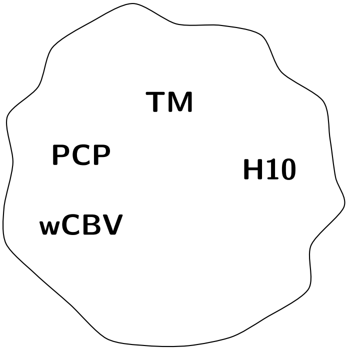
Undecidability of Higher-Order Unification Formalised in Coq

Simon Spies, Yannick Forster

20 January 2020
CPP'20

SIC Saarland Informatics Campus
SAARLAND UNIVERSITY
COMPUTER SCIENCE
UNIVERSITY OF CAMBRIDGE

Extending the Coq Library of Undecidable Problems



Library of Undecidable Problems[1]

+

U
Higher-Order Unification

[1] For an overview of the library see the talk at 4pm on Saturday at CoqPL

Higher-Order Unification in Action

Emacs-x86_64-10_14 Prelude - ~/Projects/playground/natind.v

```
Lemma plus_zero : ∀ n, n + 0 = n.
Proof.
  Check nat_ind.
  apply nat_ind with (P := λ n ⇒ n + 0 = n).
  reflexivity.
  cbn; now intros ? ->.
Qed.
```

```
-:---  natind.v      29% of 571  (11,0)     (Coq Script(1-) +2 company-    yas hs Holes Outl FlyC company i
```

```
U:%%-  *response*    All of 0    (1,0)      (Coq Response +2 FlyC company ivy EditorConfig Projectile supe
```

Higher-Order Unification in Action

Emacs-x86_64-10_14 Prelude - ~/Projects/playground/natind.v

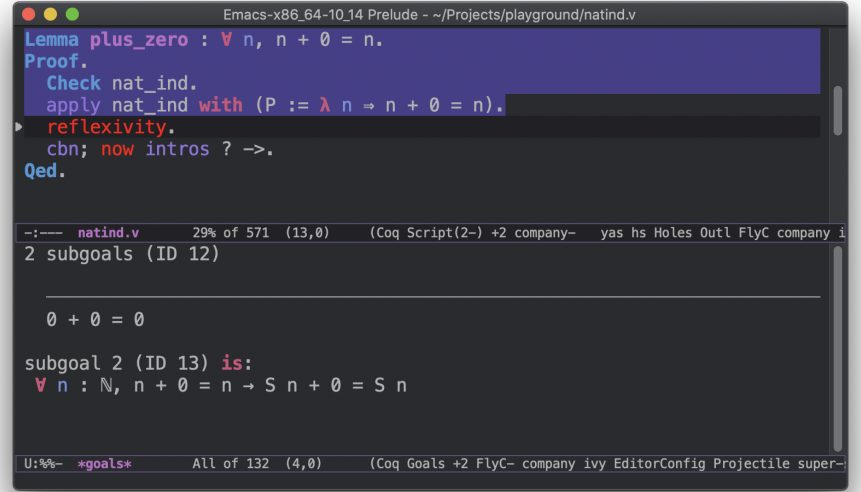
```
Lemma plus_zero : ∀ n, n + 0 = n.
Proof.
  Check nat_ind.
  apply nat_ind with (P := λ n ⇒ n + 0 = n).
  reflexivity.
  cbn; now intros ? ->.
Qed.
```

```
-:---  natind.v      29% of 571  (12,0)     (Coq Script(1-) +2 company-    yas hs Holes Outl FlyC company i
```

```
nat_ind
     : ∀ P : ℕ → ℙ, P 0 → (∀ n : ℕ, P n → P (S n)) → ∀ n : ℕ, P n
```

```
U:%%-  *response*    All of 102  (2,65)     (Coq Response +2 FlyC company ivy EditorConfig Projectile supe
```

