FIRST ORDER PREDICATE LOGIC
WITHOUT NEGATION IS NP-COMPLETE*

Dexter Kozen

TR 77-307

April 1977

Department of Computer Science
Cornell University
Ithaca, New York 14853

*This research has been supported in part by National Science Foundation Grant DCR75-09433.
FIRST ORDER PREDICATE LOGIC

WITHOUT NEGATION IS

NP-COMPLETE*

Dexter Kozen
Department of Computer Science
Cornell University
Ithaca, NY 14853

Abstract

Techniques developed in the study of the complexity of finitely presented algebras are used to show that the problem of deciding validity of positive sentences in the language of first order predicate logic with equality is $\leq_{\text{log}}$-complete for NP.

*This research has been supported in part by National Science Foundation Grant DCR75-09433.
0. Introduction

In this paper we use techniques developed in [1,2] to prove a complexity result for first order predicate logic with equality, namely that deciding validity of positive sentences (those without occurrences of $\neg$) is $\lesssim \log$-complete for $NP$. This result again attests to the power of negation, as did [3,4] previously, since the general validity problem, even without equality, is undecidable (this result is originally due to Church; see [5] for a very elegant proof, due to Floyd).

It is a little surprising that the problem would be complete for some level of the polynomial time hierarchy rather than some "even" class like P or PSPACE, since universal as well as existential quantification is allowed. This is because universal quantifiers are easy to eliminate, and existential ones not so easy, as we will see.

In [2], we approached a similar problem, that of deciding truth of a sentence of the form

$$Q_1x_1\ldots Q_kx_k \ s(\overline{x}) = t(\overline{x})$$

interpreted over a finitely presented algebra, and showed that it was complete for PSPACE. Here we reduce the validity problem to the problem of deciding truth of the sentence under a particular interpretation, a term algebra similar to the algebras of [1,2], so many of the ideas carry over.

1. Preliminaries

The definitions and results of this section are standard; see for example [6,7].

We first describe the language $L$ of first order predicate logic with equality, but without negation. Sentences of this language will be the
positive sentences of ordinary first order logic with equality.

Definition 1.0

The language L consists of the following:

Symbols

(i) a countably infinite set of variables \( x_0, x_1, \ldots; \)

(ii) a countably infinite set of function symbols \( f_0^m, f_1^m, \ldots \) for each finite arity \( m \geq 0 \) (nullary function symbols \( f_0^0, f_1^0, \ldots \) will be called constants and denoted \( a_0, a_1, \ldots \));

(iii) a countably infinite set of relational symbols \( R_0^m, R_1^m, \ldots \) for each finite arity \( m \geq 0 \);

(iv) an equality symbol \( \equiv \);

(v) logical symbols \( \land, \lor, \exists, \forall. \)

Terms \( t_1, t_2, \ldots \) are defined inductively:

(i) \( x_j, a_j \) are terms;

(ii) if \( t_1, \ldots, t_m \) are terms then \( f_j^m t_1, \ldots, t_m \) is.

Formulas \( \phi, \psi \) are defined inductively:

(i) \( t_1 \equiv t_2, \ R_j^m t_1, \ldots, t_m \) are atomic formulas;

(ii) if \( \phi, \psi \) are formulas then

\( \phi \land \psi, \ \phi \lor \psi, \ \exists x \phi, \ \forall x \phi \) are.

Sentences are closed formulas, i.e. those with no free occurrences of variables.

Definition 1.1

\( t = \) (closed terms (those not containing occurrences of variables))

\( \text{Sym}(t) = \) (symbols appearing in term \( t \))

\( \text{Sym}(\phi) = \) (symbols appearing in formula \( \phi \))

\( \text{Free}(\phi) = \) (variables with free (unquantified) occurrences in \( \phi \)).
We will write \( \phi(x_1, \ldots, x_k) \) to indicate that all the free variables of \( \phi \) are among \( x_1, \ldots, x_k \), and \( \phi(t_1, \ldots, t_k) \) to represent the formula \( \phi \) with all free occurrences of \( x_i \) replaced by \( t_i \), 1 \( \leq \) \( i \) \( \leq \) \( k \).

**Definition 1.2**

A structure for \( L \) is a pair

\[ A = \langle A, \mathcal{I} \rangle \]

where \( A \) is a set (the domain), and \( \mathcal{I} \) is a map (the interpretation) taking function symbols \( f_i^m \) to functions \( A^m \to A \) of the corresponding arity (constants go to elements of \( A \)) and relation symbols \( R_i^m \) to \( m \)-ary relations on \( A \).

We write \( f_i^m \) for \( \mathcal{I}(f_i^m) \) and \( R_i^m \) for \( \mathcal{I}(R_i^m) \).

The interpretation extends naturally to the set of closed terms, by taking

\[ (f_i^m(t_1, \ldots, t_m)_A = f_i^m(t_1_A, \ldots, t_m_A). \]

**Definition 1.3**

A valuation of variables over \( A = \langle A, \mathcal{I} \rangle \) is a map \( v: \{ \text{variables} \} \to A \).

