Term-Indexing

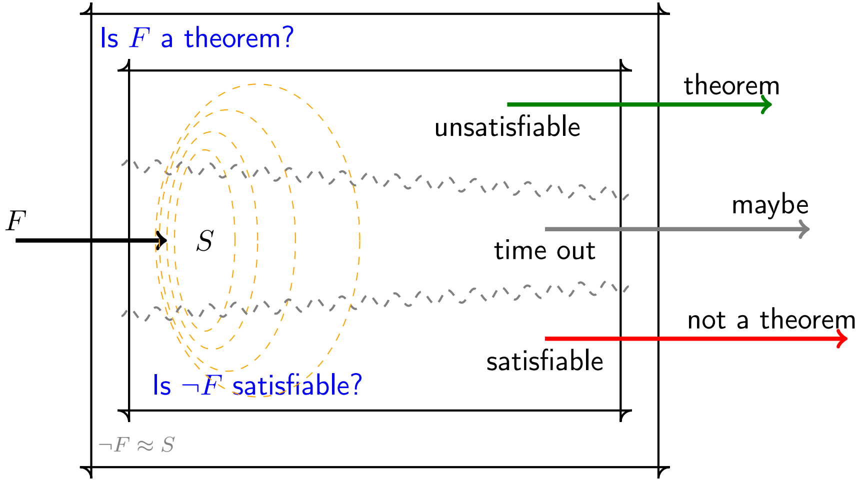
Alexander Maringele

January 27th, 2016

References

Outline

Refutation


Is F a theorem?
F
S
unsatisfiable
theorem
maybe
time out
not a theorem
satisfiable
Is $\neg F$ satisfiable?
$\neg F \approx S$

Notation

Clausal form

$$\{\ \mathsf{P}(\mathsf{f}(x)) \vee \mathsf{f}(x) \not\approx \mathsf{a},\ \ \mathsf{g}(x,y) \approx \mathsf{a} \vee \neg\mathsf{Q}(x,y),\ \ \mathcal{C}_3\ \}$$

$$\equiv$$

$$\forall x\,(\mathsf{P}(\mathsf{f}(x)) \vee \mathsf{f}(x) \not\approx \mathsf{a})$$

$$\wedge$$

$$\forall xy\,(\mathsf{g}(x,y) \approx \mathsf{a} \vee \neg\mathsf{Q}(x,y))$$

$$\wedge$$

$$\forall\,\mathcal{V}\mathsf{ar}(\mathcal{C}_3)\,(\mathcal{C}_3)$$

Goal

A sound and refutation complete calculus.

Resolution (without equality)

Resolve and factor all clauses and literals in an unsatisfiable set

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

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

and the empty clause will be derived eventually.

Observation

Usually the set grows too fast to obtain a result.

Goal

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

1. *Reduce* search space
   - Ordered Resolution, Strategies, ...
   - ... with selection functions for clauses and literals
2. *Reduce* redundancy
   - e.g. discard clauses that are subsumed by other clauses
   - ... depending on the calculus

Example (forward subsumption)

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

t_1 subsumes t_3

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

Resolution

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

InstGen / SMT

Goal

A sound, refutation complete, and effective calculus.

3. Quickly find
   - *variants* — variant removal
   - *instances* — backward subsumption
   - *generalizations* — forward subsumption
   - *unifiable terms* — resolution, demodulation

   of a query term in a given set of terms.

Observation

Deduction rate drops quickly with sequential search.

Term Indexing

Data structures and algorithms for fast retrieval of matching terms.

Position Strings

Definition

$$\mathcal{P}\mathsf{os}^{\Sigma}(t) = \begin{cases} \{\langle \epsilon, x\rangle\} & \text{if } t = x \in \mathcal{V} \\ \{\langle \epsilon, f\rangle\} \cup \{\langle ip, s\rangle \mid \langle p, s\rangle \in \mathcal{P}\mathsf{os}^{\Sigma}(t_i)\} & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Term traversals

$$\begin{tikzcd}[column sep=small] & \langle \epsilon, \mathsf{h}\rangle \arrow[d, dashed] & \\ & \langle 1, \mathsf{f}\rangle \arrow[dl, dotted] \arrow[dr, dashed] & \\ \langle 11, \mathsf{a}\rangle \arrow[rr, dotted, bend left] & & \langle 12, y\rangle \end{tikzcd}$$

$$\mathcal{P}\mathsf{os}^{\Sigma}(\mathsf{h}(\mathsf{f}(\mathsf{a}, \mathsf{y}))) = \{\langle \epsilon, \mathsf{h}\rangle, \langle 1, \mathsf{f}\rangle, \langle 11, \mathsf{a}\rangle, \langle 12, y\rangle\}$$

