ARTICLE IN PRESS FSS:7632

Available online at www.sciencedirect.com

ScienceDirect

Fuzzy Sets and Systems ••• (••••) •••–•••

FUZZY sets and systems

www.elsevier.com/locate/fss

Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach

Hassan Aït-Kaci [a], Gabriella Pasi [b,*]

[a] *HAK Language Technologies, Canada*
[b] *Università degli Studi di Milano Bicocca, Italy*

Received 8 September 2017; received in revised form 26 March 2019; accepted 28 March 2019

Abstract

Unification and generalization are operations on two terms computing respectively their greatest lower bound and least upper bound when the terms are quasi-ordered by subsumption up to variable renaming (*i.e.*, $t_1 \preceq t_2$ iff $t_1 = t_2\sigma$ for some variable substitution σ). When term signatures are such that distinct functor symbols may be related with a fuzzy equivalence (called a *similarity*), these operations can be formally extended to tolerate mismatches on functor names and/or arity or argument order. We reformulate and extend previous work with a declarative approach defining unification and generalization as sets of axioms and rules forming a complete constraint-normalization proof system. These include the Reynolds-Plotkin term-generalization procedures, Maria Sessa's "weak" unification with partially fuzzy signatures and its corresponding generalization, as well as novel extensions of such operations to signatures with weaker functor similarities (*i.e.*, with possibly different arities). One advantage of this approach is that it requires no modification of the conventional data structures for terms and substitutions. This and the fact that these declarative specifications are efficiently executable conditional Horn-clauses offers great practical potential for fuzzy information-handling applications.

Keywords: Approximate reasoning; Fuzzy inference systems; Fuzzy constraint satisfaction; Learning; Fuzzy databases; Information retrieval; Information lattices; First-order terms; Fuzzy unification; Fuzzy generalization

1. Introduction

We are motivated by the versatile use of the First-Order Term ($\mathcal{FOT}$) as a data structure in Logic Programming thanks to the unification operation ([1] and [2]) and Inductive Logic Programming thanks to the generalization (or "anti-unification") operation ([3] and [4]). We extend the formal characterization of the set of $\mathcal{FOT}$s modulo variable renaming as a lattice due to Reynolds ([5]) and Plotkin ([6]) to such an algebraic structure where similarities among distinct constructors may exist that tolerate fuzzy $\mathcal{FOT}$ approximation. We study how these notions may be formalized

* Corresponding author.
E-mail addresses: hak@acm.org (H. Aït-Kaci), pasi@disco.unimib.it (G. Pasi).

https://doi.org/10.1016/j.fss.2019.03.019

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

while abiding by a fully declarative approach based on constraint processing in the same way as crisp unification is presented in [1] and [2], as opposed to a procedural control-conscious algorithm such as Robinson's [7].

Origins of unification and generalization

Unification. The earliest printed account of the $\mathcal{FOT}$ unification operation, although not under this name, appears in 1930 in Jacques Herbrand's PhD thesis [1].[1] Later, in 1960, Dag Prawitz uses this as part of his Natural Deduction proof procedure for First-Order Logic [8]. Chap. 5 of Herbrand's thesis is first translated into English in 1967 by Jean van Heijenoort [9], and his full thesis is translated into English in 1971 by Warren Goldfarb [10].[2] In his thesis, although he does not call it "unification," Herbrand describes a declarative specification for $\mathcal{FOT}$ equation normalization more than 30 years before J. Alan Robinson actually gives the familiar name to an equivalent procedural method and dubbing it "unification" in order to extend his resolution principle from Proposition Logic to First-Order Logic [7]. This fact is already explicitly pointed out in 1976 by Gérard Huet in his French *thèse d'état* [11]. These rules are later used explicitly by Martelli and Montanari in 1982 (20 years after Robinson's paper, and 55 years after Herbrand's 1930 PhD thesis) in their method seeking to optimize Robinsons's algorithm [2]. They do not cite Herbrand's thesis, although it is explicitly cited in Huet's 1976 thesis which they cite. Probably because its name first appears in his paper on proof by resolution in First-Order Logic in [7] and this name has been used in Logic Programming, most current venues attribute the paternity of $\mathcal{FOT}$ unification to Robinson.[3] While the name was indeed his coinage, the operation however was not new.

Generalization. In 1970, John Reynolds and Gordon Plotkin publish each an article, in the same volume, each giving a different but equivalent procedure for the generalization of two $\mathcal{FOT}$s. The former calls it "*anti-unification*" ([5], page 138), and the latter calls it "*least generalization*" ([6], page 155). Each describes a method for computing the most specific $\mathcal{FOT}$ subsuming two given $\mathcal{FOT}$s in finitely many steps. The method consists in scanning them simultaneously from left to right as long as they agree, and where they disagree generating a pair of minimal generalizing substitutions by introducing a fresh variable, each time replacing the disagreeing terms with this new variable wherever they occur again simultaneously in each term.

Interestingly, in their 1982 ACM ToPLaS article on unification [2], Martelli and Montanari use a method that computes generalization of two terms implicitly (so-called "*common parts*") in preprocessing equations into congruence classes of terms (called "*multi-equations*"). This is in order to make unification more efficient by solving not just one equation but many at a time. However, they do not point out that this common-part form derived from two terms, by keeping only what is common in both, is in fact the dual of their unified form, which brings into one term what is in either.

Contribution of this work

Our objective in this work is to provide a formal and operational answer to the following question:

> "*What happens to the original Reynolds-Plotkin lattice—that is, $\mathcal{FOT}$s preordered with $\mathcal{FOT}$ subsumption, $\mathcal{FOT}$ substitutions preordered by way of composition,*[4] *and operations such as unification and generalization—when syntactic term equality is generalized into a weighted measure of similarity as done in Fuzzy Logic?*"

Starting from the original Reynolds-Plotkin lattice of First-Order Terms ordered by subsumption, we first give a constraint-based formal specification of its least upper-bound operation (generalization, or anti-unification) as has been the case for its greatest lower-bound operation (unification). This declarative semantics is new and simplifies establishing the formal correctness of the specification as well as providing an implicit operational semantics for both lattice operations by way of constraint normalization. We then allow these operations to be calibrated at approximation degrees in [0, 1] upon the existence of distinct but possibly similar function symbols (traditionally called "*functors*"

[1] Ref. [1], Chap. 5, Sec. 2.4, pp. 95–96, where it corresponds to expanding an equation into normal form that verifies what he calls "*Property A*."
[2] Chap. 5 is on pp. 148*ff*.
[3] For example, `https://en.wikipedia.org/wiki/Unification_(computer_science)`.
[4] I.e., ordered by the "*more general than*" relation whereby a substitution σ is said to be *more general than* a substitution θ if there exists a substitution δ such that $\theta = \sigma\delta \stackrel{\text{def}}{=} \delta \circ \sigma$ (see Appendix A).

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

in Logic Programming); *i.e.*, term constructor symbols related by a fuzzy equivalence relation. We first derive the operation that is the dual of so-called "weak" unification due to Maria Sessa in [12]. This dual operation corresponds to fuzzy generalization taking into account distinct but similar functors. We then extend the resulting fuzzy operations to yet a more expressive one tolerating not only distinct similar functors, but also distinct arities or misaligned argument positions. This last lattice captures a maximally expressive fuzzification of both operations of the Reynolds-Plotkin lattice. As a bonus, its formal declarative syntax-driven specification consists of logical axioms and rules over standard first-order terms and substitutions. This yields operational algorithms for these powerful fuzzy operations on a data structure used in a large number of Logic Programming idioms and systems.

Relation to other works

There have been other works dealing with the larger issue of integrating general equational theories into logical reasoning, not just the specific theory of fuzzy equivalence among terms. Among the most formally and operationally complete approach, pioneered by Goguen et al. in the seventies, is the set of works based on *initial algebras* [13].[5] Recall that an operator algebra is initial iff there exists a homomorphism from it to all other algebras that are semantic models of first-order terms freely built with these operators and variables [14]. Namely, initiality is the property that guarantees that the formal meaning of syntactic terms defined for $\mathcal{FOT}$s modulo congruence classes defined by equations is preserved for all interpretations. This result was later extended from equational systems to implicational systems by Mahr and Makowsky [15]. In this latter paper, it was also shown that Horn Logic (*i.e.*, an implicational system constituting the formal basis of the Prolog language), is the largest class of logic that admits an initial algebra semantics.

While the initial algebra approach has shown its general applicability for term unification and generalization, including more recently with the work of Alpuente et al. over order-sorted signatures [16] and with equational theories [17], its specific application to fuzzy congruences has yet to be done. While it could conceivably be specified by instantiating the general scheme of initial semantics, this must be at the expense of both formal and operational simplicity when compared with our approach which can be expressed as a direct extension of conventional operations of unification and generalization of first-order terms. This is because it is specifically adapted to a fuzzy equivalence of terms rather than general-purpose reasoning modulo equational theories. In the latter more general approach, equations can be used as term rewrite rules and unification modulo this theory is made operational using the general equation-solving technique known as "*narrowing*" [18], [19]. However, for this to work, one must have a terminating and confluent set of rewrite rules. Such may be derived in some cases based on undecidable procedures such as Knuth-Bendix completion [20], or unterminating term rewriting [21]. Rather, we limit ourselves to the obviously decidable theory generated homomorphically on $\mathcal{FOT}$s from a fuzzy equivalence relation (a similarity) on a finite signature. This follows the intuition behind Maria Sessa's formal work on fuzzy $\mathcal{FOT}$ unification as well [12]. Also, we support arity and argument position mismatch for similar operators.[6] Be that as it may, one could envisage studying how E-unification for common equational theories such as associativity or commutativity could be extended to unification of terms with similar functors. However, this is another issue and we do not do so in this document's setting.

There have been also works in logic-based databases such as using a similarity distance while comparing syntactically unequal terms; for example, Francesca Arcelli et al.'s LIKELOG database logic programming language [22], [23]. These works, however, concern only *ground* terms (*i.e.*, with no variables), not first-order terms (*i.e.*, possibly having variables). Arcelli et al.'s notion of similarity distance between terms was later extended from ground terms to first-order terms by Shroeder and Gilbert in the fuzzy logic programming language FURY [24], [25], [26]. They use the same concept as Arcelli et al.'s fuzzy equivalence but derived from dynamically evaluating so-called "*edit distance*" between strings on ground terms as well as first-order terms.[7] Thus, their objective is to derive dynamically an estimate of an "edit distance" between terms. The same comments also apply to work by Kutsia et al. [28], [29], where the objective is to check all the possibilities of dynamically matching $\mathcal{FOT}$s with *equal* function symbols having unspecified number of arguments (*i.e.*, the same sort of search objective pursued in FURY, where this is done for

[5] An initial algebra is also called *free* algebra, or *syntactic* algebra, or *tree* algebra, or *term* algebra—because its elements are the syntactic term structures one can define recursively by nesting terms as arguments of other terms (*i.e.*, the model taking as interpretation homomorphism the identity function on terms).

[6] See Section 4.1.2.

[7] See [27] this email discussion on the issue.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

unequal symbols as well). This objective is not ours in that we are not trying to infer dynamically distances between terms. Rather, we *assume given* a matrix of approximation degrees for such a static similarity relation on term constructors and use this information exactly in the same manner as done by Maria Sessa in [12]. In Sessa's context (and ours), this information is *given statically*, not inferred dynamically. Finally, the main advantage of working from a given static matrix of approximation degrees rather than estimating syntactic edit distances (whether dynamically or statically) is that similarity may be semantic among syntactically unrelated but semantically close strings. The context of dynamic syntactic distance estimation is typically for applications of purely lexical variants such as estimating gene similarity in biology [24]. Ours concerns deriving approximate solutions to fuzzy equations given similarity among term constructors [30].

Our last reference to other work related to unification and generalization of graph data and type structures, is the set of work due to Aït-Kaci et al. (*i.e.*, [31] and [32]). In this context, nodes denote sorts (that are organized in a lattice) and arrows denote features (functional attributes between sort nodes); variables denote equations among (possibly cyclic) feature paths. Just as such subsumption and its lattice operations on $\mathcal{FOT}$s can be fuzzified, so can indeed the lattice of Order-Sorted Feature terms. This is also relevant to non-aligned knowledge bases [33]. This is work we are currently pursuing soon to be published [34].

Organization of contents

The rest of this document is organized as follows. Section 2 recalls basic algebra on first-order terms and substitutions. Section 3 covers formal background on the lattice of first-order terms preordered with subsumption defined as variable instantiation by substitution application. The core of the paper is then developed in Section 4—Section 4.1 discusses fuzzy unification while fuzzy generalization is covered in Section 4.2. Section 5 concludes the paper.

This article's topic being at the junction of two well-developed independent formal computational domains (First-Order Term and Fuzzy Set algebras), readers more familiar with one domain than the other might feel short-changed if we assumed as background concepts from the field they are not, or only partially, knowledgeable about. For this reason, and in order to allow us to get to the heart of our work quickly without penalizing readers with varied backgrounds, we added an appendix summarizing all such needed material, terminology, and notation used in some essential way in this article.

2. First-order term algebra

2.1. First-order term

The first-order term ($\mathcal{FOT}$) was introduced as a data structure in software programming by the Prolog language.[8] Just like the S-expression for LISP, the $\mathcal{FOT}$ is Prolog's universal data structure. Using formal algebra notation, we write $\mathcal{T}_{\Sigma,\mathcal{V}}$ for the set of $\mathcal{FOT}$s on an operator signature $\Sigma \stackrel{\text{def}}{=} \bigcup_{n\geq 0} \Sigma_n$ where Σ_n is a set of n-ary operator symbols.[9] The set $\mathcal{V}$ is a countably infinite set of variables. Also following Prolog's tradition, we shall designate an element f in Σ as a *functor*, with $\mathbf{arity}(f)$ denoting its number of arguments.[10] This set $\mathcal{T}_{\Sigma,\mathcal{V}}$ can then be defined inductively as:

$$\mathcal{T}_{\Sigma,\mathcal{V}} \stackrel{\text{def}}{=} \mathcal{V} \cup \{\, f(t_1,\ldots,t_n) \mid f \in \Sigma_n,\ n \geq 0,\ \text{ and }\ t_i \in \mathcal{T}_{\Sigma,\mathcal{V}},\ 0 \leq i \leq n \,\}.$$

Technically, an additional condition of well-foundedness requires that $\Sigma_0 \neq \emptyset$. We write c instead of $c()$ for a constant $c \in \Sigma_0$. Also, when the set Σ of functor symbols and the set $\mathcal{V}$ of variables are implicit from the context, we simply write $\mathcal{T}$ instead of $\mathcal{T}_{\Sigma,\mathcal{V}}$.

The set $\mathbf{var}(t)$ of variables occurring in a $\mathcal{FOT}$ $t \in \mathcal{T}$ is defined as[11]:

$$\mathbf{var}(t) \stackrel{\text{def}}{=} \begin{cases} \{\, X \,\} & \text{if } t = X \in \mathcal{V} \\ \bigcup_{i=1}^{n} \mathbf{var}(t_i) & \text{if } t = f(t_1,\ldots,t_n). \end{cases}$$

8 https://en.wikipedia.org/wiki/Prolog.

9 We shall use the notation "$\stackrel{\text{def}}{=}$" to mean "*is defined as*."

10 When $\mathbf{arity}(f) = n$, this is sometimes denoted by writing f/n.

11 We shall use Prolog's convention of writing variables with capitalized symbols.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

A term t such that $\mathbf{var}(t) = \emptyset$ is called a *ground term*. We call $\mathcal{T}_0$ the subset of $\mathcal{T}$ of ground terms. The *depth* of a $\mathcal{FOT}$ t is a value in $\mathbb{N}$ defined inductively as:

$$\mathbf{depth}(t) \stackrel{\text{def}}{=} \begin{cases} 0 & \text{if } t \in \mathcal{V} \cup \Sigma_0; \\ 1 + \max_{i=1}^{n} \mathbf{depth}(t_i) & \text{if } t = f(t_1, \ldots, t_n) \text{ with } n > 0. \end{cases}$$

The **var** and **depth** notation is extended to a set of terms $T \subset \mathcal{T}$ as $\mathbf{var}(T) \stackrel{\text{def}}{=} \bigcup_{t \in T} \mathbf{var}(t)$ and $\mathbf{depth}(T) \stackrel{\text{def}}{=} \max_{t \in T} \mathbf{depth}(t)$.

2.2. Substitution

In order to express the notion of instance of a term, the concept of variable substitution σ is formalized as a functional mapping $\sigma : \mathcal{V} \to \mathcal{T}$ that is the identity function everywhere on $\mathcal{V}$ except on a finite set of n variables, $n \in \mathbb{N}$, written $\mathbf{dom}(\sigma) \stackrel{\text{def}}{=} \{ X_k \mid X_k \neq \sigma(X_k) \}_{k=1}^{k=n}$, and called the domain of σ. The range of a substitution σ is the set of terms in $\mathcal{T}$ defined as $\mathbf{ran}(\sigma) \stackrel{\text{def}}{=} \{ t \in \mathcal{T} \mid \exists X \in \mathbf{dom}(\sigma) \text{ such that } \sigma(X) = t \}$.

Such a mapping σ from V to $\mathcal{T}$ is then extended homomorphically to a mapping $\bar{\sigma}$ from $\mathcal{T}$ to $\mathcal{T}$ as follows:

$$\bar{\sigma}(t) \stackrel{\text{def}}{=} \begin{cases} \sigma(X) & \text{if } t = X \in \mathcal{V} \\ f(\bar{\sigma}(t_1), \ldots, \bar{\sigma}(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases} \tag{1}$$

which, because it coincides with σ on $\mathcal{V}$, will be written simply σ rather than $\bar{\sigma}$ even when applied to non-variable terms. In a similar fashion, substitutions may be applied to equations, as well as to sets of terms or equations in the obvious manner.

We shall denote as $\mathbf{SUBST}_{\mathcal{T}}$ the set of functions in $\mathcal{V} \to \mathcal{T}$ that are substitutions. Because it is non-identical only on a finite number of variables, we can express a substitution σ in $\mathbf{SUBST}_{\mathcal{T}}$ as a finite set of "*term/variable*" pairs of the form:

$$\sigma \stackrel{\text{def}}{=} \{ t_k / X_k \mid t_k = \sigma(X_k) \text{ and } X_k \neq \sigma(X_k) \}$$

associating each of a finite set of variables with a term not equal to it. Each pair t/X in a substitution's set notation is read "*term t is substituted for all occurrences of variable X*."

By tradition, rather than the prefix parenthesized notation usually used for functional application, substitution application to a term is written in postfix notation; *viz.*, $t\sigma$ instead of $\sigma(t)$. Thus, as defined by Expression (1), a substitution σ is a function in $\mathcal{T} \to \mathcal{T}$ mapping a term t into another one noted $t\sigma$, called its (σ-)*instance*, obtained after replacing all occurrences in t (if any) of variables in $\mathbf{dom}(\sigma)$, the domain of the substitution, by the term associated with this variable by σ. If $\mathbf{var}(\mathbf{ran}(\sigma)) = \emptyset$, σ is called a *ground substitution*, and for any term t in $\mathcal{T}$, $t\sigma \in \mathcal{T}_0$ and is called a *ground instance* of t.

We define the composition of two substitutions $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$ and $\theta \in \mathbf{SUBST}_{\mathcal{T}}$ seen as finite sets of non-identical term/variable pairs as the set of pairs written as $\sigma\theta$ and defined in terms of σ and θ as:

$$\begin{aligned} \sigma\theta \stackrel{\text{def}}{=} & \big(\{ t\theta/X \mid t/X \in \sigma \} \setminus \{ X/X \mid X \in \mathbf{dom}(\sigma) \}\big) \\ & \cup \\ & \big(\theta \setminus \{ u/Y \mid Y \in \mathbf{dom}(\sigma) \}\big). \end{aligned} \tag{2}$$

For terminology and proofs of formal properties of $\mathcal{FOT}$ substitutions as defined above and used in the remainder of this article, please refer to Appendix A.

3. First-order term subsumption lattice

The lattice-theoretic properties of $\mathcal{FOT}$s as data structures were initially and independently studied by Reynolds (in [5]) and Plotkin (in [35] and [6]). They noted that the set $\mathcal{T}$ is preordered by term subsumption (denoted as '$\preceq$'); *viz.*, $t \preceq t'$ (and we say: "*t' subsumes t*") iff there exists a variable substitution $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$ such that $t'\sigma = t$. Two $\mathcal{FOT}$s t and t' are considered "*equal up to variable renaming*" (denoted as $t \simeq t'$) whenever both $t \preceq t'$ and $t' \preceq t$.

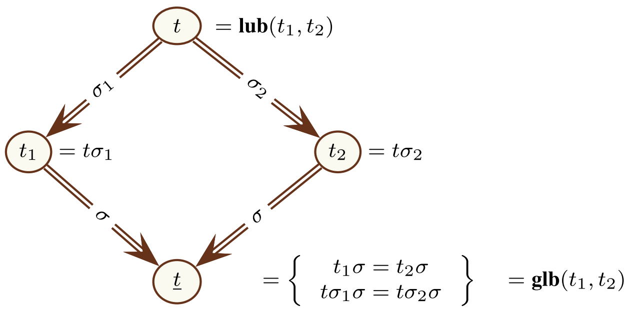


Fig. 1. Subsumption lattice operations.

Then, the quotient set of first-order terms modulo variable renaming augmented with a bottom element $\mathcal{T}_{/\simeq} \cup \{\bot_{\mathcal{T}}\}$ has a lattice structure for subsumption. It has a least element $\bot_{\mathcal{T}}$ that corresponds to no term in $\mathcal{T}$, since there exists no term that is an instance of all terms. It has a top element which is the set of all variables $\mathcal{V}$, since $\mathcal{V}$ is the class of any variable modulo renaming.

Unification corresponds to the greatest lower bound (**glb**) operation. This is the case also for failure of unification as in this case the **glb** operation results in $\bot_{\mathcal{T}}$. Given two $\mathcal{FOT}$s t_1 and t_2, unifying them is seeking to compute a most general substitution of their variables σ such that: $t_1\sigma = t_2\sigma$.[12] Such a substitution, when one exists, is not unique since any less general substitution verifies the equation; indeed, then $t_1\sigma\theta = t_2\sigma\theta$ for any $\theta \in \mathbf{SUBST}_{\mathcal{T}}$. We want only the most general such substitution. That is, for any other substitution $\theta \in \mathbf{SUBST}_{\mathcal{T}}$ such that $t_1\theta = t_2\theta$, then necessarily $\theta \preceq \sigma$. This is why it is called the Most General Unifier (**mgu**) of t_1 and t_2 [7]. If no such substitution exists, unification fails and returns $\bot_{\mathcal{T}}$ as the **glb** of t_1 and t_2, and no substitution. Formally, this is equivalent to instantiating the two terms with a bottom substitution $\bot_{\mathbf{SUBST}_{\mathcal{T}}}$ that is added to $\mathbf{SUBST}_{\mathcal{T}}$. This new substitution is a zero element in the quotient monoid of substitutions with composition. Namely, for all $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$, $\sigma\bot_{\mathbf{SUBST}_{\mathcal{T}}} = \bot_{\mathbf{SUBST}_{\mathcal{T}}}\sigma = \bot_{\mathbf{SUBST}_{\mathcal{T}}}$; which implies that $\bot_{\mathbf{SUBST}_{\mathcal{T}}} \preceq \sigma$ for all $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$. From this, it follows necessarily that, for all $t \in \mathcal{T}$, $t\bot_{\mathbf{SUBST}_{\mathcal{T}}} = \bot_{\mathcal{T}}$. Thus, when t_1 and t_2 are not unifiable, $\mathbf{mgu}(t_1, t_2) = \bot_{\mathbf{SUBST}_{\mathcal{T}}}$.

The dual operation, generalization of two terms, yields a term that is their least upper bound (**lub**) for subsumption. That is, it finds the most specific term t, and two most general substitutions σ_1 and σ_2 such that $t_i = t\sigma_i$ for $i = 1, 2$. Importantly, unlike unification, generalization cannot fail. This is because two term structures having different functors, or two unequal terms one of which is a variable, are always generalizable into a new variable (which may be construed as "*anything*"). Also, generalization yields *two* substitutions rather than just one like for unification. This is because a variable in the generalizing term t may correspond to two different instantiations in t_1 and t_2. Unification, on the other hand, seeks the *same* instantiation for all the variables in t_1 and t_2 to compute their most general common instance.

This can be summarized as the lattice diagram shown in Fig. 1. In this diagram, given a pair of terms $\langle t_1, t_2\rangle$, the pair of substitutions $\langle\sigma_1, \sigma_2\rangle$ are their respective most general generalizers, and the substitution σ is the pair's most general unifier (**mgu**).

Example 1. [***$\mathcal{FOT}$ lattice operations***] Consider the terms t_1 and t_2 defined as:

$$t_1 \stackrel{\text{def}}{=} f(a, g(X_1, b), Y_1, g(a, Y_1)),$$

$$t_2 \stackrel{\text{def}}{=} f(X_2, Y_2, g(X_2, g(X_2, b)), g(X_2, g(a, Z_2))).$$

Their most general unifier $\mathbf{mgu}(t_1, t_2)$ is the substitution σ given by:

$$\sigma = \{a/X_2, g(X_1, b)/Y_2, g(a, g(a, b))/Y_1, g(a, b)/Z_2\}$$

and so their greatest lower bound $\mathbf{glb}(t_1, t_2) = \underline{t}$ is given by:

$$\underline{t} = t_1\sigma = t_2\sigma = f(a, g(X_1, b), g(a, g(a, b)), g(a, g(a, g(a, b)))).$$

[12] See Appendix A, *First-Order Term Substitutions*.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

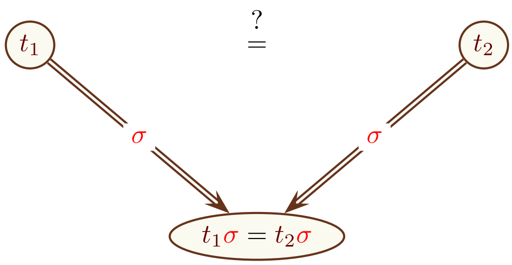


Fig. 2. $\mathcal{FOT}$ unification as a constraint.

