EQ-LOGICS WITH DELTA CONNECTIVE

M. DYBA AND V. NOVÁK

Abstract. In this paper we continue development of formal theory of a special class of fuzzy logics, called EQ-logics. Unlike fuzzy logics being extensions of the MTL-logic in which the basic connective is implication, the basic connective in EQ-logics is equivalence. Therefore, a new algebra of truth values called EQ-algebra was developed. This is a lower semilattice with top element endowed with two binary operations of fuzzy equality and multiplication. EQ-algebra generalizes residuated lattices, namely, every residuated lattice is an EQ-algebra but not vice-versa.

In this paper, we introduce additional connective \( \Delta \) in EQ-logics (analogous to Baaz delta connective in MTL-algebra based fuzzy logics) and demonstrate that the resulting logic has again reasonable properties including completeness. Introducing \( \Delta \) in EQ-logic makes it possible to prove also generalized deduction theorem which otherwise does not hold in EQ-logics weaker than MTL-logic.

1. Introduction

In this paper we continue development of the formal theory of EQ-logics that were introduced in [3] and that can be taken as a special class of fuzzy logics that differ from residuated fuzzy logics (based on MTL-logic) ([1, 6, 7, 8, 13] and elsewhere) by taking equivalence as the basic connective instead of implication. As can be demonstrated, EQ-logics lay even lower (are more fundamental) than the MTL-logic that has been till now taken as the simplest fuzzy one. Namely, the MTL-logic is obtained from the basic EQ-logic introduced in [3] by adding several axioms. Our idea of EQ-logics should be considered as a generalization of the equational (classical) logics due to Gries and Schneider (cf. [9, 10, 17]). Hence, it also naturally employs the equational style of formal proofs.

EQ-logics are based on a special algebra of truth values — the EQ-algebra introduced by Novák in [12] (cf. also [5, 14, 16]). It is a lower semilattice with top element \( 1 \) endowed with two binary operations of multiplication \( (\otimes) \) and fuzzy equality \( (\sim) \). Thus every two elements \( a \) and \( b \) are equal in a degree \( a \sim b \). The operation \( \sim \) is a natural interpretation of equivalence in logic. Axioms of EQ-algebra reflect basic properties required by the equality, namely to fit the supporting structure which is an ordered set.

EQ-algebras generalize residuated lattices, because they relax the tie between multiplication and residuation. The implication operation (operation interpreting
implication in logic) is here defined as \( a \rightarrow b = (a \land b) \sim a \). It is immediately seen that each residuated lattice gives rise to an EQ-algebra but not vice versa. Let us stress, however, that there are EQ-algebras with the same implication but different fuzzy equalities and so, the fuzzy equality, in general, cannot be reconstructed from the implication (in special cases, of course, this is possible, for example, in all linearly ordered EQ-algebras).

Another outcome of relaxation of multiplication from implication is that the multiplication does not need be commutative and still one implication is sufficient. This is not the case in residuated lattices where non-commutativity of multiplication immediately enforces two implications.

In this paper, we introduce the well-known delta connective \( \Delta \) in EQ-logic and demonstrate that the latter has reasonable properties including completeness. Recall that, because in linearly ordered algebras \( \Delta a = 1 \) iff \( a = 1 \), otherwise \( \Delta a = 0 \), this connective enables, in some sense, to "extract" a boolean substructure from the given algebra simply by pushing all values smaller than 1 to 0.

Let us emphasize that introducing \( \Delta \) in EQ-logics is not an end in itself. All the EQ-logics are not equivalent to MTL-logic presented in [3] suffer from one deficiency: the generalized deduction theorem does not hold in them. The situation changes if we introduce \( \Delta \). First of all, this enables us to prove the \( \Delta \)-deduction theorem. But there is one interesting moral follows from this: any well behaving logic requires a crisp equivalence, i.e., the \textit{general fuzzy equivalence is not sufficient}.

From an algebraic point of view, the clue lays in the extensionality of operations: if \( a = b \) and \( \odot \) is any (binary) operation then \( a \odot c = b \odot c \) for any \( c \). If we consider fuzzy equality \( \sim \) instead, then this property becomes

\[
a \sim b \leq a \odot c \sim b \odot c.
\]

In EQ-algebras, this inequality holds both for \( \odot = \land, \sim \) but not (in general) for the product \( \otimes \). Surprisingly, if we add (1) for \( \odot = \otimes \) then the EQ-algebra already becomes residuated. We can weaken the latter by introducing the \( \Delta \)-operation and assuming \( \Delta (a \sim b) \leq a \odot c \sim b \odot c \). This is already strong enough to assure validity of the \( \Delta \)-deduction theorem.

We apply standard methods developed, e.g., in [11]. From this point of view, our logic does not leave the realm of logics with algebraic semantics. On the other hand, introducing special algebra in which the main operation is symmetric fuzzy equality casts new light on development of logics, namely on the fuzzy ones where the principal connective is implication. We reveal fine nuances and interrelations and their role in logic.

It should also be emphasized that formal proofs proceed on many places in equational style. This means that the proof consists of a sequence of equivalences, each of which being derived from the previous ones using inference rules. Thus, we may start with a formula to be proved and end up with an axiom. This is the reason why equational proofs can be more easily automated.

The paper is structured as follows: the next section contains overview of EQ-algebras and EQ-logics. In Section 3, we introduce the EQ\( \Delta \)-logic (i.e., EQ-logic with the delta connective). We start with extension of EQ-algebra by the delta
operation and demonstration of its main properties as well as its representability. Then we introduce axioms and inference rules of $\text{EQ}_\Delta$-logic and prove various kinds of its properties concluded by the proof of completeness theorem.

2. EQ-logic: an Overview

In this section we recall basic definitions and properties of EQ-algebras and basic EQ-logic.

2.1. EQ-algebras. EQ-algebras were introduced in [12] and elaborated in more detail in [5, 16].

**Definition 2.1.** A non-commutative EQ-algebra $E$ is an algebra of type $(2, 2, 2, 0)$, i.e.

$$E = \langle E, \land, \otimes, \sim, 1 \rangle,$$

where for all $a, b, c, d \in E$:

(E1) $\langle E, \land, 1 \rangle$ is a commutative idempotent monoid (i.e. $\land$-semilattice with top element $1$). We put $a \leq b$ iff $a \land b = a$, as usual.

(E2) $\langle E, \otimes, 1 \rangle$ is a monoid and $\otimes$ is isotone w.r.t. $\leq$.

(E3) $a \sim a = 1$

(E4) $(a \land b) \sim c \otimes (d \sim a) \leq c \sim (d \land b)$

(E5) $(a \sim b) \otimes (c \sim d) \leq (a \sim c) \sim (b \sim d)$

(E6) $(a \land b \land c) \sim a \leq (a \land b) \sim a$

(E7) $a \otimes b \leq a \sim b$

The operation $\land$ is called meet, $\otimes$ is called multiplication and $\sim$ is the fuzzy equality. Note that in this definition $\otimes$ is not assumed to be commutative. Furthermore, we put

$$a \rightarrow b = (a \land b) \sim a$$

for all $a, b \in E$ and call this operation an implication.

The following theorem demonstrates that the properties of the fuzzy equality and implication we consider as fundamental are indeed fulfilled.

**Theorem 2.2.** Let $E$ be an EQ-algebra. The following holds for all $a, b, c \in E$:

(a) Symmetry: $a \sim b = b \sim a$, 
(b) Transitivity: $(a \sim b) \otimes (b \sim c) \leq a \sim c$,
(c) Transitivity of implication: $(a \rightarrow b) \otimes (b \rightarrow c) \leq a \rightarrow c$.
(d) Antitonicity of implication: $a \rightarrow b \leq (a \land c) \rightarrow b$.

If $E$ also contains the bottom element $0$, then we put

$$\neg a = a \sim 0,$$  \hspace{1cm}  (3)

and call $\neg a$ a negation of $a \in E$.

