ((T is HIGH [$f_{TEMP\text{-}HI}$]) =>

(VALVE-ADJUSTMENT is $-\epsilon$ [$f_{VALVE\text{-}ADJUST2}$])

then we have an ORing of the 2 rules with the same consequent. Then the truth of the consequent becomes

(VALVE-ADJUSTMENT is η) where η is the fuzzy weighted average

$$\eta = \{(f_{VALVE\text{-}ADJUST1})(-\Delta) + (f_{VALVE\text{-}ADJUST2})(-\epsilon)\} / \{(f_{VALVE\text{-}ADJUST1}) + (f_{VALVE\text{-}ADJUST2})\}$$

This is a *defuzzification* of the rule implications. The fuzzy truth is $max\{f_{VALVE\text{-}ADJUST1}, f_{VALVE\text{-}ADJUST2}\}$.

As an example of fuzzy rules, consider the rule

$$(A_1 \text{ is } LOW) \text{ AND } (A_2 \text{ is } HIGH) \Rightarrow (C \text{ is } LOW)$$

It performs the AND of the fuzzy truths of the antecedents for A_1 *and* A_2 and propagates the results to the consequent *(C is LOW)*. But if another rule, e.g.

$$(A_3 \text{ is } MED) \text{ AND } (A_4 \text{ is } LOW) \Rightarrow (C \text{ is } LOW)$$

also holds true, then *(C is LOW)* takes the maximum of the two fuzzy rule implications as its ORed fuzzy truth. Thus a min-max FNN is an AND-OR fuzzy rule-based system conceptualized in the neural network format of Figure 3.