A Tableau System for Typed Finite Sets

Second Bachelor Seminar Talk

Alexander Anisimov

Advisors: Chrisian Doczkal, Gert Smolka
Supervisor: Gert Smolka

Saarland University
Programming Systems Lab

May 9, 2015

Which problems can be expressed?

Definition

$$\begin{array}{lll} set & ::= & \emptyset \mid x \mid \{set\} \mid \{x \in set \mid rel\} \mid set \cup set \mid \mathcal{P}(set) \\ rel & ::= & set \in set \mid set \subseteq set \mid set = set \mid \neg rel \end{array}$$

- Set differences and intersections can be expressed as separations:
  $A \cap B = \{x \in A \mid x \in B\}$
  $A \backslash B = \{x \in A \mid x \notin B\}$
- logical operations inside separations can be eliminated:
  $\{x \in A \mid f\,x \vee g\,x\} = \{x \in A \mid f\,x\} \cup \{x \in A \mid g\,x\}$
  $\{x \in A \mid f\,x \wedge g\,x\} = \{x \in \{y \in A \mid f\,y\} \mid g\,x\}$
- explicitly given sets can be written as union of their elements:
  $\{x_1, x_2, \ldots, x_n\} = \{x_1\} \cup \{x_2\} \cup \cdots \cup \{x_n\}$

Tableau Refutation Systems

Definition (branch)

A *branch* is a finite set of relation statements

- tableau based refutation system
  - set of rules
  - input is a branch
  - infer further relation statements and add them to the branch
  - look for a contradiction
- proof of a proposition
  - put premisses on an empty branch
  - add negation of conclusion to the branch
  - infer a contradiction

Multi-level Syllogistic with Singletons

Definition (Language of MLSS)

$$
\begin{aligned}
\mathit{term} \quad &::= \emptyset \mid v_i \mid \{\mathit{term}\} \mid \mathit{term} \cup \mathit{term} \mid \mathit{term} \setminus \mathit{term} \qquad i \in \mathbb{N} \\
\mathit{formula} \quad &::= \mathit{term} \in \mathit{term} \mid \mathit{term} = \mathit{term} \\
&\mid \neg\, \mathit{formula} \mid \mathit{formula} \,\&\, \mathit{formula} \mid \mathit{formula} \vee \mathit{formula} \\
&\mid \mathit{formula} \rightarrow \mathit{formula} \mid \mathit{formula} \leftrightarrow \mathit{formula}
\end{aligned}
$$

- $A \subseteq B \Leftrightarrow A \cup B = B$
- $\{x_1, x_2, \ldots, x_n\} = \{x_1\} \cup \{x_2\} \cup \cdots \cup \{x_n\}$
- $A \cap B = A \cup B \setminus \big((A \setminus B) \cup (B \setminus A)\big)$
- **There is no way to express set separations $\{\mathbf{x} \in \mathbf{A} \mid \mathbf{p\,x}\}$**

Related Work

- Untyped sets with urelements and an explicit 'finite' predicate [Domenico Cantone, Rosa Ruggeri Cannata - 1995]
- Tableau calculus for an unquantified fragment of set theory [Bernhard Beckert, and Ulrike Hartmer - 1998]
- Fast tableau-based decision procedure for an unquantified fragment of set theory [Domenico Cantone, Calogero G. Zarba - 1998]
- A fragment of set theory with iterated membership [Domenico Cantone, Calogero G. Zarba, Rosa Ruggeri Cannata - 2005]
- Comprehension rules with substitution (tech report) [Benjamin Shults - 1997]

Harmless Rules

$$\frac{x \in A \qquad A \subseteq B}{x \in B} \qquad \frac{x \notin A \qquad B \subseteq A}{x \notin B}$$

$$\frac{A = B}{A \subseteq B \qquad B \subseteq A} \qquad \frac{A \in \mathcal{P}(B)}{A \subseteq B}$$

$$\frac{x \in \{y\}}{x = y} \qquad \frac{x \notin \{y\}}{x \neq y} \qquad \frac{\{x\} \subseteq A}{x \in A}$$

$$\frac{x \in A \cup B}{x \in A \mid x \in B} \qquad \frac{x \notin A \cup B}{x \notin A \qquad x \notin B}$$

Problematic Rules

generation of fresh variables

$$\frac{A \not\subseteq B}{x_{A,B} \in A \qquad x_{A,B} \notin B} \qquad \frac{A \notin \mathcal{P}(B)}{x_{A,B} \in A \qquad x_{A,B} \notin B}$$

$$\frac{A \neq B}{\begin{array}{c|c} x_{A,B} \in A & x_{B,A} \in B \\ x_{A,B} \notin B & x_{B,A} \notin A \end{array}}$$

substitution

$$\frac{y \in \{x \in A \,|\, p\}}{y \in A \qquad p^x_y} \qquad \frac{y \notin \{x \in A \,|\, p\}}{y \notin A \;|\; \neg\, p^x_y}$$