Higher-Order Unification in Action



Higher-Order Unification in Action

```
Emacs-x86_64-10_14 Prelude - ~/Projects/playground/natind.v
Lemma plus_zero : ∀ n, n + 0 = n.
Proof.
  Check nat_ind.
  apply nat_ind.
  reflexivity.
  cbn; now intros ? ->.
Qed.

-:---  natind.v       30% of 540  (13,0)     (Coq Script(2-) +2 company-   yas hs Holes Outl FlyC company i
2 subgoals (ID 9)

  ____________________________________________________________
  0 + 0 = 0

subgoal 2 (ID 10) is:
 ∀ n : ℕ, n + 0 = n → S n + 0 = S n

U:%%-  *goals*        All of 131  (4,0)      (Coq Goals +2 FlyC- company ivy EditorConfig Projectile super-
Wrote /Users/spies/Projects/playground/natind.v
```

Higher-Order Unification in Theory

Instance

$$P : \mathbb{N} \to \mathbb{P} \vdash (\forall n.\ P\ n) \stackrel{?}{=} (\forall n.\ n + 0 = n) : \mathbb{P}$$

containing constants, bound variables, free variables, and types.

Higher-Order Unification in Theory

Instance

$$P : \mathbb{N} \to \mathbb{P} \vdash (\forall n.\ P\ n) \overset{?}{=} (\forall n.\ n + 0 = n) : \mathbb{P}$$

containing constants, bound variables, free variables, and types.

Solution The substitution

$$(\lambda n. n + 0 = n)/P$$

since

$$(\forall n.\ (\lambda n. n + 0 = n)\ n) \equiv_\beta (\forall n.\ n + 0 = n)$$

Why is this hard?

Emacs-x86_64-10_14 Prelude - ~/Projects/playground/natind.v

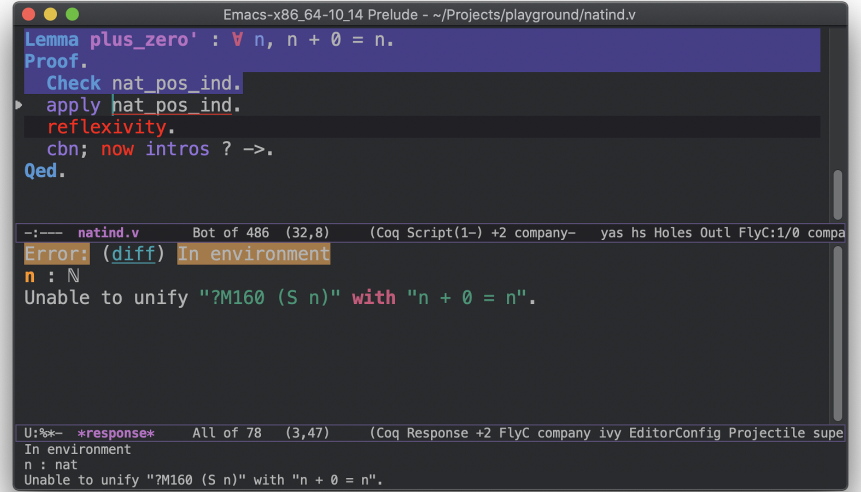
```
Lemma plus_zero' : ∀ n, n + 0 = n.
Proof.
  Check nat_pos_ind.
  apply nat_pos_ind.
  reflexivity.
  cbn; now intros ? ->.
Qed.
```

```
-:---  natind.v       Bot of 486   (32,0)      (Coq Script(1-) +2 company-    yas hs Holes Outl FlyC:1/0 compa
```

