Proposal for a second Examples Class (on 2nd ½ of Example Sheet):

THURSDAY 13 MARCH
16:00 – 17:00
SW01

Lecture 8: nominal unification

Sample αProlog code

```
id : name_type. (* variables *)
tm : type.      (* lambda terms *)
var : id -> tm.
app : tm -> tm -> tm.
lam : id\tm -> tm.
pred subst (id\tm) tm tm.
(* "subst (a\X) Y Z" holds if Z is the result of capture-avoiding substitution
   of Y for all free occurrences of var a in X *)
subst (a\var a) Y Y.
subst (a\X) Y X :- a # X.
subst (a\app X X') Y (app Z Z') :- subst (a\X) Y Z, subst (a\X') Y Z'.
subst (a\lam(b\X)) Y (lam (b\Z)) :- subst (a\X) Y Z, b # Y.
```

```
?- subst (b\lam(a\var b)) (var a) X. (* search for X satisfying X = λa.b[a/b] *)
Yes. X = lam(a'\var a) (* X is λa'.a, not λa.a *)
```

As for Prolog, search for solutions to queries involves resolution (try to unify query with head of each clause), but using nominal unification, which solves α-equivalence and freshness constraints.

Examples of unification 'mod α'

over the nominal algebraic signature Σ for λ-calculus:
name-sort Var, data-sort Term, operations

$\mathtt{V} : \mathtt{Var} \longrightarrow \mathtt{Term}$

$\mathtt{A} : \mathtt{Term}, \mathtt{Term} \longrightarrow \mathtt{Term}$

$\mathtt{L} : \mathtt{Var}.\mathtt{Term} \longrightarrow \mathtt{Term}$

Ex. 1: does there exist a $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a.\mathtt{L}(b.\mathtt{A}(t, \mathtt{V}\, b))) =_\alpha \mathtt{L}(b.\mathtt{L}(a.\mathtt{A}(\mathtt{V}\, a, t)))$$

(where $a \neq b$)?

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a.\mathtt{L}(b.\mathtt{A}(\mathtt{V}\, b, t_1))) =_\alpha \mathtt{L}(a.\mathtt{L}(a.\mathtt{A}(\mathtt{V}\, a, t_2)))$$

(where $a \neq b$)?

Alpha-equivalence

$$=_\alpha \subseteq \Sigma(\mathrm{S}) \times \Sigma(\mathrm{S})$$

$$\frac{a \in \mathbb{A}}{a =_\alpha a} \qquad \frac{t =_\alpha t'}{\mathrm{op}\, t =_\alpha \mathrm{op}\, t'} \qquad \frac{}{() =_\alpha ()}$$

$$\frac{t_1 =_\alpha t_1' \qquad t_2 =_\alpha t_2'}{t_1 , t_2 =_\alpha t_1' , t_2'}$$

$$\frac{(a_1\ a) \cdot t_1 =_\alpha (a_2\ a) \cdot t_2 \qquad a \# (a_1, t_1, a_2, t_2)}{a_1 . t_1 =_\alpha a_2 . t_2}$$

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$

(where $a \neq b$)?

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$

(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}((a\;c)\cdot t\,,\mathtt{V}\,b)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,(b\;c)\cdot t))$$

where $c \mathrel{\#} (a, b, t)$

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$

(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}((a\ c)\cdot t\,,\mathtt{V}\,b)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,(b\ c)\cdot t))$$

where $c \mathrel{\#} (a, b, t)$

$$\mathtt{A}((b\ d)(a\ c)\cdot t\,,\mathtt{V}\,d) =_\alpha \mathtt{A}(\mathtt{V}\,d\,,(a\ d)(b\ c)\cdot t))$$

where $d \mathrel{\#} c$, $d, c \mathrel{\#} (a, b, t)$

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}((a\ c)\cdot t\,,\mathtt{V}\,b)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,(b\ c)\cdot t))$$
where $c \mathrel{\#} (a,b,t)$

$$\mathtt{A}((b\ d)(a\ c)\cdot t\,,\mathtt{V}\,d) =_\alpha \mathtt{A}(\mathtt{V}\,d\,,(a\ d)(b\ c)\cdot t))$$
where $d \mathrel{\#} c$, $d,c \mathrel{\#} (a,b,t)$