We distinguish several special classes of EQ-algebras (see [16]). Among them, very important for EQ-logics, are **good** EQ-algebras which satisfy the property

$$a \sim 1 = a, \hspace{1cm} a \in E.$$

There are also other special types of EQ-algebras which we are interested in this article. We say that an EQ-algebra is:
(i) **prelinear** if for \( a, b \in E \)

\[
\sup\{a \rightarrow b, b \rightarrow a\} = 1.
\]

(ii) **residuated** if for all \( a, b, c \in E \),

\[
(a \otimes b) \land c = a \otimes b \quad \text{iff} \quad a \land ((b \land c) \sim b) = a.
\]

Equivalently:

\[
(a \otimes b) \leq c \quad \text{iff} \quad a \leq b \rightarrow c.
\]

(iii) **lattice ordered** if it has also the binary operation \( \lor \) so that \( (E, \land, \lor) \) is a lattice.

(iv) **lattice** EQ-algebra (\( \ell \)EQ-algebra) if it is a lattice-ordered EQ-algebra in which the following substitution axiom holds, for all \( a, b, c, d \in E \):

\[
(E8) \quad ((a \lor b) \sim c) \otimes (d \sim a) \leq (d \lor b) \sim c.
\]

The following theorem shows that EQ-algebra enriched with the adjunction condition leads to a residuated EQ-algebra that is commutative (i.e. with commutative multiplication).

**Theorem 2.3.** [5] Let \( E \) be a residuated EQ-algebra. Then its multiplication \( \otimes \) is commutative.

Note that in good EQ-algebras, only one side of the adjunction condition is satisfied, namely

\[
a \leq b \rightarrow c \quad \text{implies} \quad a \otimes b \leq c
\]

and so, EQ-algebras, in general, are not residuated.

There are several conditions which make good EQ-algebra residuated (see [5, Proposition 6]). We mention that only one of them will be needed later.

**Lemma 2.4.** Let \( E \) be an EQ-algebra. The following statements are equivalent:

(a) \( E \) is residuated,

(b) \( E \) is good and for all \( a, b, c \in E \) it holds that

\[
a \rightarrow b \leq (a \otimes c) \rightarrow (b \otimes c).
\]

**Theorem 2.5.** [4] If a good EQ-algebra \( E \) satisfies

\[
(a \rightarrow b) \lor (d \rightarrow (d \otimes (c \rightarrow (b \rightarrow a) \otimes c))) = 1
\]

for all \( a, b, c, d \in E \) then it is prelinear.

2.2. **Basic EQ-logic.** The basic EQ-logic was introduced in [3]. It has three basic binary connectives \( \land, \lor, \equiv \) and the truth constant \( \top \). Implication is a derived connective defined by

\[
A \Rightarrow B := (A \land B) \equiv A.
\]

The set of all formulas of the given language \( J \) is denoted by \( F_J \). The algebra of truth values is formed by a good non-commutative EQ-algebra. The following formulas are logical axioms of the basic EQ-logic:

(EQ1) \( A \equiv \top \equiv A \)

(EQ2) \( A \land B \equiv B \land A \)

(EQ3) \( (A \lor B) \lor C \equiv A \lor (B \lor C) \), \( \lor \in \{\land, \lor\} \)

(EQ4) \( A \land A \equiv A \)

(EQ5) \( A \land \top \equiv A \)
(EQ6) \( A \& \top \equiv A \)

(EQ7) \( \top \& A \equiv A \)

(EQ8a) \( (A \land B) \& C \Rightarrow (B \& C) \)

(EQ8b) \( (C \&(A \land B)) \Rightarrow (C \& B) \)

(EQ9) \( (A \land B) \equiv C \& (D \equiv A) \Rightarrow (C \equiv (D \land B)) \)

(EQ10) \( (A \equiv B) \& (C \equiv D) \Rightarrow (A \equiv C) \equiv (D \equiv B) \)

(EQ11) \( (A \Rightarrow (B \land C)) \Rightarrow (A \Rightarrow B) \)

Deduction rules of basic EQ-logic are equanimity rule \((EA)\) and Leibniz rule \((\text{Leib})\):

\[
(EA) \quad \frac{A, A \equiv B}{B}, \quad (\text{Leib}) \quad \frac{A \equiv B}{C[p := A] \equiv C[p := B]}
\]

where by \( C[p := X] \) for \( X := A \) and \( X := B \) we denote a formula (in the proofs also called as a “C-part”) resulting from \( C \) by replacing all occurrences of the variable \( p \) in \( C \) by the formula \( X \). The basic notions of truth evaluation, tautology, theory, etc. are defined as usual.

We list below some properties of the basic EQ-logic from [3] that will be used in this paper:

**Lemma 2.6.** (a) \( A \equiv \top \vdash A \), \hspace{1cm} (rule (T1))

(b) \( A \vdash A \equiv \top \), \hspace{1cm} (rule (T2))

(c) \( \vdash A \equiv A \).

(d) \( \vdash (A \Rightarrow B) \Rightarrow ((A \& C) \Rightarrow B) \),

(e) \( A, A \Rightarrow B \vdash B \), \hspace{1cm} (Modus Ponens)

(f) \( \vdash (\top \Rightarrow A) \equiv A \),

(g) \( A \Rightarrow B, B \Rightarrow C \vdash A \Rightarrow C \),

(h) \( A, B \vdash A \& B \),

(i) \( \vdash (A \equiv B) \equiv (B \equiv A) \),

(j) \( \vdash (A \equiv B) \Rightarrow (A \Rightarrow B) \),

(k) \( \vdash (A \equiv B) \& (C \equiv D) \Rightarrow (A \equiv C) \equiv (B \equiv D) \),

(l) \( \vdash (A \Rightarrow B) \& (B \Rightarrow A) \Rightarrow (A \equiv B) \),

(m) \( A \Rightarrow B, C \Rightarrow D \vdash (A \& C) \Rightarrow (B \& D) \),

(n) \( \vdash B \Rightarrow (A \Rightarrow B) \),

(o) \( \vdash (A \equiv B) \& (B \equiv C) \Rightarrow (A \equiv C) \),

(p) \( \vdash (A \&(A \equiv B)) \Rightarrow B \),

(q) \( \vdash (A \Rightarrow B) \& (B \Rightarrow C) \Rightarrow (A \Rightarrow C) \),

(r) \( \vdash (A \& B) \Rightarrow A \),

(s) \( \vdash (A \& B) \Rightarrow B \),

(t) \( A \Rightarrow (B \Rightarrow C) \vdash (A \& B) \Rightarrow C \),

(u) \( \vdash (C \Rightarrow A) \& (C \Rightarrow B) \Rightarrow (C \Rightarrow (A \land B)) \),

(v) \( \vdash A \Rightarrow ((A \Rightarrow B) \Rightarrow B) \).

Let us emphasize that the formal proofs proceed mainly in the equational style which is more natural for EQ-logics. We add a few new properties of the basic EQ-logic. From the valid algebraic counterparts of these inferences and theorems

---

*Actually, the symbol \( p \) is a metavariable for any propositional variable \( p, q, \ldots \) (see [17]).*
(see [5, 16]) and from the Completeness Theorem 2.8 noted below we can say that
they are provable.

