COMPLETE SETS OF TRANSFORMATIONS FOR GENERAL E-UNIFICATION

Jean H. Gallier and Wayne Snyder
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, Pa 19104

April 7, 2025

This research was partially supported by the National Science Foundation under Grant No. DCR-86-07156.

COMPLETE SETS OF TRANSFORMATIONS FOR GENERAL E-UNIFICATION

Jean H. Gallier and Wayne Snyder

Abstract: This paper is concerned with E-unification in arbitrary equational theories. We extend the method of transformations on systems of terms, developed by Martelli-Montanari for standard unification, to E-unification by giving two sets of transformations, $\mathcal{BT}$ and $\mathcal{T}$, which are proved to be sound and complete in the sense that a complete set of E-unifiers for any equational theory E can be enumerated by either of these sets. The set $\mathcal{T}$ is an improvement of $\mathcal{BT}$, in that many E-unifiers produced by $\mathcal{BT}$ will be weeded out by $\mathcal{T}$. In addition, we show that a generalization of surreduction (also called narrowing) combined with the computation of critical pairs is complete. A new representation of equational proofs as certain kinds of trees is used to prove the completeness of the set $\mathcal{BT}$ in a rather direct fashion that parallels the completeness of the transformations in the case of (standard) unification. The completeness of $\mathcal{T}$ and the generalization of surreduction is proved by a method inspired by the concept of *unfailing completion*, using an abstract (and simpler) notion of the completion of a set of equations.

1 Introduction

This paper is concerned with E-unification in arbitrary equational theories using the method of transformations on term systems. We present several sets of transformations and show them to be sound and complete, in the sense that given any equational theory E, a complete set of E-unifiers will be enumerated using transformations in any of these sets.

Given a (finite) set E of equations and two terms u and v, a substitution θ is an *E-unifier of u and v* iff $\theta(u)$ and $\theta(v)$ are provably equal under the equations in E, that is, congruent modulo the least stable congruence $\stackrel{*}{\longleftrightarrow}_E$ containing E. The problem of finding E-unifiers is called the *E-unification problem*. When $E = \emptyset$, σ is called a *unifier of u and v*, and the problem is called the *unification problem* (see [36]). The importance of unification and E-unification stems from the fact that unification is one of the most crucial operations used in theorem provers and logic program interpreters. For instance, unification is the basic mechanism for computing answers of queries used by Prolog. In view of the inherent inefficiency of theorem proving methods in the presence of equality, Robinson [37] and then Plotkin [34] suggested that theorem provers be stratified into a (non-equational) refutation mechanism, and an E-unification mechanism, which performs equational reasoning during unification steps. More recently, E-unification has been proposed as the theoretical basis of the incorporation of functional and equational languages into the basic paradigm of logic programming [10,12].

Unification and E-unification differ considerably in complexity. Unification is decidable and fast unification algorithms exist (in fact, linear time algorithms are known [33]), but E-unification is undecidable, due to the undecidability of the word problem for semigroups. Another major difference has to do with the existence of most general unifiers. In the case of unification, if two terms are unifiable then they have a *most general unifier*, or *mgu*, a unifier σ such that every other unifier θ may be obtained by composing σ with some other substitution ($\theta = \sigma \circ \eta$ for some η). Unification algorithms produce mgu's. Unfortunately, for an arbitrary E, if u and v are E-unifiable there may not be a single mgu. Instead, sets of mgu's must be considered. In simple terms, we say that a set U of substitutions is a *complete set of unifiers* for u, v iff every $\sigma \in U$ is a E-unifier of u, v, and for every E-unifier θ of u, v, there is some σ in U and some substitution η such that $\theta =_E \sigma \circ \eta[Var(u, v)]$. Thus, complete sets of E-unifiers play the role of mgu's. Unfortunately, complete sets of E-unifiers are not necessarily finite. At best, they are partially recursive (if E is recursive).

In the case of unification, there is an elegant and powerful method due to Martelli and Montanari [30] (but already sketched in Herbrand's thesis [13]) for finding mgu's: the method of *transformations* on term systems. This method consists of a set $\mathcal{ST}$ of four simple transformations (three, if two-element multisets instead of ordered pairs are considered) that

are used to decompose and solve simple term systems.