$(b\ d)(a\ c)\cdot t =_\alpha \mathtt{V}\,d$ and $\mathtt{V}\,d =_\alpha (a\ d)(b\ c)\cdot t$
where $d \mathrel{\#} c$, $d,c \mathrel{\#} (a,b,t)$

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}((a\ c)\cdot t\,,\mathtt{V}\,b)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,(b\ c)\cdot t))$$
where $c \mathbin{\#} (a, b, t)$

$$\mathtt{A}((b\ d)(a\ c)\cdot t\,,\mathtt{V}\,d) =_\alpha \mathtt{A}(\mathtt{V}\,d\,,(a\ d)(b\ c)\cdot t))$$
where $d \mathbin{\#} c$, $d, c \mathbin{\#} (a, b, t)$

$(b\ d)(a\ c)\cdot t =_\alpha \mathtt{V}\,d$ and $\mathtt{V}\,d =_\alpha (a\ d)(b\ c)\cdot t$
where $d \mathbin{\#} c$, $d, c \mathbin{\#} (a, b, t)$

$t =_\alpha \mathtt{V}\,b$ and $\mathtt{V}\,a =_\alpha t$
where $d \mathbin{\#} c$, $d, c \mathbin{\#} (a, b, t)$

Ex. 1: does there exist $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$

(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}((a\ c)\cdot t\,,\mathtt{V}\,b)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,(b\ c)\cdot t))$$

where $c \# (a,b,t)$

$$\mathtt{A}((b\ d)(a\ c)\cdot t\,,\mathtt{V}\,d) =_\alpha \mathtt{A}(\mathtt{V}\,d\,,(a\ d)(b\ c)\cdot t))$$

where $d \# c$, $d,c \# (a,b,t)$

$(b\ d)(a\ c)\cdot t =_\alpha \mathtt{V}\,d$ and $\mathtt{V}\,d =_\alpha (a\ d)(b\ c)\cdot t$

where $d \# c$, $d,c \# (a,b,t)$

$t =_\alpha \mathtt{V}\,b$ and $\mathtt{V}\,a =_\alpha t$

where $d \# c$, $d,c \# (a,b,t)$

$$\mathtt{V}\,b =_\alpha \mathtt{V}\,a$$

$$b = a$$

contradicting $a \neq b$ — so no such t can exist.

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$

(where $a \neq b$)?

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

$$\mathtt{A}(\mathtt{V}\,c\,,(b\ c)\cdot t_1)) =_\alpha \mathtt{A}(\mathtt{V}\,c\,,(a\ c)\cdot t_2))$$
where $c \,\#\, (a, b, t_1, t_2)$

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

$$\mathtt{A}(\mathtt{V}\,c\,,(b\ c)\cdot t_1)) =_\alpha \mathtt{A}(\mathtt{V}\,c\,,(a\ c)\cdot t_2))$$
where $c \# (a, b, t_1, t_2)$

$\mathtt{V}\,c =_\alpha \mathtt{V}\,c$ and $(b\ c)\cdot t_1 =_\alpha (a\ c)\cdot t_2$
where $c \# (a, b, t_1, t_2)$

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$

(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

$$\mathtt{A}(\mathtt{V}\,c\,,(b\;c)\cdot t_1)) =_\alpha \mathtt{A}(\mathtt{V}\,c\,,(a\;c)\cdot t_2))$$

where $c \,\#\, (a, b, t_1, t_2)$

$\mathtt{V}\,c =_\alpha \mathtt{V}\,c$ and $(b\;c)\cdot t_1 =_\alpha (a\;c)\cdot t_2$

where $c \,\#\, (a, b, t_1, t_2)$

$$t_1 = (b\;c)(a\;c)\cdot t_2 [= (a\;b)(b\;c)\cdot t_2]$$

where $c \,\#\, (a, b, (a\;b)(b\;c)\cdot t_2\,, t_2)$

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with
$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$
(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

$$\mathtt{A}(\mathtt{V}\,c\,,(b\;c)\cdot t_1) =_\alpha \mathtt{A}(\mathtt{V}\,c\,,(a\;c)\cdot t_2)$$
where $c \# (a, b, t_1, t_2)$