Term Decomposition

$$\frac{E \cup \{ f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n) \}}{E \cup \{ s_1 \doteq t_1, \ldots, s_n \doteq t_n \}} \; [n \geq 0]$$

Variable Erasure

$$\frac{E \cup \{ X \doteq X \}}{E}$$

Variable Elimination

$$\frac{E \cup \{ X \doteq t \}}{E[X \leftarrow t] \cup \{ X \doteq t \}} \; \left[\begin{array}{l} X \notin \mathbf{var}(t) \\ X \text{ occurs in } E \end{array} \right]$$

Equation Orientation

$$\frac{E \cup \{ t \doteq X \}}{E \cup \{ X \doteq t \}} \; [t \notin \mathcal{V}]$$

Fig. 3. Herbrand-Martelli-Montanari unification rules.

Dually, their least upper bound $\mathbf{lub}(t_1, t_2) = t$ is given by $t = f(X, Y, Z, g(U, V))$, with their most general generalizers $\langle \sigma_1, \sigma_2 \rangle$ such that:

$$t_1 = t\sigma_1 \quad \text{with} \;\; \sigma_1 = \{ a/X, g(X_1, b)/Y, Y_1/Z, a/U, Y_1/V \}$$

$$t_2 = t\sigma_2 \quad \text{with} \;\; \sigma_2 = \{ X_2/X, Y_2/Y, g(X_2, g(X_2, b))/Z, X_2/U, g(a, Z_2)/V \}.$$

Next, we formalize these lattice operations on $\mathcal{FOT}$s by specifying them as declarative constraint normalization.

3.1. Unification rules

Fig. 2 illustrates $\mathcal{FOT}$ unification as a commutative diagram constraint. Solving such a constraint is done by a system of equation-normalization rules that we shall call Herbrand-Martelli-Montanari [1], [2]. These rules are given in Fig. 3. Each rule can be proven correct as a solution-preserving transformation of a set of equations. In Rule **Variable Elimination**, the notation $E[X \leftarrow t]$ denotes the set of equations E in which all occurrences of variable X have been replaced with the term t.

Thus, we can use these rules to unify two $\mathcal{FOT}$s t_1 and t_2, starting with the singleton set of equations $E \stackrel{\text{def}}{=} \{ t_1 \doteq t_2 \}$.[13] Then, we transform this set of equations using any applicable rule in any order until none applies. This always terminates into a finite set of equations E'. If all the equations in E' are of the form $X \doteq t$ with X occurring nowhere else in E', then this is a most general unifying substitution (up to consistent variable renaming) $\sigma \stackrel{\text{def}}{=} \{ \, t/X \mid X \doteq t \in E' \, \}$ solving the original equation (*i.e.*, $t_1\sigma = t_2\sigma$); otherwise, there is no solution.

In the rules of Fig. 3, Rule **Variable Elimination** has the side condition $X \notin \mathbf{var}(t)$ to prevent cyclic terms (such as, *e.g.*, $X = f(X)$) whose presence indicates no $\mathcal{FOT}$ solutions. This condition could be omitted if wished, thus extending the set of $\mathcal{FOT}$s and solutions of equations to rational $\mathcal{FOT}$s—also called "*infinite trees*" (see, *e.g.*, [36], [37], [38]).

[13] In such equations, we use the notation $t_1 \doteq t_2$ not to confuse it with the equality symbol "=" (at the meta-level).

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Example 2 *($\mathcal{FOT}$ unification).* Consider the equation set $\{t_1 \doteq t_2\}$ for the terms t_1 and t_2 of Example 1:

$$\{ f(a, g(X_1, b), Y_1, g(a, Y_1)) \doteq f(X_2, Y_2, g(X_2, g(X_2, b)), g(X_2, g(a, Z_2))) \}$$

and let us apply the rules of Fig. 3:

- Rule **Term Decomposition**:

$$\{ a \doteq X_2, g(X_1, b) \doteq Y_2, Y_1 \doteq g(X_2, g(X_2, b)), g(a, Y_1) \doteq g(X_2, g(a, Z_2)) \};$$

- Rule **Equation Orientation** to $a \doteq X_2$:

$$\{ X_2 \doteq a, g(X_1, b) \doteq Y_2, Y_1 \doteq g(X_2, g(X_2, b)), g(a, Y_1) \doteq g(X_2, g(a, Z_2)) \};$$

- Rule **Variable Elimination** to $X_2 \doteq a$:

$$\{ X_2 \doteq a, g(X_1, b) \doteq Y_2, Y_1 \doteq g(a, g(a, b)), g(a, Y_1) \doteq g(a, g(a, Z_2)) \};$$

- Rule **Equation Orientation** to $g(X_1, b) \doteq Y_2$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), g(a, Y_1) \doteq g(a, g(a, Z_2)) \};$$

- Rule **Variable Elimination** to $Y_1 \doteq g(a, g(a, b))$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), g(a, g(a, g(a, b))) \doteq g(a, g(a, Z_2)) \};$$

- Rule **Term Decomposition** to $g(a, g(a, g(a, b))) \doteq g(a, g(a, Z_2))$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), a \doteq a, g(a, g(a, b)) \doteq g(a, Z_2) \};$$

- Rule **Term Decomposition** to $a \doteq a$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), g(a, g(a, b)) \doteq g(a, Z_2) \};$$

- Rule **Term Decomposition** to $g(a, g(a, b)) \doteq g(a, Z_2)$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), a \doteq a, g(a, b) \doteq Z_2 \};$$

- Rule **Term Decomposition** to $a \doteq a$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), g(a, b) \doteq Z_2 \};$$

- Rule **Equation Orientation** to $g(a, b) \doteq Z_2$:

$$\{ X_2 \doteq a, Y_2 \doteq g(X_1, b), Y_1 \doteq g(a, g(a, b)), Z_2 \doteq g(a, b) \}.$$

This last equation set is in normal form defining the substitution

$$\sigma = \{a/X_2, g(X_1, b)/Y_2, g(a, g(a, b))/Y_1, g(a, b)/Z_2\}.$$

So the greatest lower bound $\underline{t} \overset{\text{def}}{=} \mathbf{glb}(t_1, t_2)$ of:

$$t_1 \overset{\text{def}}{=} f(a, g(X_1, b), Y_1, g(a, Y_1))$$

and:

$$t_2 \overset{\text{def}}{=} f(X_2, Y_2, g(X_2, g(X_2, b)), g(X_2, g(a, Z_2)))$$

is given by:

$$\underline{t} = t_1\sigma = t_2\sigma = f(a, g(X_1, b), g(a, g(a, b)), g(a, g(a, g(a, b)))).$$

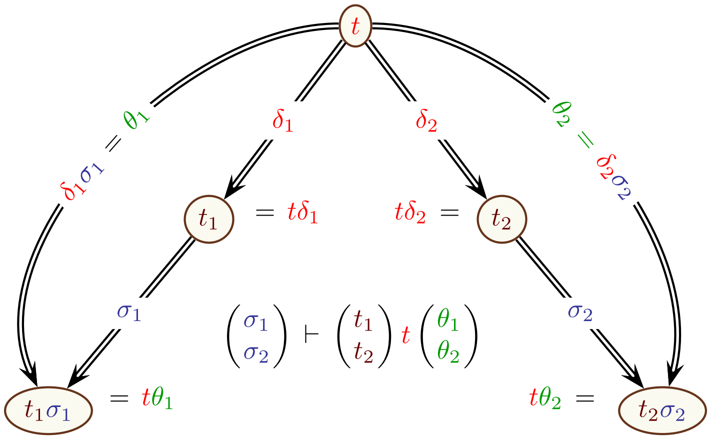


Fig. 4. $\mathcal{FOT}$ generalization judgment validity as a constraint.

3.2. Generalization rules

Next, we present a set of constraint normalization rules for $\mathcal{FOT}$ generalization which are equivalent to the procedural method of Reynolds and Plotkin. The advantage of specifying this operation in this manner rather than procedurally as done originally by Reynolds and Plotkin is that each rule or axiom relates a pair of prior substitutions to a pair of posterior substitutions based only on local syntactic-pattern properties of the terms to generalize, and this without resorting to side-effects on global structures. In this way, the terms and substitutions involved are derived as solutions of logical syntactic constraints. In addition, correctness of the so-specified operation is made much easier to establish since we only need to prove each rule's correctness independently of that of the others. Finally, the rules also provide an effective means for the derivation of an operational semantics for the so-specified operation by constraint solving, without need for control specification as any applicable rule may be invoked in any order.[14]

Definition 1 *(*GENERALIZATION JUDGMENT*). A generalization judgment is an expression of the form:*

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}t_1\\ t_2\end{pmatrix} t \begin{pmatrix}\theta_1\\ \theta_2\end{pmatrix} \tag{3}$$

where $\sigma_i \in \textbf{SUBST}_{\mathcal{T}}$, $\theta_i \in \textbf{SUBST}_{\mathcal{T}}$, $t_i \in \mathcal{T}$ ($i = 1, 2$), and $t \in \mathcal{T}$.

Informally, it reads: "*given two prior substitutions σ_1 and σ_2, the term t is the least generalization of terms $t_1\sigma_1$ and $t_2\sigma_2$ with posterior substitutions θ_1 and θ_2.*" How all the constituents of such a generalization judgment must be related to constitute what we shall consider a valid judgment, is defined next.

Definition 2 *(*GENERALIZATION JUDGMENT VALIDITY*). A $\mathcal{FOT}$ generalization judgment such as (3) is said to be valid whenever, for $i = 1, 2$:*

1. *$t_i\sigma_i = t\theta_i$; and,*
2. *$\exists\delta_i \in \textbf{SUBST}_{\mathcal{T}}$ s.t. $t_i = t\delta_i$ and $\theta_i = \delta_i\sigma_i$ (i.e., $t_i \preceq t$ and $\theta_i \preceq \sigma_i$).*

Fig. 4 illustrates the validity of a $\mathcal{FOT}$ generalization judgment as a commutative diagram constraint.

[14] Such as the Herbrand-Martelli-Montanari rules w.r.t. to Robinson's procedural unification algorithm.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Definition 3 *(*TRIVIAL $\mathcal{FOT}$ GENERALIZATION JUDGMENT*). The $\mathcal{FOT}$ generalization judgment:*

$$\mathfrak{true} \stackrel{def}{=} \begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} t \\ t \end{pmatrix} t \begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \tag{4}$$

where t is an arbitrary term in $\mathcal{T}$ is called a "trivial $\mathcal{FOT}$ generalization judgment."

Lemma 1 *(*TRIVIAL $\mathcal{FOT}$ GENERALIZATION JUDGMENT VALIDITY*). The trivial $\mathcal{FOT}$ generalization judgment $\mathfrak{true}$ is always valid.*

Proof. This follows from Definition 2 since in this particular case the equations of the first condition of Definition (2) becomes $t = t$, which is trivially true for any term $t \in \mathcal{T}$. □

Contrary to unification normalization rules which are expressed as conditional rewrite rules whereby a prior form (the "numerator") is related to a posterior form (the "denominator"), these normalization rules are more naturally rendered as (conditional) Horn clauses of judgments. This is formally as convenient as rewrite rules since a Prolog-like operational semantics can then readily provide an effective interpretation.[15] Thus, a generalization rule is of the form:

$$\frac{\begin{matrix}[\phi] \\ J_1 \dots J_n\end{matrix}}{J} \tag{5}$$

where ϕ is an optional side meta-condition, and J, $J_1, \dots, J_n$ are judgments, and it reads, "*whenever the side condition ϕ holds, if all the n antecedent judgments $J_1, \dots, J_n$ are valid, then the consequent judgment J is also valid.*" Such a generalization rule without a specified antecedent (a "numerator") is called a "*generalization axiom.*" Such an axiom is said to be valid iff its consequent (the "denominator") is valid whenever its optional side condition holds. It is equivalent to a rule where the only antecedent is the trivial generalization judgment $\mathfrak{true}$.

Definition 4 *(*GENERALIZATION RULE CORRECTNESS*). A generalization rule such as Rule (5) is correct iff J_k is a valid judgment for all $k = 1, \dots, n$ implies that J is a valid judgment, whenever the side condition ϕ holds.*

Given t_1 and t_2 two $\mathcal{FOT}$s, in order to find the most specific term t and most general substitutions σ_i, $i = 1, 2$, such that $t\sigma_i = t_i$, $i = 1, 2$, one needs to establish the generalization judgment:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} t \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}. \tag{6}$$

In other words, this expresses the upper half of Fig. 1 whereby $t = \mathbf{lub}(t_1, t_2)$, with most general substitutions σ_1 and σ_2. We give a complete set of normalization axioms and rule for generalization for all syntactic patterns in Fig. 5.

Rule "**EQUAL FUNCTORS**" specifies a sequence of judgments constrained as a sequence. It does so exactly as a so-called "Definite Clause Grammar" (or DCG) rule.[16] This rule uses an "*unapply*" operation ('$\uparrow$') on a pair of terms (t_1, t_2) given a pair of substitutions (σ_1, σ_2). It may be conceived as (and in fact is) the result of simultaneously "*unapplying*" σ_i from t_i into a common variable X only if such X is bound to t_i by σ_i, for $i = 1, 2$. If there is no such a variable, it is the identity. This operation avoids the introduction of a new variable when generalizing two already generalized terms. Formally, this is defined as:

[15] This operational semantics is also efficient because it does not need backtracking as long as the complete set of conditions of a ruleset covers all but mutually exclusive syntactic patterns.

[16] A DCG rule (see https://www.metalevel.at/prolog/dcg) is a Horn rule expressing constraints on a sequence of words constituting a sentence. The judgment sequencing in the rules we define uses exactly the same kind of constraint: the posterior pair of substitutions of a judgment must match the prior pair of substitutions of the judgment following it. Still, contrary to a DCG rule that constrains an *ordered* sequence of constituents, the order of constraints on the antecedent judgments on the arguments is arbitrary. We choose the same order as that of the arguments as it is the most natural, but it could be any of its permutations as long as the sequence's posterior/prior constraints are consistent with the chosen argument ordering.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Equal Variables

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}X\\ X\end{pmatrix} X \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}$$

Variable-Term

$[t_1 \in \mathcal{V} \text{ or } t_2 \in \mathcal{V};\ \ t_1 \neq t_2;\ \ X \text{ is new}]$

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}t_1\\ t_2\end{pmatrix} X \begin{pmatrix}\sigma_1\{t_1/X\}\\ \sigma_2\{t_2/X\}\end{pmatrix}$$

Unequal Functors

$[m \geq 0, n \geq 0;\ \ m \neq n \text{ or } f \neq g;\ \ X \text{ is new}]$

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}f(s_1,\ldots,s_m)\\ g(t_1,\ldots,t_n)\end{pmatrix} X \begin{pmatrix}\sigma_1\{f(s_1,\ldots,s_m)/X\}\\ \sigma_2\{g(t_1,\ldots,t_n)/X\}\end{pmatrix}$$

Equal Functors

$[n \geq 0]$

$$\frac{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}s_1\\ t_1\end{pmatrix}\uparrow\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1\\ \sigma_2^1\end{pmatrix} \quad \ldots \quad \begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix} \vdash \begin{pmatrix}s_n\\ t_n\end{pmatrix}\uparrow\begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix} u_n \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}}{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}f(s_1,\ldots,s_n)\\ f(t_1,\ldots,t_n)\end{pmatrix} f(u_1,\ldots,u_n) \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}}$$

Fig. 5. Generalization axioms and rule.

$$\begin{pmatrix}t_1\\ t_2\end{pmatrix}\uparrow\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \stackrel{\text{def}}{=} \begin{cases}\begin{pmatrix}X\\ X\end{pmatrix} & \text{if } \exists X \in \mathcal{V},\ t_i = X\sigma_i,\ \text{for } i = 1,2;\\[2ex] \begin{pmatrix}t_1\\ t_2\end{pmatrix} & \text{otherwise.}\end{cases} \tag{7}$$

Note also that Rule "**Equal Functors**" is defined for $n \geq 0$. For $n = 0$, it becomes the following axiom for any constant c and any two substitutions σ_i, $i = 1, 2$:

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}c\\ c\end{pmatrix} c \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}. \tag{8}$$

Referring to the axioms (seen as rules with no antecedent) and the rule of Fig. 5, we first establish the following fact.

Lemma 2. *In Rule* **Equal Functors** *of Fig. 5, taking $\sigma_i^0 \stackrel{def}{=} \sigma_i$, for $i = 1, 2$, the substitutions $\sigma_i^0, \ldots, \sigma_i^n$ are such that, for all k, $1 \leq k \leq n$, $\sigma_i^k \preceq \sigma_i^{k-1}$, for $i = 1, 2$.*

Proof. We proceed by induction on the depth d of the terms; *i.e.*, we consider only terms of depth less than or equal to d.

1. $d = 0$: This limits terms to constants and variables. The inequality between prior and posterior substitutions is verified for the three first axioms: the posterior substitutions are all either equal to the corresponding prior substitutions or of the form $\theta = \sigma\{t/X\}$ where X is a new variable and σ is the corresponding prior substitution; that is, $\theta \preceq \sigma$. As well, when limited to terms of 0 depth, Rule **Equal Functors** becomes the single judgment Axiom (8), which preserves the substitutions.
2. $d > 0$: Let us assume that this is true for all terms of depth strictly less than d. We now consider two terms at least one of which is of depth d. For Axiom **Equal Variables**, the same argument given above for the case $d = 0$ justifies concluding that $\theta \preceq \sigma$, since then $\theta = \sigma$. For Axiom **Variable-Term** and Axiom **Unequal Functors**, this true for terms t_1 and t_2 of any depth since the posterior substitutions are both less general than the corresponding prior substitutions. As for Rule **Equal Functors**, there are two possible cases for the generalized terms in its consequent (the "denominator"):

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

(a) $n = 0$: then, the conclusion follows true by Axiom (8).
(b) $n \geq 0$: since the unapply operation (7) yields either a pair of terms having the same depth as the corresponding terms it is applied to, or 0 (because it can only be a new variable), we can say that all the terms of unapplied pairs of arguments in the judgments of the rule's antecedent (the "numerator") are of depth at most $d - 1$. Therefore, all the terms in the n antecedent judgments verify our inductive hypothesis; namely: $\sigma_i^k \preceq \sigma_i^{k-1}$, for all $k = 1, \ldots, n$. Then, by transitivity of the "more general" ordering on substitutions, the conclusion follows.

Hence, this establishes that, for both $i = 1, 2$, σ_i^k is monotonically refined from more general to less as k increases from 1 to n. □

From this lemma, the following corollary follows by transitivity of the $\preceq$ preorder on substitutions.

Corollary 1. *In Rule* Equal Functors, *the substitutions* σ_i^k *are such that, for all* k, $1 \leq k \leq n$, $\sigma_i^n \preceq \sigma_i^{n-1} \preceq \ldots \preceq \sigma_i^1 \preceq \sigma_i^0$, *for* $i = 1, 2$.

It is also verified in the proof of the following theorem.

Theorem 1. *The axioms and the rule of Fig. 5 are correct.*

Proof. We must show that they verify the conditions of Definition 4. For each of the three axioms of Fig. 5 this means that they must be always valid as judgments, verifying the conditions of Definition 2, which are:

– *Condition 1*: $t_i\sigma_i = t\theta_i$,
– *Condition 2*: $t_i \preceq t$ and $\theta_i \preceq \sigma_i$

for $i = 1, 2$, for a generalization judgment such as (3) in Definition 1. These conditions for the axioms and the rule of Fig. 5 translate as the following.

Condition 1.

– Equal Variables: it amounts to the two identities $X\sigma_i = X\sigma_i$, $i = 1, 2$;
– Variable-Term: it amounts to the two identities $t_i\sigma_i = t_i\sigma_i$, $i = 1, 2$;
– Unequal Functors: it amounts to the two equations:

$$\begin{array}{rcl} f(s_1, \ldots, s_n)\sigma_1 & = & X\sigma_1\{f(s_1, \ldots, s_n)/X\}, \\ g(t_1, \ldots, t_n)\sigma_2 & = & X\sigma_2\{g(t_1, \ldots, t_n)/X\}, \end{array}$$

which, because X is a new variable that does not occur in either σ_1 or σ_2, can be simplified to the identities:

$$\begin{array}{rcl} f(s_1, \ldots, s_n) & = & f(s_1, \ldots, s_n), \\ g(t_1, \ldots, t_n) & = & g(t_1, \ldots, t_n). \end{array}$$

Condition 2. All threes cases are tautologies:

– Equal Variables: $X \preceq X$ and $\sigma_i \preceq \sigma_i$, $i = 1, 2$;
– Variable-Term: $t_i \preceq X$ and $\sigma_i\{t_i/X\} \preceq \sigma_i$, $i = 1, 2$;
– Unequal Functors:

$$\begin{array}{rcl} f(s_1, \ldots, s_n) & \preceq & X \text{ and } \sigma_1\{f(s_1, \ldots, s_n)/X\} \preceq \sigma_1, \\ g(s_1, \ldots, s_n) & \preceq & X \text{ and } \sigma_2\{g(s_1, \ldots, s_n)/X\} \preceq \sigma_2. \end{array}$$

As for Rule Equal Functors, as required by Definition 4, we must show that if all the judgments in the numerator are valid, then the judgment in the denominator must be valid too. Let us proceed by induction on the argument-position number k, for $k = 1, \ldots, n$.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

For $n = 0$, this rule becomes Axiom (8), a judgment that is trivially valid since the conditions of Definition 2 become the identity $c = c$, the term inequality $c \preceq c$, and the substitution inequalities $\sigma_i \preceq \sigma_i$, for $i = 1, 2$.

For $n > 0$, a fuzzy judgment in the rule's antecedent, for each argument-position $k = 1, \ldots, n$, is of the form:

$$\begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} \vdash \begin{pmatrix} s_k \\ t_k \end{pmatrix} \uparrow \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} u_k \begin{pmatrix} \sigma_1^{k} \\ \sigma_2^{k} \end{pmatrix}$$

that is, the form given by Definition 1, whose formal validity conditions are given by Definition 2, which in the above case is equivalent to:

$$\begin{pmatrix} v_1^k \\ v_2^k \end{pmatrix} \stackrel{\text{def}}{=} \begin{pmatrix} s_k \\ t_k \end{pmatrix} \uparrow \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} \text{and} \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} \vdash \begin{pmatrix} v_1^k \\ v_2^k \end{pmatrix} u_k \begin{pmatrix} \sigma_1^{k} \\ \sigma_2^{k} \end{pmatrix}.$$

Let us now assume that all the judgments in the rule's antecedent are valid. That is, for $k = 1, \ldots, n$, for $i = 1, 2$ (taking $\sigma_i^0 \stackrel{\text{def}}{=} \sigma_i$):

– Condition 1 of Definition 2 holds:

$$u_k \sigma_i^k \;=\; v_i^k \sigma_i^{k-1}; \tag{9}$$

– Condition 2 of Definition 2 holds:

$$v_i^k \preceq u_k \text{and} \sigma_i^k \preceq \sigma_i^{k-1}. \tag{10}$$

Condition 1. By Equation (7), this means that for all $k = 1, \ldots, n$:

$$\begin{pmatrix} v_1^k \\ v_2^k \end{pmatrix} = \begin{cases} \begin{pmatrix} X \\ X \end{pmatrix} & \text{if } s_k = X\sigma_1^{k-1} \text{ and } t_k = X\sigma_2^{k-1} \text{ for some variable} X; \\[2ex] \begin{pmatrix} s_k \\ t_k \end{pmatrix} & \text{otherwise.} \end{cases}$$

In other words, for each $k = 1, \ldots, n$, there are two cases:

1. $s_k = X\sigma_1^{k-1}$ and $t_k = X\sigma_2^{k-1}$ for some variable X; then, by Axiom **Equal Variables**, we must have $u_k = X$, and $\sigma_i^k = \sigma_i^{k-1}$ for $i = 1, 2$; and therefore Equations (9) entail:

$$\begin{aligned} s_k\sigma_1^{k-1} \;&=\; X\sigma_1^{k-1}\sigma_1^{k-1} \;=\; X\sigma_1^{k-1} \;=\; X\sigma_1^k \;=\; u_k\sigma_1^k \\ t_k\sigma_2^{k-1} \;&=\; X\sigma_2^{k-1}\sigma_2^{k-1} \;=\; X\sigma_2^{k-1} \;=\; X\sigma_2^k \;=\; u_k\sigma_2^k. \end{aligned}$$

2. There is no such variable X; in which case, Equations (9) also become:

$$\begin{aligned} s_k\sigma_1^{k-1} \;&=\; u_k\sigma_1^k \\ t_k\sigma_2^{k-1} \;&=\; u_k\sigma_2^k. \end{aligned}$$

Thus, by the only non-identical transformation relating prior and posteriors substitutions in the axioms, for any argument position k, $1 \le k \le n$, we have:

$$\sigma_i^k \;=\; \sigma_i^0\{\tau_1/X_1\}\ldots\{\tau_\ell/X_\ell\}$$

where each of the variables $X_1 \ldots X_\ell$, with $0 \le \ell$, is a variable possibly introduced in the validity of the judgment corresponding to some argument preceding position k. Therefore, for any argument position k, $1 \le k \le n$:

$$\begin{aligned} s_k\sigma_1^0 \;&=\; s_k\sigma_1^1 \;=\; \ldots \;=\; s_k\sigma_1^{k-1} \\ t_k\sigma_2^0 \;&=\; t_k\sigma_2^1 \;=\; \ldots \;=\; t_k\sigma_2^{k-1} \end{aligned}$$

as well as:

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

$$u_k\sigma_1^k = u_k\sigma_1^{k+1} = \ldots = u_k\sigma_1^n$$
$$u_k\sigma_2^k = u_k\sigma_2^{k+1} = \ldots = u_k\sigma_2^n$$

because σ_i^k affects only new variables introduced in some axioms verifying the validity of a subterm of argument at position k; and because the same variable in u_k is always instantiated by the same term, and thus as well all at higher argument positions.