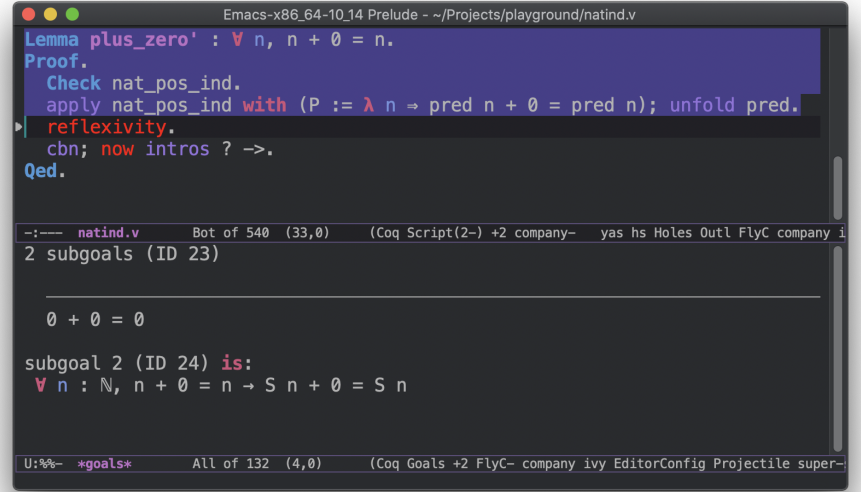
```
nat_pos_ind
     : ∀ P : ℕ → ℙ,
       P 1 → (∀ n : ℕ, P (S n) → P (S (S n))) → ∀ n : ℕ, P (S n)
```

```
U:%%-  *response*     All of 125   (3,64)      (Coq Response +2 FlyC company ivy EditorConfig Projectile supe
```

Why is this hard?



Why is this hard?



Overview

Overview

Huet (1973)

$$\mathbf{PCP} \preceq \mathbf{U}_{3+k}$$

PCP Post-correspondence problem

Overview

Huet (1973)

$\textsf{PCP} \preceq \textsf{U}_{3+k}$

Goldfarb (1981)

$\textsf{H10} \preceq \textsf{U}_2$

H10 Hilbert's tenth problem **PCP** Post-correspondence problem

Overview

Huet (1973)

$\mathsf{PCP} \preceq \mathsf{U}_{3+k}$

Goldfarb (1981)

$\mathsf{H10} \preceq \mathsf{U}_2$

Dowek (2001)

$\mathsf{H10} \preceq \mathsf{U}$

H10 Hilbert's tenth problem

PCP Post-correspondence problem

Overview

Huet (1973)

$\mathsf{PCP} \preceq \mathsf{U}_{3+k}$

Goldfarb (1981)

$\mathsf{H10} \preceq \mathsf{U}_2$

Dowek (2001)

$\mathsf{H10} \preceq \mathsf{U}$

Our Contributions

1. Coq formalisation
2. Simplification of Goldfarb's proof
3. Simplification of Huet's proof
4. Same Calculus

H10 Hilbert's tenth problem **PCP** Post-correspondence problem

Simply-Typed λ-Calculus

$$s, t ::= x \mid c \mid \lambda x.s \mid s\ t \qquad (c : \mathcal{C})$$
$$A, B ::= \alpha \mid A \to B$$
$$\Gamma, \Delta ::= x_1 : A_1, \ldots, x_n : A_n$$

Equality:	β-equivalence	$s \equiv_\beta t$
Substitution:	capture-avoiding	$s[\sigma]$
Typing:	Curry-style	$\Gamma \vdash s : A$

Higher-Order Unification

$$\mathbf{U}\ (\Gamma \vdash s \stackrel{?}{=} t : A)$$

Higher-Order Unification

$$\mathbf{U}\ (\Gamma \vdash s \stackrel{?}{=} t : A) :=$$

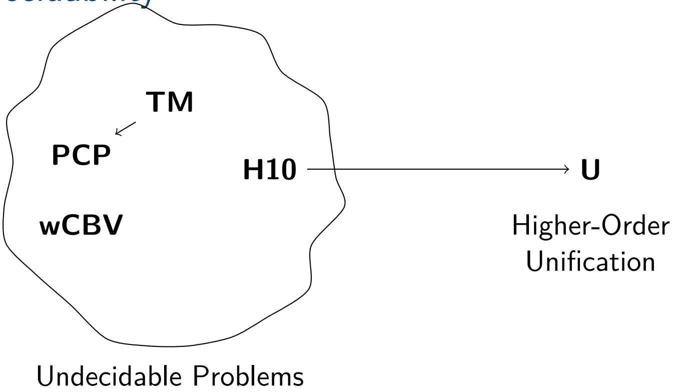
$$\exists \sigma \qquad s[\sigma] \equiv_\beta t[\sigma]$$