$\mathtt{V}\,c =_\alpha \mathtt{V}\,c$ and $(b\;c)\cdot t_1 =_\alpha (a\;c)\cdot t_2$
where $c \# (a, b, t_1, t_2)$

$$t_1 = (b\;c)(a\;c)\cdot t_2[= (a\;b)(b\;c)\cdot t_2]$$
where $c \# (a, b, (a\;b)(b\;c)\cdot t_2\,, t_2)$

$$t_1 = (a\;b)(b\;c)\cdot t_2[= (a\;b)\cdot t_2]$$
where $c \# (a, b, t_2)$ and $b \# t_2$

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$

(where $a \neq b$)?

$$\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1)) =_\alpha \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2))$$

$$\mathtt{A}(\mathtt{V}\,c\,,(b\;c)\cdot t_1)) =_\alpha \mathtt{A}(\mathtt{V}\,c\,,(a\;c)\cdot t_2))$$

where $c \,\#\, (a, b, t_1, t_2)$

$\mathtt{V}\,c =_\alpha \mathtt{V}\,c$ and $(b\;c)\cdot t_1 =_\alpha (a\;c)\cdot t_2$

where $c \,\#\, (a, b, t_1, t_2)$

$$t_1 = (b\;c)(a\;c)\cdot t_2[= (a\;b)(b\;c)\cdot t_2]$$

where $c \,\#\, (a, b, (a\;b)(b\;c)\cdot t_2\,, t_2)$

$$t_1 = (a\;b)(b\;c)\cdot t_2[= (a\;b)\cdot t_2]$$

where $c \,\#\, (a, b, t_2)$ and $b \,\#\, t_2$

$t_1 = (a\;b)\cdot t_2$, for any t_2 with $b \,\#\, t_2$

Examples of unification 'mod α'

Ex. 1: does there exist a $t \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(t\,,\mathtt{V}\,b))) =_\alpha \mathtt{L}(b\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t)))$$

(where $a \neq b$)?

Ex. 2: do there exist $t_1, t_2 \in \Sigma(\mathtt{Term})$ with

$$\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,t_1))) =_\alpha \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,t_2)))$$

(where $a \neq b$)?

Can decide all such problems (over any nominal algebraic signature) using the nominal unification algorithm [Urban+AMP+Gabbay, TCS 323(2004)473–497] $\triangleq$ NOMU.

First, need to extend the synax of terms over a nominal signature with variables...

$\Sigma(\mathtt{S})$ = raw terms over Σ of sort $\mathtt{S}$

$$\frac{a \in \mathbb{A}}{a \in \Sigma(\mathtt{N})} \qquad \frac{t \in \Sigma(\mathtt{S}) \qquad \mathtt{op} : \mathtt{S} \to \mathtt{D}}{\mathtt{op}\, t \in \Sigma(\mathtt{D})} \qquad \frac{}{() \in \Sigma(\mathtt{1})}$$

$$\frac{t_1 \in \Sigma(\mathtt{S}_1) \qquad t_2 \in \Sigma(\mathtt{S}_2)}{t_1 \,,\, t_2 \in \Sigma(\mathtt{S}_1 \,,\, \mathtt{S}_2)} \qquad \frac{a \in \mathbb{A} \qquad t \in \Sigma(\mathtt{S})}{a \,.\, t \in \Sigma(\mathtt{N} \,.\, \mathtt{S})}$$

Each $\Sigma(\mathtt{S})$ is a nominal set once equipped with the obvious **Perm** $\mathbb{A}$-action—any finite set of atoms containing all those occurring in t supports $t \in \Sigma(\mathtt{S})$.

Open nominal terms

$()$	unit	a	atomic names
$t\,,t'$	pairs	$a\,.\,t$	abstractions
$\texttt{op}\;t$	constructed	$\pi * X$	suspensions

$\pi \in \mathbf{Perm}\,\mathbb{A}$

X ranges over variables, standing for unknown terms

E.g. $\texttt{L}(a\,.\,\texttt{A}(\texttt{V}\,c\,,(a\;c) * X))$

Equality & freshness

Equality of open terms is not just

$$t =_\alpha t' \qquad \alpha\text{-equivalence}$$

Equality & freshness

Equality is in general hypothetical

$\nabla \vdash t \approx t'$ hypothetical α-equivalence

finite set of freshness assumptions, $a \# X$, each with intended meaning: 'atomic name a will not occur freely in any term substituted for X'

