Fuzzy Unification and Generalization of First-Order Terms over Similar Signatures

A Constraint-Based Approach

Hassan Aït-Kaci
Gabriella Pasi

27th LOPSTR

Namur, Belgium
October 10–12, 2017

This presentation's objective

- **Reformulate and extend** general results on (crisp & fuzzy) $\mathcal{FOT}$ **unification and generalization** ("*anti-unification*") seen **as lattice operations using** (crisp & fuzzy) **constraints**

- **Give declarative rulesets** for operational constraint-driven deductive and inductive **fuzzy inference over** $\mathcal{FOT}$**s when some signature symbols may be similar**

OK. . . And why is this interesting?. . .

- This provides a **formally clean** and **practically efficient** way to enable ***approximate reasoning*** (**deduction** and **learning**) ***with a very popular data structure*** used in logic-based data and knowledge processing systems

Some quick but important remarks about this presentation

We apologize in advance for the "*symbol soup*" in this talk ...

... but please do bear with us, as **this presentation is:**

- **only meant to give you an idea...** of what's in the paper with more examples and all proofs available here
- **necessary...** since we purport to be formal
- **not that complicated...** at least not for this audience — *we assume familiarity with Prolog's basic data structure and Fuzzy Logic notions*
- **really always the same...** once we get the basic gist

Presentation outline

- **First-Order Terms** — syntax of $\mathcal{FOT}$s
- **Subsumption** — pre-order relation on $\mathcal{FOT}$s
- **Unification** — glb operation on $\mathcal{FOT}$s
- **Generalization** — lub operation on $\mathcal{FOT}$s
- **Weak unification** — fuzzy glb of aligned $\mathcal{FOT}$s
- **Weak generalization** — fuzzy lub of aligned $\mathcal{FOT}$s
- **Full fuzzy unification** — fuzzy glb of misaligned $\mathcal{FOT}$s
- **Full fuzzy generalization** — fuzzy lub of misaligned $\mathcal{FOT}$s
- **Conclusion** — recapitulation and future work

The lattice of $\mathcal{FOT}$s

data structures that can be approximated

$\mathcal{FOT}$s on a signature of data constructors $\Sigma \stackrel{\text{def}}{=} \bigcup_{n \geq 0} \Sigma_n$

$$\mathcal{T}_{\Sigma,\mathcal{V}} \stackrel{\text{def}}{=} \mathcal{V}$$

$$\cup \; \{\, f(t_1, \; \cdots \;, t_n) \mid f \in \Sigma_n, \; n \geq 0,$$

$$t_i \in \mathcal{T}_{\Sigma,\mathcal{V}}, \; 1 \leq i \leq n \,\}$$

$\mathcal{FOT}$ subsumption pre-order relation

$$t_1 \preceq t_2$$

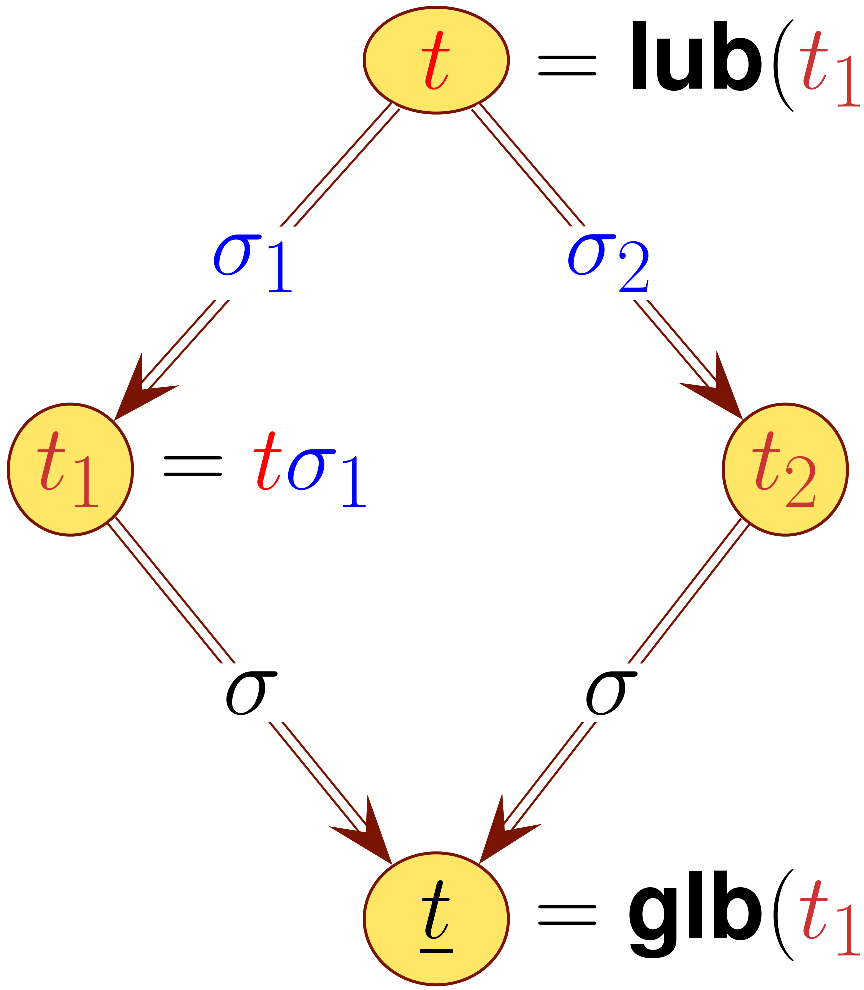
iff

$$\exists \sigma : \mathcal{V} \to \mathcal{T}_{\Sigma,\mathcal{V}}$$

s.t.

