Information Sciences 283 (2014) 229–240

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

# Implications in bounded systems

J. Dombi [a,*], O. Csiszár [b]

[a] University of Szeged, Institute of Informatics, Hungary
[b] Óbuda University, Institute of Applied Mathematics, Hungary

ARTICLE INFO

*Article history:*
Received 7 February 2014
Received in revised form 17 June 2014
Accepted 25 June 2014
Available online 7 July 2014

*Keywords:*
Implication
S-implication
R-implication
Nilpotent operator
Łukasiewicz operator system
Bounded system

ABSTRACT

A consistent connective system generated by nilpotent operators is not necessarily isomorphic to Łukasiewicz-system. Using more than one generator function, consistent nilpotent connective systems (so-called bounded systems) can be obtained with the advantage of three naturally derived negations and thresholds. In this paper, implications in bounded systems are examined. Both R- and S-implications with respect to the three naturally derived negations of the bounded system are considered. It is shown that these implications never coincide in a bounded system, as the condition of coincidence is equivalent to the coincidence of the negations, which would lead to Łukasiewicz logic. The formulae and the basic properties of four different types of implications are given, two of which fulfill all the basic properties generally required for implications.



## 1. Introduction

In our previous article [7], we showed that a consistent connective system generated by nilpotent operators is not necessarily isomorphic to Łukasiewicz-system. Using more than one generator function, consistent nilpotent connective systems can be obtained in a significantly different way with three naturally derived negations. As the class of non-strict t-norms has preferable properties that make them useful in constructing logical structures, the advantages of such systems are obvious [14]. Due to the fact that all continuous Archimedean (i.e. representable) nilpotent t-norms are isomorphic to the Łukasiewicz t-norm [11], the previously studied nilpotent systems were all isomorphic to the well-known Łukasiewicz-logic. Those consistent connective systems which are not isomorphic to Łukasiewicz logic are called bounded systems [7].

Fuzzy implications are definitely among the most important operations in fuzzy logic [2,17]. Firstly, other basic logical connectives of the binary logic can be obtained from the classical implication. Secondly, the implication operator plays a crucial role in the inference mechanisms of any logic, like modus ponens, modus tollens, hypothetical syllogism in classical logic. Fuzzy implications all generalize the classical implication with the two possible crisp values from $0, 1$, to the fuzzy concept with truth values from the unit interval $[0, 1]$ [26]. In classical logic the implication can be defined in several ways. The most well-known implications are the usual material implication from the Kleene algebra, the implication obtained as the residuum of the conjunction in Heyting algebra (also called pseudo-Boolean algebra) in the intuitionistic logic framework and the implication in the setting of quantum logic. While all these differently defined implications have identical truth tables in the classical case, the natural generalizations of the above definitions in the fuzzy logic framework are not identical. This fact has led to some throughout research on fuzzy implications [1,3–5,12,18,20,21,24,25].

* Corresponding author.
*E-mail addresses:* dombi@inf.u-szeged.hu (J. Dombi), csiszar.orsolya@nik.uni-obuda.hu (O. Csiszár).

http://dx.doi.org/10.1016/j.ins.2014.06.031
0020-0255/

Based on the results of [7], now we focus on residual and S-implication operators [3] in bounded systems. The paper is organized as follows. After some preliminaries in Section 2, we examine the residual implication in Section 3 and S-implications with special attention to the ordering property in Section 4. In Section 6 we show that in a bounded system, the minimum and maximum operators can also be expressed in terms of the conjunction, the implication and the negation. Finally in Section 5 we show that in a bounded system the implications examined in this paper can never coincide. The formulae and the properties of implications are summarized in Section 7.

## 2. Preliminaries

### 2.1. t-Norms and conorms

Now we state the basic notations and results for t-norms and t-conorms [13]. A triangular norm (*t-norm* for short) $T$ is a binary operation on the closed unit interval [0,1] such that $([0,1],T)$ is an abelian semigroup with neutral element 1 that is totally ordered; i.e., for all $x_1,x_2,y_1,y_2 \in [0,1]$ with $x_1 \leqslant x_2$ and $y_1 \leqslant y_2$ we have $T(x_1,y_1) \leqslant T(x_2,y_2)$, where $\leqslant$ is the natural order on [0,1].