Intended meaning:

'any closing substitution (= replacement of variables by terms) satisfying ∇ makes t and t' α-equivalent'

Equality & freshness

Equality is in general hypothetical

$$\nabla \vdash t \approx t' \qquad \text{hypothetical } \alpha\text{-equivalence}$$

Examples of valid judgements:

$$\{b \mathrel{\#} X\} \vdash a\,.\,X \approx b\,.\,((a\ b) * X)$$
$$\{a \mathrel{\#} X, b \mathrel{\#} X\} \vdash a\,.\,X \approx b\,.\,X$$

Equality & freshness

Also need freshness judgements

$$\nabla \vdash a \# t$$

Intended meaning:

> 'any closing substitution satisfying ∇ makes t not contain the atom a freely'

Examples of valid judgements:

$$\{b \# X\} \vdash b \# a\,.\,X$$
$$\{\ \} \vdash a \# a\,.\,X$$

Rules for $\boldsymbol{\nabla} \vdash \boldsymbol{t} \approx \boldsymbol{t'}$

Excerpt
(see NOMU, or NSB chapter 12, for full details)

Rules for $\nabla \vdash t \approx t'$

$$\frac{\nabla \vdash t \approx t'}{\nabla \vdash a\,.\,t \approx a\,.\,t'}$$

$$\frac{a \neq a' \qquad \nabla \vdash t \approx (a\ a')*t' \qquad \nabla \vdash a \,\#\, t'}{\nabla \vdash a\,.\,t \approx a'\,.\,t'}$$

Rules for $\nabla \vdash t \approx t'$

$$\frac{\nabla \vdash t \approx t'}{\nabla \vdash a\,.\,t \approx a\,.\,t'}$$

$$\frac{a \neq a' \qquad \nabla \vdash t \approx (a\ a')*t' \qquad \nabla \vdash a \# t'}{\nabla \vdash a\,.\,t \approx a'\,.\,t'}$$

$(a\ a') * t'$ is defined by recursion on the structure of t', pushing the swap down through the structure, applying it to atoms and stopping with subterms like $((a\ a') \circ \pi) * X$

Rules for $\nabla \vdash t \approx t'$

$$\frac{(a \,\#\, X) \in \nabla \text{ for all } a \text{ with } \pi(a) \neq \pi'(a)}{\nabla \vdash \pi * X \approx \pi' * X}$$

E.g.

$$\{a \,\#\, X, c \,\#\, X\} \vdash (a\ c)(a\ b) * X \approx (b\ c) * X$$

because

$$(a\ c)(a\ b) : \begin{array}{l} a \mapsto b \\ b \mapsto c \\ c \mapsto a \end{array} \qquad (b\ c) : \begin{array}{l} a \mapsto a \\ b \mapsto c \\ c \mapsto b \end{array}$$

disagree at a and c.

Rules for $\nabla \vdash a \# t$

(Excerpt)

$$\frac{a \neq a'}{\nabla \vdash a \# a'}$$

$$\frac{}{\nabla \vdash a \# a\,.\,t} \qquad \frac{a \neq a' \qquad \nabla \vdash a \# t}{\nabla \vdash a \# a'\,.\,t}$$

$$\frac{(\pi^{-1} a \# X) \in \nabla}{\nabla \vdash a \# \pi * X}$$

Correctness

[NOMU, Proposition 2.16]

Theorem. $\approx$ is an equivalence relation and agrees with $=_\alpha$ on ground terms: if t and t' contain no variables then

$$\emptyset \vdash t \approx t' \text{ is valid iff } t =_\alpha t'.$$

Furthermore

$$\emptyset \vdash a \not\approx t \text{ is valid iff } a \notin fn(t).$$

Substitution

Substitutions σ are finite maps from variables to terms, $[X_1 := t_1, \ldots, X_n := t_n]$.

Applying a substitution to a term: $\sigma\, t$ = result of replacing variables in t with terms according to σ, carrying out any permutations of atomic names that are generated.

