A Complete Gentzen-style Axiomatization for Set Constraints

Allan Cheng* and Dexter Kozen

Computer Science Department
Cornell University
Ithaca, New York 14853, USA
e-mail:{acheng,kozen}@cs.cornell.edu

Abstract. Set constraints are inclusion relations between expressions denoting sets of ground terms over a ranked alphabet. They are the main ingredient in set-based program analysis. In this paper we provide a Gentzen-style axiomatization for sequents $\Phi \vdash \Psi$, where Φ and Ψ are finite sets of set constraints, based on the axioms of termset algebra. Sequents of the restricted form $\Phi \vdash \perp$ correspond to positive set constraints, and those of the more general form $\Phi \vdash \Psi$ correspond to systems of mixed positive and negative set constraints. We show that the deductive system is (i) complete for the restricted sequents $\Phi \vdash \perp$ over standard models, (ii) incomplete for general sequents $\Phi \vdash \Psi$ over standard models, but (iii) complete for general sequents over set-theoretic termset algebras.

1 Introduction

Set constraints are inclusions between expressions denoting sets of ground terms. They have been used extensively in program analysis and type inference for many years [AM91a, AM91b, Hei93, HJ90b, JM79, Mis84, MR85, Rey69, YO88]. Considerable recent effort has focussed on the complexity of the satisfiability problem [AKVW93, AKW95, AW92, BGW93, CP94a, CP94b, GTT93a, GTT93b, HJ90a, Ste94]. Set constraints have also recently been used to define a constraint logic programming language over sets of ground terms that generalizes ordinary logic programming over an Herbrand domain [Koz94].

Set constraints exhibit a rich mathematical structure. There are strong connections to automata theory [GTT93a, GTT93b], type theory [KPS93, KPS94], first-order monadic logic [BGW93, CP94a], Boolean algebras with operators [JT51, JT52], and modal logic [Koz93]. There are algebraic and topological formulations, corresponding roughly to "soft" and "hard" typing respectively, which are related by Stone duality [Koz93, Koz95].

An axiomatization of the main properties of set constraints was proposed in [Koz93]. General models of these axioms are called *termset algebras*. In [Koz93],

* Visiting from Aarhus, **BRICS**, Basic Research in Computer Science, Center of the Danish National Research Foundation. e-mail:acheng@daimi.aau.dk

a representation theorem was proved showing that every termset algebra is isomorphic to a set-theoretic termset algebra. These models include the standard models in which set expressions are interpreted as sets of ground terms, as well as nonstandard models in which set expressions are interpreted as sets of states of *term automata* [KPS92].

In this paper we propose a Gentzen-style axiomatization involving sequents of the form $\Phi \vdash \Psi$, where Φ and Ψ are finite sets of set constraints. The intended interpretation of the sequent $\Phi \vdash \Psi$ is that if all the constraints in Φ hold of some model, then at least one of the constraints Ψ holds in that model.

This axiomatization can be thought of as a deductive system for refuting unsatisfiable systems of mixed positive and negative constraints. Deriving the sequent $\Phi \vdash \Psi$ is tantamount to refuting the mixed system $\Phi \cup \{s \neq t \mid s = t \in \Psi\}$. Systems of the restricted form $\Phi \vdash \bot$ correspond to systems of positive set constraints alone.

For this deductive system, we prove

(i) completeness over standard models for satisfiability of positive set constraints alone (if Φ is unsatisfiable, then Φ is refutable, *i.e.*, $\Phi \vdash \bot$ is derivable);
(ii) incompleteness over standard models for satisfiability of mixed positive and negative constraints (*i.e.*, not all valid sequents $\Phi \vdash \Psi$ are derivable);
(iii) completeness over nonstandard models (all set-theoretic termset algebras) for satisfiability of mixed positive and negative constraints (*i.e.*, all valid sequents $\Phi \vdash \Psi$ are derivable).

We feel that these results are of both theoretical and practical interest. Theoretically, they shed light on the distinction between exclusively positive and mixed positive and negative constraints. Although several interesting results involving the decidability and complexity of negative constraints have appeared [CP94b, GTT93b, AKW95, Ste94], the distinction between the two cases is still far from clear from a deductive standpoint.

Practically, we were interested in recasting the axioms of [Koz93] in a Gentzen style so as to take advantage of one of a number of automated deduction systems to implement a constraint solving package [Gri87]. We foresee this as being a useful alternative approach to building a set constraint solver for use in program analysis or constraint logic programming over set constraints.

This paper is organized as follows. In §2–§5, we briefly review the basic definitions and known results we will need regarding set constraints, termset algebras, term automata, and normal forms. These are included here for the sake of self-containment. In §6 we present our main results. Finally, in §7 we draw conclusions and discuss future work.

2 Set Expressions and Set Constraints

Let Σ be a finite ranked alphabet consisting of symbols f, each with an associated arity. Symbols in Σ of arity 0, 1, 2, and n are called *nullary*, *unary*, *binary*, and *n-ary*, respectively. Nullary elements are denoted by $a, b, \ldots$ and are called *constants*. The set of elements of Σ of arity n is denoted Σ_n. In the sequel, the use of expressions of the form $f(t_1, \ldots, t_n)$ carries the implicit assumption that f is of arity n.

The set of *ground terms* over Σ is denoted T_Σ. It is the least set such that if $t_1, \ldots, t_n \in T_\Sigma$ and $f \in \Sigma_n$, then $f(t_1, \ldots, t_n) \in T_\Sigma$. If $X = \{x, y, \ldots\}$ is a set of variables, then $T_\Sigma(X)$ denotes the set of ground terms over Σ and X, considering variables in X as symbols of arity 0.

Let $\mathrm{B} = (\cup,\ \cap,\ \sim,\ 0,\ 1)$ denote the usual signature of Boolean algebra. Let $\Sigma{+}\mathrm{B}$ denote the signature consisting of the disjoint union of Σ and B. Boolean operators such as $-$ (set difference) and $\oplus$ (symmetric difference) are defined from these as usual. A *set expression* over X is an element of $T_{\Sigma+\mathrm{B}}(X)$. We use $s, t, \ldots$ to denote set expressions. A typical set expression could be:

$$f(g(x \cup y), \sim(g(a) \cap b))$$

where g, f are symbols of arity 1 and 2, respectively, a, b are constants, and $x, y \in X$. A *Boolean expression* over X is an element of $T_{\mathrm{B}}(X)$.

A *positive set constraint* is a formal inclusion $s \subseteq t$, where s and t are set expressions. For notational convenience we allow equational constraints $s = t$, although inclusions and equations are interdefinable: $s \subseteq t$ is equivalent to $s \cup t = t$, and $s = t$ to $s \oplus t \subseteq 0$. A *negative set constraint* is the negation of a positive set constraint: $s \not\subseteq t$ or $s \neq t$. We use $\varphi, \psi, \ldots$ to denote set constraints and $\Phi, \Psi, \ldots$ to denote finite sets of set constraints.

3 Axioms of Termset Algebra

In [Koz93], the following axiomatization of the algebra of sets of ground terms was introduced:

$$f(\ldots, x \cup y, \ldots) = f(\ldots, x, \ldots) \cup f(\ldots, y, \ldots) \tag{1}$$

$$f(\ldots, x - y, \ldots) = f(\ldots, x, \ldots) - f(\ldots, y, \ldots) \tag{2}$$

$$\bigcup_{f \in \Sigma} f(1, \ldots, 1) = 1 \tag{3}$$

$$f(1, \ldots, 1) \cap g(1, \ldots, 1) = 0, \ f \neq g \tag{4}$$

$$f(x_1, \ldots, x_n) = 0 \Rightarrow \bigvee_{i=1}^{n} (x_i = 0) \tag{5}$$

$$\text{axioms of Boolean algebra} \tag{6}$$

The ellipses in (1) and (2) indicate that the explicitly given arguments occur in corresponding places, and that the implicit arguments in corresponding places agree. Models of the axioms are called *termset algebras*. The standard interpretation 2^{T_Σ}, where the Boolean operators have their usual set-theoretic interpretations and elements $f \in \Sigma_n$ are interpreted as

$$f : (2^{T_\Sigma})^n \rightarrow 2^{T_\Sigma}$$
$$f(A_1, \ldots, A_n) = \{f(t_1, \ldots, t_n) \,|\, t_i \in A_i,\ 1 \leq i \leq n\} \ ,$$

forms a model of these axioms.