Let \( t \in \mathcal{O} \), \( y \in A \). The map \( v[\{y\}] \) is defined by

\[ v[\{y\}](x_j) = v(x_j) \text{ if } j \neq 1, \]

\[ v[\{y\}](x_1) = y. \]

\( v \) extends naturally to the set of all terms, by taking

\[ v(a_i) = a_i_A \]

\[ v(f_i^m(t_1, \ldots, t_m)) = f_i^m(v(t_1), \ldots, v(t_m)). \]

If \( t \) is any term, we denote \( v(t) \) by \( t_A^v \). Note that for \( t \in \mathcal{O} \), \( t_A^v = t_A \).

**Definition 1.4**

A formula \( \phi \) is true in \( A \) under valuation \( v \) (notation: \( A^v \models \phi \)) if either:

1. \( \phi \) is of the form \( s = t \), \( s \), \( t \) terms, and \( s_A^v = t_A^v \).
(ii) $\phi$ is of the form $R_i^{m}t_1\ldots t_m$ and

$$R_i^m(t_1^A\ldots t_m^A)_v.$$ 

(iii) $\phi$ is of the form $\phi \land \chi$ and

$A^\phi_v$ and $A^\chi_v$;

(iv) $\phi$ is of the form $\phi \lor \chi$ and either

$A^\phi_v$ or $A^\chi_v$;

(v) $\phi$ is of the form $3x_i\phi$ and for some $y\in A,$

$A^\phi_v[(\forall y)^x]$;

(vi) $\phi$ is of the form $\forall x_i\phi$ and for all $y\in A,$

$A^\phi_v[(\forall y)^x].$

Theorem 1.5

Let $A = <A,I>, A' = <A,I'>$ be structures and $v,v'$ be valuations such that $I$ and $I'$ agree on $\text{Sym}(\phi)$ and $v$ and $v'$ agree on $\text{Free}(\phi)$. Then

$$A^\phi_v \iff A'^\phi_{v'}. $$

Proof

Induction on structure of $\phi$.

Corollary 1.6

Let $\phi$ be closed, $v,v'$ any two valuations. Then $A^\phi_v \iff A'^\phi_{v'}$. 

For this reason we may write $A^\phi$ unambiguously whenever $\phi$ is closed, and say $\phi$ is true in $A$.

Definition 1.7

A sentence $\phi$ is valid if $\phi$ is true in all structures.

The validity problem is the set
\( \phi \to \psi \) is a valid sentence of \( \mathcal{L} \).

**Theorem 1.8**

Let \( t \) be any term, and suppose \( v(x_i) = t_{A,v} \). Then \( A^f_v \phi(x_i) \) iff \( A^f_v \psi(t) \), provided no free variables of \( t \) become bound as a result of the substitution.

**Proof**

Induction on structure of \( \phi \).

**Definition 1.9**

Let \( A \) and \( B \) be structures with domains \( A \) and \( B \), respectively. A map \( h: A \to B \) is a homomorphism \( A \to B \) provided for any \( f_i^m \), \( R_i^m \), and \( y_1, \ldots, y_m \in A \),

\[
\begin{align*}
(1) & \quad h(f_i^m(y_1, \ldots, y_m)) = f_i^m(h(y_1), \ldots, h(y_m)), \\
(2) & \quad R_i^m(y_1, \ldots, y_m) = R_i^m(h(y_1), \ldots, h(y_m)).
\end{align*}
\]

If \( h: A \to B \) is a homomorphism and \( v \) is a valuation over \( A \), then \( h \cdot v \) is a valuation over \( B \), and for any term \( t \), \( h(t_{A,v}) = t_{B,h \cdot v} \).

**Theorem 1.10**

Let \( B \) be a homomorphic image of \( A \), let \( \phi \) be any formula, and let \( v \) be any valuation over \( A \). Then

\[ A^f_v \phi \iff B^f_{h \cdot v} \phi. \]

**Proof**

By assumption there is a surjective homomorphism \( h: A \to B \). Proceeding by induction on the structure of \( \phi \),

\[
\begin{align*}
A^f_v R_i^m t_1 \cdots t_m & \iff R_i^m(t_{A,v}^1, \ldots, t_{A,v}^m) \\
& \iff R_i^m(h(t_{A,v}^1), \ldots, h(t_{A,v}^m)).
\end{align*}
\]
\[ R^m_B(t_1, \ldots, t_n, h \circ v) \]
\[ \vdash_{h \circ v} t \]

and
\[ A^v s^A \Rightarrow s_{A, v} = t_{A, v} \]
\[ h(s_{A, v}) = h(t_{A, v}) \]
\[ s_{B, h \circ v} = t_{B, h \circ v} \]
\[ \vdash_{h \circ v} s \]

The induction step for \( \phi \) of the form \( \psi x \) or \( \psi x \) is trivial. Finally,
\[ A^v \forall x \phi \Rightarrow \forall y \forall A \ A^v[A \backslash y] \phi \]
\[ \vdash_{h \circ v} \forall y \forall A \ A^v[h \circ v[A \backslash y]] \phi \]
\[ \vdash_{h \circ v} \forall y \forall A \ A^v[h \circ v[A \backslash h y]] \phi \]

and since \( h \) is onto,
\[ \vdash_{h \circ v} \forall y \forall A \ A^v[h \circ v[A \backslash y]] \phi \]
\[ \vdash_{h \circ v} \forall x \forall y \phi \]

The case of \( \phi = \exists x \phi \) is similar.

