A MATHEMATICAL SETTING FOR FUZZY LOGICS

MAI GEHRKE, CAROL WALKER and ELBERT WALKER
New Mexico State University, Las Cruces, NM 88003

Received 31 March 1997

The setup of a mathematical propositional logic is given in algebraic terms, describing exactly when two choices of truth value algebras give the same logic. The propositional logic obtained when the algebra of truth values is the real numbers in the unit interval equipped with minimum, maximum and $\neg x = 1 - x$ for conjunction, disjunction and negation, respectively, is the standard propositional fuzzy logic. This is shown to be the same as three-valued logic. The propositional logic obtained when the algebra of truth values is the set $\{(a, b) \mid a \leq b \text{ and } a, b \in [0, 1]\}$ of subintervals of the unit interval with component-wise operations, is propositional interval-valued fuzzy logic. This is shown to be the same as the logic given by a certain four element lattice of truth values. Since both of these logics are equivalent to ones given by finite algebras, it follows that there are finite algorithms for determining when two statements are logically equivalent within either of these logics. On this topic, normal forms are discussed for both of these logics.

Keywords: fuzzy logic, propositional logic, normal form, de Morgan algebra, Kleene algebra

1 Introduction

In this paper we spell out in detail the setup of a mathematical propositional logic in algebraic terms, describing exactly when two choices of truth value algebras give the same logic. This analysis applies to the plethora of choices used in fuzzy systems. In particular we consider the propositional logic obtained when the algebra of truth values is $\mathbb{I}$, the real numbers in the unit interval equipped with minimum, maximum and $\neg x = 1 - x$ for conjunction, disjunction and negation, respectively. In this case we have propositional fuzzy logic, and we show that this is the same as three-valued logic. We also consider the case where the algebra of truth values is $\mathbb{I}^{[2]}$, the set $\{(a, b) \mid a \leq b$ and $a, b \in [0, 1]\}$ with component-wise operations. This is what is sometimes called interval-valued fuzzy logic. We show that the logic obtained with the algebra of truth values $\mathbb{I}^{[2]}$ is the same as the one given by a certain four element lattice of truth values. Since both of these logics are equivalent to ones given by finite algebras, it follows that there are finite algorithms for determining when two statements are logically equivalent within either of these logics. On this topic, we discuss normal forms for both of these logics. To illustrate the use of these methods in the more general setting of t-norms, we show that the conjunctive logic obtained by using the unit interval with a strict Archimedean t-norm as the truth value algebra, is the same as the logic obtained by using the unit interval with a nilpotent Archimedean t-norm.

The general setup as well as the specific algebraic results exposed and applied in this paper are well-known in universal algebra and logic, but it seems that many of the points that become transparent with this viewpoint are not well-known in fuzzy logic. For this reason we feel that it is worthwhile to spell it out in detail. In 1993, Elkan drew attention to the propositional logic of fuzzy logic by seemingly stating that it is equivalent to Boolean logic. In the wake of that paper Nguyen, Kosheleva, and Kreinovich [11] wrote a paper presenting a normal form and thus a finite algorithm for determining equivalence in interval-valued logic. This normal form was not the one for Boolean logic, and, in fact they showed that it was the same one as for logic programming propositional logic. Aside from providing a general framework for determining when two propositional logics are equivalent, we sort out in detail what was really said and done in the above mentioned papers - as far as it pertains to the fuzzy logics.

2 Propositional Logic

Propositional logic deals with the properties of some set of logical connectives, most often the connectives 'and' ($\wedge$), 'or' ($\vee$), 'not' ($\neg$), and their derivates. These can be viewed as operations defined on sets of propositions. The connective 'and' for example would yield a binary operation on the set of propositions. So the first things that have to be specified are which connectives the logic to be built will deal with. This is done by giving a collection of connective symbols, such as $\wedge$, $\vee$, and $\neg$, and arities of the corresponding operations (in this case 2, 2, and 1). When studying the logic of the connectives one does not want to deal with specific propositions from a given area, but rather propositional variables that could be replaced by any propositions. For this purpose one takes a non-empty set X of variables, whose elements we think of as propositional variables, standing for propositions. Now the formulas of our propositional logic can be built up from the propositional variables by combining them with the connective symbols. The set $\mathcal{P}$ of all well-formed formulas is usually described inductively as the smallest subset of the set of all strings in the variables, connective symbols, and parentheses satisfying

1. If x is a propositional variable then x is a well-formed formula;

2. If α and β are well-formed formulas then so are $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\neg\alpha)$.

In $\mathcal{P}$ logical meaning is not accounted for; any two formulas that are distinct as strings of symbols are not equal as formulas. For example the strings $x \wedge y$ and $y \wedge x$ are different as strings even though they are identified in most logics. Thus it also is the case that the formulas that will carry the logic are the same for any two propositional logics. The difference and the logical meaning are introduced when an *algebra of truth values* is given. This is just an algebra $\mathbb{A}$ with the same number of basic operations as there are connective symbols and with matching arities. In many cases the operations corresponding to conjunction and disjunction are taken to be idempotent or to have other special properties. However, we wish to stress that we are not placing any such restrictions on the algebra $\mathbb{A}$.

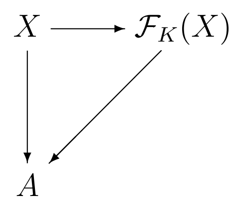
An *interpretation* or *model* of the formulas is a map $t : X \to \mathbb{A}$, that is, a function that assigns a truth value to each propositional variable. This corresponds to interpreting the variables as specific propositions that have definite truth values. Once this has been done the truth value of every proposition corresponding to a well-formed formula can be determined. That is, we get a map $\tilde{t} : \mathcal{P} \to \mathbb{A}$ extending t simply by specifying that we want $\tilde{t}\,((\alpha \wedge \beta)) = \tilde{t}\,(\alpha) \wedge \tilde{t}\,(\beta)\,, \tilde{t}\,((\alpha \vee \beta)) = \tilde{t}\,(\alpha) \vee \tilde{t}\,(\beta)$, and $\tilde{t}\,((\neg\alpha)) = \neg\tilde{t}\,(\alpha)$ for all well-formed formulas α and β. This latter condition can be expressed by saying $\tilde{t}$ is a homomorphism between the two algebras. Here we consider $\mathcal{P}$ with the operations defined simply as concatenation. For example, given formulas α and β in $\mathcal{P}$, we define the operation on $\mathcal{P}$ corresponding to the operation symbol $\wedge$ to give the string $(\alpha \wedge \beta)$ in $\mathcal{P}$. Since $\mathcal{P}$ is generated by X as an algebra, knowing the action of $\tilde{t}$ on X is enough to completely determine $\tilde{t}$. Finally, the fact that every map $t : X \to \mathbb{A}$ gives rise to a homomorphism $\tilde{t} : \mathcal{P} \to \mathbb{A}$ reflects the fact that $\mathcal{P}$ is *freely generated* by the set X. Thus, we have a one-to-one correspondence between the set $\mathbb{A}^X$ of all interpretations of the variables and the set $Hom\,(\mathcal{P}, \mathbb{A})$ of all homomorphisms from $\mathcal{P}$ to $\mathbb{A}$.

