Fuzzy types: a framework for handling uncertainty about types of objects

Tru H. Cao a,*, Peter N. Creasy b

a Department of Engineering Mathematics, Artificial Intelligence Group, University of Bristol, Bristol BS8 1TR, UK
b Department of Computer Science and Electrical Engineering, University of Queensland, Qld 4072, Australia

Received 1 November 1999; received in revised form 1 June 2000; accepted 1 July 2000

Abstract

Like other kinds of information, types of objects in the real world are often found to be filled with uncertainty and/or partial truth. It may be due to either the vague nature of a type itself or to incomplete information in the process determining it even if the type is crisp, i.e., clearly defined. This paper proposes a framework to deal with uncertainty and/or partial truth in automated reasoning systems with taxonomic information, and in particular type hierarchies. A fuzzy type is formulated as a pair combining a basic type and a fuzzy truth-value, where a basic type can be crisp or vague (in the intuitive sense). A structure for a class of fuzzy truth-value lattices is proposed for this construction. The fuzzy subtype relation satisfying intuition is defined as a partial order between two fuzzy types. As an object may belong to more than one (fuzzy) type, conjunctive fuzzy types are introduced and their lattice properties are studied. Then, for reasoning with fuzzy types, a mismatching degree of one (conjunctive) fuzzy type to another is defined as the complement of the relative necessity degree of the former to the latter. It is proved that the defined fuzzy type mismatching degree has properties similar to those of fuzzy set mismatching degree, which allow a unified treatment of fuzzy types and fuzzy sets in reasoning. The framework provides a formal basis for development of order-sorted fuzzy logic systems. © 2000 Elsevier Science Inc. All rights reserved.

Keywords: Type hierarchy; Fuzzy truth values; Information ordering; Lattice-based reasoning; Order-sorted fuzzy logic programming; Fuzzy conceptual graphs

* Corresponding author. Tel.: +44-117-928-9799; fax: +44-117-925-1154.
E-mail addresses: tru.cao@bristol.ac.uk (T.H. Cao), peter@csee.uq.edu.au (P.N. Creasy).

0888-613X/00/$ - see front matter © 2000 Elsevier Science Inc. All rights reserved.
PII: S 0 8 8 8 - 6 1 3 X ( 0 0 ) 0 0 0 5 5 - 4
1. Introduction

It is now widely accepted that taxonomic information, e.g. a type hierarchy, is an important part of a knowledge base. This is not only because objects in the real world are naturally associated with types, but also because taxonomic information helps to reduce search space and provide efficient computation through inheritance.

Order-sorted logic or, more generally, many-sorted logic [8,30,52] and order-sorted logic programming [7,31,40,59,62] have been studied and developed to provide logical foundations for automated reasoning systems with taxonomic information and inheritance. However, research on fuzzy logic in a similar direction, particularly, on order-sorted fuzzy logic and logic programming to deal with uncertainty and/or partial truth in such systems appears to be sporadic.

Order-sorted logic and its programming languages are based on type (i.e., sort) hierarchies (i.e., partially ordered sets) and inheritance through them. Let us consider the following logic program:

\[
\begin{align*}
\text{if} & \quad x \text{ is a } \text{bird} \\
\text{then} & \quad x \text{ has wings (for every } x) \\
\text{if} & \quad x \text{ is an } \text{eagle} \\
\text{then} & \quad x \text{ is a } \text{bird} \quad (\text{for every } x)
\end{align*}
\]

Object #1 is an eagle:

The answer to the query “Does object #1 have wings?” will be “Yes”. As shown in [1], if the fact that EAGLE is a subtype of BIRD is exploited, instead of using the second rule in the program above, then efficient computation is gained through integration of inheritance directly in the unification process.

Normally, inheritance is assumed to be strict, that is, a type can always inherit the properties of its supertypes. In practice, there are cases when inheritance is not so strict and has exceptions. Fuzzy logic has been applied to model these cases by fuzzifying inheritance links among types in a type hierarchy (e.g. [36]). In this paper, our attention is focused on another kind of uncertainty and/or partial truth relating to types and inheritance through a type hierarchy.

That is the uncertainty and/or partial truth about types of objects. It may be due to either the vague nature of a type itself or to incomplete information in the process determining it even if the type is crisp, i.e., clearly defined. Examples of the first case are vague types like TALL_PERSON and PRETTY_WOMAN. An example of the second case is when one sees an animal
and can only say “It is more or less true that it is a Bird”, due to some degree of indetermination in the perception process, even when Bird is a crisp type.

As such, in classical order-sorted logic an object has strictly to be or not to be of a type, whereas in fuzzy logic an object is said to be of a type with an uncertainty and/or truth degree. One could view this type and this degree collectively as a fuzzy type assigned to this object. The notion of fuzzy types here is not the same as the notion of vague types. A fuzzy type can be imagined as a basic type, which can be crisp or vague (in the intuitive sense), fuzzified by an uncertainty and/or truth degree.

Now, suppose that one has the following rules (without exception), the pattern of which is very common in fuzzy reasoning systems:

\[
\begin{align*}
\text{if } & \text{it is true that } x \text{ is a Bird} \\
\text{then } & \text{it is true that } x \text{ has wings (for every } x) \\
\text{if } & \text{it is very true that } x \text{ is an Eagle} \\
\text{then } & \text{it is true that } x \text{ is a Bird (for every } x)
\end{align*}
\]

and the fact

- It is very true that object #1 is an Eagle

then one can infer

- It is true that object #1 has wings.

Considering (Bird, true) and (Eagle, very true) as fuzzy types, one can rewrite the first rule above as follows:

\[
\begin{align*}
\text{if } x \text{ is of (Bird, true)} \\
\text{then it is true that } x \text{ has wings (for every } x)
\end{align*}
\]

Then, if (Eagle, very true) is defined to be a fuzzy subtype of (Bird, true), the same advantage of classical order-sorted logic is obtained.

There are a number of ways uncertainty and/or partial truth are measured. They can be, for instance, probability degrees, truth degrees, possibility degrees, necessity degrees or fuzzy truth-values. In this paper, for the homogeneity of vague data that are all defined by fuzzy sets capturing the meaning of natural language terms, we mainly consider fuzzy truth-values for representation of uncertainty and/or partial truth about types of objects, but the approach can be adapted for other measures as well.

Nevertheless, we recall that fuzzy truth-values, defined as fuzzy sets on the interval \([0, 1]\) of real numbers [72], express both partial truth and uncertainty [46]. Also, fuzzy truth-values can denote linguistic truth-values, which are more usual in human expressions than values in \([0, 1]\). In a fuzzy type, a fuzzy truth-value associated with a basic type of an object can be interpreted either as a
fuzzy truth qualification [74] on a basic type assertion or as a membership grade as in the definition of $L$-fuzzy sets [29], whereby a membership grade can be a value in a lattice other than $[0, 1]$.

An early work on type hierarchies with uncertainty was [60], which defined additional uncertain relations between types by rules of the form $(t)(s) : ((l_1 u_1)(l_2 u_2))$, where $t$, $s$ were types and $(l_1 u_1)$, $(l_2 u_2)$ were support pairs, each of which defined lower and upper bounds on probability [3]. Such a rule expressed “If an object is of type $s$ then it is of type $t$ with support $(l_1 u_1)$, and if it is not of type $s$ then it is of type $t$ with support $(l_2 u_2)$”. Then the support pair for a type of an object could be inferred as with a FRIL program [5].

A similar work was [48], which defined a probabilistic knowledge base to comprise a concept lattice and a set of probabilistic formulas of the form

$$A \xrightarrow{x_1, x_2} B,$$

where $A$, $B$ were concepts and $x_1$, $x_2$ were respectively the lower and the upper bounds of the conditional probability of $B$ given $A$, which were actually a support pair as used in [60]. A deduction rule was then defined to infer from the probabilistic knowledge base the support pair for a concept given another concept.

In those two works, the main concern was inferring support pairs associated with types or concepts from uncertain rules about relations between types (in addition to a type hierarchy), rather than defining fuzzy types and their partial order. Also, there was no consideration of objects in uncertainty reasoning with a type hierarchy. In contrast, we introduce the notion of fuzzy types and define their inclusion relation, possibly with mismatching degrees, to be integrated directly in the unification process as explained above. Further, our fuzzy type framework can be used to develop an order-sorted fuzzy logic language, with both objects and their types involved in uncertain reasoning. Besides, our approach exploits only the relations between basic types given by a basic type hierarchy, without assuming an additional knowledge base of uncertain rules about them.

Meanwhile, object-oriented languages have also been extended to deal with the vagueness of object attributes and uncertainty and/or partial truth of the membership of an object in a class or of the subclass relation between two classes. In [24], vague attributes were defined by fuzzy sets, and the certainty degree of a membership or of a subclass relation was defined by relative necessity degrees of vague attributes to other ones. In [67], a truth-value that was defined by a fuzzy set on the binary truth-value set $\{0, 1\}$ was used to express uncertainty about the class membership of an object. In [4], the authors described an object-oriented extension of FRIL [5], named FRIL++, which combined object-oriented programming and FRIL fuzzy logic programming. In [18], principles for a framework of fuzzy object models were outlined and
their related research issues were identified. Some recent research results in dealing with fuzziness and uncertainty in object-oriented databases were presented in [19].

In fact, order-sorted (fuzzy) logic and (fuzzy) object-oriented languages are closely related, where the former provides a logical basis for the latter with classes being treated as types and inheritance as type unification. This paper is concerned with order-sorted fuzzy logic, for which formal semantics and soundness and completeness of proof procedures are the main issues, rather than a fuzzy object-oriented language. Notable work towards order-sorted fuzzy logic is the work on fuzzy conceptual graphs [34, 50, 53, 69] and on fuzzy conceptual graph programming [14, 16, 70], which will be briefly presented in Section 6. That work has provided basic concepts for representing uncertainty about types of objects and rules for reasoning with it, although still lacking a framework of fuzzy types and their partial order.

This paper presents a general framework of fuzzy types as outlined above for handling uncertainty about types of objects, in general, and for order-sorted fuzzy logic programming, in particular. Sections 2–5 are a refinement and extension of [15], which were partially reported in [13]. Section 2 surveys different definitions of truth-values and structures of a truth-value set in existing fuzzy logic systems, then proposes a general structure in which characteristics of truth-values, being decisive for reasoning, are distinguished. Section 3 formulates a fuzzy type as a pair combining a basic type and a fuzzy truth-value, and defines the fuzzy subtype relation as a partial order between two fuzzy types. Then, in Section 4, for the fact that an object may belong to more than one (fuzzy) type, conjunctive fuzzy types are introduced and their lattice properties are studied. For fuzzy type inference, Section 5 introduces the notion of fuzzy type mismatching degrees as the complement of relative necessity degrees of fuzzy types to other ones. An application of this fuzzy type framework for development of order-sorted fuzzy logic programming systems is described in Section 6. Finally, Section 7 presents concluding remarks of the paper.

For a symbol convention, we especially use \(\leq_{i}\) as the common symbol for all orders used in this paper, under the same umbrella of information ordering, whereby \(A \leq_{i} B\) (or, equivalently, \(B \geq_{i} A\)) means \(B\) is more informative, or more specific, than \(A\). In particular, we write \(A \leq_{i} B\) if \(B\) is a subtype of \(A\). It will be clear in a specific context which order this common symbol denotes. Also, we will write \(A <_{i} B\) (or, equivalently, \(B >_{i} A\)) to indicate that \(A \leq_{i} B\) and \(A \neq B\). For an abbreviation convention, we write iff for “if and only if”.

2. Truth-value set structures

Truth-values in fuzzy logic have been expressed in various ways. They have been defined by real numbers in \([0, 1]\) (e.g. [32, 44, 55, 58], or subintervals of \([0, 1]\)
in [65], the structures of truth-value sets in more than 30 different logical systems were reviewed and their common features studied.

However truth-values are defined, it is necessary to distinguish their truth-characteristics because they are decisive for reasoning. For example, “It is very true that object #1 is an EAGLE” entails “It is very true that object #1 is a BIRD”, but “It is very false that object #1 is a BIRD” entails “It is very false that object #1 is an EAGLE”. That is, the truth-characteristics of the truth-values effect the entailment directions. The four typical truth-characteristics are TRUE, FALSE, UNKNOWN and INCONSISTENT, and their corresponding extreme values are absolutely true, absolutely false, absolutely unknown and absolutely inconsistent.

Not all the works cited above considered truth-values with different truth-characteristics. There was no mention of truth-characteristics of truth-values in [32,54,57,58,65]. Although in [44] 0.5 was used as a splitting point of \([0, 1]\) to determine whether an interpretation satisfied a formula, that was just on “fuzzy positive logic”, as discussed in [55].

In [55], \([0, 1]\) was divided into the two half-intervals \([0, 0.5]\) and \((0.5, 1]\), which corresponded to “fuzzy negative logic” and “fuzzy positive logic” as named by the authors. Implicitly, \([0, 0.5]\) and \((0.5, 1]\) were assigned to FALSE and TRUE characteristics, respectively, and 0.5 was considered an UNKNOWN-characteristic truth-value. In [9,46], TRUE- and FALSE-characteristic fuzzy truth-values like very true and very false, defined by fuzzy sets on \([0, 1]\), were used. In [20,70], a truth-value set was explicitly defined to be a union of TRUE-, FALSE- and UNKNOWN-characteristic truth-value subsets.

