A Special Case of Schematic Syntactic Unification

David M. Cerna

December 8^{th} 2021

JOHANNES KEPLER
UNIVERSITY LINZ

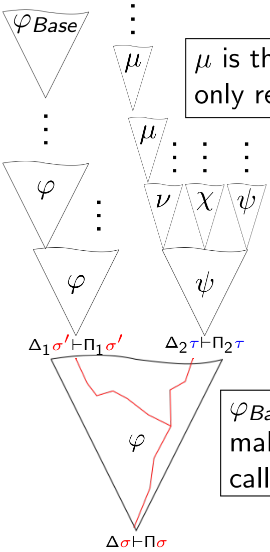
Czech Academy
of Sciences

Motivation

- Unification of term sequences (term schemata) is essential for automated reasoning driven inductive proof analysis.
- Proof analysis is removal of auxiliary lemmata from proofs.
- An interactive analysis of Fürstenburg's proof of the infinitude of primes was performed using a rudimentary schematic formalism [Baaz *et al.*, 2008].
- A formal framework for working with schematic proofs and term schemata did not exists at the time of this earlier work.
- Here we address the unification problem presented in our recent publication on the subject

  "Schematic Refutations of Formula Schemata", David M. Cerna, Alexander Leitsch, and Anela Lolic, 2021

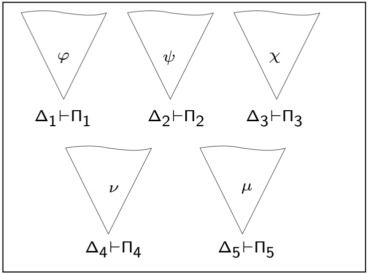
Motivation: Schematic Proofs in a Nutshell



μ is the least proof and only references itself.

Nested proof calls allowed.

φ*Base* cannot make proof calls.

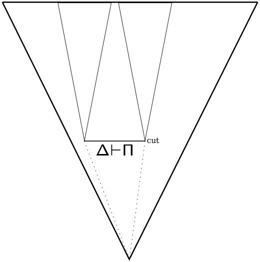


φ has free variables instantiated by σ to numerals.

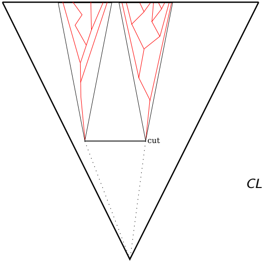
$\sigma > \sigma'$ but no order relation between σ and τ.

φ cannot be referenced in ψ, and ψ cannot be referenced in ν, etc.

Motivation: Lemmata (Cuts) as Recursive Formulas



Proof with cuts



Paths to cut ancestors

$$CL(A \vdash A) \equiv \{A\}$$
$$CL(A \vdash A) \equiv \{\neg A\}$$
$$CL(A \vdash A) \equiv \{\neg A \vee A\}$$

$$CL\left(\frac{\Delta \vdash \Pi}{\Delta' \vdash \Pi'}\ \rho\right) \equiv CL(\Delta \vdash \Pi)$$

$$CL\left(\frac{\Delta \vdash \Pi \qquad \Delta' \vdash \Pi'}{\Delta'' \vdash \Pi''}\ \rho\right) \equiv$$

$$\begin{cases} CL(\Delta \vdash \Pi) \wedge CL(\Delta' \vdash \Pi') \\ CL(\Delta \vdash \Pi) \vee CL(\Delta' \vdash \Pi') \end{cases}$$

- Proof references are denoted by defined symbols.
- The recursive formula is always unsatisfiable.
- Analysis requires refuting in a finitely representable way.
- This implies schematic unification.