E.g. if $\sigma = [X := \mathtt{A}(\mathtt{V}\, b\, , Y)]$, then

$$\begin{aligned}\sigma\,(\mathtt{L}(a\,.\,(a\ b) * X)) &= \mathtt{L}(a\,.\,(a\ b) * \mathtt{A}(\mathtt{V}\, b\, , Y)) \\ &= \mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\, a\, , (a\ b) * Y))\end{aligned}$$

Equational & freshness problems

An equational problem $\boxed{t\ {}_?{\approx}_?\ t'}$ is solved by

- a substitution σ, plus
- a set of freshness assumptions ∇

so that $\nabla \vdash \sigma\, t \approx \sigma\, t'$.

Equational & freshness problems

An equational problem $\boxed{t \;{}_?{\approx}_?\; t'}$ is solved by

- a substitution σ, plus
- a set of freshness assumptions ∇

so that $\nabla \vdash \sigma\, t \approx \sigma\, t'$.

Solving equations may entail solving freshness problems. E.g. assuming that $a \neq a'$, then $\mathtt{L}(a\,.\,t) \;{}_?{\approx}_?\; \mathtt{L}(a'\,.\,t')$ can only be solved if

$$t \;{}_?{\approx}_?\; (a\; a') * t' \quad \text{and} \quad a \;\#_?\; t'$$

can be solved.

Equational & freshness problems

An equational problem $\boxed{t \;{}_?{\approx}_?\; t'}$ is solved by

- a substitution σ, plus
- a set of freshness assumptions ∇

so that $\nabla \vdash \sigma\, t \approx \sigma\, t'$.

A freshness problem $\boxed{a \;\#_?\; t}$ is solved by

- a substitution σ, plus
- a set of freshness assumptions ∇

so that $\nabla \vdash a \;\#\; \sigma\, t$.

Existence of MGUs

Theorem. There is an algorithm which given any finite set P of equational and freshness problems (over any nominal algebraic signature), decides whether or not it has a solution (σ, ∇), and returns a most general one if it does.

straightforward definition, omitted

Existence of MGUs

Theorem. There is an algorithm which given any finite set $\boldsymbol{P}$ of equational and freshness problems (over any nominal algebraic signature), decides whether or not it has a solution (σ, ∇), and returns a most general one if it does.

Algorithm first reduces all the equations to 'solved form' (creating a substitution), possibly generating extra freshness problems, and then solves all the freshness problems (easy).

(See [NOMU, Sect. 3].)

$$\{\mathtt{L}(a.\mathtt{L}(b.\mathtt{A}(\mathtt{V}\,b, X))) \;{}_?{\approx}_?\; \mathtt{L}(a.\mathtt{L}(a.\mathtt{A}(\mathtt{V}\,a, Y)))\}$$

$$(\xrightarrow{id})^3 \quad \{b.\mathtt{A}(\mathtt{V}\,b, X) \;{}_?{\approx}_?\; a.\mathtt{A}(\mathtt{V}\,a, Y)\}$$

$$\xrightarrow{id} \quad \{\mathtt{A}(\mathtt{V}\,b, X) \;{}_?{\approx}_?\; \mathtt{A}(\mathtt{V}\,b, (b\ a) * Y),\ b \;\#_?\; \mathtt{A}(\mathtt{V}\,a, Y)\}$$

$$(\xrightarrow{id})^3 \quad \{X \;{}_?{\approx}_?\; (b\ a) * Y,\ b \;\#_?\; \mathtt{A}(\mathtt{V}\,a, Y)\}$$

$$\xrightarrow{[X:=(b\ a)*Y]} \{b \;\#_?\; \mathtt{A}(\mathtt{V}\,a, Y)\}$$

$$\xrightarrow{\varnothing} \quad \{b \;\#_?\; \mathtt{V}\,a,\ b \;\#_?\; Y\}$$

$$(\xrightarrow{\varnothing})^2 \quad \{b \;\#_?\; Y\}$$

$$\xrightarrow{\{(b\#Y)\}} \quad \varnothing$$

$$\{\mathtt{L}(a\,.\,\mathtt{L}(b\,.\,\mathtt{A}(\mathtt{V}\,b\,,X)))\ {}_?{\approx}_?\ \mathtt{L}(a\,.\,\mathtt{L}(a\,.\,\mathtt{A}(\mathtt{V}\,a\,,Y)))\}$$