**Corollary 1.11**

If \( B \) is a homomorphic image of \( A \) and \( \phi \) is closed, then
\[ A^v \phi \Rightarrow B^v \phi. \]

**Definition 1.12**

The **Herbrand** (or **free**) **structure** is the structure \( T \) with domain \( \tau \), the set of closed terms, and interpretation defined by
\[ a_{1T} = a_{i} \]
\[ f_{1\ldots m}^{T} = \lambda t_{1}\ldots t_{m}[f_{1\ldots m}^{T}t_{1}\ldots t_{m}], m \geq 1 \]
\[ R_{1\ldots m}^{T} = \lambda t_{1}\ldots t_{m}[\text{false}]. \]

Note that for any \( t \in T \), \( t_{T} = t \).

### 2. Main Results

We wish to give a nondeterministic polynomial time algorithm for deciding validity of sentences in \( L \). Our plan will be to reduce the problem of validity of \( \phi \) to truth of \( \phi \) in the Herbrand structure, then use the techniques of [2] to decide truth of \( \phi \) in this structure in nondeterministic polynomial time.

Let \( \phi \) be any sentence of \( L \).

**Theorem 2.0**

\( \phi \) is valid iff \( \vdash_{T} \phi \).

**Proof**

\((\rightarrow)\) By definition of validity.

\((\leftarrow)\) Suppose \( \phi \) is not valid. Then there is a model of \( \neg \phi \). By the Lowenheim-Skolem theorem, there is a countable or finite model of \( \neg \phi \), say \( U \). Let \( U \) be the domain of \( U \), and let \( h: T \rightarrow U \) be any map such that

\[ h(a_{i}) = a_{iU} \quad \text{for } a_{i} \in \text{Sym}(\phi) \]

and \( h \) maps constants not in \( \text{Sym}(\phi) \) onto \( U \). This is possible since \( \text{Sym}(\phi) \) is finite and \( U \) is at most countable. \( h \) then extends uniquely to domain \( \tau \) by taking

\[ h(f_{1\ldots m}^{T}t_{1}\ldots t_{m}) = f_{1\ldots m}^{U}(h(t_{1}),\ldots,h(t_{m})). \]
Thus if we define a new structure $U'$ with domain $U$ and interpretation defined by

$$a_{i_{U'}} = h(a_i)$$

$$f_{i_{U'}}^m = f_i^m, \ m \geq 1,$$

$$R_{i_{U'}}^m = R_i^m, \ m \geq 0,$$

then $h_{T-U'}$ is a surjective homomorphism. But since the interpretations of $U$ and $U'$ agree on $\text{Sym}(\phi)$ and $\mathcal{U}_{\phi}$, by Theorem 1.5, $U\models \phi$. Since $U'$ is a homomorphic image of $T$, by Corollary 1.11, $T\models \phi$.

We can also restrict our attention to sentences of a special form.

**Lemma 2.1**

There is a polynomial time algorithm which, given formula $\phi$, produces $\phi'$ such that

(i) $\phi'$ is in prenex form,

(ii) all atomic formulas of $\phi'$ are of the form $s = t$ (i.e. $\phi'$ contains no relational symbols), and

(iii) for any $v$, $T\models \phi$ iff $T\models \phi'$.

**Proof**

The standard algorithm for converting any sentence to an equivalent one in prenex form, which can be found in any logic text (e.g. [6]) is polynomial in time and will suffice for our purposes. To dispose of the relational symbols, since every $R_i^m$ is interpreted as universally false in $T$, atomic formulas of the form $R_i^m t_1 \ldots t_m$ occurring in $\phi$ may be replaced by the formula $a_0 = a_1$, which is also false in $T$. Then (iii) may be verified by induction on the structure of $\phi$. $\blacksquare$
Henceforth all sentences of $L$ we consider will be assumed to be in this form.

We have reduced the validity problem to the problem of truth in $T$ of sentences of a special form. One useful consequence of this, which we will exploit fully, is that the subtle distinction between mention and use can now be conveniently ignored, since the semantic individuals (closed terms) are actually syntactic objects as well. More precisely,

**Theorem 2.2**

(i) $\Phi \sqsubseteq s \iff s \sqsubseteq t$, $s, t \in T$;

(ii) $\Phi \sqsubseteq \forall x_1 \phi(x_1) \iff$ for all $t \in T$ $\Phi \sqsubseteq \phi(t)$;

(iii) $\Phi \sqsubseteq \exists x_1 \phi(x_1) \iff$ there is a $t \in T$ $\Phi \sqsubseteq \phi(t)$.

