Non-classical logics: theory and applications Part 1 Many-valued similarity

Esko Turunen

Tampere University of Technology, Finland

Introduction – some theoretical results

Classical logic can be studied by Boolean algebras. Similarly, for any generalized logic there is a corresponding algebra. For Lukasiewicz infinite valued logic this algebra is an *MV*-algebra. Essentintially, *a*Å*a* ¹ *a* in *MV*-algebras. We are interested in injective *MV*-algebras.

An element *b* of an *MV*-algebra *L* is called an *n-divisor* of an element *a* of *L*, if

$$\boxed{(a^* \oplus (n-1)b)^* = b \quad \text{and} \quad nb = a, \quad \text{where} \quad nb = \underbrace{b \oplus \dots \oplus b}_{n \text{ times}}.}$$

If all elements have *n*-divisors for all natural *n*, then *L* is called *divisible*.

An *MV*-algebra *L* is called *injective* if it is *complete* and *divisible*.

Injective MV-algebra are sufficient to construct fuzzy IF-THEN inrefence systems.

A canonical example of an injective MV-algebra is the *Lukasiewicz algebra* defined on the real unit interval [*0*,*1*] and endowed with the following operations:

*a*Å*b* = *min*{*a*+*b*,*1*}, aÄb = *max*{*a*+*b*-*1*, *0*},

*a** = *1* – *a*, *a*→*b* = *min*{*1*, *1* – *a* + *b*}.

Definition Let L be an injective MV-algebra and let A be a non-void set. A *fuzzy similarity* S on A is such a binary fuzzy relation that, for each a, b, and c in A,

(i) $S(a,a) = 1$ (everything is similar to itself),

(ii) $S(a,b) = S(b,a)$ (fuzzy similarity is symmetric),

(iii) $S(a,b)$ÄS$(b,c) \leq S(a,c)$ (fuzzy similarity is weakly transitive).

Recall an L-valued fuzzy subset X of A is an ordered couple (A,μ_X), where the membership function $\mu_X{:}A\rightarrow L$ tells the degree to which an element a in A belongs to the fuzzy subset X.

Given a fuzzy subset (A,μ_X), define a fuzzy relation S on A by $\boxed{S(x,y) = \mu_X(x)\leftrightarrow\mu_X(y).}$

This fuzzy relation is symmetric, reflexive and, transitive. Therefore

Any fuzzy set generates a fuzzy similarity

Proposition Consider n injective MV-algebra L-valued fuzzy similariteis S_i, $i = 1,\ldots,n$ on a set A. Then a fuzzy binary relation S on A defined by

$$\boxed{S(x,y) = \frac{S_1(x,y)}{n} \oplus \ldots \oplus \frac{S_n(x,y)}{n}}$$

is an L-valued fuzzy similarity on A. More generally, any *weighted mean* SIM is an L-valued fuzzy similarity, where

$$\boxed{SIM(x,y) = \frac{m_1 S_1(x,y)}{M} \oplus \ldots \oplus \frac{m_n S_n(x,y)}{M}, \qquad M = \sum_{i=1}^{n} m_i, \quad m_i \in \mathcal{N}.}$$

The idea of partial similarity is not new. Indeed, in 1834 *John Stuart Mill* defined:
If two objects A and B agree on k attributes and disagree on m attributes, then

$$\boxed{sim(A,B) = \frac{k}{k+m}}$$

can be taken to measure *the degree of similarity* or *partial identity* between A and B.

In fact, Mill defines an injective *MV*-algebra valued fuzzy similarity.

It is worth noting that, among all *BL*-algebras (in particular, among continuous *t*-norms) injective MV-algebras are the only structures where *'the average of similarities is a similarity*'.

Therefore the following consideration can be done only in such a structure.

An Algorithm to Construct Fuzzy IF-THEN Inference Systems

Let us now consider to a fuzzy rule based system

Rule 1: IF x_1 is in A_{11} and x_2 is in A_{12} and .... and x_m is in A_{1m} THEN y is in B_1
Rule 2: IF x_1 is in A_{21} and x_2 is in A_{22} and .... and x_m is in A_{2m} THEN y is in B_2
*
*
*
Rule n: IF x_1 is in A_{n1} and x_2 is in A_{n2} and .... and x_m is in A_{nm} THEN y is in B_n

Here all A_{ij}.s and B_j are fuzzy but can be crips actions, too. As usual, it is not necessary that the rule base is complete, some rule combinations can be missing without causing any difficulties. It is also possible that different IF-part causes equal THEN-part, but it is not possible that a fixed IF-part causes two different THEN-parts. We will not need any kind of defuzzification methods, everything is based on an experts knowledge and properties of injective MV-algebra valued similarity.

Step 1. Create the dynamics of the inference system, i.e. define the IF-THEN rules and give shapes to the corresponding fuzzy sets.