These interpretations are what specify the propositional logic. In the propositional logic determined by $\mathbb{A}$, we say that two formulas p and q are *logically equivalent*, and we write $p \sim_{\mathbb{A}} q$, if for each interpretation, the truth value assigned to p is equal to the one assigned to q. The propositional logic $L_{\mathbb{A}}$ determined by $\mathbb{A}$ is the quotient algebra $\mathcal{P}/\sim_{\mathbb{A}}$ in which formulas from $\mathcal{P}$ are identified if they are logically equivalent.

We thus see that the logic is completely determined by the algebraic properties of the algebra $\mathbb{A}$ of truth values. It is therefore of no surprise that the algebraic theory of $\mathbb{A}$ has great impact on the nature of the logic. In order to make this more precise, we need some concepts and results from universal algebra which we present in the following section.

3 Universal Algebra for Propositional Logic

The material discussed herein can be found in greater detail in any standard text on universal algebra such as [3]. When discussing the algebraic theory of a class of algebras, it is convenient to specify which kinds of basic operations we are considering on each algebra and to know which operation corresponds to which from algebra to algebra. To this end, we define a *type* of algebras to be a set τ of operation symbols of specified arity. For example, the type of lattices is a set consisting of two symbols, say $\{\vee, \wedge\}$, that are both binary. It is important here to think of $\vee$ and $\wedge$ as symbols and not as operations on a set. We will denote the class of all abstract algebras of type τ by $\mathfrak{A}_\tau$. Given a class K of algebras of some fixed type τ and a non-empty set X, the free K-algebra, freely generated by X, denoted by $\mathcal{F}_K(X)$, is the algebra (not necessarily in K) satisfying the property that there is a mapping $X \longrightarrow \mathcal{F}_K(X)$ so that given any mapping $X \longrightarrow A$, with $A \in \mathfrak{A}_\tau$, there is a unique homomorphic extension to $\mathcal{F}_K(X)$. That is, there is a unique homomorphism $\mathcal{F}_K(X) \longrightarrow A$ so that the diagram



commutes. In our setting, K will contain arbitrarily large algebras, and in this case it follows from the definition above that the map $X \longrightarrow \mathcal{F}_K(X)$ is an injection and that $\mathcal{F}_K(X)$ is generated by the image of X. It can also be shown that $\mathcal{F}_K(X)$ exists for any class K and that it is unique up to isomorphism. Given a set X, the free algebra generated by X in $\mathfrak{A}_\tau$ is called the *absolutely free algebra* of type τ over the set X. This algebra is also the term algebra of type τ over X. We denote it by $\mathcal{T}_\tau(X)$ and call its elements *terms*. To illustrate what the terms look like, we describe them inductively within the set of all strings over the alphabet $X \cup \tau$.

1. If $x \in X$ then x is a term;

2. If t_1, $t_2, ..., t_n$ are terms and $f \in \tau$ and f is n-ary, then the string $f t_1 t_2 \cdots t_n$ obtained by concatenation is again a term.

The set of terms is the smallest set of strings satisfying both of these properties. Essentially, they can be thought of as 'polynomials' of the given type in the variables in X. This inductive definition may seem similar to the one for well-formed formulas. And in fact this is a generalization of that definition. If τ is the type of our algebra of truth values, then $\mathcal{T}_\tau(X)$ is exactly the algebra $\mathcal{P}$ of well-formed formulas described above with the minor typographical change of writing $(\alpha \wedge \beta)$ as $\wedge\alpha\beta$ (this saves having to include a lot of parentheses and allows for the definition to apply to operations of greater arity than two).

A non-empty class of algebras of type τ is called a *variety* if it is closed under subalgebras, homomorphic images, and direct products. If $\mathcal{V}$ is a variety, $K \subseteq \mathcal{V}$, and $\mathcal{V}$ has no proper subvariety containing K, then $\mathcal{V}$ is generated by K, and we write $\mathcal{V}ar(K) = \mathcal{V}$ and $\mathcal{V}ar(\mathbb{A}) = \mathcal{V}$ if $K = \{\mathbb{A}\}$. When discussing algebras, a central issue is the equational theory of the algebras. In order to talk about this we first have to make clear what an equation is, and what it means for one to be satisfied in an algebra. An *equation of type τ over a set X* is an expression of the form

$$p \approx q$$

where p, q are elements of $\mathcal{T}_\tau(X)$. We will view a set E of equations of type τ over X as a subset of $\mathcal{T}_\tau(X) \times \mathcal{T}_\tau(X)$. Such a set E determines an *equational class* of algebras $\mathcal{M}od(E) = \{\mathbb{A} \in \mathfrak{A}_\tau : p \approx q$ holds in $\mathbb{A}$ for each $p \approx q \in E\}$. Here $p \approx q$ holds in $\mathbb{A}$ means that for each map $i : X \to \mathbb{A}$ the terms p and q get mapped to the same element of $\mathbb{A}$ by the unique extension of i given by the universal mapping property of $\mathcal{T}_\tau(X)$. It is now clear that in the case where the algebra $\mathbb{A}$ is the algebra of truth values for a propositional logic, to say that two terms, or in other words two well-formed formulas, α and β are logically equivalent in the logic given by $\mathbb{A}$ is the same as saying that the equation $\alpha \approx \beta$ holds in $\mathbb{A}$.