This means that in both cases we have, for all $k = 1, \ldots, n$:

$$s_k\sigma_1^0 = u_k\sigma_1^n,$$
$$t_k\sigma_2^0 = u_k\sigma_2^n.$$

Therefore, for $k = n$:

$$f(s_1, \ldots, s_n)\sigma_1^0 = f(u_1, \ldots, u_n)\sigma_1^n,$$
$$f(t_1, \ldots, t_n)\sigma_2^0 = f(u_1, \ldots, u_n)\sigma_2^n.$$

This proves Condition 1.

Condition 2. By transitivity of the $\leq$ ordering on approximation degrees and that of the $\preceq$ preorder on $\mathbf{SUBST}_{\mathcal{T}}$, both parts of Condition 2 of our induction hypothesis (10) implies that $f(s_1, \ldots, s_n) \preceq f(u_1, \ldots, u_n)$, $f(t_1, \ldots, t_n) \preceq f(u_1, \ldots, u_n)$ and $\sigma_i \preceq \sigma_i^n$, for $i = 1, 2$, which completes the proof. □

In particular, with empty prior substitutions, we obtain the following corollary.

Corollary 2 *($\mathcal{FOT}$ Generalization). Whenever the judgment*

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} \vdash \begin{pmatrix}t_1\\t_2\end{pmatrix} t \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}$$

is valid, then $t\sigma_i = t_i$, for $i = 1, 2$.

Example 3 ***(Generalization of ground $\mathcal{FOT}$s).*** Let us now consider the terms $f(a, g(b, a), b))$ and $f(b, g(a, b), a))$.

- Let us find term t and substitutions σ_1 and σ_2 such that $t\sigma_1 = f(a, g(b, a), b))$ and $t\sigma_2 = f(b, g(a, b), a)$; *i.e.*, let us try to solve the following constraint problem:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} \vdash \begin{pmatrix}f(a, g(b, a), b)\\f(b, g(a, b), a)\end{pmatrix} t \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}$$

- By Rule **Equal Functors**, we must have $t = f(u_1, u_2, u_3)$ since:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} \vdash \begin{pmatrix}f(a, g(b, a), b)\\f(b, g(a, b), a)\end{pmatrix} f(u_1, u_2, u_3) \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix} \quad \text{where:}$$

  - u_1 is the generalization of $\begin{pmatrix}a\\b\end{pmatrix} \uparrow \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}$; that is of a and b; and by Rule **Unequal Functors**:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} \vdash \begin{pmatrix}a\\b\end{pmatrix} X \begin{pmatrix}\{a/X\}\\\{b/X\}\end{pmatrix} \quad \text{and therefore } u_1 = X;$$

  - $u_2 = g(v_1, v_2)$ is the generalization of $\begin{pmatrix}g(b, a)\\g(a, b)\end{pmatrix} \uparrow \begin{pmatrix}\{a/X\}\\\{b/X\}\end{pmatrix}$; that is, of $g(b, X)$ and $g(a, X)$; and by Rule **Equal Functors**:
    - ∗ v_1 is the generalization of $\begin{pmatrix}b\\a\end{pmatrix} \uparrow \begin{pmatrix}\{a/X\}\\\{b/X\}\end{pmatrix}$; that is, of b and a; and by Rule **Unequal Functors**:

$$\begin{pmatrix}\{a/X\}\\\{b/X\}\end{pmatrix} \vdash \begin{pmatrix}b\\a\end{pmatrix} Y \begin{pmatrix}\{a/X, b/Y\}\\\{b/X, a/Y\}\end{pmatrix} \quad \text{so } v_1 = Y;$$

$*$ v_2 is the generalization of $\begin{pmatrix} a \\ b \end{pmatrix} \uparrow \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix}$; that is, of X and X; and by Rule **Equal Variables**:

$$\begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix} \vdash \begin{pmatrix} X \\ X \end{pmatrix} X \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix} \text{ so } v_2 = X;$$

therefore:

$$\begin{pmatrix} \{a/X\} \\ \{b/X\} \end{pmatrix} \vdash \begin{pmatrix} g(b, X) \\ g(a, X) \end{pmatrix} g(Y, X) \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix} \text{ so } u_2 = g(Y, X);$$

– u_3 is the generalization of $\begin{pmatrix} b \\ a \end{pmatrix} \uparrow \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix}$; that is, of Y and Y; and by Rule **Equal Variables**:

$$\begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix} \vdash \begin{pmatrix} Y \\ Y \end{pmatrix} Y \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix} \text{ so } u_3 = Y;$$

- therefore, this yields:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} f(a, g(b, a), b) \\ f(b, g(a, b), a) \end{pmatrix} f(X, g(Y, X), Y) \begin{pmatrix} \{a/X, b/Y\} \\ \{b/X, a/Y\} \end{pmatrix}$$

that is $t = f(X, g(Y, X), Y)$ with $\sigma_1 = \{a/X, b/Y\}$ such that $t\sigma_1 = f(a, g(b, a), b)$, and $\sigma_2 = \{b/X, a/Y\}$ such that $t\sigma_2 = f(b, g(a, b), a)$.

Example 4 ***(Generalization of non-ground*** $\mathcal{FOT}$***s)***. Let us apply the $\mathcal{FOT}$ generalization axioms and rules of Fig. 5 to the following $\mathcal{FOT}$s:

$$t_1 \stackrel{\text{def}}{=} h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)), \text{ and } t_2 \stackrel{\text{def}}{=} h(X_2, X_2, g(c, d)).$$

- Let us find term t and substitutions σ_1 and σ_2 such that $t\sigma_1 = h(f(a, X_1), g(X_1, b), f(Y_1, Y_1))$ and $t\sigma_2 = h(X_2, X_2, g(c, d))$; that is, let us try to solve the constraint problem:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)) \\ h(X_2, X_2, g(c, d)) \end{pmatrix} t \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}.$$

- By Rule **Equal Functors**, we must have $t = h(u_1, u_2, u_3)$ since:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)) \\ h(X_2, X_2, g(c, d)) \end{pmatrix} h(u_1, u_2, u_3) \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} \text{ where:}$$

– u_1 is the generalization of $\begin{pmatrix} f(a, X_1) \\ X_2 \end{pmatrix} \uparrow \begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}$; that is of $f(a, X_1)$ and X_2; and by Rule **Variable-Term**:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} f(a, X_1) \\ X_2 \end{pmatrix} X \begin{pmatrix} \{f(a, X_1)/X\} \\ \{X_2/X\} \end{pmatrix} \text{ so } u_1 = X;$$

– u_2 is the generalization of $\begin{pmatrix} g(X_1, b) \\ X_2 \end{pmatrix} \uparrow \begin{pmatrix} \{f(a, X_1)/X\} \\ \{X_2/X\} \end{pmatrix}$; that is, of $g(X_1, b)$ and X_2; and by Rule **Variable-Term**:

$$\begin{pmatrix} \{f(a, X_1)/X\} \\ \{X_2/X\} \end{pmatrix} \vdash \begin{pmatrix} g(X_1, b) \\ X_2 \end{pmatrix} Y \begin{pmatrix} \{\ldots, g(X_1, b)/Y\} \\ \{\ldots, X_2/Y\} \end{pmatrix} \text{ so } u_2 = Y;$$

– u_3 is the generalization of $\begin{pmatrix} f(Y_1, Y_1) \\ g(c, d) \end{pmatrix} \uparrow \begin{pmatrix} \{f(a, X_1)/X, g(X_1, b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix}$; that is, of $f(Y_1, Y_1)$ and $g(c, d)$; and by Rule **Unequal Functors**:

$$\begin{pmatrix} \{f(a, X_1)/X, g(X_1, b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix} \vdash \begin{pmatrix} f(Y_1, Y_1) \\ g(c, d) \end{pmatrix} Z \begin{pmatrix} \{\ldots, f(Y_1, Y_1)/Z\} \\ \{\ldots, g(c, d)/Z\} \end{pmatrix}$$

and so $u_3 = Z$;

- therefore, this yields:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} h(f(a,X_1),g(X_1,b),f(Y_1,Y_1)) \\ h(X_2,X_2,g(c,d)) \end{pmatrix} h(X,Y,Z) \begin{pmatrix} \{\ldots, f(Y_1,Y_1)/Z\} \\ \{\ldots, g(c,d)/Z\} \end{pmatrix}$$

that is, $t = h(X,Y,Z)$ with $\sigma_1 = \{f(a,X_1)/X, g(X_1,b)/Y, f(Y_1,Y_1)/Z\}$ such that $t\sigma_1 = h(f(a,X_1), g(X_1,b), f(Y_1,Y_1))$, and: $\sigma_2 = \{X_2/X, X_2/Y, g(c,d)/Z\}$ such that $t\sigma_2 = h(X_2,X_2,g(c,d))$.

4. Fuzzy lattice operations on first-order terms

For the formal Fuzzy Algebra notation and terminology that we use in the remainder of this work, see Appendix C.

4.1. Fuzzy unification

4.1.1. Sessa's weak unification

A fuzzy unification operation on $\mathcal{FOT}$s, dubbed "*weak unification*," was proposed by Maria Sessa in [12] which consists in normalizing fuzzy equations between conventional $\mathcal{FOT}$s modulo a similarity relation $\sim$ over functor symbols [39]. This similarity relation is then homomorphically extended to one over all $\mathcal{FOT}$s.

Example 5. [*Functor Similarity Matrix*] Given a similarity relation $\sim$ (*i.e.*, a fuzzy equivalence) on a finite signature $\Sigma = \cup_n \Sigma_n$, as explained in [39], we employ the notation $x \sim_d y$ to denote that the similarity degree between x and y as defined by the similarity relation $\sim$ is d. The similarity relation $\sim$ can be represented as a matrix in $\Sigma \times \Sigma \to [0,1]$. For example, if the signature Σ is the union of $\Sigma_0 = \{a,b,c,d\}$, $\Sigma_2 = \{f,g\}$, $\Sigma_3 = \{h\}$, and $\Sigma_n = \emptyset$ otherwise ($n = 1$ or $n \geq 4$), and with a similarity that is the reflexive, symmetric, and transitive closure of the pairs $a \sim_{0.7} b$, $c \sim_{0.6} d$, and $f \sim_{0.9} g$. This corresponds to the similarity matrix whose rows and columns are indexed by elements of Σ:

$\sim \stackrel{\text{def}}{=}$

	a	b	c	d	f	g	h
a	1	0.7	0	0	0	0	0
b	0.7	1	0	0	0	0	0
c	0	0	1	0.6	0	0	0
d	0	0	0.6	1	0	0	0
f	0	0	0	0	1	0.9	0
g	0	0	0	0	0.9	1	0
h	0	0	0	0	0	0	1

Following Maria Sessa's formal setting [12], we assume given such a similarity relation between functors of equal arity (*i.e.*, which admit the same number of arguments). Upon this basis, this similarity can be extended homomorphically from functors to $\mathcal{FOT}$s as follows. Let $\sim$ be a similarity on functors of equal arity in a signature Σ.

Definition 5 (Sessa's $\mathcal{FOT}$ Similarity). *The fuzzy relation $\sim^{\mathcal{T}}: \mathcal{T}^2_{\Sigma,\mathcal{V}} \to [0,1]$ is defined inductively on $\mathcal{T}_{\Sigma,\mathcal{V}}$ as follows:*

1. $\forall X \in \mathcal{V},\ X \sim^{\mathcal{T}}_1 X$;
2. $\forall X \in \mathcal{V}, \forall t \in \mathcal{T}$, *such that* $X \neq t$, $X \sim^{\mathcal{T}}_0 t$ *and* $t \sim^{\mathcal{T}}_0 X$;
3. *if* $f \in \Sigma_n$ *and* $g \in \Sigma_n$ *with* $f \sim_\alpha g$, *and if* $s_i \in \mathcal{T}$ *and* $t_i \in \mathcal{T}$ *are such that* $s_i \sim^{\mathcal{T}}_{\alpha_i} t_i$ *for* $i = 1, \ldots, n$, *then:*

$$f(s_1,\ldots,s_n) \sim^{\mathcal{T}}_{\alpha \wedge \bigwedge_{i=1}^n \alpha_i} g(t_1,\ldots,t_n). \tag{11}$$

Theorem 2. *The relation $\sim^{\mathcal{T}}$ defined by Definition 5 is a similarity relation on the set of $\mathcal{FOT}$s $\mathcal{T}_{\Sigma,\mathcal{V}}$.*

Proof. See proof of more general Theorem 3 below, as this is a particular case of that theorem where every similar pairs of functors have same arity and every argument position mapping is the identity. □

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Since from the above definition of similarity $\sim^{\mathcal{T}}$ extends homomorphically a similarity $\sim$ on the functors to all $\mathcal{FOT}$s in $\mathcal{T}$, we shall also assimilate $\sim^{\mathcal{T}}$ to $\sim$. This allows to define formally fuzzy subsumption among $\mathcal{FOT}$s as the fuzzy relation $\preceq$ on $\mathcal{T}$ that can be verified to be a preorder (modulo variable renaming) as a corollary of Theorem 2.

Definition 6 (Fuzzy $\mathcal{FOT}$ Subsumption). *For all terms $t_1 \in \mathcal{T}$ and $t_2 \in \mathcal{T}$, t_1 is said to be subsumed by t_2 for some α in $[0, 1]$ (and this is written $t_1 \preceq_\alpha t_2$) if and only if there exists a substitution $\sigma \in \textbf{Subst}_{\mathcal{T}}$ such that $t_1 \sim_\alpha t_2\sigma$.*

Note that, for the identity similarity on the signature and $\alpha = 1$, this reduces to the classical definition of term subsumption, as expected.

In Definition 6, the more specific term t_1 is then called a fuzzy instance of term t_2 realized with substitution σ at approximation degree α. It comes also that the "more general" relation on $\mathcal{FOT}$ substitutions extends to a "fuzzy more general" fuzzy relation on substitutions, which can also readily be verified to be a fuzzy preorder on $\textbf{Subst}_{\mathcal{T}}$ as a corollary of Theorem 2. It is formally equivalent to the relation defined in [12].[17]

Definition 7 (Fuzzy "More General" Ordering on $\mathcal{FOT}$ Substitutions). *If σ_1 and σ_2 are two substitutions in $\textbf{Subst}_{\mathcal{T}}$ and α in $[0, 1]$, we say that σ_1 is less general than σ_2 at approximation degree α (and this is written $\sigma_1 \preceq_\alpha \sigma_2$), if and only if for any term $t \in \mathcal{T}$, it is true that $t\sigma_1 \preceq_\alpha t\sigma_2$ as terms.*

Also as expected, note that for the identity similarity on the signature and $\alpha = 1$, this reduces to the classical "more general than" ordering on substitutions.

The following fuzzy relation defined on $\textbf{Subst}_{\mathcal{T}}$ can also be verified to be a similarity as a corollary of Theorem 2.[18]

Definition 8 ($\mathcal{FOT}$ Substitution Similarity). *Given an approximation degree α in $[0, 1]$, two substitutions σ and θ in $\textbf{Subst}_{\mathcal{T}}$ are said to be α-similar (written $\sigma \sim_\alpha \theta$) iff $t\sigma \sim_\alpha t\theta$ for all $\mathcal{FOT}$ t in $\mathcal{T}$.*

Therefore, referring to Definition 6 of fuzzy $\mathcal{FOT}$ subsumption, it comes as a fact that:

Lemma 3. *For any two substitutions σ and θ in $\textbf{Subst}_{\mathcal{T}}$ and approximation degree α in $[0, 1]$, $\sigma \preceq_\alpha \theta$ iff $\sigma \sim_\alpha \theta\delta$ for some substitution δ.*

Proof. Stating that $\sigma \preceq_\alpha \theta$, by Definition 7, is equivalent to stating that $t\sigma \preceq_\alpha t\theta$, for any $t \in \mathcal{T}$. By Definition 6, this is equivalent to stating that for all term t, $t\sigma \sim_\alpha t\theta\delta$, for some substitution δ; namely, again by Definition 7, that $\sigma \sim_\alpha \theta\delta$. $\square$

The following two facts regarding the fuzzy term subsumption relation on terms and the fuzzy "more general" relation on substitutions will be useful later in proof arguments.

Lemma 4. *For any two approximation degrees α and β in $[0, 1]$, for any terms t_1, t_2, and t_3 in $\mathcal{T}$, if $t_1 \preceq_\alpha t_2$ and $t_2 \preceq_\beta t_3$, then $t_1 \preceq_{\alpha\wedge\beta} t_3$.*

Proof. Let $t_1 \preceq_\alpha t_2$ and $t_2 \preceq_\beta t_3$; this is, by definition, equivalent to $t_1 \sim_\alpha t_2\sigma$, for some $\sigma \in \textbf{Subst}_{\mathcal{T}}$, and $t_2 \sim_\beta t_3\theta$, for some $\theta \in \textbf{Subst}_{\mathcal{T}}$. However, for any set S, any pair $\langle x, y\rangle$ in S^2, and any similarity $\sim: S^2 \to [0, 1]$, if $x \sim_\alpha y$ for some α in$[0, 1]$, then $x \sim_\beta y$ for all $\beta \in [0, \alpha]$.[19] This, the fact that $\alpha \wedge \beta \leq \alpha$ and $\alpha \wedge \beta \leq \beta$, together with our assumption, entail then that $t_1 \sim_{\alpha\wedge\beta} t_2\sigma$ and $t_2 \sim_{\alpha\wedge\beta} t_3\sigma$; which, by transitivity of $\sim_{\alpha\wedge\beta}$, implies that $t_1 \sim_{\alpha\wedge\beta} t_3\sigma$; that is, $t_1 \preceq_{\alpha\wedge\beta} t_3$. $\square$

[17] Ref. [12], Page 410, Definition 6.2.
[18] A equivalent definition is given in [40].
[19] See Appendix C.2.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

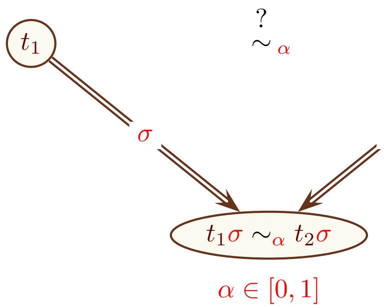


Fig. 6. Fuzzy unification as a constraint.

Weak Term Decomposition

$$\frac{(E \cup \{\, f(s_1,\ldots,s_n) \doteq g(t_1,\ldots,t_n)\,\})_\alpha}{(E \cup \{\, s_1 \doteq t_1, \ldots, s_n \doteq t_n \,\})_{\alpha\wedge\beta}} \begin{bmatrix} f \sim_\beta g \\ n \geq 0 \end{bmatrix}$$

Variable Erasure

$$\frac{(E \cup \{\, X \doteq X \,\})_\alpha}{E_\alpha}$$

Variable Elimination

$$\frac{(E \cup \{\, X \doteq t \,\})_\alpha}{(E[X \leftarrow t] \cup \{\, X \doteq t \,\})_\alpha} \begin{bmatrix} X \notin \mathbf{var}(t) \\ X \text{ occurs in } E \end{bmatrix}$$

Equation Orientation

$$\frac{(E \cup \{\, t \doteq X \,\})_\alpha}{(E \cup \{\, X \doteq t \,\})_\alpha} [t \notin \mathcal{V}]$$

Fig. 7. Normalization rules corresponding to Maria Sessa's "weak unification".

Corollary 3. *For any two approximation degrees α and β in* $[0, 1]$, *for any substitutions σ_1, σ_2, and σ_3 in* $\mathbf{SUBST}_{\mathcal{T}}$, *if $\sigma_1 \preceq_\alpha \sigma_2$ and $\sigma_2 \preceq_\beta \sigma_3$, then $\sigma_1 \preceq_{\alpha\wedge\beta} \sigma_3$.*

Proof. Let $\sigma_1 \preceq_\alpha \sigma_2$ and $\sigma_2 \preceq_\beta \sigma_3$; this is, by definition, equivalent to stating that for any term $t \in \mathcal{T}$, $t\sigma_1 \preceq_\alpha t\sigma_2$ and $t\sigma_2 \preceq_\beta t\sigma_3$. By Lemma 4, it follows that for any term $t \in \mathcal{T}$, $t\sigma_1 \preceq_{\alpha\wedge\beta} t\sigma_3$; that is, $\sigma_1 \preceq_{\alpha\wedge\beta} \sigma_3$. □

Using the definition of similarity between terms in $\mathcal{T}$ extending one on functors of equal arity, Sessa proposes to extend the $\mathcal{FOT}$ unification problem to the following fuzzy unification problem: given two $\mathcal{FOT}$s t_1 and t_2 in $\mathcal{T}$, find the most general substitution $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$ and maximum approximation degree α in $[0, 1]$ such that $t_1\sigma \sim_\alpha t_2\sigma$. Fig. 6 expresses fuzzy unification as a commutative diagram constraint.

In Fig. 7, we provide a set of declarative rewrite rules for fuzzy unification equivalent to Sessa's case-based "weak unification algorithm" [12]. To simplify the presentation of these rules while remaining faithful to Sessa's weak unification algorithm, it is assumed for now that functor symbols f/m and g/n of different arities $m \neq n$ are never similar. This follows Sessa's assumption for weak unification, which fails on term structures of different arities. (See Case (2) of the weak unification algorithm given in [12], Page 413.) Later, we will relax this and allow functors of different arities to be similar.

The rules of Fig. 7 transform E_α, a finite conjunctive set E of equations among $\mathcal{FOT}$s along with an associated approximation degree α in $[0, 1]$, into $E'_{\alpha'}$, another set of equations E' at approximation degree α' in $[0, \alpha]$. Given to solve a fuzzy unification equation $s \doteq t$ between two $\mathcal{FOT}$s s and t, we start by forming the set $\{\, s \doteq t \,\}_1$ (*i.e.*, a singleton equation set at approximation degree 1), then transform it using any applicable rules in Fig. 7 until either the approximation degree of the transformed set of equations is 0 (in which case there is no solution to the original equation, not even a fuzzy one), or the final resulting set E_α is a solution at approximation degree α in the form of a variable substitution $\sigma \overset{\text{def}}{=} \{\, t/X \mid X \doteq t \in E \,\}$ such that $s\sigma \sim_\alpha t\sigma$.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

In [12],[20] a transformation rule of a set of equation at approximation degree is considered to be correct when all the solutions of the posterior set are also solutions of the anterior set but with a possibly lesser similarity degree, which is also our Definition 10.[21]

Example 6. [$\mathcal{FOT}$ *fuzzy unification*] Taking the functor signature of Example 5, let us consider the fuzzy equation set:

$$\left\{ h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)) \doteq h(X_2, X_2, g(c, d)) \right\}_1 \tag{12}$$

and let us apply the rules of Fig. 7:

- Rule **Weak Term Decomposition** with $\alpha = 1$ and $\beta = 1$:

$$\left\{ f(a, X_1) \doteq X_2, g(X_1, b) \doteq X_2, f(Y_1, Y_1) \doteq g(c, d) \right\}_1;$$

- Rule **Equation Orientation** to $f(a, X_1) \doteq X_2$ with $\alpha = 1$:

$$\left\{ X_2 \doteq f(a, X_1), g(X_1, b) \doteq X_2, f(Y_1, Y_1) \doteq g(c, d) \right\}_1;$$

- Rule **Variable Elimination** to $X_2 \doteq f(a, X_1)$ with $\alpha = 1$:

$$\left\{ X_2 \doteq f(a, X_1), g(X_1, b) \doteq f(a, X_1), f(Y_1, Y_1) \doteq g(c, d) \right\}_1;$$

- Rule **Weak Term Decomposition** to $g(X_1, b) \doteq f(a, X_1)$ with $\alpha = 1$ and $\beta = .9$:

$$\left\{ X_2 \doteq f(a, X_1), X_1 \doteq a, b \doteq X_1, f(Y_1, Y_1) \doteq g(c, d) \right\}_{.9};$$

- Rule **Variable Elimination** to $X_1 \doteq a$ with $\alpha = .9$:

$$\left\{ X_2 \doteq f(a, a), X_1 \doteq a, b \doteq a, f(Y_1, Y_1) \doteq g(c, d) \right\}_{.9};$$

- Rule **Weak Term Decomposition** to $b \doteq a$ with $\alpha = .9$ and $\beta = .7$:

$$\left\{ X_2 \doteq f(a, a), X_1 \doteq a, f(Y_1, Y_1) \doteq g(c, d) \right\}_{.7};$$

- Rule **Weak Term Decomposition** to $f(Y_1, Y_1) \doteq g(c, d)$ with $\alpha = .7$ and $\beta = .9$:

$$\left\{ X_2 \doteq f(a, a), X_1 \doteq a, Y_1 \doteq c, Y_1 \doteq d \right\}_{.7};$$

- Rule **Variable Elimination** to $Y_1 \doteq c$ with $\alpha = .7$:

$$\left\{ X_2 \doteq f(a, a), X_1 \doteq a, Y_1 \doteq c, c \doteq d \right\}_{.7};$$

- Rule **Weak Term Decomposition** to $c \doteq d$ with $\alpha = .7$ and $\beta = .6$:

$$\left\{ X_2 \doteq f(a, a), X_1 \doteq a, Y_1 \doteq c \right\}_{.6}.$$

This last equation set is in normal form with similarity degree .6 and defines the substitution σ given by $\sigma = \{a/X_1, c/Y_1, f(a, a)/X_2\}$, so that:

$$t_1\sigma = h(f(a, a), g(a, b), f(c, c)) \sim_{.6} h(f(a, a), f(a, a), g(c, d)) = t_2\sigma. \tag{13}$$

---

[20] Ref. [12], Page 410.

[21] Note that in [12], no explicit proof for of formal correctness of "weak unification algorithm" is given: it is just mentioned that "it can be proven following the same line of the proof" for crisp unification in classible Logic Programming in [41].

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

4.1.2. A generic fuzzy unification scheme

From our perspective, a fuzzy unification operation ought to be able to fuzzify *full* $\mathcal{FOT}$ unification: whether (1) functor symbol mistmatch, and/or (2) arity mismatch, and/or (3) in which order subterms correspond. Sessa's fuzzification of unification as weak unification misses on the last two items. This is unfortunate as this can turn out to be quite useful. In real life, there is indeed no such guarantee that argument positions of different functors match similar information in data and knowledge bases, hence the need for alignment [33].

