Complexity of Linear Standard Theories

Christopher Lynch and Barbara Morawska

Department of Mathematics and Computer Science Box 5815, Clarkson University, Potsdam, NY 13699-5815, USA, E-mail: clynch@clarkson.edu,morawskb@clarkson.edu **

Abstract. We give an algorithm for deciding E-unification problems for linear standard equational theories (linear equations with all shared variables at a depth less than two) and varity 1 goals (linear equations with no shared variables). We show that the algorithm halts in quadratic time for the non-uniform E-unification problem, and linear time if the equational theory is varity 1. The algorithm is still polynomial for the uniform problem. The size of the complete set of unifiers is exponential, but membership in that set can be determined in polynomial time. For any goal (not just varity 1) we give a NEXPTIME algorithm.

1 Introduction

Automated Deduction problems frequently involve the use of equational logic. Usually, it is necessary to solve some kind of equational unification problem[1]. These problems can take three forms. The simplest is the word problem, where one must decide if two given terms are equivalent modulo an equational theory. A more difficult problem is the problem of deciding E-unification, i.e., deciding if there is a substitution that will make two terms equivalent modulo an equational theory. Finally, it is also sometimes necessary to solve an E-unification problem, which means to find a generating set of all the substitutions which make two terms equivalent modulo an equational theory.

All of these problems are undecidable in general. However, it is possible that an E-unification problem might be decidable in the equational theory of interest. Therefore, an important goal is to classify the equational theories and unification problems for which these problems can be solved. If a problem is decidable, it is also desirable to know the complexity of the problem. In particular, it would be especially useful to classify equational theories in a syntactic way, such that the decidability and complexity of these problems are easily known just by examining the equational theory.[1] This paper makes progress in that direction.

For a long time, one such syntactic class has been known: the class of ground equations (no variables). The word problem in this class is decidable in time $O(n \cdot lg(n))$[11]. The problem of deciding E-unification is NP complete.[12]. Shallow

** This work was supported by NSF grant number CCR-9712388 and ONR grant number N00014-01-1-0435. .

[1] When we refer to an equational theory, we mean a finite presentation of the theory.

theories are an extension of ground theories, where no variable in an equation occurs at a depth greater than one. This class was identified and shown decidable in [4], and also studied in [3, 18]. For shallow theories, the word problem is decidable in polynomial time, deciding E-unification is NP-complete, and the number of E-unifiers in a minimal complete set of unifiers is simply exponential. See [18] for a simple proof.

In linear standard theories[8, 18], both sides of each equation are linear, which means that no variable occurs twice on a side, and variables that are shared by both sides of the equation appear at depth 1 or 0 on both sides. Notice that non-shared variables may appear at any depth. The E-unification problem for this class has been shown to be decidable, but no complexity results are known, even for the word problem. The minimal complete set of E-unifiers has been shown to be finite, but a bound is not known. Similar results exist for standard theories[18] and semilinear theories[9].

In this paper we give a new technique for finding decidability and complexity results for E-unification problems. The technique is based on a simple algorithm, given by goal-directed inference rules. We consider linear standard theories. In particular, we examine the E-unification problem for goals of varity 1, which means that no variable occurs more than once in the goal. This problem is simpler than the general E-unification problem, but more difficult than the word problem, so all the complexity results we obtain apply directly to the word problem. We make a distinction between uniform and non-uniform E-unification problems. In the uniform problem, the input contains the goal and the equational theory, while the non-uniform problem is parameterized by the equational theory, and the input just contains the goal. We show that the complexity of the non-uniform E-unification problem is quadratic. Furthermore, if no variable occurs more than once in any equation of the equational theory, then the complexity is linear. Even in the uniform problem the complexity is still polynomial.

We show several other results. We define a set of terms, polynomial in size such that every term in the range of a substitution in the complete set of E-unifiers belongs to that set of terms. Using that and the polynomial complexity result, we get some other results that are independent of our algorithm. We show how to construct a complete set of E-unifiers whose size is at most simply exponential. We also show that it is not possible to do better, because we show an example of a ground theory where the varity 1 E-unification problem has a simply exponential minimal complete set of E-unifiers. Even though the complete set of E-unifiers we construct is exponential, we show that membership in that set can be decided in polynomial time.

Finally, we examine the general E-unification problem for linear standard theories, i.e., now the goal is unrestricted. In this case, we show that E-unification is decidable in NEXPTIME. The size of a minimal complete set is at most doubly exponential, but each term appearing in the range of a substitution in that set has linear depth. It is known that E-unification is NP hard, because of the NP completeness result for ground theories. So there is a gap here to be filled.

We would like to give some flavor of our results. The first thing we do in this paper is to give a goal-directed inference procedure. We prove that the procedure is sound and complete for any linear theory and varity 1 goal. This inference system has interest on its own. It is similar to the inference procedure for Syntactic Theories[10]. However, our inference procedure does not require the theory to be syntactic. The problem of Eager Variable Elimination is an open problem for the inference procedure for Syntactic Theories and other related inference systems[6, 10, 15]. We solve it in our context, given the restriction on theories and goals. The only other procedure known to us where Eager Variable Elimination has been shown to preserve completeness is in [17]. It is an important problem to solve because it adds determinism to the procedure.

