ICIC Express Letters
Volume **10**, Number **8**, August **2016**

ICIC International ©2016 ISSN 1881-803X
pp. 1965–1970

LINGUISTIC MODUS PONENS AND LINGUISTIC MODUS TOLLENS WITH LATTICE ORDER

Yunxia Zhang[1], Degen Huang[1] and Li Zou[2,*]

[1]School of Computer Science and Technology
Dalian University of Technology
No. 2, Linggong Road, Ganjingzi District, Dalian 116024, P. R. China
z_yunxia@163.com; huangdg@dlut.edu.cn

[2]School of Computer and Information Technology
Liaoning Normal University
No. 1, Liushu South Street, Dalian 116081, P. R. China
*Corresponding author: zoulicn@163.com

Received December 2015; accepted March 2016

Abstract. *Fuzzy modus ponens (FMP) and the fuzzy modus tollens (FMT) are two important models of fuzzy inference. In this paper, we extend the FMP and FMT based on linguistic truth-valued lattice implication algebra aiming to solve the approximate reasoning problem with linguistic values. We propose the form of linguistic modus ponens (LMP) and linguistic modus tollens (LMT). The similarity-based algorithms are obtained to make the LMP and LMT work and some essential properties are discussed.*
Keywords: Linguistic modus ponens, Linguistic modus tollens, Approximate reasoning, Linguistic truth-valued lattice

1. **Introduction.** To make reasoning with the linguistic information which is always imprecise and imperfect, many approaches have been presented. Computing with words (CWW) methodology based on fuzzy theory proposed by Zadeh [1,2] has played a key role in recent years. The basic models of deductive processes with fuzzy sets are the fuzzy modus ponens (FMP) and the fuzzy modus tollens (FMT) [3]. The most popular fundamental patterns are the composition rule of inference (CRI) [4] and the Triple I method [5] in approximate reasoning problem with fuzziness. M. X. Luo and K. Zhang extended the Triple I method by proposing a new full implication algorithm based on interval-valued fuzzy inference [6]. B. K. Zhou et al. extended the CRI with quintuple implication principle [7].

The fuzzy theory processes linguistic information with a quantitative way with a membership function while linguistic information is always qualitative and both comparable and incomparable. Lattice implication algebra (LIA) proposed by Y. Xu et al. focused on the incomparability in linguistic terms besides fuzziness [8]. Lattice implication algebra structure can imitate the uncertain and both comparable and incomparable character of the linguistic values in natural language, and therefore attracts many researchers studying in varied direction, such as logic system [9], resolution automated [10], and approximate reasoning [11].

Despite the important role that MP and MT played in the approximate reasoning, there is no research on MP and MT with linguistic values. In the present work, we study the MP and MT based on the linguistic truth-valued lattice implication algebra, which is called linguistic modus ponens (LMP) and linguistic modus tollens (LMT). It is one of the essential contents in linguistic truth-valued lattice logic system, and provides a way to make reasoning with the linguistic values for people who always use natural language when they make decision or assessment.

The paper is organized as follows. In Section 2, we briefly review the linguistic valued lattice implication algebra and show the 3D mesh of implication operator. In Section 3, we propose the linguistic modus ponens (LMP) and linguistic modus tollens (LMT) based on linguistic truth-valued lattice implication algebra and discuss some properties of them. In Section 4, some conclusions are organized.

2. **Preliminaries.** Let $(L, \vee, \wedge, O, I)$ be a bounded lattice with universal boundaries O (the least element) and I (the greatest element) respectively, and " $'$ " be an order-reversing involution. For any $x, y, z \in L$, if mapping $\rightarrow: L \times L \rightarrow L$ satisfies:

$$\begin{aligned}
&(I1): x \rightarrow (y \rightarrow z) = y \rightarrow (x \rightarrow z);\\
&(I2): x \rightarrow x = I;\\
&(I3): x \rightarrow y = y' \rightarrow x';\\
&(I4): x \rightarrow y = y \rightarrow x = I \ implies \ x = y;\\
&(I5): (x \rightarrow y) \rightarrow y = (y \rightarrow x) \rightarrow x;\\
&(I6): (x \vee y) \rightarrow z = (x \rightarrow z) \wedge (y \rightarrow z);\\
&(I7): (x \wedge y) \rightarrow z = (x \rightarrow z) \vee (y \rightarrow z);
\end{aligned}$$

then $(L, \vee, \wedge, ', \rightarrow, O, I)$ is a lattice implication algebra (LIA for short).