Lemma 2.7. (a) $A \Rightarrow (B \Rightarrow (C \Rightarrow A)) \Rightarrow (A \Rightarrow C)$,
(b) $A \Rightarrow B \Rightarrow (C \Rightarrow A) \Rightarrow (C \Rightarrow B)$,
(c) $A \Rightarrow B \Rightarrow (A \equiv B) \Rightarrow ((A \land C) \equiv (B \land C))$,
(d) $A \Rightarrow (A \equiv D) \Rightarrow (((A \land B) \equiv C) \equiv ((D \land B) \equiv C))$,
(e) $A \Rightarrow (A \equiv D) \Rightarrow ((B \Rightarrow A) \equiv (B \Rightarrow D))$,
(f) $A \Rightarrow (A \Rightarrow D) \Rightarrow ((B \Rightarrow A) \Rightarrow (B \Rightarrow D))$,
(g) $A \Rightarrow ((A \equiv B) \& (C \equiv D)) \Rightarrow ((A \land C) \equiv (B \land D))$,
(h) $A \Rightarrow (B \Rightarrow (B \Rightarrow C) \Rightarrow (A \Rightarrow C))$,
(i) $A \Rightarrow (B \Rightarrow (B \Rightarrow C)) \Rightarrow (B \Rightarrow (A \Rightarrow C))$.

The proof of the following theorem can be found in [3].

Theorem 2.8 (Completeness). The following propositions are equivalent for every
formula $A$:
(a) $\vdash A$,
(b) $e(A) = 1$ for every good non-commutative EQ-algebra $E$ and a truth evaluation
$e : F_J \to E$.

3. EQΔ-logic

In this section we introduce a propositional EQ-logic extended by a $\Delta$ connective.
First, we must extend EQ-algebra by the corresponding ($\Delta$) operation that was
originally introduced in [15]. The property that $\Delta$ pushes all values smaller than $1$
to 0 holds only in linearly ordered algebras. Therefore, this operation must be, in
general, introduced by means of several axioms.

3.1. EQΔ-algebra.

Definition 3.1. An $EQ\Delta$-algebra is an algebra

$$\mathcal{E}_\Delta = (E, \land, \otimes, \sim, \Delta, 1)$$ (6)

which is a good non-commutative EQ-algebra $E$ extended by a unary additional
operation $\Delta : E \to E$ fulfilling the following axioms:

(E\Delta1) $\Delta 1 = 1$
(E\Delta2) $\Delta a \leq \Delta a$
(E\Delta3) $\Delta (a \sim b) \leq \Delta a \sim \Delta b$
(E\Delta4) $\Delta (a \land b) = \Delta a \land \Delta b$
(E\Delta5) $\Delta a = \Delta a \otimes \Delta a$

These axioms are related to the axioms of delta operation introduced in the book
[11, Chapter two]. Axiom (E\Delta1) expresses the fundamental property that $1$
is preserved. Axioms (E\Delta3), (E\Delta4), (E\Delta5) characterize distributivity of $\Delta$
over connectives. Axiom (E\Delta2) characterizes $\Delta$ as a certain closure operation.

The following two axioms are also needed.\(^1\) They assure good behavior of the
multiplication with respect to the crisp equality:

\(^1\)These two axioms will be necessary also for predicate EQ-logic.
Lemma 3.2. EQ-\( \Delta \)-algebra becomes residuated if \( \Delta \)-operation in (E\( \Delta \)6) or (E\( \Delta \)7) is omitted.

Proof. Assume that (E\( \Delta \)6) does not contain \( \Delta \). Using Lemma 2.4, we need only to show that \( \mathcal{E}_\Delta \) satisfies (4). Thus, using (2), (E\( \Delta \)6) without \( \Delta \), property \( a \sim b \leq a \rightarrow b \) (see Lemma 2 in [3]), monotonicity of \( \rightarrow \) and isotonicity of \( \odot \) we get

\[
\begin{align*}
  a \rightarrow b &= (a \land b) \sim a \leq ((a \land b) \odot c) \sim (a \odot c) = (a \odot c) \sim ((a \land b) \odot c) \\
                     &\leq (a \odot c) \rightarrow ((a \land b) \odot c) \leq (a \odot c) \rightarrow (b \odot c).
\end{align*}
\]

The second part of the proof can be proved in a similar way. \( \square \)

Example 3.3. An example of a finite non-trivial EQ\( \Delta \)-algebra is the following: we take the lattice structure in Figure 1. The multiplication and fuzzy equality are defined as follows:

\[
\begin{array}{cccccc}
\otimes & 0 & a & b & c & d & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
a & 0 & 0 & 0 & 0 & a & 0 \\
b & 0 & 0 & 0 & a & b & 0 \\
c & 0 & 0 & c & c & c & 0 \\
d & 0 & 0 & c & c & d & 0 \\
1 & 0 & a & b & c & d & 1 \\
\end{array}
\]

\[
\begin{array}{cccccc}
\sim & 0 & a & b & c & d & 1 \\
0 & 1 & d & c & b & a & 0 \\
a & d & 1 & c & b & a & 0 \\
b & c & c & 1 & a & b & b \\
c & b & b & a & 1 & c & c \\
d & a & a & b & c & d & 0 \\
1 & 0 & a & b & c & d & 1 \\
\end{array}
\]

Using (2) we obtain the implication:

\[
\begin{array}{cccccc}
\rightarrow & 0 & a & b & c & d & 1 \\
0 & 1 & 1 & 1 & 1 & 1 & 1 \\
a & d & 1 & 1 & 1 & 1 & 1 \\
b & c & c & 1 & c & 1 & 1 \\
c & b & b & b & 1 & 1 & 1 \\
d & a & a & b & c & 1 & 1 \\
1 & 0 & a & b & c & d & 1 \\
\end{array}
\]

The \( \Delta \)-operation is defined by \( \Delta 1 = 1 \) and \( \Delta x = 0 \) otherwise.

Notice that the multiplication \( \otimes \) is not commutative. Indeed, for example \( b \otimes d = a \) but \( d \otimes b = 0 \). Moreover, this algebra is also non-residuated since, e.g., \( c = d \otimes d \leq c \), but \( d \nleq d \rightarrow c = c \).
It is easy to verify the following properties of \( \text{EQ}_\Delta \)-algebra:

**Lemma 3.4.** Let \( \mathcal{E}_\Delta \) be an \( \text{EQ}_\Delta \)-algebra. For all \( a, b \in E \) it holds that

(a) \( \Delta a \leq a \),
(b) If \( a \leq b \) then \( \Delta a \leq \Delta b \),
(c) \( \Delta (a \rightarrow b) \leq \Delta a \rightarrow \Delta b \).

### 3.2. Lattice \( \text{EQ}_\Delta \)-algebra.

If \( \text{EQ} \)-algebra is a lattice then we must add also axioms for the join operation.

**Definition 3.5.** A lattice \( \text{EQ}_\Delta \)-algebra (\( \text{lEQ}_\Delta \)-algebra) is an algebra \( E \Delta = \langle E, \land, \lor, \otimes, \sim, 0, 1 \rangle \) where \( \langle E, \land, \lor, \otimes, \sim, 0, 1 \rangle \) is a good non-commutative and bounded \( \text{lEQ} \)-algebra (0 and 1 are bottom top elements, respectively) and the \( \Delta \)-operation fulfills axioms (E\( \Delta 1 \))–(E\( \Delta 7 \)) plus

(E\( \Delta 8 \)) \( \Delta (a \lor b) \leq \Delta a \lor \Delta b \),
(E\( \Delta 9 \)) \( \Delta a \lor \sim \Delta a = 1 \).

Note that by axiom (E\( \Delta 9 \)), \( \Delta \) provides only two truth values. The following lemma shows properties of \( \lor \) which help us to prove the substitution property mentioned in Lemma 3.7.

**Lemma 3.6.** Let \( \mathcal{E}_\Delta \) be a prelinear \( \text{lEQ}_\Delta \)-algebra. For all \( a, b, c \in E \) it holds that

(a) \( a \lor (b \land c) = (a \lor b) \land (a \lor c) \),
(b) \( a \sim b \leq (a \lor c) \sim (b \lor c) \).

**Proof.** (a) See Lemma 8 in [4]. (b) is proved in Proposition 4 in [5].

**Lemma 3.7.** Let \( \mathcal{E}_\Delta \) be a prelinear \( \text{lEQ} \)-algebra. Then substitution axioms (E4) and (E8) are equivalent to the following one:

(E9) \( (((a \land b) \lor c) \sim d) \otimes (f \sim e) \otimes (e \sim a) \leq d \sim (f \lor (b \land e)) \).