After proving the completeness of the inference rules for linear theories, we tailor them for linear standard theories. First we show that when an inference rules is applied to a varity 1 goal, it remains varity 1. In other words, no variables are shared among goal equations, and they can each be considered separately. We also give a polynomial size set of terms and show that every equation generated is made up of these terms. The inference rules may seem arbitrary, but they were designed to allow these two results, which were difficult to obtain for theories containing collapse axioms (see Section 3). Since no variable occurs more than once in a subgoal, Variable Elimination does not applies. Therefore, there are no inference rules that combine goal equations. This means that each inference rule can be written as a Horn Clause, with the premise of the inference rule at the head and its conclusion as the body. Since we know that only polynomially many terms can appear in the inference, we know we only need polynomially many instances of the Horn clause, and our complexity results follow from the fact that Horn Clause implication is decidable in linear time[5]. This process is similar to what is done in stably local theories[16, 2, 7]. Results about the size of the complete set of E-unifiers and the general E-unification problem for linear standard theories follow from these results. All missing proofs, lemmas and definitions can be found in [13].

2 Preliminaries

We use standard definitions as in [1].

Given a unification problem we can either solve the unification problem or decide the unification problem. Given a goal G and a set of equations E, to *solve* the unification problem means to find a complete set of E-unifiers of G. To *decide* the unification problem simply means to answer true or false as to whether G has an E-unifier. In this paper, we consider both of these problems.

We say that a term t (or an equation or a set of equations) has *varity* n if each variable in t appears at most n times. An equation $s \approx t$ is linear if s and t are both of varity 1. Note that the equation $s \approx t$ is then of varity 2, but it might not be of varity 1. A set of equations is *linear* if each equation in the set is linear. For example, the axioms of group theory $(\{f(x, f(y, z)) \approx f(f(x, y), z), f(w, e) \approx w, f(u, i(u)) \approx e.$ are of varity 2.

If G is a set of equations then we define a *path* in G to be a sequence of equations $u_1 \approx v_1, \cdots u_n \approx v_n$ from G, such that for all j, $1 \leq j \leq n$, $u_j \approx v_j \in G$ or $v_j \approx u_j \in G$, and for all i, $1 \leq i < n$, $Vars(v_i) \cap Vars(u_{i+1}) \neq \emptyset$. In addition, we require that if $u \approx v$ is in G but $v \approx u$ is not, then they cannot both appear in the path. We call the path a *cycle* if $Vars(u_1) \cap Vars(v_n) \neq \emptyset$. For example, the sequence $f(x_1, x_2) \approx g(x_3), f(x_3, x_4) \approx g(x_5), f(x_6, x_5) \approx g(x_2)$ is a cycle. Note that a single equation $u \approx v$ forms a cycle if u and v have any variables in common. If G has a cycle, we say that G is cyclic.

3 Inference Rules

We give a set of inference rules for finding a complete set of E-unifiers of a goal G, and later we prove that for a linear equational theory E, every goal G of varity 1 and substitution θ such that $E \models G\theta$ can be converted into a *normal form* which determines a substitution more general than θ. The inference rules decompose an equational proof by choosing a potential step in the proof and leaving what is remaining when that step is removed.

We define *solved equations* recursively. An equation $x \approx t$ in a goal $x \approx t \cup G$ is *solved* if x does not appear in an unsolved equation in $G - \{x \approx t\}$. Then x is called a *solved variable.* We define the *unsolved part of* G to be the set of all equations in G that are not solved.

As in Logic Programming, we have a selection rule. For each goal G, we don't-care nondeterministically select a unsolved equation $u \approx v$ from G. We say that $u \approx v$ is *selected in* G. If all equations in G are solved, then nothing is selected, G is in normal form and a most general E-unifier can be easily determined.

The inference rules are given in Figure 1. Except for Mutate, these are the usual inference rules for syntactic unification. We assume that the equational theory is consistent, i.e., that it has no equations of the form $t \approx x$ with $x \notin t$. Therefore, in the Mutate-2 rule, $f(s_1, \cdots, s_p)$ must contain x. In that case, we call $f(s_1, \cdots, s_p) \approx x$ a *collapse axiom.* So, Mutate-2 and Mutate-3 are only applicable in theories containing collapse axioms.

The Mutate-1 rule is so-called because it is similar to the inference rule Mutate that is used in the inference procedure for Syntactic Theories[10]. The rule assumes that there is an equational proof of the goal equation with at least one step at the root. If one of the equations in this proof is $s \approx t$ then that breaks up the proof at the root into two separate parts. We see from the inference rules that this rule is applicable if the last step at the root is not a collapse axiom with a variable on the right hand side. Otherwise, Mutate-2 or Mutate-3 will apply. Mutate-2 is applicable if there is a step at the root that is not a collapse axiom with a variable on the right hand side. Otherwise, Mutate-3 is applicable.

Notice that the Mutate-1 rule decomposes $f(t_1, \cdots, t_n) \approx f(v_1, \cdots, v_n)$. The Mutate Rule for Syntactic Theories also decomposes $u \approx s$. However, that is only complete for Syntactic Theories. Our inference procedure is not just for Syntactic theories, and decomposing $u \approx s$ is not complete in our case.