Still, it has several qualities:

- *It is simple*—specified as a straightforward extension of crisp unification: only one rule (Rule "**Fuzzy Term Decomposition**") may alter the fuzziness of an equation set by tolerating similar functors.
- *It is conservative*—neither $\mathcal{FOT}$s nor $\mathcal{FOT}$ substitutions *per se* need be fuzzified; so conventional crisp representations and operations can be used; if restricted to only 0 or 1 similarity degrees, it is equivalent to crisp $\mathcal{FOT}$ unification.

We now give an extension of Sessa's weak unification which can tolerate such similarity among functors of different arities. We are given a similarity relation $\approx : \Sigma^2 \rightarrow [0,1]$ on a ranked signature $\Sigma \stackrel{\text{def}}{=} \Sigma_{n\geq 0}$. Unlike M. Sessa's equal-arity condition, we now allow similar symbols with distinct arities, or equal arities but different argument orders.

Example 7. [*Similar functors with different arities*] Consider $\texttt{person}/3$, a functor of arity 3, and $\texttt{individual}/4$, a functor of arity 4 with:

- $\texttt{person}/3 \approx_{.9} \texttt{individual}/4$; and,
- one-to-one position mapping $p : \{1,2,3\} \rightarrow \{1,2,3,4\}$:

  from $\texttt{person}/3$ to $\texttt{individual}/4$ with $p : \{1 \rightarrow 1, 2 \rightarrow 3, 3 \rightarrow 4\}$

  so that:

  $$\texttt{person}(\mathit{Name}, \mathit{SSN}, \mathit{Address}) \approx^{p}_{.9} \texttt{individual}(\mathit{Name}, \mathit{DoB}, \mathit{SSN}, \mathit{Address})$$

where we write $f \approx^{p}_{\alpha} g$ to denote a pair in the similarity relation $\approx$ consisting of a functor f and a functor g, with similarity degree α and f-to-g argument-position mapping p; in our example, $\texttt{person} \approx^{\{1\rightarrow 1, 2\rightarrow 3, 3\rightarrow 4\}}_{.9} \texttt{individual}$.

With this kind of specification, we can tolerate not only fuzzy mismatching of terms with distinct functors person and individual up to a realigning correspondence of argument positions from person to *individual* specified as p, all with a similarity degree of .9.

We formalize this by requiring that the fuzzy relation $\approx$ on Σ be such that:

- for each pair of functors $\langle f, g\rangle \in \Sigma^2$, such that $f \in \Sigma_m$ and $g \in \Sigma_n$, with $0 \leq m \leq n$, and $f \approx g$, there exists an injective (*i.e.*, one-to-one) mapping $\mu_{fg} : \{1,\ldots,m\} \rightarrow \{1,\ldots,n\}$ associating each of the m argument positions of f with a unique position among the n arguments of g, which we shall express as $f \approx^{\mu_{fg}} g$;
- argument alignment mapping between similar functors must be *consistent*; *i.e.*,
  - for any functor f/n:

    $$\mu_{ff} = \mathbb{1}_{\{1,\ldots,n\}}; \tag{14}$$

  - for any two equal-arity functors f/n and g/n:

    $$\mu_{fg} \circ \mu_{gf} = \mathbb{1}_{\{1,\ldots,n\}}; \tag{15}$$

  - for any three functors f/m, g/n, h/ℓ such that $0 \leq m \leq n \leq \ell$:

    $$\mu_{fh} = \mu_{gh} \circ \mu_{fg}. \tag{16}$$

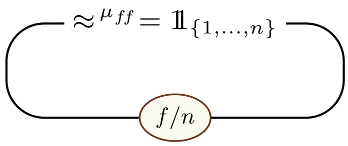


Fig. 8. Identity consistency condition for $\mathcal{FOT}$ argument alignment mapping.

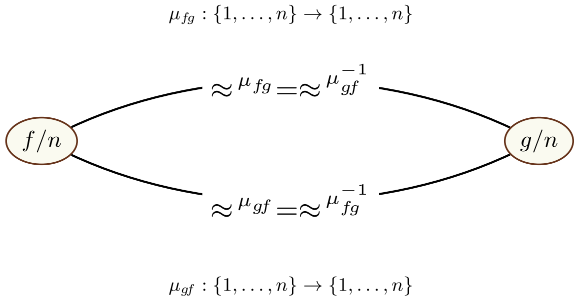


Fig. 9. Invertibility consistency condition for equal-arity $\mathcal{FOT}$ argument alignment mappings.

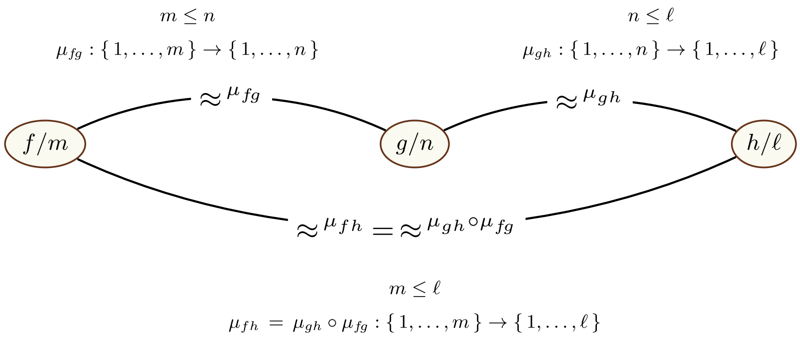


Fig. 10. Compositional consistency condition for non-aligned $\mathcal{FOT}$ argument alignment mappings.

Fig. 8 illustrates Condition (14), Fig. 9 illustrates Condition (15), and Fig. 10 illustrates Condition (16).

Note that Condition (16) applies when $0 \leq m \leq n \leq \ell$; so the one-to-one argument-position mappings always go from a smaller set to a larger set. There is no loss of generality with this assumption as this will be taken into account in the definition of non-aligned $\mathcal{FOT}$ similarity,[22] and in the normalization rules.[23] This amounts to systematically taking a $\mathcal{FOT}$ with functor of least arity as similarity class representative. Finally, note also that such a class representative is not unique because for similar functors of equal arity, it can be either terms due to Condition (15). Indeed, then the set of positions are equal and there are two injections from this set to itself in each direction which are mutually inverse bijections; *i.e.*, inverse permutations in the order of arguments realigning one's with the other's in either direction. The similarity degrees in both directions are always equal due to symmetry of similarity.

Fuzzy unification with similar functors and arity mismatch

As in the case of similarity restricted to functors of equal arities only, the similarity with argument position alignment mapping on functors can be extended homomorphically to a similarity on $\mathcal{FOT}$s. Let $\approx$ be a similarity on functors of any arity in a signature Σ. To lighten notation, rather than writing systematically $f \approx^{\mu_{fg}} g$ for two functors f and g such that $\mathbf{arity}(f) \leq \mathbf{arity}(g)$, we shall sometimes simply write $f \approx^{p}_{\alpha} g$, with p standing for the injective argument realignment mapping μ_{fg}.

[22] *Cf.*, Definition 9 below.
[23] *Cf.*, Fig. 11 below, Rule **Fuzzy Equation Orientation**.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Definition 9. *The fuzzy relation* $\approx^{\mathcal{T}}$ *on* $\mathcal{T}_{\Sigma,\mathcal{V}}$ *is defined inductively as:*

1. $\forall X \in \mathcal{V},\ \ X \approx_1^{\mathcal{T}} X$;
2. $\forall X \in \mathcal{V}, \forall t \in \mathcal{T}$, *such that* $X \neq t$, $\ X \approx_0^{\mathcal{T}} t$ *and* $t \approx_0^{\mathcal{T}} X$;
3. *if* $s = f(s_1, \ldots, s_m)$ *and* $t = g(t_1, \ldots, t_n)$ *with* $n < m$, *then* $s \approx^{\mathcal{T}} t = t \approx^{\mathcal{T}} s$;
4. *if* $f \in \Sigma_m$ *and* $g \in \Sigma_n$ *with* $m \leq n$ *and* $f \approx_\alpha^p g$, *and if* $s_i \in \mathcal{T}$, $i = 1, \ldots, m$, *and* $t_j \in \mathcal{T}$, $j = 1, \ldots, n$, *are such that* $s_i \approx_{\alpha_i}^{\mathcal{T}} t_{p(i)}$ *for all* $i \in \{1, \ldots, m\}$, *then:*

$$f(s_1, \ldots, s_m) \approx^{\mathcal{T}}_{\alpha \wedge \bigwedge_{i=1}^m \alpha_i} g(t_1, \ldots, t_n). \tag{17}$$

Theorem 3 (Non-Aligned $\mathcal{FOT}$ Similarity). *The fuzzy relation* $\approx^{\mathcal{T}}$ *on the set* $\mathcal{T}$ *of* $\mathcal{FOT}$*s specified in Definition 9 is a similarity.*

Proof. We must establish that $\approx^{\mathcal{T}}$ is reflexive, symmetric, and transitive.

Reflexivity: we must show that $t \approx_1^{\mathcal{T}} t$, for all $t \in \mathcal{T}$. We proceed by induction on the depth of the term. *Base case:* either $t = X \in \mathcal{V}$, in which case, by the first condition of Definition 9, $X \approx_1^{\mathcal{T}} X$; or, $t = c \in \Sigma_0$, in which case the fourth condition of Definition 9 and the fact that $c \approx_1 c$ implies that $c \approx_1^{\mathcal{T}} c$, for all $c \in \Sigma_0$. *Inductive case:* let us assume that $\approx^{\mathcal{T}}$ is reflexive for all terms of depth less than or equal to d, and consider the term $t = f(t_1, \ldots, t_n)$ of depth $d + 1$; then, the fourth condition of Definition 9 implies also that $t \approx^{\mathcal{T}} t$ since, by Condition (14) and the fact that $\approx$ is a similarity, $f \approx_1^{\mathbb{1}_{\{1,\ldots,n\}}} f$ for all $f \in \Sigma_n$, for any arity $n > 0$.

Symmetry: we must show that $s \approx^{\mathcal{T}} t = t \approx^{\mathcal{T}} s$ for all s and t in $\mathcal{T}$. When either of the terms is a variable, this is so by the two first cases of Definition 9. When $s = f(s_1, \ldots, s_m)$ and $t = g(t_1, \ldots, t_n)$, it is always the case that $\approx^{\mathcal{T}}$ is symmetric on such pairs since the third condition of Definition 9, states precisely that in this case $\approx^{\mathcal{T}}$ is symmetric.

Transitivity: we must show that $(s \approx^{\mathcal{T}} t \wedge t \approx^{\mathcal{T}} u) \leq s \approx^{\mathcal{T}} u$ for all terms s, t, u. There are eight possibilities:

(1)	$s \in \mathcal{V}$ and $t \in \mathcal{V}$ and $u \in \mathcal{V}$;	(5)	$s \notin \mathcal{V}$ and $t \in \mathcal{V}$ and $u \in \mathcal{V}$;
(2)	$s \in \mathcal{V}$ and $t \in \mathcal{V}$ and $u \notin \mathcal{V}$;	(6)	$s \notin \mathcal{V}$ and $t \in \mathcal{V}$ and $u \notin \mathcal{V}$;
(3)	$s \in \mathcal{V}$ and $t \notin \mathcal{V}$ and $u \in \mathcal{V}$;	(7)	$s \notin \mathcal{V}$ and $t \notin \mathcal{V}$ and $u \in \mathcal{V}$;
(4)	$s \in \mathcal{V}$ and $t \notin \mathcal{V}$ and $u \notin \mathcal{V}$;	(8)	$s \notin \mathcal{V}$ and $t \notin \mathcal{V}$ and $u \notin \mathcal{V}$.

- Case (1): $s \in \mathcal{V}, t \in \mathcal{V}, u \in \mathcal{V}$. In this case, there are five possibilities. Using different variable names to denote different variables, the corresponding similarity degrees for $s \approx^{\mathcal{T}} t$, $t \approx^{\mathcal{T}} u$, and $s \approx^{\mathcal{T}} u$, for each possibility do indeed verify the inequality. Namely:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
X	Y	Z	0	$\wedge$	0	$\leq$	0
X	Y	Y	0	$\wedge$	1	$\leq$	1
X	Y	X	0	$\wedge$	0	$\leq$	1
X	X	Y	1	$\wedge$	0	$\leq$	0
X	X	X	1	$\wedge$	1	$\leq$	1

- Case (2): $s \in \mathcal{V}, t \in \mathcal{V}, u \notin \mathcal{V}$. There are two possibilities, each verifying the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
X	X	u	1	$\wedge$	0	$\leq$	0
X	Y	u	0	$\wedge$	0	$\leq$	0

- Case (3): $s \in \mathcal{V}, t \notin \mathcal{V}, u \in \mathcal{V}$. There are two possibilities, and each verifies the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
X	t	X	0	$\wedge$	0	$\leq$	1
X	t	Y	0	$\wedge$	0	$\leq$	0

– Case (4): $s \in \mathcal{V}, t \notin \mathcal{V}, u \notin \mathcal{V}$. There is only one possibility, for any $\alpha \in [0, 1]$, which verifies the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
X	t	u	0	$\wedge$	α	$\leq$	0

– Case (5): $s \notin \mathcal{V}, t \in \mathcal{V}, u \in \mathcal{V}$. There are two possibilities, and each verifies the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
s	X	X	0	$\wedge$	1	$\leq$	0
s	X	Y	0	$\wedge$	0	$\leq$	0

– Case (6): $s \notin \mathcal{V}, t \in \mathcal{V}, u \notin \mathcal{V}$. There is only one possibility, for any $\alpha \in [0, 1]$, which verifies the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
s	X	u	0	$\wedge$	0	$\leq$	α

– Case (7): $s \notin \mathcal{V}, t \notin \mathcal{V}, u \in \mathcal{V}$. There is only one possibility, for any $\alpha \in [0, 1]$, which verifies the inequality:

s	t	u	$s \approx^{\mathcal{T}} t$	$\wedge$	$t \approx^{\mathcal{T}} u$	$\leq$	$s \approx^{\mathcal{T}} u$
s	t	X	α	$\wedge$	0	$\leq$	0

– Case (8): $s \notin \mathcal{V}, t \notin \mathcal{V}, u \notin \mathcal{V}$. In this case, we must have $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$, and $u = h(u_1, \ldots, u_\ell)$. We detail this case below.

We must then show that:

$$f(s_1, \ldots, s_m) \approx^{\mathcal{T}} g(t_1, \ldots, t_n) \;\wedge\; g(t_1, \ldots, t_n) \approx^{\mathcal{T}} h(u_1, \ldots, u_\ell) \;\leq\; f(s_1, \ldots, s_m) \approx^{\mathcal{T}} h(u_1, \ldots, u_\ell).$$

By symmetry of $\approx^{\mathcal{T}}$, all cases are equivalent to when $0 \leq m \leq n \leq \ell$, so we assume that this is so, with $f \approx_{\alpha}^{\mu_{fg}} g$ and $g \approx_{\beta}^{\mu_{gh}} h$. By the fourth condition of Definition 9, the above inequality is the same as the following one:

$$f \approx g \;\wedge\; \bigwedge_{i=1}^{m} s_i \approx^{\mathcal{T}} t_{\mu_{fg}(i)} \;\wedge\; g \approx h \;\wedge\; \bigwedge_{j=1}^{n} t_j \approx^{\mathcal{T}} u_{\mu_{gh}(j)} \;\leq\; f \approx h \;\wedge\; \bigwedge_{i=1}^{m} s_i \approx^{\mathcal{T}} u_{\mu_{fh}(i)}.$$

Using commutativity of $\wedge$, let us rearrange the different factors of the conjunction in the lefthand-side of this inequality as:

$$f \approx g \;\wedge\; g \approx h \;\wedge\; \left(\bigwedge_{i=1}^{m} s_i \approx^{\mathcal{T}} t_{\mu_{fg}(i)} \;\wedge\; t_{\mu_{fg}(i)} \approx^{\mathcal{T}} u_{\mu_{gh}(\mu_{fg}(i))}\right) \;\wedge\; \Delta \;\leq\; f \approx h \;\wedge\; \bigwedge_{i=1}^{m} s_i \approx^{\mathcal{T}} u_{\mu_{fh}(i)}$$

where Δ stands for the remaining conjunction $\bigwedge_{j \in \{1,\ldots,n\}}^{j \notin \mathbf{ran}(\mu_{fg})} t_j \approx^{\mathcal{T}} u_{\mu_{gh}(j)}$. Let us now proceed by induction on the depth of the terms to verify this inequality. For terms of depth 0, it is verified since it reduces to the transitivity inequality of $\approx$ on Σ_0. Let us assume that it holds for terms of depth less than d, and that at least one of the terms s, t, or u, is of depth d. By transitivity of $\approx$ on Σ, we have $(f \approx g) \wedge (g \approx h) \leq f \approx h$. Also, by the inductive hypothesis, the transitivity inequality for $\approx^{\mathcal{T}}$ holds for all similar subterms of depth less than or equal to d. Therefore, this assumption entails that for all $i \in \{1, \ldots, m\}$:

$$s_i \approx^{\mathcal{T}} t_{\mu_{fg}(i)} \wedge t_{\mu_{fg}(i)} \approx^{\mathcal{T}} u_{\mu_{gh}(\mu_{fg}(i))} \leq s_i \approx^{\mathcal{T}} u_{\mu_{gh}(\mu_{fg}(i))};$$

thus, since, by Condition (16), it is required that the mappings be consistent and verify $\mu_{fh} = \mu_{gh} \circ \mu_{fg}$, and by isotonicity of $\wedge$ w.r.t. $\leq$, this is equivalent to:

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

$$\bigwedge_{i=1}^{m}(s_i \approx^{\mathcal{T}} t_{\mu_{fg}(i)}) \wedge (t_{\mu_{fg}(i)} \approx^{\mathcal{T}} u_{\mu_{fh}(i)}) \leq \bigwedge_{i=1}^{m} s_i \approx^{\mathcal{T}} u_{\mu_{fh}(i)}.$$

In summary, the inequality we seek to establish is of the form $A \wedge B \wedge \Delta \leq A' \wedge B'$, and we have shown that $A \leq A'$ and $B \leq B'$. From this, the inequality follows by isotonicity of $\wedge$ w.r.t. $\leq$. □

Since we have just formally defined a new notion of similarity $\approx^{\mathcal{T}}$ on $\mathcal{T}$ extending Sessa's similarity $\sim^{\mathcal{T}}$ to non-aligned functors, all the properties we covered for $\sim^{\mathcal{T}}$ carry over to corresponding extensions for terms with non-aligned functors. Namely, Definitions 6–8 and Lemmas 3–4, as well as Corollary 3, where the term similarity $\sim^{\mathcal{T}}$ is replaced with any similarity on $\mathcal{T}$ such as $\approx^{\mathcal{T}}$ (or $\approxeq^{\mathcal{T}}$ that we shall define later and prove also to be a similarity on $\mathcal{T}$ extending $\approx^{\mathcal{T}}$). Indeed, it is easy to see that all these notions are valid algebraically when parameterized with any relation on $\mathcal{FOT}$ proven to be a similarity on $\mathcal{T}$.

Weak unification with fuzzy functor/arity mismatch

Starting with the Herbrand-Martelli-Montanari ruleset of Fig. 3, fuzziness is introduced in Sessa's weak unification by relaxing "**Term Decomposition**" to make it also tolerate possible arity or argument-order mistmatch in two structures being unified. It is the only rule that does not preserve the equation set's similarity degree. In other words, the given functor similarity relation $\approx$ on Σ is adjoined a position mapping from argument positions of a functor f to those of a functor g when $f \approx_\alpha g$ with $f \neq g$, for some α in (0, 1]. This is then taken into account in tolerating a fuzzy mismatch between two term structures $s = f(s_1, \ldots, s_m)$ and $t = g(t_1, \ldots, t_n)$. This may involve a mismatch between the terms' functor symbols (f and g), their arities (m and n), subterm ordering, or a combination. We first reorient all such equations by flipping sides so that the left-hand side is the one with lesser or equal arity. In this manner, assuming $f \approx^p_\beta g$ and $0 \leq \alpha, \beta \leq 1$, an equation set of the form: $\{\ \ldots\ ,\ f(s_1, \ldots, s_m) \doteq g(t_1, \ldots, t_n),\ \ \ldots\ \}_\alpha$ for $0 \leq m \leq n$ acquires its new similarity degree $\alpha \wedge \beta$ due to functor and arity mismatch when equated. Thus, a fully fuzzified term-decomposition rule should proceed by replacing a structure equation by the conjunction of equations between their respective subterms at corresponding indices given by the one-to-one argument mapping $p : \{1, \ldots, m\} \to \{1, \ldots, n\}$, but (possibly) decreasing the original equation set similarity degree by conjoining it with that of the decomposed terms' functor pair; that is, $\{\ldots,\ s_1 \doteq t_{p(1)},\ \ldots,\ s_m \doteq t_{p(m)},\ \ldots\}_{\alpha\wedge\beta}$. Note that all the subterms in the right-hand side term that are arguments at indices which are not p-images are ignored as they have no counterparts in the left-hand side. These terms are simply dropped as part of the approximation. This generic rule is shown in Fig. 11 along with another rule needed to make it fully effective: a rule reorienting a term equation into one with a lesser-arity term on the left.

Definition 10 *(Fuzzy Unification Rule Correctness). A fuzzy unification rule that transforms E_α, a set of equations E at a prior approximation degree α, into E'_β, a set of equations E' at a posterior approximation degree β, is said to be correct iff β is the largest degree such that $\beta \leq \alpha$ and all the substitutions of E' are also substitutions of E at approximation degree β.*

Note that this notion of correctness, contrary to that of crisp unification, does not require that all solutions of the posterior sets of equations be the same as those of the prior set. It only states that this be so as a possibly lesser posterior approximation degree.

Theorem 4. *The fuzzy unification rules of Fig. 7 where Rule "***Weak Term Decomposition***" is replaced by the rules of Fig. 11 are correct.*

Proof. Rules **Variable Elimination**, **Variable Erasure**, and **Equation Orientation** are those, unchanged, of Maria Sessa's weak unification. Their correctness follows from those of the corresponding Herbrand-Martelli-Montanari rules since all three rules keep their similarity degree α unchanged under the same side conditions as their crisp versions. As for Rule **Fuzzy Equation Orientation**, it is also correct as it simply uses the symmetry of equality or similarity denoted by the $\doteq$ relation and it leaves the similarity degree unchanged.

The correctness of Rule **Fuzzy Non-Aligned-Argument Term Decomposition** follows from the fact that it tolerates equations between two distinct but similar functors, f on the left and g on the right, by "paying a toll" as

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Fuzzy Non-Aligned-Argument Term Decomposition

$$\left[0 \leq m \leq n;\ f \approx_{\beta}^{p} g\right]$$

$$\frac{\left(E \cup \{ f(s_1, \ldots, s_m) \doteq g(t_1, \ldots, t_n) \}\right)_{\alpha}}{\left(E \cup \{ s_1 \doteq t_{p(1)}, \ldots, s_m \doteq t_{p(m)} \}\right)_{\alpha \wedge \beta}}$$

Fuzzy Equation Orientation

$$[0 \leq m < n]$$

$$\frac{\left(E \cup \{ g(t_1, \ldots, t_n) \doteq f(s_1, \ldots, s_m) \}\right)_{\alpha}}{\left(E \cup \{ f(s_1, \ldots, s_m) \doteq g(t_1, \ldots, t_n) \}\right)_{\alpha}}$$

Fig. 11. Fuzzy $\mathcal{FOT}$ unification's non-aligned decomposition and orientation rules.

the most general way this can be true is by reducing the prior equation set's similarity degree α to $\alpha \wedge \beta$. It must do so whenever a prior equation set contains an equation between two terms whose respective head functors f and g are β-similar with f having at most as many arguments as g. It collects m corresponding subterm equations from the two terms's subterms using the specific one-to-one argument mapping p that associates with each position i among f's m a unique specific position $p(i)$ among g's n, $n \geq m$. Orienting all functorial term equations to have the lesser number of arguments on the left guarantees completeness over all such syntactic patterns. By structural induction, assuming that all $s_i \approx_{\alpha\wedge\beta} t_{p(i)}$ for all $i \in \{1, \ldots, m\}$, then whenever $f \approx_{\beta}^{p} g$, we must also have $f(s_1, \ldots, s_m) \approx_{\alpha\wedge\beta} g(t_1, \ldots, t_n)$ (by definition, since $\alpha \wedge \beta \leq \alpha$) for whatever arguments of g at indices missed by p and these two terms are in the same similarity class at approximation degree $\alpha \wedge \beta$ for arbitrary arguments in these positions (by definition of $f \approx_{\beta}^{p} g$ and since $\alpha \wedge \beta \leq \beta$). The rest of the equations in E that were true at approximation degree α must now be considered true only up to approximation degree $\alpha \wedge \beta$ in order to account for f and g being functors of possibly fuzzier similarity β. Hence, all solutions of the new set of equations are also solutions of the previous one, although only at the possibly lesser approximation degree $\alpha \wedge \beta$. This approximation degree is also the greatest such degree by virtue of the $\wedge$ operation yielding the infimum of its operands.

Finally, when $m = n$ this rule is correct in either direction since a consistent similarity on a signature requires by definition that equal-arity functors f and g have arguments in bijection (inverse permutations of the set $\{1, \ldots, n\}$): $f \approx_{\alpha}^{p} g$ and $g \approx_{\alpha}^{p^{-1}} f$. In this case, the set of solutions of the new equation set is also a solution of the previous one, with equal similarity degree.

As for termination, it follows (like that of the Herbrand-Martelli-Montanari rules) from (1) the finite width and depth of $\mathcal{FOT}$s, and (2) there being no rule that is indefinitely applicable. Regarding (1), term decomposition always replaces a term equation with finitely many shallower term equations, which is a well-known well-founded process guaranteed to terminate (multiset ordering [42]). Regarding (2), Rule **Fuzzy Equation Orientation** may not be reapplied to the same functors thanks to the side condition $m < n$.