Furthermore, appropriate partial orders among truth-values are required in defining fuzzy reasoning rules. In [55] the authors defined the ambiguity partial order between two truth-values that expressed which one was more ambiguous than the other. In [20] the authors defined two partial orders on interval truth-values, one of which was called the degree-of-truth and the other the degree-of-information. In [70], the partial order between two fuzzy truth-values was defined by the fuzzy subset relation. In those works, truth-characteristics of truth-values were taken into account in defining the partial orders. Whereas, in [54], the author defined the truth order and the non-specificity order on interval truth-values, but their truth-characteristics were not mentioned.

In this paper, our attention is focused on the information partial order between truth-values, like the degree-of-information order in [20] and the non-specificity order in [54] on interval truth-values. The order defines which one is more informative or more specific than the other and thus is similar in meaning to the subtype and the fuzzy subset orders. A type is more specific than one of its supertypes in the sense, for example, that “Object #1 is an EAGLE” is more informative than “Object #1 is a BIRD”, whence the former implies the latter. The information order among truth-values can also be exploited for inference.
For example, if very true is a fuzzy subset of true, then one can infer “It is true that object #1 is a BIRD” from “It is very true that object #1 is a BIRD”.

In addition, it is desirable for a fuzzy logic system that its truth-values form a lattice. In [25], a basis for lattice fuzzy logic, inspired from [61], was discussed and a particular infinite lattice was proposed. Truth-values in [54, 58], for instance, also formed lattices. Therefore, our approach in defining truth-value sets is different from those of previous works in having all the following features:

1. We consider truth-values of different truth-characteristics, for a fuzzy logic that can express and deal with both TRUE-characteristic and FALSE-characteristic assertions.
2. We consider the information partial order among truth-values taking into account their truth-characteristics, which is used for inference.
3. We define a general structure for a class of truth-value sets having the two features above, which can be applied to different definitions of truth-values suitable for particular applications, rather than a specific truth-value set.

Firstly, we recall that the set of all fuzzy sets on a domain \( U \) forms a complete lattice with the fuzzy subset relation, denoted by \( \subseteq \), as the partial order: for a set \( S \) of fuzzy sets, the greatest lower bound \( \text{glb}(S) \) is a fuzzy set defined by \( \forall u \in U : \mu_{\text{glb}(S)}(u) = \inf_{A \in S} \mu_A(u) \) and the least upper bound \( \text{lub}(S) \) is a fuzzy set defined by \( \forall u \in U : \mu_{\text{lub}(S)}(u) = \sup_{A \in S} \mu_A(u) \), where \( \mu_A() \) denotes the membership function of a fuzzy set \( A \). The least and greatest elements are respectively the empty set \( \emptyset \), considered as a special fuzzy set whose membership function has only value 0 and the universal set \( U \), considered as a special fuzzy set whose membership function has only value 1.

Here, we apply this lattice property of fuzzy sets, but using the inverse order of the fuzzy subset one, so that “greater” means “more specific”. That is, as for all orders used in this paper, under the same umbrella of information ordering, given two fuzzy sets \( A \) and \( B \) on the same domain, we write \( A \leq_i B \) (or, equivalently, \( B \geq_i A \)) for \( B \subseteq A \). We recall that, \( B \subseteq A \) means \( B \) is more specific or more informative than \( A \) in terms of the possibility distributions [73] induced by them. For example, given young and very young as linguistic labels of fuzzy sets such that very young is a fuzzy subset of young, one has young \( \leq_i \) very young.

The following definition of a truth-value lattice was first proposed [15].

**Definition 2.1.** A truth-value lattice is a set \( S_T \cup S_F \cup \{ t_0, f_0, u_0, i_0 \} \) under the information partial order, denoted by \( \leq_i \), where

1. \( S_T \) is a set of TRUE-characteristic truth-values. The greatest element of \( S_T \) is \( t_0 \) (absolutely true).
2. \( S_F \) is a set of FALSE-characteristic truth-values. The greatest element of \( S_F \) is \( f_0 \) (absolutely false).
3. For every $v_1$ and $v_2$ in the lattice, $\text{lub}\{v_1, v_2\}$ denotes the least upper bound and $\text{glb}\{v_1, v_2\}$ denotes the greatest lower bound of $\{v_1, v_2\}$, such that:

(i) if $v_1, v_2 \in S_T$, then $\text{lub}\{v_1, v_2\} \in S_T$, else
   if $v_1, v_2 \in S_F$, then $\text{lub}\{v_1, v_2\} \in S_F$, else
   $\text{lub}\{v_1, v_2\} = i_0$

(ii) if $v_1, v_2 \in S_T$, then $\text{glb}\{v_1, v_2\} \in S_T$, else
     if $v_1, v_2 \in S_F$, then $\text{glb}\{v_1, v_2\} \in S_F$, else
     $\text{glb}\{v_1, v_2\} = u_0$

4. $u_0$ (absolutely unknown) and $i_0$ (absolutely inconsistent) are respectively the least and the greatest elements of the lattice.

The defined structure of truth-value lattices can be depicted as in Fig. 1. It can be considered as a generalization with different degrees of truth and falsity of the four-valued structure of [10]. The two elements $i_0$ and $f_0$ correspond to the binary truth-values true and false in classical logic.

In the abstract definition above, $\text{lub}\{v_1, v_2\}$ and $\text{glb}\{v_1, v_2\}$ are primarily the labels of the least upper bound and the greatest lower bound of $\{v_1, v_2\}$, rather than operations on $v_1$ and $v_2$. In a specific definition of a truth-value lattice, they may be realized by specific operations, as shown in the following examples.

Example 2.1. In this example, a truth-value is defined as a fuzzy set on $[0, 1]$, i.e., a fuzzy truth-value. TRUE and FALSE characteristics are defined as in [6]: a fuzzy truth-value is said to have the TRUE characteristic if its membership function is monotonic increasing to level unity, or the FALSE characteristic if its membership function is monotonic decreasing from level unity, where the monotonicity must be strict if the membership level is other than 0, as illustrated in Fig. 2.

Fig. 1. A general structure of truth-value lattices.
The membership functions of $t_0$, $f_0$, $u_0$ and $i_0$ are defined as the fuzzy truth-values absolutely true, absolutely false, absolutely unknown and absolutely inconsistent as usual:

\[
\mu_{\text{absolutely true}}(1) = 1 \quad \text{and} \quad \mu_{\text{absolutely true}}(u) = 0 \quad \text{for all} \quad u \neq 1
\]
\[
\mu_{\text{absolutely false}}(0) = 1 \quad \text{and} \quad \mu_{\text{absolutely false}}(u) = 0 \quad \text{for all} \quad u \neq 0
\]
\[
\forall u \in [0, 1] : \mu_{\text{absolutely inconsistent}}(u) = 0
\]
\[
\forall u \in [0, 1] : \mu_{\text{absolutely unknown}}(u) = 1
\]

and $\leq$, $\text{lub}$ and $\text{glb}$ are defined like those for a fuzzy set lattice as recalled above. This truth-value lattice definition is still not very specific, and can have variations. For example, there is a variety of fuzzy set intersection and union definitions [42], and $\text{lub}$ and $\text{glb}$ can be defined to be any corresponding ones of them. Also, $\mathbf{S}_T$ and $\mathbf{S}_F$ can be countable collections of linguistic truth-values, e.g. in [9], then $\text{lub}\{v_1, v_2\}$ and $\text{glb}\{v_1, v_2\}$ are the labels of the least upper bound and the greatest lower bound of $\{v_1, v_2\}$ rather than fuzzy set operations on them.

**Example 2.2.** In this example, a truth-value is defined as a real number in $[0, 1]$. Systems based on real numbers in $[0, 1]$ and having truth-characteristics distinguished, e.g. [55] and [20], commonly use 0.5 as the splitting point between $\text{false}$- and $\text{true}$-characteristic regions, where 0.5 is considered an unknown-characteristic truth-value. Then, a truth-value set can be defined as follows:

\[
\mathbf{S}_T = (0.5, 1], \quad \mathbf{S}_F = [0, 0.5), \quad t_0 = 1, \quad f_0 = 0, \quad u_0 = 0.5
\]
\[
\forall v_1, v_2 \in \mathbf{S}_T : v_1 \leq v_2 \iff v_1 \leq v_2
\]
\[
\text{lub}\{v_1, v_2\} = \max\{v_1, v_2\}
\]
\[
\text{glb}\{v_1, v_2\} = \min\{v_1, v_2\}
\]
where $\leq$ is the usual real number less-than-or-equal-to relation. The meaning of the order here is similar to the ambiguity order [55], as illustrated in Fig. 3, that is, the nearer a truth-value to 0.5 the less informative it is. However, there is no straightforward definition of inconsistent-characteristic truth-values by real numbers in $[0, 1]$, whereas fuzzy sets on $[0, 1]$ are more expressive in that they can represent even unknown- and inconsistent-characteristic truth-values with various degrees. This is simply because fuzzy set values are “two-dimensional” in comparison with “one-dimensional” real numbers in $[0, 1]$.

Based on the structure proposed above, in [13], we defined a structure for fuzzy truth-value lattices, which is presented below, to be used to formulate fuzzy types. It also contains two distinguished true- and false-characteristic truth-value subsets, but is not restricted in having only absolutely unknown and absolutely inconsistent as unknown- and inconsistent-characteristic truth-values, respectively. In fact, it may include an arbitrary fuzzy set on $[0, 1]$ as a truth-value and thus unknown- and inconsistent-characteristic truth-values with various degrees.

In the following definition, for every fuzzy set $A$ on a domain $U$ and every $\varepsilon \in [0, 1]$, $A + \varepsilon$ denotes the fuzzy set whose membership function is defined by:

$$\forall u \in U : \mu_{A+\varepsilon}(u) = \min\{1, \mu_A(u) + \varepsilon\}$$

representing $A$ being pervaded overall with an indetermination degree $\varepsilon \in [0, 1]$, where there is no information which could allow one to assign different degrees of indetermination to different elements of $U$ [49].

**Definition 2.2.** A fuzzy truth-value lattice is a lattice of fuzzy sets on $[0, 1]$ that includes two complete sublattices $T$ and $F$ such that:

1. $\forall v_1 \in T \forall v_2 \in F : v_1$ and $v_2$ are incomparable, and
2. $\forall S \in T : \text{lub}(S) \in T$ and $\text{glb}(S) \in T$
   $\forall S \in F : \text{lub}(S) \in F$ and $\text{glb}(S) \in F$, and
3. \( \forall v \in T \ \forall e \in [0, 1] : \text{ if } \exists v^* \in T : v^* \leq v + e \text{ then } v + e \in T, \)
\( \forall v \in F \ \forall e \in [0, 1] : \text{ if } \exists v^* \in F : v^* \leq v + e \text{ then } v + e \in F, \)
where \( T \) and \( F \) respectively denote the set of all \text{true}-characteristic truth-values and the set of all \text{false}-characteristic truth-values in the lattice.

The basis of Condition 3 is that \( v + e \) represents \( v \) being pervaded overall with an indetermination degree \( e \), as defined above. Thus, if \( v \) is a \text{true}-characteristic truth-value (i.e., \( v \in T \)) and \( v + e \) still implies a \text{true}-characteristic truth-value (i.e., \( \exists v^* \in T : v^* \leq v + e \)), then \( v + e \) should still be a \text{true}-characteristic truth-value. The case of \text{false}-characteristic truth-values is similar. Like Definition 2.1, Definition 2.2 gives a structure rather than a specific definition of a fuzzy truth-value lattice. Here, for generality, we deliberately leave \( T \) and \( F \) to be specifically defined in a particular implementation. For instance, the \text{true} and the \text{false} characteristics can be defined as in Example 2.1.

3. Single fuzzy types

The formulation of a fuzzy type as a pair combining a basic type and a truth-value, as described in Section 1, was first proposed in [15] to be a formal basis for development of order-sorted fuzzy logic systems. Therein, a truth-value was either a \text{true}-characteristic or a \text{false}-characteristic one or, otherwise, \text{absolutely unknown} or \text{absolutely inconsistent}. Here, based on the structure of fuzzy truth-value lattices given in Definition 2.2, we extend the framework by allowing arbitrary fuzzy truth-values, including partially unknown and partially inconsistent ones with various degrees.

In the following definition, we assume a partially ordered set of basic types and a fuzzy truth-value lattice to be given. Also, under the same umbrella of information ordering, given two basic types \( t_1 \) and \( t_2 \), we write \( t_1 \leq t_2 \) to denote that \( t_2 \) is a subtype of \( t_1 \). In accordance with this, a type hierarchy is drawn with a type being nearer to the top than its supertypes. This convention with information ordering on types was also adopted in [17]. Therefore, we use the terms \text{the least specific common subtype} and \text{the most specific common supertype}, instead of \text{the maximal common subtype} and \text{the minimal common supertype}, respectively.