$$(\overset{id}{\rightarrow})^3 \quad \{b\,.\,\mathtt{A}(\mathtt{V}\,b\,,X)\ {}_?{\approx}_?\ a\,.\,\mathtt{A}(\mathtt{V}\,a\,,Y)\}$$

$$\overset{id}{\rightarrow} \quad \{\mathtt{A}(\mathtt{V}\,b\,,X)\ {}_?{\approx}_?\ \mathtt{A}(\mathtt{V}\,b\,,(b\ a)*Y),\ b\ \#_?\ \mathtt{A}(\mathtt{V}\,a\,,Y)\}$$

$$(\overset{id}{\rightarrow})^3 \quad \{X\ {}_?{\approx}_?\ (b\ a)*Y,\ b\ \#_?\ \mathtt{A}(\mathtt{V}\,a\,,Y)\}$$

$$\overset{[X:=(b\ a)*Y]}{\rightarrow} \{b\ \#_?\ \mathtt{A}(\mathtt{V}\,a\,,Y)\}$$

$$\overset{\varnothing}{\rightarrow} \quad \{b\ \#_?\ \mathtt{V}\,a,\ b\ \#_?\ Y\}$$

$$(\overset{\varnothing}{\rightarrow})^2 \quad \{b\ \#_?\ Y\}$$

$$\overset{\{(b\#Y)\}}{\rightarrow} \quad \varnothing$$

most general solution $= [X := (b\ a)*Y], \{(b\ \#\ Y)\}$

Existence of MGUs

Theorem. There is an algorithm which given any finite set P of equational and freshness problems (over any nominal algebraic signature), decides whether or not it has a solution (σ, ∇), and returns a most general one if it does.

- Current best NOMU algorithm is quadratic [Levy & Villaret, Proc. RTA 2010].
- NOMU is (quadratically) inter-reducible with Dale Miller's higher-order pattern unification, which uses variables that depend on names $X(a_1, \ldots, a_n)$ rather than NOMU's variables that are fresh for names ($\{a_1, \ldots, a_n\} \# X$). (Higher-order patterns form a subset of Church's simply typed λ-calculus.)

Other applications of nominal sets

- **Computational logic**
  - Higher-order logic: Urban & Berghofer's Nominal package for the interactive theorem-prover Isabelle/HOL.
  - Equational logic: rewriting for nominal terms [Fernandez+Gabbay+Calves+$\cdots$]

Other applications of nominal sets

- **Computational logic**
  - Higher-order logic: Urban & Berghofer's Nominal package for the interactive theorem-prover Isabelle/HOL.
  - Equational logic: rewriting for nominal terms [Fernandez+Gabbay+Calves+· · ·]
- **Automata theory & verification**
  - HD-automata [Montanari el al]
  - fresh-register automata [Tzevelekos]
  - orbit-finite computation theory [Bojańczyk et al]

Other applications of nominal sets

- **Homotopy Type Theory** (HoTT)

  Cubical sets [Bezem-Coquand-Huber] model of Voevodsky's axiom of univalence makes use of nominal sets equipped with an operation of substitution $x \mapsto x(i/a)$ where $i \in \{0,1\}$.

  - names are names of directions (cartesian axes)
    (so e.g., if an object has support $\{a, b, c\}$ it is 3-dimensional)
  - freshness $(a \# x)$ = degeneracy $(x(i/a) = x)$
  - identity types are modelled by name-abstraction: $\langle a \rangle x$ is a proof that $x(0/a)$ is equal to $x(1/a)$.

HoTT and univalence is about (computable) *mathematical foundations* (a topic no longer very popular with mathematicians). That's where the mathematics of nominal sets came from...

Impact can take a long time

The mathematics behind nominal sets goes back a long way...

Abraham Fraenkel, *Der Begriff "definit" und die Unabhängigkeit des Auswahlsaxioms*, Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse (1922), 253–257.

Andrzej Mostowski, *Uber die Unabhängigkeit des Wohlordnungssatzes vom Ordnungsprinzip*, Fundamenta Mathematicae 32 (1939), 201–252.

Impact can take a long time

The mathematics behind nominal sets goes back a long way...

...and it's still too early to tell what will be the impact of the applications of it to CS developed over the last 15 years.