Motivation: Example Extracted Formula

$$
\begin{aligned}
\hat{O}(x,y,n,m) &\Longrightarrow \hat{D}(x,n,m) \wedge\ \hat{P}(x,y,n,m)\\
\hat{D}(x,n,0) &\Longrightarrow f(x) = \hat{S}(n,a)\ \vee\ f(x) < \hat{S}(n,a)\\
\hat{D}(x,n,s(m)) &\Longrightarrow f(\hat{S}(s(m),x) = \hat{S}(n,a)\ \vee\ f(x) < \hat{S}(n,a)) \wedge\ \hat{D}(x,n,m)\\
\hat{P}(x,y,0,m) &\Longrightarrow \hat{C}(y,0,m)\ \wedge\ f(a) \not< 0\\
\hat{P}(x,y,s(n),m) &\Longrightarrow (\hat{C}(y,s(n),m))\ \wedge (\hat{T}(x,n,m))\ \wedge\ \hat{P}(x,z,n,m)\\
\hat{C}(x,n,0) &\Longrightarrow f(x) \neq \hat{S}(n,a)\\
\hat{C}(x,n,s(m)) &\Longrightarrow f(\hat{S}(s(m),x)) \neq \hat{S}(n,a)\ \vee\ \hat{C}(x,n,m)\\
\hat{T}(x,n,0) &\Longrightarrow f(x) \not< \hat{S}(s(n),a)\ \vee\ f(x) = \hat{S}(n,a)\ \vee\ f(x) < \hat{S}(n,a)\\
\hat{T}(x,n,s(m)) &\Longrightarrow f(\hat{S}(s(m),x)) \not< \hat{S}(s(n),a)\ \vee\ f(\hat{S}(s(m),x)) = \hat{S}(n,a) \vee\\
&\qquad f(x) < \hat{S}(n,a) \wedge\ \hat{T}(x,n,m)\\
\hat{S}(0,x) &\Longrightarrow x \qquad \hat{S}(s(n),x) \Longrightarrow suc(\hat{S}(s(n),x))
\end{aligned}
$$

- Yes, quite ugly! Goal is to handle mostly automatically.
- We need to provide unifiers for “instances” of x and y.

Simplified Representation

- Technical motivation, but can be presented in simpler terms:
- Let V be a countable set of variables symbol, and
- Let $\hat{a}, \hat{b}$ be a special variable symbol not in V
- For $x \in V$,let $V^x_{\mathbb{N}} = \{x_i \mid i \in \mathbb{N}\}$,
- $S(x_i) = x_{i+1}$, $\mathrm{ex}^s_{\hat{a}}(\hat{a}) = s$, and
- $\sigma(s,t)$ is a substitution s.t. $s\sigma(s,t) = t\sigma(s,t)$ and σ is an m.g.u without renaming variables.
- Let s and t be first-order terms such that:
  - $Var(s) \subset V^x_{\mathbb{N}} \cup \{\hat{a}\}$
  - $Var(t) \subset V^y_{\mathbb{N}} \cup \{\hat{b}\}$
  - $x, y, \hat{b}, \hat{a}$ are all distinct.
  - We will refer to pairs of such terms are Loops (denoted $\langle s, t \rangle$)

How to think about Loop Unification?

function $\textsc{Loop}(s, t, c)$

  if $\hat{a}$ *or* $\hat{b} \in dom(\sigma(s,t)) \wedge \hat{a}\sigma(s,t), \hat{b}\sigma(s,t) \notin V)$ **then**

    $\textsc{Loop}(\mathrm{ex}^{s}_{\hat{a}}(S(s)), \mathrm{ex}^{s}_{\hat{a}}(S(t)), \mathrm{ex}^{s}_{\hat{a}}, \mathrm{ex}^{t}_{\hat{b}})$

  end if

end function

Question: Is termination of $\mathrm{Loop}(s, t, \mathrm{ex}^{s}_{\hat{a}}, \mathrm{ex}^{t}_{\hat{b}})$ decidable ?

- Can we finitely represent the unifier of all extensions?
- Cannot be easily reduced to Narrowing, nor Primal Grammars.
- We can also think about this as follows:

Definition of Loop Unification

Definition (Loop Unification Problem)

Decide if for every extension of a loop $\langle s, t \rangle$, the corresponding terms are unifiable. If for any extension the terms are not unifiable then the Loop is not Unifiable.