**Proof**

(i) is a direct consequence of the fact that $s_T = s$ and $t_T = t$;

(ii) and (iii) follow from the definition of $\sqsubseteq$ and Theorem 1.8. □

Thus we may write

$$Q_1x_1\ldots Q_kx_k \phi(x_1,\ldots,x_k) \quad (*)$$

for

$$\Phi \sqsubseteq Q_1x_1\ldots Q_kx_k \phi(x_1,\ldots,x_k) \quad (**)$$

Here (**) is an assertion about truth of a sentence of $L$ in $T$, whereas (*) is a metastatement about elements of $T$. In (*), all $= $ have been changed to $\in$, variables range over $T$, and the $Q_i$ are no longer symbols of $L$, but represent the English "for all" and "there is" in (ii) and (iii) of the previous theorem. Henceforth we shall in general not distinguish between (*) and the right side of the $\sqsubseteq$ in (**).

Now we show how to get rid of leading universal quantifiers.
Theorem 2.3

\[ \vdash \forall x_j \phi(x_j) \iff \vdash \phi(a_j), \]

where \( a_j \models \text{Sym}(\phi) \).

Proof

Let \( v \) be any valuation with \( v(x_i) = a_j \).

\( (+) \quad \vdash \forall x_i \phi(x_i) \rightarrow \vdash v_\phi(x_i) \)

\[ \quad + \vdash \phi(a_j), \]

by Theorem 1.8.

\( (+) \quad \text{Let } \vdash \phi(a_j). \text{ For arbitrary } y, \text{ define} \)

\[ h(a_i) = a_i \text{ for } a_i \models \text{Sym}(\phi) \]

\[ h(a_j) = y \]

and let \( h \) map \( (a_i | a_i \models \text{Sym}(\phi) \) and \( i \neq j) \) onto \( y \). Extend \( h \) to a homomorphism \( T \rightarrow T' \), where \( T' \) is just \( T \) with some of the \( a_i \)'s not appearing in \( \phi \) reinterpreted, as in the proof of Theorem 2.0.

Since \( h \) is surjective, by Corollary 1.11,

\[ T \vdash \phi(a_j), \]

thus

\[ T \vdash v_{[J \setminus y]} \phi(x_j), \]

by Theorem 1.8. By Theorem 1.5,

\[ T \vdash v_{[J \setminus y]} \phi(x_j). \]
As \( y \) was arbitrary,

\[ T \forall x_j \phi_j(x_j). \]

The above theorem indicates why universal quantifiers are so easy to eliminate in this setting: there are an infinite number of unused constant symbols which are ripe for reinterpretation. In [2] this was not possible, since the number of symbols was finite. The problem studied in [2], namely the truth of sentences of the form

\[ Q_1x_1 \ldots Q_kx_k \text{s}^t(\vec{x}) \]

in a finitely presented algebra, appears to correspond to the validity problem for sentences in \( L \) when a certain kind of bounded quantification is allowed, but the exact correspondence is unclear (see §3).

Let us further restrict our attention to formulas with conjunctive matrices. Let \( \phi \) be in prenex form with no relational symbols besides \( \to \); i.e., \( \phi \) looks like

\[ Q_1x_1 \ldots Q_kx_k \text{B}(\phi_1(\vec{x}), \ldots, \phi_n(\vec{x})) \]

where \( B \) is a monotone Boolean tree with leaves \( \phi_1(\vec{x}), \ldots, \phi_n(\vec{x}) \), each \( \phi_i \) an atomic formula \( s_1 \text{ct}_1 \), and \( \vec{x} = < x_1, \ldots, x_k >. \)

**Lemma 2.4**

\[ Q_1x_1 \ldots Q_kx_k \text{B}(\phi_1(\vec{x}), \ldots, \phi_n(\vec{x})) \]

iff there is a subset of the \( \phi_i \)'s, MLOG say \( \phi_1 \ldots \phi_m \), such that

(1) \( B(\text{true, \ldots, true, false, \ldots, false}) = \text{true} \), and

(2) \( \text{false} \)}
\[
(i) \ \exists_{i=1}^{m} x_{i} \ \forall_{k=1}^{n} x_{k} \ \phi_{i}(\bar{x}).
\]

**Proof**

Induction on the number of quantifiers. The basis is easy. The induction step has two cases:

**Case 1** leading existential quantifier.

\[
3_{x_1} Q_{x_2} \ldots Q_{x_k} B(\phi_1(\bar{x}), \ldots, \phi_n(\bar{x}))
\]

iff

for some \( x_1 \epsilon \tau, Q_{x_2} \ldots Q_{x_k} B(\phi_1(\bar{x}), \ldots, \phi_n(\bar{x})) \)

iff \quad (by induction hypothesis)

for some \( x_1 \epsilon \tau \) and some subset \( \phi_1, \ldots, \phi_m \) of the \( \phi_i \)'s,

(1) \( B(\text{true}, \ldots, \text{true}, \text{false}, \ldots, \text{false}) \), and

(11) \( Q_x \exists_{x_2} \ldots Q_{x_k} \ \bigwedge_{i=1}^{m} \phi_i(\bar{x}) \)

iff

for some subset \( \phi_1, \ldots, \phi_m \) of the \( \phi_i \)'s,

(1) \( B(\text{true}, \ldots, \text{true}, \text{false}, \ldots, \text{false}) \), and

(11) \( 3_{x_1} Q_{x_2} \ldots Q_{x_k} \ \bigwedge_{i=1}^{m} \phi_1(\bar{x}) \).