**Definition 3.1.** A fuzzy type is defined to be a pair \((t, v)\), where \( t \) is a basic type in a partially ordered set of basic types and \( v \) is a fuzzy truth-value in a fuzzy truth-value lattice.

The intended meaning of a fuzzy type assertion “\( x \) is of fuzzy type \((t, v)\)” is “\((x \text{ is of } t) \text{ is } v\)”, or “\( \text{It is } v \text{ that } x \text{ is of } t \)”. A basic type can be regarded as a
special fuzzy type whose fuzzy truth-value is absolutely true. For example, given EAGLE and BIRD as basic types, (BIRD, true) and (EAGLE, very false) are fuzzy types. Basic types may also be vague in nature, such as TALL_MAN and YOUNG_MAN. Then one has (TALL_MAN, very true) and (YOUNG_MAN, false) as fuzzy types. An assertion “John is of fuzzy type (TALL_MAN, very true)” means “It is very true that John is a TALL_MAN”, and “John is of fuzzy type (YOUNG_MAN, false)” means “It is false that John is a YOUNG_MAN”.

As a basis for defining the fuzzy subtype relation, we establish the two following assumptions, which were implicitly applied to the definition of the projection from a fuzzy concept to another in [70], where fuzzy truth-values were used as compatibility degrees of concept referents to concept types. They were then explicitly stated and discussed in [15] generally for any truth-value set of the structure of Definition 2.1.

Assumption 3.1. Given a type assertion “x is of t” and two truth-values v₁ and v₂ under the information order ≤₁, one has “(x is of t) is v₁” entails “(x is of t) is v₂” if v₂ ≤₁ v₁.

With v₁ and v₂ being fuzzy truth-values, this assumption is a special case of the entailment principle [74] for fuzzy sets, which states that if A is a fuzzy subset of B (i.e., B ≤₁ A) then “x is A” entails “x is B”. For examples, “(Object #1 is a BIRD) is very true” entails “(Object #1 is a BIRD) is true”, and “(Object #1 is a BIRD) is very false” entails “(Object #1 is a BIRD) is false”, provided that true ≤₁ very true and false ≤₁ very false.

Assumption 3.2. Given two type assertions “x is of t₁” and “x is of t₂” where t₁ is a subtype of t₂ and a truth-value v, one has:
1. “(x is of t₁) is v” entails “(x is of t₂) is v” if v has the TRUE characteristic.
2. “(x is of t₂) is v” entails “(x is of t₁) is v” if v has the FALSE characteristic.

For examples, “(Object #1 is an EAGLE) is very true” entails “(Object #1 is a BIRD) is very true”, and “(Object #1 is a BIRD) is very false” entails “(Object #1 is an EAGLE) is very false”, provided that BIRD ≤₁ EAGLE. The assumption here is that, if one can assign a type to an object with a TRUE-characteristic degree, then one can assign a supertype of this type to the object with at least the same truth degree (i.e., it is possibly truer), which is actually the least specific solution subsuming all other possible solutions of the case. Dually, if one can assign a type to an object with a FALSE-characteristic degree, then one can assign a subtype of this type to the object with at least the same falsity degree (i.e., it is possibly falser).

In [60], the authors assumed that, if a type was assigned to an object with a support pair, then every supertype of it could be assigned to the object with the same support pair. However, while this assumption is reasonable with a
support pair like \([1, 1]\), which corresponds to absolutely true, it is not reason-
able with a support pair like \([0, 0]\), which corresponds to absolutely false. This
fault was due to the fact that the authors did not take into account “positive”
or “negative” characteristics of support pairs.

On the basis of Assumptions 3.1 and 3.2, we define the fuzzy subtype re-
lation that has the same intuitive idea as the ordinary subtype relation. That is,
if \(\tau_1\) and \(\tau_2\) are fuzzy types and \(\tau_1\) is a fuzzy subtype of \(\tau_2\), then “\(x\) is of \(\tau_1\)”
entails “\(x\) is of \(\tau_2\)”. The following definition of the fuzzy subtype relation is
more general than the one in [15], with consideration of arbitrary fuzzy truth-
values, including unknown-characteristic and inconsistent-characteristic
ones with various degrees. In the definition, \(T\) and \(F\) are respectively the true-
characteristic and the false-characteristic complete sublattices of a fuzzy
truth-value lattice of discourse (Definition 2.2).

**Definition 3.2.** Given two fuzzy types \((t_1, v_1)\) and \((t_2, v_2)\), \((t_2, v_2)\) is said to be a
fuzzy subtype of \((t_1, v_1)\), denoted by \((t_1, v_1) \leq (t_2, v_2)\), iff:
1. \(t_1 = t_2\) and \(v_1 \leq v_2\), or
2. \(t_1 < t_2\) and \(\exists v \in T: v \leq v_2\) and \(v_1 \leq \text{lub}\{v \in T | v \leq v_2\}\), or
3. \(t_1 > t_2\) and \(\exists v \in F: v \leq v_2\) and \(v_1 \leq \text{lub}\{v \in F | v \leq v_2\}\).

The intuition of Definition 3.2 is as follows. For case 1, “\(x\) is of \((t_2, v_2)\)”
entails “\(x\) is of \((t_1, v_1)\)” due to \(t_1 = t_2\) and Assumption 3.1. In case 2, one has
\(\text{lub}\{v \in T | v \leq v_2\} \in T\) due to \(\exists v \in T: v \leq v_2\) and Definition 2.2, whence the
case is based on the following entailment chain:
1. “\(x\) is of \((t_2, v_2)\)” entails “\(x\) is of \((t_2, \text{lub}\{v \in T | v \leq v_2\})\)” due to
\(\text{lub}\{v \in T | v \leq v_2\} \leq v_2\) and Assumption 3.1, and
2. “\(x\) is of \((t_2, \text{lub}\{v \in T | v \leq v_2\})\)” entails “\(x\) is of \((t_1, \text{lub}\{v \in T | v \leq v_2\})\)” due to
\(\text{lub}\{v \in T | v \leq v_2\} \in T, t_1 < t_2\) and Assumption 3.2, and
3. “\(x\) is of \((t_1, \text{lub}\{v \in T | v \leq v_2\})\)” entails “\(x\) is of \((t_1, v_1)\)” due to
\(v_1 \leq \text{lub}\{v \in T | v \leq v_2\}\) and Assumption 3.1.

Case 3 is similar. We note that, in case 2, if \(v_2 \in T\) then \(\text{lub}\{v \in T | v \leq v_2\} = v_2\).
Similarly, in case 3, if \(v_2 \in F\) then \(\text{lub}\{v \in F | v \leq v_2\} = v_2\). Also, if
\((t_1, v_1) \leq (t_2, v_2)\) by either case 1, 2 or 3, then \(v_1 \leq v_2\).

**Example 3.1.** Suppose the following basic type hierarchy:

```
HANDSOME_MAN
   /\
  /   \
TALL_MAN YOUNG_MAN
   |   |
   MAN
```

which assumes that a handsome man must be tall and young. Then one has
\((TALL\_MAN, \text{true}) \leq (HANDSOME\_MAN, \text{very true})\) and \((HANDSOME\_MAN, 
false) \leq (YOUNG\_MAN, \text{very false})\), provided that \text{true} \leq \text{very true} and
false \leq_i very false. That is, “It is very true that John is a HANDSOME\_MAN” entails “It is true that John is a TALL\_MAN”, and “It is very false that John is a YOUNG\_MAN” entails “It is false that John is a HANDSOME\_MAN”.

The following proposition states that the fuzzy subtype relation is a partial order on the set of all fuzzy types defined over a partially ordered set of basic types and a fuzzy truth-value lattice (see Appendix A for the proof).

**Proposition 3.1.** The fuzzy subtype relation is a partial order.

We note that, although the definitions of fuzzy types and fuzzy subtype relation here employ fuzzy truth-values, they can be adapted for other measures of uncertainty and/or partial truth as well. Indeed, instead of considering the **true** and the **false** characteristics, one can generally consider the “positive” and the “negative” characteristics of uncertainty and/or partial truth values. In particular, lower and upper bounds of probability, truth, possibility, or necessity degrees, which are commonly used in uncertainty reasoning systems (e.g. [5,22,56,57]) can be considered as “positive” characteristic and “negative” characteristic values, respectively.

For example, with the uncertain type (EAGLE, (\geq 0.9)) where 0.9 is a probability degree, the fact “Object #1 is of (EAGLE, (\geq 0.9))” means “It is probable at least to degree 0.9 that object #1 is an EAGLE”. In fact, a lattice of the structure of Definition 2.1 can be defined for lower and upper bounds of probability, truth, possibility, or necessity degrees, where \(S_T\) is considered as the set of “positive” characteristic values of the form \((\geq v)\) with \(v \in (0, 1]\), and \(S_F\) as the set of “negative” characteristic values of the form \((\leq v)\) with \(v \in [0, 1)\). The specific definition is as follows:

\[
S_T = \{(\geq v) | v \in (0, 1]\}, \quad S_F = \{(\leq v) | v \in [0, 1)\},
\]

\[
t_0 = (\geq 1), \quad f_0 = (\leq 0),
\]

\[
\forall v_1, v_2 \in S_T : (\geq v_1) \leq_i (\geq v_2) \text{ iff } v_1 \leq v_2,
\]

\[
lub\{(\geq v_1), (\geq v_2)\} = (\geq \max\{v_1, v_2\}),
\]

\[
glb\{(\geq v_1), (\geq v_2)\} = (\geq \min\{v_1, v_2\}),
\]

\[
\forall v_1, v_2 \in S_F : (\leq v_1) \leq_i (\leq v_2) \text{ iff } v_2 \leq v_1,
\]

\[
lub\{(\leq v_1), (\leq v_2)\} = (\leq \min\{v_1, v_2\}),
\]

\[
glb\{(\leq v_1), (\leq v_2)\} = (\leq \max\{v_1, v_2\}).
\]

Then it can be applied to Definition 3.2 for the information partial order between uncertain types. For example, (EAGLE, (\geq 0.9)) is a subtype of (BIRD, (\geq 0.7)), where 0.9 and 0.7 are probability degrees. The intuition is that, if “It is probable at least to degree 0.9 that an object is an EAGLE” then “It is probable
at least to degree 0.7 that it is a BIRD”, provided that EAGLE is a subtype BIRD.

4. Conjunctive fuzzy types

In fact, there can be more than one fuzzy type assertion for an object. For example, the expression “It is true that John is a TALL_MAN and it is very false that he is a YOUNG_MAN” implies two fuzzy types associated with John, which are (TALL_MAN, true) and (YOUNG_MAN, very false). To have a single type label as a lattice-based value associated with an object in such a case, we apply the conjunctive type construction technique in [1,17], whereby a conjunctive type is defined to be a non-empty finite set of pairwise incomparable types (i.e., every pair is incomparable).

Definition 4.1. A conjunctive fuzzy type is defined to be a non-empty finite set of pairwise incomparable fuzzy types.

For example, with the basic type hierarchy in Example 3.1, \{(TALL_MAN, true), (YOUNG_MAN, very false)\} is a conjunctive fuzzy type, which can be assigned to John to reflect the expression above. However, \{(HANDSOME_MAN, false), (YOUNG_MAN, very false)\} is not a valid conjunctive fuzzy type, because (HANDSOME_MAN, false) \(\leq_s\) (YOUNG_MAN, very false), when false \(\leq_f\) very false. Conjunctive fuzzy types that express (partial) inconsistency are also allowed, such as \{(TALL_MAN, false), (HANDSOME_MAN, more or less true)\}. Since a fuzzy type can be regarded as a special conjunctive fuzzy type that consists of only one fuzzy type, we may omit the bracelet brackets in writing such a special conjunctive fuzzy type.

As mentioned above, given a conjunctive fuzzy type \(T = \{\tau_1, \tau_2, \ldots, \tau_n\}\), the intended meaning of “\(x\) is of \(T\)” is “\(x\) is of \(\tau_1\) and \(x\) is of \(\tau_2\) and \ldots and \(x\) is of \(\tau_n\)”’. Thus, the conjunctive fuzzy subtype relation can be defined in a straightforward manner on the basis of the fuzzy subtype relation as follows.

Definition 4.2. Given two conjunctive fuzzy types \(T_1\) and \(T_2\), \(T_2\) is said to be a conjunctive fuzzy subtype of \(T_1\), denoted by \(T_1 \leq_{\bigcap} T_2\), iff \(\forall \tau_1 \in T_1 \exists \tau_2 \in T_2 : \tau_1 \leq_{\bigcap} \tau_2\).

For example, with the basic type hierarchy in Example 3.1, one has:

\[
\{(TALL_MAN, true), \ (HANDSOME_MAN, \ more \ or \ less \ false)\} \leq_{\bigcap} \{(TALL_MAN, \ very \ true), \ (YOUNG_MAN, \ false)\}
\]

because (TALL_MAN, true) \(\leq_{\bigcap}\) (TALL_MAN, very true) and (HANDSOME_MAN, more or less false) \(\leq_{\bigcap}\) (YOUNG_MAN, false), provided that true \(\leq_{\bigcap}\) very true and more or less false \(\leq_{\bigcap}\) false.
The following proposition states that the conjunctive fuzzy subtype relation is a partial order on the set of all conjunctive fuzzy types defined over a partially ordered set of basic types and a fuzzy truth-value lattice (see Appendix A for the proof).