Decomposition:

$$\frac{\{f(s_1, \cdots, s_n) \approx f(t_1, \cdots, t_n)\} \cup G}{\{s_1 \approx t_1, \cdots, s_n \approx t_n\} \cup G}$$

Mutate 1:

$$\frac{\{u \approx f(v_1, \cdots, v_n)\} \cup G}{\{u \approx s, t_1 \approx v_1, \cdots, t_n \approx v_n\} \cup G}$$

where $s \approx f(t_1, \cdots, t_n) \in E$. [a] [b]

Mutate 2:

$$\frac{\{u \approx v\} \cup G}{\{u \approx s, t_1 \approx s_1, \cdots, t_p \approx s_p, x \approx v\} \cup G}$$

where $s \approx f(t_1, \cdots, t_p), f(s_1, \cdots, s_p) \approx x \in E$.

Mutate 3:

$$\frac{\{f(u_1, \cdots, u_m) \approx v\} \cup G}{\{u_1 \approx s_1, \cdots, u_m \approx s_m, x \approx v\} \cup G}$$

where $f(s_1, \cdots, s_m) \approx x \in E$.

Variable Elimination:

$$\frac{\{x \approx t\} \cup G \cup H}{\{x \approx t\} \cup G[x \mapsto t] \cup H}$$

where $x \in G$, G is unsolved, and H is solved.

Orient:

$$\frac{\{t \approx x\} \cup G}{\{x \approx t\} \cup G} \quad \text{where } t \text{ is not a variable.}$$

Trivial:

$$\frac{\{t \approx t\} \cup G}{G}$$

[a] To be exact, $s \approx t$ is renamed to have no variables in common with the goal.

[b] For simplicity, we assume that E is closed under symmetry.

Fig. 1. The inference rules

We will write $G \longrightarrow G'$ to indicate that G goes to G' by one application of an inference rule. Then $\stackrel{*}{\longrightarrow}$ is the reflexive, transitive closure of $\longrightarrow$.

We want our inference rules to be applied deterministically or don't-care nondeterministically whenever possible. Therefore, we allow the Trivial, Orient and Variable Elimination rules to be performed eagerly. It is usual in inference systems for the Trivial and Orient rules to be performed eagerly. However, it is an open question in many inference systems whether the Variable Elimination rule can be applied eagerly. Since we restrict our inference rules to the case where E is linear and G is of varity 1, we can prove the completeness when Variable Elimination is performed eagerly. Eager inferences are a form of determinism, because when inferences are performed eagerly, that means that there is no need to backtrack and try other rules.

Inferences must be performed on a selected equation. This selection is a source of don't-care non-determinism in our procedure. However, there are still some sources of don't-know nondeterminism. If Decomposition and a Mutate rule (or two different Mutate rules) are both applicable to the selected equation, we don't know which one to do, and therefore have to try them both. Similarly, there may be more than one equation we can use in order to perform a Mutate rule on the selected equation. In that case, we must also try all the possibilities.

We will prove that the above inference rules solve a goal G by transforming it into normal forms representing a complete set of E-unifiers of G.

In order for the final result of the procedure to determine a unifier, it must not be cyclic. We will consider a goal G of varity 1 and a set of linear equations. Since G is of varity 1, it is not cyclic. We show that property is preserved.

Lemma 1. *Suppose that E is linear and $G \longrightarrow H$. If G is of varity 2, each term in G is of varity 1 and G is not cyclic, then H is of varity 2, each term in H is of varity 1 and H is not cyclic.*

A goal G is in *normal form* if the equations of G are all of the form $x \approx t$, where x is a variable, and the equations of G can be arranged in the form $\{x_1 \approx t_1, \cdots, x_n \approx t_n\}$ such that for all $i \leq j$, x_i is not in t_j. Then define θ_G to be the substitution $[x_1 \mapsto t_1][x_2 \mapsto t_2]\cdots[x_n \mapsto t_n]$. θ_G is a most general E-unifier of G. Notice that if a noncyclic goal has no selected equation, then the goal is in normal form, since Variable Elimination is applied eagerly.

4 A Bottom Up Inference System

In order to prove the completeness of this procedure, we first define an equational proof using Congruence and Equation Application rules. We prove that this equational proof is equivalent to the usual definition of equational proof, which involves Reflexivity, Symmetry, Transitivity and Congruence.

We will define a bottom-up inference system for ground terms, using the following rules of inference from a set of equations closed under symmetry:

$$\text{Congruence:} \quad \frac{s_1 \approx t_1 \cdots s_n \approx t_n}{f(s_1, \cdots, s_n) \approx f(t_1, \cdots, t_n)}$$

$$\text{Equation Application:} \quad \frac{u \approx s \quad t \approx v}{u \approx v},$$

if $\quad s \approx t \quad$ is a ground instance of an equation in E.

For Congruence $n \geq 0$. In the special case where $n = 0$, f is a constant.

We define $E \vdash u \approx v$ if there is a proof of $u \approx v$ using the Congruence and Equation Application rules. If π is a proof, then $|\pi|_E$ is the ordered pair (m, n), where m is the number of Equation Application steps in π and n is the number of Congruence steps. These ordered pairs are compared lexicographically, and addition is defined by components, i.e., $(m, n) + (p, q) = (m + n, p + q)$. $|u \approx v|_E$ is the pair (m, n), which is the minimum $|\pi|_E$ such that π is a proof of $u \approx v$.

