

# Term-Indexing

## First-order terms

Alexander Maringele

January 27th, 2016

# References



R. Sekar, I. V. Ramakrishnan, and Andrei Voronkov, *Term indexing*, Handbook of Automated Reasoning (Alan Robinson and Andrei Voronkov, eds.), Elsevier Science Publishers B. V., Amsterdam, The Netherlands, 2001, pp. 1853–1964.

# Outline

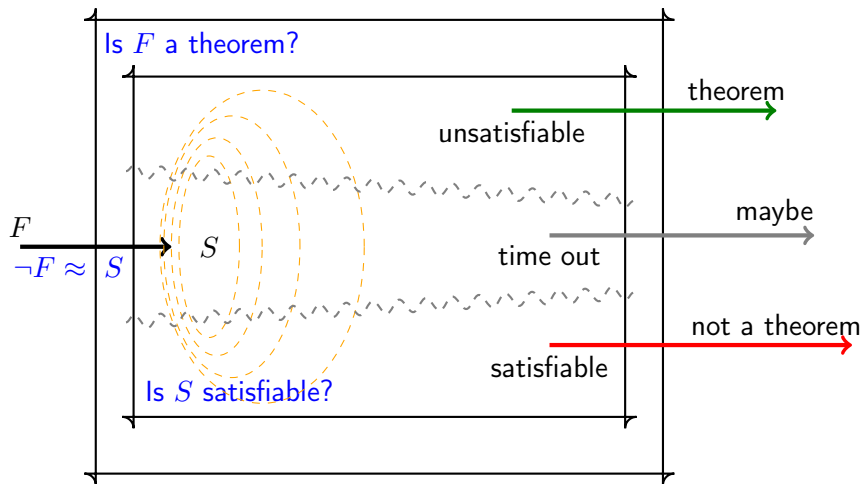
- 1 Motivation
- 2 Position
- 3 Path-Indexing
- 4 Discrimination Trees
- 5 Experiences

# Notation

## Notation

$\Sigma = (\mathcal{V}, \mathcal{F}, \mathcal{P})$	signature
$\mathcal{V} = \{x, y, z, \dots\}$	variables
$\mathcal{F} = \{a, f, g, h, \dots\}$	function symbols
$\mathcal{P} = \{P, Q, \dots\}$	predicate symbols
$\mathcal{T} = \mathcal{V} \cup \{f(t_1, \dots, t_n) \mid f \in \mathcal{F}, t_i \in \mathcal{T}\}$	terms
$\mathcal{A} = \{s \approx t, P(t_1, \dots, t_n) \mid P \in \mathcal{P}, s, t, t_i \in \mathcal{T}\}$	atoms
$\mathcal{L} = \mathcal{A} \cup \{\neg A \mid A \in \mathcal{A}\}$	literals

# Refutation-based theorem proving



## Goal

A sound and refutation complete calculus.

## Example

Apply resolution and factoring

$$\frac{A \vee \mathcal{C} \quad \neg B \vee \mathcal{D}}{(\mathcal{C} \vee \mathcal{D})\sigma} (\sigma) \text{ R} \qquad \frac{A \vee B \vee \mathcal{C}}{(A \vee \mathcal{C})\sigma} (\sigma) \text{ F}$$

$$\sigma = \text{mgu}(A, B)$$

eventually on all clauses and literals until the empty clause is derived.

## Observation

Search space grows too fast.

## Goal

A sound, refutation complete, and *effective* calculus.

- 1 *Reduce* search space
  - e.g. Ordered Resolution
  - ... selection functions for clauses and literals
- 2 *Reduce* redundancy
  - e.g. ignore clause  $\mathcal{D}$ , if  $\mathcal{C}$  subsumes  $\mathcal{D}$ , i.e.  $\mathcal{C}\tau \subseteq \mathcal{D}$ .
  - ... depends on the calculus

## Example (forward subsumption)

$$S = \{^1P(x, y), ^2\neg P(a, z)\} \cup \{^3P(a, z')\}$$

$t_1$  subsumes  $t_3$

$$\frac{P(x, y) \quad \neg P(a, z)}{\square} \quad \{x \mapsto a, y \mapsto z\}$$

Resolution

$$S\perp = \{P(\perp, \perp), \neg P(a, \perp), P(a, \perp)\}$$

InstGen/SMT

## Goal

A sound, refutation complete, and *effective* calculus.

### 3 Quickly find

- *variants*
- *instances*
- *generalizations*
- *unifiable terms*

variant removal  
backward subsumption  
forward subsumption  
resolution, etc.

of a query term in a given set of terms.

## Observation

Deduction rate drops quickly with linear search.