Higher-Order Unification

$$\mathbf{U}\ (\Gamma \vdash s \stackrel{?}{=} t : A) :=$$
$$\exists \sigma \Delta.\ \Delta \vdash \sigma : \Gamma \quad \text{and} \quad s[\sigma] \equiv_\beta t[\sigma]$$

where $\Delta \vdash \sigma : \Gamma := \forall (x : A) \in \Gamma.\ \Delta \vdash \sigma x : A$

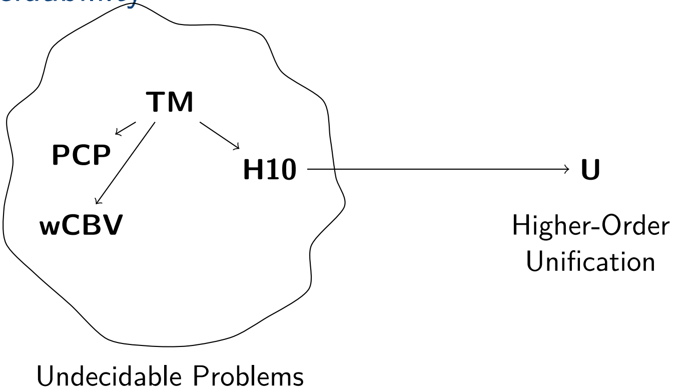
Traditional Undecidability



P undec. iff there is no TM deciding **P**

P $\preceq$ **Q** iff there is a TM computable function f such that $\forall x.$ **P**(x) iff **Q**$(f(x))$

Synthetic Undecidabililty



P undec.	iff	$\mathbf{TM} \preceq \mathbf{P}$
$\mathbf{P} \preceq \mathbf{Q}$	iff	there is a Coq function f such that
		$\forall x.\ \mathbf{P}(x)$ iff $\mathbf{Q}(f(x))$

Reduction

$$\mathbf{H10} \quad \preceq \quad \mathbf{U}$$

Reduction

$$\textbf{H10} \preceq \textbf{SU} \preceq \textbf{U}$$

$$\textbf{SU}\ (\{\Gamma \vdash s_i \stackrel{?}{=} t_i : A_i \mid i = 1, \ldots, n\}) :=$$
$$\exists \sigma \Delta.\ \Delta \vdash \sigma : \Gamma \quad \text{and} \quad \forall i.\ s_i[\sigma] \equiv_\beta t_i[\sigma]$$

Hilbert's tenth problem

Diophantine Equations

$$
\begin{aligned}
d ::= &\; x \doteq 1 \\
\mid &\; x + y \doteq z \\
\mid &\; x \cdot y \doteq z
\end{aligned}
$$

$$
\begin{aligned}
\theta \vDash x \doteq 1 \quad &\text{iff} \quad \theta x = 1 \\
\theta \vDash x + y \doteq z \quad &\text{iff} \quad \theta x + \theta y = \theta z \\
\theta \vDash x \cdot y \doteq z \quad &\text{iff} \quad \theta x \cdot \theta y = \theta z
\end{aligned}
$$

Hilbert's tenth problem

$$\mathbf{H10}(D) := \exists\theta.\forall d \in D.\ \theta \vDash d$$

Diophantine Equations

$$d ::= x \doteq 1 \mid x + y \doteq z \mid x \cdot y \doteq z$$

$$\begin{aligned} \theta \vDash x \doteq 1 &\quad \text{iff} \quad \theta x = 1 \\ \theta \vDash x + y \doteq z &\quad \text{iff} \quad \theta x + \theta y = \theta z \\ \theta \vDash x \cdot y \doteq z &\quad \text{iff} \quad \theta x \cdot \theta y = \theta z \end{aligned}$$

H10 $\to$ **SU** following Dowek (2001)

$$\mathbf{H10} \preceq \mathbf{SU}$$

H10 → SU following Dowek (2001)

$$\textbf{H10}(D) \quad \text{iff} \quad \textbf{SU}(f(D))$$

H10 $\to$ SU following Dowek (2001)

$$\mathbf{H10}(D) \quad \text{iff} \quad \mathbf{SU}(f(D))$$

where f is given by

$$\begin{aligned} f(x \doteq 1) &:= x \stackrel{?}{=} [\![1]\!]_{\mathsf{cn}} \\ f(x + y \doteq z) &:= x \oplus y \stackrel{?}{=} z \\ f(x \cdot y \doteq z) &:= x \otimes y \stackrel{?}{=} z \end{aligned}$$

and for every variable x a characteristic equation CN x.