We need to prove that $\{u \approx v \,|\, E \vdash u \approx v\}$ is closed under Reflexivity, Symmetry and Transitivity Also, we need to prove that certain rotations of a proof can be done without making the proof any larger (see [13]).

Theorem 1. *If u and v are ground and $E \models u \approx v$, then $E \vdash u \approx v$.*

5 Completeness

In this section, we will state the completeness of the inference rules given in Figure 1, where E is linear, and G is of varity 1. See [13] for the proof. First we define a measure on the equations in the goal, which will be used in the completeness proof.

Definition 1. *Let E be an equational theory and G be a goal. Let θ be a substitution such that $E \models G\theta$. We will define a measure μ, parameterized by θ and G. Let m (resp. q) be the sum of all the first (resp. second) components of $|u\theta \approx v\theta|_E$, where $u \approx v$ is an unsolved equation of G. Let n be the number of unsolved variables in G. Let p be the number of equations of the form $t \approx x$, where x is a variable and t is not. Then Define $\mu(G, \theta)$ to be the quadruple (m, n, q, p). We will compare these quadruples lexicographically.*

Now we come to the completeness theorem, which says that every E-unifier can be gotten from our algorithm. The proof of the theorem shows that if there is a goal which is not in normal form, then an inference can be performed to reduce the measure of the goal. Variable Elimination, Orient and Trivial always reduce the measure of the goal.

Theorem 2. *Suppose that E is an equational theory, G is a set of goal equations, E is linear, G is of varity 2, every term in G is varity 1, G is not cyclic, and θ is a ground substitution. If $E \models G\theta$ then there exists a goal H in normal form such that $G \stackrel{*}{\longrightarrow} H$ and $\theta_H \leq_E \theta[Var(G)]$.*

Corollary 1. *Suppose that E is an equational theory, G is a set of goal equations, E is linear, G is of varity 1, and θ is a ground substitution. If $E \models G\theta$ then there exists a goal H such that $G \stackrel{*}{\longrightarrow} H$ and $\theta_H \leq_E \theta[Var(G)]$.*

6 Linear Standard Theories

In this section we consider Linear Standard (LS) theories.

Definition 2. *An equation $u \approx v$ is LS if u and v are linear, and every variable that is shared by u and v is at depth 1 or 0 in u and also at depth 1 or 0 in v. A set of equations E is LS if every equation in E is LS.*

For example, $f(g(h(x_1)), x_2, h(g(x_3)), x_4) \approx k(x_2, x_4, k(x_5, a, x_6))$ is LS. So is the the collapsing equation $f(x) \approx x$. Some examples that are not LS are $f(x, x) \approx g(a)$ and $f(x, f(y)) \approx g(y)$. The first one is not LS because $f(x, x)$ is not linear. The second one is not LS, because y appears on both sides, but is at depth 2 on the right side.

Throughout this section, we will refer to equational theories E that are LS and goals G that are varity 1. We consider the E-unification problem for such theories and goals. For simplicity, since no variable is repeated in a goal, we will consider goals consisting of a single equation, because each equation can be E-unified separately, and the results can be combined.

We have defined inference rules for linear theories, and shown their completeness and soundness. For LS theories and varity 1 goals, we will derive an algorithm that always halts, and show therefore that this kind of E-unification is decidable, and then analyze its complexity.

In our completeness result, we proved that Variable Elimination and Orient and Trivial can be performed eagerly. Therefore, we will refer to Mutate+ inference rules (Mutate 1+, Mutate 2+ and Mutate 3+). These inference rule will consist of Mutate, plus some eager Variable Eliminations.

Mutate 1+:

$$\frac{\{u \approx f(v_1, \cdots, v_n)\} \cup G}{\{u \approx s\sigma, t_1 \approx v_1, \cdots, t_n \approx v_n\} \cup G}$$

where $s \approx f(t_1, \cdots, t_n) \in E$ and $\sigma = \{t_j \mapsto v_j \mid t_j \in Vars, 1 \leq j \leq n\}$.

Mutate 2+:

$$\frac{\{u \approx v\} \cup G}{\{u \approx s\sigma, t_1 \approx s_1, \cdots, t_i \approx v, \cdots, t_p \approx s_p, x \approx v\} \cup G}$$

where $s \approx f(t_1, \cdots, t_p), f(s_1, \cdots, s_p) \approx x \in E$, $s_i = x$, and $\sigma = \{t_j \mapsto s_j \mid t_j \in Vars, 1 \leq j \leq n\}$.

Mutate 3+:

$$\frac{\{f(u_1, \cdots, u_m) \approx v\} \cup G}{\{u_1 \approx s_1, \cdots, u_i \approx v, \cdots, u_m \approx s_m, x \approx v\} \cup G}$$

where $f(s_1, \cdots, s_m) \approx x \in E$ and $s_i = x$.

Next we prove that the property that the unsolved part of a goal is varity 1 is preserved by the inference rules. Recall that this means that no variable is repeated anywhere else in the entire goal.