We define for a collection $K \subseteq \mathfrak{A}_\tau$ *the theory of K over X* to be

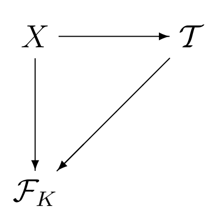
$$\mathcal{T}h_X(K) = \{p \approx q \in \mathcal{T}_\tau(X) \times \mathcal{T}_\tau(X) : p \approx q \text{ holds in } \mathbb{A} \text{ for each } \mathbb{A} \in \mathbb{K}\}.$$

Thus the equivalence relation $\sim_{\mathbb{A}}$ of logical equivalence defined in the previous section is exactly the set $\mathcal{T}h_X(\mathbb{A})$ and the *propositional logic over X determined by* $\mathbb{A}$ is

$$L_{\mathbb{A}}(X) = \mathcal{T}_\tau(X)/\mathcal{T}h_X(\mathbb{A}).$$

One of the most celebrated theorems in universal algebra is Birkhoff's theorem which says that varieties are exactly the equational classes, that is, classes of the form $\mathcal{M}od(E)$ for some set E of equations. This important theorem follows from the remarkable relationship between the free algebra, freely generated by X in a variety $\mathcal{V}$, and the theory of $\mathcal{V}$ over X.

In the following we will consider a fixed type τ and a non-empty set X, and we will suppress them in the notation. If $K \subseteq \mathfrak{A}$ is a variety then the free K-algebra is a quotient of $\mathcal{T}$ due to the universal mapping property



together with the fact that X generates $\mathcal{F}_K$. Thus $\mathcal{F}_K$ must be a quotient of $\mathcal{T}$. Indeed the following theorem holds.

Theorem 1 *(Birkhoff) For any $K \subseteq \mathfrak{A}$ the following statements are true:*

1. *$\mathcal{T}h(K)$ is a congruence relation on $\mathcal{T}$;*
2. *$\mathcal{F}_K$ exists and $\mathcal{F}_K = \mathcal{T}/\mathcal{T}h(K)$;*
3. *$\mathcal{F}_K = \mathcal{F}_{\mathcal{V}ar(K)} \in \mathcal{V}ar(K)$;*
4. *$\mathcal{T}h(K) = \mathcal{T}h(\mathcal{V}ar(K))$.*

In terms of propositional logics this means the following.

Theorem 2 *Let X be a non-empty set and $\mathbb{A}$ and $\mathbb{B}$ algebras of the type τ as described above. Then the following statements are true about the propositional logics over X determined by $\mathbb{A}$ and $\mathbb{B}$:*

1. *$L_{\mathbb{A}}(X) = \mathcal{F}_{\mathbb{A}}(X) = \mathcal{F}_{\mathcal{V}ar(\mathbb{A})}(X)$;*
2. *$L_{\mathbb{A}}(X) = L_{\mathbb{B}}(X)$ if and only if $\mathcal{V}ar(\mathbb{A}) = \mathcal{V}ar(\mathbb{B})$;*
3. *$L_{\mathbb{A}}(X) = \mathcal{T}(X)/\mathcal{T}h_X(\mathbb{A}) = \mathcal{T}(\mathbb{X})/\mathcal{T}h(\mathcal{V}ar(\mathbb{A}))$.*

Example 3 *Classical propositional logic is obtained when we choose $\mathbb{A}$ to be the two element Boolean algebra $\mathbf{2} = (\{0,1\}, \wedge, \vee, \neg, 0, 1)$[1]. In this case*

$$L_{\mathbb{A}}(X) = \mathcal{T}(X)/\mathcal{T}h_X(\mathbf{2}) = \mathcal{T}(X)/\mathcal{T}h_X(\mathcal{V}ar(\mathbf{2})) = \mathcal{F}_{\mathbb{B}}(X)$$

is the free Boolean algebra freely generated by X. And we get the exact same propositional logic no matter which Boolean algebra we choose as the set of truth values.

Other examples are provided by fuzzy logic and other non-classical logics presented in the next section.

Notice that if $\mathbb{A}$ is any finite algebra then there is a finite process for checking whether or not $\alpha \sim_{\mathbb{A}} \beta$. The variables occurring in α together with the ones occurring in β form a finite subset Y of X. It follows that $\alpha \sim_{\mathbb{A}} \beta$ if and only if $\tilde{t}(\alpha) = \tilde{t}(\beta)$ for each $t \in A^Y$, and A^Y is finite since both A and Y are finite. Thus the logical equivalence of the propositional logic given by an algebra $\mathbb{A}$ is finitely checkable if the variety generated by $\mathbb{A}$ is also generated by a finite algebra. We state this result as a corollary.

Corollary 4 *The logical equivalence relation $\sim_{\mathbb{A}}$ of the logic generated by $\mathbb{A}$ is given by a finite algorithm if there is a finite algebra $\mathbb{B}$ with $\mathcal{V}ar(\mathbb{A}) = \mathcal{V}ar(\mathbb{B})$.*

[1] We will use the same label for the operation in a specific algebra as for the corresponding function symbol, assuming that it will be clear from the context which we are talking about.

Before closing this section we want to stress the consequences of Theorem 2. When looking for an appropriate propositional logic structure for a particular situation, one just needs to study the universal algebraic properties of the corresponding algebras of truth values. If it has already been decided that the algebra of truth values is to be in a certain variety for example, then there will be as many choices of distinct logics as there are subvarieties of that variety. A common choice of truth value structure in fuzzy logic is a de Morgan system (see [5]). In [5, 6] we determined which de Morgan systems on $\mathbb{I}$ and on $\mathbb{I}^{[2]}$ are isomorphic. Of course isomorphic algebras generate the same variety, but non-isomorphic algebras can generate the same variety also, and hence determine the same logic. Any two Boolean algebras generate the same variety, for example, and the same propositional logic. This is also illustrated by examples in the next section and by the following example from which we can conclude that the logic determined by the system consisting of the unit interval as a bounded lattice together with a continuous t-norm is independent of the property that the t-norm is strict or nilpotent.