Definition

Let a loop $\langle s, t \rangle$ be loop unifiable. We say $\langle s, t \rangle$ is infinitely loop unifiable if every extension is extendably unifiable. Otherwise, we say $\langle s, t \rangle$ is finitely loop unifiable.

- We focus on semiloops, only one term is extended.
- Doesn't seem hard, let's look at some examples.

Terminating and Unifiable

- Consider $\langle s, t| = \langle h(h(x_2, x_1), \hat{a}), \quad h(y_1, h(y_2, y_3))|$ together with the function $\mathrm{ex}^s_{\hat{a}}$.
- $\sigma = \{y_1 \mapsto h(x_2, x_1)\} \cup \{\hat{a} \mapsto h(y_2, y_3)\}$ unifies $\langle s, t|$.
- We refer such term pairs as extendably unifiable.
- Now consider $\langle s, t|_1 = \langle \mathrm{ex}^s_{\hat{a}}(S(s)), \quad h(y_1, h(y_2, y_3))|$.
- $\mathrm{ex}^s_{\hat{a}}(S(s)) = h(h(x_3, x_2), h(h(x_2, x_1), \hat{a}))$
- $\sigma' = \{y_1 \mapsto h(x_3, x_2)\} \cup \{y_2 \mapsto h(x_2, x_1)\} \cup \{y_3 \mapsto \hat{a}\}$ unifies $\langle s, t|_1$.
- This semiloop is finitely Loop unifiable.
- All extensions are unified by a substitution similar to σ'.
- What about terminating and not unifiable?

Terminating and Not unifiable

- $\langle s, t| = \langle h(h(h(x_2, x_1), h(x_2, x_3)), \hat{a}), \quad h(h(y_3, y_1), h(y_4, y_4))|$ together with the function $\mathrm{ex}^s_{\hat{a}}$.
- $\sigma_0 = \{y_3 \mapsto h(x_2, x_1)\ ,\ y_1 \mapsto h(x_2, x_3)\ ,\ \hat{a} \mapsto h(y_4, y_4)\}$ unifies $\langle s, t|_1$.
- $\langle s, t|_2$ is unified by $\sigma_1 =$

$$\{y_3 \mapsto h(x_3, x_2),\ y_4 \mapsto h(h(x_2, x_1), h(x_2, x_3)),$$
$$y_1 \mapsto h(x_3, x_4),\ \hat{a} \mapsto h(h(x_2, x_1), h(x_2, x_3))\}.$$

- However, the irreducible form derived from $\langle s, t|_3$ is

$$\{y_3 \stackrel{?}{=} h(x_4, x_3),\ y_4 \stackrel{?}{=} h(h(x_3, x_2), h(x_3, x_4)),$$
$$y_1 \stackrel{?}{=} h(x_4, x_5),\ \hat{a} \stackrel{?}{=} h(x_3, x_4),\ x_3 \stackrel{?}{=} h(x_2, x_1),$$
$$x_2 \stackrel{?}{=} h(x_2, x_3)\}.$$

- After finite steps we know some extensions are not unifiable.
- Are there infinitely Loop unifiable term pairs?

Non-terminating, but Unifiable!

- Consider $\langle s, t| = \langle h(h(x_1, x_1), \hat{a}), h(y_1, y_1)|$
- $\sigma_0 = \{\hat{a} \mapsto h(x_1, x_1)\}$ unifies $\langle s, t|_1$.
- $\sigma_1 = \{\hat{a} \mapsto h(x_1, x_1)\}$ unifies $\langle s, t|_2$.
- $\ldots$
- Cyclic behavior is also possible:

$$\langle s, t| = \langle (\hat{a}, h(h(h(x_1, x_1), x_1), x_1)), \quad h(h(h(h(y_1, y_1), y_1), y_1), y_1)|$$

- There are three types of unifiers depending on the extension.
  - The solved form of $\langle s, t|_{3n}$ contains $\hat{a} \stackrel{?}{=} h(h(t(1), t(1)), t(1))$,
  - the solved form of $\langle s, t|_{3n+1}$ contains $\hat{a} \stackrel{?}{=} h(t(1), t(1))$,
  - the solved form of $\langle s, t|_{3n+2}$ contains $\hat{a} \stackrel{?}{=} t(1)$,
  - where $t(n) = h(h(h(x_{n+1}, x_{n+1}), x_{n+1}), x_{n+1})$.