**Proof.** By (E8) and Lemma 3.6(a) we have \( (((a \land b) \lor c) \sim d) \otimes (f \sim c) = ((c \lor (a \land b)) \sim d) \otimes (f \sim c) \leq (f \lor (a \land b)) \sim d = ((f \lor a) \land (f \lor b)) \sim d. \) Thus using isotonicity of \( \otimes \), Lemma 3.6(b), (E4) and again Lemma 3.6(a) we get
EQ-logics with Delta Connective

49

(((a ∧ b) ∨ c) ∼ d) ⊗ (f ∼ c)) ⊗ (e ∼ a) ≤ (((f ∨ a) ∧ (f ∨ b)) ∼ d) ⊗ (e ∼ a) ≤ (((f ∨ a) ∧ (f ∨ b)) ∼ d) ⊗ ((f ∨ e) ∼ (f ∨ a)) ≤ d ∼ ((f ∨ e) ∧ (f ∨ b)) = d ∼ (f ∨ (e ∧ b)).

Conversely assume (E9). Firstly, we prove (E4): (((a ∧ b) ∼ c) ⊗ (d ∼ a) = (((a ∧ b) ∨ 0) ∼ c) ⊗ (d ∼ a)) ≤ c ∼ (0 ∨ (b ∧ d)) = c ∼ (d ∧ b). Now we prove (E8): ((a ∨ b) ∼ c) ⊗ (d ∼ a) = (((a ∧ 1) ∨ b) ∼ c) ⊗ (b ∼ b)) ⊗ (d ∼ a) ≤ c ∼ (b ∨ (1 ∧ d)) = (d ∨ b) ∼ c.

This lemma suggests a slight simplification: in ℓEQ-algebras we may consider only axiom (E9) instead of two axioms (E4) and (E8).

Figure 2. Eight Element ℓEQΔ-algebra

The following example demonstrates how the operation Δ can be introduced in non-linearly ordered EQ-algebras.

Example 3.8. Consider an example of a finite non-trivial EQΔ-algebra whose lattice structure is depicted in the Figure 2. The multiplication and fuzzy equality are defined as follows:

<table>
<thead>
<tr>
<th>⊗</th>
<th>0</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>a</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>a</td>
<td>0</td>
<td>a</td>
<td>a</td>
<td>a</td>
</tr>
<tr>
<td>b</td>
<td>0</td>
<td>0</td>
<td>b</td>
<td>0</td>
<td>b</td>
<td>0</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>0</td>
<td>0</td>
<td>c</td>
<td>0</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
</tr>
<tr>
<td>d</td>
<td>0</td>
<td>0</td>
<td>b</td>
<td>a</td>
<td>b</td>
<td>a</td>
<td>d</td>
<td>d</td>
</tr>
<tr>
<td>e</td>
<td>0</td>
<td>a</td>
<td>0</td>
<td>c</td>
<td>a</td>
<td>e</td>
<td>c</td>
<td>e</td>
</tr>
<tr>
<td>f</td>
<td>0</td>
<td>b</td>
<td>c</td>
<td>b</td>
<td>c</td>
<td>f</td>
<td>f</td>
<td>f</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
<td>e</td>
<td>f</td>
<td>1</td>
</tr>
</tbody>
</table>
Using (2) we obtain the implication:
\[
\rightarrow 0 \ a \ b \ c \ d \ e \ f \ 1
\]

The Δ-operation is defined by
\[
\Delta x = \begin{cases} 
1 & \text{if } x = 1 \\
e & \text{if } x = e \\
b & \text{if } x = b, d, f \\
0 & \text{if } x = 0, a, c
\end{cases}
\]

Note that the multiplication ⊗ is not commutative. Moreover, this algebra is non-residuated since, e.g., 0 = a ⊗ d ≤ 0, but a ≰ d → 0 = 0. Notice also that the algebra is prelinear (unlike the algebra from Example 3.3), since it satisfies the assumptions of Theorem 2.5.

The following theorem characterizes a representable class of ℓEQΔ-algebras.

**Theorem 3.9.** Let \( \mathcal{E}_\Delta \) be ℓEQΔ-algebra. Then the following properties are equivalent:

(a) \( \mathcal{E}_\Delta \) is subdirectly embeddable into a product of linearly ordered ℓEQΔ-algebras (i.e., \( \mathcal{E}_\Delta \) is representable).

(b) The equality (5) in \( \mathcal{E}_\Delta \) is satisfied for all \( a, b, c, d \in E \).

**Proof.** This theorem was proved in [5] for good EQΔ-algebras. We will therefore follow this proof and focus on different points only. Obviously, if \( \mathcal{E}_\Delta \) is representable then it satisfies condition (5) (since either \( x \rightarrow y = 1 \) or \( y \rightarrow x = 1 \) holds for all \( x, y \) in a linearly ordered EQΔ-algebra). Conversely, due to Theorem 2.5, \( \mathcal{E}_\Delta \) is prelinear. Then, the proof proceeds in the same way as that of Theorem 11 in [5].
It is only sufficient to verify that axioms of $\ell\text{EQ}_\Delta$-algebra hold also in the factor algebra $\overline{\mathcal{E}}_\Delta/\cong_F$ where $\cong_F$ is a congruence defined for a given filter $F$ as $a \cong_F b$ iff $a \sim b \in F$.

We will demonstrate this, e.g. for $(E\Delta 6)$. Then $\Delta([a]_F \sim [b]_F) = [\Delta(a \sim b)]_F \leq [(a \otimes c) \sim (b \otimes c)]_F = ([a]_F \otimes [c]_F) \sim ([b]_F \otimes [c]_F)$ because $[\Delta(a \sim b)]_F \leq [(a \otimes c) \sim (b \otimes c)]_F$ iff $\Delta(a \sim b) \to ((a \otimes c) \sim (b \otimes c)) = 1 \in F$ which holds because $(E\Delta 6)$ holds in $\mathcal{E}_\Delta$. □

3.3. Basic $\text{EQ}_\Delta$-logic. Let $J$ be a language of $\text{EQ}_\Delta$-logic, $F_J$ the set of all formulas in the language $J$ and $E_\Delta = \langle E, \land, \otimes, \sim, \Delta, 1 \rangle$ be a good non-commutative $\text{EQ}_\Delta$-algebra. The language of $\text{EQ}_\Delta$-logic is the language of basic $\text{EQ}$-logic expanded by the unary connective $\Delta$.

A truth evaluation of formulas is a mapping $e : F_J \to E$ defined as follows: if $A$ is a propositional variable $p$ then $e(p) \in E$. Otherwise,

$$
eq (\top) = 1,$$
$$e(A \land B) = e(A) \land e(B),$$
$$e(A \& B) = e(A) \otimes e(B),$$
$$e(A \equiv B) = e(A) \sim e(B),$$
$$e(\Delta A) = \Delta e(A)$$

for all formulas $A, B \in F_J$.

3.3.1. Logical axioms and inference rules. Now we are ready to introduce axioms and inference rules of the $\text{EQ}_\Delta$-logic. Clearly, they copy axioms of he underlying $\text{EQ}_\Delta$-algebras but not exactly.