This paper addresses the problem of finding complete sets of transformations $\mathcal{T}$ extending the set $\mathcal{ST}$ to account for the presence of arbitrary equations. We say that a set $\mathcal{T}$ is *complete* iff for every set E of equations, a complete set of E-unifiers can be enumerated using transformations in $\mathcal{T}$. In addition, we would like to find complete sets of transformations which eliminate as many redundant E-unifiers as possible. This is a very difficult task, because under a reasonable definition of minimality, minimal complete sets of E-unifiers may not exist [7]. We present two sets of transformations $\mathcal{BT}$ and $\mathcal{T}$ and prove that they are complete for arbitrary sets of equations. The set $\mathcal{T}$ is an improvement of $\mathcal{BT}$, in that many redundant E-unifiers produced by $\mathcal{BT}$ will be weeded out by $\mathcal{T}$. In addition, we show that a generalization of surreduction (also called narrowing) combined with the computation of critical pairs is complete.

Although $\mathcal{BT}$ only contains two more transformations than $\mathcal{ST}$, and $\mathcal{T}$ one more transformation than $\mathcal{ST}$, proving the completeness of $\mathcal{BT}$ and $\mathcal{T}$ (and also of the generalization of surreduction) turned out to be quite difficult. We were led to define a new representation of equational proofs as certain kinds of (sets of) trees. These proof trees are used to prove the completeness of the set $\mathcal{BT}$ in a rather direct fashion that parallels the completeness of the simple set $\mathcal{ST}$ in the case of (standard) unification. In order to prove the completeness of $\mathcal{T}$, inspired by the concept of *unfailing completion* [1,2,3], we developed an abstract (and simpler) notion of the completion of a set of equations that allowed us to use the previous completeness proof. The completeness of the generalization of surreduction also uses this abstract completion. We then give a second proof of the completeness of $\mathcal{T}$ based on the completeness of the generalization of surreduction. In a sense made precise when these results are proved, the first completeness result about $\mathcal{T}$ (theorem 6.8) is stronger than the second completeness result (theorem 8.3).

This paper generalizes the approach initiated in the pioneering work of Kirchner [22,23] to arbitrary theories. One of the main important technical differences between our work and Claude Kirchner's is that we use transformations extending naturally those proposed by Herbrand [13], whereas Kirchner uses transformations closer to those Martelli and Montanari developed for multiequations [30]. Also, Kirchner's transformations are only complete for a subclass of all equational theories, the strict theories. Nevertheless, our work would not have been possible without Claude Kirchner's previous contribution. Another concept that inspired us at a crucial time is the idea of unfailing completion, due to Bachmair, Dershowitz, Hsiang, and Plaisted [1,2,3]. Without this research, we would not have been able to show the completeness of our improved set of transformations $\mathcal{T}$.

This paper is an expanded and corrected version of [11], where our results are pre-

sented in preliminary form. In particular, the set of transformations $\mathcal{T}'$ used in [11] is equivalent to $\mathcal{T}$ but with some additional restrictions. The proof that $\mathcal{T}'$ is a complete set of transformations turned out to have a serious gap that remains to be filled. The difficulty has to do with the strategy of "eager" variable elimination discussed in section 10.

The plan of this paper is as follows. After presenting a number of preliminary definitions, we proceed to develop in §3 an abstract view of standard unification, due to [30], as a set $\mathcal{ST}$ of transformation rules for non-deterministically transforming a unification problem into an explicit representation of its solution (if such exists). This set of rules is then extended in the next section to a basic set of transformations $\mathcal{BT}$ which accounts for the presence of arbitrary equations in a unification problem. In §5 we develop techniques which allow us to restrict rewriting at or below variable occurrences, and which we then use in §6 to prove the completeness of an improved set of transformations $\mathcal{T}$. In sections §7 and §8, a weaker version of the completeness proof for this set is established using the notion of a surreduction (or narrowing) step. The final sections of the paper discuss previous work on more general forms of E-unification, open problems, and our current research.

2 Preliminaries

In order that this paper be self-contained, this section contains an outline of the major definitions and results related to E-unification, and is basically consistent with [17] and [9]. We begin with the basic algebraic notions of trees and substitutions.