A triangular conorm (*t-conorm* for short) $S$ is a binary operation on the closed unit interval [0,1] such that $([0,1],S)$ is an abelian semigroup with a neutral element 0 that is totally ordered.

A continuous t-norm $T$ is said to be *Archimedean* if $T(x,x) < x$ holds for all $x \in (0,1)$. A continuous Archimedean $T$ is called *strict* if $T$ is strictly monotone; i.e. $T(x,y) < T(x,z)$ whenever $x \in (0,1]$ and $y < z$ , and *nilpotent* if there exist $x,y \in (0,1)$ such that $T(x,y) = 0$.

From the duality between t-norms and t-conorms, we can easily derive the following properties. A continuous t-conorm $S$ is said to be *Archimedean* if $S(x,x) > x$ holds for every $x,y \in (0,1)$. A continuous Archimedean $S$ is called *strict* if $S$ is strictly monotone; i.e. $S(x,y) < S(x,z)$ whenever $x \in [0,1)$ and $y < z$, and *nilpotent* if there exist $x,y \in (0,1)$ such that $S(x,y) = 1$.

The following well-known results provide important single variable representations for t-norms and t-conorms.

**Proposition 1** (*[15,18]*). *A function $T : [0,1]^2 \to [0,1]$ is a continuous Archimedean t-norm iff it has a continuous additive generator; i.e. there exists a continuous strictly decreasing function $t : [0,1] \to [0,\infty]$ with $t(1) = 0$, which is uniquely determined up to a positive multiplicative constant, such that*

$$T(x,y) = t^{-1}(\min(t(x)+t(y),t(0))), \quad x,y \in [0,1]. \tag{1}$$

**Proposition 2** (*[15,18]*). *A function $S : [0,1]^2 \to [0,1]$ is a continuous Archimedean t-conorm iff it has a continuous additive generator; i.e. there exists a continuous strictly increasing function $s : [0,1] \to [0,\infty]$ with $s(0) = 0$, which is uniquely determined up to a positive multiplicative constant, such that*

$$S(x,y) = s^{-1}(\min(s(x)+s(y),s(1))), \quad x,y \in [0,1]. \tag{2}$$

**Proposition 3.** [11]

*A t-norm $T$ is strict if and only if $t(0) = \infty$ holds for each continuous additive generator $t$ of $T$.*
*A t-norm $T$ is nilpotent if and only if $t(0) < \infty$ holds for each continuous additive generator $t$ of $T$.*
*A t-conorm $S$ is strict if and only if $s(1) = \infty$ holds for each continuous additive generator $s$ of $S$.*
*A t-conorm $S$ is nilpotent if and only if $s(1) < \infty$ holds for each continuous additive generator $s$ of $S$.*

**Proposition 4** [11]. *Let $T$ be a continuous Archimedean t-norm.*

*If $T$ is strict, then it is isomorphic to the product t-norm $T_{\mathbf{P}}$ , i.e., there exists an automorphism $\phi$ of the unit interval such that $T_\phi = \phi^{-1}(T(\phi(x),\phi(y))) = T_{\mathbf{P}}$.*

*If $T$ is nilpotent, then it is isomorphic to the Łukasiewicz t-norm $T_{\mathbf{L}}$, i.e., there exists an automorphism of the unit interval $\phi$ such that $T_\phi = \phi^{-1}(T(\phi(x),\phi(y))) = T_{\mathbf{L}}$.*

From the definitions of t-norms and t-conorms it follows immediately that t-norms are conjunctive (i.e. $T(x,y) \leqslant \min(x,y)$), while t-conorms are disjunctive (i.e. $S(x,y) \geqslant \max(x,y)$) aggregation functions. This is why they are widely used as conjunctions and disjunctions in multivalued logical structures.

The use of the so-called cutting function makes the formulae simpler.

**Definition 1** (*[7,19]*). Let us define the cutting operation [ ] by

$$[x] = \begin{cases} 0 & \textit{if} \quad x < 0 \\ x & \textit{if} \quad 0 \leqslant x \leqslant 1 \\ 1 & \textit{if} \quad 1 < x \end{cases}$$

and let the notation [ ] also act as 'brackets' when writing the argument of an operator, so that we can write $f[x]$ instead of $f([x])$.