Some immediate consequences of these axioms are

$$\begin{aligned}
f(\ldots, 0, \ldots) &= 0 & (7)\\
f(\ldots, \sim x, \ldots) &= f(\ldots, 1, \ldots) - f(\ldots, x, \ldots) & (8)\\
f(\ldots, x \oplus y, \ldots) &= f(\ldots, x, \ldots) \oplus f(\ldots, y, \ldots) & (9)\\
f(\ldots, x \cap y, \ldots) &= f(\ldots, x, \ldots) \cap f(\ldots, y, \ldots) & (10)\\
x \subseteq y \Rightarrow f(\ldots, x, \ldots) &\subseteq f(\ldots, y, \ldots) \ . & (11)
\end{aligned}$$

Also, a *generalized DeMorgan law* can be derived:

$$\begin{aligned}
\sim f(x_1, \ldots, x_n) = & \bigcup_{g \neq f} g(1, \ldots, 1) \\
& \cup \bigcup_{i=1}^{n} f(\underbrace{1, \ldots, 1}_{i-1}, \sim x_i, \underbrace{1, \ldots, 1}_{n-i}) \qquad (12)
\end{aligned}$$

The law intuitively says that a ground term *not* having head symbol f and i^{th} subterm satisfying x_i either has head symbol different from f or has head symbol f but one of its i^{th} subterms does not satisfy x_i. The law is useful for pushing occurrences of the negation operator $\sim$ inward.

4 Term Automata and Models

Following Courcelle [Cou83], we define *(Σ-)terms*.

Definition 1. Let ω denote the set of natural numbers and let Σ be a finite ranked alphabet. A (Σ-)*term* is a partial function $t : \omega^* \rightarrow \Sigma$ whose domain is nonempty, prefix-closed, and respects arities in the sense that if $t(\gamma)$ is defined then

$$\{i \mid t(\gamma i) \text{ is defined}\} = \{1, 2, \ldots, \text{arity}(t(\gamma))\} \ .$$

If α is in the domain of t, the subterm of t rooted at α is the the term $\lambda\beta.t(\alpha\beta)$. A term is (in)finite if its domain is (in)finite, and is *regular* if it has only finitely many subterms.

Example 1. The finite term $f(g(a), f(a, g(b)))$ is formally a partial map t with domain $\{\epsilon, 1, 2, 11, 21, 22, 221\}$ such that $t(\epsilon) = t(2) = f$, $t(1) = t(22) = g$, $t(11) = t(21) = a$, and $t(221) = b$. The infinite term $f(a, f(a, f(a, \ldots)))$ is formally a map s whose domain is the infinite set described by the regular expression $2^* + 2^*1$ such that $s(\alpha) = f$ for $\alpha \in 2^*$ and $s(\alpha) = a$ for $\alpha \in 2^*1$. The infinite term s is regular since it has only two subterms, namely s and a.

4.1 Term Automata

It is well known that an infinite regular term can be represented by a finite labeled graph such that the infinite term is obtained by "unwinding" the graph (see [Cou83, Col82]). We use the automata-theoretic formulation introduced in [KPS92] of this idea.

Definition 2. A *term automaton* over Σ is a tuple $\mathcal{M} = (Q,\ \Sigma,\ \ell,\ \delta)$ where:

- Q is a set of *states* (not necessarily finite)
- Σ is a ranked alphabet
- $\ell : Q \to \Sigma$ is a *labeling*
- $\delta : Q \times \omega \to Q$ is a partial function such that for all $q \in Q$,

$$\{i \mid \delta(q,i) \text{ is defined}\} = \{1, 2, \ldots, \text{arity}(\ell(q))\}\ .$$

The function δ extends uniquely to a partial function $\widehat{\delta} : Q \times \omega^* \to Q$ according to the inductive definition

$$\begin{aligned}\widehat{\delta}(q,\epsilon) &= q \\ \widehat{\delta}(q,\gamma i) &= \delta(\widehat{\delta}(q,\gamma), i)\ ,\end{aligned}$$

with the understanding that δ is strict (undefined if one of its arguments is undefined). For each $q \in Q$, the partial function

$$t_q = \lambda\gamma.\ell(\widehat{\delta}(q,\gamma))$$

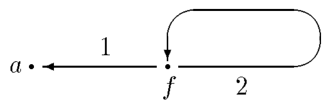
is a Σ-term in the sense of Definition 1. Notice that t_p may equal t_q even though $p \neq q$.

Every term in the sense of Definition 1 is t_q for some state q of some term automaton. In fact, $t = t_t$ in the syntactic term automaton

$$I_\Sigma = (\{\Sigma\text{-terms}\},\ \Sigma,\ \ell,\ \delta)$$

where $\ell(t) = t(\epsilon)$ and $\delta(t,i) = \lambda\gamma.t(i\gamma)$, $1 \le i \le \text{arity}(\ell(t))$. In this sense the notion of term automaton (Definition 2) is a generalization of the notion of term (Definition 1).

A term is regular if and only if it is t_q for some state q of some finite term automaton [KPS93, Lemma 8]. For example, if q is the state labeled f in the term automaton



then t_q is the infinite regular term s of Example 1.

A term automaton $\mathcal{M}$ is *closed* if for any $f \in \Sigma_n$ and $q_1, \ldots, q_n \in Q$ there exists a $q \in Q$ such that

$$\ell(q) = f \text{ and } \delta(q,i) = q_i\ ,\ 1 \le i \le n\ . \tag{13}$$

A *model* is a closed term automaton $\mathcal{M}$. We refer to the states of $\mathcal{M}$—rather then their associated partial functions t_q—as the *terms* of $\mathcal{M}$, and use the notation $\mathbf{t} \in \mathcal{M}$ to indicate $\mathbf{t} \in Q$. A term $\mathbf{t}'$ of $\mathcal{M}$ is a *subterm* of $\mathbf{t}$ at depth k if there exist a $\gamma \in \omega^k$ such that $\widehat{\delta}(\mathbf{t}, \gamma) = \mathbf{t}'$. A term $\mathbf{t}$ of $\mathcal{M}$ is (in)finite if $t_{\mathbf{t}}$ is (in)finite, and said to be labeled by t' if $t_{\mathbf{t}} = t'$. The model is *standard* if the function $q \mapsto t_q : Q \to T_\Sigma$ is a bijection. We denote a standard model by T_Σ.

Remark. For any term automaton $\mathcal{M} = (Q,\ \Sigma,\ \ell,\ \delta)$ there is a closed term automaton $\mathcal{M}' = (Q',\ \Sigma,\ \ell',\ \delta')$ such that $Q \subseteq Q'$, ℓ' and δ' coincide with ℓ and δ on states from Q, and Q' is a minimal set of states—with respect to subset inclusion—with these properties; $\mathcal{M}'$ is said to be a *minimal closure* of $\mathcal{M}$. $\mathcal{M}'$ can be obtained as follows: Let $\mathcal{M}_0 = \mathcal{M}$ and let $\mathcal{M}_{i+1}$ be obtained from $\mathcal{M}_i$ by adding exactly one new term $\mathbf{t}$ to Q_i for every $f \in \Sigma_n$ and $\mathbf{t}_1, \ldots, \mathbf{t}_n \in Q_i$ for which (13) doesn't hold. ℓ_{i+1} is the extension of ℓ_i that maps $\mathbf{t}$ to f and δ_{i+1} is the extension of δ_i that maps $(\mathbf{t}, i)$ to $\mathbf{t}_i$, $1 \le i \le n$. Define $\mathcal{M}'$ as the ω-limit of these term automata.

4.2 Term Automata and Set-Theoretic Termset Algebras

Let $\mathcal{M}$ be the term automaton $(Q,\ \Sigma,\ \ell,\ \delta)$. For $f \in \Sigma_n$, define the partial function $R_f^{\mathcal{M}} : Q \to Q^n$ and the set-theoretic function $f^{\mathcal{M}} : (2^Q)^n \to 2^Q$ by

$$R_f^{\mathcal{M}}(q) = \begin{cases} (\delta(q,1), \ldots, \delta(q,n)) \text{ , if } \ell(q) = f \\ \text{undefined ,} \qquad\qquad\quad \text{otherwise.} \end{cases} \tag{14}$$

$$\begin{aligned} f^{\mathcal{M}}(A_1, \ldots, A_n) &= \{q \in Q \mid \ell(q) = f \text{ and } \delta(q,i) \in A_i,\ 1 \le i \le n\} \\ &= (R_f^{\mathcal{M}})^{-1}(A_1 \times \cdots \times A_n)\ . \end{aligned} \tag{15}$$

Set expressions are interpreted over 2^Q, the powerset of Q, which forms an algebra of signature Σ+B, where the Boolean operators have their usual set-theoretic interpretations and elements $f \in \Sigma$ are interpreted as $f^{\mathcal{M}}$. If $\mathcal{M}$ is closed, one can show that this gives a termset algebra. Such an algebra, or a subalgebra of such an algebra, is called a *set-theoretic termset algebra*.

