Fuzzy Type Theory, Descriptions, and Partial Functions

Vilém Novák

University of Ostrava, IRAFM, 30. dubna 22. 701 03 Ostrava 1, Czech Republic

Abstract

This paper studies the possibility to deal with partial functions in fuzzy type theory. Among various ways how to represent them we chose introduction of a special value “undefined” laying outside the corresponding domain. In FTT, we can quite naturally utilize the description operator by extending its action also to subnormal fuzzy sets.

Keywords: Fuzzy type theory, EQ-algebra, partial function, description operator.

1. Introduction

When applying type theory to the logical analysis of natural language (cf. [1]), one may meet the requirement to deal with partial functions. Several problems arise when trying to include them in the formalism of type theory. One of the main ones is failure of the \( \lambda \)-conversion and so, the power of the resulting theory is significantly reduced. Thus, several ways how this problem can be overcome were proposed. For example, W. Farmer in his paper [2] describes 8 of them. For type theory, we can consider two basic approaches: either to introduce a special functional value “undefined”, or to modify the syntactic system to distinguish between general equality and equality of defined values.

In this paper, we decided for the first option. The problem, though, arises with the type \( o \) of truth values. The reason is that truth values must form an algebra and so, any additional element must fulfill all its properties. The value “undefined” (in the sequel, we will denote it by \( # \)), however, is quite specific and requires exceptional properties making it distinct from any other value of the considered algebra. Consequently, we sooner or later arrive at a discrepancy. We conclude that it is not possible to extend the algebra of truth values by a value representing “undefined”. The remedy is either to develop a specific kind of algebra of truth values, or simply to replace a formula of type \( oo \) by a formula of type \( oo \alpha \) where the former is expected to represent a partial function (i.e. to replace a function by a (fuzzy) relation). Moreover, it is usually accepted by logicians that predicates (i.e., functions whose range are truth values) should always have a truth value (cf. [2]). Consequently, if the argument is “undefined” then the resulting functional value is \textit{false}. Hence, there are no partial functions into truth values.

Introduction of \( # \) into other types distinct from \( o \) is still possible. In fuzzy type theory, there is one specific formula which requires treatment with respect to partiality, namely the description operator \( t_{\alpha(o)} \). In [3, 4], it is considered in such a way that its interpretation is just a partial function giving a value from the kernel of the corresponding fuzzy set if the latter is normal, and giving nothing otherwise. This suggests an idea how \( # \) can be naturally included: the value \( # \) of type \( \alpha \) is defined on the basis of \( t_{\alpha(o)} A_{oo} \) for \( A_{oo} \) representing a subnormal fuzzy set. The problem, however, still remains with the formula \( t_{\alpha(o)} \) since there can also be subnormal fuzzy sets on truth values ans consequently, we face the above mentioned impossibility to extend the algebra of truth values.

There are two other solutions.

(i) We can modify slightly the definition of \( t_{\alpha(o)} \) so that each considered fuzzy set is normalized and so, the value “undefined” in truth values is excluded. Unfortunately, if we use this definition together with the definition of \( # \) using the description operator for the other types, we arrive at contradiction.

(ii) Introduce \( # \) systematically for all types and in case of the type \( o \), identify \( #_\alpha \) with the falsity \( \bot \). The reason for the latter comes from the above mentioned general assumption that partial functions assigning truth values should give falsity when arguments are undefined.

In this paper, we will focus on the option (ii) and prove that it may be consistently used for the definition of partial functions.

2. Fuzzy Type Theory

We will consider the basic FTT with good \( EQ_\Delta \)-algebra of truth values which was introduced in [4]. This is the most general kind of FTT which can be extended to various specific versions including the IMTL-algebra based one originally introduced in [5].

---

\*This work was supported by the Grant IAA108270901 of the GA AV ČR and the Institutional Research Plan MSM 6198898701.
2.1. Truth values

The truth values form a good EQ-algebra (see [6, 4]) extended by the delta operation $\Delta$:

$$\mathcal{E} = \langle E, \land, \lor, \sim, 1, \Delta \rangle,$$

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 the top element 1). We put $a \leq b$ iff $a \land b = a$, as usual.