Example 5 *All algebras $(\mathbb{L}, \circ)$ with $\mathbb{L} = ([0,1], \wedge, \vee, 0, 1)$ and $\circ$ a strict (continuous, Archimedean) t-norm are isomorphic to the algebra $\mathbb{A} = (\mathbb{L}, \cdot)$ where $\cdot$ is multiplication on the unit interval. And all algebras $(\mathbb{L}, \circ)$ with $\circ$ a nilpotent (continuous, Archimedean) t-norm are isomorphic to the system $\mathbb{B} = (\mathbb{L}, *)$ where $x * y = (x + y - 1) \vee 0$. The two algebras $\mathbb{A} = (\mathbb{L}, \cdot)$ and $\mathbb{B} = (\mathbb{L}, *)$ are not isomorphic (see [5]). Let $\mathcal{V}(\mathbb{A})$ be the variety generated by $\mathbb{A}$, $\mathcal{V}(\mathbb{B})$ the variety generated by $\mathbb{B}$, and let $a \in (0,1)$. The relation $\sim$ on $\mathbb{A}$ given by $x \sim y$ if $x, y \in [0, a]$ is a congruence, and the quotient algebra $\overline{\mathbb{A}} = \mathbb{A}/\sim$ satisfies*

$$x \circ y = \begin{cases} x \cdot y & \text{if } x \cdot y > a \\ \bar{0} = \bar{a} & \text{if } x \cdot y \leq a \end{cases}$$

There are numbers $x, y > a$ with $x \cdot y \leq a$, so this t-norm is nilpotent and $\overline{\mathbb{A}}$ is isomorphic to $\mathbb{B}$. Since varieties are closed under quotients, we see that $\mathcal{V}(\mathbb{B}) \subseteq \mathcal{V}(\mathbb{A})$.

Now we find a subalgebra of $\prod_{\mathbb{Z}^+} \mathbb{B}$ of the form $(\mathbb{L}, \circ)$ with $\circ$ a strict t-norm. For n, m positive integers, r a positive real number, and $y \in (0,1)$, the powers $y^{[n]}$, $y^{\left[\frac{1}{m}\right]}$, and $y^{[r]}$ are defined by

$$y^{[n]} = \overbrace{y * y * \cdots * y}^{n \text{ times}} \qquad \left(y^{\left[\frac{1}{m}\right]}\right)^{[m]} = y \qquad y^{[r]} = \lim_{\frac{m}{n} \to r} \left(y^{\left[\frac{1}{m}\right]}\right)^{[n]}$$

Let $x_n \in \mathbb{B}$ with $x_n^{[n]} \neq 0$, $n \in \mathbb{Z}^+$. Set $a = (x_n) \in \prod_{\mathbb{Z}^+} \mathbb{B}$ and set $S = \left\{a^{[-\ln x]} : x \in [0,1]\right\}$. Then $\mathbb{S} = (S, \wedge, \vee, 0, 1, \circ)$, where $\wedge, \vee, 0, 1, \circ$ are the operations inherited from the coordinatewise operations on $\prod_{\mathbb{Z}^+} \mathbb{B}$, is a subalgebra of $\prod_{\mathbb{Z}^+} \mathbb{B}$ isomorphic to $\mathbb{A}$. Since varieties are closed under both products and subalgebras, we see that $\mathcal{V}(\mathbb{B}) = \mathcal{V}(\mathbb{A})$.

4 Some Fuzzy Logics

The propositional logic underlying what people loosely call fuzzy logic, or Lee-Chang fuzzy logic [9], is the logic you get when you take $\mathbb{I}$ to be the algebra of truth values. Here $\mathbb{I} = ([0,1], \wedge, \vee, 0, 1, \neg)$ is the unit interval of real numbers with the lattice operations induced by the natural order, and $\neg x = 1 - x$ as the negation. The papers [11, 4] address the properties of the underlying propositional logic. The main question motivating both papers seems to be the existence of a finite algorithm for determining logical equivalence. In the language of this paper, we can now put in perspective the work of these authors. Elkan essentially keeps the same algebra of truth values (the only difference is that he does not retain the constants 0 and 1 in the type). However, he changes the class of truth valuations or interpretations, thus deviating from the natural construction of a propositional logic. Recall that logical equivalence for $p, q \in \mathcal{P}$, in the logic given by $\mathbb{A}$, is $p \equiv_{\mathbb{A}} q$ if and only if $t(p) = t(q)$ for all $t \in Hom(\mathcal{P}, \mathbb{A})$. Since $\mathbf{2}$ is a subalgebra of $\mathbb{J} = ([0,1], \wedge, \vee, \neg)$, $Hom(\mathcal{P}, \mathbf{2}) \subseteq Hom(\mathcal{P}, \mathbb{J})$ and thus $p \equiv_{\mathbb{J}} q$ implies $p \equiv_{\mathbf{2}} q$. Now Elkan chooses to define the collection $\mathcal{T}$ of admissible truth valuations as $\mathcal{T} = \{t \in Hom(\mathcal{P}, \mathbb{J}) : p \equiv_{\mathbf{2}} q$ implies $t(p) = t(q)\}$. What he proves is that $\mathcal{T}$ is essentially equal to $Hom(\mathcal{P}, \mathbf{2})$! In effect, this says that he is just dealing with the natural classical two-valued logic as obtained with $\mathbb{A} = \mathbf{2}$. This is neither surprising nor difficult to prove.

Proposition 6 *For $t \in \mathcal{T}$, $\operatorname{Im}(t)$ is either the one or two element Boolean algebra.*

Proof. For $t \in \mathcal{T}$, $\equiv_{\mathbf{2}} \subseteq \{(p,q) : t(p) = t(q)\} = \operatorname{Ker}(t)$. This implies that the natural map $\mathcal{P}/\equiv_{\mathbf{2}} \to \mathcal{P}/\operatorname{Ker}(t)$ is a surjective homomorphism of type $(\wedge, \vee, \neg)$. Since $\mathcal{P}/\equiv_{\mathbf{2}}$ is a Boolean algebra, so is its image $\mathcal{P}/\operatorname{Ker}(t)$. So $\operatorname{Im}(t)$, which is isomorphic to $\mathcal{P}/\operatorname{Ker}(t)$ is a Boolean subalgebra of $\mathbb{J}$. But the only such subalgebras of $\mathbb{J}$ are the one and two element Boolean algebras.$\square$

As far as we can tell, the reason for this strange choice of admissible interpretations is to have a logical equivalence that can be checked with a finite algorithm. But since this collapses to classical propositional logic, it is of no interest for fuzzy logic considerations. In [11], the authors stay within the natural framework of propositional logic as described here, but they change the algebra of truth values from $\mathbb{J}$ to the algebra $\mathbb{J}^{[2]}$ of subintervals of $\mathbb{J}$. (We will describe this algebra in more detail later). For this logic, which they call *applied fuzzy propositional logic*, they develop a normal form and thus a formal way of checking logical equivalence. The normal form they arrive at is the normal form for de Morgan algebras (with the bounds left out). We will see below why this is the case. In the process of developing the normal form, they show that it is the appropriate normal form both for their applied fuzzy logic and for logic programming. Whereas the normal form and its relation to the logic generated by $\mathbb{J}^{[2]}$ is essentially well-known universal algebraic information, the fact that the logic obtained from interval-valued fuzzy logic (applied fuzzy logic) and the logic obtained from logic programming coincide seems to be a very interesting result.