In other words, applying this modified ruleset to $E_1 \stackrel{\text{def}}{=} \{ s \doteq t \}_1$, an equation set of similarity degree 1 (in any order as long as a rule applies and its similarity degree is not zero) always terminates. And when the final equation set is a substitution σ at approximation degree α, σ is the most general substitution (up to a variable renaming) that is a solution at approximation degree α (*i.e.*, $s\sigma \approx_{\alpha} t\sigma$), and α is the greatest approximation degree for which this is true. □

Example 8. [***$\mathcal{FOT}$ fuzzy unification with similar functors of different arities***] Take a functor signature such that: $\{a, b, c, d\} \subseteq \Sigma_0$, $\{f, g, \ell\} \subseteq \Sigma_2$, $\{h\} \subseteq \Sigma_3$; and let us further assume the functor similarity that is the reflexive symmetric transitive closure of[24]:

$$a \approx_{.7} b,\ \ c \approx_{.6} d,\ \ f \approx_{.9}^{\{1\to 2, 2\to 1\}} g,\ \ g \approx_{.9}^{\{1\to 2, 2\to 1\}} f,\ \text{ and } \ell \approx_{.8}^{\{1\to 2, 2\to 3\}} h.$$

[24] Recall that the argument mapping is the identity by default.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Let us consider the fuzzy equation set $\{ t_1 \doteq t_2 \}_1$:

$$\left\{ h(X, g(Y, b), f(Y, c)) \doteq \ell(f(a, Z), g(d, c)) \right\}_1 \tag{18}$$

and let us apply the rules of Fig. 7 where rule **Weak Term Decomposition** has been replaced by the rules of Fig. 11:

Rule **Fuzzy Equation Orientation** with $\alpha = 1$ because $\mathbf{arity}(\ell) < \mathbf{arity}(h)$; new set: $\{ \ell(f(a, Z), g(d, c)) \doteq h(X, g(Y, b), f(Y, c)) \}_1$;

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** with $\alpha = 1$ and $\beta = .8$ since $\ell \approx^{\{1\to 2, 2\to 3\}}_{.8} h$; new set: $\{ f(a, Z) \doteq g(Y, b), g(d, c) \doteq f(Y, c) \}_{.8}$;

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** to $f(a, Z) \doteq g(Y, b)$ with $\alpha = .8$ and $\beta = .9$ since $f \approx^{\{1\to 2, 2\to 1\}}_{.9} g$; new set: $\{ a \doteq b, Z \doteq Y, g(d, c) \doteq f(Y, c) \}_{.8}$;

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** to $a \doteq b$ with $\alpha = .8$ and $\beta = .7$ since $a \approx_{.7} b$; new set: $\{ Z \doteq Y, g(d, c) \doteq f(Y, c) \}_{.7}$;

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** to $g(d, c) \doteq f(Y, c)$ with $\alpha = .7$ and $\beta = .9$ since $f \approx^{\{1\to 2, 2\to 1\}}_{.9} g$; new set: $\{ Z \doteq Y, d \doteq c, c \doteq Y \}_{.7}$;

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** to $d \doteq c$ with $\alpha = .7$ and $\beta = .6$ since $d \approx_{.6} c$; new set: $\{ Z \doteq Y, c \doteq Y \}_{.6}$;

Rule **Equation Orientation** to $c \doteq Y$ with $\alpha = .6$; new set:: $\{ Z \doteq Y, Y \doteq c \}_{.6}$.

Rule **Variable Elimination** to $Y \doteq c$ with $\alpha = .6$; new set: $\{ Z \doteq c, Y \doteq c \}_{.6}$.

This last equation set at approximation degree .6 is in normal form and defines the substitution $\sigma = \{ c/Z, c/Y \}$ so that: $t_1\sigma = h(X, g(Y, b), f(Y, c))\sigma \approx_{.6} \ell(f(a, Z), g(d, c))\sigma = t_2\sigma$; that is: $t_1\sigma = h(X, g(c, b), f(c, c)) \approx_{.6} \ell(f(a, c), g(d, c)) = t_2\sigma$.

Rule **Fuzzy Non-Aligned-Argument Term Decomposition** is a very general rule for normalizing fuzzy equations over $\mathcal{FOT}$ structures. It has the following convenient properties:

1. it accounts for fuzzy mismatches of similar functors of possibly different arity or order of arguments;
2. when restricted to tolerating only similar equal-arity functors with matching argument positions, it reduces to Sessa's weak unification's **Weak Term Decomposition** rule;
3. when similarity degrees are further restricted to be in $\{0, 1\}$, it is the Herbrand-Martelli-Montanari **Term Decomposition** rule;
4. it requires no alteration of the standard notions of $\mathcal{FOT}$s and $\mathcal{FOT}$ substitutions: similarity among $\mathcal{FOT}$s is derived from that of signature symbols;
5. finally, and most importantly, it keeps fuzzy unification in the same complexity class as crisp unification: that of Union-Find [43].[25]

As a result, it is more general than all other extant approaches we know which propose a fuzzy $\mathcal{FOT}$ unification operation. The same will be established for the fuzzification of the dual operation: first a limited "*functor-weak*" $\mathcal{FOT}$ generalization corresponding to the dual operation of Sessa's "weak" unification, then to a more expressive "*functor/arity-weak*" $\mathcal{FOT}$ generalization corresponding to our extension of Sessa's unification to functor/arity weak unification.

4.2. *Fuzzy generalization*

While there has been relatively intense interest in devising a fuzzy $\mathcal{FOT}$ unification operation, we know of ***no*** work regarding its dual operation, fuzzy $\mathcal{FOT}$ generalization. This comes as no surprise since even in the crisp case only marginal attention has been paid to generalization (*a.k.a.* anti-unification) as compared to unification.

The Reynolds-Plotkin characterization of $\mathcal{FOT}$ subsumption as a lattice ordering relies on formalizing this ordering as $\mathcal{FOT}$ instantiation. Namely, $t_1 \preceq t_2$ iff there exists a variable substitution σ such that $t_1 = t_2\sigma$. Then, unification

[25] Quasi-linear; *i.e.*, linear with a log … log coefficient [44].

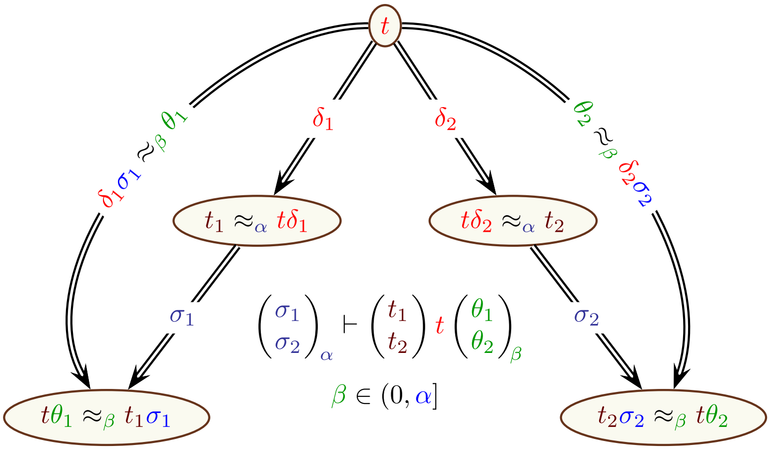


Fig. 12. Fuzzy generalization judgment validity as a constraint.

and generalization are respectively the **glb** and **lub** operations for this ordering and are specified in terms of variable substitutions.

It is clear however, as overviewed in the previous section, that there are several ways one can propose to fuzzify $\mathcal{FOT}$ unification. As a consequence of this, for each specific fuzzification of $\mathcal{FOT}$ unification, and therefore of associated specific fuzzy subsumption ordering on $\mathcal{FOT}$s, there should also correspond a dual operation of fuzzy generalization of $\mathcal{FOT}$s.

In what follows, we first elaborate some lattice-theoretic consequences for Maria Sessa's "weak unification" fuzzy operation on $\mathcal{FOT}$s presented in [12]. In particular, we derive its corresponding fuzzy dual lattice operation that we shall dub "weak $\mathcal{FOT}$ generalization." We then extend this lattice to signatures admitting similar functors with differing arity or argument order.

Fuzzy functor-weak generalization

Let t_1 and t_2 be two $\mathcal{FOT}$s in $\mathcal{T}$ to generalize. We shall use the following notation for a fuzzy generalization judgment:

$$\begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha \vdash \begin{pmatrix}t_1\\t_2\end{pmatrix} t \begin{pmatrix}\theta_1\\\theta_2\end{pmatrix}_\beta \tag{19}$$

given:

- $\sigma_i \in \mathbf{SUBST}_\mathcal{T}$ $(i = 1, 2)$: two prior substitutions with prior similarity degree α,
- t_i $(i = 1, 2)$: two prior $\mathcal{FOT}$s,
- t: a posterior $\mathcal{FOT}$,
- $\theta_i \in \mathbf{SUBST}_\mathcal{T}$ $(i = 1, 2)$: two posterior substitutions with similarity degree β.

Definition 11 *(*Fuzzy $\mathcal{FOT}$ Generalization Judgment Validity*). A fuzzy $\mathcal{FOT}$ generalization judgment such as* (19) *is valid whenever, for $i = 1, 2$:*

1. $\beta \in (0, \alpha]$;
2. $t_i\sigma_i \approx_\beta t\theta_i$;
3. $\exists\delta_i \in \mathbf{SUBST}_\mathcal{T}$ *s.t.* $t_i \approx_\alpha t\delta_i$ *and* $\theta_i \approx_\beta \delta_i\sigma_i$ *(i.e.,* $t_i \preceq_\alpha t\sigma_i$ *and* $\theta_i \preceq_\beta \sigma_i$*).*

Fig. 12 shows an illustration of a valid fuzzy generalization judgment constraint as a commutative diagram.

Fuzzy Equal Variables

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_\alpha \vdash \begin{pmatrix}X\\ X\end{pmatrix} X \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_\alpha$$

Fuzzy Variable-Term

$[t_1 \in \mathcal{V} \text{ or } t_2 \in \mathcal{V};\; t_1 \neq t_2;\; X \text{ is new}]$

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_\alpha \vdash \begin{pmatrix}t_1\\ t_2\end{pmatrix} X \begin{pmatrix}\sigma_1\{t_1/X\}\\ \sigma_2\{t_2/X\}\end{pmatrix}_\alpha$$

Dissimilar Functors

$[f \nsim g;\; m \geq 0, n \geq 0;\; X \text{ is new}]$

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_\alpha \vdash \begin{pmatrix}f(s_1,\ldots,s_m)\\ g(t_1,\ldots,t_n)\end{pmatrix} X \begin{pmatrix}\sigma_1\{f(s_1,\ldots,s_m)/X\}\\ \sigma_2\{g(t_1,\ldots,t_n)/X\}\end{pmatrix}_\alpha$$

Similar Functors

$\left[f \sim_\beta g;\; \beta > 0;\; n \geq 0;\; \alpha_0 \stackrel{\text{def}}{=} \alpha \wedge \beta\right]$

$$\frac{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix}s_1\\ t_1\end{pmatrix} \Uparrow_{\alpha_0} \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1\\ \sigma_2^1\end{pmatrix}_{\alpha_1} \;\ldots\; \begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix}_{\alpha_{n-1}} \vdash \begin{pmatrix}s_n\\ t_n\end{pmatrix} \Uparrow_{\alpha_{n-1}} \begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix} u_n \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}_{\alpha_n}}{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_\alpha \vdash \begin{pmatrix}f(s_1,\ldots,s_n)\\ g(t_1,\ldots,t_n)\end{pmatrix} f(u_1,\ldots,u_n) \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}_{\alpha_n}}$$

Fig. 13. Functor-weak generalization axioms and rule.

Definition 12 (Fuzzy Generalization Rule Correctness). *A fuzzy generalization rule is correct iff, whenever the side condition holds, if all the fuzzy generalization judgments making up its antecedent are valid, then necessarily the fuzzy generalization judgment in its consequent is valid.*

In Fig. 13, we give a fuzzy version of the generalization rules of Fig. 5. As was the case in Sessa's weak unification, we assume as well for now that we are given a similarity relation $\sim: \Sigma \times \Sigma \to [0, 1]$ on the signature $\Sigma = \cup_{n\geq 0}\Sigma_n$ such that for all $m \geq 0$ and $n \geq 0$, $m \neq n$ implies $f \nsim g$. In other words, functors of different arities may not be similar.

Rule **Similar Functors** uses a "*fuzzy unapply*" operation ('$\Uparrow_\alpha$') on a pair of terms (t_1, t_2) given a pair of substitutions (σ_1, σ_2) and a similarity degree α. It is the result of "*unapplying*" σ_i from t_i, for $i = 1, 2$, into a common variable X, if any such exists such that the terms $X\sigma_i$ are respectively similar to t_i with similarity degrees α_i. It returns a fuzzy pair of terms and a similarity degree in $(0, \alpha]$ defined as:

$$\begin{pmatrix}t_1\\ t_2\end{pmatrix} \Uparrow_\alpha \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \stackrel{\text{def}}{=} \begin{cases} \begin{pmatrix}X\\ X\end{pmatrix}_{\alpha\wedge\alpha_1\wedge\alpha_2} & \text{if } \exists X \in \mathcal{V},\; t_i \sim_{\alpha_i} X\sigma_i \text{ for some } \alpha_i \in (0,1]\; i = 1,2; \\ \begin{pmatrix}t_1\\ t_2\end{pmatrix}_\alpha & \text{otherwise.} \end{cases} \tag{20}$$

In Equation (20), the "*if*" condition is "$\exists X \in \mathcal{V},\; t_i \sim_{\alpha_i} X\sigma_i$" for some $\alpha_i \in (0, 1]$ $(i = 1, 2)$. But could there be two such variables? That is,

$$\exists X \in \mathcal{V}, \exists Y \in \mathcal{V},\; X \neq Y, \text{ s.t. } t_i \sim_{\alpha_i} X\sigma_i \text{ and } t_i \sim_{\beta_i} Y\sigma_i \tag{21}$$

for some $\alpha_i \in (0, 1]$ and $\beta_i \in (0, 1]$ $(i = 1, 2)$. Note that a new variable is introduced in the generalizing pair of substitutions only in Axiom **Fuzzy Variable-Term** and Axiom **Dissimilar Functors**. Then, each axiom binds the new variable in the two substitutions to two terms that are dissimilar at any similarity degree (as required by their side conditions). However, by Lemma 4, Condition (21) would imply that:

$$t_i \sim_{\alpha_i\wedge\beta_i} X\sigma_i \sim_{\alpha_i\wedge\beta_i} Y\sigma_i$$

with $\alpha_i \wedge \beta_i \in (0, 1]$, for $i = 1, 2$. This would mean that X or Y was introduced while the side condition of neither Axiom **Fuzzy Variable-Term** nor Axiom **Dissimilar Functors** was verified; which is impossible. Thus, there can be at most only one such variable.

Importantly, note that fuzzy unapplication defined by Equation (20) returns a pair of terms and a (possibly lesser) approximation degree, unlike crisp unapplication defined by Equation (7) which returns only a pair of terms. Because of this, when we write a fuzzy judgment such as:

$$\begin{pmatrix}\sigma\\ \sigma'\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}t\\ t'\end{pmatrix} \uparrow_{\alpha} \begin{pmatrix}\sigma\\ \sigma'\end{pmatrix} u \begin{pmatrix}\theta\\ \theta'\end{pmatrix}_{\beta} \tag{22}$$

as we do in Rule **Similar Functors**, this is shorthand to indicate that the posterior similarity degree β is *at most* the one returned by the fuzzy unapplication $\begin{pmatrix}t\\ t'\end{pmatrix} \uparrow_{\alpha} \begin{pmatrix}\sigma\\ \sigma'\end{pmatrix}$. Formally, the notation of the fuzzy judgment (22) is equivalent to:

$$\begin{pmatrix}t\\ t'\end{pmatrix} \uparrow_{\alpha} \begin{pmatrix}\sigma\\ \sigma'\end{pmatrix} = \begin{pmatrix}s\\ s'\end{pmatrix}_{\beta'} \quad \text{and} \quad \begin{pmatrix}\sigma\\ \sigma'\end{pmatrix}_{\beta'} \vdash \begin{pmatrix}s\\ s'\end{pmatrix} u \begin{pmatrix}\theta\\ \theta'\end{pmatrix}_{\beta} \tag{23}$$

for some β' such that $\beta \leq \beta' \leq \alpha$. This is because a fuzzy unapplication invoked while proving the validity of a fuzzy judgment may require, by Expression (20), lowering the *prior* approximation degree of the judgment.

Note also that Rule "**Similar Functors**" is defined for $n \geq 0$. For $n = 0$, it becomes the following fuzzy judgment:

$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}c\\ c\end{pmatrix} c \begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha} \tag{24}$$

which can be verified to be an axiom since it is valid at any approximation degree α in [0, 1], for any constant c in Σ_0, and any substitutions σ_1 and σ_2 in $\mathbf{Subst}_{\mathcal{T}}$, thanks to the reflexivity of the similarity $\sim_\alpha$ on $\mathcal{T}$.

Referring to the axioms (seen as rules with no antecedent) and the rule of Fig. 13 on Page 28, we establish the following fact corresponding to Lemma 2 (taking $\sigma_i^0 \stackrel{\text{def}}{=} \sigma_i$, for $i = 1, 2$), where the fuzzy ordering on substitutions is defined in Definition 7.

Lemma 5. *In Rule* **Similar Functors** *of Fig. 13, taking $\sigma_i^0 \stackrel{\text{def}}{=} \sigma_i$, for $i = 1, 2$, the approximation degrees $\alpha_i^0, \ldots, \alpha_i^n$ are such that $\alpha_i^k \leq \alpha_i^{k-1}$, and the substitutions $\sigma_i^0, \ldots, \sigma_i^n$ are such that $\sigma_i^k \preceq_{\alpha_i^k} \sigma_i^{k-1}$, for all k, $1 \leq k \leq n$ $(i = 1, 2)$.*

Proof. We proceed by induction on the depth d of the terms; *i.e.*, we consider only terms of depth less than or equal to d.

1. $d = 0$: This limits terms to constants and variables. The inequality between prior and posterior substitutions is verified for the three first axioms of Fig. 13: each preserves the prior approximation degree and the posterior substitutions are all either equal to the corresponding prior substitutions or of the form $\theta = \sigma\{t/X\}$ where X is a new variable and σ is the corresponding prior substitution. As well, when limited to terms of 0 depth, Rule **Similar Functors** becomes the Axiom (24), which preserves both the approximation degree and the substitutions.
2. $d > 0$: Let us assume now that this is true for all terms of depth less than d. That is, we consider two terms to generalize, at least one of which is of depth d. The same argument given above for when $d = 0$ for the three first axioms of Fig. 5 justifies concluding that $\theta \preceq \sigma$, since this is true in these cases for terms of any depth. As for Rule **Equal Functors**, there are two possible cases for the two terms in its consequent (the "denominator"):
   (a) $n = 0$: then, the conclusion follows true by Axiom (24);
   (b) $n \geq 0$: since the fuzzy unapply operation (20) yields either a pair of terms having the same depth as the corresponding terms it is applied to, or 0 (because it can only be a new variable), we can say that all the terms resulting from fuzzy-unapplied pairs of arguments in the judgments of the rule's antecedent (the "numerator") are of depth at most $d - 1$. Therefore, this fact, together with our induction hypothesis being verified for depths less than d and the expression of a fuzzy judgment (23) involving only terms of such depths, we can conclude that all the judgments in the rule's antecedent can only reduce their prior approximation degree. Therefore, $\alpha_i^k \leq \alpha_i^{k-1}$ and $\sigma_i^k \preceq_{\alpha_i^k} \sigma_i^{k-1}$, for all $k = 1, \ldots, n$. Then, by Corollary 3 and transitivity of the "more general" ordering on substitutions $\preceq_\alpha$ at fixed α, the conclusion follows.

Hence, this establishes that, for both $i = 1, 2$, the approximation degree α_i^k is monotonically decreasing and the substitution σ_i^k is monotonically refined from more general to less, as k increases from 1 to n; which concludes our proof. □

And the corresponding corollary also follows.

Corollary 4. *In Rule* Similar Functors *of Fig. 13, for all k,* $1 \leq k \leq n$*:*

- *the approximation degrees* α_i^k *are such that* $\alpha_i^n \leq \alpha_i^{n-1} \leq \ldots \alpha_i^1 \leq \alpha_i^0$*, and*
- *the substitutions* σ_i^k *are such that* $\sigma_i^n \preceq_{\alpha_i^n} \sigma_i^{n-1} \preceq_{\alpha_i^{n-1}} \ldots \sigma_i^1 \preceq_{\alpha_i^1} \sigma_i^0$*,*

for $i = 1, 2$.

Theorem 5 *(*Functor-Weak Generalization Correctness*). The fuzzy generalization rules of Fig. 13 are correct.*

Proof. We must show that they verify the conditions of Definition 12. For each of the three axioms of Fig. 13, this means that they must be valid as fuzzy judgments, verifying the three conditions of Definition 11, which are:

– *Condition 1*: $\beta \in (0, \alpha]$,
– *Condition 2*: $t_i\sigma_i \sim_\beta t\theta_i$,
– *Condition 3*: $t_i \preceq_\alpha t$ and $\theta_i \preceq_\beta \sigma_i$,

for $i = 1, 2$, for a fuzzy $\mathcal{FOT}$ generalization judgment such as (19). These conditions for the axioms and the rule of Fig. 13 translate as the following.

Condition 1. All three axioms verify this condition because they preserve the approximation degree.

Condition 2. This condition becomes the following for each of the three axioms (for $i = 1, 2$):

– **Fuzzy Equal Variables**: Condition 2 becomes the similarity $X\sigma_i \sim_\alpha X\sigma_i$, which is true by reflexivity of $\sim_\alpha$ for all X, σ_i, and α;
– **Fuzzy Variable-Term**: it becomes the similarity $t_i\sigma_i \sim_\alpha t_i\sigma_i$, which is true also by reflexivity of $\sim_\alpha$, for all t_i, σ_i, and α;
– **Dissimilar Functors**: Condition 2 becomes:

$$\begin{array}{rcl} f(s_1,\ldots,s_m)\sigma_1 & \sim_\alpha & X\sigma_1\{f(s_1,\ldots,s_m)/X\} \\ g(t_1,\ldots,t_n)\sigma_2 & \sim_\alpha & X\sigma_2\{g(t_1,\ldots,t_n)/X\} \end{array}$$

which, because X is a new variable that does not occur in either σ_1 or σ_2, simplify respectively to the similarities:

$$\begin{array}{rcl} f(s_1,\ldots,s_m) & \sim_\alpha & f(s_1,\ldots,s_m) \\ g(t_1,\ldots,t_n) & \sim_\alpha & g(t_1,\ldots,t_n) \end{array}$$

which hold by reflexivity of $\sim_\alpha$ at any approximation degree α.

Condition 3. The three axioms verify the following at all approximation degrees α and β (for $i = 1, 2$):

– **Fuzzy Equal Variables**: $X \preceq_\alpha X$ and $\sigma_i \preceq_\beta \sigma_i$;
– **Fuzzy Variable-Term**: $t_i \preceq_\alpha X$ and $\sigma_i\{t_i/X\} \preceq_\beta \sigma_i$;
– **Dissimilar Functors**:

$$\begin{array}{rclcrcl} f(s_1,\ldots,s_m) & \preceq_\alpha & X & \text{and} & \sigma_1\{f(s_1,\ldots,s_m)/X\} & \preceq_\beta & \sigma_1, \\ g(t_1,\ldots,t_n) & \preceq_\alpha & X & \text{and} & \sigma_2\{g(t_1,\ldots,t_n)/X\} & \preceq_\beta & \sigma_2. \end{array}$$

As for Rule **Similar Functors**, as required by Definition 12, we must show that if all the fuzzy judgments in the numerator are valid, then the fuzzy judgment in the denominator is valid too. For all three conditions, let us proceed by induction on the arity n:

Condition 1. For $n = 0$, the conclusion follows also because Axiom (24) applies and it also preserves the approximation degree; for $n > 0$, if we assume that $0 \leq \alpha_k \leq \alpha_{k-1} \leq 1$ for all $k = 1, \ldots, n$, by transitivity of $\leq$ on [0, 1], it follows that $0 \leq \alpha_n \leq \alpha_0 \leq 1$, which verifies the definition.

Condition 2. For $n = 0$, this rule becomes Axiom (24). Since it preserves the approximation degree, Condition 1 is verified. Also, this fuzzy judgment is trivially valid at all approximation degrees: the conditions of Definition 11 become the reflexive similarity $c \sim_\alpha c$, and the conjunction of reflexive fuzzy inequality $c \preceq_\alpha c$ and reflexive substitution fuzzy inequalities $\sigma_i \preceq_\alpha \sigma_i$, for $i = 1, 2$. Thus, this verifies both Condition 2 and Condition 3 for $n = 0$.