Let $\mathcal{M}$ be a model. A *set valuation* over $\mathcal{M}$ is a map

$$\sigma : X \to 2^Q$$

assigning a subset of terms of $\mathcal{M}$ to each variable in X. We can extend any set valuation σ uniquely to a (Σ+B)-homomorphism

$$\sigma : T_{\Sigma+\mathrm{B}}(X) \to 2^Q$$

by induction on the structure of set expressions in the usual way. A set valuation σ over $\mathcal{M}$ satisfies the positive set constraint $s \subseteq t$ if $\sigma(s) \subseteq \sigma(t)$, and satisfies the negative set constraint $s \not\subseteq t$ if $\sigma(s) \not\subseteq \sigma(t)$. We write $\sigma \models_{\mathcal{M}} \Phi$ if σ satisfies all set constraints in Φ; Φ is said to be *satisfiable* in $\mathcal{M}$ and σ a *solution* to Φ. The set Φ is satisfiable if it is satisfiable over some model. We write $\Phi \models_{\mathcal{M}} \Psi$ if $\sigma \models_{\mathcal{M}} \Phi$ implies $\sigma \models_{\mathcal{M}} \psi$ for some $\psi \in \Psi$. When no confusion is possible, we suppress the subscript $\mathcal{M}$.

5 Systems in Normal Form and Solutions

Let $X' \subseteq X$. *Positive (negative) literals* from X' are expressions x ($\sim x$) for $x \in X'$. A maximal conjunction of literals from X' is a conjunction of positive and negative literals from X', where each variable in X' occurs exactly once.

A triple $(t_B,\ \Phi,\ \Delta)$ is a system of set constraints in *normal form* (or just a system in normal form) if there is a finite set $X' \subseteq X$ such that (i) $t_B \in T_{\mathrm{B}}(X')$ is of the form $\cup_{\alpha \in U} \alpha$, for some set U of maximal conjunctions of literals from X', (ii) for each $f \in \Sigma_n$ and $\alpha_1, \ldots, \alpha_n \in U$ there is exactly one set constraint in Φ of the form $f(\alpha_1, \ldots, \alpha_n) \subseteq \bigcup_{\alpha \in E_{f(\alpha_1,\ldots,\alpha_n)}} \alpha$, where $E_{f(\alpha_1,\ldots,\alpha_n)} \subseteq U$, and (iii) Δ is a finite set of Boolean expressions $\{\bigcup_{\alpha \in I_1} \alpha, \ldots, \bigcup_{\alpha \in I_m} \alpha\}$, where $I_k \subseteq U$ for $1 \leq k \leq m$. The set U is referred to as the set of atoms[2] specified by t_B.

The triple $(t_B,\ \Phi,\ \Delta)$ corresponds to the set of set constraints $\{t_B = 1\} \cup \Phi \cup \{\bigcup_{\alpha \in I_1} \alpha \neq 0, \ldots, \bigcup_{\alpha \in I_m} \alpha \neq 0\}$ and is said to be (un)satisfiable if the latter is. A set valuation satisfies $(t_B,\ \Phi,\ \Delta)$ if it satisfies the corresponding set constraints. If Δ is empty, we denote the system in normal form by $(t_B,\ \Phi)$ and call it a system of positive set constraints in normal form (or just a positive system in normal form). Every system of mixed positive and negative set constraints is equivalent to a system in normal form [AKW95].

Each positive system in normal form $(t_B,\ \Phi)$ has an associated hypergraph; the nodes are the elements of U and the hyperedges are specified by the sets $E_{f(\alpha_1,\ldots,\alpha_n)}$. Let $\mathcal{M}$ be a model. A *run* over $\mathcal{M}$ through the hypergraph is a function $\theta : Q \to U$ such that

$$\theta(\mathbf{t}) \in E_{f(\theta(\mathbf{t}_1),\ldots,\theta(\mathbf{t}_n))} \ ,$$

where $\ell(\mathbf{t}) = f \in \Sigma_n$ and $\delta(\mathbf{t}, i) = \mathbf{t}_i$, for $1 \leq i \leq n$. Each subset $U' \subseteq U$ induces a subhypergraph by restricting the nodes and hyperedges to U'. The subhypergraph induced by U' is *closed* if for each $f \in \Sigma_n$ and $\alpha_1, \ldots, \alpha_n \in U'$ the set $E_{f(\alpha_1,\ldots,\alpha_n)} \cap U'$ is nonempty. It can be proved that $(t_B,\ \Phi)$ is satisfiable over a standard model if and only if there is a nonempty $U' \subseteq U$ that induces a closed subhypergraph in the hypergraph associated with $(t_B,\ \Phi)$. Intuitively, from a run θ one can obtain a set valuation σ_θ over a standard model satisfying $(t_B,\ \Phi)$, and—vice versa—from a set valuation σ satisfying $(t_B,\ \Phi)$ one can obtain a run θ_σ over a standard model through the hypergraph associated with $(t_B,\ \Phi)$. For details see [AKVW93, Koz93, Koz95].

6 Completeness and Incompleteness

In this section we give a Gentzen-style axiomatization for sequents $\Phi \vdash \Psi$, based on the axioms of termset algebra. The intended interpretation of the sequent $\Phi \vdash \Psi$ is that if all the constraints in Φ hold of some model, then at least one of the constraints Ψ holds in that model. We prove (i) completeness over standard

[2] The elements of U are the atoms of the free Boolean algebra on generators X' modulo t_B.

models for satisfiability of positive set constraints (if Φ is unsatisfiable, then Φ is refutable, *i.e.*, $\Phi \vdash \bot$ is derivable), (ii) incompleteness over standard models for satisfiability of mixed positive and negative set constraints (*i.e.*, not all true sequents $\Phi \vdash \Psi$ are derivable), and (iii) completeness over nonstandard models.

Any set constraint can be represented as an inclusion $s \subseteq t$, or an equation $u = 0$, or an equation $v = 1$. In the following, any set expression s occurring in a context expecting a set constraint denotes the set constraint $s = 1$. An inclusion $s \subseteq t$ can then be represented as the term $\sim s \cup t$, denoting the set constraint $\sim s \cup t = 1$, and an equation $s = t$ as the term $(\sim s \cup t) \cap (\sim t \cup s)$. A set Φ denotes the conjunction or disjunction of its elements, depending on whether it occurs on the left or right side of a $\vdash$, respectively. A comma denotes conjunction or disjunction, depending on whether it occurs on the left or right side of a $\vdash$, respectively. We use $\bot$ for the empty disjunction on the right side of $\vdash$; $\bot$ can be read as 0. The rules are:

$$\Phi \vdash \Phi \text{ (ident)} \qquad \frac{\Phi \vdash \Psi}{\Phi', \Phi \vdash \Psi, \Psi'} \text{ (weakening)}$$

$$\frac{\Phi, \sim t_i \vdash \Psi, 1 \leq i \leq n}{\Phi, \sim f(t_1, \ldots, t_n) \vdash \Psi} \text{ (f-intro $\vdash$)}$$

$$\frac{\Phi, s, t \vdash \Psi}{\Phi, s \cap t \vdash \Psi} \text{ ($\cap$-intro $\vdash$)} \qquad \frac{\Phi, s \cap t \vdash \Psi}{\Phi, s, t \vdash \Psi} \text{ ($\cap$-elim $\vdash$)}$$

$$\frac{\Phi, \varphi[t \leftarrow t'], t = t' \vdash \Psi}{\Phi, \varphi, t = t' \vdash \Psi} \text{ (substitution $\vdash$)}$$

$$\frac{\Phi, t = t' \vdash \psi[t \leftarrow t'], \Psi}{\Phi, t = t' \vdash \psi, \Psi} \text{ ($\vdash$ substitution)}$$

For x not in Φ, t:

$$\frac{\Phi, x = t \vdash \Psi}{\Phi \vdash \Psi} \text{ (x-elim $\vdash$)}$$

For any instance $s = t$ of the termset algebra axioms:

$$\frac{\Phi, s \vdash \Psi}{\Phi, t \vdash \Psi} \text{ (termset $\vdash$)} \qquad \frac{\Phi \vdash \Psi, s}{\Phi \vdash \Psi, t} \text{ ($\vdash$ termset)}$$

The sequents above and under a bar are referred to as the premises and conclusion of the rule, respectively. $\varphi[t \leftarrow t']$ denotes the substitution of all occurrences of the expression t in φ by the expression t'.