In linguistic hedges set $H = \{h_i \mid i = 0, 1, 2, 3, 4\}$, where h_0 means "slightly", h_1 means "somewhat", h_2 means "exactly", h_3 means "very" and h_4 means "absolutely". Evaluating values set is $\{c_1, c_2\}$, where c_1 means "incredibility" and c_2 means "credibility". As for ten-element lattice implication algebra $(V, \vee, \wedge, \rightarrow)$, the operation "$\vee$" and "$\wedge$" are shown in Figure 1. And the operation " $'$ " is $(h_i, c_1)' = (h_i, c_2)$, $(h_i, c_2)' = (h_i, c_1)$.

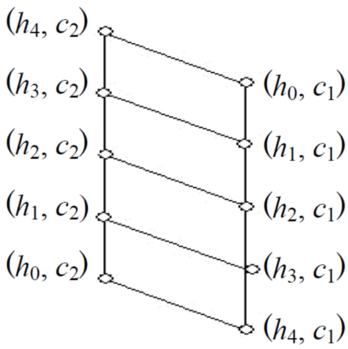


FIGURE 1. The Hasse diagram of $\mathcal{L}_5 \times \mathcal{L}_2$

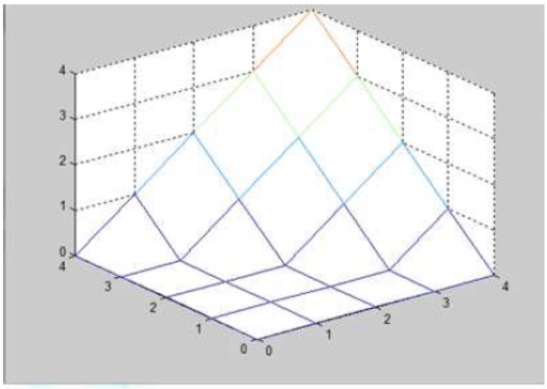
The operation "$\rightarrow$" is as the following (Figure 2).

$$\begin{cases}
(h_i, c_2) \rightarrow (h_j, c_1) = (h_{\max\{0, i+j-4\}}, c_1)\\
(h_i, c_1) \rightarrow (h_j, c_2) = (h_{\min\{4, i+j\}}, c_2)\\
(h_i, c_2) \rightarrow (h_j, c_2) = (h_{\min\{4, 4-i+j\}}, c_2)\\
(h_i, c_1) \rightarrow (h_j, c_1) = (h_{\min\{4, 4-j+i\}}, c_2).
\end{cases}$$

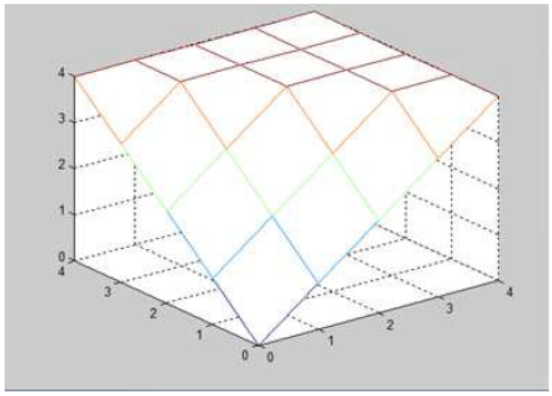
Then $(V, \vee, \wedge, \rightarrow)$ is ten linguistic-valued credibility factors LIA.