For $n > 0$, for each argument-position $k = 1, \ldots, n$, a fuzzy judgment in the rule's antecedent is of the form:

$$\begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix}_{\alpha_{k-1}} \vdash \begin{pmatrix} s_k \\ t_k \end{pmatrix} \uparrow_{\alpha_{k-1}} \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} u_k \begin{pmatrix} \sigma_1^{k} \\ \sigma_2^{k} \end{pmatrix}_{\alpha_k};$$

that is, the form of Expression (22), whose formal meaning is given as Expression (23), which in the above case is equivalent to:

$$\begin{pmatrix} v_1^{k} \\ v_2^{k} \end{pmatrix}_{\beta_k} \stackrel{\text{def}}{=} \begin{pmatrix} s_k \\ t_k \end{pmatrix} \uparrow_{\alpha_{k-1}} \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} \sigma_1^{k-1} \\ \sigma_2^{k-1} \end{pmatrix}_{\beta_k} \vdash \begin{pmatrix} v_1^{k} \\ v_2^{k} \end{pmatrix} u_k \begin{pmatrix} \sigma_1^{k} \\ \sigma_2^{k} \end{pmatrix}_{\alpha_k}$$

for some β_k s.t. $\alpha_{k-1} \leq \beta_k \leq \alpha_k$. Let us now assume that all the fuzzy judgment in the rule's antecedent are valid. That is, for $k = 1, \ldots, n$ (defining $\alpha_0 \stackrel{\text{def}}{=} \alpha \wedge \beta$), for $i = 1, 2$:

$$u_k \sigma_i^k \sim_{\alpha_k} v_i^k \sigma_i^{k-1} \tag{25}$$

and (defining $\sigma_i^0 \stackrel{\text{def}}{=} \sigma_i$):

$$v_i^k \preceq_\alpha u_k \text{ and } \sigma_i^k \preceq_\beta \sigma_i^{k-1}. \tag{26}$$

By Equation (20), this means:

$$\begin{pmatrix} v_1^{k} \\ v_2^{k} \end{pmatrix}_{\alpha_k} \stackrel{\text{def}}{=} \begin{cases} \begin{pmatrix} X \\ X \end{pmatrix}_{\alpha_{k-1} \wedge \beta_1^k \wedge \beta_2^k} & \text{if } \exists X \in \mathcal{V} \text{ s.t. } s_k \sim_{\beta_1^k} X\sigma_1^{k-1} \text{ and } t_k \sim_{\beta_2^k} X\sigma_2^{k-1}; \\ \begin{pmatrix} s_k \\ t_k \end{pmatrix}_{\alpha_{k-1}} & \text{otherwise.} \end{cases}$$

for some β_1^k and β_2^k in (0, 1]. In other words, for each $k = 1, \ldots, n$, there are two cases:

1. $s_k \sim_{\beta_1^k} X\sigma_1^{k-1}$ and $t_k \sim_{\beta_2^k} X\sigma_2^{k-1}$ for some variable X; then, by Axiom **Fuzzy Equal Variables**, we must have $\alpha_k = \alpha_{k-1} \wedge \beta_1^k \wedge \beta_2^k$, $u_k = X$, and $\sigma_i^k = \sigma_i^{k-1}$ for $i = 1, 2$; thus, $\alpha_k \leq \alpha_{k-1}$ and Similarity (25) becomes $u_k \sigma_i^k \sim_{\alpha_k} X\sigma_i^{k-1}$ So that:

$$\begin{array}{lllllllll} s_k\sigma_1^{k-1} & \sim_{\alpha_k} & X\sigma_1^{k-1}\sigma_1^{k-1} & = & X\sigma_1^{k-1} & = & X\sigma_1^k & \sim_{\alpha_k} & u_k\sigma_1^k, \\ t_k\sigma_2^{k-1} & \sim_{\alpha_k} & X\sigma_2^{k-1}\sigma_2^{k-1} & = & X\sigma_2^{k-1} & = & X\sigma_2^k & \sim_{\alpha_k} & u_k\sigma_2^k. \end{array}$$

2. There is no such variable X; in which case, $\alpha_k = \alpha_{k-1}$ and Similarity (25) becomes:

$$\begin{array}{lll} s_k\sigma_1^{k-1} & \sim_{\alpha_k} & u_k\sigma_1^k, \\ t_k\sigma_2^{k-1} & \sim_{\alpha_k} & u_k\sigma_2^k. \end{array}$$

Thus, by the only non-identical transformation relating prior and posteriors substitutions in the axioms, for any argument position k, $1 \leq k \leq n$, we have:

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

$$\sigma_i^k \sim_{\alpha_k} \sigma_i^0\{\tau_1/X_1\}\ldots\{\tau_\ell/X_\ell\}$$

where each of the variables $X_1 \ldots X_\ell$, with $0 \leq \ell$, is a variable possibly introduced in proving the validity of the fuzzy judgment corresponding to some argument position k. Therefore, since for any argument position k, $1 \leq k \leq n$:

1. σ_i^k affects only a *new* variable introduced in one of the axioms verifying the validity of the subterm at argument position k; and,
2. such a newly introduced variable now occurring in u_k is always instantiated by the same term;

it comes that, at approximation degree α_k:

$$s_k\sigma_1^0 \sim_{\alpha_k} s_k\sigma_1^1 \sim_{\alpha_k} \ldots \sim_{\alpha_k} s_k\sigma_1^{k-1}$$
$$t_k\sigma_2^0 \sim_{\alpha_k} t_k\sigma_2^1 \sim_{\alpha_k} \ldots \sim_{\alpha_k} t_k\sigma_2^{k-1}$$

as well as, at approximation degree α_n:

$$u_k\sigma_1^k \sim_{\alpha_n} u_k\sigma_1^{k+1} \sim_{\alpha_n} \ldots \sim_{\alpha_n} u_k\sigma_1^n$$
$$u_k\sigma_2^k \sim_{\alpha_n} u_k\sigma_2^{k+1} \sim_{\alpha_n} \ldots \sim_{\alpha_n} u_k\sigma_2^n$$

which shows that in both cases we have, for all $k = 1, \ldots, n$:

$$s_k\sigma_1^0 \sim_{\alpha_k} u_k\sigma_1^n$$
$$t_k\sigma_2^0 \sim_{\alpha_k} u_k\sigma_2^n.$$

Therefore, for $k = n$:

$$f(s_1, \ldots, s_n)\sigma_1^0 \sim_{\alpha_n} f(u_1, \ldots, u_n)\sigma_1^n$$
$$f(t_1, \ldots, t_n)\sigma_2^0 \sim_{\alpha_n} f(u_1, \ldots, u_n)\sigma_2^n$$

which completes the proof of Condition 2.

Condition 3. This condition becomes, for all $k = 1, \ldots, n$:

$$\begin{array}{rl} f(s_1, \ldots, s_n) \preceq_{\alpha_{k-1}} f(u_1, \ldots, u_n) & \text{and} \quad \sigma_1^k \preceq_{\alpha_k} \sigma_1^{k-1} \\ g(t_1, \ldots, t_n) \preceq_{\alpha_{k-1}} g(u_1, \ldots, u_n) & \text{and} \quad \sigma_2^k \preceq_{\alpha_k} \sigma_2^{k-1} \end{array}$$

from which, since $\alpha_k \leq \alpha_{k-1}$ for all $k = 1, \ldots, n$, it follows that:

$$\begin{array}{rl} f(s_1, \ldots, s_n) \preceq_{\alpha_n} f(u_1, \ldots, u_n) & \text{and} \quad \sigma_1^n \preceq_{\alpha_n} \sigma_1^0 \\ g(t_1, \ldots, t_n) \preceq_{\alpha_n} g(u_1, \ldots, u_n) & \text{and} \quad \sigma_2^n \preceq_{\alpha_n} \sigma_2^0 \end{array}$$

or indifferently, using the same similarity class representative in both cases since $f \sim_{\alpha_n} g$ (because $f \sim_\beta g$ and $\alpha_n \leq \beta$):

$$g(t_1, \ldots, t_n) \preceq_{\alpha_n} f(u_1, \ldots, u_n) \quad \text{and} \quad \sigma_2^n \preceq_{\alpha_n} \sigma_2^0$$

which completes the proof of Condition 3, and the proof of Theorem 5. □

Example 9. [*Fuzzy generalization with similar functors of same arities*] Consider the signature Σ containing $\Sigma_0 = \{a, b, c, d\}$, and $\Sigma_2 = \{f, g\}$, and the closure $\sim$ of the similar pairs $a \sim_{.7} b$, $c \sim_{.6} d$, and $f \sim_{.8} g$. Let us apply the functor-weak generalization axioms and rule Fig. 13 to $t_1 \stackrel{\text{def}}{=} g(c, d)$, and $t_2 \stackrel{\text{def}}{=} f(a, b)$; that is, let us find term t, substitutions $\sigma_i \in \textbf{SUBST}_{\mathcal{T}}$ $(i = 1, 2)$, and similarity degree α in $[0, 1]$ such that $t\sigma_1 \sim_\alpha g(c, d)$ and $t\sigma_2 \sim_\alpha f(a, b)$. This is expressed as the following fuzzy judgment:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} g(c, d) \\ f(a, b) \end{pmatrix} t \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha .$$

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

By Rule **Similarity Functors**, we infer that $t = g(u_1, u_2)$[26]:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} g(c,d) \\ f(a,b) \end{pmatrix} g(u_1,u_2) \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha$$

which, replaced by the antecedents of Rule **Similarity Functors**, becomes (since $g \sim_{.8} f$):

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_{.8} \vdash \begin{pmatrix} c \\ a \end{pmatrix} \Uparrow_{.8} \begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} u_1 \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix}_{\alpha'}, \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix}_{\alpha'} \vdash \begin{pmatrix} d \\ b \end{pmatrix} \Uparrow_{\alpha'} \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix} u_2 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha .$$

Since the prior substitutions of the first judgment are empty, evaluating its fuzzy unapplication (using Expression (23) in which $\beta' = \alpha$) yields the sequence:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_{.8} \vdash \begin{pmatrix} c \\ a \end{pmatrix} u_1 \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix}_{\alpha'}, \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix}_{\alpha'} \vdash \begin{pmatrix} d \\ b \end{pmatrix} \Uparrow_{\alpha'} \begin{pmatrix} \sigma_1' \\ \sigma_2' \end{pmatrix} u_2 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha .$$

By Axiom **Dissimilar Functors**, it comes that $u_1 = X_1$, a new variable, and the sequence becomes:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_{.8} \vdash \begin{pmatrix} c \\ a \end{pmatrix} X_1 \begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix}_{.8}, \begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix}_{.8} \vdash \begin{pmatrix} d \\ b \end{pmatrix} \Uparrow_{.8} \begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix} u_2 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha .$$

The validity of the first fuzzy judgment is thereby established. We proceed with the remaining fuzzy judgment evaluating its fuzzy unapplication using Equation (20). Since X_1 is such that $d \sim_{.6} X_1\{c/X_1\} = c$ and $a \sim_{.7} X_1\{b/X_1\} = b$, it verifies the first of the conditions of Equation (20). Therefore, the new approximation degree of the judgment is $.8 \wedge .6 \wedge .7 = .6$, and $u_2 = X_1$ so that the judgment becomes:

$$\begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix}_{.6} \vdash \begin{pmatrix} d \\ b \end{pmatrix} \Uparrow_{.6} \begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix} X_1 \begin{pmatrix} \{\, c/X_1 \,\} \\ \{\, a/X_1 \,\} \end{pmatrix}_{.6} .$$

This validates the last judgment completing the fuzzy generalization whereby $t = g(X_1, X_1)$ is the least fuzzy generalizer of $t_1 = g(c,d)$, and $t_2 = f(a,b)$ at approximation degree .6 with $\sigma_1 = \{\, c/X_1 \,\}$ so that $t\sigma_1 = g(c,c) \sim_{.6} t_1$; and, $\sigma_2 = \{\, a/X_1 \,\}$ so that $t\sigma_2 = g(a,a) \sim_{.6} t_2$.

Fuzzy functor/arity-weak generalization

In Fig. 14, we give a fuzzy version of the generalization rules taking into account mismatches not only in functors, but also in arities; *i.e.*, number and/or order of arguments. We now assume that we are not only given a similarity relation $\sim : \Sigma \times \Sigma \to [0, 1]$ on the signature $\Sigma = \cup_{n \geq 0} \Sigma_n$, but also that functors of different arities may be similar with some non-zero similarity degree as specified by a one-to-one argument-position mapping for each pair of so-similar functors associating each argument position of the functor of least arity with a distinct argument position of the functor of larger arity. The only rule among those of Fig. 13 that differs is the last one (**Similar Functors**) which is now a pair of rules called **Functor/Arity Similarity Left** and **Functor/Arity Similarity Right** as they account for non-identical correspondence among similar functors's argument positions whether in the left or in the right of the pair of terms to generalize, depending on which side has less arguments. If the arities are the same, the two rules are equivalent (each and all the arguments of the two terms are paired in bijection by a position permutation).

Theorem 6 (Functor/Arity-Weak Generalization Correctness). *The fuzzy generalization rules of Fig. 13 where Rule "***Similar Functors***" is replaced with the rules in Fig. 14 are correct.*

Proof. The argument in this proof has exactly the same structure as the argument for the proof of Rule **Similar Functors** of Fig. 13. The only difference is that structural induction on a pair of terms with similar functors to generalize is always limited to the largest possible set of pairs of corresponding argument positions as specified by a one-to-one argument map from all the argument positions of the functor of lesser arity to those of the functor of larger arity, rather than the identity on equal cardinality sets of argument positions. Thus, in the following, parts of the proof

[26] This is a non-deterministic choice of a functor's similarity-class representative. We shall always take the left (or upper, in this notation) term's functor. This, of course, will also result in a non-deterministic choice of representative for any term elaborated in generalization modulo functor similarity. The lower the approximation degree, the larger the similarity class.

Functor/Arity Similarity Left

$$\left[f \approx^p_\beta g;\ \ \beta > 0;\ \ 0 \le m \le n;\ \ \alpha_0 \stackrel{\text{def}}{=} \alpha \wedge \beta\right]$$

$$\frac{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix}s_1\\ t_{p(1)}\end{pmatrix}\uparrow\alpha_0\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1\\ \sigma_2^1\end{pmatrix}_{\alpha_1} \ \cdots\ \begin{pmatrix}\sigma_1^{m-1}\\ \sigma_2^{m-1}\end{pmatrix}_{\alpha_{m-1}} \vdash \begin{pmatrix}s_m\\ t_{p(m)}\end{pmatrix}\uparrow\alpha_{m-1}\begin{pmatrix}\sigma_1^{m-1}\\ \sigma_2^{m-1}\end{pmatrix} u_m \begin{pmatrix}\sigma_1^m\\ \sigma_2^m\end{pmatrix}_{\alpha_m}}{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}f(s_1,\dots,s_m)\\ g(t_1,\dots,t_n)\end{pmatrix} f(u_1,\dots,u_m) \begin{pmatrix}\sigma_1^m\\ \sigma_2^m\end{pmatrix}_{\alpha_m}}$$

Functor/Arity Similarity Right

$$\left[g \approx^p_\beta f;\ \ \beta > 0;\ \ 0 \le n \le m;\ \ \alpha_0 \stackrel{\text{def}}{=} \alpha \wedge \beta\right]$$

$$\frac{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix}s_{p(1)}\\ t_1\end{pmatrix}\uparrow\alpha_0\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1\\ \sigma_2^1\end{pmatrix}_{\alpha_1} \ \cdots\ \begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix}_{\alpha_{n-1}} \vdash \begin{pmatrix}s_{p(n)}\\ t_n\end{pmatrix}\uparrow\alpha_{n-1}\begin{pmatrix}\sigma_1^{n-1}\\ \sigma_2^{n-1}\end{pmatrix} u_n \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}_{\alpha_n}}{\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}f(s_1,\dots,s_m)\\ g(t_1,\dots,t_n)\end{pmatrix} g(u_1,\dots,u_n) \begin{pmatrix}\sigma_1^n\\ \sigma_2^n\end{pmatrix}_{\alpha_n}}$$

Fig. 14. Functor/arity-weak generalization axioms and rule.

that are omitted are identical to their corresponding parts in the proof of Rule **Similar Functors**. Also, for reason of obvious symmetry, we need only provide the detailed proof of correctness of Rule **Functor/Arity Similarity Left**. The proof of correctness of Rule **Functor/Arity Similarity Right** is the pointwise similar dual argument in the other direction.

Considering Rule **Functor/Arity Similarity Left**, as required by Definition 12, we must show that if all the fuzzy judgments in the numerator are valid, then the fuzzy judgment in the denominator is valid too. Since the proofs of Condition 1 and Condition 3 are the same for equal-arity functor similarity, we need only provide a proof of Condition 2 of Definition 12. Let us proceed by induction on the argument-position number k, for $k = 1, \dots, m$, where m is the arity of f (the first of the two terms' functor, with the same or a smaller arity as required by the side condition).

For $m = 0$, this rule becomes Axiom (24). This fuzzy judgment is trivially valid at all approximation degrees: Condition 2 of Definition 11 becomes the reflexive similarity $c \approx_\alpha c$ and Condition 3 becomes the conjunction $c \preceq_\alpha c$ and $\sigma_i \preceq_\alpha \sigma_i$, for $i = 1, 2$. Thus, this verifies both Condition 2 and Condition 3 for $m = 0$.

For $m > 0$, for each argument-position $k = 1, \dots, m$, a fuzzy judgment in the rule's antecedent is of the form:

$$\begin{pmatrix}\sigma_1^{k-1}\\ \sigma_2^{k-1}\end{pmatrix}_{\alpha_{k-1}} \vdash \begin{pmatrix}s_k\\ t_{p(k)}\end{pmatrix}\uparrow\alpha_{k-1}\begin{pmatrix}\sigma_1^{k-1}\\ \sigma_2^{k-1}\end{pmatrix} u_k \begin{pmatrix}\sigma_1^k\\ \sigma_2^k\end{pmatrix}_{\alpha_k};$$

that is, the form of Expression (22), whose formal meaning is given as Expression (23), which in the above case is equivalent to:

$$\begin{pmatrix}v_1^k\\ v_2^k\end{pmatrix}_{\beta_k} \stackrel{\text{def}}{=} \begin{pmatrix}s_k\\ t_{p(k)}\end{pmatrix}\uparrow\alpha_{k-1}\begin{pmatrix}\sigma_1^{k-1}\\ \sigma_2^{k-1}\end{pmatrix} \ \text{and}\ \begin{pmatrix}\sigma_1^{k-1}\\ \sigma_2^{k-1}\end{pmatrix}_{\beta_k} \vdash \begin{pmatrix}v_1^k\\ v_2^k\end{pmatrix} u_k \begin{pmatrix}\sigma_1^k\\ \sigma_2^k\end{pmatrix}_{\alpha_k}$$

for some β_k s.t. $\alpha_{k-1} \le \beta_k \le \alpha_k$. Let us now assume that all the fuzzy judgment in the rule's antecedent are valid. That is, for $k = 1, \dots, m$ (defining $\alpha_0 \stackrel{\text{def}}{=} \alpha \wedge \beta$), for $i = 1, 2$:

$$u_k\sigma_i^k \approx_{\alpha_k} v_i^k\sigma_i^{k-1} \tag{27}$$

and (defining $\sigma_i^0 \stackrel{\text{def}}{=} \sigma_i$):

$$v_i^k \preceq_\alpha u_k \text{ and } \sigma_i^k \preceq_\beta \sigma_i^{k-1}. \tag{28}$$

By Equation (20), this means that for all $k = 1, \dots, m$, v_1^k, v_2^k, and α_k are defined by:

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

$$\begin{pmatrix} v_1^k \\ v_2^k \end{pmatrix}_{\alpha_k} \stackrel{\text{def}}{=} \begin{cases} \begin{pmatrix} X \\ X \end{pmatrix}_{\alpha_{k-1}\wedge\beta_1^k\wedge\beta_2^k} & \text{if } \exists X \in \mathcal{V} \text{ s.t.} \begin{pmatrix} s_k & \approx_{\beta_1^k} & X\sigma_1^{k-1} \\ t_{p(k)} & \approx_{\beta_2^k} & X\sigma_2^{k-1} \end{pmatrix}; \\ \begin{pmatrix} s_k \\ t_{p(k)} \end{pmatrix}_{\alpha_{k-1}} & \text{otherwise.} \end{cases}$$

for some β_1^k and β_2^k in (0, 1]. In other words, for each $k = 1, \ldots, m$, there are two cases:

1. $s_k \approx_{\beta_1^k} X\sigma_1^{k-1}$ and $t_{p(k)} \approx_{\beta_2^k} X\sigma_2^{k-1}$ for some variable X; then, by Axiom **Fuzzy Equal Variables**, we must have $\alpha_k = \alpha_{k-1} \wedge \beta_1^k \wedge \beta_2^k$, $u_k = X$, and $\sigma_i^k = \sigma_i^{k-1}$ for $i = 1, 2$; thus, $\alpha_k \leq \alpha_{k-1}$ and Similarity (25) becomes $u_k\sigma_i^k \approx_{\alpha_k} X\sigma_i^{k-1}$ So that:
$$\begin{array}{lllllll} s_k\sigma_1^{k-1} & \approx_{\alpha_k} & X\sigma_1^{k-1}\sigma_1^{k-1} & = & X\sigma_1^{k-1} & = & X\sigma_1^k & \approx_{\alpha_k} & u_k\sigma_1^k, \\ t_{p(k)}\sigma_2^{k-1} & \approx_{\alpha_k} & X\sigma_2^{k-1}\sigma_2^{k-1} & = & X\sigma_2^{k-1} & = & X\sigma_2^k & \approx_{\alpha_k} & u_k\sigma_2^k. \end{array}$$
2. There is no such variable X; in which case, $\alpha_k = \alpha_{k-1}$ and Similarity (25) becomes:
$$\begin{array}{lll} s_k\sigma_1^{k-1} & \approx_{\alpha_k} & u_k\sigma_1^k, \\ t_{p(k)}\sigma_2^{k-1} & \approx_{\alpha_k} & u_k\sigma_2^k. \end{array}$$

Thus, by the only non-identical transformation relating prior and posteriors substitutions in the axioms, for any argument position k, $1 \leq k \leq m$, we have:

$$\sigma_i^k \approx_{\alpha_k} \sigma_i^0\{\tau_1/X_1\}\ldots\{\tau_\ell/X_\ell\}$$

where each of the variables $X_1 \ldots X_\ell$, with $0 \leq \ell$, is a variable possibly introduced in proving the validity of the fuzzy judgment corresponding to some argument position preceding k. Therefore, since for any argument position k, $1 \leq k \leq m$:

1. σ_i^k affects only a *new* variable introduced in one of the axioms verifying the validity of the subterm at argument position k; and,
2. such a newly introduced variable now occurring in u_k is always instantiated by the same term;

it comes that, at approximation degree α_k:

$$\begin{array}{lllllll} s_k\sigma_1^0 & \approx_{\alpha_k} & s_k\sigma_1^1 & \approx_{\alpha_k} & \ldots & \approx_{\alpha_k} & s_k\sigma_1^{k-1} \\ t_{p(k)}\sigma_2^0 & \approx_{\alpha_k} & t_{p(k)}\sigma_2^1 & \approx_{\alpha_k} & \ldots & \approx_{\alpha_k} & t_{p(k)}\sigma_2^{k-1} \end{array}$$

as well as, at approximation degree α_m:

$$\begin{array}{lllllll} u_k\sigma_1^k & \approx_{\alpha_m} & u_k\sigma_1^{k+1} & \approx_{\alpha_m} & \ldots & \approx_{\alpha_m} & u_k\sigma_1^m \\ u_k\sigma_2^k & \approx_{\alpha_m} & u_k\sigma_2^{k+1} & \approx_{\alpha_m} & \ldots & \approx_{\alpha_m} & u_k\sigma_2^m \end{array}$$

This means that in both cases we have, for all $k = 1, \ldots, m$:

$$\begin{array}{lll} s_k\sigma_1^0 & \approx_{\alpha_m} & u_k\sigma_1^m \\ t_{p(k)}\sigma_2^0 & \approx_{\alpha_m} & u_k\sigma_2^m. \end{array}$$

Therefore, for $k = m$:

$$\begin{array}{lll} f(s_1, \ldots, s_m)\sigma_1^0 & \approx_{\alpha_m} & f(u_1, \ldots, u_m)\sigma_1^m \\ f(t_{p(1)}, \ldots, t_{p(m)})\sigma_2^0 & \approx_{\alpha_m} & f(u_1, \ldots, u_m)\sigma_2^m \end{array}$$

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

which completes the proof of Condition 2 of Theorem 6, and that of the theorem because of the facts stated at the outset regarding all other cases each of whose proof is identical to when arities are equal. □

Example 10. [*Fuzzy generalization with similar functors of different arities*] Consider the signature Σ containing $\Sigma_0 = \{a, b, c, d\}$, $\Sigma_2 = \{f, g, l\}$, and $\Sigma_3 = \{h\}$, and the closure $\sim$ of the similar pairs $a \sim_{.7} c \sim_{.6} d$, $f \sim_{.8} g$, and $l \sim_{.9} h$. Let us take all argument-position mappings as the default (identity on least-arity set). Let us apply the fuzzy generalization axioms of Fig. 13 and the rule of Fig. 14 to $t_1 \stackrel{\text{def}}{=} h(g(b, Y), f(Y, c), V)$, and $t_2 \stackrel{\text{def}}{=} l(f(a, Z), g(c, d))$; that is, let us find term t, substitutions $\sigma_i \in \textbf{SUBST}_{\mathcal{T}}$ $(i = 1, 2)$, and similarity degree α in [0, 1], such that $t\sigma_1 \sim_\alpha h(g(b, Y), f(Y, c), V)$ and $t\sigma_2 \sim_\alpha l(f(a, Z), g(c, d))$. This is expressed as the following fuzzy judgment:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_1 \vdash \begin{pmatrix}h(g(b,Y), f(Y,c), V)\\ l(f(a,Z), g(c,d))\end{pmatrix} t \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

By Rule **Functor/Arity Similarity Right**, we can infer that $t = l(u_1, u_2)$:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_1 \vdash \begin{pmatrix}h(g(b,Y), f(Y,c), V)\\ l(f(a,Z), g(c,d))\end{pmatrix} l(u_1, u_2) \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha$$

which, when replaced by the rule's antecedents, since $h \sim_{.9} l$ and $1 \wedge .9 = .9$, becomes the sequence:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.9} \vdash \begin{pmatrix}g(b,Y)\\ f(a,Z)\end{pmatrix} \uparrow_{.9} \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} u_1 \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'}, \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'} \vdash \begin{pmatrix}f(Y,c)\\ g(c,d)\end{pmatrix} \uparrow_{\alpha'} \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix} u_2 \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

By evaluating the fuzzy unapplication in its first judgment, this sequence becomes:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.9} \vdash \begin{pmatrix}g(b,Y)\\ f(a,Z)\end{pmatrix} u_1 \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'}, \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'} \vdash \begin{pmatrix}f(Y,c)\\ g(c,d)\end{pmatrix} \uparrow_{\alpha'} \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix} u_2 \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