The main justification provided in [11] for the switch from $\mathbb{J}$ to $\mathbb{J}^{[2]}$ was that this allows for a finite algorithm for checking logical equivalence. However, this is not a reasonable justification as there is a finite algorithm for checking logical equivalence in 'classical' fuzzy logic also. This has been known for a long time [10] and we will show it here as a direct consequence of well-known algebraic facts. According to the corollary in the last section, we just need to show the existence of a finite algebra generating the same variety. To this end we need to situate the algebras in question within algebraic theory.

The algebras in question are *de Morgan algebras:*

Definition 7 *A* de Morgan algebra *is an algebra* $\mathbb{L}$ *of the type* $(2, 2, 0, 0, 1)$ *satisfying*

1. $(L, \vee, \wedge, 0, 1)$ *is a distributive lattice with* $0, 1$.
2. $\neg(x \vee y) = \neg x \wedge \neg y$ *and* $\neg(x \wedge y) = \neg x \vee \neg y$ *are identities.*
3. $\neg(\neg x) = x$ *is an identity.*

We denote the equational class of de Morgan algebras by $\mathcal{M}$. A unary operation on a distributive lattice satisfying 2) and 3) is called an *involution.*

Example 8 *Let* **3** *denote the three element chain* $\{0, u, 1\}$ *with its unique involution. When we choose* $\mathbb{A}$ *to be the de Morgan algebra* **3** *then we get what is known as three-valued logic.*

Example 9 *The algebra* $\mathbb{I} = ([0, 1], \vee, \wedge, \neg, 0, 1)$ *as described above forms a de Morgan algebra. Using* $\mathbb{I}$ *as the algebra of truth values we get the propositional logic known as (classical) fuzzy logic.*

Example 10 *Given any de Morgan algebra* $\mathbb{L}$, *let* $\mathbb{L}^{[2]} = (L^{[2]}, \vee, \wedge, \neg, 0, 1)$ *where* $L^{[2]} = \{(x, y) : x, y \in L \text{ and } x \leq y\}$, $\vee$ *and* $\wedge$ *are defined coordinate-wise,* $\neg(x, y) = (\neg y, \neg x)$, $0_{L^{[2]}} = (0_L, 0_L)$ *and* $1_{L^{[2]}} = (1_L, 1_L)$. *Then* $\mathbb{L}^{[2]}$ *is again a de Morgan algebra. Since the pairs* (x, y) *satisfy* $x \leq y$, *they can be thought of as subintervals of* L. *Indeed the de Morgan algebra* $\mathbb{I}^{[2]}$ *is the one used as the algebra of truth values in the propositional logic known as interval fuzzy logic or practical fuzzy logic [11].*

The nature of each of the logics above is completely determined by which variety of de Morgan algebras is generated by the truth value algebra $\mathbb{A}$ in question. Therefore it is of interest to know the subvarieties, that is, the equational subclasses of the class of de Morgan algebras. This has long since been worked out and it turns out that there are very few subvarieties.

We denote the variety of de Morgan algebras by $\mathcal{M}$; the trivial subvariety of $\mathcal{M}$, consisting of all one-element algebras, by $\mathcal{M}_{-1}$; the subvariety of $\mathcal{M}$ generated by **2**, **3**, and $\diamond$, by $\mathcal{M}_0$, $\mathcal{M}_1$, and $\mathcal{M}_2$, respectively, where

$$\diamond \;=\; \begin{array}{ccccc} & & 1 & & \\ & / & & \backslash & \\ v & & & & w \\ & \backslash & & / & \\ & & 0 & & \end{array}$$

with $\neg v = v$, and $\neg w = w$.

Theorem 11 *[7] The subvarieties of $\mathcal{M}$ are*

$$\mathcal{M}_{-1} \subseteq \mathcal{M}_0 \subseteq \mathcal{M}_1 \subseteq \mathcal{M}_2 = \mathcal{M}$$

Moreover, for $L \in \mathcal{M}$, $L \in \mathcal{M}_0$ if and only if $x \wedge \neg x = 0$ is an identity in L (i.e. $\mathcal{M}_0$ is the class of Boolean algebras), and $L \in \mathcal{M}_1$ if and only if

$$[(x \vee \neg x) \wedge (y \wedge \neg y)] \vee [(y \vee \neg y) \wedge (x \wedge \neg x)] = (y \wedge \neg y) \vee (x \wedge \neg x)$$

is an identity in L.

The latter equality is equivalent to the inequality

$$x \wedge \neg x \leq y \vee \neg y$$

[7] and consequently to the equality

$$(x \wedge \neg x) \wedge (y \vee \neg y) = (x \wedge \neg x)\,.$$

A distributive lattice in which the inequality $x \wedge \neg x \leq y \vee \neg y$ holds is called a *normal i-lattice* [7] or a *Kleene algebra* [2, 8].

Theorem 12 *The de Morgan algebras* **3** *and $\mathbb{I}$ both generate the subvariety $\mathcal{M}_1$ of $\mathcal{M}$.*

Proof. We know that **3** generates $\mathcal{M}_1$ by the definition of $\mathcal{M}_1$. Also, since $\mathbb{I}$ satisfies the equation $(x \wedge \neg x) \wedge (y \vee \neg y) = (x \wedge \neg x)$ we have $\mathcal{V}ar(\mathbb{I}) \subseteq \mathcal{M}_1$. On the other hand, $\mathbb{I}$ is not Boolean so $\mathcal{M}_0$ is a proper subvariety of $\mathcal{V}ar(\mathbb{I})$. It follows that $\mathcal{V}ar(\mathbb{I}) = \mathcal{M}_1$.$\square$

Theorem 13 *The de Morgan algebras $\mathbf{3}^{[2]}$ and $\mathbb{I}^{[2]}$ both generate the equational class $\mathcal{M}$ of all de Morgan algebras.*