Sufficient Condition for Finite Unifiability

- Not enough to be unifiable and non-extendable.
  - $\langle s, t| = \langle h(x_2, h(x_4, \hat{a})), h(y_1, y_1)|$
  - A unifier of $\langle s, t|_1$ is $\{y_1 \mapsto h(x_4, \hat{a})\ ,\ x_2 \mapsto h(x_4, \hat{a})\}$
  - A unifier of $\langle s, t|_2$ from the above unifier:

$$\{y_1 \mapsto h(x_5, h(x_2, h(x_4, \hat{a}))),\ x_3 \mapsto h(x_5, h(x_2, h(x_4, \hat{a})))\}$$

  - However, generating the unifier for $\langle s, t|_3$ this way fails:

$$\{y_1 \mapsto h(x_6, h(x_3, h(x_5, h(x_2, h(x_4, \hat{a}))))),$$
$$x_4 \mapsto h(x_6, h(x_3, h(x_5, h(x_2, h(x_4, \hat{a})))))\}$$

  - Extension results in an occurrence check.
  - Every variable must be large enough not to cause occurrence checks through extension.
  - Or, variables indices form an interval without gaps.

Sufficient Condition for Infinite Unifiability: Decomposition

- Given enough information about the extensions of $\langle s, t|$ one can decomposed the unifier of $\langle s, t|_k$.
- We transform the unifier of $\langle s, t|_k$ into a compositions of unifiers for the semiloops $\langle s, t_1|, \cdots \langle s, t_{k-1}|$.
- Too technical to present here, instead we provide an example.

Sufficient Condition for Infinite Unifiability: Example

- Consider the following: $\langle s, t| = \langle h(t(0), \hat{a})\ ,\ h(y_1, h(y_2, y_1))|$ where $t(n) = h(x_{n+6}, h(x_{n+1}, x_{n+6}))$.
- $\langle s, t|_3 = \langle h(t(2), h(t(1), h(t(0), \hat{a})))\ ,\ h(y_1, h(y_2, y_1))\rangle$
- The solved form of $h(t(2), h(t(1), h(t(0), \hat{a}))) \stackrel{?}{=} t$ is

$$\{y_1 \stackrel{?}{=} h(x_8, h(x_3, x_8)),\ y_2 \stackrel{?}{=} h(x_7, h(x_2, x_7))$$
$$x_8 \stackrel{?}{=} h(x_6, h(x_1, x_6)),\ \hat{a} \stackrel{?}{=} h(x_3, h(x_6, h(x_1, x_6)))\}$$

- The unifier of $\langle s, t|_3$ can be written as

$$D(Id, h(t(0), \hat{a}), h(y_1, h(y_2, y_1)), 3) =$$
$$sh^2(\sigma^2)D(sh^1(\sigma^2), s, h(y_2, t(1))), 2) =$$
$$sh^2(\sigma^2)sh^1(\sigma^1)D(sh^1(\sigma^1), s, t(2), 1) =$$
$$sh^2(\sigma^2)sh^1(\sigma^1)\sigma^0 D(sh^1(\sigma^0), s, h(x_4, t(1)), 0) =$$
$$sh^2(\sigma^2)sh^1(\sigma^1)\sigma^0\{\hat{a} \mapsto h(x_3, h(h(x_6, h(x_1, x_6))\}$$

Sufficient Condition for Infinite Unifiability: Example

- where

$$\sigma^2 = \{y_1 \mapsto h(x_6, h(x_1, x_6))\}$$
$$\sigma^1 = \{y_2 \mapsto h(x_6, h(x_1, x_6))\}$$
$$\sigma^0 = \{x_8 \mapsto h(x_6, h(x_1, x_6))\}$$

- Surprisingly, this loop is not infinitely unifiable as the 14-extension is not unifiable.