By Rule **Functor/Arity Similarity Left**,[27] it comes that $u_1 = g(u_3, u_4)$ and, since $g \sim_{.8} f$ and $.9 \wedge .8 = .8$, the sequence becomes:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.8} \vdash \begin{pmatrix}b\\ a\end{pmatrix} \uparrow_{.8} \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} u_3 \begin{pmatrix}\sigma_1''\\\sigma_2''\end{pmatrix}_{\alpha''}, \begin{pmatrix}\sigma_1''\\\sigma_2''\end{pmatrix}_{\alpha''} \vdash \begin{pmatrix}Y\\ Z\end{pmatrix} \uparrow_{\alpha''} \begin{pmatrix}\sigma_1''\\\sigma_2''\end{pmatrix} u_4 \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'},$$
$$\begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'} \vdash \begin{pmatrix}f(Y,c)\\ g(c,d)\end{pmatrix} \uparrow_{\alpha'} \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix} u_2 \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

By evaluating the fuzzy unapplication in the first judgment, and using Rule **Functor/Arity Similarity Left** in the 0-arity case as Axiom (24), since $b \sim_{.7} a$ and $.8 \wedge .7 = .7$, we have $u_3 = b$, and the sequence becomes:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.7} \vdash \begin{pmatrix}b\\ a\end{pmatrix} b \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.7}, \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.7} \vdash \begin{pmatrix}Y\\ Z\end{pmatrix} \uparrow_{.7} \begin{pmatrix}\emptyset\\\emptyset\end{pmatrix} u_4 \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'},$$
$$\begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{\alpha'} \vdash \begin{pmatrix}f(Y,c)\\ g(c,d)\end{pmatrix} \uparrow_{\alpha'} \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix} u_2 \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

The validity of the first fuzzy judgment is thereby established. We proceed with the remaining sequence of fuzzy judgments evaluating the fuzzy unapplication in the first of its judgments, which sets $\alpha' = .7$:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.7} \vdash \begin{pmatrix}Y\\ Z\end{pmatrix} u_4 \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{.7}, \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix}_{.7} \vdash \begin{pmatrix}f(Y,c)\\ g(c,d)\end{pmatrix} \uparrow_{.7} \begin{pmatrix}\sigma_1'\\\sigma_2'\end{pmatrix} u_2 \begin{pmatrix}\sigma_1\\\sigma_2\end{pmatrix}_\alpha .$$

By Axiom **Fuzzy Variable-Term**, we infer from this that $u_4 = X_1$, a new variable, and the judgments become:

$$\begin{pmatrix}\emptyset\\\emptyset\end{pmatrix}_{.7} \vdash \begin{pmatrix}Y\\ Z\end{pmatrix} X_1 \begin{pmatrix}\{\,Y/X_1\,\}\\ \{\,Z/X_1\,\}\end{pmatrix}_{.7},$$

[27] Since f and g have equal arities, we could also use Rule **Functor/Arity Similarity Right**. This would end in an equivalent final result, modulo functor similarities at the final approximation degree. In the remainder of this example, we shall omit making this remark, and choose the left rule over the right for equal-arity functors.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

$$\begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} f(Y,c) \\ g(c,d) \end{pmatrix} \uparrow_{.7} \begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix} u_2 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha}.$$

The validity of the first fuzzy judgment of the above sequence is thereby established. We proceed with the remainder evaluating the fuzzy unapplication in the first of its judgments, which returns the same pair of terms with the similarity degree kept at .7:

$$\begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} f(Y,c) \\ g(c,d) \end{pmatrix} u_2 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha},$$

and by Rule **Functor/Arity Similarity Left** with $u_2 = f(u_5, u_6)$, this becomes:

$$\begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} Y \\ c \end{pmatrix} \uparrow_{.7} \begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix} u_5 \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}_{\beta}, \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}_{\beta} \vdash \begin{pmatrix} c \\ d \end{pmatrix} \uparrow_{\beta} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} u_6 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha}.$$

Evaluating the fuzzy unapplication gives $\beta = .7$:

$$\begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} Y \\ c \end{pmatrix} u_5 \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}_{.7}, \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix}_{.7} \vdash \begin{pmatrix} c \\ d \end{pmatrix} \uparrow_{.7} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} u_6 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha},$$

and by Axiom **Fuzzy Variable-Term**, we infer from this that $u_5 = X_2$, a new variable, which yields:

$$\begin{pmatrix} \{Y/X_1\} \\ \{Z/X_1\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} Y \\ c \end{pmatrix} X_2 \begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix}_{.7},$$
$$\begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix}_{.7} \vdash \begin{pmatrix} c \\ d \end{pmatrix} \uparrow_{.7} \begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix} u_6 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha},$$

and establishes the penultimate judgment. The last remaining judgment, after evaluating its fuzzy unapplication, since $c \sim_{.6} d$ and $.7 \wedge .6 = .6$, is:

$$\begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix}_{.6} \vdash \begin{pmatrix} c \\ d \end{pmatrix} u_6 \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_{\alpha},$$

for which Axiom **Fuzzy Variable-Term** allows us to infer that $u_6 = c$ and $\alpha = .6$:

$$\begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix}_{.6} \vdash \begin{pmatrix} c \\ d \end{pmatrix} c \begin{pmatrix} \{Y/X_1, Y/X_2\} \\ \{Z/X_1, c/X_2\} \end{pmatrix}_{.6}.$$

This validates the last judgment and completes the fuzzy generalization whereby $t = l(g(b, X_1), f(X_2, c))$ is the least fuzzy generalizer of $t_1 = h(g(b, Y), f(Y, c), V)$ and $t_2 = l(f(a, Z), g(c, d))$ at approximation degree .6, with:

- $\sigma_1 = \{Y/X_1, Y/X_2\}$ so that $t\sigma_1 = l(g(b, Y), f(Y, c)) \sim_{.6} t_1$; and,
- $\sigma_2 = \{Z/X_1, c/X_2\}$ so that $t\sigma_2 = l(g(b, Z), f(c, c)) \sim_{.6} t_2$.

5. Conclusion

We have developed a formal derivation of fuzzy lattice operations for the data structure known as first-order term. This is achieved by means of syntax-driven constraint normalization rules for both unification and generalization. These operations are then extended to enable arbitrary mismatch between similar terms whether functor-based, arity-based (number and order), or combinations.

We studied three lattice structures over $\mathcal{FOT}$s (one crisp and two fuzzy), gave declarative axioms and rules for their operations expressing the six corresponding dual lattice operations as constraint solving in these algebraic structures. Using the "✓" symbol to indicate our original contribution, this article's contents may be summarized, in each of the three lattice structures on $\mathcal{FOT}$s, as follows:

- **for conventional signatures** (no operator similarity besides identity):
  — we presented unification's declarative rules due to Herbrand, and Martelli and Montanari;
  ✓ we provided a declarative constraint-based version of generalization equivalent to the original procedural methods due to Reynolds and Plotkin;

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

- **for signatures with "weak" similarity** (all pairs of similar operators have the same number and order of arguments):
  — we presented "weak" fuzzy unification as constraint normalization using declarative rules due to Maria Sessa;
  ✓ we provided a "weak" fuzzy generalization as a constraint solving using a declarative specification for the dual operation of Sessa's "weak" unification;
- **for signatures with possibly misaligned similarity** (similar operators possibly with different number or order of arguments):
  ✓ we extended the above constraint-driven declarative "weak" fuzzy unification to $\mathcal{FOT}$s with possible different/mixed arities;
  ✓ we extended the above constraint-driven declarative "weak" fuzzy generalization of $\mathcal{FOT}$s with possible different/mixed arities.

This third pair of lattice operations on $\mathcal{FOT}$ modulo a similarity involving operators with misaligned or unordered arguments extends the previous pair of "weak" operations given argument mappings specified for so-similar operators. That is, a similar pair has a similarity degree as well as an injective argument-realigning mapping for each pair of operators in the signature. If unspecified, this mapping's default is the identity from the term with less arguments to the one of more arguments. In effect, this third lattice of $\mathcal{FOT}$s permits Fuzzy Logic Programming querying misaligned databases, or more generally Information Retrieval (using fuzzy unification) and Approximate Knowledge Acquisition (using fuzzy generalization) over heterogeneous but similar data models.

We have started an implementation in Java of the operational semantics derived from the axioms and rules that we presented and proved correct in this article which has allowed us to confirm our results on concrete examples [45].[28] This was eased by the fact that the fuzzy lattice operations do no require altering these conventional first-order structures.

As for future work, there are several avenues to explore. There are several other disciplines where this technology has potential for fuzzifying applications wherever $\mathcal{FOT}$s are used for their lattice-theoretic properties such as linguistics and learning. Finally, most promising is using this work's approach to more generic and more expressive knowledge structures for applications such as Information Retrieval (*e.g.*, in the line of [47]), or Data and Knowledge Base Management, Ontology Alignment, *etc.*, ... Finally, we are currently extending this work to similar functors where similarity may be restricted to possibly *partial* mappings of subterms between similar functors. As well, we are also developing the same formal constructions for fuzzy lattice operations over order-sorted feature ($\mathcal{OSF}$) graphs [48]. Encouraging initial results are being reported in [34].

Acknowledgements

The authors wish to thank the anonymous referees for their shrewd observations, and useful suggestions, which contributed to improve greatly the presentation and contents of the paper we had originally submitted.

Appendix

The following appendix sections provide formal material, terminology, and notation constituting background for the issues developed in this work. Appendix A recalls basic definitions and properties of $\mathcal{FOT}$ substitutions represented, as we do in this work, as finitely non-identical variable-to-terms mappings. Appendix B gives the basic formal set-theoretic characterization properties of relations as sets of pairs. Appendix C extends these to their fuzzy generalizations: Appendix C.1 to fuzzy relations; Appendix C.2 to fuzzy equivalence relations (called similarities); and Appendix C.3 to fuzzy partial orders.

Appendix A. First-order term substitutions

This section gives basic terminology and properties of $\mathcal{FOT}$ substitutions as defined in 2.2 where the set-theoretic definition of substitutions as finitely non-identical variable-to-term mappings is given as Expression (2).

[28] See also [46], a recent extension of the Bousi-Prolog system based on our similarity-based unification tolerating functors of differing arities.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Lemma 6. *Given two substitutions σ and θ in $\mathbf{SUBST}_{\mathcal{T}}$, the operation defined by Expression (2) always results in a substitution in $\mathbf{SUBST}_{\mathcal{T}}$.*

Proof. It must be verified that, given σ and θ two finitely non-identical mappings from $\mathcal{V}$ to $\mathcal{T}$, the notation $\sigma\theta$ defined in set-theoretic terms from the set structure of σ and θ by Expression (2) always results in a finitely non-identical mapping from $\mathcal{V}$ to $\mathcal{T}$. This is an elementary exercise from the very set-theoretic definition of substitution composition given as Expression (2). □

Lemma 7. *For any term t in $\mathcal{T}$ and any substitutions σ and θ in $\mathbf{SUBST}_{\mathcal{T}}$, the expression $\sigma\theta$ defined by Expression (2) is a substitution that has the same effect as first applying σ to t, and then applying θ to the result; that is, $\forall t \in \mathcal{T}, \forall \sigma \in \mathbf{SUBST}_{\mathcal{T}}, \forall \theta \in \mathbf{SUBST}_{\mathcal{T}},\ t(\sigma\theta) = (t\sigma)\theta$.*

Proof. Expression (2) consists of two parts of a (disjoint) set union. The first part of this union consists in the set of pairs t/X in σ transformed into the set of pairs $t\theta/X$ for each pair t/X in σ. This has for effect to "capture" any potential variables in $\mathbf{var}(t\sigma) \cap \mathbf{dom}(\theta)$ by mapping directly to $t\sigma\theta$ any variable mapped to t by σ. This corresponds to precomputing the necessary "shortcut" of instantiating X directly into to $t\sigma\theta$ for all such concerned variables in $\mathbf{dom}(\theta)$. Note that since this may possibly introduce identical pairs X/X, which must then be eliminated.

The second part of the union in Expression (2) simply completes the resulting substitution with pairs t/Y in θ concerning those variables Y which are not affected by σ (*i.e.*, all $Y \in \mathbf{dom}(\theta)$ such that $Y \notin \mathbf{dom}(\sigma)$). Indeed, these variables are taken care of in the first part in the terms mapping the variables in $\mathbf{dom}(X)$ by further instantiating by θ as need be.

These two cases clearly cover the only possibilities for variable mapping by σ and θ, and by construction in each case, this results in a finite set of term/variable pairs, thus completely specified by Expression (2) on all $\mathcal{V}$, when applied to any term t, has the same effect of first applying σ to t and then applying θ to the result. □

Corollary 5. *Substitution composition as defined by Expression (2) is an associative operation; i.e., for all σ, θ, and δ in $\mathbf{SUBST}_{\mathcal{T}}$, $\sigma(\theta\delta) = (\sigma\theta)\delta$.*

Proof. Let t be any term in $\mathcal{T}$, and σ, θ, and δ be three substitutions in $\mathbf{SUBST}_{\mathcal{T}}$. Applying Lemma 7 successively, we have $t(\sigma(\theta\delta)) = (t\sigma)(\theta\delta) = ((t\sigma)\theta)\delta = (t(\sigma\theta))\delta = t((\sigma\theta)\delta)$. Since both sides applied to any term are equal, this means that $\sigma(\theta\delta) = (\sigma\theta)\delta$. □

Note that, as a set of term/variable pairs, the substitution which is the identity everywhere on $\mathcal{V}$ is the empty set of pairs—which is why it is called the empty substitution and denoted as the empty set $\emptyset$. It is easy to verify that this empty substitution is also the unique identity element on $\mathbf{SUBST}_{\mathcal{T}}$. Namely, for all substitution $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$, $\sigma\emptyset = \emptyset\sigma = \sigma$ and if $\sigma\theta = \theta\sigma = \sigma$ for some $\theta \in \mathbf{SUBST}_{\mathcal{T}}$, then $\theta = \emptyset$. Therefore, $\mathbf{SUBST}_{\mathcal{T}}$ with composition and $\emptyset$ is a monoid. Note finally that substitution composition is not commutative since in general $\sigma\theta \neq \theta\sigma$.[29] Therefore, the set $\mathbf{SUBST}_{\mathcal{T}}$ with substitution composition is a non-commutative monoid.

Like all monoids, the set $\mathbf{SUBST}_{\mathcal{T}}$ of substitutions inherits a relation $\preceq$ defined as follows.

Definition 13. $\sigma \preceq \theta$ *iff* $\exists \delta \in \mathbf{SUBST}_{\mathcal{T}}$ *s.t.* $\sigma = \theta\delta$.

The expression "$\sigma \preceq \theta$" is read "σ *refines* θ" or "θ *is more general than* σ."

Lemma 8. *The relation $\preceq$ is a preorder on the set of first-order term substitutions $\mathbf{SUBST}_{\mathcal{T}}$.*

Proof. We must show that $\preceq$ is reflexive and transitive. **Reflexivity:** For any $\sigma \in \mathbf{SUBST}_{\mathcal{T}}$, there exists $\delta = \emptyset$ such that $\sigma = \sigma\delta$, which means by definition of $\preceq$ that $\sigma \preceq \sigma$. **Transitivity:** Assume $\sigma_1 \preceq \sigma_2$ and $\sigma_2 \preceq \sigma_3$; this means that there exist δ_1 and δ_2 such that $\sigma_1 = \sigma_2\delta_1$ and $\sigma_2 = \sigma_3\delta_2$. Replacing σ_2 by its value in the expression of σ_1, it comes as a result that $\sigma_1 = \sigma_3\delta_2\delta_1$. And so, there exists $\delta_3 = \delta_2\delta_1$ such that $\sigma_1 = \sigma_3\delta_3$; which means that $\sigma_1 \preceq \sigma_3$. □

[29] Take for example $\sigma = \{a/X\}$ and $\theta = \{b/X\}$, for which $\sigma\theta = \{a/X\}$ and $\theta\sigma = \{b/X\}$.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

Note that $\preceq$ is not an order relation because it is not anti-symmetric. Indeed, if we have both $\sigma \preceq \theta$ and $\theta \preceq \sigma$, this does not necessarily imply that $\sigma = \theta$. However, this defines an equivalence relation on substitutions.

Lemma 9. *The relation* $\simeq \stackrel{def}{=} \preceq \cap \preceq^{-1}$ *is an equivalence on the set of substitutions* $\textbf{SUBST}_{\mathcal{T}}$.

Proof. Let us verify that $\simeq$ has three properties of an equivalence. **Reflexivity:** Clearly, for any $\sigma \in \textbf{SUBST}_{\mathcal{T}}$, $\sigma \simeq \sigma$ since this is equivalent to $\sigma \preceq \sigma$ and $\sigma \preceq \sigma$, which is always true since $\preceq$ is reflexive because it is a preorder. **Symmetry:** Also, for any $\sigma \in \textbf{SUBST}_{\mathcal{T}}$ and $\theta \in \textbf{SUBST}_{\mathcal{T}}$, if $\sigma \simeq \theta$, this is equivalent by definition to $\sigma \preceq \theta$ and $\theta \preceq \sigma$; which is also equivalent to $\theta \simeq \sigma$. Therefore, $\simeq$ is symmetric. **Transitivity:** Let us now assume that (1) $\sigma \simeq \theta$ and (2) $\theta \simeq \delta$. This implies in particular, by definition of $\simeq$ and $\simeq$: (1) ($\sigma \preceq \theta$ and $\theta \preceq \delta$), which by transitivity of $\preceq$ implies $\sigma \preceq \delta$; and (2) ($\delta \preceq \theta$ and $\theta \preceq \sigma$); which by transitivity of $\preceq$ implies $\delta \preceq \sigma$. Hence, we have both $\sigma \preceq \delta$ and $\delta \preceq \sigma$, which is equivalent to $\sigma \simeq \delta$. Therefore, $\simeq$ is transitive. □

Definition 14. *A variable renaming* ρ *is a substitution in* $\textbf{SUBST}_{\mathcal{T}} \cap (\mathcal{V} \to \mathcal{V})$ *that is injective. That is,*

- $\rho = \{X'_i/X_i\}_{i=1}^n$ *with* $X_i \in \mathcal{V}$ *and* $X'_i \in \mathcal{V}$*; and,*
- *if* $X_i \neq X_j$ *then* $X'_i \neq X'_j$*, for any* $i, j = 1, \ldots, n$ *such that* $i \neq j$.

Corollary 6. *If both* $\sigma \preceq \theta$ *and* $\theta \preceq \sigma$*, this entails that* σ *and* θ *are equal up to a renaming of their variables. Namely,* $\exists\, \rho : \mathcal{V} \to \mathcal{V}$ *bijective such that* $\theta = \rho\sigma$ *and* $\sigma = \rho^{-1}\theta$.

Proof. If $\sigma \preceq \theta$ and $\theta \preceq \sigma$ then, by definition, there exist two substitutions ρ and ρ' such that $\sigma = \theta\rho$ and $\theta = \sigma\rho'$. In other words:

$$\begin{cases} \sigma &= \sigma\,\rho\,\rho', \\ \theta &= \theta\,\rho'\rho\,; \end{cases} \quad \text{which is equivalent to:} \quad \begin{cases} \rho\,\rho' &= \emptyset, \\ \rho'\rho &= \emptyset; \end{cases} \quad \text{and therefore to:} \quad \begin{cases} \rho &= \rho'^{-1}, \\ \rho' &= \rho^{-1}. \end{cases}$$

Note also that since ρ and ρ' are mutual inverses on $\mathcal{V}$, it must be that ρ and ρ' are injective. This follows from the axiom of functionality for ρ and ρ', which states that for every pair of variables X and X' in $\mathcal{V}$, if $X = X'$ then necessarily $X\rho = X'\rho$ and $X\rho' = X'\rho'$. But since ρ are mutual inverses on $\mathcal{V}$, this means that whenever $Y\rho' = Y'\rho'$ for any pair of variables Y and Y' in $\mathcal{V}$, then necessarily $Y\rho'\rho = Y'\rho'\rho$; *i.e.*, $Y\emptyset = Y'\emptyset$, and thus $Y = Y'$, which means that ρ' must be injective. The same reasoning in the other direction will entail that ρ must be injective as well. Note finally that ρ is also surjective on $\mathcal{V}$, since any variable $X \in \mathcal{V}$ is such that $X\rho'\rho = X$, therefore there exists $Y = X\rho'$ such that $Y\rho = X$. The same applies to ρ' in the other direction. Therefore, ρ and ρ' are bijective inverses. □

Appendix B. Crisp relations

In (crisp) Set Theory, a binary relation r on a set S is a subset of $S \times S$. We say that r is:

- *reflexive* iff:

$$\mathbb{1}_{S\times S} \subseteq r \tag{B.1}$$

  where $\mathbb{1}_S$ is the *identity* relation on S defined as: $\mathbb{1}_{S\times S} \stackrel{\text{def}}{=} \{\langle x, x\rangle \mid x \in S\}$;
- *symmetric* iff:

$$r = r^{-1} \tag{B.2}$$

  where r^{-1} is the *inverse* relation of r defined as: $r^{-1} \stackrel{\text{def}}{=} \{\langle y, x\rangle \in S \times S \mid \langle x, y\rangle \in r\}$;
- *antisymmetric* iff:

$$r \cap r^{-1} \subseteq \mathbb{1}_{S\times S} \tag{B.3}$$

  where $r \cap r'$ is the intersection of r and r'; *viz.*, the relation on S defined as: $r \cap r' \stackrel{\text{def}}{=} \{\langle x, y\rangle \in S \times S \mid \langle x, y\rangle \in r \textbf{ and } \langle x, y\rangle \in r'\}$;

- *transitive* iff:

$$r \circ r \subseteq r \tag{B.4}$$

where $r \circ r'$ is the composition of r and r'; *viz.*, the relation on S defined as: $r \circ r' \stackrel{\text{def}}{=} \{ \langle x, y\rangle \in S \times S \mid \langle x, z\rangle \in r$ **and** $\langle z, y\rangle \in r'$ for some $z \in S \}$.

Definition 15 *(PREORDER). A relation r on a set S is a preorder on S iff it is reflexive and transitive; i.e., iff r verifies conditions* (B.1) *and* (B.4).

Definition 16 *(EQUIVALENCE). A symmetric preorder r on a set S is called an equivalence on S; that is, r verifies conditions* (B.1), (B.2), *and* (B.4).

Such an equivalence relation $\equiv$ on a set S defines a partition of this set; namely, a collection of non-empty subsets S_i, $1 \le i \le \mathbf{I}_{\equiv} \in \mathbb{N}$, of S (the equivalence classes) such that:

$$1 \le i \ne j \le \mathbf{I}_{\equiv} \implies S_i \cap S_j = \emptyset \tag{B.5}$$

and:

$$S = \bigcup_{i \le \mathbf{I}_{\equiv}} S_i \tag{B.6}$$

where $\mathbf{I}_{\equiv}$, the *index* of $\equiv$, is the number of equivalence classes of $\equiv$ forming the partition of S. The equivalence class of an element of $x \in S$ is denoted $[x]^{\equiv}$ and is defined as:

$$[x]^{\equiv} \stackrel{\text{def}}{=} \{ y \in S \mid x \equiv y \}. \tag{B.7}$$

Definition 17 *(PARTIAL ORDER). A relation r on a set S is a partial order on S iff it is an antisymmetric preorder on S; i.e., iff r verifies conditions* (B.1), (B.3), *and* (B.4).

Appendix C. Fuzzy set algebra

In this section, we recall some essential terminology and notation on Fuzzy Set algebra used in this document. The symbol we shall use for fuzzy conjunction (a canonical model of which is the family of T-norms[30]) is $\wedge$ (resp., $\vee$ for its fuzzy dual operation). This is generally interpreted as **min** (resp., **max**); *e.g.*, in Zadeh's seminal paper [49]. But other T-norms (T-conorms) can be considered depending on the desired effect. For example, here below the definitions of three popular definitions for $\wedge$ and $\vee$ on [0, 1], which are used in practice [39], [50]:

- "Gödel" fuzzy operators:

$$\begin{cases} \alpha_1 \wedge_G \alpha_2 \stackrel{\text{def}}{=} \mathbf{min}(\alpha_1, \alpha_2) \\ \alpha_1 \vee_G \alpha_2 \stackrel{\text{def}}{=} \mathbf{max}(\alpha_1, \alpha_2) \end{cases} \tag{C.1}$$

- "Product" (or "probabilistic") fuzzy operators:

$$\begin{cases} \alpha_1 \wedge_P \alpha_2 \stackrel{\text{def}}{=} \alpha_1\alpha_2 \\ \alpha_1 \vee_P \alpha_2 \stackrel{\text{def}}{=} \alpha_1 + \alpha_2 - \alpha_1\alpha_2 \end{cases} \tag{C.2}$$

- "Łukasiewicz" fuzzy operators:

$$\begin{cases} \alpha_1 \wedge_L \alpha_2 \stackrel{\text{def}}{=} \mathbf{max}(0, \alpha_1 + \alpha_2 - 1) \\ \alpha_1 \vee_L \alpha_2 \stackrel{\text{def}}{=} \mathbf{min}(\alpha_1 + \alpha_2, 1) \end{cases} \tag{C.3}$$

30 See https://en.wikipedia.org/wiki/T-norm.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

The choice of one of the previous definitions affects the semantics of the fuzzy conjunction (fuzzy disjunction) operator. For example, contrary to the "Gödel" fuzzy conjunction $\wedge_G$ that imposes the value of one over the other of two values, the "Product" version $\wedge_P$ is less "drastic" and will take a more balanced consideration of the values of both arguments. There are a few other fuzzy operators that have been given specific denominations that correspond to particular situations.[31] But one may design their adequate $\wedge$ operator (or $\vee$ operator since one can be derived from the other by duality).[32]

However, in all the actual numerical examples provided in this document for illustration, we use **min** (resp., **max**) as fuzzy conjunction (resp., fuzzy disjunction).