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

(E3) $a \sim a = 1$ (reflexivity)

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

(E5) $(a \sim b) \lor (c \sim d) \leq (a \sim c) \lor (b \sim d)$ (congruence)

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

(E7) $a \land b \leq a \sim b$ (boundedness)

(E8) $a \sim 1 = a$ (goodness)

The delta operation in $\mathcal{E}$ is an operation $\Delta : E \rightarrow E$ having the following properties:

(i) $\Delta 1 = 1$,

(ii) $\Delta a \leq a$,

(iii) $\Delta a \leq \Delta \Delta a$,

(iv) $\Delta (a \sim b) \leq \Delta a \sim \Delta b$,

(v) $\Delta (a \land b) = \Delta a \land \Delta b$.

If $\mathcal{E}$ is also lattice ordered then, moreover, $\Delta$ must fulfill the following:

(vi) $\Delta (a \lor b) \leq \Delta a \lor \Delta b$,

(vii) $\Delta a \lor \neg \Delta a = 1$.

It should be emphasized that every residuated lattice is a good EQ-algebra with fuzzy equality being the biresiduation (i.e., $a \sim b = (a \rightarrow b) \land (b \rightarrow a)$).

In the EQ-fuzzy type theory we assume that the truth values form a linearly ordered good EQ-$\Delta$-algebra

$$\mathcal{E}_\Delta = \langle E, \land, \lor, \sim, 0, 1, \Delta \rangle$$

where 0 is the bottom element. Therefore, the delta operation reduces to a simple function keeping the truth value 1 and sending all other truth values to 0.

2.2. Syntax

The basic syntactical objects of FTT are classical — see [7], namely the concepts of type and formula. The atomic types are $\epsilon$ (elements) and $o$ (truth values). Complex types ($\beta\alpha$) are formed from previously formed ones $\beta$ and $\alpha$. The set of all types is denoted by $\text{Types}$.

The language of FTT denoted by $J$ consists of variables $x_\alpha, \ldots$. special constants $c_{\alpha, \ldots}$ ($\alpha \in \text{Types}$), auxiliary symbols $\lambda$ and brackets. Formulas are formed of variables, constants (each of specific type), and the symbol $\lambda$. Thus, each formula $A$ is assigned a type (we write $A_\alpha$). A set of formulas of type $\alpha$ is denoted by $\text{Form}_\alpha$, the set of all formulas by $\text{Form}$. Interpretation of a formula $A_{\beta\alpha}$ is a function from the set of objects of type $\alpha$ into the set of objects of type $\beta$. Thus, if $B \in \text{Form}_\beta$ and $A \in \text{Form}_\alpha$ then $(BA) \in \text{Form}_\beta$. Similarly, if $A \in \text{Form}_\beta$ and $x_\alpha \in J$, $\alpha \in \text{Types}$, is a variable then $\lambda x_\alpha A_{\beta\alpha}$ is a formula whose interpretation is a function which assigns to each object of type $\alpha$ an object of type $\beta$ represented by the formula $A_{\beta\alpha}$.

It is specific for type theories that connectives are also formulas. Thus, special formulas (connectives) introduced in FTT are fuzzy equality/equivalence $\equiv$, conjunction $\land$, strong conjunction $\&$, disjunction $\lor$, implication $\Rightarrow$ and the delta connective $\Delta$ which is interpreted by the delta operation.

Specific constants always present in the language of FTT are the following: $E_{(\alpha\alpha)}$, (fuzzy equality), $G_{\alpha\alpha}$ (generating function), $C_{(\alpha\alpha)}$ (conjunction), $S_{(\alpha\alpha)}$ (strong conjunction), $D_{\alpha\alpha}$ (delta).

The fundamental connective in FTT is the fuzzy equality syntactically defined as follows:

(i) $\equiv_{(\alpha\alpha)} := \lambda x_\alpha \lambda y_\alpha (E_{(\alpha\alpha)}x_\alpha y_\alpha x_\alpha y_\alpha)$

(ii) $\equiv_{(\alpha\epsilon)} := \lambda x_\alpha \lambda y_\epsilon (E_{(\alpha\epsilon)}x_\alpha y_\epsilon)(G_{\alpha\epsilon}x_\epsilon)$

(iii) $\equiv_{(\alpha(\beta\alpha))}(\beta\epsilon) := \lambda f_{\beta\alpha} \lambda g_{\beta\alpha} (E_{(\alpha(\beta\alpha))}(\beta\epsilon) g_{\beta\alpha}) f_{\beta\alpha}$

The quantifiers are defined by

$$(\forall x_\alpha)A_\alpha := (\lambda x_\alpha A_\alpha \equiv \lambda x_\alpha \top)$$

$$(\exists x_\alpha)A_\alpha := (\forall y_\alpha)((\forall x_\alpha)A_\alpha \Rightarrow y_\alpha \Rightarrow y_\alpha)$$

$(y_\alpha$ does not occur in $A_\alpha$).