3. **Linguistic Modus Ponens (LMP) and Linguistic Modus tollens (LMT).** As one of the basic inference models of fuzzy reasoning, FMP and FMT have the following form:

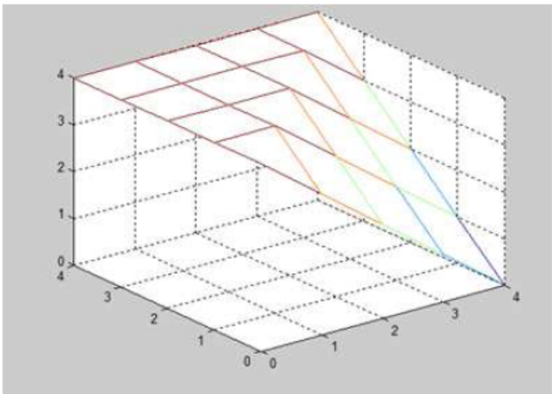
$$\frac{\begin{array}{l} A \rightarrow B \\ A^* \end{array}}{\qquad B^*}$$



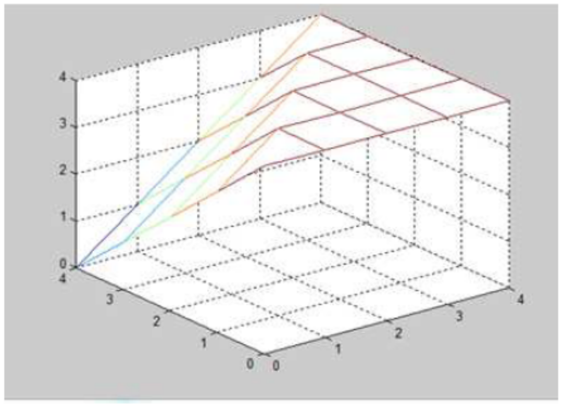
(a) $(h_i, c_2) \to (h_j, c_1)$



(b) $(h_i, c_1) \to (h_j, c_2)$



(c) $(h_i, c_2) \to (h_j, c_2)$



(d) $(h_i, c_1) \to (h_j, c_1)$

Figure 2. The 3D mesh of " $\to$ "

and

$$\frac{\begin{array}{l} A \to B \\ \qquad\ B^* \end{array}}{A^*}.$$

For any (h_i, c_a), (h_j, c_b), (h_k, c_d) and $(h_l, c_e) \in L_{V(n\times 2)}$, $i, j, k, l \in \{0, 1, 2, \dots, n\}$, $a, b, d, e \in \{1, 2\}$, the LMP form is similar as FMP, represented as

$$\frac{\begin{array}{l} (h_i, c_a) \to (h_j, c_b) \\ (h_k, c_d) \end{array}}{\qquad\qquad (h_l, c_e)}.$$

Now the (h_i, c_a), (h_j, c_b), and (h_k, c_d) are given, and then we obtain the similarity-based operator to get (h_l, c_e).

If $c_a = c_b = c_d$, then

$$l = (j - i + k) \wedge n \vee 0,\ c_e = c_b. \tag{1}$$

If $c_a = c_b \neq c_d$, then

$$l = (i - j + k) \wedge n \vee 0,\ c_e = c_d. \tag{2}$$

If $c_a = c_d \neq c_b$, then

$$l = (i + j - k) \wedge n \vee 0,\ c_e = c_b. \tag{3}$$

If $c_a \neq c_b = c_d$, then

$$l = (k + i + j + 1 - n) \wedge n \vee 0,\ c_e = c_d. \tag{4}$$

As to the LMT, it has the similar form with FMT.

$$\begin{array}{l} (h_i, c_a) \rightarrow (h_j, c_b) \\ \qquad\qquad\quad (h_l, c_e) \\ \hline (h_k, c_d) \end{array}$$

The operator to get (h_k, c_d) can be obtained according to LMP easily. For example, if $c_a = c_b = c_e$, then we obtain

$$k = (i - j + l) \wedge n \vee 0,\ c_d = c_a \tag{5}$$

from Equation (1).

In the same way, the equations are given in other cases.

If $c_a = c_b \neq c_e$, then

$$k = (j - i + l) \wedge n \vee 0,\ c_d = c_e \tag{6}$$

from Equation (2).

If $c_a \neq c_b = c_e$, then

$$k = (i + j - l) \wedge n \vee 0,\ c_d = c_a \tag{7}$$

from Equation (3).

If $c_a = c_e \neq c_b$, then

$$k = (l + i + j + 1 - n) \wedge n \vee 0,\ c_d = c_e \tag{8}$$

from Equation (4).