Lemma 2. *Let G be a goal such that the unsolved part of G is of varity 1, and E is LS. Suppose $G \longrightarrow G'$. Then the unsolved part of G' is of varity 1.*

Furthermore, suppose that no variable appears more than twice in G, and any variable y that does appear twice in G appears in a term t in a solved equation of the form $x \approx t$ with x a variable from E. Then the same thing is true in G'

Corollary 2. *Suppose E is LS. If G is of varity 1, and $G \stackrel{*}{\longrightarrow} G'$, then the unsolved part of G' is of varity 1. Furthermore, the Variable Elimination rule is not used in the derivation of G', except as part of a Mutate+ inference rule.*

Next, we give a description of what terms can appear in a derivation. First we give a recursive definition of a *decomposition* of an equation $u \approx v$

Definition 3. Decomposition *of an equation $u \approx v$ is defined recursively as*

1. *$u \approx v$ is a* decomposition of $u \approx v$.
2. *If $s \approx t$ is a decomposition of $u \approx v$ then $t \approx s$ is a* decomposition of $u \approx v$.
3. *If $f(u_1, \cdots, u_n) \approx f(v_1, \cdots, v_n)$ is a decomposition of $u \approx v$ then $u_i \approx v_i$ is a* decomposition of $u \approx v$ *for all i, $1 \leq i \leq n$.*

For example, the equation $f(g(x), h(y)) \approx f(g(h(a)), f(z))$ has four decompositions. They are $f(g(x), h(y)) \approx f(g(h(a)), f(z))$, $g(x) \approx g(h(a))$, $h(y) \approx f(z)$ and $x \approx h(a)$. Notice that if $s \approx t$ is a decomposition of $u \approx v$ then there exists a position i such that $u|_i = s$ and $v|_i = t$, or $v_| = s$ and $u|_i = t$.

Decompositions of the goal can appear in a derivation. We need some more definitions before we can say what else can appear in a derivation.

Definition 4.
- *$St(t)$ is the set of all subterms of t. $St(s \approx t) = St(s) \cup St(t)$. $St(E) = \bigcup\{St(e) \mid e \in E\}$.*
- *$Pr(t)$ is the set of all proper subterms of t. $Pr(s \approx t) = Pr(s) \cup Pr(t)$. $Pr(E) = \bigcup\{Pr(e) \mid e \in E\}$.*
- *$Im(t)$ is the set of all immediate subterms of t, i.e., $t_i \in Im(f(t_1, \cdots, t_n))$ for all i, $1 \leq i \leq n$.*
- *$Ren(t)$ is the set of all renamings(variants) of t*

Some instances of terms can appear in the derivation, called shallow instances, because only the shared shallow variables are instantiated.

Definition 5. *$s \in Sh(t, E, u)$ (s is a shallow (t, E)* instance *of u) if there is a variant $u' \approx v'$ of an equation $u \approx v \in E$ and a substitution σ such that*

1. *The domain of σ is the set of all shared variables in $u' \approx v'$.*
2. *$Ran(\sigma) \subseteq Im(t) \cup Ren(Pr(E))$.*
3. *$s = u'\sigma$.*

$s \in Sh(t, E)$ if there is a u such that $s \in Sh(t, E, u)$.

For example, suppose that t is $f(h(x, y), f(c, d))$, and E is $\{g(h(x, y), z, w) \approx f(z, w), h(f(a, x), y) \approx f(y, g(a, a, b))\}$. Then $g(h(x', y'), z', w')$ and $g(h(x', y'), h(x, y), f(a, x''))$ are both shallow (t, E) instances of $g(h(x, y), z, w)$.

The next definition shows what can appear in a derivation.

Definition 6. – *$s \in Der(E, e)$ if there is a t in $St(e)$ with a symbol from E as its top symbol, such that $s \in Im(t) \cup Ren(St(E)) \cup Sh(t, E)$.*
– *$s \approx s' \in Der(E, e)$ if s and s' are in $Der(E, e)$.*
– *$G \in Der(E, e)$ if every $s \approx s' \in G$ is in $Der(E, e)$.*

We need a small proposition about this definition.

Proposition 1. *Let $s \in Der(E, e)$. There there is a $t \in St(e)$ such that $Im(s) \subseteq Im(t) \cup Ren(Pr(E))$.*

From Proposition 1, we see that all immediate subterms (and therefore all subterms) of a term in $Der(E, e)$ are also in $Der(E, e)$.

We prove that only those kinds of equations can appear in a derivation.

Lemma 3. *Let E be LS. Let G be varity 1 and $G \in Der(E, e)$. Suppose $G \longrightarrow G'$. Then $G' \in Der(E, e)$.*

Corollary 3. *Suppose E is LS. If e is of varity 1, and $e \stackrel{*}{\longrightarrow} G'$, then $G' \in Der(E, e)$.*

We get a better complexity result when the equations of E are varity 1.