Further details can be found in [5, 4].

2.3. Axioms and inference rules

Fundamental axioms

(FT-fund1) $\Delta(x_\alpha \equiv y_\alpha) \Rightarrow (f_{\beta\alpha} x_\alpha \equiv f_{\beta\alpha} y_\alpha)$

(FT-fund2) $\forall x_\alpha (f_{\beta\alpha} x_\alpha \equiv g_{\beta\alpha} x_\alpha) \Rightarrow (f_{\beta\alpha} \equiv g_{\beta\alpha})$

(FT-fund3) $(f_{\beta\alpha} \equiv g_{\beta\alpha}) \Rightarrow (f_{\beta\alpha} x_\alpha \equiv g_{\beta\alpha} x_\alpha)$,
Axioms of truth values. As usual in fuzzy logic, we have two kinds of conjunction, namely the “ordinary” conjunction \( \& \) and the strong conjunction \( \&\&\& \). Let \( \circ \in \{ \&, \&\&\& \} \).

\[ (\text{FT-tval1}) \quad (x_o \& y_o) \equiv (y_o \& x_o), \]
\[ (\text{FT-tval2}) \quad (x_o \& y_o) \circ z_o \equiv x_o \& (y_o \& z_o), \]
\[ (\text{FT-tval3}) \quad x_o \equiv \top \equiv y_o, \]
\[ (\text{FT-tval4a}) \quad (x_o \& y_o) \equiv x_o, \]
\[ (\text{FT-tval4b}) \quad (\top \& x_o) \equiv x_o, \]
\[ (\text{FT-tval5}) \quad (x_o \& x_o) \equiv x_o, \]
\[ (\text{FT-tval6}) \quad ((x_o \& y_o) \equiv z_o) \& (t_o \equiv x_o) \Rightarrow (z_o \equiv (t_o \& y_o)), \]
\[ (\text{FT-tval7}) \quad (x_o \equiv y_o) \& (z_o \equiv t_o) \Rightarrow (x_o \equiv z_o) \equiv (y_o \equiv t_o), \]
\[ (\text{FT-tval8}) \quad (x_o \Rightarrow (y_o \& z_o)) \Rightarrow (x_o \Rightarrow y_o), \]
\[ (\text{FT-tval9}) \quad (x_o \Rightarrow y_o) \Rightarrow ((x_o \& z_o) \Rightarrow y_o), \]
\[ (\text{FT-tval10a}) \quad \Delta(x_o \Rightarrow y_o) \Rightarrow (x_o \& z_o) \Rightarrow y_o \& z_o, \]
\[ (\text{FT-tval10b}) \quad \Delta(x_o \Rightarrow y_o) \Rightarrow (z_o \& x_o) \Rightarrow z_o \& y_o, \]
\[ (\text{FT-tval11}) \quad ((x_o \Rightarrow y_o) \Rightarrow z_o) \Rightarrow ((y_o \Rightarrow x_o) \Rightarrow z_o) \Rightarrow z_o, \]

\[ Axioms \ of \ delta \]
\[ (\text{FT-delta1}) \quad (g_{oo}(\Delta x_o) \& g_{oo}(\neg\Delta x_o)) \equiv (\forall y_o)g_{oo}(\Delta y_o), \]
\[ (\text{FT-delta2}) \quad \Delta(x_o \& y_o) \equiv \Delta x_o \& \Delta y_o, \]
\[ (\text{FT-delta3}) \quad \Delta(x_o \& y_o) \Rightarrow \Delta x_o \& y_o, \]
\[ (\text{FT-delta4}) \quad \Delta x_o \& \neg\Delta x_o, \]