$\langle \epsilon, \mathsf{h}\rangle\langle 1, \mathsf{f}\rangle\langle 12, y\rangle$ path from root to leaf

$\langle \epsilon, \mathsf{h}\rangle\langle 1, \mathsf{f}\rangle\langle 11, \mathsf{a}\rangle\langle 12, y\rangle$ pre-order traversal

Variables

Variants of terms generate the same position strings

- if variable names are ignored $\mathsf{f}(y, z) \Rightarrow \langle \epsilon, \mathsf{f} \rangle \langle 1, * \rangle \langle 2, * \rangle$
- or normalized $\mathsf{f}(y, z) \Rightarrow \langle \epsilon, \mathsf{f} \rangle \langle 1, x_1 \rangle \langle 2, x_2 \rangle$
  $\mathsf{f}(y, y) \Rightarrow \langle \epsilon, \mathsf{f} \rangle \langle 1, x_1 \rangle \langle 2, x_1 \rangle$

In the first case even non-variants of terms generate the same strings.

Notation

We abbreviate

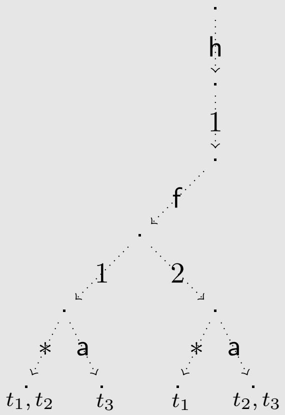
- path strings $\langle \epsilon, \mathsf{h} \rangle \langle 1, \mathsf{f} \rangle \langle 12, * \rangle$ $\mathsf{h}.1.\mathsf{f}.2.*$
- and pre-order traversal strings $\langle \epsilon, \mathsf{h} \rangle \langle 1, \mathsf{f} \rangle \langle 11, * \rangle \langle 12, * \rangle$ $\mathsf{h}.\mathsf{f}.\mathsf{a}.*$
  when the arities of function symbols are fixed.

Path Indexing

Build

$^{t_1:}\mathsf{h}(\mathsf{f}(x,y)),{}^{t_2:}\mathsf{h}(\mathsf{f}(x,\mathsf{a})),{}^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{a},\mathsf{a}))$

$$t_1 \Rightarrow \{\mathsf{h.1.f.1.}*, \mathsf{h.1.f.2.}*\}$$
$$t_2 \Rightarrow \{\mathsf{h.1.f.1.}*, \mathsf{h.1.f.2.a}\}$$
$$t_3 \Rightarrow \{\mathsf{h.1.f.1.a}, \mathsf{h.1.f.2a}\}$$



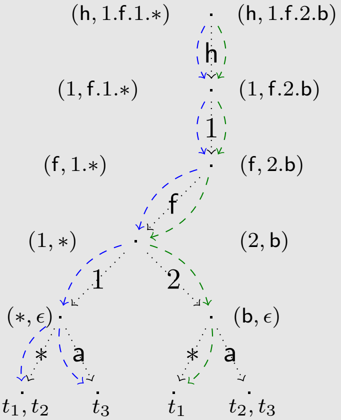
Retrieve

$^{t_1:}\mathsf{h}(\mathsf{f}(x,y)), ^{t_2:}\mathsf{h}(\mathsf{f}(x,\mathsf{a})), ^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{a},\mathsf{a}))$

$$\mathsf{h}(\mathsf{f}(z,\mathsf{b}))) \Rightarrow \{\mathsf{h}.1.\mathsf{f}.1.*, \mathsf{h}.1.\mathsf{f}.2.\mathsf{b}\}$$

$$u : \mathsf{h}(\mathsf{f}(z,\mathsf{b})) \mapsto \{t_1, t_2, t_3\} \cap \{t_1\}$$
$$i : \mathsf{h}(\mathsf{f}(z,\mathsf{b})) \mapsto \{t_1, t_2, t_3\} \cap \{\}$$
$$g : \mathsf{h}(\mathsf{f}(z,\mathsf{b})) \mapsto \{t_1, t_2\} \cap \{t_1\}$$
$$v : \mathsf{h}(\mathsf{f}(z,\mathsf{b})) \mapsto \{t_1, t_2\} \cap \{\}$$