Definition 2.1 Let $\mathbf{N}$ be the set of natural numbers. A *ranked alphabet* is a set Σ with an associated function $arity : \Sigma \to \mathbf{N}$ assigning a *rank* or *arity* n to each symbol f in Σ. We denote the set of symbols of arity n by Σ_n. (For example, the set of constants is just Σ_0.)

Definition 2.2 Let $\mathbf{N}_+$ denote the set of positive natural numbers. A *tree domain* D is a nonempty subset of strings in $\mathbf{N}_+^*$ satisfying the conditions:

(i) For all $\alpha, \beta \in \mathbf{N}_+^*$, if $\alpha\beta \in D$ then $\alpha \in D$.

(ii) For all $\alpha \in \mathbf{N}_+^*$, for every $i \in \mathbf{N}_+$, if $\alpha i \in D$ then, for every j, $1 \leq j \leq i$, $\alpha j \in D$.

Definition 2.3 Given a ranked alphabet Σ, a Σ-*tree* (or *term*) is any function $t : D \to \Sigma$ where D is a tree domain denoted by $Dom(t)$ and if $\alpha \in Dom(t)$ and $\{i \mid \alpha i \in Dom(t)\} = \{1, \ldots, n\}$, then $arity(t(\alpha)) = n$. We shall denote the symbol $t(\epsilon)$ by $Root(t)$. Given a tree t and some tree address $\alpha \in Dom(t)$, the *subtree of* t *rooted at* α is the tree,

denoted t/α, whose domain is the set $\{\beta \mid \alpha\beta \in Dom(t)\}$ and such that $t/\alpha(\beta) = t(\alpha\beta)$ for all $\beta \in Dom(t/\alpha)$. Given two trees t_1 and t_2 and a tree address α in t_1 the *result of replacing* t_2 at α in t_1, denoted by $t_1[\alpha \leftarrow t_2]$, is the function whose graph is the set of pairs $\{(\beta, t_1(\beta)) \mid \beta \in Dom(t_1)$ such that α is not a prefix of $\beta\} \cup \{(\alpha\beta, t_2(\beta)) \mid \beta \in Dom(t_2)\}$.

The set of all finite trees is denoted by T_Σ. Given a countably infinite set of variables $X = \{x_0, x_1, \ldots\}$, we can form the set of trees $T_\Sigma(X)$ by adjoining the set X to the set Σ_0. Thus, $T_\Sigma(X)$ is the set of all terms formed from the constant and function symbols in Σ and the variables in X.

The *size* of a term t is the number of occurrences of function and constant symbols and variables in the term, i.e., the cardinality of $Dom(t)$. We shall denote the *depth* of a term t, i.e., the length of the longest path in t (or, equivalently, the length of the longest string in $Dom(t)$), by $|t|$. For example, $|f(a)| = 1$ and $|c| = 0$. The set of variables occurring in a term t is the set

$$Var(t) \;=\; \{x \in X \mid t(\alpha) = x \text{ for some } \alpha \in Dom(t)\,\}.$$

Any term t for which $Var(t) = \emptyset$ is called a *ground term*.

In the rest of this paper, we shall use the letters a, b, c, and d to denote constants; f, g, and h to denote functions; l, r, s, t, u, v, and w for terms; and α, β, and γ for tree addresses.

In order that $T_\Sigma(X)$ be non-empty, we assume that $\Sigma_0 \cup X \neq \emptyset$. Thus $T_\Sigma(X)$ is the free Σ-algebra generated by X. This property allows us to define substitutions.

Definition 2.4 A *substitution* is any function $\theta : X \to T_\Sigma(X)$ such that $\theta(x) \neq x$ for only finitely many $x \in X$. Since $T_\Sigma(X)$ is freely generated by X, every substitution $\theta : X \to T_\Sigma(X)$ has a unique homomorphic extension $\widehat{\theta} : T_\Sigma(X) \to T_\Sigma(X)$. In the sequel, we will identify θ and its homomorphic extension $\widehat{\theta}$.