\[ Axioms \ of \ quantifiers \]
\[ (\text{FT-quant1}) \quad \Delta(\forall x_o)(A_o \Rightarrow B_o) \Rightarrow (A_o \Rightarrow (\forall x_o)B_o), \quad x_o \text{ is not free in } A_o, \]
\[ (\text{FT-quant2}) \quad (\forall x_o)(A_o \Rightarrow B_o) \Rightarrow ((\exists x_o)A_o \Rightarrow B_o), \quad x_o \text{ is not free in } B_o, \]
\[ (\text{FT-quant3}) \quad (\forall x_o)(A_o \& B_o) \Rightarrow ((\forall x_o)A_o \& (\forall x_o)B_o), \quad x_o \text{ is not free in } B_o, \]

\[ Axioms \ of \ descriptions \]
\[ (\text{FT-descr1}) \quad \iota_{\alpha(oa)}(E_{o(oa)}o) \equiv y_o, \quad \alpha = o, \epsilon \]

From \( A_o \equiv A'_o \) and \( B \in \text{Form}_o \) infer \( B' \)
where \( B' \) comes from \( B \) by replacing one occurrence of \( A_o \), which is not preceded by \( \lambda \), by \( A'_o \).

Let \( A_o \in \text{Form}_o \). Then, from \( A_o \) infer \( \Delta A_o \).

A theory \( T \) is a set of formulas of type \( o \) (determined by a subset of special axioms, as usual). Provability is defined as usual. If \( T \) is a theory and \( A_o \) a formula then \( T \vdash A_o \) means that \( A_o \) is provable in \( T \). A theory \( T \) is contradictory if \( T \vdash \bot \). Otherwise it is consistent.

2.4. Semantics

Interpretation of formulas is realized in a frame which is a tuple

\[ \mathcal{M} = \langle \{ M_o \, : \, o \in \text{Types} \}, \mathcal{E}_\Delta, G \rangle \]

so that the following holds:

(i) The \( \mathcal{E}_\Delta \) is an algebra of truth values (EQ-\( \Delta \)-algebra). We put \( M_o = E \) and assume that each set \( M_{oo} \cup M_{vo}o \) contains all the operations from \( \mathcal{E}_\Delta \).

(ii) \( =_o : M_o \times M_o \rightarrow L \) is a fuzzy equality on \( M_o \) and \( =_o \in M_{oo}o \) for every \( o \in \text{Types} \), which is reflexive, symmetric and transitive (i.e. \( =_o(x,y) \equiv =_o(x,z) \leq =_o(x,z), \ x, y \in M_o \)\(^*\)).

Furthermore, we define: \( =_o := _\sim, \ =_\epsilon \) is given explicitly, and

\[ [f =_\beta f'] = \bigwedge_{m \in M_o} [f(m) =_\beta f'(m)] \]

where \( f, f' \in M_{\beta_o} \).

(iii) \( G : M_o \rightarrow M_o \) is a generating function using which the fuzzy equality \( =_\epsilon \) is generated by

\[ [m =_\epsilon m'] = G(m) \sim G(m') \]

Interpretation of a formula \( A_o \) is an element from the corresponding set \( M_o \). It is defined recurrently starting with the assignment \( p \) of elements from \( M_o \) to variables (of the same type). We then write \( \mathcal{M}_p(A_o) \in M_o \).

More specifically, let the type \( \alpha \) be \( \alpha = \gamma \beta \). Then \( \mathcal{M}(A_{\beta,\gamma}) = f \in M_{\beta,\gamma} \) where \( f \) is a function \( f : M_{\beta,\gamma} \rightarrow M_\gamma \). If, moreover, \( B_\beta \) is a formula of type \( \beta \) having an interpretation \( \mathcal{M}(B_\beta) = b \in M_\beta \) then

\[ ^*\text{As usual, we will write } x =_\alpha y \text{ instead of } =_o(x,y). \text{ Sometimes, we also write } [x =_\alpha y] \text{ to denote a (arbitrary) truth value of } x =_\alpha y. \]
interpretation $\mathcal{M}(A_\beta B_\beta)$ of the formula $A_\beta B_\beta$ is a functional value $f(b) \in M_1$ of the function $f$ at point $b$.

A model of a theory $T$ is a general frame $\mathcal{M}$ for which $\mathcal{M}_{\alpha}(A_\alpha) = 1$ holds for all axioms $A_\alpha$ of $T$. Analogously as in [8], we say that it is safe if any interpretation $\mathcal{M}_{\alpha}(A_\alpha)$ is defined. A formula $A_\alpha$ is true in the theory $T$, $T \models A_\alpha$ if it is true in the degree 1 in all its models.