Proof. We know that $\diamond$ generates $\mathcal{M}$. And it is easy to see that $\diamond$ can be realized as a subalgebra of $\mathbf{3}^{[2]}$ which can be realized as a subalgebra of $\mathbb{I}^{[2]}$. Thus $\mathcal{M} = \mathcal{V}ar(\diamond) \subseteq \mathcal{V}ar(\mathbf{3}^{[2]}) \subseteq \mathcal{V}ar(\mathbb{I}^{[2]})$. On the other hand, $\mathbb{I}^{[2]}$ is a de Morgan algebra, so $\mathcal{V}ar(\mathbb{I}^{[2]}) \subseteq \mathcal{M}$. It follows that $\mathcal{V}ar(\mathbf{3}^{[2]}) = \mathcal{V}ar(\mathbb{I}^{[2]}) = \mathcal{M}$.$\square$

Thus (classical) fuzzy propositional logic is the same as three-valued logic, and interval (practical) fuzzy propositional logic is the same as the four-valued logic with the algebra $\diamond$ of truth values. Now what import does this have for fuzzy set theory? The following example illustrates one of the consequences.

Example 14 *The theorem implies that given two expressions involving variables, say A, B, and C, connected with $\wedge$, $\vee$, and $\neg$ (where $\neg$ denotes the usual negation), such as*

$$A \wedge ((\neg A \wedge B) \vee (\neg A \wedge \neg B) \vee (\neg A \wedge C)) \text{ and } A \wedge \neg A,$$

the equality

$$A \wedge ((\neg A \wedge B) \vee (\neg A \wedge \neg B) \vee (\neg A \wedge C)) = A \wedge \neg A$$

will hold for all fuzzy sets A, B, and $C : X \rightarrow [0,1]$ if and only if the equality holds for all choices of A, B, and C elements in $\mathbf{3} = \{0, u, 1\}$. In this example, there are up to 27 checks that need to be made. We check for the case $A = u$, $B = 0$, and $C = u$. In that case, the right hand side has value $u \wedge u = u$, and the left side is

$$u \wedge ((u \wedge 0) \vee (u \wedge 1) \vee (u \wedge u)) = u \wedge (0 \vee u \vee u) = u.$$

The other 26 cases also yield equalities, so the expression is an identity for fuzzy sets. No matter what the fuzzy sets A, B, and C are, for every $x \in X$, the equality

$$\begin{aligned} &A(x) \wedge ((\neg A(x) \wedge B(x)) \vee (\neg A(x) \wedge \neg B(x)) \vee (\neg A(x) \wedge C(x))) \\ &\quad = A(x) \wedge \neg A(x) \end{aligned}$$

holds.

So there is an algorithm for checking the equality of two expressions in fuzzy sets involving the connectives max, min, and the usual negation $x \rightarrow 1 - x$. We now turn to the related topic of normal forms.

5 Canonical Forms

Even though the existence of an algorithm for checking logical equivalence, allows us to check whether two formulae are equivalent, it does not give a convenient, natural choice for which among equivalent expressions to work with. A normal form is a description of a canonical representative of each equivalence class and a process for reducing (or expanding as the case might be) an arbitrary expression to the normal form that is equivalent to it.

Since the propositional logic given by the algebra $\mathbb{A}$ of truth values is equal to the free algebra generated by the set of variables X in the variety generated by $\mathbb{A}$, a normal form for the logic is the same as a normal form for the free algebra.

The normal form for the free de Morgan algebra is well-known and is described in detail in the paper [11] (except that they leave out the bounds): every element is uniquely an irredundant disjunction of conjunctions, each of which involves only literals (i.e., variables and their negations). Here we of course consider the order in which the conjunctions appear, as well as the order of the literals within each conjunction, as immaterial. The irredundancy lies in discarding any conjunction of literals involving fewer literals than another of the conjunctions present in the disjunction and in discarding repetitions of literals within each conjunction. Further, to incorporate the extremes, 0 and 1, we have to discard any conjunction involving 0 or $\neg 1$ (if no conjunctions are left, then the normal form for the element is 0); within each conjunction we must discard any occurrence of 1 and $\neg 0$ unless this literal makes up an entire conjunction (in which case the normal form for the entire formula is 1).

The normal form for the free Boolean algebra is of course also well-known: every element is uniquely a disjunction of complete conjunctions of literals. Here a complete conjunction of literals is a conjunction of literals in which each variable occurs exactly once. The empty disjunction is 0, and the disjunction of all the complete conjunctions is 1.

This leaves the middle case of the free Kleene algebra. It is very likely that the normal form for this algebra also is well-known, but we were not able to find it in the literature. So we present here in slightly more detail than the two preceding cases a normal form for the free Kleene algebra.

The fact that is of fundamental use in finding all of these normal forms is that in each case (Boolean, Kleene, de Morgan) the free algebra generated by $x_1, x_2, \ldots, x_n$ is a bounded distributive lattice generated by the (finite) set of literals: x_1, x_2, …, x_n, $\neg x_1$, $\neg x_2$, …, $\neg x_n$. It is well known (and we will take this for granted here) that each element of a finite distributive lattice is the join of all the join irreducibles (i.e., elements that cannot be written as a proper join) below it. This is the content of Birkhoff's duality for finite distributive lattices. We will take advantage of a slight refinement of this: every element is the join of the maximal join irreducibles below that element.

Thus getting a normal form just boils down to determining the join irreducibles (and the ordering between them). It is clear that in each of the above cases, an element in the free algebra only stands a chance of being join irreducible if it is equal to a conjunction of literals or is equal to 1. The normal form for de Morgan algebras stems from realizing that all conjunctions of literals as well as 1, are join irreducible. The normal form for Boolean algebras stems from realizing that the only join irreducible elements in the Boolean case are the complete conjunctions of literals. We now determine which elements are join irreducible in the Kleene case.