Definition 2.5 Given a substitution σ, the *support* (or *domain*) of σ is the set of variables $D(\sigma) = \{x \mid \sigma(x) \neq x\}$. A substitution whose support is empty is termed the *identity substitution*, and is denoted by Id. The set of variables *introduced by* σ is $I(\sigma) = \bigcup_{x \in D(\sigma)} Var(\sigma(x))$. Given a substitution σ, if its support is the set $\{x_1, \ldots, x_n\}$, and if $t_i = \sigma(x_i)$ for $1 \leq i \leq n$, then σ is also denoted by $[t_1/x_1, \ldots, t_n/x_n]$. Given a term r, we also denote $\sigma(r)$ as $r[t_1/x_1, \ldots, t_n/x_n]$. A substitution ρ is a *renaming substitution away from* W if $\rho(x)$ is a variable for every $x \in D(\rho)$, $I(\rho) \cap W = \emptyset$, and for every $x, y \in D(\theta)$, $\rho(x) = \rho(y)$ implies that $x = y$. If W is unimportant, then ρ is simply called a *renaming*. The *restriction* of a substitution θ to some V, denoted $\theta|_V$, is

the substitution θ' such that

$$\theta'(x) = \begin{cases} \theta(x), & \text{if } x \in V; \\ x, & \text{otherwise.} \end{cases}$$

Definition 2.6 The *union* of two substitutions σ and θ, denoted by $\sigma \cup \theta$, is defined by

$$\sigma \cup \theta(x) = \begin{cases} \sigma(x), & \text{if } x \in D(\sigma); \\ \theta(x), & \text{if } x \in D(\theta); \\ x, & \text{otherwise,} \end{cases}$$

and is only defined if $D(\sigma) \cap D(\theta) = \emptyset$. The *composition* of σ and θ is the substitution denoted by $\sigma \circ \theta$ such that for every variable x we have $\sigma \circ \theta(x) = \widehat{\theta}(\sigma(x))$. Given a set V of variables, we say that two substitutions σ and θ are *equal over* V, denoted $\sigma = \theta[V]$ iff $\forall x \in V,\ \sigma(x) = \theta(x)$. We say that σ is *more general than* θ *over* V, denoted by $\sigma \leq \theta[V]$, iff there exists a substitution η such that $\theta = \sigma \circ \eta[V]$. When V is the set of all variables, we drop the notation $[V]$.

A substitution σ is *idempotent* if $\sigma \circ \sigma = \sigma$. A necessary and sufficient condition for idempotency is given by

Lemma 2.7 A substitution σ is idempotent iff $I(\sigma) \cap D(\sigma) = \emptyset$.

Idempotent substitutions are easier to manipulate and the assumption of idempotency often simplifies a proof. That we may often restrict our attention to idempotent substitutions without loss of generality is formally justified by our next result, which shows that any substitution is equivalent (over an arbitrary superset of its support) up to renaming with an idempotent substitution.

Lemma 2.8 For any substitution σ and set of variables W such that $D(\sigma) \subseteq W$, there exists an idempotent substitution σ' such that $D(\sigma) = D(\sigma')$, $\sigma \leq \sigma'$, and $\sigma' \leq \sigma[W]$.

Proof. Let $D(\sigma) \cap I(\sigma) = \{x_1, \ldots, x_n\}$, let $\{y_1, \ldots, y_n\}$ be a set of *new* variables disjoint from W, $D(\sigma)$, and $I(\sigma)$, let $\rho_1 = [y_1/x_1, \ldots, y_n/x_n]$, and let $\rho_2 = [x_1/y_1, \ldots, x_n/y_n]$. Now let $\sigma' = \sigma \circ \rho_1$, where clearly $\sigma \leq \sigma'$ and $D(\sigma) = D(\sigma')$ as required. Since $\rho_1 \circ \rho_2 = Id[W \cup I(\sigma)]$, then $\sigma = \sigma \circ \rho_1 \circ \rho_2 = \sigma' \circ \rho_2[W]$, and thus $\sigma' \leq \sigma[W]$. Finally, by our previous lemma, σ' must be idempotent, since $D(\sigma') = D(\sigma)$ is disjoint from $I(\sigma') = (I(\sigma) - \{x_1, \ldots, x_n\}) \cup \{y_1, \ldots, y_n\}$. $\square$

Since most uses of substitutions in this paper are modulo renaming, this lemma will allow us to assume that substitutions are idempotent if necessary. We shall prove specific results related to the use of idempotent unifiers in later sections.