**Definition 3.10.** Axioms of $\text{EQ}_\Delta$-logic are those of the basic $\text{EQ}$-logic and the following ones:

$(\text{EQ}\Delta 1)$ $\Delta A \Rightarrow \Delta \Delta A$

$(\text{EQ}\Delta 2)$ $\Delta (A \equiv B) \Rightarrow (\Delta A \equiv \Delta B)$

$(\text{EQ}\Delta 3)$ $\Delta (A \land B) \equiv (\Delta A \land \Delta B)$

$(\text{EQ}\Delta 4)$ $\Delta A \equiv (\Delta A \& \Delta A)$

$(\text{EQ}\Delta 5)$ $\Delta (A \equiv B) \Rightarrow ((A \& C) \equiv (B \& C))$

$(\text{EQ}\Delta 6)$ $\Delta (A \equiv B) \Rightarrow ((C \& A) \equiv (C \& B))$

Deduction rules of $\text{EQ}_\Delta$-logic are *equanimity rule*, *Leibniz rule* and the *necessitation rule*

$$(N) \quad \frac{A}{\Delta A}.$$ 

3.3.2. Main properties. Now we introduce main properties of $\text{EQ}_\Delta$-logic, including completeness theorem.

**Lemma 3.11.** All axioms of $\text{EQ}_\Delta$-logic are tautologies.

**Proof.** This is straightforward using the axioms and properties of $\text{EQ}_\Delta$-algebra. □
Lemma 3.12. The deductive rules of EQΔ-logic are sound in the following sense. Let e : FJ → E be a truth evaluation:
(a) If e(A) = 1 and e(A ⇒ B) = 1 then e(B) = 1.
(b) If e(B ⊃ C) = 1 then e(A[p := a] ⇒ A[p := b]) = 1 for any formula A.
(c) If e(A) = 1 then e(ΔA) = 1.

Proof. By the straightforward verification. □

The following formulas are provable in EQΔ-logic.

Lemma 3.13. (a) ⊢ ΔA ⇒ A,
(b) ⊢ ΔT ∨ T,
(c) ⊢ (ΔA ⇒ B) ⇒ (ΔA ⇒ ΔB),
(d) ⊢ (ΔA & B) ⇒ (ΔA & ΔB),
(e) ⊢ ΔA ⇔ Δ(A & A),
(f) ⊢ Δ((A ⇒ B) & (C ⇒ D)) ⇒ ((A & C) ⇒ (B & D)).

Proof. The proofs proceed in equational style.
(a) $Δ(A ≜ T) ⇒ ((A & T) ≜ (T & T))$ (EQΔ5)

(b) $(ΔA ≜ T) ⇒ (A ≜ (T & T))$

(c) $(ΔA ≜ T) ⇒ (A ≜ (T & T))$

(d) By Lemma 2.6(r) and 2.6(s) using rule (N), Lemma 3.13(c) and Lemma 2.6(e) we get ⊢ Δ(A & B) ⇒ ΔA and ⊢ Δ(A & B) ⇒ ΔB, respectively. Next, we use these formulas as assumptions in Lemma 2.6(m) to obtain ⊢ (Δ(A & B) & Δ(A & B)) ⇒ (ΔA & ΔB) and therefore ⊢ Δ(A & B) ⇒ (ΔA & ΔB) (see (EQΔ4)).

(e) From (EQΔ5) ⊢ Δ(T ≜ A) ⇒ ((T & A) ≜ (A & A)) and thus ⊢ ΔA ⇒ (A ≜ (A & A)). By Lemma 2.6(j) and applying transitivity of implication ⊢ ΔA ⇒ (A ⇒ (A & A)). By rule (N), double using of Lemma 3.13(c), Lemma 2.6(c) and transitivity of implication observe ⊢ ΔA ⇒ (ΔA ⇒ (A & A)) and thus ⊢ ΔA ⇒ (ΔA ⇒ Δ(A & A)) (use (EQΔ4) and transitivity of implication again). Then, from Lemma 2.6(c) ⊢ (ΔA & ΔA) ⇒ Δ(A & A) and finally, (EQΔ4) gives ⊢ ΔA ⇒ Δ(A & A).

We prove the converse implication as follows:

$Δ(A & A) ⇒ (ΔA & ΔA)$ (Lemma 3.13(d))

$Δ(A & A) ⇒ (ΔA & ΔA)$ (Lemma 3.13(d))
\(\Delta(A \& A) \Rightarrow \Delta A\)

put together Both implications by Lemma 2.6(h) and using Lemmas 2.6(l) and 2.6(e) we get the desired formula.

(f) First using Lemma 3.13(d) we have

\[\vdash \Delta((A \equiv B) \& (C \equiv D)) \Rightarrow (\Delta(A \equiv B) \& \Delta(C \equiv D)).\]

Then using assumptions (EQ\(\Delta\)5) and \(\vdash \Delta(C \equiv D) \Rightarrow \Delta(C \equiv D)\) in Lemma 2.6(m) we obtain

\[\vdash (\Delta(A \equiv B) \& \Delta(C \equiv D)) \Rightarrow (((A \& C) \equiv (B \& C)) \& \Delta(C \equiv D)).\]

In the same way (but using (EQ\(\Delta\)6)) we get

\[\vdash (((A \& C) \equiv (B \& C)) \& \Delta(C \equiv D)) \Rightarrow ((A \& C) \equiv (B \& D)).\]

Finally from the previous formulas and Lemma 2.6(o) in the form

\[\vdash ((A \& C) \equiv (B \& D)) \Rightarrow ((A \& C) \equiv (B \& D)).\]

using Lemma 2.6(g) we find desired formula.

The following is \(\Delta\)-deduction theorem. Surprisingly, it can be formulated in the same way as in residuated logics.

**Theorem 3.14 (Deduction theorem).** For each theory \(T\) and formulas \(A, B\) it holds that

\[T \cup \{A\} \vdash B \iff T \vdash \Delta A \Rightarrow B.\]

**Proof.** Let \(T \cup \{A\} \vdash B\). The proof proceeds by induction on the length of the proof of \(B\).

(a) If \(B := A, B \in T\) or \(B\) is a logical axiom then it follows from Lemma 3.13(a), Lemma 2.6(n) and Lemma 2.6(e).

(b) Let \(B\) is obtained using rule (EA) by the proof

\[\ldots, C, C \equiv B, B.\]

Then we proceed as follows:

(L.1) \(T \vdash \Delta A \Rightarrow (\Delta A \& \Delta A)\) \hspace{1cm} ((EQ\(\Delta\)4), Lemma 2.6(j), (e))

(L.2) \(T \vdash (\Delta A \& \Delta A) \Rightarrow (C \& (C \equiv B))\) \hspace{1cm} (2x inductive assumption, Lemma 2.6(m))

(L.3) \(T \vdash (C \& (C \equiv B)) \Rightarrow B\) \hspace{1cm} (Lemma 2.6(p))

(L.4) \(T \vdash \Delta A \Rightarrow B\) \hspace{1cm} (L.1, L.2, L.3 and Lemma 2.6(g))

(c) Let \(B := D[p := E] \equiv D[p := F]\) have been obtained using rule (Leib) by the proof

\[\ldots, E \equiv F, D[p := E] \equiv D[p := F].\]

Then the proof proceeds by induction on the complexity of the formula \(D\).

(i) If \(D\) is either \(\top\) or \(q\) (other than \(p\)) then \(D[p := E] \equiv D[p := F] = D \equiv D\) and \(T \vdash \Delta A \Rightarrow (D \equiv D)\) follows from Lemma 2.6(c), Lemma 2.6(n) and Lemma 2.6(e).
(ii) If \( D \) is \( p \) then it follows immediately from the inductive assumption.