Lemma 4. *Let E be varity 1. Let G be varity 1. Suppose $G \longrightarrow G'$. Suppose that every side of an equation in G is in $Ren(St(E)) \cup Im(t)$ for some $t \in St(e)$. Furthermore, suppose that each equation in G is a Decomposition or one side is in $Ren(St(E))$. Then G' has those same properties.*

Corollary 4. *Suppose E and G are varity 1. If $e \stackrel{*}{\longrightarrow} G'$, Then every side of an equation in G is in $Ren(St(E)) \cup Im(t)$ for some $t \in St(e)$. Also, each equation in G is a Decomposition or one side is in $Ren(St(E))$.*

Since we have shown that Variable Elimination is not applicable, all of our inference rules can be expressed as Horn clauses, where the head of the clause is the selected literal, and the body is the result of the inference on the selected literal. In fact the variables which are introduced in the body of the Horn clause can be skolemized, again because of the fact that the Variable Elimination rule is not applicable. We define a Skolem function Sk which turns a variable into a constant, i.e., $Sk(t) = t\theta$, where $\theta = \{x \mapsto c \mid x \in Vars(t)\}$. Note that every variable maps to the same constant, since the constant is not important. We also add Horn clauses to eliminate solved variables. Therefore, the inference rules are expressed by the following Horn clauses.

Decomposition, Orient and Trivial are expressed as:

$$f(x_1, \cdots, x_n) \approx f(y_1, \cdots, y_n) \leftarrow x_1 \approx y_1, \cdots, x_n \approx y_n$$

$$c \approx y \leftarrow y \approx c$$

$$x \approx x \leftarrow$$

The Mutate 1+ rule is:

$$y \approx f(x_1, \cdots, x_n) \leftarrow (y \approx s\sigma, t_1 \approx x_1, \cdots, t_n \approx x_n)\theta$$

where $s \approx f(t_1, \cdots, t_n) \in E$, $\sigma = \{t_j \mapsto x_j \mid s_j \in Vars, 1 \leq j \leq n\}$, and $\theta = \{x' \mapsto c \mid x' \in Vars(s \approx f(t_1, \cdots, t_n)$.

The Mutate 2+ rule is:

$$y \approx z \leftarrow (y \approx s\sigma, t_1 \approx s_1, \cdots, t_i \approx z, \cdots, t_p \approx s_p, x \approx z)\theta$$

where $s \approx f(t_1, \cdots, t_p), f(s_1, \cdots, s_p) \approx x \in E$, $s_i = x$, $\sigma = \{t_j \mapsto s_j \mid s_j \in Vars, 1 \leq j \leq n\}$, and $\theta = \{x' \mapsto c \mid x' \in Vars(s \approx f(t_1, \cdots, t_p)) \cup Vars(f(s_1, \cdots, s_p) \approx x)$.

The Mutate 3+ rule is:

$$f(x_1, \cdots, x_m) \approx y \leftarrow (x_1 \approx s_1, \cdots, x_i \approx v, \cdots, x_m \approx s_m, x \approx v)\theta$$

where $f(s_1, \cdots, s_m) \approx x \in E$, $s_1 = x$, and
$\theta = \{x' \mapsto c \mid x' \in Vars(f(s_1, \cdots, s_m) \approx x)\}$.

We also add an inference rule to remove solved variables:

$$c \approx y \leftarrow$$

Therefore, each equational theory E determines a particular set of Horn clauses. Let us denote this set of Horn clauses as $HC(E)$. Note that the equality in these Horn clauses is now interpreted just as a binary predicate symbol with no special meaning. We have the following result:

Theorem 3. *Let e be a goal of varity 1 and E be LS. Then e is E-unifiable if and only if there is an SLD derivation of $Sk(e)$ in $HC(E)$.*

Corollary 5. *Let e be a goal of varity 1 and E be LS. Then e is E-unifiable if and only if $HC(E) \models Sk(e)$*

We notice that only certain ground instances will arise in the SLD refutation. Let $HC(e, E)$ be the set of all instances of $HC(E)$ such that the head of the clause is in $Sk(Der(E, e))$. Then we have the following theorem:

Theorem 4. *Let e be a goal of varity 1 and E be LS. Then e is E-unifiable if and only if there is an SLD derivation of $Sk(e)$ in $HC(e, E)$.*

Corollary 6. *Let e be a goal of varity 1 and E be LS. Then e is E-unifiable if and only if $HC(e, E) \models Sk(e)$*

Finally we have the decidability and complexity theorem of E-unification for varity 1 goals in LS theories.[2]

Theorem 5. *Suppose that E is LS and e is varity 1. Then*

1. *It is decidable in polynomial time whether e is E-unifiable.*
2. *If E is considered constant, then it is decidable in $O(|e|^2)$ whether e is E-unifiable.*
3. *If E is considered constant, and all equations in E are varity 1, then it is decidable in $O(|e|)$ whether e is E-unifiable.*

[2] Note: $|e|$ refers to the size of e (number of symbols).

7 Most General E-unifiers

In this section we extend our results on LS theories and varity 1 goals to examine how efficient it is to compute a complete set of E-unifiers. First, we show that, there is a complete set of unifiers such that the range of every substitution in the set only contains terms from $Der(E, e)$. This result, along with the polynomial time algorithm for deciding E-unifiability leads us to all the rest of the results of the section, which are independent of the algorithm we have given.