We now proceed to review the basic notions of relations, orderings, and equational rewriting.

Definition 2.9 Let $\Longrightarrow \subseteq A \times A$ be a binary relation on a set A. The *converse* (or *inverse*) of the relation $\Longrightarrow$ is the relation denoted as $\Longrightarrow^{-1}$ or $\Longleftarrow$, defined such that $u \Longleftarrow v$ iff $v \Longrightarrow u$. The symmetric closure of $\Longrightarrow$, denoted by $\Longleftrightarrow$, is the relation $\Longrightarrow \cup \Longleftarrow$. The transitive closure, reflexive and transitive closure, and the reflexive, symmetric, and transitive closure of $\Longrightarrow$ are denoted respectively by $\overset{+}{\Longrightarrow}$, $\overset{*}{\Longrightarrow}$, and $\overset{*}{\Longleftrightarrow}$.

Definition 2.10 A relation $\succ$ on a set A is *Noetherian* or *well founded* iff there are no infinite sequences $\langle a_0, \ldots, a_n, a_{n+1}, \ldots \rangle$ of elements in A such that $a_n \succ a_{n+1}$ for all $n \geq 0$.[1]

Definition 2.11 A *preorder* $\preceq$ on a set A is a binary relation $\preceq \subseteq A \times A$ that is reflexive and transitive. A *partial order* $\preceq$ on a set A is a preorder that is also antisymmetric. The converse of a preorder (or partial order) $\preceq$ is denoted as $\succeq$. A *strict ordering* (or *strict order*) $\prec$ on a set A is a transitive and irreflexive relation. Given a preorder (or partial order) $\preceq$ on a set A, the strict ordering $\prec$ associated with $\preceq$ is defined such that $s \prec t$ iff $s \preceq t$ and $t \not\preceq s$. Conversely, given a strict ordering $\prec$, the partial ordering $\preceq$ associated with $\prec$ is defined such that $s \preceq t$ iff $s \prec t$ or $s = t$. The converse of a strict ordering $\prec$ is denoted as $\succ$. Given a preorder (or partial order) $\preceq$, we say that $\preceq$ is well founded iff $\succ$ is well founded.[2]

Definition 2.12 Let $\longrightarrow$ be a binary relation $\longrightarrow \subseteq T_\Sigma(X) \times T_\Sigma(X)$ on terms. The relation $\longrightarrow$ is *monotonic* iff for every two terms s, t and every function symbol f, if $s \longrightarrow t$ then $f(\ldots, s, \ldots) \longrightarrow f(\ldots, t, \ldots)$. The relation $\longrightarrow$ is *stable* (under substitution) if $s \longrightarrow t$ implies $\sigma(s) \longrightarrow \sigma(t)$ for every substitution σ.

Definition 2.13 A strict ordering $\prec$ has the *subterm property* iff $s \prec f(\ldots, s, \ldots)$ for every term $f(\ldots, s, \ldots)$. A *simplification ordering* $\prec$ is a strict ordering that is monotonic and has the subterm property (since we are considering symbols having a fixed rank, the deletion property is superfluous, as noted in Dershowitz [6]). A *reduction ordering* $\prec$ is a strict ordering that is monotonic, stable, and such that $\succ$ is well founded. With a slight abuse of language, we will also say that the converse $\succ$ of a strict ordering $\prec$ is a simplification ordering (or a reduction ordering). It is shown in Dershowitz [6] that there are simplification orderings that are total on ground terms.

---

[1] We warn the readers that this is not the usual way of defining a well founded relation in set theory, as for example in Levy [29]. In set theory, the condition is stated in the form $a_{n+1} \prec a_n$ for all $n \geq 0$, where $\prec = \succ^{-1}$. It is the dual of the condition we have used, but since $\prec = \succ^{-1}$, the two definitions are equivalent. When using well founded relations in the context of rewriting systems, we are usually interested in the reduction relation $\Longrightarrow$ and the fact that there are no infinite sequences $\langle a_0, \ldots, a_n, a_{n+1}, \ldots \rangle$ such that $a_n \Longrightarrow a_{n+1}$ for all $n \geq 0$. Thus, following other authors, including Dershowitz, we adopt the dual of the standard set theoretic definition.