**Proposition 4.1.** The conjunctive fuzzy subtype relation is a partial order.

Given a non-empty finite set $S$ of fuzzy types, an element $\tau_1$ is said to be **redundant** in $S$ iff there is an element $\tau_2$ in $S$ such that $\tau_1 \prec \tau_2$. The conjunctive fuzzy type constructed from $S$ by removing all redundant elements in $S$ is denoted by $\text{con}(S)$. For example, with the basic type hierarchy in Example 3.1, one has:

$$\text{con}\{(\text{HANDSOME\_MAN, false}), (\text{YOUNG\_MAN, very false})\} = \{(\text{YOUNG\_MAN, very false})\}$$

provided that $\text{false} \leq_i \text{very \ false}$.

While the least upper bound of two fuzzy types does not always exist, that of two conjunctive fuzzy types does. For instance, there is no least upper bound of two fuzzy types $(\text{TALL\_MAN, true})$ and $(\text{YOUNG\_MAN, very false})$, in general. Whereas, as stated in the following proposition, the least upper bound of two conjunctive fuzzy types $T_1$ and $T_2$ is $\text{con}(T_1 \cup T_2)$, which always exists (see Appendix A for the proof).

**Proposition 4.2.** The set of all conjunctive fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, forms an upper semi-lattice under the conjunctive fuzzy subtype relation where, for two conjunctive fuzzy types $T_1$ and $T_2$:

$$\text{lub}_f T_1; T_2 \hat{=} \text{con}\{\text{glb}\_f T_1; T_2\}$$

The greatest lower bound of two conjunctive fuzzy types, however, does not always exist, neither does that of two fuzzy types. For instance, there is no greatest lower bound of two conjunctive fuzzy types $\{(\text{TALL\_MAN, true})\}$ and $\{(\text{YOUNG\_MAN, very false})\}$, in general. Nevertheless, we are interested in whether it exists when the two types do have a common lower bound. The significance is that, if a finite set $S = \{T_1, T_2, \ldots, T_n\}$ of (conjunctive) fuzzy types has this property and the constraint $X \leq, T_1 & X \leq, T_2 & \ldots & X \leq, T_n$ does have a solution for the fuzzy type variable $X$, then the most specific solution is $\text{glb}(S)$, which exists. Firstly, we prove that two conjunctive fuzzy types $T_1$ and $T_2$ have this property if every pair $\tau_1 \in T_1$ and $\tau_2 \in T_2$ has this property, as stated in the following proposition (see Appendix A for the proof).

**Proposition 4.3.** Let $T_1$ and $T_2$ be two conjunctive fuzzy types such that $\forall \tau_1 \in T_1 \forall \tau_2 \in T_2 : \text{glb}\{\tau_1, \tau_2\}$ exists if $\{\tau_1, \tau_2\}$ has a lower bound. Then $\text{glb}(T_1, T_2) = \text{con}\{\text{glb}\{\tau_1, \tau_2\} \mid \tau_1 \in T_1, \tau_2 \in T_2 \text{ and } \text{glb}\{\tau_1, \tau_2\} \text{ exists}\}$. 
We now identify classes of fuzzy types that have the above-mentioned property.

**Definition 4.3.** A fuzzy type \((t, v)\) is said to be **non-negative** iff there does not exist \(v' \in F\) such that \(v' \leq v\), or **non-positive** iff there does not exist \(v' \in T\) such that \(v' \leq v\). A conjunctive fuzzy type is said to be non-negative (respectively non-positive) iff it consists of only non-negative (respectively non-positive) fuzzy types.

For example, \((\text{TALL\_MAN, very true})\) is a non-negative fuzzy type and \((\text{YOUNG\_MAN, false})\) is a non-positive one. Whereas \{\((\text{TALL\_MAN, true}), (\text{YOUNG\_MAN, very false})\)\} is neither a non-negative nor non-positive conjunctive fuzzy type. Then, the following proposition holds (see Appendix A for the proof).

**Proposition 4.4.** Let \(\tau_1\) and \(\tau_2\) be two fuzzy types, defined over a basic type lattice and a fuzzy truth-value lattice, such that both are either:
1. Constructed from the same basic type, or
2. Non-negative, or
3. Non-positive

Then \(\text{glb}\{\tau_1, \tau_2\}\) exists if \(\{\tau_1, \tau_2\}\) has a lower bound.

For the proposition above, a basic type lattice (not just a partially ordered set) is required for cases 2 and 3, so that \(\text{glb}\{t_1, t_2\}\) and \(\text{lub}\{t_1, t_2\}\) of two basic types \(t_1\) and \(t_2\) exist. In particular, if \(v_1, v_2 \in T\) then \(\text{glb}\{(t_1, v_1), (t_2, v_2)\} = (\text{glb}\{t_1, t_2\}, \text{glb}\{v_1, v_2\})\), or if \(v_1, v_2 \in F\) then \(\text{glb}\{(t_1, v_1), (t_2, v_2)\} = (\text{lub}\{t_1, t_2\}, \text{glb}\{v_1, v_2\})\). For example, with the basic type hierarchy in Example 3.1, one has:

\[
\text{glb}\{(\text{TALL\_MAN, true}), (\text{TALL\_MAN, more or less true})\} = (\text{TALL\_MAN, more or less true}),
\]
\[
\text{glb}\{(\text{TALL\_MAN, very false}), (\text{YOUNG\_MAN, false})\} = (\text{lub}\{\text{TALL\_MAN, YOUNG\_MAN}\}, \text{glb}\{\text{very false, false})\}
\]
\[
= (\text{HANDSOME\_MAN, false}).
\]

Therefore, by Propositions 4.3 and 4.4, if \(T_1\) and \(T_2\) are two conjunctive fuzzy types, defined over a basic type lattice and a fuzzy truth-value lattice, such that both are either:
1. Constructed from the same basic type, or
2. Non-negative, or
3. Non-positive

and have a common lower bound, then \(\text{glb}\{T_1, T_2\}\) exists.
For example, with the basic type hierarchy in Example 3.1, one has:
\[
\text{glb}\{(\text{TALL MAN, very true}), (\text{YOUNG MAN, true})\},
\]
\[
\text{glb}\{(\text{HANSDOME MAN, more or less true})\} = \text{con}\{\text{glb}\{(\text{TALL MAN, very true}), (\text{HANSDOME MAN, more or less true})\}\},
\]
\[
\text{glb}\{(\text{YOUNG MAN, true}), (\text{HANSDOME MAN, more or less true})\}
\]
\[
= \{(\text{TALL MAN, more or less true}), (\text{YOUNG MAN, more or less true})\}
\]
promised that more or less true \(\leq\) true \(\leq\) very true.

5. Fuzzy type mismatching degrees

In reasoning with taxonomic information, the question is what the necessity of “\(x\) is of \(\tau_1\)” given “\(x\) is of \(\tau_2\)” is. If \(\tau_1\) and \(\tau_2\) are crisp types, it is either absolutely necessary (when \(\tau_2\) is a subtype of \(\tau_1\)) or, in general, undefined. If \(\tau_1\) and \(\tau_2\) are fuzzy types, it is a matter of degree. In each case 1, 2 or 3 of Def-

inition 3.2 of the fuzzy subtype relation, if the last condition (e.g. \(v_1 \leq\) \(v_2\) in case 1) does not hold, then there is a mismatching degree of \((t_1, v_1)\) to \((t_2, v_2)\). Before defining the mismatching degree of one fuzzy type to another, we present the notion of the mismatching degree of one fuzzy set to another, which was introduced in [70] and applied in [12,14]. Mismatching degrees are taken to be the complement of relative necessity degrees and are used in place of such complements for simplicity of expressions.

**Definition 5.1.** Let \(A\) and \(A^*\) be two fuzzy sets on a domain \(U\). The mismatching degree of \(A\) to \(A^*\) is denoted by \(\Delta(A \mid A^*)\) and defined by:

\[
\Delta(A \mid A^*) = \sup_{u \in U} \{\max\{0, \mu_A(u) - \mu_{A^*}(u)\}\}.
\]

We note that \(\Delta(A \mid A^*) = 1 - N(A \mid A^*)\), where \((NA \mid A^*)\) denotes the relative necessity degree of \(A\) given \(A^*\), whence \(N(A \mid A^*) = \inf_{u \in U} \{\min\{1, 1 - \mu_A(u) + \mu_{A^*}(u)\}\}\). This definition of the relative necessity degree (or certainty degree) of one fuzzy set to another was proposed in [49] which, as analysed in [28], avoids counter-intuitive behaviour problems of other definitions. The defined fuzzy set mismatching degree function \(\Delta\) has the following properties (see Appendix A for the proofs).

**Proposition 5.1.** For every fuzzy set \(A\) and \(\varepsilon \in [0, 1]\), \(A + \varepsilon\) is the least specific solution for \(A^*\) such that \(\Delta(A \mid A^*) \leq \varepsilon\).

**Proposition 5.2.** Let \(A, A^*, A_1\) and \(A_2\) be fuzzy sets on the same domain. Then the following properties hold:

1. \(\Delta(A \mid A^*) = 0\) iff \(A \leq A^*\), i.e., \(A^* \subseteq A\).
2. If \(A_1 \subseteq A_2\) then \(\Delta(A \mid A_2) \leq \Delta(A \mid A_1)\).
3. \(A + \varepsilon \leq A^*\) iff \(\Delta(A \mid A^*) \leq \varepsilon\), for every \(\varepsilon \in [0, 1]\).
We recall that $\Delta(A \mid A^*) \leq \varepsilon$ is equivalent to $N(A \mid A^*) \geq 1 - \varepsilon$. The significance of Proposition 5.1 is that from “It is certain at least to degree $\alpha$ that $x$ is $A$” one can infer “$x$ is $A + (1 - \alpha)$” as the least specific solution, in accordance with the principle of minimum specificity [23]. On the other hand, by property 3 in Proposition 5.2, which implies Proposition 5.1, the solution $\varepsilon$ for the constraint $A + \varepsilon \leq A^*$, given $A$ and $A^*$, is $\Delta(A \mid A^*) \leq \varepsilon$.

Given two fuzzy types $\tau_1$ and $\tau_2$, we also denote the mismatching degree of $\tau_1$ to $\tau_2$ by $\Delta(\tau_1 \mid \tau_2)$, which is a value in $[0, 1]$. If $\Delta(\tau_1 \mid \tau_2) = 0$ then $\tau_1 \leq \tau_2$. When $\Delta(\tau_1 \mid \tau_2) \neq 0$, “$x$ is of $\tau_2$” does not fully entail “$x$ is of $\tau_1$”, but rather $1 - \Delta(\tau_1 \mid \tau_2)$ measures the relative necessity degree of “$x$ is of $\tau_1$” given “$x$ is of $\tau_2$”. If $\tau_1$ and $\tau_2$ do not satisfy the conditions except for the last one of any case of Definition 3.2, then $\Delta(\tau_1 \mid \tau_2)$ is undefined and $\tau_1$ is said to be not matchable to $\tau_2$, as formally defined below.

**Definition 5.2.** A fuzzy type $(t_1, v_1)$ is said to be matchable to a fuzzy type $(t_2, v_2)$ iff:
1. $t_1 = t_2$, or
2. $t_1 < t_2$ and $\exists v \in T : v \leq_v v_2$, or
3. $t_1 > t_2$ and $\exists v \in F : v \leq_v v_2$.

The mismatching degree of one fuzzy type to another is then formally defined as follows, where $T$ and $F$ are respectively the true-characteristic and the false-characteristic complete sublattices of a fuzzy truth-value lattice of discourse.

**Definition 5.3.** Let $\tau_1 = (t_1, v_1)$ and $\tau_2 = (t_2, v_2)$ be two fuzzy types such that $\tau_1$ is matchable to $\tau_2$. Then the mismatching degree of $\tau_1$ to $\tau_2$, denoted by $\Delta(\tau_1 \mid \tau_2)$, is defined to be either:
1. $\Delta(v_1 \mid v_2)$ if $t_1 = t_2$, or
2. $\Delta(v_1 \mid \text{lub}\{v \in T \mid v \leq_v v_2\})$ if $t_1 < t_2$ and $\exists v \in T : v \leq_v v_2$, or
3. $\Delta(v_1 \mid \text{lub}\{v \in F \mid v \leq_v v_2\})$ if $t_1 > t_2$ and $\exists v \in F : v \leq_v v_2$.

For example, with the basic type hierarchy in Example 3.1, one has:

$\Delta(\text{TALL\_MAN, very true} \mid \text{HANDSOME\_MAN, true}) = \Delta(\text{very true} \mid \text{true})$ where $1 - \Delta(\text{very true} \mid \text{true})$ measures the necessity of “It is very true that John is a TALL\_MAN” given “It is true that John is a HANDSOME\_MAN”. Meanwhile, (TALL\_MAN, very true) is not matchable to (HANDSOME\_MAN, false), as given “It is false that John is a HANDSOME\_MAN”, one cannot say anything about the necessity of “It is very true that John is a TALL\_MAN”. For simplicity, given two fuzzy types $\tau_1$ and $\tau_2$, we may write $\Delta(\tau_1 \mid \tau_2)$ without explicitly stating that $\tau_1$ is matchable to $\tau_2$.