$$v : \mathsf{h}(\mathsf{f}(z,z)) \mapsto \{t_1, t_2\} \cap \{t_1\}$$



Unit Superposition Inference Rules

$$\frac{s \approx t \quad L[s']}{(L[t]) \cdot \sigma} \text{ unit paramodulation}$$

where $\sigma = \mathrm{mgu}(s, s')$, $s' \notin \mathcal{V}$, $t\sigma \not\succ s\sigma$

$$\frac{s \approx t \quad u[s'] \not\approx v}{(u[t] \not\approx v) \cdot \sigma} \text{ unit superposition } \frac{s \approx t \quad u[s'] \approx v}{(u[t] \approx v) \cdot \sigma}$$

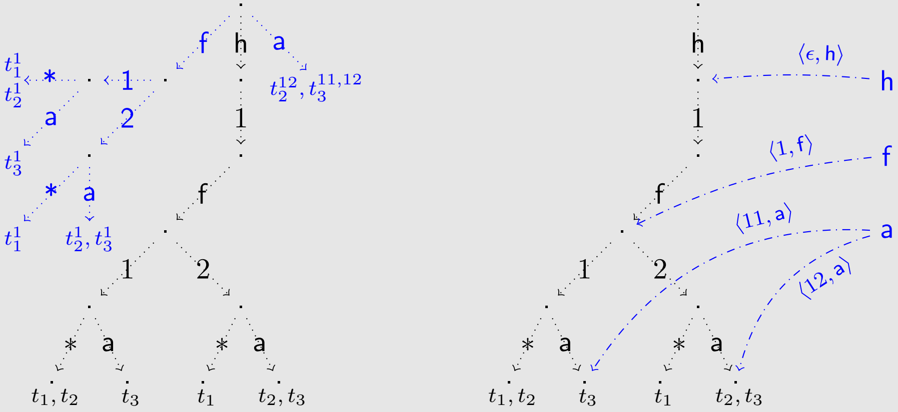
where $\sigma = \mathrm{mgu}(s, s')$, $s' \notin \mathcal{V}$, $t\sigma \not\succ s\sigma$, $v\sigma \not\succ u[s']\sigma$

$$\frac{s \not\approx t}{\square} \text{ unit equality resolution} \qquad \frac{A \quad \neg B}{\square} \text{ unit resolution}$$

where s and t (A and B respectively) are unifiable

Demodulation

$^{t_1:}\mathsf{h}(\mathsf{f}(x,y)), {}^{t_2:}\mathsf{h}(\mathsf{f}(x,\mathsf{a})), {}^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{a},\mathsf{a})), \ldots, \mathsf{f}(x,\mathsf{a}) \approx x$



Discrimination Trees

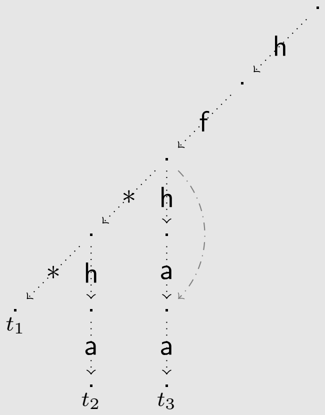
Insert

${}^{t_1:}\mathsf{h}(\mathsf{f}(x, y)), {}^{t_2:}\mathsf{h}(\mathsf{f}(x, \mathsf{h}(\mathsf{a}))), {}^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{h}(\mathsf{a}), \mathsf{a}))$

$$t_1 \Rightarrow \mathsf{h.f.}{*}.{*}$$

$$t_2 \Rightarrow \mathsf{h.f.}{*}\mathsf{.h.a}$$

$$t_3 \Rightarrow \mathsf{h.f.h.a.a}$$

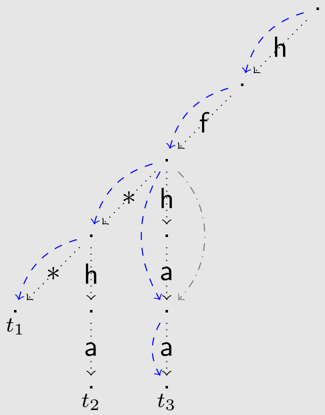


Retrieve

$^{t_1:}\mathsf{h}(\mathsf{f}(x, y)), ^{t_2:}\mathsf{h}(\mathsf{f}(x, \mathsf{h}(\mathsf{a}))), ^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{h}(\mathsf{a}), \mathsf{a}))$