[2] Again, we caution our readers that in standard set theory it is $\prec$ that is well founded! However, our definition is equivalent to the standard set-theoretic definition of a well founded partial ordering.

Definition 2.14 Let $E \subseteq T_\Sigma(X) \times T_\Sigma(X)$ be a binary relation on terms. We define the relation $\longleftrightarrow_E$ over $T_\Sigma(X)$ as the smallest symmetric, stable, and monotonic relation that contains E. This relation is defined explicitly as follows: Given any two terms $t_1, t_2 \in T_\Sigma(X)$, then $t_1 \longleftrightarrow_E t_2$ iff there is some variant[3] (s,t) of a pair in $E \cup E^{-1}$, some tree address α in t_1, and some substitution σ, such that

$$t_1/\alpha = \sigma(s), \quad \text{and} \quad t_2 = t_1[\alpha \leftarrow \sigma(t)].$$

(In this case, we say that σ is a *matching substitution* of s onto t_1/α.) Note that the pair (s,t) is used as a two-way rewrite rule (that is, non-oriented). In such a case, we denote the pair (s,t) as $s \doteq t$ and call it an *equation*. When $t_1 \longleftrightarrow_E t_2$, we say that we have an *equality step*. It is well known that the reflexive and transitive closure $\overset{*}{\longleftrightarrow}_E$ of $\longleftrightarrow_E$ is the smallest stable congruence on $T_\Sigma(X)$ containing E. When we want to fully specify an equality step, we use the notation

$$t_1 \longleftrightarrow_{[\alpha, s \doteq t, \sigma]} t_2$$

(where some of the arguments may be omitted).

Definition 2.15 When a pair $(s,t) \in E$ is used as an oriented equation (from left to right), we call it a *rule* and denote it as $s \to t$. The *reduction relation* $\longrightarrow_E$ is the smallest stable and monotonic relation that contains E. We can define $t_1 \longrightarrow_E t_2$ explicitly as in definition 2.14, the only difference being that (s,t) is a variant of a pair in E (and not in $E \cup E^{-1}$). When $t_1 \longrightarrow_E t_2$, we say that t_1 *rewrites* to t_2, or that we have a *rewrite step*. When we want to fully specify a rewrite step, we use the notation

$$t_1 \longrightarrow_{[\alpha, s \to t, \sigma]} t_2$$

(where some of the arguments may be omitted).

When $Var(r) \subseteq Var(l)$, then a rule $l \to r$ is called a *rewrite rule*; a set of such rules is called a *rewrite system*.

In what follows, we shall usually for simplicity refer to both equality steps and rewrite steps by the generic term 'rewrite step' and similarly the term 'rewriting' will usually be used generically for the application of either rewrite steps or equality steps. The context should prevent any conflusion.

---

[3] In what follows we shall assume that before a pair (i.e., an equation) is used it has been renamed apart from all variables in current use. This is essential to prevent clashes among the variables. Thus we shall always state that a *variant* of an equation is being used.

Definition 2.16 Let $\longrightarrow \subseteq T_\Sigma(X) \times T_\Sigma(X)$ be a binary relation on $T_\Sigma(X)$. We say that $\longrightarrow$ is *Church Rosser* iff for all $t_1, t_2 \in T_\Sigma(X)$, if $t_1 \stackrel{*}{\longleftrightarrow}_E t_2$, then there is some $t_3 \in T_\Sigma(X)$ such that $t_1 \stackrel{*}{\longrightarrow} t_3$ and $t_2 \stackrel{*}{\longrightarrow} t_3$. We say that $\longrightarrow$ is *confluent* iff for all $t, t_1, t_2 \in T_\Sigma(X)$, if $t \stackrel{*}{\longrightarrow} t_1$ and $t \stackrel{*}{\longrightarrow} t_2$, then there is some $t_3 \in T_\Sigma(X)$ such that $t_1 \stackrel{*}{\longrightarrow} t_3$ and $t_2 \stackrel{*}{\longrightarrow} t_3$. A term s is *irreducible w.r.t.* $\longrightarrow$ iff there is no term t such that $s \longrightarrow t$.