Proposition 15 *Consider the free Kleene algebra $\mathbb{F}$ over the variables*

$$X = \{x_1, x_2, \ldots, x_n\}$$

An element of $\mathbb{F}$ is join irreducible if and only if it is equal to 1 or it is a conjunction of literals satisfying at least one of the following two conditions:

1. *It contains at most one of the literals for each variable.*

2. *It contains at least one of the literals for each variable.*

Proof. We know that $\mathbb{F} = \mathbb{T}(X)/\mathcal{T}h_X(\mathbf{3})$. So for two terms $s, t \in T(X)$ we have $[s]_\mathbb{F} = [t]_\mathbb{F}$ (which we will denote by $s =_\mathbb{F} t$) if and only if the equation $s \approx t$ holds in the Kleene algebra $\mathbf{3}$. In particular, since $s \leq_\mathbb{F} t$ is equivalent to the term equation $s \wedge t \approx t$, this holds if and only if $s(a_1, \ldots, a_n) \leq t(a_1, \ldots, a_n)$ for all $(a_1, \ldots, a_n) \in \mathbf{3}^n$. The following sequence of lemmas completes the proof.

Lemma 16 *Given two conjunctions of literals c_1 and c_2 we have $c_1 \leq_\mathbb{F} c_2$ if and only if each literal in c_2 occurs in c_1.*

Proof. The 'if' part is clear. On the other hand, if there is a literal $l(x_i)$ ($= x_i$ or $\neg x_i$) which occurs in c_2 but not in c_1, then pick the interpretation (or truth valuation)

$$\begin{array}{rcl} t & : & X \to \mathbf{3} \\ x_i & \mapsto & \begin{cases} 0 \text{ if } l(x_i) = x_i \\ 1 \text{ if } l(x_i) = \neg x_i \end{cases} \\ x & \mapsto & u \text{ for } x \neq x_i \end{array}$$

We see that $\tilde{t}(c_2) = 0$ whereas $\tilde{t}(c_1) \geq u$, so $c_1 \not\leq_\mathbb{F} c_2$.

Lemma 17 *If a conjunction of literals, c, contains at least one of the literals for each variable, then c is join irreducible.*

Proof. Let $d_1, \ldots, d_k$ be conjunctions strictly below c. Then there are variables $y_1, \ldots, y_k$ in X so that $l_j(y_j)$ is in d_j but $l_j(y_j)$ is not in c. Since, for each $j \in \{1, \ldots, k\}$, the conjunction c contains one of the literals y_j or $\neg y_j$, it follows that all the y_j's are distinct. Therefore we can define the following truth valuation:

$$\begin{array}{rcl} t & : & X \to \mathbf{3} \\ y_j & \mapsto & \begin{cases} 0 \text{ if } l_j(y_j) = y_j \\ 1 \text{ if } l_j(y_j) = \neg y_j \end{cases} \\ y & \mapsto & u \text{ for } y \neq y_j,\ j = 1, \ldots, k \end{array}$$

Then we have $\widetilde{t}(d_1 \vee \ldots \vee d_k) = \widetilde{t}(d_1) \vee \ldots \vee \widetilde{t}(d_k) = 0 \vee \ldots \vee 0 = 0$ whereas $\widetilde{t}(c) \geq u$, so $d_1 \vee \ldots \vee d_k <_\mathbb{F} c$, and it follows that c is join irreducible.

Lemma 18 *If c is 1 or a conjunction of literals containing at most one of the literals for each variable, then c is join irreducible.*

Proof. Consider the truth valuation:

$$\begin{array}{rcl} t & : & X \to \mathbf{3} \\ x & \mapsto & 1 \text{ if } x \text{ occurs in } c \\ x & \mapsto & 0 \text{ if } \neg x \text{ occurs in } c \\ x & \mapsto & u \text{ otherwise} \end{array}$$

Since c is such that not both x and $\neg x$ are contained in c, t is well-defined, and it follows that $\widetilde{t}(c) = 1$. On the other hand, if $d_1, \ldots, d_k$ are conjunctions strictly below c, then $\widetilde{t}(d_1 \vee \ldots \vee d_k) = \widetilde{t}(d_1) \vee \ldots \vee \widetilde{t}(d_k) \leq u$, so $d_1 \vee \ldots \vee d_k <_{\mathbb{F}} c$, and it follows that c is join irreducible.

Lemma 19 *If a conjunction of literals, c, contains both literals for some variable but does not contain at least one of the literals for each variable, then c is join reducible.*

Proof. Let $c = x \wedge \neg x \wedge c'$, and let y be some variable that is not involved in c. Now, since in a Kleene algebra $a \wedge \neg a \leq b \vee \neg b$ for any two elements a and b, we have $c = x \wedge \neg x \wedge c' = (x \wedge \neg x \wedge (y \vee \neg y)) \wedge c' = c \wedge (y \vee \neg y) = (c \wedge y) \vee (c \wedge \neg y)$and since by our assumption c does not contain either literal for y, it follows from Lemma161 above that $c \wedge y$ as well as $c \wedge \neg y$ are conjunctions strictly below c. That is, c is join reducible.$\square$

Now we are ready to describe the normal form for Kleene algebras and the procedure for obtaining the normal form.

Definition 20 *A formula w in the term (or formula) algebra $\mathcal{T}(x_1, ..., x_n)$ of type $(2, 2, 0, 0, 1)$ is in* Kleene normal form *provided it is equal to the string 0 or 1 or a (non-empty) disjunction of literals satisfying:*

1. *each of the conjunctions of literals involved is join-irreducible as an element of $\mathbb{F} = \mathcal{T}(x_1, ..., x_n)/\mathcal{T}h(\mathbf{3})$;*
2. *The conjunctions of literals involved are pairwise incomparable as elements of $\mathbb{F} = \mathcal{T}(x_1, ..., x_n)/\mathcal{T}h(\mathbf{3})$;*
3. *There is no repetition of literals within any one of the conjunctions.*

Two terms (or formulae) in normal form are considered equal if the only difference between them occurs in the order and the association among the disjunctions or within the individual conjunctions (this makes sense since both join and meet are both commutative and associative).

Notice that the properties (1) and (2) as stated are not syntactic in nature. However, they are made so by Proposition 15 and by Lemma 16 in its proof.

Theorem 21 *Every element in $\mathcal{T}(x_1, ..., x_n)$ is equivalent in $\mathbb{F} = \mathcal{T}(x_1, ..., x_n)/\mathcal{T}h(3)$ to a unique term (or formula) in Kleene normal form.*

As mentioned earlier, once we have seen that this corresponds to the representation of each element, uniquely, as the join of the maximal join irreducible elements below it, this theorem follows from Birkhoff's representation theorem for finite distributive lattices. Instead of getting into this, we will describe the procedure for putting an arbitrary word in $\mathcal{T}(x_1, ..., x_n)$ in Kleene normal form:

1. Given a formula (or word) w in $\mathcal{T}(x_1, x_2, \ldots, x_n)$, first use de Morgan's laws to move all the negations in, so that the word is rewritten as a word w_1 which is of lattice type in the literals, 0, and 1.