The following theorem generalizes Henkin theorem for classical type theory [9] (cf. also [7]).

**Theorem 1 ([4, 5])**

(a) A theory $T$ is consistent iff it has a safe general model $\mathcal{M}$.

(b) For every theory $T$ and a formula $A_\alpha$,

$$T \vdash A_\alpha \quad \text{iff} \quad T \models A_\alpha.$$

The following is an auxiliary lemma.

**Lemma 1**

(a) $\vdash (\exists z_0)(z_0 \equiv y_0)

(b) $\vdash y_0 \equiv (\exists z_0)(z_0 \& \equiv (y_0))

(c) $\vdash y_0 \equiv (\exists z_0)(z_0 \& \Delta(z_0 \equiv y_0))

(d) $\vdash (\exists z_0)z_0

(e) If $r_0$ is a constant then $\vdash (\forall z_0)r_0 \equiv r_0$ and $\vdash (\exists z_0)r_0 \equiv r_0.$

**PROOF:** (a) follows from reflexivity of $\equiv$ and $\exists$-substitution. The proof of (b) is the same as in [3] where the provable property $A_\alpha \Rightarrow (B_\alpha \Rightarrow C_\alpha)$ must be used. (c) follows from (b). (d) follows from (a) when putting $y_0 := \top$. Similarly for (e) using quantifier axioms and the properties of fuzzy equality.

3. **Description operator**

As mentioned, the description operator $t_{\alpha}(\alpha)$ is originally defined for normal fuzzy sets only, i.e. it can be applied to a formula $A_\alpha$ provided that $\vdash (\exists x_\alpha)\Delta(A_\alpha x_\alpha)$. The operator is defined for all types $\alpha$ recursively, starting with the elementary ones.

In [3], the following simple definition for the type of truth values was considered: $t_{\alpha}(\alpha) := \lambda g_\alpha . g_\alpha. \top$. Since it sometimes gives inconvenient results, we should use the following definition:

$$t_{\alpha}(\alpha) := \lambda g_\alpha . (\exists z_\alpha)(z_\alpha \& \Delta(g_\alpha z_\alpha \equiv \top)). \tag{4}$$

Its interpretation is as follows:

$$\mathcal{M}_p(t_{\alpha}(\alpha)A_\alpha) = \bigvee \{\mathcal{M}_p'(z_0) \otimes \Delta \mathcal{M}_p'(A_\alpha z_0) \mid p'(z_0) \in E, p' = p \setminus z_0\} \tag{5}$$

where $p, p'$ are assignments to variables. Clearly, (5) gives supremum of all the truth values belonging to the kernel of the fuzzy set $\mathcal{M}_p(A_\alpha)$ if it is nonempty and gives 0 otherwise.

**Lemma 2**

Let the description operator $t_{\alpha}(\alpha)$ be defined by (4). Then

(a) $t_{\alpha}(\alpha)(E_\alpha y_0) \equiv y_0$.

(b) $\vdash (\forall x_\alpha)((\exists \Delta(A_\alpha x_\alpha \equiv \top))$ then $\vdash t_{\alpha}(\alpha)A_\alpha \equiv \bot$.

**PROOF:** (a)

(L.1) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv \lambda g_\alpha . (\exists z_\alpha)(z_\alpha \& \Delta(g_\alpha z_\alpha \equiv \top))(E_\alpha y_0)$ (definition of $t$)

(L.2) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv (\exists z_\alpha)(z_\alpha \& \Delta z_\alpha \equiv y_0)$

(L.3) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv y_0$

(L.2, Lemma 1(c), rule (R))

(L.1) $\vdash t_{\alpha}(\alpha)A_\alpha \equiv (\exists z_\alpha)(z_\alpha \& \Delta(A_\alpha z_\alpha \equiv \top))$ (definition of $t_{\alpha}(\alpha)$)

(L.2) $\vdash (\forall x_\alpha)((\exists \Delta(A_\alpha x_\alpha \equiv \top))$ (assumption)

(L.3) $\Delta A_\alpha z_\alpha \equiv \bot$ (L.2, properties of FTT)

(L.4) $\vdash t_{\alpha}(\alpha)A_\alpha \equiv (\exists z_\alpha)(z_\alpha \& \bot)$

(L.1, L.3, properties of FTT)