**Case 2** leading universal quantifier.

\( \forall_{x_1} Q_{x_2} \ldots Q_{x_k} B(\phi_1(\bar{x}), \ldots, \phi_n(\bar{x})) \)

iff \quad (by Theorem 2.3)
where $a_j \in \text{Sym}(\phi)$.

iff (by induction hypothesis)

for some subset $\phi_1, \ldots, \phi_m$ of the $\phi_i$'s,

(i) $B(\underbrace{\text{true}, \ldots, \text{true}}_{m}, \underbrace{\text{false}, \ldots, \text{false}}_{n-m})$, and

(ii) $\bigwedge_{i=1}^{m} Q_{2x_2^k \ldots Q_{kx_k^k} \quad i=1 \phi_i(a_j, x_2^k, \ldots, x_m)$

iff

for some subset $\phi_1, \ldots, \phi_m$ of the $\phi_i$'s,

(i) $B(\underbrace{\text{true}, \ldots, \text{true}}_{m}, \underbrace{\text{false}, \ldots, \text{false}}_{n-m})$

(ii) $\forall \bar{x} \bigwedge_{i=1}^{m} Q_{2x_2^k \ldots Q_{kx_k^k} \quad i=1 \phi_i(\bar{x})$.

Lemma 2.4 is not as trivial as it first may appear; some of the variables are universally quantified, and different valuations of these variables could cause different atomic formulas of the matrix to be true. The object of the lemma is to uniformize the set of atomic formulas which can be true, so that our nondeterministic polynomial time algorithm can initially guess this set of atomic formulas, verify that $B$ is true with those formulas true, and then verify the conjunctive formula

$\bigwedge_{i=1}^{m} Q_{1x_1^k \ldots Q_{kx_k^k} \quad i=1 \phi_i(\bar{x})$.

The following definitions and lemmas are simplified versions of ones
appearing in [2], which the reader may consult for a more thorough treatment.

Definition 2.5

Let $\alpha \in \{f_1, m_k | 1 \leq k \leq m\}^*$ be a string of symbols on a path through the tree representation of a term. E.g. if $s, t$ are terms,