$$t_1 = t_2\sigma$$

$\mathcal{FOT}$ subsumption lattice operations



$$t = \textbf{lub}(t_1, t_2)$$

$$t_1 = t\sigma_1$$

$$t_2 = t\sigma_2$$

$$\underline{t} = \textbf{glb}(t_1, t_2) = \left\{ \begin{array}{c} t_1\sigma = t_2\sigma \\ t\sigma_1\sigma = t\sigma_2\sigma \end{array} \right\}$$

Declarative lattice operations on $\mathcal{FOT}$s. . .

using constraints

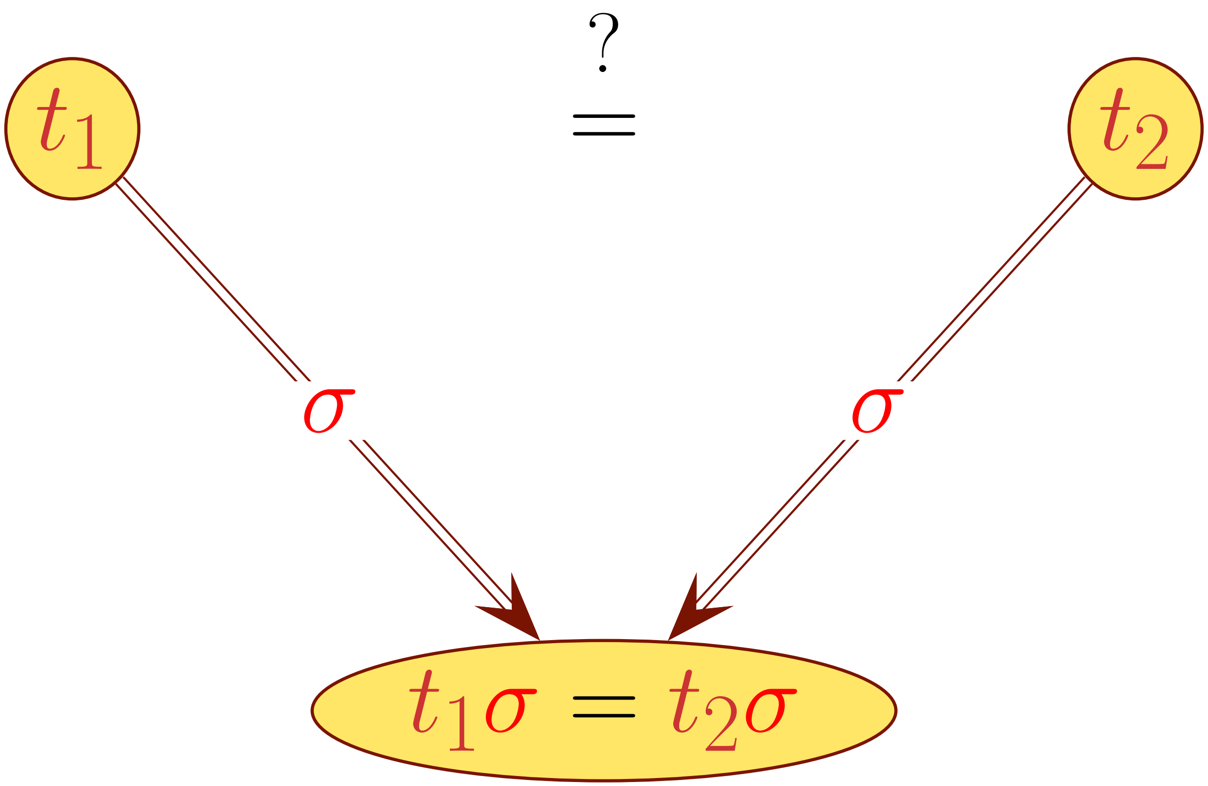
- **1930** – Jacques Herbrand gives normalization rules for sets of term equalities in his PhD thesis (*Chap. 5, Sec. 2.4, pp. 95 – 96*) but does not call this "*unification*"
- **1960** – Dag Prawitz expresses this as reduction rules as part of proof normalization procedure for Natural Deduction in F.O. Logic (Gentzen, 1934)
- **1965** – J. Alan Robinson gives a procedural algorithm and uses it to lift the resolution principle from Propositional Logic to F.O. Logic — calling it "*unification*"
- **1967** – Jean van Heijenoort translates Chap. 5 of Herbrand's thesis into English
- **1971** – Warren Goldfarb translates Herbrand's full thesis into English

- **1976** – Gérard Huet dates the first $\mathcal{FOT}$ unification algorithm to initial equation normalization in Herbrand's 1930 PhD thesis (*also in Chap. 5 in Huet's thesis!*)

- **1982** – Alberto Martelli *&* Ugo Montanari give unification rules (with no mention of Herbrand's thesis, although Huet's thesis is cited)

Interestingly, Martelli *&* Montanari use a preprocessing method that uses generalization implicitly (to compute "*common parts*" in preprocessing equations into congruence classes of equations called "*multi-equations*") — but do not point out that it is dual to unification

$\mathcal{FOT}$ unification as a constraint



Declarative unification rule

A unification rule rewrites a prior set of equations E into a posterior set of equations E' whenever an optional meta-condition holds:

RULE NAME:

$$\frac{\textit{Prior set of equations} \quad E}{\textit{Posterior set of equations} \quad E'} \quad [\textit{Optional meta-condition}]$$

Herbrand–Martelli-Montanari $\mathcal{FOT}$ unification rules

Term Decomposition:

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

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:

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

Variable Erasure:

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

Moving on to. . .

declarative constraint-based generalization

Generalization

- The lattice-theoretic properties of $\mathcal{FOT}$s as data structures pre-ordered by subsumption were exposed independently and simultaneously by Reynolds and Plotkin in **1970**
- Both gave a formal definition of $\mathcal{FOT}$ generalization and each proved correct a *procedural* specification for computing it
- *However*, ...so far, a ***declarative*** formal specification was lacking — which we provide here
- **Why should we care?...** Well, because:
  - **syntax-driven rules give an operational semantics as constraint solving needing no control specification** (use any rule that applies in any order)
  - **each rule's correctness is independent of that of the others** (they share no global context)
  - **eases the formal specification of more expressive approximation over the same data structure** (such as ***fuzzy constraints*** on $\mathcal{FOT}$s)