Definition 5.3 can be considered as a generalization, with arbitrary fuzzy truth-values, of the definition of fuzzy concept projection mismatching degrees in [70], with only true-, false- and unknown-characteristic fuzzy
truth-values. In case 2 of Definition 5.3 (case 3 is similar), if \( v_2 \in T \) then \( \text{lub}\{v \in T \mid v \leq_1 v_2\} = v_2 \), whence the result is \( \Delta(v_1 \mid v_2) \). Whereas, if \( v_2 \not\in T \) and \( \{v \in T \mid v \leq_1 v_2\} \) is infinite, then \( \text{lub}\{v \in T \mid v \leq_1 v_2\} \) may not be computable. Fig. 4 illustrates such a fuzzy truth-value \( v_2 \) where \( v_2 \not\in T \) but \( \exists v \in T : v \leq_1 v_2 \).

However, in practice, with a membership function diagram defined by straight line segments like this, \( \text{lub}\{v \in T \mid v \leq_1 v_2\} \) can be simply computed, even when \( \{v \in T \mid v \leq_1 v_2\} \) is infinite. Indeed, its diagram comprises the increasing parts of the diagram of \( v_2 \) and the level line segments drawn from its (relative) maximal points, as illustrated by the dash diagram in Fig. 4, assuming that a fuzzy truth-value with a membership function being monotonic increasing to level unity has the \text{TRUE}-characteristic. Nevertheless, in general, to guarantee the computability of \( \Delta((t_1, v_1) \mid (t_2, v_2)) \), one has to avoid the case when \( v_2 \not\in T \) and \( \exists v \in T : v \leq_1 v_2 \), or \( v_2 \not\in F \) and \( \exists v \in F : v \leq_1 v_2 \). Such a fuzzy truth-value \( v_2 \) and fuzzy type \((t_2, v_2)\) are said to be \textit{abnormal}.

In [26], the authors defined the similarity degree between two adjacent types in a type hierarchy to be one. Then the similarity degree between any two types was defined to be the sum of the lengths of the paths from the two types to their most specific common supertype. In [50,51], the author defined a similarity measure on types in a like manner and a similarity measure on fuzzy sets, then combined them to define a similarity degree between two fuzzy conceptual graphs. Since such a similarity degree function is commutative, it can be used for similarity reasoning but not for lattice-based reasoning. In contrast, our fuzzy type mismatching degree function, as intended for lattice-based reasoning, is not commutative and takes into account the information partial order between fuzzy types.

The notions of matchability and the mismatching degree of one fuzzy type to another can also be extended to conjunctive fuzzy types. For the extension, the min and the max functions are used on the basis that “\( x \) is of
\{\tau_1, \tau_2, \ldots, \tau_n\}\) means the conjunction of “\(x\) is of \(\tau_1\)”, “\(x\) is of \(\tau_2\)”, ..., and “\(x\) is of \(\tau_n\)||

**Definition 5.4.** A conjunctive fuzzy type \(T_1\) is said to be matchable to a conjunctive fuzzy type \(T_2\) iff \(\forall \tau_1 \in T_1 \exists \tau_2 \in T_2 : \tau_1\) is matchable to \(\tau_2\).

**Definition 5.5.** Let \(T_1\) and \(T_2\) be two conjunctive fuzzy types such that \(T_1\) is matchable to \(T_2\). Then the mismatching degree of \(T_1\) to \(T_2\), denoted by \(\Delta(T_1\mid T_2)\), is defined to be \(\max_{\tau_1 \in T_1} \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2) \mid \tau_1\) is matchable to \(\tau_2\}\).

For example, with the basic type hierarchy in Example 3.1 and \(true \leq_{\text{v}} very\ true\), one has:

\[
\Delta(\{(\text{TALL\_MAN, true}), (\text{HANDSOME\_MAN, false})\})
\]

\[
\mid \{(\text{TALL\_MAN, very true}), (\text{YOUNG\_MAN, more or less false})\})\}
\]

\[
= \max\{\Delta((\text{TALL\_MAN, true}) \mid (\text{TALL\_MAN, very true}))
\]

\[
\Delta((\text{HANDSOME\_MAN, false}) \mid (\text{YOUNG\_MAN, more or less false}))\}
\]

\[
= \max\{\Delta(\text{true} \mid \text{very true}), \Delta(\text{false} \mid \text{more or less false})\}
\]

\[
= \max\{0, \Delta(\text{false} \mid \text{more or less false})\} = \Delta(\text{false} \mid \text{more or less false}).
\]

Meanwhile, \(\{(\text{TALL\_MAN, very true}), (\text{HANDSOME\_MAN, false})\}\) is not matchable to \(\{(\text{YOUNG\_MAN, very false})\}\), because \((\text{TALL\_MAN, very true})\) is not matchable to \((\text{YOUNG\_MAN, very false})\).

As for fuzzy types, given two conjunctive fuzzy types \(T_1\) and \(T_2\), we may write \(\Delta(T_1 \mid T_2)\) without explicitly stating that \(T_1\) is matchable to \(T_2\). Also, \(\Delta(T_1 \mid T_2)\) may not be computable if \(T_2\) contains an abnormal fuzzy type. A conjunctive fuzzy type that contains an abnormal fuzzy type is said to be abnormal too.

We now prove that the fuzzy type and the conjunctive fuzzy type mismatching degree functions have properties similar to those of the fuzzy set mismatching degree function stated by Propositions 5.1 and 5.2. Firstly, for every \(\varepsilon \in [0, 1]\) and fuzzy type \(\tau = (t, v)\), we define \(\tau + \varepsilon\) to be \((t, v + \varepsilon)\). Then the following propositions hold (see Appendix A for the proofs).

**Proposition 5.3.** For every fuzzy type \(\tau_1\) and \(\varepsilon \in [0, 1]\), \(\tau_1 + \varepsilon\) is the least specific solution for \(\tau_2\) such that \(\Delta(\tau_1 \mid \tau_2) \leq_{\varepsilon}\).

**Proposition 5.4.** Let \(\tau_1, \tau_2\) and \(\tau_3\) be fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, such that \(\tau_1\) is matchable to \(\tau_2\). Then the following properties hold:

1. \(\Delta(\tau_1 \mid \tau_2) = 0\) iff \(\tau_1 \leq_{\varepsilon}, \tau_2\).
2. If \(\tau_2 \leq_{\varepsilon}, \tau_3\), then \(\tau_1\) is matchable to \(\tau_3\) and \(\Delta(\tau_1 \mid \tau_3) \leq \Delta(\tau_1 \mid \tau_2)\).
3. \(\tau_1 + \varepsilon \leq_{\varepsilon}, \tau_2\) iff \(\Delta(\tau_1 \mid \tau_2) \leq \varepsilon\), for every \(\varepsilon \in [0, 1]\).
For a conjunctive fuzzy type $T$, we define $T + \varepsilon$ to be $\text{con}\{\tau + \varepsilon | \tau \in T\}$. Then similar properties are obtained as stated in the following propositions (see Appendix A for the proofs).

**Proposition 5.5.** For every conjunctive fuzzy type $T_1$ and $\varepsilon \in [0, 1]$, $T_1 + \varepsilon$ is the least specific solution for $T_2$ such that $\Delta(T_1 | T_2) \leq \varepsilon$.

**Proposition 5.6.** Let $T_1, T_2$ and $T_3$ be conjunctive fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, such that $T_1$ is matchable to $T_2$. Then the following properties hold:

1. $\Delta(T_1 | T_2) = 0$ iff $T_1 \leq T_2$.
2. If $T_2 \leq T_3$, then $T_1$ is matchable to $T_3$ and $\Delta(T_1 | T_3) \leq \Delta(T_1 | T_2)$.
3. $T_1 + \varepsilon \leq T_2$ iff $\Delta(T_1 | T_2) \leq \varepsilon$, for every $\varepsilon \in [0, 1]$.

The significance of Propositions 5.3 and 5.4 and Propositions 5.5 and 5.6 is that fuzzy types and conjunctive fuzzy types can be treated in the same way as fuzzy set values, with regard to mismatching degree qualification, propagation and modification. In particular, for any fuzzy value $v$, which is a fuzzy set, a fuzzy type or a conjunctive fuzzy type, and $\varepsilon \in [0, 1]$, $v + \varepsilon$ is the least specific solution for $v'$ such that $\Delta(v | v') \leq \varepsilon$. This then provides a basis for order-sorted fuzzy logic programming as presented in Section 6.

### 6. Order-sorted fuzzy logic programming

The framework of fuzzy types presented above can be applied to develop order-sorted fuzzy logic programming systems. Firstly, we note that fuzzy logic programming systems can be roughly classified into two groups with respect to whether they involve fuzzy sets in programs or not. Systems that do not involve fuzzy sets usually have formulas weighted by real numbers in the interval $[0, 1]$, interpreted as truth or uncertainty degrees, e.g. [2,21,35,41,44,55,71]. Systems that involve fuzzy sets, which we call fuzzy set logic programming, include those of [5,28,33,43,45,66,68].

In fact, on the basis of the lattice property of fuzzy sets (under the information order as presented in Section 2), the entailment principle, and the principle of minimum specificity, fuzzy set logic programming can be studied and developed in a lattice-based reasoning framework. We recall that, by the entailment principle, if $A \subseteq B$ then “$x$ is $A$” entails “$x$ is $B$” and, by the principle of minimum specificity, if “$x$ is $A$” and “$x$ is $B$” then “$x$ is $A \cap B$”, where $x$ is a variable and $A$ and $B$ are fuzzy sets on the same domain.

With this point of view, in [11,12], annotated fuzzy logic programming was developed as a framework for fuzzy set logic programming. It extends the lattice-based reasoning framework of classical annotated logic programming
by considering both atoms and terms of predicate logic as objects, which all can be annotated, and using multiple annotation upper semi-lattices, which are ones of fuzzy sets on different domains. An annotated fuzzy logic program consists of Horn-like clauses of the following form:

\[ \text{Obj} : H \leftarrow \text{Obj}_1 : B_1 \land \text{Obj}_2 : B_2 \land \cdots \land \text{Obj}_n : B_n, \]

where \( \text{Obj} \) and \( \text{Obj}_i \)'s are first-order predicate logic terms or atoms, and \( H \) and \( B_i \)'s are fuzzy set constants.

For example, in the following program, \text{buy}() and \text{like}() are atoms while \text{price}() is a term, in classical predicate logic terminology, and \text{true}, \text{very true}, \text{not expensive} and \text{very cheap} are linguistic labels of fuzzy sets:

\begin{align*}
\text{buy}(\text{John}, x) : \text{very true} & \leftarrow \text{like}(\text{John}, x) : \text{true} \land \text{price}(x) : \text{not expensive} \\
\text{like}(\text{John}, \#36) : \text{very true} & \\
\text{price}(\#36) : \text{fairly cheap}. \\
\end{align*}

Assuming \( x \) and \#36 to stand for cars, the rule says “If it is true that John likes a car and the car’s price is not expensive, then it is very true that he buys it”. The first fact says “It is very true that John likes car \#36”. The second fact says “The price of car \#36 is fairly cheap”.

The declarative semantics was defined, then a sound and complete SLD-resolution style proof procedure was developed for annotated fuzzy logic programs. The proof procedure selects reductants rather than clauses of a program in resolution steps, and involves solving constraints on fuzzy value terms. For efficient computation, the meta-level fuzzy rule model of [49] was applied, with the following inference pattern of certainty degree qualification, propagation and modification as in [28]:

\begin{align*}
\text{Obj} : H & \leftarrow \text{Obj}_1 : B_1 \land \text{Obj}_2 : B_2 \land \cdots \land \text{Obj}_n : B_n \\
\text{Obj}_1 : B_1^* & \land \text{Obj}_2 : B_2^* \land \cdots \land \text{Obj}_n : B_n^* \\
\text{Obj} : H^* \\
\end{align*}

where \( H^* = H + \max_{i=1,n} \{ \Delta B_i | B_i^* \} \).