$$s = f_1^3f_1^2a_1a_2f_1^2a_3a_4f_1^1a_1x_5$$

$$t = f_1^1x_5,$$

then their tree representation are

```
  f_1^3
  / \  \
 1   2  3
 /     \   \
 f_1^2  f_1^2  f_1^1
 / \   / \   / \\
1   2  1  2  1   1
 / \   / \   / \\
 a_1 a_2 a_3 a_4 f_1^1
    /       /   \
   /  x_5   x_5
```

```
and the path from the root of \( s \) to the root of \( t \) is

\[
a = f_1^3 f_2^1.
\]

We write \( s \triangleleft t \) to indicate that term \( t \) appears as a subterm of term \( s \) at the position specified by \( a \).

The empty string is denoted \( \lambda \); thus \( s \triangleleft t \) iff \( s = t \).

As in [2], we will allow terms to be represented by dags instead of trees, by "factoring out" common subterms; e.g.

The reason for this representation, as opposed to a tree representation, is that sometimes we will want to replace all occurrences of some variable with some term; the dag representation allows us to do this by readjusting edges, so that the representation does not grow any bigger.

Let the sentence
be so represented. Extra undirected edges between terms may be used to represent $\sim$. E.g., the sentence

$$\forall x_0 \exists x_1 f_1 \ x_0 \sim f_1 \ x_1 \land f_1 \ x_0 \ x_1 \sim a_3 \land x_0 \sim x_1$$

could be represented by

In the following, let

$$\phi = Q_1 x_1 \ldots Q_k x_k \ s_j \sim t_i$$

be given.

**Definition 2.6**

$\sim$ is the smallest equivalence relation on terms satisfying

(i) $s_i \sim t_i$, $1 \leq i \leq n$

(ii) if $r^m u_1 \ldots u_m \sim r^m v_1 \ldots v_m$ then

$$u_i \sim v_i, 1 \leq i \leq m.$$  

**Lemma 2.7**

If $\bar{y} \in \mathcal{X}^k$ is such that $\bigwedge_{i=1}^n s_i(\bar{y}) = t_i(\bar{y})$, and if $u \sim v$, then $u(\bar{y}) = v(\bar{y})$.

**Proof**

Induction on definition of $\sim$.  

\[\]
Definition 2.8

Let \( x_i, x_j \) be variables. Define \( x_i \preceq x_j \) if either

1. \( \exists u \ x_i \sim u \land u \preceq x_j \), or
2. \( \exists x_k, \alpha, \gamma \ a = \gamma, x_i \preceq x_k, \land x_k \preceq x_j \).

A variable \( x_i \) is principal if \( x_i \preceq x_j \) implies \( a = \lambda \).

Lemma 2.9

If \( \bar{y} = <y_1, \ldots, y_k> \in \varepsilon^k \) such that \( \bigwedge_{i=1}^n s_i(\bar{y}) = t_i(\bar{y}) \), and if \( x_i \preceq x_j \), then \( y_i \preceq y_j \).

Proof

Induction on definition of \( x_i \preceq x_j \).

Definition 2.10

\( R^+ = (\text{subterms of } s_1 \text{ and } t_1, 1 \leq i \leq n) \).

\( R = R^+ \tau \).

Lemma 2.11

If \( x_i \) is not principal then there is a proper term \( w \in R^+ \) (a proper term is one that is not a variable or a constant) such that \( x_i \sim u \). Moreover, there is a polynomial time algorithm to determine whether \( x_i \) is principal, and if not, supply a proper \( w \in R^+ \) such that \( x_i \sim u \).

Proof

The first part follows from the definition of \( x_i \preceq x_j \). For the second part, construct the relation \( \sim \) inductively on the dag representation of \( \bigwedge_{i=1}^n s_i \sim t_i \).

Lemma 2.12

If \( x_i \) is principal, \( y_i \in R \), and \( z_i \in R \), then
\[
\begin{align*}
q_1x_1 \cdots q_nx_n & \quad \prod_{i=1}^n s_i(y_1, x_2, \ldots, x_k) = t_i(y_1, x_2, \ldots, x_k) \\
\text{iff} \\
q_1x_1 \cdots q_nx_n & \quad \prod_{i=1}^n s_i(z_1, x_2, \ldots, x_k) = t_i(z_1, x_2, \ldots, x_k).
\end{align*}
\]

Proof

Suppose \(x_1\) is principal, \(\tilde{y} \in \mathbb{R}^k\) with \(y_1 \in \mathbb{R}\), and \(\prod_{i=1}^n s_i(\tilde{y}) = t_i(\tilde{y})\). Let \(z_1 \in \mathbb{R}\) be arbitrary. For \(1 \leq i \leq k\), define \(g_i(y_1) = y_1\) with all occurrences of \(y_1\) in \(y_1\) at a position \(\alpha\) such that \(x_1x_1\) replaced by \(z_1\). For example, if

\[
\begin{align*}
\tilde{y} = y_1 & \quad y_3 = \\
& \quad f_1^2
\end{align*}
\]

and \(x_3(f_1^2)z_1\) but not \(x_3(f_1^2)z_1\) then

\[
\begin{align*}
g_3(y_3) & = \\
& \quad f_1^2
\end{align*}
\]

Note that \(g_1(y_1) = z_1\). Let \(<g_1(y_1), \ldots, g_k(y_k)>\) be denoted by \(g(\tilde{y})\).

We claim that \(\prod_{i=1}^n s_i(g(\tilde{y})) = t_i(g(\tilde{y}))\). WLOG, it suffices to show that whenever \(s_i(\tilde{y})_\alpha y_1\) and \(s_i(g(\tilde{y}))_\alpha z_1\) then \(t_i(\tilde{y})_\alpha y_1\) and \(t_i(g(\tilde{y}))_\alpha z_1\), since then