Derivation trees are inductively defined finite trees whose nodes are labeled with sequents $\Phi \vdash \Psi$. A single node labeled with any sequent $\Phi \vdash \Psi$ is a derivation tree, and if there exist derivation trees $\mathcal{T}_1, \ldots, \mathcal{T}_n$ whose roots are labeled with sequents matching the premises of a rule, then the tree whose root is labeled with the conclusion of that rule and has $\mathcal{T}_1, \ldots, \mathcal{T}_n$ as immediate subtrees is itself a derivation tree. A sequent $\Phi \vdash \Psi$ is *derivable from a set S* of sequents if and only if there is a derivation tree all of whose leaves are labeled by sequents in S and whose root is labeled $\Phi \vdash \Psi$. If S only contains sequents of the form $\Delta \vdash \Delta$ or $\Delta, \sim c \vdash \Gamma$ (corresponding to the rules (ident) and (f-intro $\vdash$) for $n = 0$, respectively), then the derivation tree is called a *tableau* and $\Phi \vdash \Psi$ is said to be *derivable*.

Example 2. As an example of how the rules are used, let us consider how $\Phi, \sim t \vdash \Psi$ can be derived from $\Phi, \sim f(\ldots, t, \ldots) \vdash \Psi$, hence it is not necessary to postulate as an axiom the corresponding rule

$$\frac{\Phi, \sim f(\ldots, t, \ldots) \vdash \Psi}{\Phi, \sim t \vdash \Psi} \ (f\text{-elim} \vdash)$$

Assume $f(\ldots, t, \ldots)$ is $f(t_1, \ldots, t_{i-1}, t, t_i, \ldots, t_{n-1})$ and let $x_1, \ldots, x_{n-1}$ be distinct new variables not occurring in $\Phi, f(\ldots, t, \ldots), \Psi$. A derivation (sketch) could be:

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\Phi, \sim f(\ldots, t, \ldots) \vdash \Psi}{\Phi, \sim t, \sim f(\ldots, t, \ldots), x_1 = t_1, \ldots, x_{n-1} = t_{n-1} \vdash \Psi}}{\Phi, \sim t, \sim f(x_1, \ldots, x_{i-1}, 0, x_i, \ldots, x_{n-1}), x_1 = t_1, \ldots, x_{n-1} = t_{n-1} \vdash \Psi}}{\Phi, \sim t, 1, x_1 = t_1, \ldots, x_{n-1} = t_{n-1} \vdash \Psi}}{\Phi, \sim t, 1 \vdash \Psi}}{\Phi, \sim t \cap 1 \vdash \Psi}}{\Phi, \sim t \vdash \Psi}$$

The rules applied—bottom-up—are (termset $\vdash$), ($\cap$-intro $\vdash$), (x-elim $\vdash$) (several times), (termset $\vdash$) ((7) applied to 1), (substitution $\vdash$) (several times, $\sim t$ can be rewritten into $t = 0$, substitute t for 0, then t_1 for x_1, *etc.*), and finally (weakening).

Lemma 3. *All rules are sound.*

Proof. The proof is straightforward. As an example, assume we are given a model $\mathcal{M}$. Let us consider the (f-intro $\vdash$) rule. Assume we have a set valuation σ (over $\mathcal{M}$) which satisfies $\Phi, \sim f(t_1, \ldots, t_n)$ and that $\Phi, \sim t_i \models \Psi$ holds for $1 \leq i \leq n$. Since $\sigma(f(t_1, \ldots, t_n)) = \varnothing$, we conclude by $\mathcal{M}$ being closed and the definition of set valuations that $\sigma(t_{i_0}) = \varnothing$ for some $1 \leq i_0 \leq n$. But then σ satisfies $\Phi, \sim t_{i_0}$, and by our assumptions σ must also satisfy a set constraint in Ψ. Hence, $\Phi, \sim f(t_1, \ldots, t_n) \models \Psi$. $\square$

The following theorem shows that the deductive system is complete over standard models for satisfiability of positive set constraints.

Theorem 4. *If a finite set of positive set constraints Φ is unsatisfiable in any standard model, then $\Phi \vdash \bot$ is derivable.*

Proof. We construct a tableau whose root is labeled $\Phi \vdash \bot$ in two stages. In the first stage we show how one can obtain an equivalent finite set of set constraints $\{t_B = 1\} \cup \Phi'$ from Φ, such that $(t_B,\ \Phi')$ is a positive system in normal form. Simultaneously, we show how to derive $\Phi \vdash \bot$ from $t_B, \Phi' \vdash \bot$. This is essentially a formalization of the normal form algorithm of [AKVW93] in terms of the sequent rules.

Given Φ, replace all occurrences of a subexpression $f(t_1, \ldots, t_n)$ in a set constraint in Φ by occurrences of a variable x and add the set constraints

$$\begin{aligned} x &= f(y_1, \ldots, y_n) \\ y_i &= t_i,\ 1 \le i \le n\ , \end{aligned} \tag{16}$$

where $x, y_1, \ldots, y_n$ are new variables. We refer to this as *flattening*. Repeat this until all set constraints are purely Boolean or of the form (16). Let Δ denote the obtained completely flattened set of set constraints. Notice that Δ is equivalent to Φ. Using (x-elim $\vdash$) and (substitution $\vdash$) we can derive $\Phi \vdash \bot$ from $\Delta \vdash \bot$.

Any set constraint of the form (16) in Δ can be replaced by two inclusions

$$f(y_1, \ldots, y_n) \subseteq x \tag{17}$$

$$\sim f(y_1, \ldots, y_n) \subseteq \sim x. \tag{18}$$

Applying the generalized DeMorgan law (12), the inclusion (18) is equivalent to

$$\bigcup_{\substack{g \neq f \\ g \in \Sigma}} g(1, \ldots, 1) \cup \bigcup_{i=1}^{n} f(\underbrace{1, \ldots, 1}_{i-1}, \sim y_i, \underbrace{1, \ldots, 1}_{n-i}) \subseteq \sim x$$

and can be replaced by the inclusions

$$\begin{aligned} g(1, \ldots, 1) &\subseteq \sim x \qquad g \neq f \\ f(\underbrace{1, \ldots, 1}_{i-1}, \sim y_i, \underbrace{1, \ldots, 1}_{n-i}) &\subseteq \sim x \qquad 1 \le i \le n\ . \end{aligned} \tag{19}$$

Let Δ' denote the current set of set constraints. Since $x = f(y_1, \ldots, y_n)$ may be represented as the term $(\sim x \cup f(y_1, \ldots, y_n)) \cap (\sim f(y_1, \ldots, y_n) \cup x)$ in the sequents, use ($\cap$-intro $\vdash$) and (termset $\vdash$) to obtain the inclusions (17) and (18). Use (termset $\vdash$) to replace $\sim(\sim f(y_1, \ldots, y_n)) \cup \sim x$ (corresponding to (18)) by

$$\Big(\bigcap_{\substack{g \neq f \\ g \in \Sigma}} \sim g(1, \ldots, 1) \cup \sim x\Big) \cap \Big(\bigcap_{i=1}^{n} \sim f(\underbrace{1, \ldots, 1}_{i-1}, \sim y_i, \underbrace{1, \ldots, 1}_{n-i}) \cup \sim x\Big) \tag{20}$$

and use ($\cap$-intro $\vdash$) to split this term into terms

$$
\begin{array}{rl}
\sim g(1,\ldots,1) \cup \sim x & g \neq f \\
\sim f(\underbrace{1,\ldots,1}_{i-1}, \sim y_i, \underbrace{1,\ldots,1}_{n-i}) \cup \sim x & 1 \leq i \leq n \ ,
\end{array}
$$

corresponding to the inclusions (19). This shows that $\Delta \vdash \bot$ can be derived from $\Delta' \vdash \bot$.