The annotated fuzzy logic programming framework is, however, general for systems with other models to be studied and developed. Inherited from annotated logic programming, the framework has two main advantages, as compared with previous fuzzy set logic programming approaches. First, it can deal with local inconsistency, that is, a program containing local inconsistencies does not arbitrarily entail everything. Second, in annotated fuzzy logic programs, fuzzy set values as annotations are separated from symbolic objects, whence symbolic manipulation such as pattern matching and unification can be performed as in classical logic programming, while lattice-based deduction based on a particular computation model can be studied independently.
Furthermore, since conjunctive fuzzy types form an upper semi-lattice, as stated in Proposition 4.2, and can be treated in the same way as fuzzy set values in fuzzy reasoning, as noted after Proposition 5.6, they can be added to annotated fuzzy logic programs as *type annotations* to extend the programs to order-sorted ones. An example of such an extended annotated fuzzy logic program is as follows:

\[
\text{buy}(x, y) : \text{very true} \leftarrow \text{type}(x): (\text{RICH\_MAN}, \text{very true}) \land \text{type}(y): \text{CAR} \\
\land \text{like}(x, y): \text{more or less true}
\]

\[\text{type(John): (RICH\_MAN, not false)}\]
\[\text{type(#36): CAR}\]
\[\text{like(John, #36): very true}\]

Here, *type()* is a reserved unary predicate added to the annotated fuzzy logic programming formalism. Also, we briefly write *CAR* for *(CAR, absolutely true)*, and a fuzzy type like *(RICH\_MAN, very true)*, for instance, for the conjunctive fuzzy type *(RICH\_MAN, very true)*. The rule says “If it is very true that a person is a RICH\_MAN and it is more or less true that he likes a car, then it is very true that he buys it”. The first fact says “It is not false that John is a RICH\_MAN”. The second fact says “#36 is a CAR”. The third fact says “It is very true that John likes car #36”.

In fact, annotated (fuzzy) logic programming provides an abstract framework rather than a concrete language for studying lattice-based reasoning. That is, even though in a particular language lattice-based data may not be syntactically so clearly separated, they still can be abstractly considered as annotations. In particular, it was applied in [13,14] to the development of fuzzy conceptual graph programs (cf. [16,70]), where fuzzy concept/relation types were regarded as lattice-based annotations in the structure of conceptual graphs [63].

As defined therein, a fuzzy conceptual graph is a bipartite graph of *concept* vertices alternate with (conceptual) relation vertices, where edges connect relation vertices to concept vertices. Each concept vertex, drawn as a box, represents either an *entity concept* or an *attribute concept*. An entity concept vertex is labelled by a pair of a concept type and a concept referent, whereas an attribute concept vertex is labelled by a triple of a concept type, a concept referent, and an *attribute-value*. Each relation vertex, drawn as a circle and labelled by a relation type, represents a relation of the entities represented by the concept vertices connected to it. A concept/relation type is defined by a conjunctive fuzzy type as presented above, and an attribute-value by a fuzzy set.

For example, the fuzzy conceptual graph in Fig. 5 says “It is very true that John is an American man who is *young*, and it is more or less true that he likes a car whose colour is *blue*”, where *very true*, *more or less true*, *young* and *blue* are linguistic labels of fuzzy sets.

In this example, *[AMERICAN\_MAN, very true]: John* and *[CAR: *] are entity concepts, and *[AGE: *@ young] and *[COLOUR: *@blue] are attribute concepts,
with \texttt{(AMERICAN\_MAN, very true)}, \texttt{CAR}, \texttt{AGE}, \texttt{COLOUR} being concept types. Whereas, \texttt{((LIKE, more or less true))}, \texttt{(ATTR\_1)} and \texttt{(ATTR\_2)} are relations with \texttt{(LIKE, more or less true)}, \texttt{ATTR\_1} and \texttt{ATTR\_2} being relation types. The individual concept \texttt{[(AMERICAN\_MAN, very true): John]} refers to an entity specified by the referent John, whereas each of the generic concepts \texttt{[CAR: *]}, \texttt{[AGE: *@young]} and \texttt{[COLOUR: *@blue]} with the referent \texttt{*} refers to an unspecified entity.

A fuzzy conceptual graph program is then defined to be a finite set of Horn-like clauses of the form \texttt{if } \texttt{u} \texttt{then } \texttt{v}, where \texttt{u} and \texttt{v} are fuzzy conceptual graphs. Fig. 6 shows an example fuzzy conceptual graph program consisting of one rule saying “If a building has \textit{original} architecture, then it is \textit{true} that it is a building worth seeing and it is \textit{more or less true} that its designer is proud of it” and one fact saying “Sydney Opera House is a building that has \textit{quite original} architecture”. The dotted line connecting the two concepts \texttt{[BUILDING: *]} and \texttt{[(BUILDING\_WORTH\_SEEING, true): *]} is called a coreference link, denoting that these two generic concepts refer to the same unspecified entity.

The declarative semantics was defined in [13], then a sound and complete graph-based SLD-resolution style proof procedure was developed in [14] for fuzzy conceptual graph programs. On the one hand, to our knowledge, this forms the first sound and complete order-sorted fuzzy set logic programming system, which can deal with uncertainty about types of objects. On the other hand, it adds to the efforts of the fusion of conceptual graphs and fuzzy logic towards a knowledge representation and reasoning language that approaches

\begin{center}
\begin{tikzpicture}
  \node (Building) {\texttt{BUILDING: *}};
  \node (Attr) [below right of=Building] {\texttt{ATTR}};
  \node (Architecture) [below right of=Attr] {\texttt{ARCHITECTURE: *@original}};
  \path (Building) -- (Attr) node [midway, left] {\texttt{if}};
  \path (Attr) -- (Architecture) node [midway, left] {\texttt{then}};
  \end{tikzpicture}
\end{center}

\begin{center}
\begin{tikzpicture}
  \node (Building) {\texttt{BUILDING: Sydney\_Opera\_House}};
  \node (Attr) [below right of=Building] {\texttt{ATTR}};
  \node (Architecture) [below right of=Attr] {\texttt{ARCHITECTURE: *@quite\_original}};
  \path (Building) -- (Attr) node [midway, left] {\texttt{if}};
  \path (Attr) -- (Architecture) node [midway, left] {\texttt{then}};
  \end{tikzpicture}
\end{center}

Fig. 5. An example fuzzy conceptual graph.

Fig. 6. An example fuzzy conceptual graph program.
human expression and reasoning. At this juncture, conceptual graphs offers a structure for a smooth mapping to and from natural language [64], while fuzzy logic offers a methodology for approximate reasoning with words [75].

7. Conclusion

Uncertainty and/or partial truth about object types can be represented by fuzzy types as pairs combining a basic type and a fuzzy truth-value. For this formulation of fuzzy types, we have proposed a structure of lattices of fuzzy truth-values of different truth-characteristics and various degrees. A conjunctive fuzzy type is constructed as a set of fuzzy types, to provide a single type label associated with an object that has more than one fuzzy type assertion.

The fuzzy subtype and the conjunctive fuzzy subtype relations are defined on those of basic types and fuzzy truth-values, taking into account the truth-characteristics of fuzzy truth-values. We have proved that these relations are partial orders, under the same umbrella of information ordering as those of basic types and fuzzy truth-values. The presented formulation of (conjunctive) fuzzy types and the definition of the (conjunctive) fuzzy subtype relation have been shown to be adaptable for other measures of uncertainty and/or partial truth as well.

We have also proved that the set of all conjunctive fuzzy types, defined on a partially ordered set of basic types and a fuzzy truth-value lattice, forms an upper semi-lattice. In general, the greatest lower bound of two (conjunctive) fuzzy types may not exist, but we have identified those such that, if two of them have a common lower bound then their greatest lower bound does exist.

For reasoning with fuzzy types, we have defined the mismatching degree of one (conjunctive) fuzzy type to another as the complement of the relative necessity degree of a fuzzy type assertion with the former to that with the latter. We have proved that (conjunctive) fuzzy types can be treated in the same way as fuzzy set values with regard to qualification, propagation, and modification of mismatching degrees.

Grouping a basic type and a fuzzy truth-value, expressing uncertainty and/or partial truth about the basic type assigned to an object, into a fuzzy type is advantageous both for the theoretical study of lattice and mismatching degree properties of fuzzy types and for the machinery computation with them as lattice-based values. In fact, since conjunctive fuzzy types form an upper semi-lattice and can be treated in the same way as fuzzy set values in fuzzy reasoning, they can be added to annotated fuzzy logic programs as type annotations to extend the programs to order-sorted ones. We have then shown the application of the presented fuzzy type framework in the development of a sound and complete order-sorted fuzzy set logic programming system in the conceptual graph notation.
Acknowledgements

This study was carried out when the first author was doing a Ph.D. at the Department of Computer Science and Electrical Engineering, University of Queensland, supported by an Australian Overseas Postgraduate Research Scholarship and a University of Queensland Postgraduate Research Scholarship. We would like to thank the anonymous referees for their useful comments.

Appendix A.

Proposition 3.1. The fuzzy subtype relation is a partial order.

Proof.
1. Reflexivity: It is obvious that the relation is reflexive (case 1 of Definition 3.2).
2. Transitivity: From Definition 3.2, if \((t_1, v_1) \leq_i (t_2, v_2)\) then \(v_1 \leq_i v_2\). Thus, supposing that \((t_1, v_1) \leq_i (t_2, v_2) \leq_i (t_3, v_3)\), one has \(v_1 \leq_i v_2 \leq_i v_3\). Regarding the basic subtype relations between \(t_1\) and \(t_2\) and between \(t_2\) and \(t_3\), there are nine cases in total. We now prove that \((t_1, v_1) \leq_i (t_3, v_3)\) by considering them case by case:

2.1. \(t_1 = t_2\):
   - \(t_2 = t_3\): one has \((t_1, v_1) \leq_i (t_3, v_3)\) by case 1 of Definition 3.2.
   - \(t_2 \leq_i t_3\) and \(v_2 \leq_i \text{lub}\\{v \in T \mid v \leq_i v_3\}\): one has \(t_1 \leq_i t_3\) and \(v_1 \leq_i v_2 \leq_i \text{lub}\\{v \in T \mid v \leq_i v_3\}\), whence \((t_1, v_1) \leq_i (t_3, v_3)\) by case 2 of Definition 3.2.
   - \(t_2 \geq_i t_3\) and \(v_2 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_3\}\): one has \(t_3 \leq_i t_1\) and \(v_1 \leq_i v_2 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_3\}\), whence \((t_1, v_1) \leq_i (t_3, v_3)\) by case 3 of Definition 3.2.
2.2. \(t_1 \leq_i t_2\) and \(v_1 \leq_i \text{lub}\\{v \in T \mid v \leq_i v_2\}\):
   - \(t_2 = t_3\): one has \(t_1 \leq_i t_3\) and \(v_1 \leq_i \text{lub}\\{v \in T \mid v \leq_i v_2\} \leq_i \text{lub}\\{v \in T \mid v \leq_i v_3\}\), whence \((t_1, v_1) \leq_i (t_3, v_3)\) by case 2 of Definition 3.2.
   - \(t_2 \leq_3 t_3\) and \(v_2 \leq_3 \text{lub}\\{v \in T \mid v \leq_i v_3\}\): one has \(t_1 \leq_i t_3\) and \(v_1 \leq_i \text{lub}\\{v \in T \mid v \leq_i v_2\} \leq_i \text{lub}\\{v \in T \mid v \leq_i v_3\}\), whence \((t_1, v_1) \leq_i (t_3, v_3)\) by case 2 of Definition 3.2.
   - \(t_2 \geq_i t_3\) and \(v_2 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_3\}\): this case does not occur because it would lead to \(\text{lub}\\{v \in T \mid v \leq_i v_2\} \leq_i v_2 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_3\}\), which would violate Definition 2.2 requiring that a \text{TRUE}-characteristic fuzzy truth-value and a \text{FALSE}-characteristic one are not comparable.
2.3. \(t_1 \leq_i t_2\) and \(v_1 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_2\}\):
   - \(t_2 = t_3\): one has \(t_2 \leq_i t_1\) and \(v_1 \leq_i \text{lub}\\{v \in F \mid v \leq_i v_2\} \leq_i \text{lub}\\{v \in F \mid v \leq_i v_3\}\), whence \((t_1, v_1) \leq_i (t_3, v_3)\) by case 3 of Definition 3.2.
3. There is an element \( T \) such that it would lead to \( \text{lub}\{v \in T \mid v \leq s, v_3\} \), which would violate Definition 2.2 of a fuzzy truth-value lattice.

4. \( T_2 > T_3 \) and \( v_2 \leq \text{lub}\{v \in F \mid v \leq s, v_3\} \): one has \( T_3 \leq T_1 \) and \( v_1 \leq \text{lub}\{v \in F \mid v \leq s, v_3\} \), whence \((t_1, v_1) \leq (t_3, v_3)\) by case 3 of Definition 3.2.

3. **Antisymmetry:** Supposing that \((t_1, v_1) \leq (t_2, v_2)\) and \((t_2, v_2) \leq (t_1, v_1)\), one has \( v_1 \leq v_2 \text{ and } v_2 \leq v_1 \), whence \( v_1 = v_2 \). Furthermore:

   3.1. If \( T_1 < T_2 \), then \( v_1 = v_2 \leq \text{lub}\{v \in T \mid v \leq s, v_2\} \) due to \((t_1, v_1) \leq (t_2, v_2)\), and \( v_1 = v_2 \leq \text{lub}\{v \in F \mid v \leq s, v_1\} \) due to \((t_2, v_2) \leq (t_1, v_1)\), whence \( v_1 = v_2 = \text{lub}\{v \in T \mid v \leq s, v_2\} = \text{lub}\{v \in F \mid v \leq s, v_1\} \), which would violate Definition 2.2.