It is well known that a relation is confluent iff it is Church Rosser [16]. We say that a rewrite system R is Noetherian, Church Rosser, or confluent, iff the relation $\longrightarrow_R$ associated with R given in definition 2.15 has the corresponding property. We say that R is *canonical* iff it is Noetherian and confluent.

Finally, before we proceed with the transformation method for the first-order case, we present the notion of a *multiset* and of the *multiset ordering*.

Definition 2.17 Given a set A, a *multiset* over A is an unordered collection of elements of A which may have multiple occurrences of identical elements. More formally, a multiset over A is a function $M : A \to \mathbf{N}$ (where $\mathbf{N}$ is the set of natural numbers) such that an element a in A has exactly n occurrences in M iff $M(a) = n$. In particular, a does not belong to M when $M(a) = 0$, and we say that $a \in M$ iff $M(a) > 0$. The *union* of two multisets M_1 and M_2, denoted by $M_1 \cup M_2$, is defined as the multiset M such that for all $a \in A$, $M(a) = M_1(a) + M_2(a)$.

To avoid confusion between multisets and sets, we shall always state carefully when an object is considered to be a multiset. Note that multiset union is a distinct notion from the union of sets, since for example, if A is a non-empty multiset, then $A \cup A \neq A$.

Definition 2.18 Let $<$ be a strict partial order on a set A, let M be some finite multiset on A, and finally let $n, n'_1, \ldots, n'_k \in A$. Define the relation $\Leftarrow_m$ on finite multisets as

$$M \cup \{n'_1, \ldots, n'_k\} \Leftarrow_m M \cup \{n\},$$

where $k \geq 0$ and $n'_i < n$ for all i, $1 \leq i \leq k$. Then the multiset ordering $\ll$ is simply the transitive closure $\stackrel{+}{\Leftarrow}_m$. In other words, $N' \ll N$ iff N' is produced from N by removing one or more elements and replacing them with any finite number of elements, each of which is strictly smaller than at least one element removed.

Lemma 2.19 Let $M(A)$ denote the set of all *finite* multisets on A, and let $<$ be a strict partial order on A. Then the multiset ordering $\ll$ is a strict partial ordering on $M(A)$ which is total (respectively, well-founded) iff $<$ is total (respectively, well-founded).

In this paper we use only the multiset ordering on multisets of natural numbers.

3 Unification by Transformations on Systems

We now define unification of terms and present an abstract view of the unification process as a set of non-deterministic rules for transforming a unification problem into an explicit representation of its solution, if such exists; in the next section this will be extended to E-unification. This elegant approach is due to [30], but was implicit in Herbrand's thesis [13].[4] Our representation for unification problems is the following.

Definition 3.1 A *term pair* or just a *pair* is a multiset of two terms, denoted, e.g., by $\langle s, t\rangle$, and a substitution θ is called a *standard unifier* (or just a *unifier*) of a pair $\langle s, t\rangle$ if $\theta(s) = \theta(t)$. A *term system* (or *system*) is a multiset of such pairs, and a substitution θ is a unifier of a system if it unifies each pair. The set of unifiers of a system S is denoted $U(S)$, and if S consists of only a single pair $\langle s, t\rangle$, the set of unifiers is denoted by $U(s, t)$.

Definition 3.2 A substitution σ is a *most general unifier*, or *mgu*, of a system S iff

(i) $D(\sigma) \subseteq Var(S)$;
(ii) $\sigma \in U(S)$;
(iii) For every $\theta \in U(S)$, $\sigma \leq \theta$.

It is well known that *mgu*'s always exist for unifiable systems, and it can be shown that *mgu*'s are unique up to composition with a renaming substitution, and so we shall follow the common practice of glossing over this distinction by referring to *the mgu* of a system, denoted by $mgu(S)$.

Definition 3.3 A pair $\langle x, t\rangle$ is in *solved form* in a system S and x in this pair is called a *solved variable* if x is a variable which does not occur anywhere else in S; in particular, $x \notin Var(t)$. A system is in solved form if all its pairs are in solved form; a variable is *unsolved* if it occurs in S but is not solved.