(iii) Let \( D \) be \( G \land H \), where \( \land \in \{ \land, \&\&\&, \equiv \} \). Then we must prove
\[
T \vdash \Delta A \Rightarrow ((G \land H)[p := E] \equiv (G \land H)[p := F]),
\]
that is
\[
T \vdash \Delta A \Rightarrow ((G' \land H') \equiv (G'' \land H'')) \quad (7)
\]

In case that \( \land \in \{ \land, \equiv \} \), then (7) can be proved by the following steps:

(L.1) \( T \vdash \Delta A \Rightarrow (\Delta A \& \Delta A) \) \((\text{EQ} \Delta 4), \text{Lemma } 2.6(j), (e))

(L.2) \( T \vdash (\Delta A \& \Delta A) \Rightarrow ((G' \equiv G'') \& (H' \equiv H'')) \)
\((2x \text{ inductive assumption, Lemma } 2.6(m))

(L.3) \( T \vdash ((G' \equiv G'') \& (H' \equiv H'')) \Rightarrow ((G' \land H') \equiv (G'' \land H'')) \)
\((\text{Lemma } 2.7(g), \text{Lemma } 2.6(k))

(L.4) \( T \vdash \Delta A \Rightarrow ((G' \land H') \equiv (G'' \land H'')) \)
\((\text{L.1, L.2, L.3 and Lemma } 2.6(g))

In case that \( \land \) is \&\&\& then the proof of (7) proceeds as follows:

(L.1) \( T \vdash \Delta A \Rightarrow ((G' \equiv G'') \& (H' \equiv H'')) \) \((\text{see above})

(L.2) \( T \vdash \Delta A \Rightarrow \Delta (\Delta A) \)
\((\text{EQ} \Delta 1))

(L.3) \( T \vdash \Delta (\Delta A) \Rightarrow \Delta ((G' \equiv G'') \& (H' \equiv H'')) \)
\((\text{L.1, rule } \text{(N), Lemma } 3.13(c), \text{Lemma } 2.6(e))

(L.4) \( T \vdash \Delta ((G' \equiv G'') \& (H' \equiv H'')) \Rightarrow ((G' \& H') \equiv (G'' \& H'')) \)
\((\text{Lemma } 3.13(f))

(L.5) \( T \vdash \Delta A \Rightarrow ((G' \& H') \equiv (G'' \& H'')) \)
\((\text{L.2, L.3, L.4 and Lemma } 2.6(g))

(iv) Let \( D \) be \( \Delta H \). Then we have

(L.1) \( T \vdash \Delta A \Rightarrow (H' \equiv H'') \)
\((\text{inductive assumption})

(L.2) \( T \vdash \Delta (\Delta A) \Rightarrow \Delta (H' \equiv H'') \)
\((\text{L.1, rule } \text{(N), Lemma } 3.13(c), \text{Lemma } 2.6(e))

(L.3) \( T \vdash \Delta (H' \equiv H'') \Rightarrow (\text{EQ} \Delta 1), \text{L.2 and Lemma } 2.6(g))

(L.4) \( T \vdash \Delta (H' \equiv H'') \Rightarrow (\text{EQ} \Delta 2) \) and Lemma 2.6(g)

(d) Let \( B := \Delta C \) have been obtained using rule (N) by the proof
\[
\ldots, C, \Delta C.
\]

Then \( T \vdash \Delta A \Rightarrow \Delta C \) follows from inductive assumption using rule (N), Lemma 3.13(c), 2.6(e), (EQ\Delta 1) and Lemma 2.6(g).

The converse implication follows using rule (N) and Lemma 2.6(e). \( \square \)

**Definition 3.15.** Put
\[
A \equiv B \quad \text{iff} \quad \vdash A \equiv B, \quad A, B \in F_J.
\]

It follows from Lemmas 2.6(c), (i) and (o) that \( \equiv \) is an equivalence on \( F_J \). Let us denote by \([A]\) an equivalence class of \( A \) and put \( \bar{E} = \{ [A] \mid A \in F_J \} \). Finally we
define

\[ 1 = [\top], \]
\[ [A] \land [B] = [A \land B], \]
\[ [A] \otimes [B] = [A \& B], \]
\[ [A] \sim [B] = [A \equiv B], \]
\[ \Delta[A] = [\Delta A]. \]

**Lemma 3.16.** The algebra \( \mathcal{E}_\Delta = \langle E, \land, \otimes, \sim, \Delta, 1 \rangle \) is a good non-commutative \( EQ_\Delta \)-algebra.

**Proof.** We have to verify only axioms of \( EQ_\Delta \)-algebra: for axioms (E1)–(E7) and the “goodness property” see the proof of Lemma 10 in [3]. For axioms (E∆1)–(E∆7) see Lemma 3.13(b) and (EQ∆∆∆1)–(EQ∆∆∆6). \( \Box \)

**Theorem 3.17** (Soundness). The basic \( EQ_\Delta \)-logic is sound.

**Proof.** This follows from Lemmas 3.11 and 3.12. \( \Box \)

**Theorem 3.18** (Completeness). The following is equivalent for every formula \( A \):

(a) \( \vdash A \),

(b) \( e(A) = 1 \) for every good non-commutative \( EQ_\Delta \)-algebra \( \mathcal{E}_\Delta \) and a truth evaluation \( e : F \rightarrow E \).

**Proof.** The implication \( (a) \Rightarrow (b) \) is soundness.

\( (b) \Rightarrow (a) \): By Lemma 3.16 the algebra \( \mathcal{E}_\Delta \) of equivalence classes of formulas is a good non-commutative \( EQ_\Delta \)-algebra. Thus, if \( (b) \) holds then it holds also for \( e : F \rightarrow E \). If \( e(A) = 1 \) then it means that \( [A] = [\top] \), i.e. \( \vdash A \equiv \top \) and so, \( \vdash A \) by rule (T1). \( \Box \)

3.4. **Prelinear \( EQ_\Delta \)-logic.** This logic is interesting because stronger form of the completeness theorem holds in it. The language of this logic is that of the basic \( EQ_\Delta \)-logic extended by the logical connective \( \lor \) and logical constant \( \bot \) (falsum). We also extend the language by the short

\( \neg A := A \equiv \bot. \) \hspace{1cm} (8)

Formula (8) is the definition of *negation* in this logic.

The complete list of logical axioms of the prelinear \( EQ_\Delta \)-logic is the following:

- (EQ1) \( (A \equiv \top) \equiv A \)
- (EQ2) \( A \land B \equiv B \land A \)
- (EQ3) \( (A \lor B) \lor C \equiv A \lor (B \lor C), \) \( \lor \in \{\land, \&\} \)
- (EQ4) \( A \land A \equiv A \)
- (EQ5) \( A \land \top \equiv A \)
- (EQ6) \( A \& \top \equiv A \)
- (EQ7) \( \top \& A \equiv A \)
- (EQ8a) \( ((A \land B) \& C) \Rightarrow (B \& C) \)
- (EQ8b) \( (C \& (A \land B)) \Rightarrow (C \& B) \)
- (EQ9) \( (((A \land B) \lor C) \equiv D) \& (F \equiv C)) \& (E \equiv A) \Rightarrow (D \equiv (F \lor (B \land E))) \)
Remark 3.19. Axioms of the basic EQΔ-logic are extended by axioms (EQ12)–(EQ14) and (EQΔ7)–(EQΔ8) which reflect the join-semilattice structure. Moreover, axiom (EQ15) stands for the prelinearity. Note also that axiom (EQ9) expresses a common substitution axiom both for ∧ and for ∨ and thus it replaces the original substitution axiom in EQ-logics.

Semantics of this logic is formed by a non-commutative ℓEQΔ-algebra in which condition (5) is fulfilled.

The main properties of the prelinear EQΔ-logic, with the emphasis on disjunction connective are introduced in the following lemma. We will also give a proof of the substitution of ∧ connective are introduced in the following lemma. We will also give a proof of the condition (5) is fulfilled.