C.1. Fuzzy relation

Let us now fuzzify the conventional set theoretic definitions recalled in Appendix B. It is a straightforward homomorphic extension of the conventional view of (crisp) sets as $\{0, 1\}$-valued functions to $[0, 1]$-valued functions. Indeed, the former are just a particular case of the more general (fuzzy) sets seen as $[0, 1]$-valued characteristic functions.[33] That is, all the fuzzy notions are obtained as straightforward extensions of their crisp counterparts through a Boolean lattice homomorphism. The advantage of the fuzzy extension over conventional sets is that, being structurally richer, it is more expressive. It is a homomorphic extension insofar as all the formal algebraic properties of fuzzy sets and fuzzy-set connectives reduce to their conventional crisp versions when reducing the membership value $\phi(x)$ of every element $\phi(x)/x$ of a fuzzy set ϕ into a crisp value in $\{0, 1\}$ for values which, when compared to a given value α in $[0, 1]$, are either strictly less (assimilated to 0), or greater or equal (assimilated to 1). Informally, this is the crisp set of elements with "*at least*" α as membership value. This is called a fuzzy set's "α-*cut*" ϕ_α such that $\phi_\alpha(x) \stackrel{\text{def}}{=} 0$ whenever $\phi(x) < \alpha$ and $\phi_\alpha(x) \stackrel{\text{def}}{=} 1$ whenever $\phi(x) \geq \alpha$, for any *threshold value* α in $[0, 1]$.

Definition 18 (Fuzzy Relation). *A fuzzy relation on a set S is a fuzzy subset on $S \times S$.*

The following properties generalize those of crisp binary relations seen in Appendix B. Like in the crisp case, we will look closer at essentially two kinds of fuzzy binary relations: fuzzy orders and fuzzy equivalences.[34]

Recall that a fuzzy set ϕ on a set S is a function $\phi : S \to [0, 1]$. Let $\rho : S \times S \to [0, 1]$ be a fuzzy relation on S. We say that ρ is:

- *reflexive* iff:

$$\mathbb{1}_{S\times S} \leq \rho \tag{C.4}$$

  where $\mathbb{1}_{S\times S}$ is the *fuzzy identity* relation on S defined as: $\mathbb{1}_{S\times S}(x, y) = 1$ if $x = y$ and 0 if $x \neq y$, for all x and y in S; and $\leq$ is *fuzzy set inclusion* defined as: $\rho \leq \rho'$ iff $\rho(x, y) \leq \rho'(x, y)$, for all x and y in S;
- *symmetric* iff:

$$\rho = \rho^{-1} \tag{C.5}$$

  where ρ^{-1} is the *fuzzy inverse* of ρ; *viz.*, the fuzzy relation on S defined as: $\rho^{-1}(x, y) \stackrel{\text{def}}{=} \rho(y, x)$, for all x and y in S;
- *antisymmetric* iff:

$$\rho \wedge \rho^{-1} \leq \mathbb{1}_{S\times S} \tag{C.6}$$

  where the *fuzzy meet* $\rho \wedge \rho'$ is the fuzzy relation on S defined as: $\big(\rho \wedge \rho'\big)(x, y) \stackrel{\text{def}}{=} \rho(x, y) \wedge \rho'(x, y)$, for all x and y in S;

[31] See, *e.g.*: http://www.nicodubois.com/bois5.2.htm.

[32] Designing specific fuzzy norms can be done visually in 3D using publicly available tools such as, *e.g.*, http://www.math.uri.edu/~bkaskosz/flashmo/graph3d2/.

[33] Such a fuzzy characteristic function is called a "*membership function*" in the literature following Zadeh's original terminology [49].

[34] See [51] for even finer and more expressive kinds of useful fuzzy relations that can be defined algebraically.

- *transitive* iff:

$$\rho \circ \rho \;\leq\; \rho \tag{C.7}$$

where the *fuzzy composition* $\rho \circ \rho'$ is the fuzzy relation on S defined as: $\big(\rho \circ \rho'\big)(x, y) \stackrel{\text{def}}{=} \bigvee_{z \in S}\big(\rho(x, z) \wedge \rho'(z, y)\big)$, for all x and y in S.

Definition 19 (Fuzzy Preorder). *A fuzzy relation ρ on a set S is a fuzzy preorder on S iff it is reflexive and transitive; i.e., iff r verifies conditions* (C.4) *and* (C.7).

C.2. Similarity

Definition 20 (Fuzzy Equivalence). *A fuzzy equivalence ρ on a set S is a fuzzy relation on S which is a symmetric fuzzy preorder on S—that is, ρ verifies conditions* (C.4), (C.5), *and* (C.7).

A fuzzy equivalence relation is also called "*similarity*" relation in the literature [50]. For this reason, we speak of "*similarity degree*" to denote the membership value of a pair so related.

A similarity relation $\sim$ on a set S is a fuzzy equivalence relation on S; *i.e.*, a fuzzy set of pairs of $S \times S$. When S is a finite discrete set, say indexed over $\{1, \ldots, n\}$, since a similarity relation $\sim$ on S is a fuzzy subset of $S \times S$, the three conditions of an equivalence can be visualized on a square $n \times n$ matrix $\sim \in \{1, \ldots, n\}^2 \rightarrow [0, 1]$ as follows. For all $i, j, k = 1, \ldots, n$:

- *reflexivity*: $i \sim i \;=\; 1$ (*i.e.*, entries on the diagonal are equal to 1);
- *symmetry*: $i \sim j \;=\; j \sim i$ (*i.e.*, all symmetric entries on either side of the diagonal are equal);
- *transitivity*: $i \sim k \wedge k \sim j \;\leq\; i \sim j$, for any $k \in \{1, \ldots, n\}$ (*i.e.*, going via an intermediate element will always result in a smaller or equal similarity degree than going directly).[35]

Given a similarity relation $\sim$ on a set S, the subset of $[0, 1]$ denoted $\textbf{Degrees}^{\sim}$ and defined as $\textbf{Degrees}^{\sim} \stackrel{\text{def}}{=} \{\alpha \in [0, 1] \mid x \sim_\alpha y, \text{ for some } x, y \in S\}$ is called the "*similarity degree set*" of $\sim$. A similarity degree $\alpha \in \textbf{Degrees}^{\sim}$ can thus be used as an approximation threshold, and a similarity can be rendered a crisp equivalence on S by keeping only pairs in $\sim$ with similarity degree greater than or equal to α (*i.e.*, the α-cut of the similarity).

The similarity class $[x]^{\sim}_{\alpha}$ of an element $x \in S$ at an approximation threshold α in $[0, 1]$ given a similarity $\sim$ on S is defined as:

$$[x]^{\sim}_{\alpha} \stackrel{\text{def}}{=} \{\, y \in S \mid x \sim_\beta y, \text{ for some } \beta \in [\alpha, 1]\,\}.$$

Thus, for lower values of α, more similarity degrees between pairs of distinct elements of S are considered than those related in α-cuts of $\sim$ for greater values of α.

C.3. Fuzzy partial order

Definition 21 (Fuzzy Partial Order). *A fuzzy relation ρ on a set S is a fuzzy partial order on S iff it is an antisymmetric fuzzy preorder; i.e., iff ρ verifies conditions* (C.4), (C.6), *and* (C.7).

For a fuzzy partial order, as in the case of a fuzzy equivalence relation, when S is a finite discrete set $\{x_1, \ldots, x_n\}$, the three conditions of the above definition can be visualized on a square $n \times n$ matrix $\preceq in\, [0, 1]^2$ as follows:

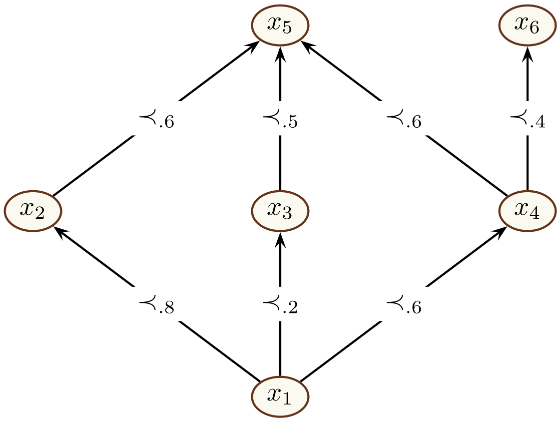
- reflexivity and transitivity (just as for a similarity matrix);

[35] Here and elsewhere in this article, we shall use $\wedge/\vee$ for fuzzy conjunction/disjunction in generic formulas. We prefer using these more general symbols in our formalization since which specific fuzzy operations are used is irrelevant. However, we shall use **min**/**max** in all the illustrative examples we give that use actual numbers.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

- antisymmetry: the matrix must be triangular (up to reordering of columns and lines); this is because $\preceq_{ij} > 0$ implies $\preceq_{ji} = 0$, for all $i, j = 1, \ldots, n$ (*i.e.*, all symmetric entries on either side of the diagonal may not be both non-zero).

For example, the fuzzy binary relation $\preceq$ on the 6-element set $\{x_1, \ldots, x_6\}$ defined as the fuzzy **min/max** reflexive-transitive closure of the following weighted acyclic graph[36]:



corresponds to the following fuzzy matrix:

$$\preceq \ \stackrel{\text{def}}{=} \begin{bmatrix} 1 & 0.8 & 0.2 & 0.6 & 0.6 & 0.4 \\ 0 & 1 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 & 0.6 & 0.4 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.8}$$

upon which these conditions can be verified—which means that the fuzzy relation $\preceq$ so defined is a fuzzy partial order on the set $\{x_1, x_2, x_3, x_4, x_5, x_6\}$.

Note that, just as in the crisp case, any fuzzy preorder $\preceq$ on a set S (*i.e.*, a fuzzy relation on S that is reflexive and transitive) always implicitly defines the following fuzzy relations:

- a similarity $\sim$ on S defined, for any $\alpha \in [0, 1]$, as:

$$\sim_\alpha \ \stackrel{\text{def}}{=} \ \preceq_\alpha \wedge \succeq_\alpha \tag{C.9}$$

  where $\succeq_\alpha$ is the fuzzy relation defined as: $\succeq_\alpha \ \stackrel{\text{def}}{=} \ \preceq_\alpha^{-1}$;
- a fuzzy partial order $\preceq$, a fuzzy set of partial orders $\preceq_\alpha$ on each partition $\Pi_\alpha^\sim$ of S generated by $\sim$ in the fuzzy partition $\Pi^\sim \stackrel{\text{def}}{=} \{ \Pi_\theta^\sim \mid \theta \in \textbf{Degrees}^\sim \}$, such that:

$$[x]_\alpha^\sim \ \preceq_\alpha \ [y]_\alpha^\sim \ \text{ iff } \ x \sim_\alpha x' \text{ and } x' \preceq_\alpha y' \text{ and } y' \sim_\alpha y \tag{C.10}$$

  for some $x' \in [x]_\alpha^\sim$ and some $y' \in [y]_\alpha^\sim$.

References

[1] Jacques Herbrand, Recherches sur la théorie de la démonstration, PhD thesis, Faculté des sciences de l'université de Paris, Paris (France), 1930, online: http://archive.numdam.org/article/THESE_1930__110__1_0.pdf, English translation in [10].
[2] Alberto Martelli, Ugo Montanari, An efficient unification algorithm, ACM Trans. Program. Lang. Syst. 4 (2) (April 1982) 258–282, online: http://moscova.inria.fr/~levy/courses/X/IF/03/pi/levy2/martelli-montanari.pdf.

[36] This example is from [52].

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

[3] Mathieu Serrurier, Thomas Sudkamp, Didier Dubois, Henri Prade, Fuzzy inductive logic programming: learning fuzzy rules with their implication, in: Proceedings of the 14th IEEE International Conference on Fuzzy Systems (FUZZ'05), Reno, NV (USA), IEEE, May 25–25 2005, pp. 613–618.

[4] Yann-Michaël De Hauwere, Declarative programming: inductive logic programming. Tutorial lecture slides, online: http://ai.vub.ac.be/~ydehauwe/decl_prog/7_InductiveLogicProgramming.pdf, 2015.

[5] John C. Reynolds, Transformational systems and the algebraic nature of atomic formulas, in: Bernard Metzer, Donald Michie (Eds.), Machine Intelligence 5, Edinburgh University Press, Edinburgh, Scotland (UK), 1970, pp. 135–151, chapter 7, online: http://www.cs.cmu.edu/afs/cs/user/jcr/ftp/transysalg.pdf.

[6] Gordon D. Plotkin, A note on inductive generalization, in: Bernard Metzer, Donald Michie (Eds.), Machine Intelligence 5, Edinburgh University Press, Edinburgh, Scotland (UK), 1970, pp. 154–163, chapter 8, online: http://homepages.inf.ed.ac.uk/gdp/publications/MI5_note_ind_gen.pdf.

[7] John Alan Robinson, A machine-oriented logic based on the resolution principle, J. ACM 12 (January 1965) 23–41, online: https://www.academia.edu/[...]/A_[...]_Logic_Based_on_the_Resolution_Principle.

[8] Dag Prawitz, An improved proof procedure, Theoria 26 (August 1960) 102–139, online: http://onlinelibrary.wiley.com/doi/10.1111/j.1755-2567.1960.tb00558.x/full.

[9] Jean van Heijenoort, From Frege to Gödel—A Source Book in Mathematical Logic, 1879–1931, Harvard University Press, Cambridge, MA (USA), 1967.

[10] Jacques Herbrand, Logical Writings, Harvard University Press, Cambridge MA, 1971, Edited by Warren D. Goldfarb.

[11] Gérard Huet, Résolution d'Équations dans des Langages d'Ordre $1, 2, \ldots, \omega$, Thèse d'état, Université de Paris VII, Paris (France), 1976, online: https://www.researchgate.net/[...]_languages_d'ordre_1_2_omega.

[12] Maria I. Sessa, Approximate reasoning by similarity-based SLD resolution, Theor. Comput. Sci. 275 (2002) 389–426, online: http://www.sciencedirect.com/science/article/pii/S0304397501001888.

[13] Joseph A. Goguen, James W. Thatcher, Eric G. Wagner, Jesse B. Wright, Initial algebra semantics and continuous algebras, J. Assoc. Comput. Mach. 24 (1) (January 1977) 68–95, online: https://dl.acm.org/citation.cfm?doid=321992.321997.

[14] Joseph Goguen, What is unification?, in: Aït-Kaci Hassan, Maurice Nivat (Eds.), Resolution of Equations in Algebraic Structures—Algebraic Techniques, Academic Press, 1989, pp. 217–261, chapter 6, online: http://www-cse.ucsd.edu/~goguen/pps/subs.ps.

[15] Bernd Mahr, Johann A. Makowsky, Characterizing specification languages which admit initial semantics, Theor. Comput. Sci. 31 (1–2) (1984) 49–59, online: https://www.sciencedirect.com/science/article/pii/0304397584901257.

[16] María Alpuente, Santiago Escobar, Javier Espert, José Meseguer, A modular order-sorted equational generalization algorithm, Inf. Comput. 235 (2014) 98–136, online: https://www.ideals.illinois.edu/bitstream/handle/2142/25871/journal.pdf.

[17] María Alpuente, Santiago Escobar, José Meseguer, Pedro Ojeda, Order-sorted generalization, Electron. Notes Theor. Comput. Sci. 246 (2009) 27–38, online: https://pdfs.semanticscholar.org/049a/6d3eed8d5cc3cd02a4aa0ef5283a0a368952.pdf.

[18] M.J. Fay, First order unification in an equational theory, in: W.H. Joyner (Ed.), Proceedings of the 4th Workshop on Automated Deduction (Austin, TX), Academic Press, London, UK, 1979, pp. 161–167.

[19] Santiago Escobar, José Meseguer, Prasanna Thati, Narrowing and rewriting logic: from foundations to applications, Electron. Notes Theor. Comput. Sci. 177 (2007) 5–33, online: https://www.sciencedirect.com/science/article/pii/S1571066107002113.

[20] Donald E. Knuth, Peter B. Bendix, Simple word problems in universal algebras, in: J. Leech (Ed.), Computational Problems in Abstract Algebra, Pergamon Press, Oxford, UK, 1970, pp. 263–297, reprinted in: Automatic Reasoning, vol. 2, Springer-Verlag, 1983, pp. 342–376.

[21] Nachum Dershowitz, Jean-Pierre Jouannaud, Rewrite systems, in: Jan van Leeuwen (Ed.), Handbook of Theoretical Computer Science—Volume B: Formal Methods and Semantics, chapter 6, North-Holland, Amsterdam, The Netherlands, 1990, pp. 243–320, online: http://www.cs.tau.ac.il/~nachum/papers/survey-draft.pdf.

[22] Francesca Arcelli, Ferrante Formato, Giangiacomo Gerla, Extending unification through similarity relations, BUSEFAL (1997) 3–12, online: http://int.polytech.univ-smb.fr/[...]/listic/busefal/Papers/70.zip/70_01.pdf.

[23] Francesca Arcelli, Ferrante Formato, A fuzzy logic programming language, in: Luca Console, Evelina Lamma, Paola Mello, Michela Milano (Eds.), Programmazione Logica e Prolog, UTET Università, 1997, pp. 319–332, online: http://www.programmazionelogica.it/wp-content/uploads/1997/06/319_Fontana1.pdf.

[24] David Gilbert, Michael Schroeder, FURY: fuzzy unification and resolution based on edit distance, in: Nikolaos G. Bourbakis (Ed.), Proceedings of the 1st IEEE International Symposium on Bioinformatics and Biomedical Engineering (BIBE 2000), Arlington, VA (USA), November 8–10, 2000, IEEE Computer Society, pp. 330–336, online: https://pdfs.semanticscholar.org/ce6c/380e804ac52124fbf72dcf57338aa660c307.pdf.

[25] Michael Schroeder, Ralf Schweimeier, Arguments and misunderstandings: fuzzy unification for negotiating agents, Electron. Notes Theor. Comput. Sci. 70 (5) (October 2002) 1–19, online: http://www.sciencedirect.com/science/article/pii/S1571066104805851.

[26] Ralf Schweimeier, Michael Schroeder, Fuzzy unification and argumentation for well-founded semantics, in: Peter Van Emde Boas, Július Štuller Jaroslav Pokorný, Mária Bieliková (Eds.), Proceedings of the 30th Conference on Current Trends in Theory and Practice of Computer Science, Měřín, Czech Republic, 24–30 January 2004, in: Lecture Notes in Computer Science, vol. 2932, Springer, pp. 102–121, online: https://pdfs.semanticscholar.org/a623/f8302765ee433fae2f0291a6383f8a1818c5.pdf.

[27] Michael Schroeder, Re: your work on FURY, private communication (email), October 20, 2017, online: http://hassan-ait-kaci.net/eml/Re-Your-work-on-FURY.eml.

[28] Kutsia Temur, Pattern unification with sequence variables and flexible arity symbols, Electron. Notes Theor. Comput. Sci. 66 (5) (2002) 52–69, online: https://www.risc.jku.at/people/tkutsia/papers/Kutsia-UNCL02.pdf.

[29] Temur Kutsia, Jordi Levy, Mateu Villaret, Anti-unification for unranked terms and hedges, J. Autom. Reason. 52 (2) (2014) 155–190, online: https://link.springer.com/content/pdf/10.1007/s10817-013-9285-6.pdf.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019

[30] Aït-Kaci Hassan, Gabriella Pasi, Lattice operations on terms over similar signatures, in: Fabio Fioravanti, John Gallagher (Eds.), Proceedings of the 27th International Symposium on Logic-Based Program Synthesis and Transformation (LOPSTR 2017), Namur, Belgium, October 20-12 2017, online: http://www.hassan-ait-kaci.net/pdf/fuzfotlats-lopstr2017.pdf.

[31] Aït-Kaci Hassan, Andreas Podelski, Towards a meaning of LIFE, J. Log. Program. 16 (3–4) (1993) 195–234, online: http://hassan-ait-kaci.net/pdf/meaningoflife.pdf.

[32] Aït-Kaci Hassan, Yutaka Sasaki, An axiomatic approach to feature term generalization, in: Peter Flach, Luc de Raedt (Eds.), Proceedings of the 12th European Conference on Machine Learning (ECML'01), Berlin Heidelberg, in: LNCS, vol. 2167, Springer-Verlag, September 2001, pp. 1–12, online: http://www.hassan-ait-kaci.net/pdf/ecml01.pdf.

[33] Simon Lacoste-Julien, Konstantina Palla, Alex Davies, Gjergji Kasneci, Thore Graepel, Zoubin Ghahramani, SiGMa: simple greedy matching for aligning large knowledge bases, in: Inderjit S. Dhillon, Yehuda Koren, Rayid Ghani, Ted E. Senator, Paul Bradley, Rajesh Parekh, Jingrui He, Robert L. Grossman, Ramasamy Uthurusamy (Eds.), Proceedings of the 19th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD 2013—Chicago, IL, USA), New York, NY (USA), August 11–14, 2013, Association for Computing Machinery, ACM, pp. 572–580, online: http://snap.stanford.edu/social2012/papers/lacostejulien-palla-etal.pdf, see also online: https://arxiv.org/pdf/1207.4525.pdf.

[34] Aït-Kaci Hassan, Gabriella Pasi, Fuzzy Lattice-Theoretic Operations over Data and Knowledge Structures, Technical Report, HAK Language Technologies, 2017, in preparation, online: http://hassan-ait-kaci.net/pdf/fuzlatopdks.pdf.

[35] Gordon D. Plotkin, Lattice Theoretic Properties of Subsumption, Technical Memo MIP-R-77, Department of Machine Intelligence and Perception, University of Edinburgh, Edinburgh, Scotland (UK), June 1970.

[36] Maarten H. van Emden, John W. Lloyd, A logical reconstruction of Prolog II, J. Log. Program. 2 (1984) 143–149, online: https://www.sciencedirect.com/science/article/pii/0743106684900013.

[37] Joxan Jaffar, Efficient unification over infinite terms, New Gener. Comput. 2 (3) (September 1984) 207–219, online: https://link.springer.com/article/10.1007/BF03037057.

[38] Alain Colmerauer, Prolog and infinite trees, in: Keith L. Clark, Sten Åke Tärnlund (Eds.), Logic Programming, Academic Press, 1982, pp. 231–251.

[39] Didier Dubois, Henri Prade, Fuzzy sets and systems: theory and applications, in: William F. Ames (Ed.), Mathematics in Science and Engineering, vol. 144, Georgia Institute of Technology, Academic Press, 180 pp., online: ftp://[...]/Fuzzy%20Sets%20And%20Systems%20Theory%20And%20Applications.pdf.

[40] Pascual Julián Iranzo, Clemente Rubio Manzano, A sound and complete semantics for a similarity-based logic programming language, Fuzzy Sets Syst. 317 (2017) 1–26.

[41] Krzysztof R. Apt, Logic programming, in: Jan van Leeuwen (Ed.), Handbook of Theoretical Computer Science, Elsevier, 1990, pp. 492–574.

[42] Aït-Kaci Hassan, An algorithm for finding a minimal recursive path ordering, RAIRO Inform. Théor. 19 (4) (1985) 359–382, online: http://hassan-ait-kaci.net/pdf/rairo85.pdf.

[43] Kevin Wayne, Union-find, Tutorial lecture slides based on book "Algorithm Design" by Jon Kleinberg and Éva Tardos, Addison-Wesley, 2015, online: https://www.cs.princeton.edu/~wayne/kleinberg-tardos/pdf/UnionFind.pdf.

[44] Alfred Aho, John Hopcroft, Jeffrey Ullmann, The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, MA, 1974.

[45] Hassan Aït-Kaci, FFF: a fuzzy facility for first-order terms, Java Software, August 2018, online: https:[...]/hlt/src/hlt/fot/fuz/syntax/docs/FuzzyFOTLatticeDoc/000StartHere.html.

[46] Maria Eugenia Cornejo, Jesús Medina Moreno, Clemente Rubio-Manzano, Towards a full fuzzy unification in the bousi prolog system, in: Fernando Gomide (Ed.), Proceedings of the International Conference on Fuzzy Systems (FUZZ-IEEE 2018), July 2018, online: https://[...]Towards_a_Full_Fuzzy_Unification_in_the_Bousi_Prolog_system.

[47] Mustapha Baziz, Mohand Boughanem, Gabriella Pasi, Henri Prade, A fuzzy set approach to concept-based information retrieval, in: E. Montseny, P. Sobrevilla (Eds.), Proceedings of the Joint 4th Conference of the European Society for Fuzzy Logic and Technology and the 11èmes Rencontres Francophones sur la Logique Floue et ses Applications, EUSFLAT/LFA, September 7–9, 2005, Barcelona (Spain), pp. 1287–1292, online: https://www.irit.fr/publis/ADRIA/BougPetal001a.pdf, September 2005.

[48] Hassan Aït-Kaci, Andreas Podelski, Seth C. Goldstein, Order-sorted feature theory unification, J. Log. Program. 30 (2) (1997) 99–124, online: http://www.hassan-ait-kaci.net/pdf/osf-theory-unification.pdf.

[49] Lotfi A. Zadeh, Fuzzy sets, Inf. Control 8 (1965) 338–353, online: https://people.eecs.berkeley.edu/~zadeh/papers/Fuzzy Sets[...].pdf.

[50] Pascual Julián Iranzo, Ginés Moreno, Jaime Penabad, Carlos Vázquez, A fuzzy logic programming environment for managing similarity and truth degrees, in: Santiago Escobar (Ed.), Proceedings XIV Jornadas sobre Programación y Lenguajes (PROLE 2014), in: Electronic Proceedings in Theoretical Computer Science, EPTCS, vol. 173, January 2015, pp. 71–86, online: https://arxiv.org/pdf/1501.02034.pdf.

[51] Raymond T. Yeh, Toward an Algebraic Theory of Fuzzy Relational Systems, Technical Report TR-25, Department of Computer Sciences and Electronics Research Center, the University of Texas at Austin, Austin, TX (USA), July 1973, online: ftp://www.cs.utexas.edu/pub/techreports/tr73-25.pdf.

[52] Lotfi A. Zadeh, Similarity relations and fuzzy orderings, Inf. Sci. 3 (1971) 177–200, online: https://people.eecs.berkeley.edu/~zadeh/papers/Similarity Relations[...].pdf.

Please cite this article in press as: H. Aït-Kaci, G. Pasi, Fuzzy lattice operations on first-order terms over signatures with similar constructors: A constraint-based approach, Fuzzy Sets Syst. (2019), https://doi.org/10.1016/j.fss.2019.03.019