Let X' denote the set of variables occurring in Δ'. At this point, Δ' only contains either purely Boolean set constraints or set constraints of the form

$$
f(x_1, \ldots, x_n) \subseteq x \tag{21}
$$

where $x_1, \ldots, x_n, x$ are either positive or negative literals from X' or the constant 1. Collect all purely Boolean set constraints and rewrite them, using the laws of Boolean algebra, into one equivalent Boolean set constraint $\bigcup_{\alpha \in U} \alpha = 1$, where U is a set of maximal conjunctions of literals from X'. Let t_B denote the left side of this set constraint. The current set of set constraints is now of the form $\{t_B = 1\} \cup \Delta''$, where all set constraints in Δ'' are of the form (21). Using ($\cap$-elim $\vdash$) to collect all Boolean terms into one Boolean term and (termset $\vdash$) to replaced it by t_B, we derive $\Delta' \vdash \bot$ from $t_B, \Delta'' \vdash \bot$.

For $x \in X'$, let $U(x)$ and $U(\sim x)$ denote the set of atoms from U in which x occurs positively and negatively, respectively. Also, let $U(1)$ denote $\bigcup_{\alpha \in U} \alpha$. Using the set constraint $t_B = 1$ we can replace each set constraint of the form (21) in $\{t_B = 1\} \cup \Delta''$ by

$$
f(\bigcup_{\alpha \in U(x_1)} \alpha, \ldots, \bigcup_{\alpha \in U(x_n)} \alpha) \subseteq \bigcup_{\alpha \in U(x)} \alpha \ , \tag{22}
$$

which can be rewritten as separate inclusions

$$
f(\alpha_1, \ldots, \alpha_n) \subseteq \bigcup_{\alpha \in U(x)} \alpha \ , \ \alpha_i \in U(x_i), \ 1 \leq i \leq n \ . \tag{23}
$$

For any $f \in \Sigma_n$ and $\alpha_1, \ldots, \alpha_n \in U$, collect all inclusions of the form (23) and rewrite them into one

$$
f(\alpha_1, \ldots, \alpha_n) \subseteq \bigcup_{\alpha \in E_{f(\alpha_1, \ldots, \alpha_n)}} \alpha \ , \tag{24}
$$

where $E_{f(\alpha_1, \ldots, \alpha_n)}$ is the intersection of all sets $U(x)$ from the right sides of the collected inclusions. The resulting set of set constraints $\{t_B = 1\} \cup \Phi'$ is equivalent to Φ and $(t_B, \ \Phi')$ is a positive system in normal form.

The rewriting of inclusion of the form (21) into inclusions of the form (23) can be done using (termset $\vdash$) to rewrite x_i into $x_i \cap 1$, $1 \leq i \leq n$, (substitution

$\vdash$) to substitute t_B for 1, and (termset $\vdash$) and ($\cap$-intro $\vdash$) to obtain the separate inclusions. To obtain the inclusions corresponding to (24), use ($\cap$-elim $\vdash$) to collect the appropriate inclusions and (termset $\vdash$) to rewrite them into (24). Hence we can derive $t_B, \Delta'' \vdash \bot$ from $t_B, \Phi' \vdash \bot$. This completes the derivation of $\Phi \vdash \bot$ from $t_B, \Phi' \vdash \bot$.

For the second stage we construct a tableau with root $t_B, \Phi' \vdash \bot$, which together with the derivation of $\Phi \vdash \bot$ from $t_B, \Phi' \vdash \bot$ completes the proof.

Since $(t_B,\ \Phi')$ is a positive system in normal form and equivalent to Φ, there must exist an inclusion in Φ' of the form (24) with $E_{f(\alpha'_1,\ldots,\alpha'_n)} = \varnothing$, else the corresponding hypergraph would be closed and Φ would be satisfiable in a standard model.

If there exists such an inclusion with $n = 0$, then we have found the desired tableau. Otherwise, use (f-intro $\vdash$) to derive $t_B, \sim f(\alpha'_1, \ldots, \alpha'_n), \Phi'' \vdash \bot$ from $t_B, \sim \alpha'_i, \Phi'' \vdash \bot, 1 \leq i \leq n$, where $\sim f(\alpha'_1, \ldots, \alpha'_n), \Phi''$ is Φ'. Each of these sequents represents the discarding of one of the atoms in U. Consider any $1 \leq i \leq n$ and let U_i denote the set $U - \{\alpha'_i\}$ and t_{B_i} denote the conjunction of all elements in U_i. Use ($\cap$-elim $\vdash$) and (termset $\vdash$) to derive the sequent $t_B, \sim \alpha'_i, \Phi'' \vdash \bot$ from the sequent $t_{B_i}, \sim \alpha'_i, \Phi'' \vdash \bot$, which itself can be derived from $t_{B_i}, \Phi''_i \vdash \bot$, where Φ''_i contains all inclusions of the form (24) in which α'_i does not occur on the left of the inclusion and α'_i has been removed from the sets $E_{f(\alpha_1,\ldots,\alpha_n)}$; this can be done using (weakening), (substitution $\vdash$), and (termset $\vdash$). Notice that $(t_{B_i},\ \Phi''_i)$ is a positive system in normal form and is unsatisfiable because $(t_B,\ \Phi')$ is unsatisfiable.

By repeatedly applying the above procedure to all $t_{B_i}, \Phi''_i \vdash \bot$ we conclude that there must exist a tableau deriving $t_B, \Phi' \vdash \bot$ from sequents of the form $\Psi', \sim c \vdash \bot$. $\square$

Now suppose we are given a set of mixed positive and negative set constraints $\Phi = \{s_1 = t_1, \ldots, s_n = t_n\} \cup \{s'_1 \neq t'_1, \ldots, s'_m \neq t'_m\}$. Observe that Φ is unsatisfiable if and only if $\{s_1 = t_1, \ldots, s_n = t_n\} \models \{s'_1 = t'_1, \ldots, s'_m = t'_m\}$. The following theorem shows that the deductive system is incomplete over standard models for satisfiability of mixed positive and negative set constraints.

Theorem 5. *The axiomatization is incomplete for systems of mixed positive and negative set constraints over standard models.*

Proof. The sequent $x = f(x) \models x = 0$ certainly holds in all standard models. However, $x = f(x) \vdash x = 0$ cannot be derived, since the rules are sound for nonstandard models as well, and if infinite terms are allowed then $x = f(x) \models x = 0$ is no longer valid: in any model containing an infinite term labeled $f(f(f(\ldots)))$, the set of terms labeled $f(f(f(\ldots)))$ is a nontrivial solution to the set constraint $x = f(x)$. $\square$

We continue by considering nonstandard models.

Lemma 6. *A system of set constraints in normal form $(t_B,\ \Phi,\ \Delta)$, where $\Delta = \{\bigcup_{\alpha \in I_1} \alpha, \ldots, \bigcup_{\alpha \in I_m} \alpha\}$, is satisfiable if and only if there exists a set $U' \subseteq U$ such that*

$$\forall f \in \Sigma_n .\, \forall \alpha_1, \ldots, \alpha_n \in U' .\, E_{f(\alpha_1,\ldots,\alpha_n)} \cap U' \neq \varnothing \ , \tag{25}$$

$$\forall \alpha \in U' .\, \exists f \in \Sigma_n .\, \exists \alpha_1, \ldots, \alpha_n \in U' .\, \alpha \in E_{f(\alpha_1,\ldots,\alpha_n)} \ , \ \textit{and} \tag{26}$$

$$\forall 1 \leq k \leq m .\, I_k \cap U' \neq \varnothing \ , \tag{27}$$

where U are the atoms corresponding to t_B.

Proof. For the "only if" direction, assume $(t_B,\ \Phi,\ \Delta)$ is satisfiable and let $\sigma : X \to 2^Q$ denote a satisfying set valuation over $\mathcal{M}$. Then $U' = \{\alpha \in U \mid \sigma(\alpha) \neq \varnothing\}$ satisfies the properties (25)–(27). To see this, notice (i) that (25) follows from σ being a satisfying set valuation, the definition of U', and axiom (5), (ii) that (26) follows from $\sigma(1) = \sigma(t_B) = \sigma(\bigcup_{\alpha \in U} \alpha) = \sigma(\bigcup_{\alpha \in U'} \alpha)$ and axiom (3), and (iii) that (27) follows from σ being a satisfying set valuation and the definition of U'.