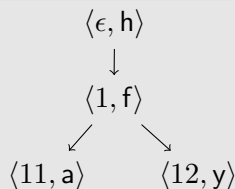


# Position-Strings

## Positions of a term

$$\mathcal{P}\text{os}(t) = \begin{cases} \{\epsilon\} & \text{if } t = x \in \mathcal{V} \\ \{\epsilon\} \cup \{ip \mid 1 \leq i \leq n \wedge p \in \mathcal{P}\text{os}(t_i)\} & \text{if } t = f(t_1, \dots, t_n) \end{cases}$$

## Traversals of $h(f(a, y))$



$$\mathcal{P}\text{os}(h(f(a, y))) = \{\epsilon, 1, 11, 12\}$$

$$h(f(a, y))|_{12} = y \qquad \langle 12, y \rangle$$

$$\langle \epsilon, h \rangle \langle 1, f \rangle \langle 12, y \rangle \quad \text{root to leaf } y \quad (h1f2y)$$

$$\langle \epsilon, h \rangle \langle 1, f \rangle \langle 11, a \rangle \langle 12, y \rangle \quad \text{pre-order} \quad (hfay)$$

# Normalization of variables

## Forget variable names

$$\text{path} : f(x, y) \mapsto \{f1*, f2*\}$$

$$f(x, x) \mapsto \{f1*, f2*\}$$

$$\text{pre-order} : f(x, y) \mapsto f**$$

$$f(x, x) \mapsto f**$$

## Enumerate variable names

$$\text{path} : f(x, y) \mapsto \{f1*_1, f2*_2\}$$

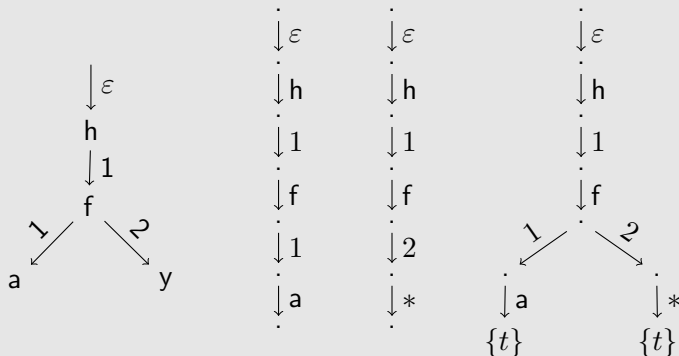
$$f(x, x) \mapsto \{f1*_1, f2*_1\}$$

$$\text{pre-order} : f(x, y) \mapsto f*_1*_2$$

$$f(x, x) \mapsto f*_1*_1$$

# Path-Strings

The path-strings of  $h(f(a, y))$  are **h1f1a** and **h1f2\***.

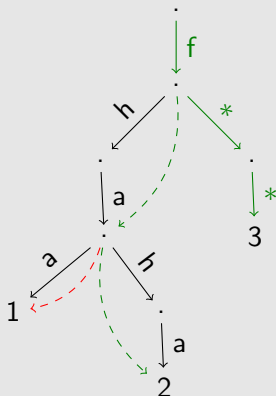


# Pre-Order Strings





## Non-linear terms



$$\{^1: f(h(a), a), ^2: f(h(a), h(a))\}, ^3: f(x, y)\}$$

The terms  $f(x, y)$  and  $f(z, z)$  share the preorder-term  $f^{**}$ .

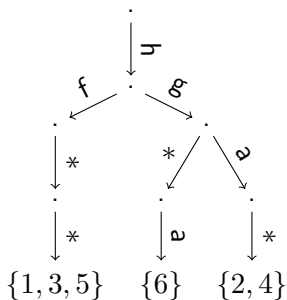
But  $f(h(a), a)$  is not an instance of  $f(z, z)$ .

$f(a, x)$  $f$  $a$  $x$  $f(a, x) \approx f(x, a)$  $\approx$  $f$  $a$  $x$  $f$  $x$  $a$

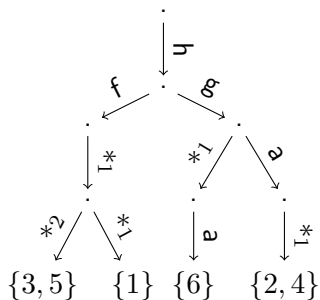


## Perfect filtering

$$\left\{ \begin{array}{lll} {}^1\text{h}(\text{f}(x, x)), & {}^2\text{h}(\text{g}(\text{a}, x)), & {}^3\text{h}(\text{f}(y, z)), \\ {}^4\text{h}(\text{g}(\text{a}, y)), & {}^5\text{h}(\text{f}(y, x)), & {}^6\text{h}(\text{g}(y, a)) \end{array} \right\}$$



$$h(f(y, x)) \mapsto h.f.*.*$$



$$h(f(y, x)) \mapsto' h.f.*_1.*_2$$