Theorem 3.1. *For any (h_i, c_a), (h_j, c_b), (h_k, c_d) and $(h_l, c_e) \in L_{V(n\times 2)}$, if $(h_k, c_d) = (h_i, c_a)$, then $(h_l, c_e) = (h_j, c_b)$, where (h_l, c_e) is got from LMP as below.*

$$\begin{array}{l} (h_i, c_a) \rightarrow (h_j, c_b) \\ (h_k, c_d) \\ \hline \qquad\qquad\quad (h_l, c_e) \end{array}.$$

Proof: If $(h_k, c_d) = (h_i, c_a)$, then $h_k = h_i$, $c_d = c_a$. Two cases satisfy $c_d = c_a$ when $c_a = c_b = c_d$ and $c_a = c_d \neq c_b$. We know $k = i$ from $h_k = h_i$, then when $c_a = c_b = c_d$,

$$l = (j - i + k) \wedge n \vee 0 = j,\ c_e = c_b.$$

When $c_a = c_d \neq c_b$,

$$l = (i + j - k) \wedge n \vee 0 = j,\ c_e = c_b.$$

Hence, $h_l = h_j$, $c_e = c_b$. That is to say, $(h_l, c_e) = (h_j, c_b)$.

Corollary 3.1. *For any (h_i, c_a), (h_j, c_b), (h_k, c_d) and $(h_l, c_e) \in L_{V(n\times 2)}$, if $(h_l, c_e) = (h_j, c_b)$, then $(h_k, c_d) = (h_i, c_a)$, where (h_k, c_d) is got from LMT as below.*

$$\begin{array}{l} (h_i, c_a) \rightarrow (h_j, c_b) \\ \qquad\qquad\quad (h_l, c_e) \\ \hline (h_k, c_d) \end{array}.$$

Proof: If $(h_l, c_e) = (h_j, c_b)$, then $h_l = h_j$, $c_e = c_b$. Two cases satisfy $c_e = c_b$ when $c_a = c_b = c_e$ and $c_a \neq c_b = c_e$. When $c_a = c_b = c_e$, we know $j = l$ from $h_l = h_j$, then

$$k = (i - j + l) \wedge n \vee 0 = i,\ c_d = c_a.$$

When $c_a \neq c_b = c_e$

$$k = (i + j - l) \wedge n \vee 0 = i,\ c_d = c_a.$$

Hence, $h_k = h_i$, $c_a = c_d$. That is to say, $(h_k, c_d) = (h_i, c_a)$.

Theorem 3.2. *For any* (h_i, c_a), (h_j, c_b), (h_k, c_d), $(h_{k'}, c_{d'})$, (h_l, c_e) *and* $(h_{l'}, c_{e'}) \in L_{V(n\times 2)}$, *it is known that*

$$\frac{\begin{array}{l}(h_i, c_a) \to (h_j, c_b)\\ (h_k, c_d)\end{array}}{\qquad\qquad (h_l, c_e)},$$

$$\frac{\begin{array}{l}(h_i, c_a) \to (h_j, c_b)\\ (h_{k'}, c_{d'})\end{array}}{\qquad\qquad (h_{l'}, c_{e'})}.$$

Suppose (h_i, c_a), (h_j, c_b) are given, if $(h_k, c_d) < (h_{k'}, c_{d'})$, then $(h_l, c_e) \le (h_{l'}, c_{e'})$.

Proof: Here we discuss the case that $c_a = c_b = 2$.

When $c_d = 2$, then $c_{d'} = 2$, $k' > k$ for $(h_k, c_d) < (h_{k'}, c_{d'})$.

$$l = (j - i + k) \wedge n \vee 0,\ c_e = c_b = 2;$$

$$l' = (j - i + k') \wedge n \vee 0,\ c_{e'} = c_b = 2.$$

Obviously, $l \le l'$, $c_e = c_{e'}$. So $(h_l, c_e) \le (h_{l'}, c_{e'})$.

When $c_d = 1$, then $c_{d'} = 1$, $k' < k$ or $c_{d'} = 2$, $k' \ge n - k$. We prove it in the two cases, respectively.

For the case that $c_{d'} = 1$, $k' < k$,

$$l = (i - j + k) \wedge n \vee 0,\ c_e = c_d = 1;$$

$$l' = (i - j + k') \wedge n \vee 0,\ c_{e'} = c_{d'} = 1.$$

Obviously, $l \ge l'$, $c_e = c_{e'}$. So $(h_l, c_e) \le (h_{l'}, c_{e'})$.