Lemma 3.20.
(a) ⊢ A ∨ B ≡ B ∨ A,
(b) ⊢ ((A ∨ B) ≡ C) & (D ≡ A) ⇒ ((D ∨ B) ≡ C),
(c) ⊢ A ∨ ⊥ ≡ A,
(d) ⊢ ((A ∧ B) ≡ C) & (D ≡ A) ⇒ (C ≡ (D ∧ B)),
(e) ⊢ (A ⇒ B) ∨ (B ⇒ A),
(f) ⊢ ⊤ ∨ A,
(g) ⊢ A ⇒ (A ∨ B),
(h) ⊢ A ∨ (A ∨ B) ≡ A,
(i) ((A ∨ B) ≡ B) ⇒ (A ⇒ B),
(j) ((A ⇒ C) & (B ⇒ C) ⇒ ((A ∨ B) ⇒ C),
(k) ((A ∨ B) ⇒ C) ⇒ (A ⇒ C),
(l) (A ⇒ B) ⇒ (A ⇒ (B ∨ C)),
(m) (A ∨ B) ≡ (((A ⇒ B) ⇒ B) ∧ ((B ⇒ A) ⇒ A)),
(n) (A ≡ B) ⇒ ((A ∨ C) ≡ (B ∨ C)),
(o) (A ⇒ B) ⇒ ((A ∨ C) ⇒ (B ∨ C)),
(p) (A ⇒ B) & (C ⇒ D) ⇒ ((A ∨ C) ⇒ (B ∨ D)).
Proof. (a) By (EQ9) we get (((((A ∧ ⊤) ∨ B) ≡ ⊤) & (B ≡ B)) & (A ≡ A) ⇒ (⊤ ≡ (B ∨ (⊤ ∧ A)))). Using (Leib), (EQ5), (EQ1), Lemma 2.6(c) + rule (T2), (EQ6), (EQ2) and Lemma 2.6(i) we get ⊢ (A ∨ B) ⇒ (B ∨ A). In the same way we obtain ⊢ (B ∨ A) ⇒ (A ∨ B). Finally using Lemma 2.6(h), (l) and (e) we have ⊢ (A ∨ B) ≡ (B ∨ A).

(b) From (EQ9) we have (((((A ∧ ⊤) ∨ B) ≡ C) & (D ≡ A) ⇒ (C ≡ (B ∨ ((⊤ ∧ D)))). Using (Leib), (EQ5), Lemma 2.6(c) + rule (T2), (EQ6), (EQ2), Lemma 2.6(a) and Lemma 2.6(i) we obtain (b).

c) A ⊥
⇔ ((Leib) + (EQ14) + Lemma 2.6(i); “C-part”: A p)
A ∨ (A ∧ ⊥)
⇔ ((EQ13))
A

d) From (EQ9) we have (((((A ∧ B) ∨ ⊥) ≡ C) & (⊥ ≡ ⊥)) & (D ≡ A) ⇒ (C ≡ (⊥ ∨ (B ∧ D)))). Using (Leib), Lemma 3.20(c), Lemma 2.6(c) + rule (T2), (EQ6), Lemma 3.20(a) and (EQ2) we get (d).

e) Directly from (EQ15) in the form (A ⇒ B) ∨ (⊤ ⇒ (⊤ & (⊤ ⇒ ((B ⇒ A) & ⊤)))) using (Leib), (EQ6), (EQ7) and Lemma 2.6(f).
(f) (A ⇒ ⊤) ∨ (⊤ ⇒ A) (Lemma 3.20(e))
⇔ ((Leib) + Lemma 2.6(f); “C-part”: A ⊤ p)
A ⇒ ⊤ ∨ A
⇔ ((Leib) + (EQ5): (A ∧ ⊤) ≡ A + rule (T2); “C-part”: p ∨ A)
⊤ ∨ A

g) Just from Lemma 3.20(b) ⊢ (((⊤ ∨ B) ≡ ⊤) & (A ≡ ⊤) ⇒ ((A ∨ B) ≡ ⊤) using Lemma 3.20(f) + rule (T2), Lemma 2.6(c) + rule (T2), (EQ7) and (EQ1).
(h) Immediately from implication (gin Lemma 3.20).
(i) The equivalence will be obtained when proving the implications left to right and vice versa. From Lemma 3.20(d) in the form

⊢ (((A ∨ B) ∧ A) ≡ A) & (B ≡ (A ∨ B)) ⇒ (A ≡ (B ∧ A))

and by clear modifications using the Leibniz rule we have ⊢ ((A ∨ B) ≡ B) ⇒ (A ⇒ B). The converse implication follows in the same way from Lemma 3.20(b) in the form

⊢ (((A ∧ B) ∨ B) ≡ B) & (A ≡ (A ∧ B)) ⇒ ((A ∨ B) ≡ B).

Both implications, Lemma 2.6(h), (l) and (e) complete the proof.
(j) We start with Lemma 3.20(b) in the form

⊢ ((C ∨ A) ≡ C) & ((B ∨ C) ≡ C) ⇒ ((B ∨ C) ∨ A) ≡ C).

Using Leibniz rule and Lemma 3.20(a), (EQ12) and Lemma 3.20(i) we finish the proof.
(k) Follows immediately from Lemma 2.6(d) in the form

⊢ ((A ∨ B) ⇒ C) ⇒ (((A ∨ B) ∧ A) ⇒ C)
using (Leib), (EQ2) and Lemma 3.20(h).

   (l) Immediately from (EQ11) in the form

   \[ \vdash (A \Rightarrow ((B \lor C) \land B)) \Rightarrow (A \Rightarrow (B \lor C)) \]

   using (Leib), (EQ2) and Lemma 3.20(h).

   (m) We prove two implications which yield equivalence using standard technique. Denote \( C := ((A \Rightarrow B) \Rightarrow B) \land ((B \Rightarrow A) \Rightarrow A) \). From Lemma 2.6(v) and also 2.6(u) in the form \( \vdash (A \Rightarrow ((B \Rightarrow A) \Rightarrow A)) \) using Lemma 2.6(h), 2.6(u) and 2.6(e) we have \( \vdash A \Rightarrow C \). By a similar way we get \( \vdash B \Rightarrow C \). Thus, using Lemma 2.6(h), 3.20(j) and 2.6(e) we obtain

   \[ (A \lor B) \Rightarrow (((A \Rightarrow B) \Rightarrow B) \land ((B \Rightarrow A) \Rightarrow A)). \]

   Now, we show the converse implication. Let us consider the following formulas:

   \( \vdash (A \Rightarrow B) \Rightarrow (((A \Rightarrow B) \Rightarrow B) \Rightarrow B) \) \hspace{1cm} (Lemma 2.6(v))
   \( \vdash ((((A \Rightarrow B) \Rightarrow B) \Rightarrow B) \Rightarrow (C \Rightarrow B)) \) \hspace{1cm} (Lemma 2.6(d))
   \( \vdash (C \Rightarrow B) \Rightarrow (C \Rightarrow (A \lor B)) \) \hspace{1cm} (Lemma 3.20(l))

   The transitivity of \( \Rightarrow \) results in \( \vdash (A \Rightarrow B) \Rightarrow (C \Rightarrow (A \lor B)) \). Similarly, we get \( \vdash (B \Rightarrow A) \Rightarrow (C \Rightarrow (A \lor B)) \). Using Lemma 2.6(h), 3.20(j) and 2.6(e) we have

   \( \vdash ((A \Rightarrow B) \lor (B \Rightarrow A)) \Rightarrow (C \Rightarrow (A \lor B)) \)

   and therefore, using Lemma 3.20(e) and 2.6(e),

   \( \vdash (((A \Rightarrow B) \Rightarrow B) \land ((B \Rightarrow A) \Rightarrow A)) \Rightarrow (A \lor B) \).

   (n) Let us consider the following variant of Lemma 3.20(b):

   \( \vdash ((A \lor C) \equiv (A \lor C)) \land (B \equiv A) \Rightarrow ((B \lor C) \equiv (A \lor C)) \).

   Using (Leib), Lemma 2.6(c) + rule (T2), (EQ7) Lemma 2.6(i) we obtain (n).