(L.5) $\vdash t_{\alpha}(\alpha)A_\alpha \equiv \bot$ (L.4, properties of FTT)

Definition (4) gives no result for subnormal fuzzy set $g_\alpha$. Since we work in the algebra of truth values, we can modify (4) as follows:

$$t_{\alpha}(\alpha) := \lambda g_\alpha . (\exists z_\alpha)(z_\alpha \& (g_\alpha z_\alpha \equiv (\exists v_\alpha)(g_\alpha v_\alpha))). \tag{6}$$

**Lemma 3**

Let the description operator $t_{\alpha}(\alpha)$ be defined by (6). Then

$$t_{\alpha}(\alpha)(E_\alpha y_0) \equiv y_0.$$

**PROOF:**

(L.1) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv \lambda g_\alpha . (\exists z_\alpha)(z_\alpha \& (g_\alpha z_\alpha \equiv (\exists v_\alpha)(g_\alpha v_\alpha))(E_\alpha y_0)$ (definition of $t$)

(L.2) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv (\exists z_\alpha)(z_\alpha \& (z_\alpha \equiv y_0) \equiv (\exists v_\alpha)(v_\alpha \equiv y_0))$

(L.1, $\lambda$-conversion)

(L.3) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv (\exists z_\alpha)(z_\alpha \& ((z_\alpha \equiv y_0) \equiv \top))$

(L.2, Lemma 1(a), (FT-tval3), rule (R))

(L.4) $\vdash t_{\alpha}(\alpha)(E_\alpha y_0) \equiv y_0$

(L.3, Lemma 1(b), rule (R))
Hence, using definition (6), the description operator \( \tau_{o(o)} \) can be applied to any formula \( o_{oo} \), i.e. also if the corresponding fuzzy set is subnormal. Note that \( \tau_{o(o)} \) assigns this fuzzy set supremum of all the truth values belonging to it in the greatest degree.

**Lemma 4**

Let the description operator \( \tau_{o(o)} \) be defined by (6). Then

\[ \vdash \tau_{o(o)} \cdot \lambda x_o \perp \equiv \top. \]

**Proof:** This follows from definition (6), the fact that \( \vdash A_{\beta_0} \equiv A_{\beta_0} \) for all types and formulas, properties of \( \Lambda \) and Lemma 1(e). \( \square \)

Note that this lemma has a good sense. Namely, it assigns the function \( \lambda x_o \perp \) the greatest truth value from all the truth values being assigned \( 0 \) which, of course, is \( 1 \) (interpreting the constant \( \top \)). Unfortunately, as will be seen below, it prevents using (6) if we want to introduce the values “undefined” using the description operator.

4. Fuzzy type theory with partially defined functions

4.1. The value “undefined”

As already emphasized, partially defined functions will be represented by a special functional value \( \# \) which lays outside the common domain and which represents “undefined” functional value. It follows from the above results that we can make the description operator always defined when considering truth values. Moreover, since we cannot consider a special value \( #_o \) outside the truth values, we can only take it equal to some specific truth value, which naturally should be \( \perp \). But then Lemma 4 implies that definition (6) cannot be used since we arrive at contradiction. Therefore, we will in the sequel consider definition (4) for the description operator \( \tau_{o(o)} \).

In the language \( J \) we will introduce new special constants \( \{ #_\alpha | \alpha = o, \epsilon \} \) and the following definition for the other types:

\[ #_\alpha := \tau_{o(oa)} #_{oa}, \quad \alpha = o, \epsilon \]

\[ #_{\beta o} := \lambda x_o #_{\beta}. \]

Furthermore, we introduce the following new logical axioms:

(FT-U1) \( #_o \equiv \perp \),

(FT-U2) \( \forall x_o (\neg (f_o x_o \equiv \top) \equiv (\tau_{o(a)} f_{oa} \equiv #_o) \equiv #_\epsilon) \)

(FT-U3) \( \exists x_o (\exists y_o (\neg (f_{o(oa)} x_o \equiv f_{oa} y_o)) \Rightarrow (f_{o(oa)} #_a \equiv #_\beta). \)

Axiom (FT-U1) enables us to work technically with the “undefined” value for all types. This complies with our discussion above about impossibility to extend the algebra of truth values by some additional value, and also, it is in accordance with the general assumption that partial functions on undefined arguments should give falsity. Axiom (FT-U2) makes it possible to apply the description operator to all functions (which, however, may give the value “undefined”). Axiom (FT-U3) excludes constant functions since otherwise we might arrive at contradiction.