Step 2. If necassary, give weights to various IF-parts to emphasize their importance.

Step 3. List the rules with respect to the mutual importance

Step 4. For each THEN-part, give a criteria on how to distinguish outputs with equal degree of membership.

A general framework for the inference system is now ready.
Asssume then that we have an actual input Actual = $(X_1,\ldots,X_m)$. A corresponding output Y is counted in the following way.

(1) Consider each IF-part of each rule as a crisp case, that is $\mu_{A_{ij}}(x_j) = 1$ holds.

(2) Count the degree of similarity between Actual and the IF-part of Rule i, $i = 1,\ldots,n$.
Since $\mu_{A_{ij}}(X_j)\leftrightarrow\mu_{A_{ij}}(x_j) = \mu_{A_{ij}}(X_j)\leftrightarrow 1 = \mu_{A_{ij}}(X_j)$, we only need to calculate averages or weighted averages of membership degrees!

(3) Fire an output Y such that $\mu_{B_k}(Y)$ = Similarity(Actual, Rule k) corresponding to the greatest similarity degree between the input Actual and the IF-part of a Rule k. If such a maximal rule is not unique, then use the preference list given in Step (3), and if there are several such outputs Y, use a creteria given in Step (4).

Note that counting the actual output can be viewed as an instance of *Generalized Modus Ponens* in the sense of (injective MV-algebra valued) Lukasiewicz-Pavelka logic;

$$R_{GMP} = \frac{\alpha, \alpha \Rightarrow \beta}{\beta}, \frac{a, b}{a \otimes b}$$

where α corresponds to the IF-part of a Rule, β corresponds to the THEN-part of the Rule, a is the value Similarity(Actual, Rule k) and $b = 1$. This gives a many-valued logic based theoretical justification to fuzzy inference.

In the rest part of our talk deals with real world case studies where we have applied the above metodology and algorithm.

Example 1. Signalized isolated pedestrian crossing

Acrobat Document

As long as there are no pedestrians, vehicles have green signal. If a pedestrian pushes a button and no cars are approaching the pedestrian will have immediately green signal.

In case there are vehicles approaching and pedestrians waiting, then vehicles's green depends on the following factors:
* how long time have pedestrians been waiting for [a short/long/too long time]
* how many vehicles are approaching [few/some/many]
* what is the shortes cap between approaching vehicles [short/large]
The situation is updated after every half second.