   3.2. If \( T_1 > T_2 \), then \( v_1 = v_2 \leq \text{lub}\{v \in F \mid v \leq s, v_2\} \) due to \((t_1, v_1) \leq (t_2, v_2)\), and \( v_1 = v_2 \leq \text{lub}\{v \in T \mid v \leq s, v_1\} \) due to \((t_2, v_2) \leq (t_1, v_1)\), whence \( v_1 = v_2 = \text{lub}\{v \in F \mid v \leq s, v_2\} = \text{lub}\{v \in T \mid v \leq s, v_1\} \), which would also violate Definition 2.2.

Thus, \( t_1 = t_2 \), whence \((t_1, v_1) = (t_2, v_2)\). \( \square \)

**Proposition 4.1.** The conjunctive fuzzy subtype relation is a partial order.

**Proof.**

1. **Reflexivity:** It is obvious that the relation is reflexive.
2. **Transitivity:** Supposing that \( T_1 \leq T_2 \leq T_3 \), one has \( \forall t_1 \in T_1 \exists t_2 \in T_2 : t_1 \leq t_2 \) and \( \forall t_2 \in T_2 \exists t_3 \in T_3 : t_2 \leq t_3 \), whence \( \forall t_1 \in T_1 \exists t_3 \in T_3 : t_1 \leq t_3 \), which means \( T_1 \leq T_3 \).
3. **Antisymmetry:** Supposing that \( T_1 \leq T_2 \) and \( T_2 \leq T_1 \), one has \( \forall t_1 \in T_1 \exists t_2 \in T_2 : (t_1 \leq t_2 \text{ and } \exists t_1^* \in T_1 : t_2 \leq t_1^*) \), where \( t_1 = t_1^* \) and thus \( t_1 = t_2 \), because otherwise \( t_1 < t_1^* \), which would violate Definition 4.1. Thus, \( \forall t_1 \in T_1 \exists t_2 \in T_2 : t_1 = t_2 \), which means \( T_1 \subseteq T_2 \). Similarly, one has \( T_2 \subseteq T_1 \), whence \( T_1 = T_2 \). \( \square \)

**Proposition 4.2.** The set of all conjunctive fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, forms an upper semi-lattice under the conjunctive fuzzy subtype relation where, for two conjunctive fuzzy types \( T_1 \) and \( T_2 \), \( \text{lub}\{T_1, T_2\} = \text{con}(T_1 \cup T_2) \).

**Proof.** As it is described, \( \text{con}(T_1 \cup T_2) \) is constructed from \( T_1 \cup T_2 \) by just removing the elements that are less specific than others in \( T_1 \cup T_2 \). Thus, for every element \( t \) in \( T_1 \) or in \( T_2 \), there is an element \( t^* \) in \( \text{con}(T_1 \cup T_2) \) such that \( t \leq \tau \), whence \( \text{con}(T_1 \cup T_2) \) is an upper bound of \( \{T_1, T_2\} \).

On the other hand, every element in \( \text{con}(T_1 \cup T_2) \) is an element in \( T_1 \) or in \( T_2 \). Thus, if \( T \) is an upper bound of \( \{T_1, T_2\} \), then for every element \( t \) in \( \text{con}(T_1 \cup T_2) \) there is an element \( t^* \) in \( T \) such that \( t \leq \tau \), whence \( \text{con}(T_1 \cup T_2) \leq T \). Therefore, \( \text{con}(T_1 \cup T_2) \) is the least upper bound of \( \{T_1, T_2\} \). \( \square \)
Proposition 4.3. Let $T_1$ and $T_2$ be two conjunctive fuzzy types such that $\forall \tau_1 \in T_1 \forall \tau_2 \in T_2 : glb\{\tau_1, \tau_2\}$ exists if $\{\tau_1, \tau_2\}$ has a lower bound. Then $glb\{T_1, T_2\} = con\{glb\{\tau_1, \tau_2\} | \tau_1 \in T_1, \tau_2 \in T_2 \text{ and } glb\{\tau_1, \tau_2\} \text{ exists}\}.$

Proof. Let $T_0 = con\{glb\{\tau_1, \tau_2\} | \tau_1 \in T_1, \tau_2 \in T_2 \text{ and } glb\{\tau_1, \tau_2\} \text{ exists}\}.$ It is obvious that $T_0 \leq T_1$ and $T_0 \leq T_2.$ Also, if $T$ is a lower bound of $\{T_1, T_2\},$ then $\forall \tau \in T \exists \tau_1 \in T_1 \exists \tau_2 \in T_2 : \tau \leq \tau_1 \text{ and } \tau \leq \tau_2.$ Thus, $\forall \tau \in T \exists \tau_1 \in T_1 \exists \tau_2 \in T_2 : glb\{\tau_1, \tau_2\}$ exists and $\tau \leq glb\{\tau_1, \tau_2\},$ whence $T \leq T_0.$ Therefore, $glb\{T_1, T_2\} = T_0.$

Proposition 4.4. Let $\tau_1$ and $\tau_2$ be two fuzzy types, defined over a basic type lattice and a fuzzy truth-value lattice, such that both are either:
1. Constructed from the same basic type, or
2. Non-negative, or
3. Non-positive.

Then $glb\{\tau_1, \tau_2\}$ exists if $\{\tau_1, \tau_2\}$ has a lower bound.

Proof. Let $(t, v_1)$ and $(t, v_2)$ be two fuzzy types constructed from the same basic type $t.$ It is obvious that $(t, glb\{v_1, v_2\})$ is a lower bound of $\{(t, v_1), (t, v_2)\}.$ We now prove that, for any lower bound $(t_0, v_0)$ of $\{(t, v_1), (t, v_2)\},$ one has $(t_0, v_0) \leq (t, glb\{v_1, v_2\}).$

1. $t_0 = t$: one has $v_0 \leq v_1$ and $v_0 \leq v_2,$ whence $v_0 \leq glb\{v_1, v_2\}$ and thus $(t_0, v_0) \leq (t, glb\{v_1, v_2\}).$

2. $t_0 < t$: one has $v_0 \leq lub\{v \in T | v \leq v_1\} \leq v_1$ and $v_0 \leq lub\{v \in T | v \leq v_2\} \leq v_2,$ whence $v_0 \leq glb\{lub\{v \in T | v \leq v_1\}, lub\{v \in T | v \leq v_2\}\} \leq glb\{v_1, v_2\}.$ Since $glb\{lub\{v \in T | v \leq v_1\}, lub\{v \in T | v \leq v_2\}\} \in T,$ one has $v_0 \leq lub\{v \in T | v \leq glb\{v_1, v_2\}\}$ and thus $(t_0, v_0) \leq (t, glb\{v_1, v_2\}).$

3. $t_0 > t$: one has $v_0 \leq lub\{v \in F | v \leq v_1\} \leq v_1$ and $v_0 \leq lub\{v \in F | v \leq v_2\} \leq v_2,$ whence $v_0 \leq glb\{lub\{v \in F | v \leq v_1\}, lub\{v \in F | v \leq v_2\}\} \leq glb\{v_1, v_2\}.$ Since $glb\{lub\{v \in F | v \leq v_1\}, lub\{v \in F | v \leq v_2\}\} \in F,$ one has $v_0 \leq lub\{v \in F | v \leq glb\{v_1, v_2\}\}$ and thus $(t_0, v_0) \leq (t, glb\{v_1, v_2\}).$

Therefore, $glb\{(t, v_1), (t, v_2)\} = (t, glb\{v_1, v_2\}).$

2. Let $(t_0, v_0)$ be a lower bound of $\{(t_1, v_1), (t_2, v_2)\}.$ Regarding the basic subtype relations between $t_0$ and $t_1$ and between $t_0$ and $t_2,$ there are totally nine cases:

1. $t_0 = t_1 = t_2$ and $v_0 \leq v_1$ and $v_0 \leq v_2.$

2. $t_0 = t_1 \leq t_2$ and $v_0 \leq v_1$ and $v_0 \leq lub\{v \in T | v \leq v_2\}.$

3. $t_0 = t_1 \geq t_2$ and $v_0 \leq v_1$ and $v_0 \leq lub\{v \in T | v \leq v_2\}.$

4. $t_0 = t_2 \leq t_1$ and $v_0 \leq lub\{v \in F | v \leq v_1\}$ and $v_0 \leq v_2.$

5. $t_0 = t_2 \geq t_1$ and $v_0 \leq lub\{v \in F | v \leq v_1\}$ and $v_0 \leq v_2.$
2.6. \( t_0 \preceq t_1 \) and \( t_0 \preceq t_2 \) and \( v_0 \preceq \text{lub}\{v \in T \mid v \preceq v_1\} \) and \( v_0 \preceq \text{lub}\{v \in T \mid v \preceq v_2\} \).

2.7. \( t_0 \succeq t_1 \) and \( t_0 \succeq t_2 \) and \( v_0 \preceq \text{lub}\{v \in F \mid v \preceq v_1\} \) and \( v_0 \preceq \text{lub}\{v \in F \mid v \preceq v_2\} \).

2.8. \( t_1 \preceq t_0 \preceq t_2 \) and \( v_0 \preceq \text{lub}\{v \in F \mid v \preceq v_1\} \) and \( v_0 \preceq \text{lub}\{v \in T \mid v \preceq v_2\} \).

2.9. \( t_1 \succeq t_0 \succeq t_2 \) and \( v_0 \preceq \text{lub}\{v \in T \mid v \preceq v_1\} \) and \( v_0 \preceq \text{lub}\{v \in F \mid v \preceq v_2\} \).

With \((t_1, v_1)\) and \((t_2, v_2)\) being non-negative fuzzy types, cases 2.3, 2.5, 2.7, 2.8, 2.9 do not occur, so \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} \) is defined as follows:

- \( t_1 = t_2 \): only cases 2.1, 2.6 are involved, whence \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_1, \text{glb}\{ (v_1, v_2) \}) \), or else
- \( t_1 \preceq t_2 \): only cases 2.2, 2.6 are involved, whence \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_1, \text{glb}\{ v \in T \mid v \preceq v_2 \}) \), or else
- \( t_1 \succeq t_2 \): only cases 2.4, 2.6 are involved, whence \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_2, \text{glb}\{ v \in T \mid v \preceq v_1 \}) \), or else
- \( t_1 \) and \( t_2 \) are not comparable: only case 2.6 is involved, whence \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (\text{glb}\{ t_1, t_2 \}, \text{glb}\{ v \in F \mid v \preceq v_1 \}) \).

Similarly, with \((t_1, v_1)\) and \((t_2, v_2)\) being non-positive fuzzy types, \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} \) is defined as follows:

- \( t_1 = t_2 \): \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_1, \text{glb}\{ v_1, v_2 \}) \), or else
- \( t_1 \preceq t_2 \): \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_2, \text{glb}\{ v \in F \mid v \preceq v_1 \}) \), or else
- \( t_1 \succeq t_2 \): \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (t_1, \text{glb}\{ v \in F \mid v \preceq v_2 \}) \), or else
- \( t_1 \) and \( t_2 \) are not comparable: \( \text{glb}\{(t_1, v_1), (t_2, v_2)\} = (\text{glb}\{ t_1, t_2 \}, \text{glb}\{ v \in F \mid v \preceq v_1 \}) \).

Proposition 5.1. For every fuzzy set \( A \) and \( \varepsilon \in [0, 1] \), \( A + \varepsilon \) is the least specific solution for \( A^* \) such that \( \Delta(A \mid A^*) \leq \varepsilon \).

Proof. Let \( U \) be the domain of \( A \) and \( A^* \). Firstly, by definition, one has:

\[
\Delta(A \mid A + \varepsilon) = \sup_{u \in U} \{ \max \{0, \mu_{A+\varepsilon}(u) - \mu_A(u)\} \}
\]

\[
= \sup_{u \in U} \{ \max \{0, \min \{1, \mu_A(u) + \varepsilon\} - \mu_A(u)\} \}
\]

\[
= \sup_{u \in U} \{ \min \{1, \mu_A(u) + \varepsilon\} - \mu_A(u) \}
\]

\[
= \min_{u \in U} \left\{ \sup_{u \in U} \{1 - \mu_A(u)\}, \varepsilon \right\}
\]

\[\leq \varepsilon.\]
One has $\Delta(A \mid A + \varepsilon) = \varepsilon$ iff $\sup_{u \in U} \{1 - \mu_A(u)\} \geq \varepsilon$. In particular, this occurs when $\exists u \in U : 1 - \mu_A(u) > \varepsilon$, that is, $A + \varepsilon \neq U$.

We now prove that, if $\Delta(A \mid A^*) \leq \varepsilon$ then $A^* \subseteq A + \varepsilon$. By definition, one has:

$$\Delta(A \mid A^*) = \sup_{u \in U} \{\max\{0, \mu_A^*(u) - \mu_A(u)\}\}.$$ 

Thus, if $\Delta(A \mid A^*) \leq \varepsilon$ then $\forall u \in U : \max\{0, \mu_A^*(u) - \mu_A(u)\} \leq \varepsilon$, whence $\forall u \in U : \mu_A^*(u) \leq \min\{1, \mu_A(u) + \varepsilon\}$, that is, $A^* \subseteq A + \varepsilon$. □

**Proposition 5.2.** Let $A, A^*, A_1$ and $A_2$ be fuzzy sets on the same domain. Then the following properties hold:

1. $\Delta(A \mid A^*) = 0$ iff $A \subseteq A^*$, i.e., $A^* \subseteq A$.
2. If $A_1 \subseteq A_2$ then $\Delta(A \mid A_2) \leq \Delta(A \mid A_1)$.
3. $A + \varepsilon \subseteq A^*$ iff $\Delta(A \mid A^*) \leq \varepsilon$, for every $\varepsilon \in [0, 1]$.