$\mathcal{FOT}$ generalization judgment

Statement of the form:

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

where (for $i = 1, 2$):

- $t \in \mathcal{T}$ and $t_i \in \mathcal{T}$ are $\mathcal{FOT}$s
- $\sigma_i : \mathcal{V} \to \mathcal{T}$ and $\theta_i : \mathcal{V} \to \mathcal{T}$ are substitutions

$\mathcal{FOT}$ generalization judgment validity

A generalization judgment:

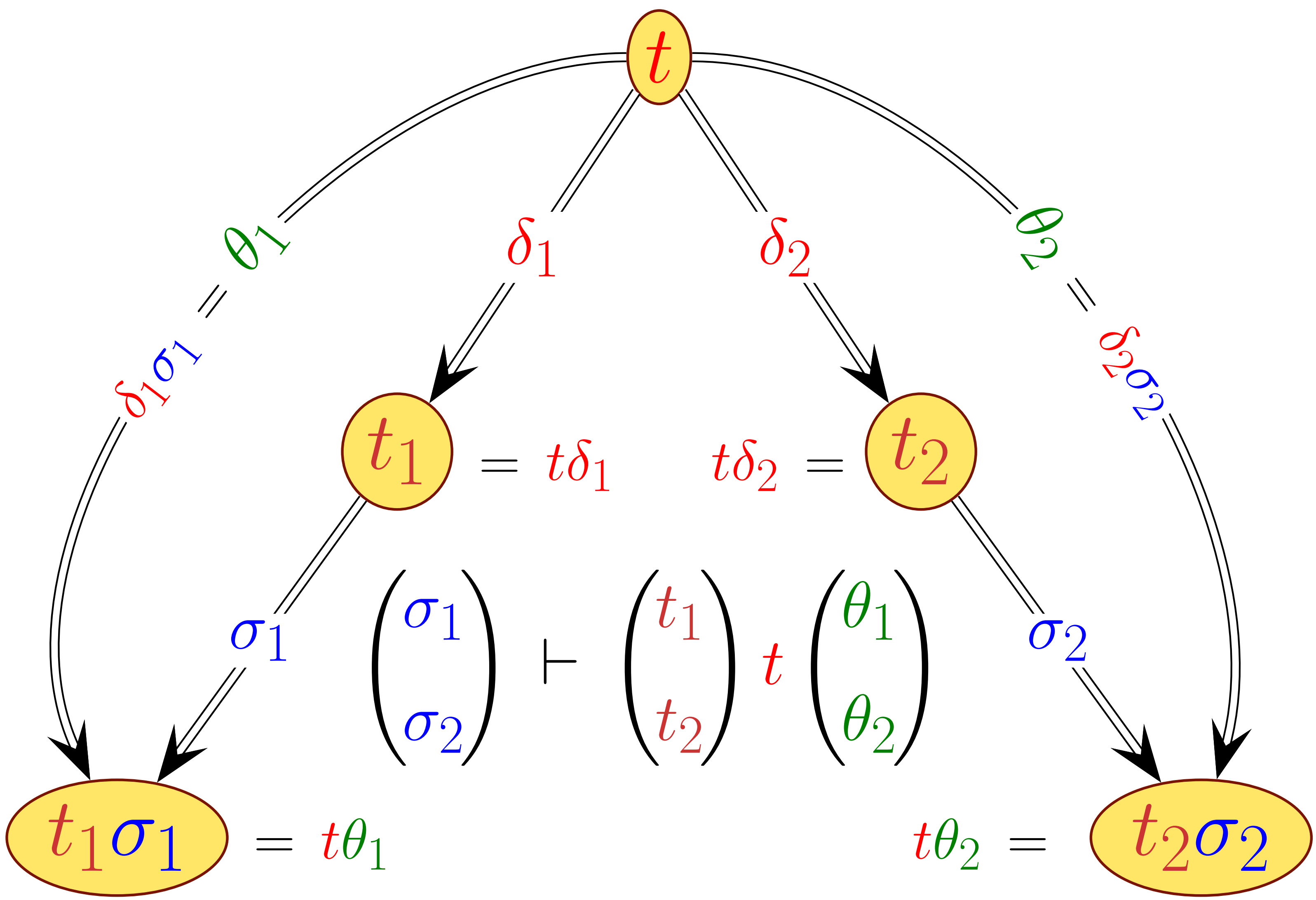
$$\begin{pmatrix}\sigma_1\\ \sigma_2\end{pmatrix} \vdash \begin{pmatrix}t_1\\ t_2\end{pmatrix} t \begin{pmatrix}\theta_1\\ \theta_2\end{pmatrix}$$

is deemed **valid** whenever:

$$\boxed{t_i\sigma_i = t\theta_i}$$

with $\theta_i \preceq \sigma_i$ (*i.e.*, $\exists\delta_i$ s.t. $\theta_i = \delta_i\sigma_i$) for $i = 1, 2$

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



Declarative generalization axiom

Statement of the form:

AXIOM NAME:

[*Optional meta-condition*]

Judgment J

which reads:

> "*whenever the optional meta-condition holds, judgement* J *is always valid*"

$\mathcal{FOT}$ generalization axioms

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 :

[$t_1 \in \mathcal{V}$ or $t_2 \in \mathcal{V}$; $t_1 \neq t_2$; X 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 :

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

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

Declarative generalization inference rule

Conditional Horn rule of generalization judgments of the form:

RULE NAME:

$$\frac{[\textit{Optional Meta-Condition}] \quad \textit{Prior Judgment } J_1 \quad \cdots \quad \textit{Prior Judgment } J_n}{\textit{Posterior Judgment } J}$$