We show that there is a complete set of E-unifiers no bigger than simply exponential. However, we give an example of a ground theory where the size of the minimal complete set of E-unifiers is simply exponential. So the bound is tight. Even though the size can be exponential, it still has some nice properties. We define a complete set of E-unifiers $CSU_E(u \approx v)$, and show that given a substitution σ, it can be decided in polynomial time whether $\sigma \in CSU_E(u \approx v)$.

Using those results, we finally move to the general E-unification problem for LS theories. In other words, we no longer restrict ourselves to varity 1 goals. We give a NEXPTIME algorithm for deciding unifiability. It is known that E-unification is NP hard, even for ground theories[18]. This leaves us with a gap for the actual complexity of the problem. The complete set of E-unifiers that we construct may be doubly exponential in size. However, all terms appearing in a substitution in the complete set of E-unifiers have depth linear in the maximum of the depths of the terms in the goal and the equational theory.

First we show that it is decidable in polynomial time if σ is an E-unifier of $u \approx v$ We will define $Gr(u \approx v)$ to be an instance of $u \approx v$ such that each variable in $u \approx v$ is replaced by a different new constant.

Theorem 6. *Let E be LS. Then it is decidable in polynomial time whether σ is an E-unifier of $u \approx v$.*

The next result in this section refers to the earlier completeness results.

Theorem 7. *Suppose E is LS and $u \approx v$ is of varity 1. Then there is a complete set of E-unifiers Θ of $u \approx v$ such that if $\sigma \in \Theta$ then $Ran(\sigma) \subseteq Der(E, e)$.*

Using that result, we can show that there is a complete set of E-unifiers with at most simply exponentially many members.

Theorem 8. *If E is LS and $u \approx v$ is varity 1 then there is a simply exponential size complete set of E-unifiers of $u \approx v$.*

There are goals which have simply exponential sized minimal complete sets of E-unifiers, even in ground theories, so this is a tight bound.

Theorem 9. *There is a ground theory E and a goal $u \approx v$ of varity 1, such that every minimal complete set of E-unifiers of $u \approx v$ has exponentially many members.*

A complete set of E-unifiers of $u \approx v$ can be described as follows: $CSU_E(u \approx v) = \{\sigma \mid Dom(\sigma) = Vars(u \approx v), Ran(\sigma) \subseteq Der(E, e)$, and σ is an E-unifier of $u \approx v$ $\}$. We can decide in polynomial time whether a given σ is a member of $CSU_E(u \approx v)$.

Theorem 10. *Let E be LS, and e be varity 1. Let σ be a substitution. Then it is decidable in polynomial time if $\sigma \in CSU_E(e)$*

Finally, we move to the most general case of E-unification, where E is LS but the goal e is not necessarily varity 1. It can be in any form.

Theorem 11. *Let E by LS and e be a goal. Then*

1. *E-unification for e is decidable in NEXPTIME.*
2. *There is a complete set, Θ, of E-unifiers of e which is of doubly exponential size.*
3. *Every term appearing in the range of a substitution of Θ has depth linear in the maximum depth of the terms in E and e.*

8 Conclusion

This paper presents a new technique for showing decidability and complexity results for E-unification problems. It makes it easy to analyze what forms of subgoals will arise from the initial goal equation. That can give useful information used to make the procedure halt, and then an examination of what kinds of equations are generated allows us to determine the complexity.

One application of the results of this paper is for approximating E-unification problems. For any theory and goal, we can rename all the variables in the theory and goal to new variables until they are varity 1.[3] As we showed, a linear time algorithm can be run on the new problem, and if the algorithm says "not E-unifiable", then that is also true of the initial E-unification problem. This is useful in the context of automated deduction problems that require lots of E-unification, and would allow a quickly discarding many E-unification problems.

There are some relationships with some of our other papers. In [15], we gave a goal directed inference system for E-unification in a similar style. The method of showing soundness and completeness in that paper is similar to the method in this paper. However, this time the inference system is different, and the Eager Variable Elimination rules make the proof more difficult. That paper had no decidability or complexity results.

Another recent work of ours[14] also develops decidability and complexity results for a class of equational theories and goals of varity 1. However, the class of problems in that paper is not a syntactic class, and the complexity results are not as good. We have actually shown in this paper that Linear Standard theories are in that class, because any E-unification problem whose goal contains only

[3] This is more renaming than necesary, because it removes all variables shared by both sides of an equation.

subterms of E or general terms will have a complete set of E-unifiers, such that the range of every substitution in the complete set only contains goal terms.

We would also like to compare our work to other recent works, which show E-unification decidable for syntactic classes such as linear standard theories.[4, 3, 8, 18, 9]. There are three basic approaches to the problem: saturation based theorem proving methods like completion[18, 9], tree automata techniques[3, 8], and goal-directed inference rule methods[4]. Actually, [9] shows a relationship between their saturation methods and tree automata techniques. The methods of [4] are quite different from ours, even though they both use goal directed inference rule methods. One difference is that [4] saturate E by completion-like inference rules to make it syntactic. We do not do that, since our inference procedure is not limited to Syntactic Theories. In fact, all of the methods except ours pre-process E using something like completion. Therefore our method benefits from a memoization technique (as opposed to dynamic programming) that the other methods may not benefit from. On the other hand, there is one thing that all the methods share in common. They all are based on the fact that only certain terms will appear during the procedure.