2. Next use the distributive law to obtain a new word w_2 from w_1 which is a disjunction of conjunctions involving the literals, 0, and 1. Given a word in lattice-disjunctive normal form, such as w_2, we will call the conjunctions that the word is written in, a disjunction of 'full' conjunctions of that word. At this point, discard any full conjunction in which 0 or $\neg 1$ appears as one of the conjuncts. Also discard any repetition of literals from any full conjunction, as well as 1 and $\neg 0$ from any full conjunction in which they do not appear alone (if a full conjunction consists entirely of 1's and $\neg 0$'s, then replace the whole thing by 1). This yields a word w_3.

3. Now discard all non-maximal conjunctions among the full conjunctions that w_3 is a disjunction of. The type of conjunctions we now are dealing with are either conjunctions of literals or 1 by itself. Of course 1 is above all the others and, as is described in Lemma 16 of the above proof, one conjunction of literals is below another if and only if the former contains all the literals contained in the latter. This process yields a word w_4.

4. At this point, replace any full conjunction of literals, c, which contains both literals for at least one variable by the disjunction of all the conjunctions of literals below c that contain exactly one of the literals for each variable not occurring in c.

5. Finally, again discard all non-maximal conjunctions among the full conjunctions that are left, and if no conjunctions are left, then replace the word by 0. The word thus obtained is now in the normal form described above.

Example 22 *We illustrate the Kleene normal form with the two equivalent expressions*

$$\begin{aligned} w &= A \wedge ((\neg A \wedge B) \vee (\neg A \wedge \neg B) \vee (\neg A \wedge C)) \\ w' &= A \wedge \neg A \end{aligned}$$

used in Example 4.9 and the three variables, A*,* B*, and* C*.*

(a) *Because* w *and* w' *are both of bounded lattice type in the literals,* 0*, and* 1*, we have* $w_1 = w$ *and* $w'_1 = w'$*.*

(b) *Applications of the distributive law lead to disjunctions of conjunctions involving the literals:*

$$\begin{aligned} w_2 &= (A \wedge \neg A \wedge B) \vee (A \wedge \neg A \wedge \neg B) \vee (A \wedge \neg A \wedge C) \\ w'_2 &= A \wedge \neg A \end{aligned}$$

(c) *Neither of the expressions in #2 contains any non-maximal conjunctions, so* $w_3 = w_2$ *and* $w'_3 = w'_2$*.*

(d) *Replace*

$$\begin{array}{lll} A \wedge \neg A \wedge B & \textit{by} & (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge B \wedge \neg C) \\ A \wedge \neg A \wedge \neg B & \textit{by} & (A \wedge \neg A \wedge \neg B \wedge C) \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \\ A \wedge \neg A \wedge C & \textit{by} & (A \wedge \neg A \wedge C \wedge B) \vee (A \wedge \neg A \wedge C \wedge \neg B) \end{array}$$

and

$$\begin{array}{ll} A \wedge \neg A \quad \textit{by} & (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge \neg B \wedge C) \\ & \vee\, (A \wedge \neg A \wedge B \wedge \neg C) \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \end{array}$$

to get

$$\begin{aligned} w_4 &= (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge B \wedge \neg C) \vee (A \wedge \neg A \wedge \neg B \wedge C) \\ &\quad \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \vee (A \wedge \neg A \wedge C \wedge B) \vee (A \wedge \neg A \wedge C \wedge \neg B) \\ w'_4 &= (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge \neg B \wedge C) \\ &\quad \vee (A \wedge \neg A \wedge B \wedge \neg C) \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \end{aligned}$$

(e) *Discarding all non-maximal conjunctions among the full conjunctions that are left means in this case, simply discarding repetitions, leading to the normal forms*

$$\begin{aligned} w_5 &= (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge B \wedge \neg C) \\ &\quad \vee (A \wedge \neg A \wedge \neg B \wedge C) \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \\ w'_5 &= (A \wedge \neg A \wedge B \wedge C) \vee (A \wedge \neg A \wedge \neg B \wedge C) \\ &\quad \vee (A \wedge \neg A \wedge B \wedge \neg C) \vee (A \wedge \neg A \wedge \neg B \wedge \neg C) \end{aligned}$$

References

[1] Balbes, R., and P. Dwinger, *Distributive Lattices*, University of Missouri Press, Columbia, Missouri (1974).

[2] Brignole, D., and A. Monteiro, Characterization des algèbres de Nelson par des égalités, *Notas de Lógica Matemática*, Instituto de Matemática Universidad del sur Bahía Blanca **20** (1964).

[3] Burris, S., and H. D. Sankappanavar, *A Course in Universal Algebra*, Springer Verlag, N. Y. (1980).

[4] Elkan, C., The paradoxical success of fuzzy logic, *Proceedings of AAAI-93*, American Association for Artificial Intelligence 1993 Conference, pp. 698-703.

[5] Gehrke, M., C. Walker, and E. Walker, de Morgan Systems on the Unit Interval, *International Journal of Intelligent Systems*, 11(1996), 733-750.

[6] Gehrke, M., C. Walker, and E. Walker, Some Comments on Interval Valued Fuzzy Sets, *International Journal of Intelligent Systems*, 11(1996), 751-759.

[7] Kalman, J. A., Lattices with involution, *Trans. Amer. Math. Soc.* 87 (1958), 485-491.

[8] Kleene, S. C., *Introduction to metamathematics*, North Holland Pub. Co., Amsterdam (1952).

[9] Lee, R. C. T., and C. L. Chang, Some properties of fuzzy logic, *Information and Control*, **19**(1) (1971), 417-431.

[10] Muller, D. E., Treatment of Transition Signals in Electronic Switching Circuits by Algebraic Methods, *IRE Transactions on Electronic Computers*, EC-**8**, No. 3, (September 1959).

[11] Nguyen„ H. T., O. M. Kosheleva, and V. Kreinovich, Is the success of fuzzy logic really paradoxical? or What is the actual logic behind expert systems?, *International Journal of Intelligent Systems*, **11** (1996).

[12] Zadeh, L. A., Fuzzy Sets, *Information and Control*, **8** (1965), pp. 338-353.

International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 5(1997) 223-238