(for $n \geq 0$) — which reads:

"*whenever the optional meta-condition holds, if all the n prior judgements J_n are valid, then the posterior judgement J is also valid*"

Declarative $\mathcal{FOT}$ generalization rule for equal functors

EQUAL FUNCTORS :

$[n \geq 0]$

$$\frac{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix} \vdash \begin{pmatrix}s_1' \\ t_1'\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1 \\ \sigma_2^1\end{pmatrix} \quad \cdots \quad \begin{pmatrix}\sigma_1^{n-1} \\ \sigma_2^{n-1}\end{pmatrix} \vdash \begin{pmatrix}s_n' \\ t_n'\end{pmatrix} u_n \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}}{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix} \vdash \begin{pmatrix}f(s_1, \cdots, s_n) \\ f(t_1, \cdots, t_n)\end{pmatrix} f(u_1, \cdots, u_n) \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}}$$

where $\begin{pmatrix}s_i' \\ t_i'\end{pmatrix} \stackrel{\text{def}}{=} \begin{pmatrix}s_i \\ t_i\end{pmatrix} \uparrow \begin{pmatrix}\sigma_1^{i-1} \\ \sigma_2^{i-1}\end{pmatrix}$ for $i = 1, \ldots, n$.

"Unapplying" a pair of substitutions on a pair of $\mathcal{FOT}$s

Rule "**EQUAL FUNCTORS**" uses operation "***unapply***" '$\uparrow$' on a pair of terms t_1, t_2 given a pair of substitutions σ_1, σ_2:

$$\begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \uparrow \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} \stackrel{\text{def}}{=} \begin{cases} \begin{pmatrix} X \\ X \end{pmatrix} & \text{if } t_i = X\sigma_i, \text{ for } i = 1, 2 \\ \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} & \text{otherwise} \end{cases}$$

Declarative $\mathcal{FOT}$ generalization rule for $n = 0$

NB: for $n = 0$, the rule **EQUAL FUNCTORS** becomes an axiom; *viz.*, for any constant c:

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

for any pair σ_1, σ_2

$\mathcal{FOT}$ generalization example

Consider the terms $f(a, a, a)$ and $f(b, c, c)$ to generalize; *i.e.*:

- Find term t and substitutions σ_1 and σ_2 such that $t\sigma_1 = f(a, a, a)$ and $t\sigma_2 = f(b, c, c)$:

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

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

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix} \vdash \begin{pmatrix} f(a, a, a) \\ f(b, c, c) \end{pmatrix} f(u_1, u_2, u_3) \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}$$

  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**:

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

$\mathcal{FOT}$ generalization example (ctd.)

- u_2 is the generalization of $\begin{pmatrix} a \\ c \end{pmatrix} \uparrow \begin{pmatrix} \{a/X\} \\ \{b/X\} \end{pmatrix}$; that is, of a and c;

  and by Rule **Unequal Functors**:

$$\begin{pmatrix} \{a/X\} \\ \{b/X\} \end{pmatrix} \vdash \begin{pmatrix} a \\ c \end{pmatrix} Y \begin{pmatrix} \{a/X, a/Y\} \\ \{b/X, c/Y\} \end{pmatrix} \quad \text{therefore} \quad u_2 = Y$$

- u_3 is the generalization of $\begin{pmatrix} a \\ c \end{pmatrix} \uparrow \begin{pmatrix} \{a/X, a/Y\} \\ \{b/X, c/Y\} \end{pmatrix}$; that is, of Y and Y;

  and by Rule **Equal Variables**:

$$\begin{pmatrix} \{a/X, a/Y\} \\ \{b/X, c/Y\} \end{pmatrix} \vdash \begin{pmatrix} Y \\ Y \end{pmatrix} Y \begin{pmatrix} \{a/X, a/Y\} \\ \{b/X, c/Y\} \end{pmatrix} \quad \text{therefore} \quad u_3 = Y$$

- therefore, the overall constraint is thus solved proving the overall judgment valid as:

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

i.e., $t = f(X, Y, Y)$, with $\sigma_1 = \{a/X, a/Y\}$ s.t. $t\sigma_1 = f(a, a, a)$,

and $\sigma_2 = \{b/X, c/Y\}$ and $t\sigma_2 = f(b, c, c)$

Going from crisp to fuzzy. . .

extending the foregoing to fuzzy lattice operations as fuzzy constraints

Fuzzy equivalence relation on a (crisp) set (fuzzy set of pairs)

When S is a finite discrete set $\{x_1, \ldots, x_n\}$, 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 $\sim \subseteq [0,1]^2$ as follows; $\forall\, i, j, k = 1, \ldots, n$:

- *reflexivity*: $\sim_{ii} = 1$ entries on the diagonal are equal to 1
- *symmetry*: $\sim_{ij} = \sim_{ji}$ symmetric entries on either side of the diagonal are equal
- *transitivity*: $\sim_{ik} \wedge \sim_{kj} \leq \sim_{ij}$ going via an intermediate will always result in a smaller or equal truth value than going directly

N.B.: if $x_i \sim_\alpha x_j$ for some $\alpha \in (0,1]$, then $x_i \sim_\beta x_j$ for all $\beta \in (0, \alpha]$

Similar functors and terms

Sessa, *TCS* 2002

Given a similarity relation $\sim$ on signature Σ Sessa extends it homomorphically to $\mathcal{FOT}$s as follows:

- for all $X \in \mathcal{V}$: $X \sim_1 X$
- for all $X \in \mathcal{V}$ and $t \in \mathcal{T}$ s.t. $t \neq X$: $X \sim_0 t$ and $t \sim_0 X$
- for $f \in \Sigma_n$ and $g \in \Sigma_n$ s.t. $f \sim_\alpha g$ and $s_i \sim_{\alpha_i} t_i$:

$$f(s_1, \cdots, s_n) \sim_{\alpha \wedge \bigwedge_{i=1}^{n} \alpha_i} g(t_1, \cdots, t_n)$$