**Remark 1**

Indeed, let \( r_\beta \) be a constant and consider Axiom (FT-U1) in the following simple form:

\[ f_{\beta o} #_\alpha \equiv #_\beta. \]

Let \( f_{\beta o} := \lambda x_o r_\beta \). Then by \( \lambda \)-conversion we immediately obtain

\[ \vdash r_\beta \equiv #_\beta. \]

For example, for \( \beta = o \) and \( f_{oo} := \lambda x_o \top \) we would obtain \( \vdash \top \equiv \perp \), which is a contradiction.

With respect to Axiom (FT-U3) we must also add axiom assuring that the generating function does not lead to degenerated fuzzy equality:

(FT-G1) \( (\exists x_o)(\exists y_o) (\neg (f_{oo}(x_o) \equiv f_{oo}(y_o))) \).

The following lemma demonstrates that the definition of the missing values works correctly.

**Lemma 5**

(a) \( \vdash #_\gamma \equiv \#_\gamma. \)

(b) \( \vdash #_\gamma \equiv \#_\gamma. \)

**Proof:** (a) follows immediately from (8) using \( \lambda \)-conversion.

(b) If \( \gamma = o \) then the identity (b) is obtained from (FT-U1) and the proved property \( \vdash \perp \equiv \perp \) using rule (R). If \( \gamma = \epsilon \) then (b) follows from the previous identity, (7) and (FT-fund1). If \( \gamma = \beta_0 \) then (b) follows from the inductive assumption and (FT-fund1). \( \square \)

Recall from [3] that the following definition introduces description operator for type \( \beta_0 \):

\[ \tau_{(\beta_0 o)(\alpha_0 o)} h_{o(\beta_0)}(\lambda z_\beta (\exists f_\beta_0 (\Delta (h_{o(\beta_0)} f_{\beta_0}) \& (f_{\beta_0} x_o \equiv z_\beta))). \]

**Theorem 2**

For all types \( \gamma \),

\[ \vdash \tau_{(\gamma(o))} #_{\gamma o} \equiv #_{\gamma}. \]

**Proof:** By induction on the length of type. Let \( \gamma = o \). Then
Let $\gamma = \epsilon$. Then

(L.1) $\vdash (\lambda x_\epsilon \#_o)x_\epsilon \equiv (\lambda x_\epsilon \#_o)x_\epsilon$ (properties of FTT)

(L.2) $\vdash \Delta(\Delta((\lambda x_\epsilon \#_o)x_\epsilon \equiv \top)) \equiv \top$. (L.1, \lambda-conversion)

(L.3) $\vdash t_o(oo)\#_o \equiv \#_o$ (L.2, \lambda-conversion, properties of FTT, definition of $\#_o$)

Let $\gamma = \beta\alpha$. By (9) and the induction assumption we obtain:

(L.1) $\vdash t_{(\beta\alpha)(o)(\beta\alpha)} \#_{(\beta\alpha)} \equiv \lambda x_\alpha \cdot \beta(\beta\alpha)(\lambda z_\beta)(\Delta(\#_{(\beta\alpha)}f_{\beta\alpha}) \& (f_{\beta\alpha}x_\alpha \equiv z_\beta))$ (definition (9))

(L.2) $\vdash t_{(\beta\alpha)(o)(\beta\alpha)} \#_{(\beta\alpha)} \equiv \lambda x_\alpha \cdot \beta(\beta\alpha)(\lambda z_\beta)(\Delta(\#_{(\beta\alpha)}f_{\beta\alpha}) \& (f_{\beta\alpha}x_\alpha \equiv z_\beta))$ (L.1, \lambda-conversion, def. of $\#_{(\beta\alpha)}$)

(L.3) $\vdash \gamma_{(\psi)}(\lambda x_\beta \approx \lambda x_\beta \approx \Delta(A_{(\alpha\beta)}f_{\beta\alpha} \equiv \top))$ (L.3, L.4, properties of FTT)

(L.4) $\vdash \Delta(\Delta(\#_{(\beta\alpha)}f_{\beta\alpha}) \& (f_{\beta\alpha}x_\alpha \equiv z_\beta)) \equiv \top$. (L.3, properties of FTT)