For the "if" direction, assume $U' \subseteq U$ satisfies (25)–(27). Since U' induces a closed subhypergraph in the hypergraph associated with $(t_B,\ \Phi)$ there exist set valuations over standard models—whose associated runs map terms into U'—satisfying $(t_B,\ \Phi)$. Let $U'' \subseteq U'$ be the set of atoms α for which there exists such a set valuation σ with $\sigma(\alpha) \neq \varnothing$. Let $l = |U''|$, and let $\sigma_i : X \to 2^{Q_i}$ be set valuations over standard models $\mathcal{M}_i$, $1 \leq i \leq l$, satisfying $(t_B,\ \Phi)$ such that $U'' = \{\alpha \in U' \mid \exists 1 \leq i \leq l .\, \sigma_i(\alpha) \neq \varnothing\}$. Moreover, we may assume that the states of the standard models $\mathcal{M}_1, \ldots, \mathcal{M}_l$ are all mutually disjoint. Note that there exists a model $\mathcal{M}$ whose set of terms (states) contains $\bigcup_{i=1}^{l} Q_i$ and is minimal with respect to subset-inclusion. Moreover, its functions ℓ and δ restricted to Q_i coincide with ℓ_i and δ_i; see the remark in §4.1. Let $\theta_i : Q_i \to U''$, $1 \leq i \leq l$, be the runs corresponding to the set valuations σ_i, *i.e.*, for $\mathbf{t} \in Q_i$, $\theta_i(\mathbf{t})$ is the unique α such that $\mathbf{t} \in \sigma_i(\alpha)$. We define a run $\varrho : Q \to U''$ over $\mathcal{M}$, whose image is U'', as the limit of a chain of partial functions from Q to U''. Let

$$\varrho_0(\mathbf{t}) = \begin{cases} \theta_i(\mathbf{t}) \ , \ \text{if } \mathbf{t} \in Q_i,\ 1 \leq i \leq l, \\ \text{undefined, otherwise.} \end{cases}$$

Given ϱ_j, define ϱ_{j+1} as follows. For $\mathbf{t} \in Q$,

- if ϱ_j is defined on $\mathbf{t}$, then $\varrho_{j+1}(\mathbf{t})$ is defined as $\varrho_j(\mathbf{t})$
- else, if $\ell(\mathbf{t}) = f \in \Sigma_n$ and $\delta(\mathbf{t}, i) = \mathbf{t}_i$ on which ϱ_j is defined, for $1 \leq i \leq n$, then pick any $\alpha \in E_{f(\varrho_j(\mathbf{t}_1),\ldots,\varrho_j(\mathbf{t}_n))} \cap U'$ and define $\varrho_{j+1}(\mathbf{t}) = \alpha$
- otherwise, ϱ_{j+1} is undefined on $\mathbf{t}$.

Now define ϱ as the limit of $\varrho_0, \varrho_1, \ldots$ It is easy to see that

$$\varrho(\mathbf{t}) \in E_{f(\varrho(\mathbf{t}_1),\ldots,\varrho(\mathbf{t}_n))} \cap U' \tag{28}$$

for any $\mathbf{t} \in Q$, where $\ell(\mathbf{t}) = f \in \Sigma_n$ and $\delta(\mathbf{t}, i) = \mathbf{t}_i$, for $1 \leq i \leq n$. Hence, ϱ is a run in the closed subhypergraph induced by U' and the corresponding set valuation $\varsigma_\varrho : X \to Q$ satisfies $(t_B,\ \Phi)$.

If $U'' = U'$ the set valuation satisfies $(t_B,\ \Phi,\ \Delta)$. So assume $U''' = U' \backslash U''$ is nonempty. Pick any $\alpha_{j_1} \in U'''$. We construct a finite tree structure $\mathcal{T}_{j_1}$ whose nodes are labeled by symbols from Σ. The tree structure is expanded from the root and down as long as certain conditions are met. Also, to each node of the tree we associate an element from U'. From $\mathcal{T}_{j_1}$ we obtain a new term $\mathbf{t}_{j_1}$ which will be added to the terms of $\mathcal{M}$. The term $\mathbf{t}_{j_1}$ will then be mapped to α_{j_1} by an extension of ς_ϱ.

By (26) there exist $f \in \Sigma_n$ and $\alpha_1, \ldots, \alpha_n \in U'$ such that $\alpha_{j_1} \in E_{f(\alpha_1,\ldots,\alpha_n)}$. The root of the tree structure is labeled by f and α_{j_1} is the associated atom.

For all $1 \leq i \leq n$ such that $\varsigma_\varrho(\alpha_i) \neq \varnothing$, pick a $\mathbf{t}_i \in \varsigma_\varrho(\alpha_i)$. Such a $\mathbf{t}_i$ corresponds to the i^{th} child of the root. The atom associated with the node $\mathbf{t}_i$ is $\varrho(\mathbf{t}_i)$. The nodes $\mathbf{t}_i$ are not expanded further and are referred to as $\mathcal{M}$-nodes.

For all $1 \leq i \leq n$ such that $\varsigma_\varrho(\alpha_i) = \varnothing$, add a new i^{th} child n whose associated atom is α_i. If there is another node n' on the path from n to the root whose associated atom is α_i, then label n by the symbol $f \in \Sigma$ that labels n'. The node n is not be expanded further; n is referred to as a *repeat-node* and n' as its *twin-node*.

Repeat the above procedure for the leaves n which are neither $\mathcal{M}$-nodes nor repeat-nodes.

Since U' is finite, we obtain a finite tree structure $\mathcal{T}_{j_1}$, all of whose internal nodes are labeled by symbols in Σ whose arities respect the branching structure. Moreover, any path from the root either ends at an $\mathcal{M}$-node or in a repeat-node. If there are any repeat nodes, the tree structure corresponds to an infinite regular term.

From $\mathcal{M}$ we obtain a new term automaton $\mathcal{M}'_1$ by adding new nodes to Q for all nodes of $\mathcal{T}_{j_1}$ that are not $\mathcal{M}$-nodes or repeat-nodes and by defining ℓ'_1 and δ'_1 to be the obvious extensions of ℓ and δ obtained by $\mathcal{T}_{j_1}$, when repeat-nodes are identified with their twin-nodes. Also, $\mathcal{T}_{j_1}$ permits ϱ to be extended to a function $\varrho'_1 : Q'_1 \rightarrow U'$ such that the inclusion (28) is still valid if ϱ'_1 is defined on the occurring terms. Notice that this function is not a run since $\mathcal{M}'_1$ is not a model.

Applying the same procedure for the remaining atoms in U''' we obtain a sequence of term automata $\mathcal{M}, \mathcal{M}'_1, \ldots, \mathcal{M}'_p$ and a corresponding sequence of functions $\varrho, \varrho'_1, \ldots, \varrho'_p$, where $p = |U'|$, each one extending the previous in the sequence as described above (except for $\mathcal{M}$ and ϱ).

Let $\mathcal{M}''$ be a minimal closure of $\mathcal{M}'_p$. We define a run $\theta_{\mathcal{M}''} : Q'' \rightarrow U'$ as the limit of a chain of partial functions from Q'' to U'. Let $\eta_0 = \varrho'_p$ and define η_{i+1} from η_i as follows. For $\mathbf{t} \in Q''$,

- if η_i is defined on $\mathbf{t}$, then $\eta_{i+1}(\mathbf{t})$ is defined as $\eta_i(\mathbf{t})$
- else, if $\ell''(\mathbf{t}) = f \in \Sigma_n$ and $\delta''(\mathbf{t}, i) = \mathbf{t}_i$ on which η_i is defined, for $1 \leq i \leq n$, then pick any $\alpha \in E_{f(\eta_i(\mathbf{t}_1),\ldots,\eta_i(\mathbf{t}_n))} \cap U'$ and define $\eta_{i+1}(\mathbf{t}) = \alpha$
- otherwise, η_{i+1} is undefined on $\mathbf{t}$.