**Proof.** Properties 1 and 2 are straightforward from Definition 5.1.

For property 3, by Proposition 5.1, if $\Delta(A \mid A^*) \leq \varepsilon$ then $A + \varepsilon \subseteq A^*$. On the other hand, if $A + \varepsilon \subseteq A^*$ then, by property 2, $\Delta(A \mid A^*) \leq \Delta(A \mid A + \varepsilon)$. Since $\Delta(A \mid A + \varepsilon) \leq \varepsilon$ by Proposition 5.1, one has $\Delta(A \mid A^*) \leq \varepsilon$. □

**Proposition 5.3.** For every fuzzy type $\tau_1$ and $\varepsilon \in [0, 1]$, $\tau_1 + \varepsilon$ is the least specific solution for $\tau_2$ such that $\Delta(\tau_1 \mid \tau_2) \leq \varepsilon$.

**Proof.** Firstly, one has $\Delta(\tau_1 \mid \tau_1 + \varepsilon) = \Delta(v_1 \mid v_1 + \varepsilon)$. By Proposition 5.1, $\Delta(v_1 \mid v_1 + \varepsilon) \leq \varepsilon$, whence $\Delta(\tau_1 \mid \tau_1 + \varepsilon) \leq \varepsilon$, for every $\varepsilon \in [0, 1]$.

We now prove that, if $\Delta(\tau_1 \mid \tau_2) \leq \varepsilon$ then $\tau_1 + \varepsilon \subseteq \tau_2$. Let $\tau_1 = (t_1, v_1)$ and $\tau_2 = (t_2, v_2)$. By Definition 5.3, there are three cases:

1. $t_1 = t_2$: one has $\Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid v_1 + \varepsilon) \leq \varepsilon$, whence $\tau_1 + \varepsilon \subseteq \tau_2$ (Proposition 5.1) and thus $\tau_1 + \varepsilon \subseteq \tau_2$ by case 1 of Definition 3.2.

2. $t_1 \prec t_2$ and $\exists v \in T : v \leq v_2$: one has $\Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid \lub\{v \in T \mid v \leq v_2\}) \leq \varepsilon$, whence $\tau_1 + \varepsilon \subseteq \lub\{v \in T \mid v \leq v_2\}$ and thus $\tau_1 + \varepsilon \subseteq \tau_2$ by case 2 of Definition 3.2.

3. $t_1 \succ t_2$ and $\exists v \in F : v \leq v_2$: one has $\Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid \lub\{v \in F \mid v \leq v_2\}) \leq \varepsilon$, whence $\tau_1 + \varepsilon \subseteq \lub\{v \in F \mid v \leq v_2\}$ and thus $\tau_1 + \varepsilon \subseteq \tau_2$ by case 3 of Definition 3.2. □

**Proposition 5.4.** Let $\tau_1, \tau_2$ and $\tau_3$ be fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, such that $\tau_1$ is matchable to $\tau_2$. Then the following properties hold:

1. $\Delta(\tau_1 \mid \tau_2) = 0$ iff $\tau_1 \leq \tau_2$.
2. If $\tau_2 \leq \tau_3$, then $\tau_1$ is matchable to $\tau_3$ and $\Delta(\tau_1 \mid \tau_3) \leq \Delta(\tau_1 \mid \tau_2)$.
3. $\tau_1 + \varepsilon \leq \tau_2$ iff $\Delta(\tau_1 \mid \tau_2) \leq \varepsilon$, for every $\varepsilon \in [0, 1]$. 
3. By Proposition 5.3, if \( \Delta(\tau_1 \mid \tau_3) \leq \Delta(\tau_1 \mid \tau_2) \) by considering them case by case:

2.1. \( t_1 = t_2 \): one has \( \Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid v_2) \)

\[
\begin{align*}
& t_2 = t_3 \text{ and } v_2 \leq i, v_3: \text{ one has } \Delta(\tau_1 \mid \tau_3) = \Delta(v_1 \mid v_3) \leq \Delta(v_1 \mid v_2) = \\
& \Delta(\tau_1 \mid \tau_2). \\
& t_2 <, t_3 \text{ and } v_2 \leq i, \text{lub}\{v \in T \mid v \leq i, v_3\}: \text{ one has } \Delta(\tau_1 \mid \tau_3) = \Delta(v_1 \mid lub\{v \in T \mid v \leq i, v_3\}) \leq \Delta(v_1 \mid v_2) = \Delta(\tau_1 \mid \tau_2).
\end{align*}
\]

2.2. \( t_1 <, t_2 \text{ and } \exists v \in T : v \leq i, v_2 \)

\[
\begin{align*}
& t_2 = t_3 \text{ and } v_2 \leq i, v_3: \text{ one has } \Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid lub\{v \in T \mid v \leq i, v_3\}) \geq \\
& \Delta(v_1 \mid lub\{v \in T \mid v \leq i, v_3\}) = \Delta(\tau_1 \mid \tau_3). \\
& t_2 <, t_3 \text{ and } v_2 \leq i, lub\{v \in T \mid v \leq i, v_3\}: \text{ one has } \Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid lub\{v \in T \mid v \leq i, v_3\}) \geq \\
& \Delta(v_1 \mid lub\{v \in T \mid v \leq i, v_3\}) = \Delta(\tau_1 \mid \tau_3). \\
& t_2 \geq t_3 \text{ and } v_2 \leq lub\{v \in F \mid v \leq i, v_3\}: \text{ this case does not occur because it would lead to } \exists v \in F : v \leq i, v_2 \leq lub\{v \in F \mid v \leq i, v_3\} \in F, \text{ which would violate Definition 2.2 of a fuzzy truth-value lattice.}
\end{align*}
\]

2.3. \( t_1 >, t_2 \text{ and } \exists v \in F : v \leq i, v_2 \)

\[
\begin{align*}
& t_2 = t_3 \text{ and } v_2 \leq i, v_3: \text{ one has } \Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid lub\{v \in F \mid v \leq i, v_3\}) \geq \\
& \Delta(v_1 \mid lub\{v \in F \mid v \leq i, v_3\}) = \Delta(\tau_1 \mid \tau_3). \\
& t_2 <, t_3 \text{ and } v_2 \leq lub\{v \in T \mid v \leq i, v_3\}: \text{ this case does not occur because it would lead to } \exists v \in F : v \leq i, v_2 \leq lub\{v \in T \mid v \leq i, v_3\} \in T, \text{ which would violate Definition 2.2 of a fuzzy truth-value lattice.}
\end{align*}
\]

\[
\begin{align*}
& t_2 \geq t_3 \text{ and } v_2 \leq lub\{v \in F \mid v \leq i, v_3\}: \text{ one has } \Delta(\tau_1 \mid \tau_2) = \Delta(v_1 \mid lub\{v \in F \mid v \leq i, v_3\}) \geq \\
& \Delta(v_1 \mid lub\{v \in F \mid v \leq i, v_3\}) = \Delta(\tau_1 \mid \tau_3). \\
& \text{3. By Proposition 5.3, if } \Delta(\tau_1 \mid \tau_2) \leq \epsilon \text{ then } \tau_1 + \epsilon \leq \tau_2. \text{ On the other hand, if } \tau_1 + \epsilon \leq \tau_2 \text{ then, by property 2 above, } \Delta(\tau_1 \mid \tau_2) \leq \Delta(\tau_1 \mid \tau_1 + \epsilon). \text{ Since } \Delta(\tau_1 \mid \tau_1 + \epsilon) \leq \epsilon \text{ by Proposition 5.3, one has } \Delta(\tau_1 \mid \tau_2) \leq \epsilon. \quad \square
\end{align*}
\]

**Proposition 5.5.** For every conjunctive fuzzy type \( T_1 \) and \( \epsilon \in [0, 1] \), \( T_1 + \epsilon \) is the least specific solution for \( T_2 \) such that \( \Delta(T_1 \mid T_2) \leq \epsilon. \)

**Proof.** Let \( S = \{ \tau_1 + \epsilon \mid \tau_1 \in T_1 \} \). By Definition 5.5, one has \( \Delta(T_1 \mid T_1 + \epsilon) = \max_{\tau_1 \in T_1} \min_{\tau_2 \in \tau_1 + \epsilon} \{ \Delta(\tau_1 \mid \tau_2) \} \). Since \( T_1 + \epsilon = con(S) \) is obtained from \( S \) by just removing the elements that are less specific than others in \( S \), one has \( \forall \tau_2 \in S \ast \tau_2 \in T_1 + \epsilon : \tau_2 \leq \tau_2 \). By property 2 in Proposition 5.4, for any \( \tau_1 \), if \( \tau_1 \) is matchable to \( \tau_2 \), then \( \tau_1 \) is matchable to \( \tau_2 \) and \( \Delta(\tau_1 \mid \tau_2) \leq \Delta(\tau_1 \mid \tau_2) \). Thus, for every \( \tau_1 \in T_1 \), \( \min_{\tau_2 \in \tau_1 + \epsilon} \{ \Delta(\tau_1 \mid \tau_2) \} = \min_{\tau_2 \in \tau_1 + \epsilon} \{ \Delta(\tau_1 \mid \tau_2) \} \leq \Delta(\tau_1 \mid \tau_1 + \epsilon) \leq \epsilon \), whence \( \Delta(T_1 \mid T_1 + \epsilon) = \max_{\tau_1 \in T_1} \min_{\tau_2 \in T_1 + \epsilon} \{ \Delta(\tau_1 \mid \tau_2) \} \leq \epsilon. \)
We now prove that, if $\Delta(T_1 \mid T_2) \leq \varepsilon$ then $T_1 + \varepsilon \leq_i T_2$. One has $\Delta(T_1 \mid T_2) = \max_{\tau_1 \in T_1} \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\} \leq \varepsilon$ whence $\forall \tau_1 \in T_1 : \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\} \leq \varepsilon$. Thus, $\forall \tau_1 \in T_1 \exists \tau_2 \in T_2 : \Delta(\tau_1 \mid \tau_2) \leq \varepsilon$ whence $\tau_1 + \varepsilon \leq_i \tau_2$, which means $T_1 + \varepsilon \leq_i T_2$. \hfill $\Box$

**Proposition 5.6.** Let $T_1$, $T_2$ and $T_3$ be conjunctive fuzzy types, defined over a partially ordered set of basic types and a fuzzy truth-value lattice, such that $T_1$ is matchable to $T_2$. Then the following properties hold:

1. $\Delta(T_1 \mid T_2) = 0$ iff $T_1 \leq_i T_2$.
2. If $T_2 \leq_i T_3$, then $T_1$ is matchable to $T_3$ and $\Delta(T_1 \mid T_3) \leq \Delta(T_1 \mid T_2)$.
3. $T_1 + \varepsilon \leq_i T_2$ iff $\Delta(T_1 \mid T_2) \leq \varepsilon$, for every $\varepsilon \in [0, 1]$.

**Proof.**

1. $\Delta(T_1 \mid T_2) = 0$ iff $\max_{\tau_1 \in T_1} \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\} = 0$ iff $\forall \tau_1 \in T_1 : \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\} = 0$ iff $\forall \tau_1 \in T_1 \exists \tau_2 \in T_2 : \tau_1 \leq_i \tau_2$ iff $T_1 \leq_i T_2$.

2. If $T_2 \leq_i T_3$ then $\forall \tau_2 \in T_2 \exists \tau_3 \in T_3 : \tau_2 \leq_i \tau_3$ whence, by property 2 in Proposition 5.4, $\Delta(\tau_1 \mid \tau_3) \leq \Delta(\tau_1 \mid \tau_2)$ for any $\tau_1$ that is matchable to $\tau_2$. Thus, $\forall \tau_1 \in T_1 : \min_{\tau_3 \in T_3} \{\Delta(\tau_1 \mid \tau_3)\} \leq \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\}$, whence $\Delta(T_1 \mid T_3) = \max_{\tau_1 \in T_1} \min_{\tau_3 \in T_3} \{\Delta(\tau_1 \mid \tau_3)\} \leq \max_{\tau_1 \in T_1} \min_{\tau_2 \in T_2} \{\Delta(\tau_1 \mid \tau_2)\} = \Delta(T_1 \mid T_2)$.

3. By Proposition 5.5, if $\Delta(T_1 \mid T_2) \leq \varepsilon$ then $T_1 + \varepsilon \leq_i T_2$. On the other hand, if $T_1 + \varepsilon \leq_i T_2$ then, by property 2 above, $\Delta(T_1 \mid T_2) \leq \Delta(T_1 \mid T_1 + \varepsilon)$. Since $\Delta(T_1 \mid T_1 + \varepsilon) \leq \varepsilon$ by Proposition 5.5, one has $\Delta(T_1 \mid T_2) \leq \varepsilon$. \hfill $\Box$

**References**