$\alpha \in [0,1]$, $\alpha_i \in [0,1]$ $(i = 1, \ldots, n)$

Unification degree of pair of terms (0 for dissimilar pairs)

NB: (1) for Sessa's "weak" similarity on Σ: $n \neq m \rightarrow (\sim \cap \Sigma_m \times \Sigma_n = \emptyset)$, for all $m, n \geq 0$ and (2) operation $\wedge$ is $\min$ — but other interpretations are possible

Fuzzy subsumption

$$\alpha \in (0, 1]$$

$$t_1 \preceq_\alpha t_2$$

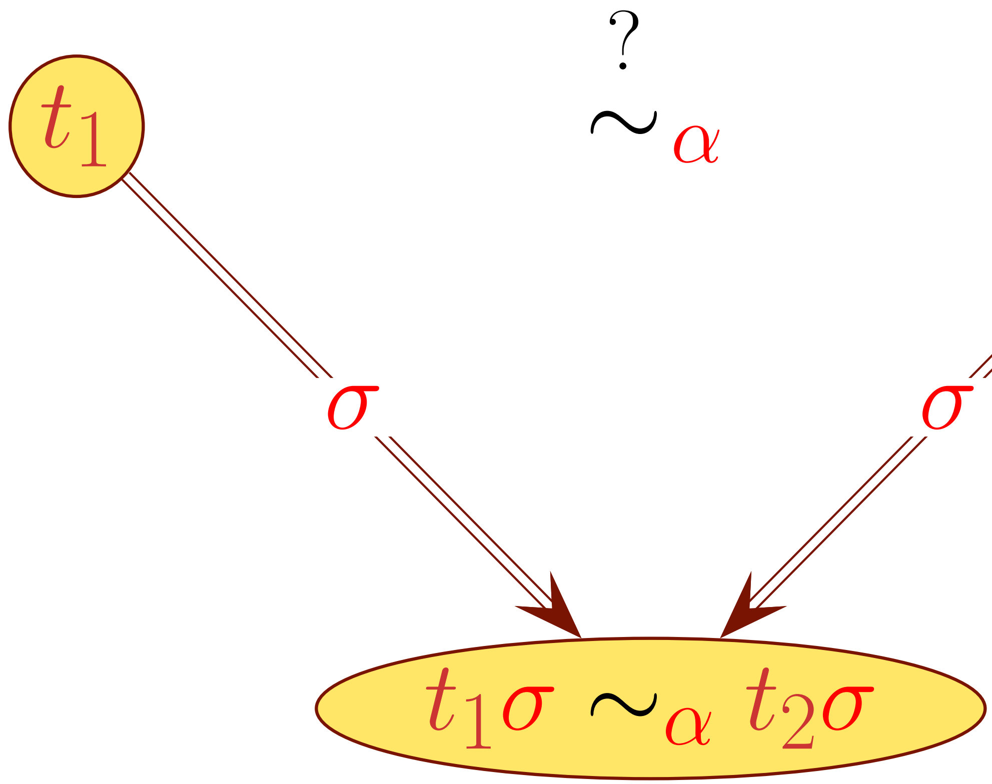
iff

$$\exists \sigma : \mathcal{V} \to \mathcal{T}_{\Sigma,\mathcal{V}}$$

s.t.

$$t_1 \sim_\alpha t_2\sigma$$

Fuzzy unification as a constraint



$$\alpha \in (0, 1]$$

Fuzzy unification rule

A fuzzy unification rule rewrites E_α, a prior set of equations E with truth value $\alpha \in (0, 1]$, into $E'_{\alpha'}$, a posterior set of equations E' with truth value $\alpha' \in [0, \alpha]$, when an optional meta-condition holds:

RULE NAME:

$$\frac{\textit{Prior set of equations } E_\alpha}{\textit{Posterior set of equations } E'_{\alpha'}} \text{ [Optional meta-condition]}$$

Sessa's "weak" fuzzy unification

Variable Elimination:

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

Crisp version is HMM's:

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

Variable Erasure:

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

Crisp version is HMM's:

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

Equation Orientation:

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

Crisp version is HMM's:

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

WEAK TERM DECOMPOSITION:

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

NB: only unification rule among HMM's that constrains the overall unification degree upon equating similar terms with different constructors

CRISP VERSION IS ALSO HMM'S:

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

Fuzzy unification example

Let $\{a, b, c, d\} \subseteq \Sigma_0$, $\{f, g\} \subseteq \Sigma_2$, $\{h\} \subseteq \Sigma_3$; with $a \sim_{.7} b$, $c \sim_{.6} d$, $f \sim_{.9} g$.

- Fuzzy equational constraint to normalize:

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

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

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

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

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

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

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

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

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

Fuzzy unification example (ctd.)

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

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

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

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

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

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

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

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

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

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

This is in normal form, yielding substitution σ:

$$\sigma = \{\, X_1 = a,\ Y_1 = c,\ X_2 = f(a,a) \,\}$$

with truth value $.6$ so that:

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

Moving on to. . .

fuzzy generalization

Fuzzy generalization judgment

Statement of the form:

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

where (for $i = 1, 2$):

- $t \in \mathcal{T}$ and $t_i \in \mathcal{T}$ are $\mathcal{FOT}$s
- $\sigma_i : \mathcal{V} \to \mathcal{T}$ are substitutions and $\alpha \in [0, 1]$
- $\theta_i : \mathcal{V} \to \mathcal{T}$ are substitutions and $\beta \in [0, 1]$

Fuzzy generalization judgment validity

A fuzzy generalization judgment:

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

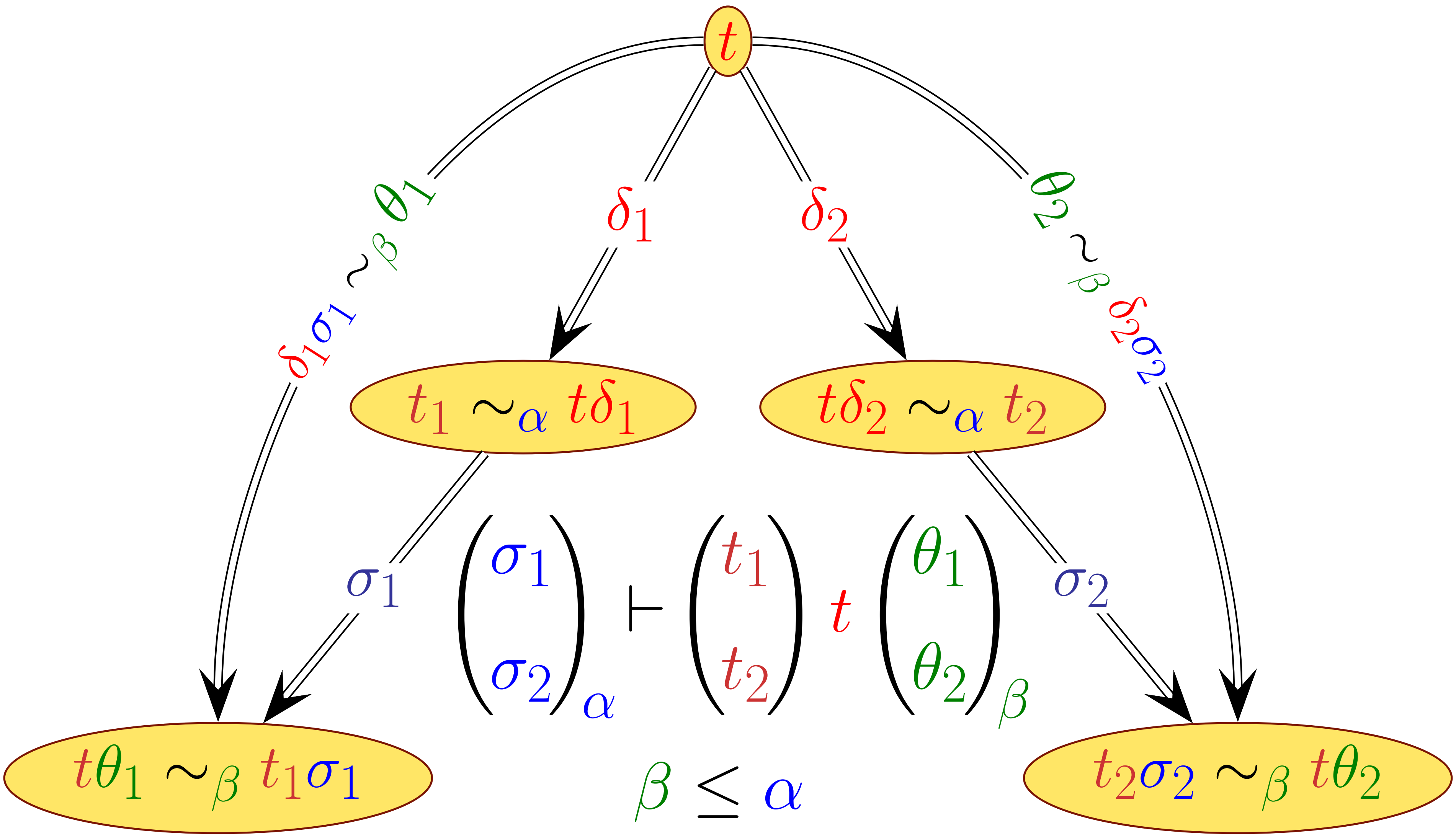
is deemed **valid** whenever ($i = 1, 2$):

$$t_i\sigma_i \sim_\beta t\theta_i$$

with: $0 \leq \beta \leq \alpha \leq 1$

and: $\theta_i \preceq_\beta \sigma_i$ (*i.e.*, $\exists \delta_i$ s.t. $\theta_i \sim_\beta \delta_i\sigma_i$)

Fuzzy generalization judgment validity as a constraint



Fuzzy generalization axioms

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 :

$[t_1 \in \mathcal{V}$ or $t_2 \in \mathcal{V};\ \ t_1 \neq t_2;\ \ X$ 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 :

$[f \not\sim g;\ m \geq 0, n \geq 0;\ X$ is new$]$

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

Fuzzy generalization rule for similar functors

SIMILAR FUNCTORS :

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

$$\frac{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix}s_1' \\ t_1'\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1 \\ \sigma_2^1\end{pmatrix}_{\alpha_1} \quad \cdots \quad \begin{pmatrix}\sigma_1^{n-1} \\ \sigma_2^{n-1}\end{pmatrix}_{\alpha_{n-1}} \vdash \begin{pmatrix}s_n' \\ t_n'\end{pmatrix} u_n \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}_{\alpha_n}}{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}f(s_1, \cdots, s_n) \\ g(t_1, \cdots, t_n)\end{pmatrix} f(u_1, \cdots, u_n) \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}_{\alpha_n}}$$

where $\begin{pmatrix}s_i' \\ t_i'\end{pmatrix} \stackrel{\text{def}}{=} \begin{pmatrix}s_i \\ t_i\end{pmatrix} \uparrow_{\alpha_i} \begin{pmatrix}\sigma_1^{i-1} \\ \sigma_2^{i-1}\end{pmatrix}$ for $i = 1, \ldots, n$.