all the same occurrences of \( y \) in \( s_1(\tilde{y}) \) and \( t_1(\tilde{y}) \) are replaced by \( z \). Suppose \( s_1(\tilde{y}) \in y_1 \) and \( s_1(g(\tilde{y})) \in z \). We know \( t_1(\tilde{y}) = y_1 \), since \( s_1(\tilde{y}) = t_1(\tilde{y}) \). It must be that \( \alpha = \beta y \), \( s_1 \alpha x_j \), \( x_j y x_1 \), and \( y_1 y y_1 \), for some \( \beta, \gamma, x_j \). If \( t_1 \), \( w \) is a proper term, and \( w \in x_k \) for some \( x_k \), use the definition of \( \alpha \) to show that \( x_1 \notin x_k \), contradicting the principality of \( x_1 \). If \( t_1 \), \( w \in R \), then \( y_1 \notin R \), contradicting an assumption. The only possibility remaining is that \( t_1 \notin x_k \) and \( \delta \in \alpha \), for some \( \delta, \alpha, x_k \). A case argument of two cases (one in which \( \delta \) is a substring of \( \beta \), the other in which \( \beta \) is a substring of \( \delta \)) shows that \( x_k \notin x_1 \), thus \( y_1 \notin y_1 \) by Lemma 2.9. Then \( g_1(y_k) \in x_1 \) and \( t_1(g(\tilde{y})) \in z_1 \), and the claim is verified.

Proceeding by induction on quantifiers, suppose for any \( y_2, \ldots, y_k \),

\[
Q_{k+1} y_{k+1} \ldots Q_k y_k \phi(y_1, \ldots, y_k, y_{k+1}, \ldots, y_k)
\]

where \( \phi = \bigwedge_{i=1}^{\infty} s_i \in t_i \). Then

\[
3y_k Q_{k+1} y_{k+1} \ldots Q_k y_k \phi(y_1, \ldots, y_k, y_{k+1}, \ldots, y_k) \quad (\ast)
\]

and

\[
3y_k Q_{k+1} y_{k+1} \ldots Q_k y_k \phi(g_1(y_1), \ldots, g_k(y_k), y_{k+1}, \ldots, y_k) \quad (\ast\ast)
\]

the \( y_k \) satisfying \((\ast\ast)\) is obtained by applying \( g_k \) to the \( y_k \) satisfying \((\ast)\). If

\[
V y_k Q_{k+1} y_{k+1} \ldots Q_k y_k \phi(y_1, \ldots, y_{k-1}, y_k, \ldots, y_k)
\]

then it cannot be the case that \( x_k \notin x_1 \) for any \( \alpha \), by Lemma 2.9. Thus
\[ g_k(y_k) = y_k \text{ for all } y_k < \bot. \] Then
\[
\forall y_k \forall y_{k+1} \forall y_{k+2} \ldots \forall y_k g_k(y_1, y_2, y_{k+1}, \ldots, y_k)
\]
\[
\rightarrow
\forall y_k \forall y_{k+1} \forall y_{k+2} \ldots \forall y_k g_k(y_1), g_k(y_2), g_k(y_{k+1}), \ldots, y_k.
\]
\[
\forall y_k \forall y_{k+1} \forall y_{k+2} \ldots \forall y_k g_k(y_1), g_k(y_2), g_k(y_{k+1}), \ldots, y_k.
\]

We have shown
\[
\bigwedge_{i=1}^{n} Q_1 x_1 \cdots Q_k x_k s_i(y_1, x_2, \ldots, x_k) = t_i(y_1, x_2, \ldots, x_k)
\]
\[
\bigwedge_{i=1}^{n} Q_1 x_1 \cdots Q_k x_k s_i(z_1, x_2, \ldots, x_k) = t_i(z_1, x_2, \ldots, x_k).
\]

and the converse follows from symmetry.

Now we are ready to show how to eliminate leading existential quantifiers.

**Theorem 2.13**

There is a nondeterministic polynomial time algorithm which, given
\[
\bigwedge_{i=1}^{n} Q_1 x_1 \cdots Q_k x_k s_i(\bar{x}) = t_i(\bar{x}), \quad (*)
\]
produces a true formula of the same size as (*) but with one fewer quantifier
iff (*) is true.

**Proof**

Given (*), if \( x_1 \) is principal, guess whether some \( y_1 \in \mathcal{R} \) will satisfy (*) when substituted for \( x_1 \). If guessed yes, guess which one, replace all occurrences of \( x_1 \) in \( s_i \) and \( t_i \) with \( y_1 \) (this is done by redirecting all edges into occurrences of \( x_1 \) to the root of \( y_1 \), thus the size of the representation
does not increase) and output

$$Q_2 x_2 \ldots Q_k x_k \bigwedge_{i=1}^{n} s_i(y_1, x_2, \ldots, x_k) = t_i(y_1, x_2, \ldots, x_k).$$

If guessed no, output

$$Q_2 x_2 \ldots Q_k x_k \bigwedge_{i=1}^{n} s_i(a_j, x_2, \ldots, x_k) = t_i(a_j, x_2, \ldots, x_k)$$

for some $a_j \in \text{Sym}(\phi)$. This suffices, by Lemma 2.12. If $x_1$ is not principal, Lemma 2.11 guarantees us a proper term $u \in R^+$ such that $x_1 - u$. Thus (*) is equivalent to

$$\bigwedge_{i=1}^{n} 3x_1 Q_2 x_2 \ldots Q_k x_k s_i(\bar{x}) = t_i(\bar{x}) \land x_1 = u(\bar{x}),$$

by Lemma 2.7. If $x_1$ appears in $u$, the sentence is false, and we may immediately reject. Otherwise replace all occurrences of $x_1$ in $s_i$ and $t_i$ with $u$ (redirect edges incident to $x_1$ to the root of $u$) and call the resulting terms $s_i^{-}, t_i^{-}$. Now we have the equivalent formula

$$\bigwedge_{i=1}^{n} 3x_1 Q_2 x_2 \ldots Q_k x_k s_i^{-}(\bar{x}) = t_i^{-}(\bar{x}) \land x_1 = u(\bar{x}). \quad (***)$$

Let $x_j, \ldots, x_r$ be the variables appearing in $u$, and suppose $y_1$ satisfies (**). If any of the $x_j$ are universally quantified, the sentence is immediately false. Otherwise, if $u \equiv x_j \alpha$, the only value of $x_j$ which can satisfy (**) is the subterm of $y_1$ occurring at position $\alpha$. As this is uniform in the universally quantified variables occurring before $3x_j$ in the quantifier list, the $3x_j$ may be moved to the front. Thus (**) implies

$$\bigwedge_{i=1}^{n} 3x_1 3x_j \ldots 3x_r Q_{r+1} x_{r+1} \ldots Q_k x_k s_i(\bar{x}) = t_i(\bar{x}) \land x_1 = u(\bar{x}). \quad (***)$$