Nontermination of the Full Tableau System

- We start with the branch
  $F := \left\{a \in A \mid B \not\subseteq \{a\} \cup C\right\}$
  $x \in F$
  $B \subseteq F$
- . . . and add stepwise the following formulas
  $x \in A, \quad B \not\subseteq \{x\} \cup C$
  $y \in B, \quad y \notin \{x\} \cup C$
  $y \in F$
  $y \in A, \quad B \not\subseteq \{y\} \cup C$
- For legibility reasons: y instead of $x_{B,\{x\} \cup C}$
- From $B \not\subseteq \{y\} \cup C$ we can generate an $x_{B,\{y\} \cup C}$ that behaves like $y \Rightarrow$ **The system doesn't terminate!**

The Restricted System

- interaction between fresh variable generation and substitution may cause divergence
- interested in a terminating system
- define the **restricted system**

- remove separation rules

$$\frac{y \in \{x \in A \mid p\}}{y \in A \qquad p^x_y} \qquad \frac{y \notin \{x \in A \mid p\}}{y \notin A \mid \neg p^x_y}$$

- add rules for intersection and set difference instead to limit the loss of expressive power

$$\frac{x \in A \cap B}{x \in A \qquad x \in B} \qquad \frac{x \notin A \cap B}{x \notin A \mid x \notin B}$$

$$\frac{x \in A \backslash B}{x \in A \qquad x \notin B} \qquad \frac{x \notin A \backslash B}{x \notin A \mid x \in B}$$

Theorem

The restricted system terminates.

Level

Definition (Level)

The *level* of a set expression is the number of toplevel `fset` constructors in its Type.

Example

Base type `T` without `fset` constructors, `A : {fset T}`
$\mathcal{P}$(`A`) has level 2 as its type is {`fset {fset T}`}.

Definition

Let Γ be a branch. $S_l(\Gamma)$ is the set of all set expressions of level l occurring somewhere in Γ.

Example

`T` base type without `fset` constructors in it
$\Gamma := \{x \notin A \cup B\}$ for some `(x:T)`, `(A B:{fset T})`. Then

$S_0(\Gamma) = \{x\}$
$S_1(\Gamma) = \{A, B, A \cup B\}$
$S_2(\Gamma) = \emptyset$

- Every branch Γ has a maximal level L_Γ s.t. $S_{L_\Gamma} \neq \emptyset$ and $\forall L > L_\Gamma.\ S_L = \emptyset$.
- $S_l(\Gamma)$ is finite for every $l \in \mathbb{N}$ and every branch Γ

Definition (set expression closure)

$$S_l^+(\Gamma) := \begin{cases} \emptyset & \textit{if } l > L_\Gamma \\ S_l(\Gamma) \cup f_l(\Gamma) & \textit{otherwise} \end{cases}$$

$$f_l(\Gamma) := \left\{ x_{uv} \textit{ at level } l \mid (u,v) \in \left(S_{l+1}^+(\Gamma)\right)^2 \right\}$$

$$\mathcal{S}(\Gamma) := \bigcup_{l=0}^{L_\Gamma} S_l^+(\Gamma)$$

- $S_l^+(\Gamma)$ is finite if $f_l(\Gamma)$ is
- $f_l(\Gamma)$ is finite if $S_{l+1}^+(\Gamma)$ is
- $f_{L_\Gamma}(\Gamma) = \emptyset$

$\Rightarrow$ $\forall l \in \mathbb{N}.\ S_l^+(\Gamma)$ is finite
$\Rightarrow$ the set expression closure $\mathcal{S}(\Gamma)$ is finite

- every relation inferred from Γ is of the form $X \circ Y$ for some $X, Y \in \mathcal{S}(\Gamma)$ and $\circ \in \{\in, \notin, \subseteq, \not\subseteq, =, \neq\}$
  $\Rightarrow$ there are at most $6 * |\mathcal{S}(\Gamma)|^2$ relations to generate
- application of a rule adds at least one new relation statement to the branch
- no relation statement is added twice
- no relation statement is ever removed from the branch

$\Rightarrow$ at some point no new relation statements can be added and the system terminates

Proof Search in `Ltac`

- current implementation uses the unrestricted tableau ruleset
  - more expressive power
  - may diverge
- differences between procedure and tableau system
  - relation statements are added with respect to their information content, but not to the names of the variables (e.g. in the branch
    $A \not\subseteq B$
    $y \in A$
    $y \notin B$
    the procedure wouldn't generate any fresh $x_{A,B}$ as suggested by the tableau system)
  - call to `subst` if a set or an urelement is equal to some variable
    $\Rightarrow$ reduces the number of generated relations

Future Work

- investigate decidability of the modelled fragment of set theory
- prove or disprove completeness of the full tableau system
- investigate necessity of cut rules
- improve implementation

References

Domenico Cantone, Calogero G. Zarba
A New Fast Tableau-Based Decision Procedure for an Unquantified Fragment of Set Theory
FTP (LNCS Selection) 1998: 126-136

Bernhard Beckert, Ulrike Hartmer
A Tableau Calculus for Quantifier-Free Set Theoretic Formulae
TABLEAUX 1998: 93-107

Domenico Cantone, Rosa Ruggeri Cannata
Deciding set-theoretic formulae with the predicate 'finite' by a tableau calculus
Le Matematiche Vol 50, No 1 (1995)

References

Domenico Cantone, Calogero G. Zarba, Rosa Ruggeri Cannata
A Tableau-Based Decision Procedure for a Fragment of Set Theory with Iterated Membership.
J. Autom. Reasoning 34(1): 49-72 (2005)

Benjamin Shults
Comprehension and Description in Tableaux
1997