(L.5) $\vdash f_{\beta\alpha} \approx \Delta_{(\beta\alpha)}f_{\beta\alpha} \& (f_{\beta\alpha}x_\alpha \equiv z_\beta)$ (L.4, properties of FTT)

(L.6) $\vdash t_{(\beta\alpha)(o)(\beta\alpha)} A_{(\beta\alpha)} \equiv \lambda x_\alpha \#_\beta$ (L.1, L.6, \lambda-conversion, ind. assumption)

(L.7) $\vdash t_{(\beta\alpha)(o)(\beta\alpha)} A_{(\beta\alpha)} \equiv \Delta_{(\beta\alpha)}f_{\beta\alpha} \& (f_{\beta\alpha}x_\alpha \equiv z_\beta)$ (L.7, def. of $\#_{(\beta\alpha)}$)

Both theorems above demonstrate that our definition of the “undefined” value of complex types is correct.

4.2. Canonical model

The canonical model of FTT with partial functions can be obtained by extension of the canonical model for the basic FTT (see [4, 5]). We will denote it by $M^\#$. Recall that in its construction, we consider a function $\mathcal{V}$ defined on the set of all formulas.

Let $T$ be a theory and $\approx \equiv \rightarrow$ be the canonical equivalence on formulas: $A_\alpha \approx B_\alpha$ if $T \vdash A_\alpha \equiv B_\alpha$. $\alpha \in \text{Types}$. The corresponding equivalence class is denoted by $[A_\alpha]$.

Canonical frame. This is defined in the same way as in [4, 5], namely, we put $M_\alpha = \{V(A_\alpha) \mid A_\alpha \in \text{Form}_\alpha\}$, $\alpha \in \text{Types}$ and define the fuzzy equality on it by

$$[V(A_\alpha) =_\alpha V(B_\alpha)] = [A_\alpha \equiv B_\alpha].$$

Furthermore we define:

(a) $\nu(\#_\alpha) = \{\top\}$.

(b) $\nu(\beta_\beta) = \{B_\alpha \mid T \vdash B_\alpha \equiv t_{(\alpha\beta)}A_{(\alpha\beta)} \& \text{for some } A_{(\alpha\beta)}, T \vdash (\forall x_\beta)(\Delta(A_{(\alpha\beta)}f_{\beta\alpha} \equiv \top))\}$.

(c) $\nu(\#_{(\beta\alpha)}) = [\lambda x_\alpha \#_\beta]$.

(i) $\nu(A_\alpha) = [A_\alpha]$ and $M^\#_\alpha = M_\alpha = \{V(A_\alpha) \mid A_\alpha \in \text{Form}_\alpha\}$.

(ii) $(\forall x_\beta) : M_\beta \rightarrow M_\alpha$ such that

$$\nu(A_{(\beta\alpha)}) : \nu(B_\beta) \rightarrow \nu(A_{(\beta\alpha)}B_\beta).$$

(iii) $M^\#_\alpha = M_\alpha \cup \nu(\beta_\beta)$.

(iv) $M^\#_{(\beta\alpha)} = M_\alpha \cup \nu(\#_{(\beta\alpha)})$.

(v) The generating function $G_T$ is defined by $G_T : \nu(A_\alpha) \rightarrow \nu(G_{(\alpha\beta)}A_\alpha)$ for all $A_\alpha \in \text{Form}_\alpha$. 

\[\]
Theorem 4

A theory \( T \) of EQ-FTT with partially defined functions is consistent iff it has a safe general model \( M \).

PROOF: This follows from the above construction of the canonical frame using the same arguments as in [4]. \( \square \)

5. Conclusion

In this paper, we studied possibilities how partial function can be considered in fuzzy type theory. We have chosen the most general kind of FTT based on EQ-algebra of truth values introduced in [4]. Note that there also exists propositional logic based on EQ-algebra [10].

From several options how a special value “undefined” (denoted by \( # \)) can be introduced, we decided to represent it by an element laying outside of the given domain (for each specific type). However, this cannot be done with the type of truth values, unless \( # \) is identified with a specific truth value, namely the falsity \( \bot \). Further idea is to introduce \( \#_\alpha \) for the other types using the description operator \( \iota_{\alpha(\alpha)} \) applied to formulas representing subnormal fuzzy sets. We have done this construction in this paper.

References