Fuzzy "unapplication" of a pair of substitutions on a pair of $\mathcal{FOT}$s

Rule "**Similar Functors**" uses operation "***fuzzy unapply***" '$\uparrow_\alpha$' on a pair of terms t_1, t_2 given a pair of substitutions σ_1, σ_2 and truth value $\alpha \in [0, 1]$:

$$\begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \uparrow_\alpha \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} \stackrel{\text{def}}{=} \begin{cases} \begin{pmatrix} X \\ X \end{pmatrix} & \text{if } t_i \sim_\alpha X\sigma_i, \text{ for } i = 1, 2 \\ \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} & \text{otherwise} \end{cases}$$

Fuzzy generalization example

Again, let $\{a, b, c, d\} \subseteq \Sigma_0$, $\{f, g\} \subseteq \Sigma_2$, $\{h\} \subseteq \Sigma_3$; with $a \sim_{.7} b$, $c \sim_{.6} d$, $f \sim_{.9} g$.

- Terms to generalize:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)) \\ h(X_2, X_2, g(c, d)) \end{pmatrix} t \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha$$

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

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} h(f(a, X_1), g(X_1, b), f(Y_1, Y_1)) \\ h(X_2, X_2, g(c, d)) \end{pmatrix} h(u_1, u_2, u_3) \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix}_\alpha$$

  where:

  - u_1 is the fuzzy generalization of $\begin{pmatrix} f(a, X_1) \\ X_2 \end{pmatrix} \uparrow_1 \begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}$; that is, of $f(a, X_1)$ and X_2;

    by Rule **Fuzzy Variable-Term**:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} f(a, X_1) \\ X_2 \end{pmatrix} X \begin{pmatrix} \{f(a, X_1)/X\} \\ \{X_2/X\} \end{pmatrix}_1 \quad \text{so } u_1 = X$$

Fuzzy generalization example (ctd.)

– u_2 is the fuzzy generalization of $\begin{pmatrix} g(X_1,b) \\ X_2 \end{pmatrix} \uparrow_1 \begin{pmatrix} \{f(a,X_1)/X\} \\ \{X_2/X\} \end{pmatrix}$; *i.e.*, $g(X_1,b)$ and X_2 by Rule **Fuzzy Variable-Term**:

$$\begin{pmatrix} \{f(a,X_1)/X\} \\ \{X_2/X\} \end{pmatrix}_1 \vdash \begin{pmatrix} g(X_1,b) \\ X_2 \end{pmatrix} \; Y \; \begin{pmatrix} \{\cdots, g(X_1,b)/Y\} \\ \{\cdots, X_2/Y\} \end{pmatrix}_1 \quad \text{so } u_2 = Y$$

– $u_3 = f(v_1, v_2)$ is the fuzzy generalization of $\begin{pmatrix} f(Y_1,Y_1) \\ g(c,d) \end{pmatrix} \uparrow_{.9} \begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix}$; that is, of $f(Y_1,Y_1)$ and $g(c,d)$ with truth value $.9$, because of Rule **Similar Functors** and $f \sim_{.9} g$, where:

∗ v_1 is the fuzzy generalization of $\begin{pmatrix} Y_1 \\ c \end{pmatrix} \uparrow_{.9} \begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix}$; *i.e.*, Y_1 and c by Rule **Fuzzy Variable-Term**:

$$\begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix}_{.9} \vdash \begin{pmatrix} Y_1 \\ c \end{pmatrix} \; Z \; \begin{pmatrix} \{\cdots, Y_1/Z\} \\ \{\cdots, c/Z\} \end{pmatrix}_{.9} \quad \text{so } v_1 = Z$$

Fuzzy generalization example (ctd.)

* v_2 is the fuzzy generalization of $\begin{pmatrix} Y_1 \\ d \end{pmatrix} \uparrow_{.9} \begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y, Y_1/Z\} \\ \{X_2/X, X_2/Y, c/Z\} \end{pmatrix}$; *i.e.*, Y_1 and d; by Rule **Fuzzy Variable-Term**:

$$\begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y, Y_1/Z\} \\ \{X_2/X, X_2/Y, c/Z\} \end{pmatrix}_{.9} \vdash \begin{pmatrix} Y_1 \\ d \end{pmatrix} U \begin{pmatrix} \{\cdots, Y_1/U\} \\ \{\cdots, d/U\} \end{pmatrix}_{.9}$$

so, $v_2 = U$

in other words, $u_3 = f(Z,U)$ since:

$$\begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y\} \\ \{X_2/X, X_2/Y\} \end{pmatrix}_1 \vdash \begin{pmatrix} f(Y_1,Y_1) \\ g(c,d) \end{pmatrix} f(Z,U) \begin{pmatrix} \{\cdots, Y_1/Z, Y_1/U\} \\ \{\cdots, c/Z, d/U\} \end{pmatrix}_{.9}$$

Therefore:

$$\begin{pmatrix} \emptyset \\ \emptyset \end{pmatrix}_1 \vdash \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} h(X,Y,f(Z,U)) \begin{pmatrix} \{f(a,X_1)/X, g(X_1,b)/Y, Y_1/Z, Y_1/U\} \\ \{X_2/X, X_2/Y, c/Z, d/U\} \end{pmatrix}_{.9}$$

whereby

$$t\sigma_1 = h(f(a,X_1), g(X_1,b), f(Y_1,Y_1)) = t_1,$$
$$t\sigma_2 = h(X_2,X_2,f(c,d)) \sim_{.9} h(X_2,X_2,g(c,d)) = t_2$$

So we now have fuzzy lattice operations on $\mathcal{FOT}$...

but, aren't we missing something?

Hey! . . . *but what about similar functors with different arities?*

. . . or equal arities but different order of arguments?