For the case that $c_{d'} = 2$, $k' \ge n - k$,

$$\begin{aligned} l' &= (j - i + k') \wedge n \vee 0,\\ &\ge (j - i + n - k) \wedge n \vee 0,\\ &= (n - (i - j + k)) \wedge n \vee 0,\\ &\ge (n - l).\\ c_{e'} &= c_b = 2. \end{aligned}$$

Obviously, $l \ge n - l'$. So $(h_l, c_e) \le (h_{l'}, c_{e'})$.

Other cases can be proved in the same way.

Corollary 3.2. *For any* (h_i, c_a), (h_j, c_b), (h_k, c_d), $(h_{k'}, c_{d'})$, (h_l, c_e) *and* $(h_{l'}, c_{e'}) \in L_{V(n\times 2)}$, *it is known that*

$$\frac{\begin{array}{r}(h_i, c_a) \to (h_j, c_b)\\ (h_l, c_e)\end{array}}{(h_k, c_d)\qquad\qquad},$$

$$\frac{\begin{array}{r}(h_i, c_a) \to (h_j, c_b)\\ (h_{l'}, c_{e'})\end{array}}{(h_{k'}, c_{d'})\qquad\qquad}.$$

Suppose (h_i, c_a), (h_j, c_b) are given, if $(h_l, c_e) < (h_{l'}, c_{e'})$, then $(h_k, c_d) \le (h_{k'}, c_{d'})$.

It can be proved in the same way with Theorem 3.2.

4. **Conclusions.** We extend the FMP and FMT to linguistic MP (LMP) and linguistic MT (LMT) based on linguistic truth-valued lattice implication algebra. It processes the linguistic information qualitatively and provides a basis method to linguistic approximate reasoning by LMP and LMT. We will study the Triple I method in linguistic approximate reasoning problem and discuss the performances of different reasoning methods in the future work.

Acknowledgment. This work is partially supported by National Natural Science Foundation of China (Nos. 61372187, 61175055, and 61173100), and Natural Science Foundation of Liaoning Province (No. 2015020059).

REFERENCES

[1] L. A. Zadeh, Fuzzy logic = computing with words, *IEEE Trans. Fuzzy Systems*, vol.4, no.2, pp.103-111, 1996.

[2] L. A. Zadeh, The concept of linguistic variable and application to approximate reasoning, Part I, Part II, Part III, *Information Science*, vol.8, pp.199-249, pp.301-357, vol.9, pp.43-80, 1975.

[3] L. A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans. Systems, Man and Cybernetics*, vol.3, no.1, pp.28-44, 1973.

[4] C. C. Lee, Fuzzy logic in control systems: Fuzzy logic controller, *IEEE Trans. Systems, Man and Cybernetics*, vol.20, no.2, pp.405-435, 1990.

[5] G. J. Wang, Formalized theory of general fuzzy reasoning, *Information Sciences*, vol.160, pp.251-266, 2004.

[6] M. X. Luo and K. Zhang, Robustness of full implication algorithms based on interval-valued fuzzy inference, *International Journal of Approximate Reasoning*, vol.62, pp.61-72, 2015.

[7] B. K. Zhou, G. Q. Xu and S. J. Li, The quintuple implication principle of fuzzy reasoning, *Information Sciences*, vol.297, pp.202-215, 2015.

[8] Y. Xu, S. Chen and J. Ma, Linguistic truth-valued lattice implication algebra and its properties, *Proc. of CESA'06*, pp.1123-1129, 2006.

[9] X. X. He, Y. Xu, J. Liu et al. A unified algorithm for finding k-IESFs in linguistic truth-valued lattice-valued propositional logic, *Soft Computing*, vol.18, no.11, pp.2135-2147, 2014.

[10] X. M. Zhong and Y. Xu, Alpha-group quasi-lock semantic resolution method based on lattice-valued propositional logic LP(X), *Journal of Multiple-Valued Logic and Soft Computing*, vol.22, nos.4-6, pp.581-598, 2014.

[11] L. Zou, Y. X. Zhang and X. Liu, Linguistic-valued approximate reasoning with lattice ordered linguistic-valued credibility, *International Journal of Computational Intelligence Systems*, vol.8, no.1, pp.53-61, 2015.