Define $\theta_{\mathcal{M}''}$ as the limit of $\eta_0, \eta_1, \ldots$ Since $\mathcal{M}''$ is the minimal closure of $\mathcal{M}'_p$ and ϱ'_p is defined on all of Q'_p, any $\mathbf{t} \in Q'' \backslash Q'_p$ has the property that for some natural number k, ϱ'_p is defined on all subterms of $\mathbf{t}$ at depth k or more. This ensures that $\theta_{\mathcal{M}''}$ is defined everywhere on Q''. It is easy to see that

$$\theta_{\mathcal{M}''}(f(\mathbf{t})) \in E_{f(\theta_{\mathcal{M}''}(\mathbf{t}_1),\ldots,\theta_{\mathcal{M}''}(\mathbf{t}_n))} \cap U' \tag{29}$$

for any $\mathbf{t} \in Q''$, where $\ell''(\mathbf{t}) = f \in \Sigma_n$ and $\delta''(\mathbf{t}, i) = \mathbf{t}_i$, for $1 \leq i \leq n$. Hence, $\theta_{\mathcal{M}''}$ is a run through the hypergraph associated with $(t_B,\ \Phi)$ and the set valuation $\sigma_{\theta_{\mathcal{M}''}}$ corresponding to $\theta_{\mathcal{M}''}$ satisfies $(t_B,\ \Phi,\ \Delta)$, since the image of $\theta_{\mathcal{M}''}$ is U' and (27) holds. □

The last theorem shows that our deductive system is complete for satisfiability of mixed positive and negative set constraints.

Theorem 7. *If a finite set of mixed positive and negative set constraints*

$$\{s_1 = t_1, \ldots, s_n = t_n\} \cup \{s'_1 \neq t'_1, \ldots, s'_m \neq t'_m\}$$

is unsatisfiable, then

$$s_1 = t_1, \ldots, s_n = t_n \vdash s'_1 = t'_1, \ldots, s'_m = t'_m$$

is derivable.

Proof. Assume $\{s_1 = t_1, \ldots, s_n = t_n\} \cup \{s'_1 \neq t'_1, \ldots, s'_m \neq t'_m\}$ is not satisfiable in any model. We show how to derive

$$s_1 = t_1, \ldots, s_n = t_n \vdash s'_1 = t'_1, \ldots, s'_m = t'_m\ . \tag{30}$$

Notice that by repeatedly using ($\vdash$ termset), (x-elim $\vdash$), and ($\vdash$ substitution) we can derive (30) from

$$\begin{gathered} s_1 = t_1, \ldots, s_n = t_n, \\ x_1 = (s'_1 \cap \sim t'_1) \cup (\sim s'_1 \cap t'_1), \ldots, x_m = (s'_m \cap \sim t'_m) \cup (\sim s'_m \cap t'_m) \vdash \\ x_1 = 0, \ldots, x_m = 0\ , \end{gathered} \tag{31}$$

where $x_1, \ldots, x_m$ are new variables. Now apply the procedure from the proof of Theorem 4 to derive (31) from

$$t_B, \Phi \vdash x_1 = 0, \ldots, x_m = 0\ , \tag{32}$$

where $(t_B,\ \Phi)$ is a positive system in normal form such that $\{t_B = 1\} \cup \Phi$ is equivalent to $s_1 = t_1, \ldots, s_n = t_n, x_1 = (s'_1 \cap \sim t'_1) \cup (\sim s'_1 \cap t'_1), \ldots, x_m = (s'_m \cap \sim t'_m) \cup (\sim s'_m \cap t'_m)$.

Let U denote the set of atoms specified by t_B. Applying ($\vdash$ substitution) and ($\vdash$ termset) we derive (32) from

$$t_B, \Phi \vdash \bigcup_{\alpha \in I_1} \alpha = 0, \ldots, \bigcup_{\alpha \in I_m} \alpha = 0\ , \tag{33}$$

where I_j is $U(x_j)$, the set of atoms in U in which x_j occurs positively. Notice that $(t_B,\ \Phi,\ \{\bigcup_{\alpha \in I_1} \alpha, \ldots, \bigcup_{\alpha \in I_m} \alpha\})$ is a system in normal form. If the set constraints corresponding to the left of $\vdash$ in (33) are not satisfiable, we can derive

$$t_B, \Phi \vdash \bot\ , \tag{34}$$

using the technique from Theorem 4, and using (weakening) we can derive (33). In fact, in the following, whenever the set constraints to the left of a $\vdash$ in any sequent considered are unsatisfiable, we conclude that the sequent is derivable. So assume the $(t_B,\ \Phi)$ is satisfiable. Using (termset $\vdash$), ($\cap$-elim $\vdash$), and (3) we derive (33) from

$$t_B, \Phi, 1 = \bigcup_{\substack{f \in \Sigma \\ \alpha_1, \ldots, \alpha_n \in U}} f(\alpha_1, \ldots, \alpha_n) \vdash \bigcup_{\alpha \in I_1} \alpha = 0, \ldots, \bigcup_{\alpha \in I_m} \alpha = 0\ , \tag{35}$$

which can be derived using (termset $\vdash$), ($\cap$-intro $\vdash$), ($\cap$-elim $\vdash$), and (weakening) from

$$t_B, \Phi, 1 = \bigcup_{\substack{f \in \Sigma \\ \alpha_1, \ldots, \alpha_n \in U}} \bigcup_{\alpha \in E_{f(\alpha_1, \ldots, \alpha_n)}} \alpha \vdash \bigcup_{\alpha \in I_1} \alpha = 0, \ldots, \bigcup_{\alpha \in I_m} \alpha = 0\ , \tag{36}$$

which again can be derived from

$$t'_B, \Phi' \vdash \bigcup_{\alpha \in I_1} \alpha = 0, \ldots, \bigcup_{\alpha \in I_m} \alpha = 0\ , \tag{37}$$

using (substitution $\vdash$), (termset $\vdash$), and (weakening), where $U' \subseteq U$ is the set

$$\bigcup_{f \in \Sigma} \bigcup_{\alpha_1, \ldots, \alpha_n \in U} \bigcup_{\alpha \in E_{f(\alpha_1, \ldots, \alpha_n)}} \alpha\ ,$$

t'_B is the term $\bigcup_{\alpha \in U'} \alpha$, Φ' consists of all inclusions of the form

$$f(\alpha_1, \ldots, \alpha_n) \subseteq \bigcup_{\alpha \in E'_{f(\alpha_1, \ldots, \alpha_n)}} \alpha\ ,$$

where $f \in \Sigma_n$, $\alpha_1, \ldots, \alpha_n \in U'$, and $E'_{f(\alpha_1, \ldots, \alpha_n)} = E_{f(\alpha_1, \ldots, \alpha_n)} \cap U'$.

Using ($\vdash$ termset) and ($\vdash$ substitution) we can derive (37) from

$$t'_B, \Phi' \vdash \bigcup_{\alpha \in I'_1} \alpha = 0, \ldots, \bigcup_{\alpha \in I'_m} \alpha = 0\ , \tag{38}$$

where $I'_j = I_j \cap U'$, $1 \leq j \leq m$. Notice $(t'_B,\ \Phi',\ \{\bigcup_{\alpha \in I'_1} \alpha, \ldots, \bigcup_{\alpha \in I'_m} \alpha\})$ is a system in normal form and is satisfiable only if $(t_B,\ \Phi,\ \{\bigcup_{\alpha \in I_1} \alpha, \ldots, \bigcup_{\alpha \in I_m} \alpha\})$ is.