Church Numerals

$[\![n]\!]_{\mathsf{cn}} := \lambda a f. f^n\ a$ $\oplus$ faithful $\otimes$ faithful

Fragments

$$x \oplus y \stackrel{?}{=} z \qquad\qquad \lambda a f. x\ (y\ a\ f)\ f \stackrel{?}{=} z$$

where

$$x, y, z : \alpha \to (\alpha \to \alpha) \to \alpha$$

Fragments

Third-Order Unification

$$x \oplus y \stackrel{?}{=} z \qquad\qquad \lambda a f. x\ (y\ a\ f)\ f \stackrel{?}{=} z$$

where

$$x, y, z : \alpha \to (\alpha \to \alpha) \to \alpha$$

Fragments

First-Order Unification

$$\mathsf{g}\ u\ \mathsf{a} \stackrel{?}{=} \mathsf{g}\ \mathsf{a}\ v \qquad \mathsf{g}\ \mathsf{a}\ v \stackrel{?}{=} w \qquad \mathsf{g}\ u\ \mathsf{a} \stackrel{?}{=} u$$

Third-Order Unification

$$x \oplus y \stackrel{?}{=} z \qquad \lambda a f. x\ (y\ a\ f)\ f \stackrel{?}{=} z$$

where

$$\mathsf{g} : \alpha \to \alpha \to \alpha \qquad \mathsf{a} : \alpha \qquad x, y, z : \alpha \to (\alpha \to \alpha) \to \alpha \qquad u, v, w : \alpha$$

Fragments

First-Order Unification

$$\mathsf{g}\ u\ \mathsf{a} \stackrel{?}{=} \mathsf{g}\ \mathsf{a}\ v \qquad \mathsf{g}\ \mathsf{a}\ v \stackrel{?}{=} w \qquad \mathsf{g}\ u\ \mathsf{a} \stackrel{?}{=} u$$

Second-Order Unification

$$\mathsf{g}\ \mathsf{a} \stackrel{?}{=} h\ \mathsf{a} \qquad h\ \mathsf{a} \stackrel{?}{=} h\ (h\ \mathsf{a})$$

Third-Order Unification

$$x \oplus y \stackrel{?}{=} z \qquad \lambda a f. x\ (y\ a\ f)\ f \stackrel{?}{=} z$$

where

$$\mathsf{g} : \alpha \to \alpha \to \alpha \qquad \mathsf{a} : \alpha \qquad x, y, z : \alpha \to (\alpha \to \alpha) \to \alpha \qquad u, v, w : \alpha \qquad h : \alpha \to \alpha$$

Nth-Order Unification

$$\mathbf{U}_n\ (\Gamma \vdash_n s \overset{?}{=} t : A) :=$$
$$\exists \sigma \Delta.\ \Delta \vdash_n \sigma : \Gamma \quad \text{and} \quad s[\sigma] \equiv_\beta t[\sigma]$$

where $\Delta \vdash_n \sigma : \Gamma := \forall (x : A) \in \Gamma.\ \Delta \vdash_n \sigma x : A$

Conservativity

Conservativity

$$\mathbf{U}_n \preceq_{\mathsf{id}} \mathbf{U}_{n+k} \preceq_{\mathsf{id}} \mathbf{U} \quad \text{for } n \geq 1, k \geq 0$$

Corollary

$$\mathbf{U}_1 \preceq_{\mathsf{id}} \mathbf{U}_2 \preceq_{\mathsf{id}} \mathbf{U}_{2+k} \preceq_{\mathsf{id}} \mathbf{U}$$

Second-Order Undecidability following Goldfarb (1981)

$$\mathbf{H10} \preceq \mathbf{U}_2$$

with constants $\mathsf{g} : \alpha \to \alpha \to \alpha$ and $\mathsf{a} : \alpha$.