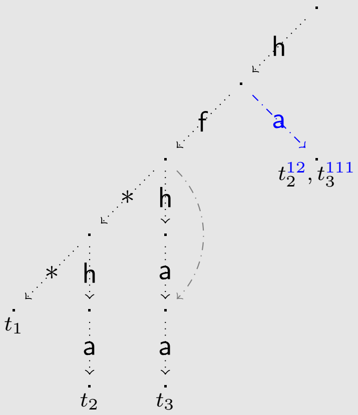
$$\mathsf{h}(\mathsf{f}(x', \mathsf{a})) \Rightarrow \mathsf{h.f.}{*}\mathsf{.a}$$

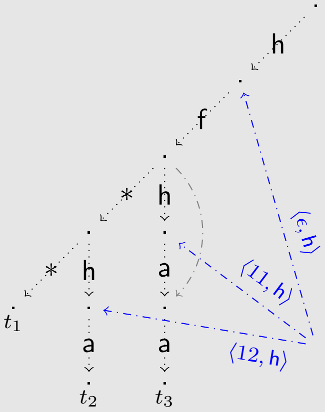
$$u : \mathsf{h}(\mathsf{f}(x', \mathsf{a})) \mapsto \{t_1, t_3\}$$
$$i : \mathsf{h}(\mathsf{f}(x', \mathsf{a})) \mapsto \{t_3\}$$
$$g : \mathsf{h}(\mathsf{f}(x', \mathsf{a})) \mapsto \{t_1\}$$
$$v : \mathsf{h}(\mathsf{f}(x', \mathsf{a})) \mapsto \{ \}$$



Subterms

$^{t_1:}$ $\mathsf{h}(\mathsf{f}(x, y))$, $^{t_2:}$ $\mathsf{h}(\mathsf{f}(x, \mathsf{h}(\mathsf{a})))$, $^{t_3:}$ $\mathsf{h}(\mathsf{f}(\mathsf{h}(\mathsf{a}), \mathsf{a}))$





Substitution Trees

Build

$$^{t_1:}\mathsf{h}(\mathsf{f}(x,y)),\ ^{t_2:}\mathsf{h}(\mathsf{f}(x,\mathsf{h}(\mathsf{a}))),\ ^{t_3:}\mathsf{h}(\mathsf{f}(\mathsf{h}(\mathsf{a}),\mathsf{a})),\ ^{t_4:}\mathsf{h}(\mathsf{f}(\mathsf{a},\mathsf{a}))$$

$$\downarrow$$

$$\begin{tikzcd}
 & & & *_0 \mapsto \mathsf{h}(*_1) \arrow[d] & & & \\
 & & & *_1 \mapsto \mathsf{f}(*_2, *_3) \arrow[dll] \arrow[drr] & & & \\
 & *_2 \mapsto x \arrow[dl] \arrow[dr] & & & & *_3 \mapsto \mathsf{a} \arrow[dl] \arrow[dr] & \\
*_3 \mapsto y & & *_3 \mapsto \mathsf{h}(\mathsf{a}) & & *_2 \mapsto \mathsf{h}(\mathsf{a}) & & *_2 \mapsto \mathsf{a} \\
t_1 & & t_2 & & t_3 & & t_4
\end{tikzcd}$$

`TPTP/Problems/HWV/HWV134-1.p`
`2 332 428 formulae, 6 570 884 literals`

checking 1000 new literals			sequential	path	speed
afterwards	(ℓ_1, ℓ_2)	$A, \neg B$	search	index	up
1 000	500 000	761	726ms	70ms	10
2 000	1 500 000	812	2s	69ms	29
4 000	3 500 000	723	4s	75ms	53
8 000	7 500 000	433	9s	125ms	72
16 000	15 500 000	742	21s	221ms	95
32 000	31 500 000	592	40s	489ms	81
64 000	63 500 000	1 167	*80s*	697ms	114
128 000	127 500 000	1 479	*160s*	13s	12
256 000	255 500 000	1 097	*320s*	440s	<1
512 000	511 500 000	1 440	*640s*	348s	<2
1 024 000	1023 500 000	1 534	*1280s*	330s	<4