If any $I_j = \varnothing$, $1 \leq j \leq m$, (38) is easily seen to be derivable. So assume this is not the case. By repeating the steps from (33) to (38) we eventually obtain a sequent of the form (38), where some $I_j = \varnothing$ or all atoms $\alpha \in U'$ occur in some set $E'_{f(\alpha_1,\ldots,\alpha_n)}$. Now assume the latter is the case. Since $(t'_B,\ \Phi',\ \{\bigcup_{\alpha\in I'_1}\alpha,\ldots,\bigcup_{\alpha\in I'_m}\alpha\})$ is unsatisfiable, we conclude by Lemma 6 that there exist $f \in \Sigma$ and $\alpha_1,\ldots,\alpha_n \in U'$ such that $E'_{f(\alpha_1,\ldots,\alpha_n)} = \varnothing$. So using (termset $\vdash$) and (f-intro $\vdash$) we derive (38) from

$$t'_B, \Phi'', \sim\alpha_i \vdash \bigcup_{\alpha\in I'_1} \alpha = 0, \ldots \bigcup_{\alpha\in I'_m} \alpha = 0\ ,\ 1 \leq i \leq n\ , \tag{39}$$

where Φ'' is Φ' without the inclusion $f(\alpha_1,\ldots,\alpha_n) \subseteq \bigcup_{\alpha\in E'_{f(\alpha_1,\ldots,\alpha_n)}} \alpha$. The sequents in (39) whose set constraints to the left of $\vdash$ are unsatisfiable can be derived using the technique from the proof of Theorem 4. The remaining sequents can be derived by repeating steps similar to those used in phase two in the proof of Theorem 4 and those used to derive (37) from (38) to eliminate the atom α_i, and then repeating steps similar to those used to derived (33) from (39). This procedure eventually terminates, since atoms are being discarded in each iteration. □

7 Conclusion

In this paper we have introduced and investigated a deductive system for deriving sequents $\Phi \vdash \Psi$, where Φ and Ψ are finite sets of set constraints. Using standard and nonstandard models involving set-theoretic termset algebras as introduced in [Koz93], we have shown that the deductive system is (i) complete for restricted sequents of the form $\Phi \vdash \bot$ over standard models, (ii) incomplete for general sequents $\Phi \vdash \Psi$ over standard models, but (iii) complete for general sequents over nonstandard models.

Having chosen term automata as the basis for our models, we naturally get models that allow "multiple copies" of a term t, *i.e.* we may have $t_p = t_q$ for different states p and q of the term automaton. One natural and interesting question that remains is whether the system is complete for general sequents over models that forbid such "multiple copies" but allow infinite terms.

Acknowledgments

The support of the Danish Research Academy and Danish Research Council under contract SNF-journal number 11-0773 grant 5100.7314, the National Science Foundation under grant CCR-9317320, and the U.S. Army Research Office through the ACSyAM branch of the Mathematical Sciences Institute of Cornell University under contract DAAL03-91-C-0027 is gratefully acknowledged.

References

[AKVW93] Alexander Aiken, Dexter Kozen, Moshe Vardi, and Edward Wimmers. The complexity of set constraints. In E. Börger, Y. Gurevich, and K. Meinke, editors, *Proc. 1993 Conf. Computer Science Logic (CSL'93)*, volume 832 of *Lect. Notes in Comput. Sci.*, pages 1–17. Eur. Assoc. Comput. Sci. Logic, Springer, September 1993.

[AKW95] Alexander Aiken, Dexter Kozen, and Edward Wimmers. Decidability of systems of set constraints with negative constraints. *Infor. and Comput.*, 1995. To appear. Also Cornell University Tech. Report 93-1362, June, 1993.

[AM91a] A. Aiken and B. Murphy. Implementing regular tree expressions. In *Proc. 1991 Conf. Functional Programming Languages and Computer Architecture*, pages 427–447, August 1991.

[AM91b] A. Aiken and B. Murphy. Static type inference in a dynamically typed language. In *Proc. 18th Symp. Principles of Programming Languages*, pages 279–290. ACM, January 1991.

[AW92] A. Aiken and E. Wimmers. Solving systems of set constraints. In *Proc. 7th Symp. Logic in Computer Science*, pages 329–340. IEEE, June 1992.

[BGW93] L. Bachmair, H. Ganzinger, and U. Waldmann. Set constraints are the monadic class. In *Proc. 8th Symp. Logic in Computer Science*, pages 75–83. IEEE, June 1993.

[Col82] A. Colmerauer. PROLOG and infinite trees. In S.-A. Tärnlund and K. L. Clark, editors, *Logic Programming*, pages 231–251. Academic Press, January 1982.

[Cou83] Bruno Courcelle. Fundamental properties of infinite trees. *Theor. Comput. Sci.*, 25:95–169, 1983.

[CP94a] W. Charatonik and L. Pacholski. Negative set constraints with equality. In *Proc. 9th Symp. Logic in Computer Science*, pages 128–136. IEEE, July 1994.

[CP94b] W. Charatonik and L. Pacholski. Set constraints with projections are in *NEXPTIME*. In *Proc. 35th Symp. Foundations of Computer Science*, pages 642–653. IEEE, November 1994.

[Gri87] Timothy G. Griffin. An environment for formal systems. Technical Report TR87-846, Cornell University, June 1987.

[GTT93a] R. Gilleron, S. Tison, and M. Tommasi. Solving systems of set constraints using tree automata. In *Proc. Symp. Theor. Aspects of Comput. Sci.*, volume 665, pages 505–514. Springer-Verlag Lect. Notes in Comput. Sci., February 1993.

[GTT93b] R. Gilleron, S. Tison, and M. Tommasi. Solving systems of set constraints with negated subset relationships. In *Proc. 34th Symp. Foundations of Comput. Sci.*, pages 372–380. IEEE, November 1993.

[Hei93] Nevin Heintze. *Set Based Program Analysis*. PhD thesis, Carnegie Mellon University, 1993.

[HJ90a] N. Heintze and J. Jaffar. A decision procedure for a class of set constraints. In *Proc. 5th Symp. Logic in Computer Science*, pages 42–51. IEEE, June 1990.

[HJ90b] N. Heintze and J. Jaffar. A finite presentation theorem for approximating logic programs. In *Proc. 17th Symp. Principles of Programming Languages*, pages 197–209. ACM, January 1990.

[JM79] N. D. Jones and S. S. Muchnick. Flow analysis and optimization of LISP-like structures. In *Proc. 6th Symp. Principles of Programming Languages*, pages 244–256. ACM, January 1979.

[JT51] B. Jónsson and A. Tarski. Boolean algebras with operators. *Amer. J. Math.*, 73:891–939, 1951.

[JT52] B. Jónsson and A. Tarski. Boolean algebras with operators. *Amer. J. Math.*, 74:127–162, 1952.

[Koz93] Dexter Kozen. Logical aspects of set constraints. In E. Börger, Y. Gurevich, and K. Meinke, editors, *Proc. 1993 Conf. Computer Science Logic (CSL'93)*, volume 832 of *Lect. Notes in Comput. Sci.*, pages 175–188. Eur. Assoc. Comput. Sci. Logic, Springer, September 1993.

[Koz94] Dexter Kozen. Set constraints and logic programming (abstract). In J.-P. Jouannaud, editor, *Proc. First Conf. Constraints in Computational Logics (CCL'94)*, volume 845 of *Lect. Notes in Comput. Sci.*, pages 302–303. ESPRIT, Springer, September 1994.

[Koz95] Dexter Kozen. Rational spaces and set constraints. In Peter D. Mosses, Mogens Nielsen, and Michael I. Schwartzbach, editors, *Proc. Sixth Int. Joint Conf. Theory and Practice of Software Develop. (TAPSOFT'95)*, volume 915 of *Lect. Notes in Comput. Sci.*, pages 42–61. Springer, May 1995.

[KPS92] Dexter Kozen, Jens Palsberg, and Michael I. Schwartzbach. Efficient inference of partial types. In *Proc. 33rd Symp. Found. Comput. Sci.*, pages 363–371. IEEE, October 1992.

[KPS93] Dexter Kozen, Jens Palsberg, and Michael I. Schwartzbach. Efficient recursive subtyping. In *Proc. 20th Symp. Princip. Programming Lang.*, pages 419–428. ACM, January 1993.

[KPS94] Dexter Kozen, Jens Palsberg, and Michael I. Schwartzbach. Efficient inference of partial types. *J. Comput. Syst. Sci.*, 49(2):306–324, October 1994.

[Mis84] P. Mishra. Towards a theory of types in PROLOG. In *Proc. 1st Symp. Logic Programming*, pages 289–298. IEEE, 1984.

[MR85] P. Mishra and U. Reddy. Declaration-free type checking. In *Proc. 12th Symp. Principles of Programming Languages*, pages 7–21. ACM, 1985.

[Rey69] J. C. Reynolds. Automatic computation of data set definitions. In *Information Processing 68*, pages 456–461. North-Holland, 1969.

[Ste94] K. Stefánsson. Systems of set constraints with negative constraints are NEXPTIME-complete. In *Proc. 9th Symp. Logic in Computer Science*, pages 137–141. IEEE, June 1994.

[YO88] J. Young and P. O'Keefe. Experience with a type evaluator. In D. Bjørner, A. P. Ershov, and N. D. Jones, editors, *Partial Evaluation and Mixed Computation*, pages 573–581. North-Holland, 1988.

This article was processed using the LaTeX macro package with LLNCS style