Goldfarb Numerals

$\boxed{[\![n]\!]_{\mathsf{cn}}}$

$$[\![n]\!]_{\mathsf{cn}} := \lambda a f.f^n\ a$$

to

$\boxed{[\![n]\!]_{\mathsf{gn}}}$

$$[\![n]\!]_{\mathsf{gn}} := \lambda a.\mathsf{S}^n\ a$$

where $\mathsf{S} := \mathsf{g}\ \mathsf{a}$ with $\mathsf{g} : \alpha \to \alpha \to \alpha$ and $\mathsf{a} : \alpha$

Goldfarb Numerals

$\boxed{[\![n]\!]_{\mathsf{cn}}}$

$$[\![n]\!]_{\mathsf{cn}} := \lambda a f. f^n\ a$$

to

$\boxed{[\![n]\!]_{\mathsf{gn}}}$

$$[\![n]\!]_{\mathsf{gn}} := \lambda a. \mathsf{S}^n\ a$$

where $\mathsf{S} := \mathsf{g}\ \mathsf{a}$ with $\mathsf{g} : \alpha \to \alpha \to \alpha$ and $\mathsf{a} : \alpha$

Operations

- ✓ addition
- ✓ characteristic equation
- ✗ multiplication

$$s \otimes t := \lambda a f. \underbrace{s\ a\ (\lambda b. t\ b\ f)}_{\text{3rd-order}}$$

Multiplication

Following Goldfarb (1981), the equation

$$x \cdot y \doteq z$$

is encoded as

$$\begin{aligned} & \lambda uv.G_{xyz}\ (\mathsf{g}\ (\mathsf{g}\ (z\ u)\ (x\ v))\ \mathsf{a})\ u\ v \\ \stackrel{?}{\doteq}\ & \lambda uv.\mathsf{g}\ (\mathsf{g}\ u\ v)\ (G_{xyz}\ \mathsf{a}\ (y\ u)\ (\mathsf{S}\ v)) \end{aligned}$$

where $G_{xyz} : \alpha \to \alpha \to \alpha \to \alpha$ and $x, y, z : \alpha \to \alpha$

Why? Explanation in the paper.

Contributions in the context of the library

$$\textbf{TM} \preceq \textbf{H10} \preceq \textbf{U}_2 \preceq \textbf{U}_{2+k} \preceq \textbf{U}$$

Library | Goldfarb | Conservativity

Recall

P undec. iff **TM** $\preceq$ **P**

P $\preceq$ **Q** iff there is a Coq function f such that $\forall x.\ \textbf{P}(x)$ iff $\textbf{Q}(f(x))$

Furthermore. . .

- First-Order Unification

  $\mathbf{U}_1$ is decidable

- Simplifying Huet (1973)

  $\mathbf{PCP} \preceq \mathbf{U}_3$ simplified to $\mathbf{MPCP} \preceq \mathbf{U}_3$

- Techniques for treating constants similar to Statman (1981)

$$\mathbf{U}_2^{\{\mathrm{g},\mathrm{a}\}} \preceq \mathbf{U}_2^{\{\mathrm{g}\}} \preceq \mathbf{U}_3^{\{\mathrm{g}\}} \preceq \mathbf{U}_3^{\emptyset}$$

- $\mathbf{U}, \mathbf{SU}, \mathbf{U}_n,$ and $\mathbf{SU}_n$ are enumerable

Future Work

- Decidability of monadic 2nd-order unification; Farmer (1988)
- Huet's unification procedure; Huet (1975)

Formalisation

Details

- De Bruijn indices
- Normalisation for the STLC
- Constant Replacement
- Meta Theory of the STLC

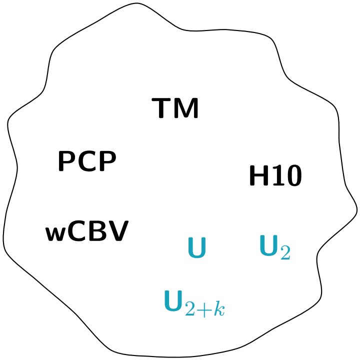
Coq Code

Unification	3000
Undecidability	450
Second-Order	1000
Other	3000
Total	7450

Tools

- *Autosubst 2* used for generating terms and substitution
- *Equations* used for defining first-order unification algorithm
- *Setoid Rewriting* used for reasoning about β-equivalence

Coq Library of Undecidable Problems



+

???
Your Contribution

Library under uds-psl on Github

and *4pm on Saturday at CoqPL*

References I

Dowek, G.
2001. Higher-order unification and matching. *Handbook of automated reasoning*, 2:1009–1062.