where \( Q'_{r+1} x'_{r+1} \ldots Q_k x_k \) is the quantifier list \( Q_2 x_2 \ldots Q_k x_k \) with all the \( 3x_j \) removed. Clearly the implication goes the other way as well, since for any \( \psi(x_1, x_2) \),

\[
3x_1 \forall x_2 \psi(x_1, x_2) \rightarrow \forall x_2 3x_1 \psi(x_1, x_2).
\]

But (***) is equivalent to

\[
3x_1 \ldots 3x_{j_r} [3x_1 x_1 = u(x)] \wedge \bigwedge_{i=1}^n Q_{r+1} x'_{r+1} \ldots Q_k x_k s_i(x) = t_i(x), (\ast)
\]

since \( x_1 \) does not occur in any \( s_i \) or \( t_i \). But (\ast) is trivially equivalent to

\[
3x_1 \ldots 3x_{j_r} Q_{r+1} x'_{r+1} \ldots Q_k x_k s_i(x) = t_i(x),
\]

which is of the desired form. As all manipulations took polynomial time, we are done.

\[\square\]

Theorem 2.14

The validity problem is in \( \mathbf{NP} \).

Proof

Given \( \psi \), we need only check that \( \psi \models \psi \), by Theorem 2.0. We can eliminate relational symbols and convert to prenex form in polynomial time, by Lemma 2.1. By Lemma 2.4, we can convert to a formula with a conjunctive matrix in nondeterministic polynomial time. Theorems 2.3 and 2.13 allow us to eliminate quantifiers in nondeterministic polynomial time. Finally, we are left with a sentence of the form

\[
\bigwedge_{i=1}^n s_i = t_i
\]
where \( s, t \in \tau \), which can certainly be verified in polynomial time. \( \square \)

Theorem 2.15

The validity problem is \( L_{\log} \)-complete for NP.

Proof

We reduce a well-known NP-complete problem, the satisfiability of Boolean formulas, to the validity problem.

Let \( B \) be a Boolean formula with variables \( x_1, \ldots, x_k \). Let \( B^- \) be formed from \( B \) by replacing each literal \( x_i \) by \( x_i = a_1 \) and each literal \( \neg x_i \) by \( x_i = a_0 \). E.g. if

\[
B = (x_1 \lor \neg x_2 \lor x_3) \land (x_1 \lor x_2 \lor \neg x_3)
\]

then

\[
B^- = (x_1 = a_1 \lor x_2 = a_0 \lor x_3 = a_1) \land (x_1 = a_1 \lor x_2 = a_1 \lor x_3 = a_0).
\]

If \( B \) is satisfiable over \((\text{true}, \text{false})\) then \( B^- \) is satisfiable over \((a_0, a_1)\) in the obvious way. If \( B^- \) is satisfiable over \( \tau \) then \( B^- \) is satisfiable over \((a_0, a_1)\), by reassigning any variable in \( B^- \) not assigned to \( a_0 \) or \( a_1 \) to either \( a_0 \) or \( a_1 \). The monotonicity of \( B^- \) guarantees that the new assignment also satisfies \( B^- \). From this we get a satisfying assignment for \( B \) in the obvious way. Thus

\( B \) is satisfiable over \((\text{true}, \text{false})\)

iff

\( B^- \) is satisfiable over \( \tau \)

iff (by Theorem 2.2)

\[ \varphi \equiv \exists x_1 \ldots \exists x_k \ B^- \]
iff \hspace{1cm} \text{(by Theorem 2.0)}

$3x_1...3x_k B$ is a valid sentence of $L$.

Theorem 2.15 is a special case of a more general result:

**Theorem 2.16**

Let $I$ be a finite set of sentences of the form $s=t$, $s,t \in \tau$, and let $\phi$ be a sentence of $L$. The problem,

"Is $\phi$ true in all models of $I$?"

is $\mathcal{L}_{\text{log}}$-complete for NP.

A more extensive use of the techniques of [2] is needed, but all the main ideas are here. The Herbrand domain for the more general case is the quotient structure $T/I$, whose domain is the set of closed terms $\tau$ modulo the congruence relation induced by $I$.

It is conjectured that Theorem 2.16 holds even if $I$ is allowed to contain atomic formulas $R_{i}^{m}t_{1}...t_{m}$, $t_{i} \in \tau$, $1 \leq i \leq m$.

3. Problems

(i) Prove the conjecture at the end of §2. What other generalizations can be made?

(ii) Let $I$ be given. Suppose that in addition to $\forall \exists$ we allow bounded quantifiers of the form

$\forall t_{1},...,t_{n},f_{1}^{m_{1}},...,f_{k}^{m_{k}}$ and $3 t_{1},...,t_{n},f_{1}^{m_{1}},...,f_{k}^{m_{k}}$.

The meaning of $\forall t_{1},...,t_{n},f_{1}^{m_{1}},...,f_{k}^{m_{k}} x$ would be, "for all elements $x$ of the substructure of $A$ generated by $t_{1},...,t_{n}$ \text{ under } f_{1}^{m_{1}},...,f_{k}^{m_{k}}$.

We apparently now have enough power to force variables to range only over
the algebra presented by ε, instead of all of A, thus the validity problem
is at least PSPACE-hard (see [2]). Use the fact that deciding membership
in a finitely generated substructure is complete for P [2] to show that
this is all the power you get; i.e., show that with bounded quantifiers,
the validity problem for sentences with n alternations of quantifiers, the
outermost a 3(\forall), is complete for \Sigma^n_P(nP), and the validity problem in
general is complete for PSPACE.