   (o) From (n) we have \( \vdash ((A \lor B) \equiv B) \Rightarrow (((A \lor B) \lor C) \equiv (B \lor C)) \). By (Leib), (i), (a) and (EQ12) we get \( \vdash (A \Rightarrow B) \Rightarrow (((A \lor C) \lor B) \equiv (B \lor C)) \) and thus by Lemma 2.6(j) and transitivity of \( \Rightarrow \) \( \vdash (A \Rightarrow B) \Rightarrow (((A \lor C) \lor B) \Rightarrow (B \lor C)) \). From this and \( \vdash (((A \lor C) \lor B) \Rightarrow (B \lor C)) \Rightarrow (((A \lor C) \Rightarrow (B \lor C)) \) (i.e. (k)) using transitivity of \( \Rightarrow \) again we conclude (o).

   (p) Using (o) and (a) we obtain \( \vdash (A \Rightarrow B) \Rightarrow ((A \lor C) \Rightarrow (B \lor C)) \) and

   \( \vdash (C \Rightarrow D) \Rightarrow ((A \lor C) \Rightarrow (B \lor C))) \& ((B \lor C) \Rightarrow (B \lor D)). \)

   Finally, using Lemma 2.6(q) and transitivity of \( \Rightarrow \) we complete the proof. \( \Box \)

The following lemma shows properties of the delta connective in the prelinear EQAΔ-logic.

**Lemma 3.21.**

(a) \( \vdash \Delta \bot \equiv \bot \),
(b) \( \vdash \Delta (\neg A) \Rightarrow \neg (\Delta A) \),
(c) \( \vdash \Delta (A \lor B) \equiv (\Delta A \lor \Delta B) \),
(d) \( \vdash \Delta (A \Rightarrow B) \lor \Delta (B \Rightarrow A) \),
(e) \( \vdash (\Delta A \& \Delta (A \Rightarrow B)) \Rightarrow \Delta B \).
Proof. (a) $\Delta \bot \Rightarrow \bot$ follows from Lemma 3.13(a). If we use (EQ14) and rewrite it in the implication form, we obtain the converse implication. Both implications together with Lemma 2.6(h), 2.6(l) and 2.6(e) yield the equivalence.

(b) $\Delta(A \equiv \bot) \Rightarrow (\Delta(A \equiv p))$

$\Delta(A \equiv \bot) \Rightarrow (\Delta(A \equiv \bot))$

$\Rightarrow (\Delta(A \equiv \bot))$

$\Delta(A \equiv \bot)$

(c) The implication left to right is just (EQ7). The proof of the converse implication proceeds as follows: From Lemma 3.20(g), rule (N), Lemma 3.13(c) and 2.6(e) we obtain $\vdash \Delta(A \equiv \bot)$ and $\vdash \Delta(A \equiv p)$. Both formulas are connected by &&& using Lemma 2.6(h) and finally, we complete it using Lemma 3.20(j) and 2.6(e).

(d) Follows from Lemma 3.20(e) using rule (N), (EQ7) and Lemma 2.6(e).

(e) This can be found from Lemma 3.13(c) using Lemma 2.7(i) and 2.6(t).

□

Remark 3.22. Similarly as in Definition 3.15, we construct Lindenbaum algebra of equivalence classes of formulas. In addition to it, we define

0 = [⊥],

$[A] \lor [B] = [A \lor B]$.

Lemma 3.23. The algebra $\mathcal{E}_\Delta = \langle \mathcal{E}, \land, \lor, \otimes, \sim, \Delta, 0, 1 \rangle$ is an $\mathcal{L}EQ\Delta$-algebra satisfying condition (5).

Proof. Axiom (E9) which substitutes axioms (E4) and (E8), follows from (EQ9). For the least element of $\mathcal{E}$ see (EQ14). For join-semilattice structure see (EQ12)–(EQ13) and Lemma 3.20(a). For axioms (E\Delta8) and (E\Delta9) see (EQ\Delta7), (EQ\Delta8) and rule (T2). For condition (5) see axiom (EQ15). The remaining axioms have already proved in Lemma 3.16.

□

Theorem 3.24 (Completeness). For every formula $A \in F_j$ and every theory $T$ the following is equivalent:

(a) $T \vdash A$.

(b) $e(A) = 1$ for every truth evaluation $e : F_j \rightarrow E$ and every linearly ordered, $\mathcal{L}EQ\Delta$-algebra $\mathcal{E}_\Delta$.

(c) $e(A) = 1$ for every truth evaluation $e : F_j \rightarrow E$ and every $\mathcal{L}EQ\Delta$-algebra $\mathcal{E}_\Delta$ satisfying condition (5).

Proof. (a) $\Rightarrow$ (c): All axioms of $T$ are true in all models of $T$ (For axiom (EQ9) see Lemma 3.7; axioms (EQ12)–(EQ14) follow from properties of lattice; axiom (EQ15) is a tautology because of the property (5); to verify (EQ\Delta7) and (EQ\Delta8) observe (E\Delta8) and (E\Delta9); for the remaining axioms see Lemma 3.11).

(c) $\Rightarrow$ (a): If (c) holds then it holds also for $e : F_j \rightarrow \mathcal{E}$, where $\mathcal{E} = \langle \mathcal{E}, \land, \lor, \otimes, \sim, \Delta, 0, 1 \rangle$ is an $\mathcal{L}EQ\Delta$-algebra satisfying (5) (the Lindenbaum algebra). If $e(A) = 1$ then $[A] = [\top]$, i.e. $T \vdash A \equiv \top$ and so, $T \vdash A$ by rule (T1).
(c) ⇒ (b): According to Definition 3.15 and Remark 3.22 we use the Lindenbaum-
Tarski technique to construct the algebra $\bar{E} = \langle \bar{E}, \wedge, \vee, \otimes, \sim, \Delta, 0, 1 \rangle$. This algebra
satisfies (5) (see (EQ15)), hence it is representable by Theorem 3.9. Thus we have
also deduced (b) ⇒ (c).

\[\square\]

4. Conclusion

EQ-logics form a special class of fuzzy logics that contains the basic EQ-logic
that is more fundamental that the MTL-(fuzzy) logic. Recall that the former are
based on the fuzzy equality/equivalence instead of implication. In classical logic
both approaches are equivalent. However, this is not the case in fuzzy logic.

In this paper, we introduce a special subclass of EQ-logics that are extended by
the additional unary connective of $\Delta$. Note that this connective is very useful in
fuzzy logics because it makes it possible to distinguish crisp formulas (formulas that
are always evaluated by the truth value $1$) from the other ones. Moreover, it also
enables to prove the $\Delta$-deduction theorem because in EQ-logics weaker than MTL,
no deduction theorem holds. Besides this theorem we also demonstrated that the
EQ-$\Delta$-logic has many reasonable properties and enjoys completeness.

Let us remark that completeness of the prelinear EQ$_\Delta$-logic follows also from
the general results presented in [2] since this logic is a weakly implicative and even
algebraically implicative. However, the technical details of the proof and proofs of
many interesting properties have to be given.

Acknowledgements. This research was elaborated in the framework of the IT4-
Innovations Centre of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070 sup-
ported by Operational Programme 'Research and Development for Innovations'
funded by Structural Funds of the European Union and state budget of the Czech
Republic.

References

[1] P. Cintula, P. Hájek, R. Horčík, Formal systems of fuzzy logic and their fragments, Annals
tula, P. Hájek, C. Noguera, eds., Handbook of Mathematical Fuzzy Logic - volume 1, Studies
103-207.
[6] F. Esteva and L. Godo, Monoidal t-norm based logic: towards a logic for left-continuous
ment, R. Mesiar (Eds.), Logical, Algebraic, Analytic, and Probabilistic Aspects of Triangular
1993.


M. Dyba*, University of Ostrava, NSC IT4Innovations, 30. dubna 22, 702 00 Ostrava, Czech Republic

E-mail address: Martin.Dyba@osu.cz

V. Novák, University of Ostrava, NSC IT4Innovations, 30. dubna 22, 702 00 Ostrava, Czech Republic

E-mail address: Vilem.Novak@osu.cz

*Corresponding author