- Disallowed in Sessa's weak unification, even though this would be of great convenience; *e.g.*, in **approximate data retrieval and mining** in **non-aligned databases**

For example:

$$\texttt{person}(\texttt{Name}, \texttt{SSN}, \texttt{Address}) \sim_{\alpha} \texttt{individual}(\texttt{Name}, \texttt{DoB}, \texttt{SSN}, \texttt{Address})$$

for $\alpha \in (0, 1]$ would allow fuzzy matching of non-aligned similar records

Similar terms with different argument number or order

Given $\sim : \Sigma^2 \rightarrow [0, 1]$ similarity on $\Sigma \stackrel{\text{def}}{=} \bigcup_{n \geq 0} \Sigma_n$, s.t.:

- $\sim \cap \Sigma_m \times \Sigma_n \neq \emptyset$ for some $m \geq 0$, $n \geq 0$, with $m \neq n$
- for $f \in \Sigma_m$, $g \in \Sigma_n$, $0 \leq m \leq n$, whenever $f \sim_\alpha g$ there is an *injective mapping* $p : \{1, \ldots, m\} \rightarrow \{1, \ldots, n\}$ that is denoted as $f \sim^p_\alpha g$; *e.g.*:

$$\mathtt{person}(\mathtt{Name}, \mathtt{SSN}, \mathtt{Address}) \sim^{\{1 \rightarrow 1, 2 \rightarrow 3, 3 \rightarrow 4\}}_{.9} \mathtt{individual}(\mathtt{Name}, \mathtt{DoB}, \mathtt{SSN}, \mathtt{Address})$$

N.B.: m and n are such that $0 \leq m \leq n$; so the one-to-one argument-position mapping goes from the lesser set to the larger set

Unifying similar functors w/ different arg. number/order

GENERIC WEAK TERM DECOMPOSITION :

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

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

FUZZY EQUATION REORIENTATION :

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

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

Generalizing similar functors w/ different arg. number/order

FUNCTOR/ARITY SIMILARITY LEFT :

$$\left[f \sim_{\beta}^{p} g;\ \ 0 \le m \le n;\ \ \alpha_0 \overset{\text{def}}{=} \alpha \wedge \beta \right]$$

$$\frac{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix}s'_1 \\ t'_1\end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1 \\ \sigma_2^1\end{pmatrix}_{\alpha_1} \quad \cdots \quad \begin{pmatrix}\sigma_1^{m-1} \\ \sigma_2^{m-1}\end{pmatrix}_{\alpha_{m-1}} \vdash \begin{pmatrix}s'_m \\ t'_m\end{pmatrix} u_m \begin{pmatrix}\sigma_1^m \\ \sigma_2^m\end{pmatrix}_{\alpha_m}}{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha} \vdash \begin{pmatrix}f(s_1,\ldots,s_m) \\ g(t_1,\ldots,t_n)\end{pmatrix} f(u_1,\ldots,u_m) \begin{pmatrix}\sigma_1^m \\ \sigma_2^m\end{pmatrix}_{\alpha_m}}$$

where $\begin{pmatrix}s'_i \\ t'_i\end{pmatrix} \overset{\text{def}}{=} \begin{pmatrix}s_i \\ t_{p(i)}\end{pmatrix} \uparrow_{\alpha_i} \begin{pmatrix}\sigma_1^i \\ \sigma_2^i\end{pmatrix}$ for $i = 1, \ldots, m$.

Generalizing similar functors w/ different arg. number/order (ctd.)

FUNCTOR/ARITY SIMILARITY RIGHT :

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

$$\frac{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha_0} \vdash \begin{pmatrix} s'_1 \\ t'_1 \end{pmatrix} u_1 \begin{pmatrix}\sigma_1^1 \\ \sigma_2^1\end{pmatrix}_{\alpha_1} \quad \cdots \quad \begin{pmatrix}\sigma_1^{n-1} \\ \sigma_2^{n-1}\end{pmatrix}_{\alpha_{n-1}} \vdash \begin{pmatrix} s'_n \\ t'_n \end{pmatrix} u_n \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}_{\alpha_n}}{\begin{pmatrix}\sigma_1^0 \\ \sigma_2^0\end{pmatrix}_{\alpha} \vdash \begin{pmatrix} f(s_1, \ldots, s_m) \\ g(t_1, \ldots, t_n) \end{pmatrix} g(u_1, \ldots, u_n) \begin{pmatrix}\sigma_1^n \\ \sigma_2^n\end{pmatrix}_{\alpha_n}}$$

where $\begin{pmatrix} s'_i \\ t'_i \end{pmatrix} \stackrel{\text{def}}{=} \begin{pmatrix} s_{p(i)} \\ t_i \end{pmatrix} \uparrow_{\alpha_i} \begin{pmatrix}\sigma_1^i \\ \sigma_2^i\end{pmatrix}$ for $i = 1, \ldots, n$.

OK — we've had enough for now!. . .

let us recap and conclude

Recapitulation

We overviewed 3 lattice structures over $\mathcal{FOT}$s (1 crisp and 2 fuzzy), gave declarative axioms and rules, and expressed the 6 corresponding dual lattice operations as constraints (✔ indicates original contribution):

- **Conventional signature**
  - Unification (*Herbrand–Martelli&Montanari's*)
  - ✔ Generalization (*declarative version of Reynolds–Plotkin's*)
- **Signature with aligned similarity**
  - "Weak" fuzzy unification (*Sessa's*)
  - ✔ "Weak" fuzzy generalization (*dual to Sessa's*)
- **Signature with misaligned similarity**
  - ✔ Full fuzzy unification (*different/mixed arities*)
  - ✔ Full fuzzy generalization (*different/mixed arities*)

Future Work?

- **Implement!**
  - ☞ Java/Scala Libraries
  - ☞ Extend Bousi∼Prolog?
  - ☞ Applications!
  - ☞ *Etc.*, . . .
- ***OK...*** But can all this be made more expressive somehow?

  Yes! — Extend these results to the lattice of Order-Sorted Feature terms (fuzzy $\mathcal{OSF}$ constraints?)

  We’re working on it. . .

Coming soon to a ~~theater~~ ***conference*** near you!. . . ☺

Thank You For Your Attention !

Hassan Aït-Kaci

hak@acm.org

Gabriella Pasi

pasi@disco.unimib.it