Experienced traffic engineers described the above fuzzy set as follows

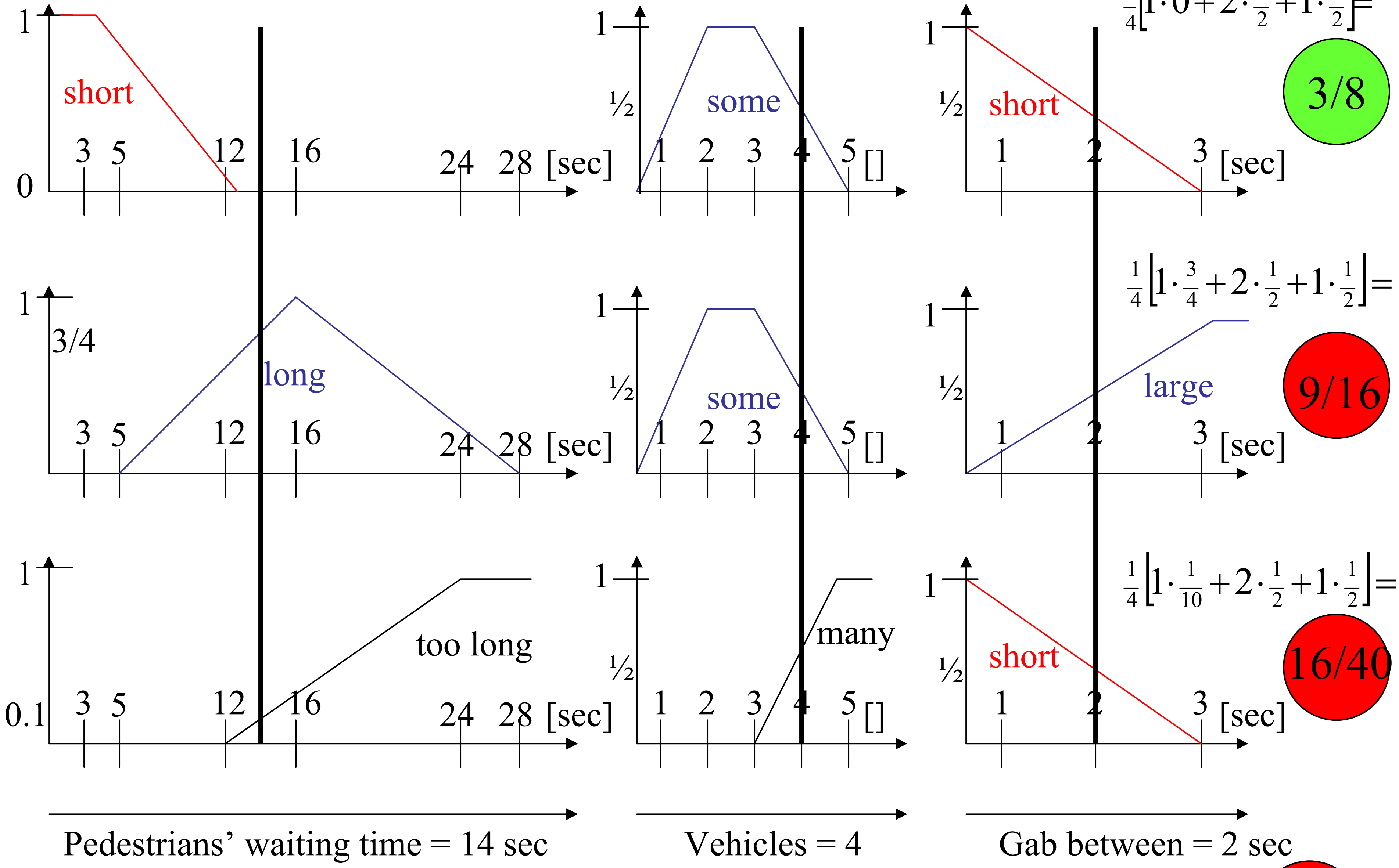
Acrobat Document

There are 18 rules in our fuzzy IF-THEN rule base (green extension is prefered)
This corresponds to all possible rules (3x3x2 = 18), therefore, the rule base is complete.

```
An example of  a rule (given by traffic engineers):
   IF (pedestrians' waiting time = short) [weight = 1]
       AND (approaching vehicles = few) [weight = 2]
          AND (gab between approaching vehicles = short) [weight = 1]
                                      THEN (extend vehicles green signal)
```

The output is always a crisp action (red or green). In a '50-50 situation' it is green.

An example (assuming we would have only three rules)



$$\tfrac{1}{4}\left[1\cdot 0+2\cdot\tfrac{1}{2}+1\cdot\tfrac{1}{2}\right]= 3/8$$

$$\tfrac{1}{4}\left[1\cdot\tfrac{3}{4}+2\cdot\tfrac{1}{2}+1\cdot\tfrac{1}{2}\right]= 9/16$$

$$\tfrac{1}{4}\left[1\cdot\tfrac{1}{10}+2\cdot\tfrac{1}{2}+1\cdot\tfrac{1}{2}\right]= 16/40$$

The actual traffic situation is most similar to the 2. rule, thus...

Example 2. Determining Athlete’s Aerobic and Anaerobic Thresholds

100 metres sprinter has to run a short distance very fact, therefore, he has to have much training in the *anaerobic* zone (where his pulse close to maximal value), while a long distance runner needs endurance, thus, he needs training in the *aerobic zone*.
It is important for an athlete to test his aerobic and anaerobic thresholds regularly, these tests can be done e.g. on a running track ergometer, see Acrobat Document

Aerobic and anaerobic thresholds are functions of blood lactate [mmol/l], ventilation; CO_2 [l/min] and O_2 uptake [%]. They, in turn, are functions of heartbeat [b/min].
A typical test starts with a 3 minutes worm up [pulse around 100 beat/min], then the load is increased every 2 - 3 minutes and blood lactate, ventilation, CO_2, O_2 uptake and heartbeat are measured. A test lasts until volitional exhaustion [pulse near 200 b/min], this takes usually 20-30 minutes [A typical test protocol, see Acrobat Document]

	Lactate	Ventilation	Pulse	Order
Aerobic	* lowest value ↑	*VE increasing	* max - 40 +/-	lactate, VE
threshold	* Δ=0.2 mmol/l	* O_2% highest	10 +/- 5 [b/min]	pulse
Anaerobic	* rapid increasing	* VE clearly ↑	* max - 20 +/-	lactate, VE
threshold	* 3 mmol/l	* VCO_2 ↑	5 +/- 3 [b/min]	VCO_2,
		*O_2% decreasing		O_2%
		* VE/VO2 ↑		VE/VO2

Clearly, all these consepts are fuzzy and can be expressed by fuzzy intervals. They are context dependent, too. For example 'Maximal pulse – 40' depends on a respective measurement. We used the following membership function

FuzzyMaxMinus40.mw

To mimic a skilled sport medicine spacialist's action when she/he is determing an aerobic threshold, we need only one rule (namely that one given above!)

Indeed, we count the degree of total fuzzy similarity between each measured value and that given be the rule.

Examples: open Matlab7

```
> cd C:\Matlab7\Mika\hemi
> aek
```

Samples 33, 78