Sufficient Condition for Infinite Unifiability

- The second and fourth argument of the decomposition do not directly influence the construction of the unifier.
- This leaves the substitution and the non-extendable term.
- When a unifier is large enough it may decompose as follows:

$$D'(Id, s, t, r+1) = \Theta(r+1)D'(\sigma_1^{\Delta}, s, t_1, r)$$

$$\vdots$$

$$D'(\sigma_{r-i+1}, s, t_{r-i+1}, i) = \Theta(i)D'(\sigma^*, s, t^*, i-1)$$

$$\vdots$$

$$D'(\sigma_{r-j+1}, s, t_{r-j+1}, j) = \Theta(j)D'(\sigma^*, s, t^*, j-1)$$

- We can use this to construct a primitive recursive definition of a unifier for any extension.

Example with a Cycle

- Consider the semiloop

$$\langle s, t| = \langle h(\hat{a}, h(h(x_1, x_1), x_1)) \ , \ \ h(h(h(y_1, y_1), y_1), y_1)|,$$

and we define $t(n) = h(h(x_{n+1}, x_{n+1}), x_{n+1}))$.

- Now consider the decomposition of $\langle s, t|_5$:

$$D(Id, s, t, 5) =$$

$$sh^4(\sigma_4)D(Id, s, h(h(t(1), t(1)), t(1)), 4) =$$

$$sh^4(\sigma_4)sh^3(\sigma_3)D'(\mathrm{Id}, s, h(t(1)), t(1))), 3) =$$

$$sh^4(\sigma_4)sh^3(\sigma_3)sh^2(\sigma_2)D'(Id, s, t(1), 2) =$$

$$sh^4(\sigma_4)sh^3(\sigma_3)sh^2(\sigma_2)sh^1(\sigma_1)D(\mathrm{Id}, s, h(t(1), t(1)), 1) =$$

$$sh^4(\sigma_4)sh^3(\sigma_3)sh^2(\sigma_2)sh^1(\sigma_1)\sigma_0\{\hat{a} \mapsto h(h(x_2, x_2), x_2)\}$$

Example with a Cycle

Where the substitutions σ_i are as follows:

$$\begin{aligned}
\sigma_4 =& \{y_2 \mapsto h(h(x_1, x_1), x_1)\} \\
\sigma_3 =& \{x_2 \mapsto x_1\} \\
\sigma_2 =& \{x_2 \mapsto x_1\} \\
\sigma_1 =& \{x_2 \mapsto h(h(x_1, x_1), x_1)\} \\
\sigma_0 =& \{x_2 \mapsto x_1\}
\end{aligned}$$

- Cycle repeats within the unifiers of large extensions of $\langle s, t|$.
- If a cycle is found within the decomposition of a large enough extension, then the semiloop is infinite loop unifiable.

When Extension is not Large Enough

- Consider $\langle s, t|$ where:

$$s = h(h(x_1, h(x_{16}, h(x_{32}, h(x_1, h(x_{16}, x_{32}))))), \hat{a})$$

$$t = h(y_1, h(y_2, h(y_3, h(y_1, h(y_2, y_3)))))$$

- The unifier of $\langle s, t|_{11}$ decomposes such that

$$D(Id, s, h(x_4, h(x_{19}, h(x_{35}, h(x_4, h(x_{19}, x_{35}))))), 9)$$

$$D(Id, s, h(x_4, h(x_{19}, h(x_{35}, h(x_4, h(x_{19}, x_{35}))))), 4)$$

occur.

- This fits the cycle requirement, yet, $\langle s, t|_{28}$ is not unifiable.
- Large enough: 2n+1 where n is the length of the interval containing all variables.

Conclusion

- We present a sufficient condition for semiloop unification.
- Evidence suggest this condition is necessary, still open.
- This concerns only a fragment of the loop unification problem.
- Future work: Extending results to full loop unification with a restricted number of variable classes.