Farmer, W. M.
1988. A unification algorithm for second-order monadic terms. *Annals of Pure and applied Logic*, 39(2):131–174.

Forster, Y., D. Kirst, and G. Smolka
2019. On synthetic undecidability in Coq, with an application to the Entscheidungsproblem. In *International Conference on Certified Programs and Proofs*.

Goldfarb, W. D.
1981. The undecidability of the second-order unification problem. *Theoretical Computer Science*, 13:225–230.

References II

Huet, G. P.
1973. The undecidability of unification in third order logic. *Information and control*, 22(3):257–267.

Huet, G. P.
1975. A unification algorithm for typed λ-calculus. *Theoretical Computer Science*, 1(1):27–57.

Snyder, W. and J. H. Gallier
1989. Higher order unification revisited: Complete sets of transformations. *Technical Reports (CIS)*, P. 778.

Statman, R.
1981. On the existence of closed terms in the typed λ calculus II: Transformations of unification problems. *Theoretical Computer Science*, 15(3):329–338.

Characteristic Equation

Iteration fulfills

$$f^n(fa) = f(f^n a)$$

We can show: Let s be normal.

$$\lambda af.s\ (f\ a)\ f \equiv_\beta \lambda af.f\ (s\ a\ f) \quad \text{iff} \quad s = [\![n]\!]_{\mathsf{cn}} \quad \text{for some } n : \mathbb{N}$$

where $[\![n]\!]_{\mathsf{cn}} := \lambda af.f^n\ a$.

Characteristic Equation

$$\mathsf{CN}\ x := \lambda af.x\ (f\ a)\ f \stackrel{?}{=} \lambda af.f\ (x\ a\ f)$$

SU $\preceq$ U

$$\textbf{SU}(E) \quad \text{iff} \quad \textbf{U}(f(E))$$

Proof.

Pick $f := \{\Gamma \vdash s_i \stackrel{?}{=} t_i : A_i \mid i = 1, \ldots, n\} \mapsto$

$$\Gamma \vdash \lambda h.h\ s_1 \cdots s_n \stackrel{?}{=} \lambda h.h\ t_1 \cdots t_n : A$$

where $A = (A_1 \to \cdots \to A_n \to \alpha) \to \alpha$. Follows with:

$$h\ u_1 \cdots u_n \equiv_\beta h\ v_1 \cdots v_n \ \text{ iff } \ \forall i.\ u_i \equiv_\beta v_i$$

Multiplication

Multiplication sequence

$$(0,0),(n,1),(2n,2),\cdots,(m\cdot n,m)$$

generated by

$$m\cdot n=p \quad \text{iff} \quad \exists X.\ (0,0) :: \mathsf{map\ step}\ X = X +\!\!+ [(p,m)]$$

where $\mathsf{step}(a,i) := (a+n, i+1)$.

Modified Post Correspondence Problem — **MPCP**

Given

$$\boxed{\frac{l_0}{r_0}} \quad \text{and} \quad \boxed{\frac{l_1}{r_1}} \cdots \boxed{\frac{l_n}{r_n}}$$

$$\textcircled{0} \qquad\qquad \textcircled{1} \qquad \textcircled{n}$$

Find Ordering

$$i_1, \ldots, i_k$$

Such that

$$l_0 l_{i_1} \cdots l_{i_k} = r_0 r_{i_1} \cdots r_{i_k}$$

Simplification of Huet's Proof

Original

$$\lambda u_1 u_0 h.h\ (f\ \overline{l_0} \cdots \overline{l_n})\ (f\ u_1 \cdots u_1) \stackrel{?}{=} \lambda u_1 u_0 h.h\ (f\ \overline{l_0} \cdots \overline{l_n})\ (u_1\ (d\ u_1))$$

where $f : (\alpha \to \alpha)^{n+1} \to \alpha \to \alpha$ and $d : (\alpha \to \alpha) \to \alpha$.

vs.