Most of those other techniques have been used to show decidability results. Complexity results and bounds on the size of the minimal complete set of unifiers are not usually addressed. However, these issues are addressed quite nicely for shallow theories in [18]. The main benefit of our paper is to focus more on complexity. Our quadratic bound is interesting, because the results on shallow theories give a polynomial bound for the word problem, but not the exact polynomial. We suspect that the techniques used in this paper to analyze complexity could be used in other methods. We also discuss E-unification for varity 1 goals. There is a mention of E-unification for varity 1 goals in shallow linear theories in [9], where they give a simple decidability result using tree automata.

Since we reduced our E-unification problem to a Horn clause implication problem, and then showed only certain instances of the Horn Clauses are necessary for the derivation, it is natural to ask whether these Horn clauses are stably local[7], i.e. if variables only need to be substituted by terms appearing in the theory and goal. The presented Horn clause theory is not stably local, but if the single variables appearing in the horn clause were replaced by all possible terms of the form $f(x_1, \cdots, x_n)$, then the theory would be stably local. The initial equational theory is not stably local, because the shallow instances take us out of the set of subterms, and also because the shared variables may need to unify with terms that are not even in $Der(E, e)$.

We plan to extend the algorithm and techniques presented in this paper to get other decidability and complexity results. We would like to know what other classes can be shown decidable and efficient using this method. We have already started analyzing other syntactic forms of linear theories. But we also will consider non-linear theories. We left a gap in the complexity results for general E-unification of linear standard theories, which needs to be filled. Also, we are interested in finding better ways of approximating equational theories. Finally, there should be a closer examination of the relationship between our goal-directed method and the saturation-based and tree automata methods. Can our complexity techniques be used there? Maybe all these methods are encodings of the same process.

References

1. F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge, 1998.
2. D. Basin and H. Ganzinger. Automated complexity analysis based on ordered resolution. In *J. Association for Computing Machinery* 48(1), 70-109, 2001.
3. H. Comon. Sequentiality, second order monadic logic and tree automata. In *Proceedings 10th IEEE Symposium on Logic in Computer Science, LICS'95*, IEEE Computer Society Press, 508-517, 1995.
4. H. Comon, M. Haberstrau and J.-P. Jouannaud. Syntacticness, Cycle-Syntacticness and shallow theories. In *Information and Computation* 111(1), 154-191, 1994.
5. W. Dowling and J. Gallier. Linear-time algorithms for testing the satisfiability of propositional horn formulae. In *Journal of Logic Programming 3*, 267-284, 1984.
6. J. Gallier and W. Snyder. Complete sets of transformations for general E-unification. In *TCS*, vol. 67, 203-260, 1989.
7. H. Ganzinger. Relating Semantic and Proof-Theoretic Concepts for Polynomial Time Decidability of Uniform Word Problems. In *Proceedings 16th IEEE Symposium on Logic in Computer Science, LICS'2001*, Boston, 2001.
8. F. Jacquemard. Decidable approximations of term rewriting systems. In H. Ganzinger, ed., *Rewriting Techniques and Applications, 7th International Conference, RTA-96*, Vol. 1103 of LNCS, Springer, 362-376, 1996.
9. F. Jacquemard, Ch. Meyer, Ch. Weidenbach. Unification in Extensions of Shallow Equational Theories. In T. Nipkow, ed., *Rewriting Techniques and Applications, 9th International Conference, RTA-98*, Vol. 1379, LNCS, Springer, 76-90, 1998.
10. C. Kirchner. Computing unification algorithms. In *Proceedings of the Fourth Symposium on Logic in Computer Science*, Boston, 200-216, 1990.
11. D. Kozen. Complexity of finitely presented algebras. In *Proc. 9th STOC*, 164-177, 1977.
12. D. Kozen. Positive first order logic is NP-complete. *IBM Journal of Res. Developp.*, 25(4):327-332, July 1981.
13. C. Lynch and B. Morawska. Complexity of Linear Standard Theories. `http://www.clarkson.edu/~clynch/papers/standard_full.ps/`, 2001.
14. C. Lynch and B. Morawska. Decidability and Complexity of Finitely Closable Linear Equational Theories. In R. Goré, A. Leitsch and T. Nipkow, eds., *Automated Reasoning. First International Joint Conference, IJCAR 2001*, Vol. 2083 of LNAI, 499-513, Springer, 2001.
15. C. Lynch and B. Morawska. Goal Directed E-Unification. In *RTA 12*, ed. A. Middeldorp, LNCS vol. 2051, 231-245, 2001.
16. D. McAllester. Automated Recognition of Tractability in Inference Relations. In *Journal of the ACM*, vol.40(2) , pp. 284-303, 1993.
17. A. Middeldorp, S. Okui, T. Ida. Lazy Narrowing: Strong Completeness and Eager Variable Elimination. In *Theoretical Computer Science* 167(1,2), pp. 95-130, 1996.
18. R. Nieuwenhuis. Basic paramodulation and decidable theories. (Extended abstract), In *Proceedings 11th IEEE Symposium on Logic in Computer Science, LICS'96*, IEEE Computer Society Press, 473-482, 1996.