Simplification

$$\lambda u_1 u_0.\overline{l_0}\ (f\ \overline{l_0}\ \cdots\ \overline{l_n}) \stackrel{?}{=} \lambda u_1 u_0.\overline{r_0}\ (f\ \overline{r_0}\ \cdots\ \overline{r_n})$$

where $f : (\alpha \to \alpha)^{n+1} \to \alpha \to \alpha$.

Meta Theory of STLC

Small Challenges

$$\textit{If } h\ s \equiv_\beta h\ t,\ \textit{then } s \equiv_\beta t.$$

- *If* $s \succ s'$ *and* $\text{isLam}(\text{head } s')$ *then* $\text{isLam}(\text{head } s)$.
- *If* $s\ t \succ^* v$ *then* $s \succ^* s', t \succ^* t'$, *and* $v = s'\ t'$ *for some* s', t'
  or $s \succ^* \lambda x.s'$ *and* $\text{isLam}\ (\text{head } s)$ *for some* s'.
- *If* $s_1\ s_2 \equiv_\beta t_1\ t_2$, $\text{isVar}\ (\text{head } s_1)$, *and* $\text{isVar}\ (\text{head } t_1)$, *then* $s_1 \equiv_\beta t_1$ *and* $s_2 \equiv_\beta t_2$.

List Operations

$\boxed{S\ t}$	$\boxed{s\ T}$	$\boxed{\Lambda X.\ s}$
$\text{nil}\ t = t$	$s\ \text{nil} = s$	$\Lambda\text{nil}.\ s = s$
$(s :: S)\ t = s\ (S\ t)$	$s\ (t :: T) = (T\ s)\ t$	$\Lambda x :: X.\ s = \lambda x.\Lambda X.\ s$

Conservativity — $\mathbf{U}_n \subseteq \mathbf{U}$

Let $\Gamma \vdash_n s \overset{?}{=} t : A$.

$$s[\sigma] \equiv_\beta t[\sigma] \text{ for some } \Sigma \vdash_n \sigma : \Gamma$$
$$\text{iff}$$
$$s[\sigma] \equiv_\beta t[\sigma] \text{ for some } \Delta \vdash \sigma : \Gamma$$

Proof Sketch.
Replace free variables and constants not of order n with first-order terms. For example, $x : (\alpha \to \alpha) \to \alpha$ is replaced by $\lambda x_1.z$ where $z : \alpha$ and $\mathsf{g} : \alpha \to \alpha \to \alpha$ is replaced by $\lambda x_1 x_2.z$. Normalise the result. □

Adding Constants

$$\mathbf{U}_n^{\mathcal{C}} \preceq \mathbf{U}_n^{\mathcal{D}} \quad \text{if } \mathcal{C} \subseteq \mathcal{D}$$

Proof Sketch.
Replace constants $d \in \mathcal{D} - \mathcal{C}$ with first-order terms, see conservativity. □

Removing Constans

$$\mathbf{U}_n^{\mathcal{D}} \preceq \mathbf{U}_n^{\mathcal{C}} \quad \text{if } \mathcal{C} \subseteq \mathcal{D} \text{ and } \forall d \notin C.\ \text{ord}\ (\Omega d) < n$$

Proof Sketch.
Let $\mathcal{C} = \{\mathsf{g}\}$ and $\mathcal{D} = \{\mathsf{a}, \mathsf{g}\}$.

$$\begin{array}{lcl} \mathsf{g}\ x \stackrel{?}{=} \mathsf{g}\ \mathsf{a} & & \lambda x_{\mathsf{a}}.\mathsf{g}\ (x\ x_{\mathsf{a}}) \stackrel{?}{=} \lambda x_{\mathsf{a}}.\mathsf{g}\ x_{\mathsf{a}} \\ & \rightsquigarrow & \\ \text{where} \quad x : \alpha & & \text{where} \quad x : \alpha \to \alpha \end{array}$$

□