Note that a solved form system is always a *set* of solved pairs. The importance of solved form systems is shown by

Lemma 3.4 Let $S = \{\langle x_1, t_1\rangle, \ldots, \langle x_n, t_n\rangle\}$ be a system in solved form, where the $x_1, \ldots, x_n$ are solved variables. If $\sigma = [t_1/x_1, \ldots, t_n/x_n]$, then σ is an idempotent *mgu* of S. Furthermore, for any substitution $\theta \in U(S)$, we have $\theta = \sigma \circ \theta$.

Proof. We simply observe that for any θ, $\theta(x_i) = \theta(t_i) = \theta(\sigma(x_i))$ for $1 \leq i \leq n$, and

[4] It is remarkable that in his thesis, Herbrand gave all the steps of a (nondeterministic) unification algorithm based on transformations on systems of equations. These transformations are given at the end of the section on property A, page 148 of Herbrand [13].

$\theta(x) = \theta(\sigma(x))$ otherwise. Clearly σ is an mgu, and since $D(\sigma) \cap I(\sigma) = \emptyset$ by the definition of solved forms, it is idempotent. □

Strictly speaking the substitution σ here is ambiguous in the case that there is at least one pair in S consisting of two solved variables; but since mgu's are considered unique up to renaming, and such pairs can be arbitrarily renamed, we denote this substitution by σ_S. As a special case, note that $\sigma_\emptyset = Id$.

We may analyse the process of finding mgu's as follows. If $\theta(u) = \theta(v)$, then either (i) $u = v$ and no unification is necessary; or (ii) $u = f(u_1, \ldots, u_n)$ and $v = f(v_1, \ldots, v_n)$ for some $f \in \Sigma$, and $\theta(u_i) = \theta(v_i)$ for $1 \leq i \leq n$; or (iii) u is a variable not in $Var(v)$ or vice versa. If u is a variable and $u \notin Var(v)$, then $[v/u] \in U(u, v)$ and $[v/u] \leq \theta$. By extending this analysis to account for systems of pairs, we have a set of transformations for finding mgu's.

Definition 3.5 (The set of transformation rules $\mathcal{ST}$) Let S denote any system (possibly empty), $f \in \Sigma$, and u and v be two terms. We have the following transformations.

Trivial:

$$\{\langle u, u\rangle\} \cup S \implies_{\text{triv}} S$$

Term Decomposition: For any $f \in \Sigma_n$ for some $n > 0$,

$$\{\langle f(u_1, \ldots, u_n), f(v_1, \ldots, v_n)\rangle\} \cup S \implies_{\text{dec}} \{\langle u_1, v_1\rangle, \ldots, \langle u_n, v_n\rangle\} \cup S$$

Variable Elimination:

$$\{\langle x, v\rangle\} \cup S \implies_{\text{vel}} \{\langle x, v\rangle\} \cup \sigma(S),$$

where $\langle x, v\rangle$ is not a solved pair in S such that $x \notin Var(v)$, and $\sigma = [v/x]$.

Recall that systems are multisets, so the unions here are multiset unions; the intent of the left-hand side of each of these rules is to isolate a single pair to be transformed. The symbol $\implies$ will be used for an arbitrary transformation from the set $\mathcal{ST}$. We shall say that $\theta \in Unify(S)$ iff there exists some sequence of transformations

$$S \implies \ldots \implies S',$$

where S' is in solved form and $\theta = \sigma_{S'}$. (If no transformation applies, but the system is not in solved form, the procedure given here fails.)

Clearly, by choosing $S = \{\langle u, v\rangle\}$, we can attempt to find a unifier for two terms u, and v, as the following example shows.[5]

[5] In examples, we shall often drop set brackets around systems, e.g., $S = \langle x_1, t_1\rangle, \ldots, \langle x_n, t_n\rangle$.

Example 3.6

$$\begin{aligned}
&\langle f(x, g(a, y)),\ f(x, g(y, x))\rangle \\
&\Longrightarrow_{\text{dec}}\ \langle x, x\rangle,\ \langle g(a, y), g(y, x)\rangle \\
&\Longrightarrow_{\text{triv}}\ \langle g(a, y), g(y, x)\rangle \\
&\Longrightarrow_{\text{dec}}\ \langle a, y\rangle,\ \langle y, x\rangle \\
&\Longrightarrow_{\text{vel}}\ \langle a, y\rangle,\ \langle a, x\rangle\,.
\end{aligned}$$

The sense in which these transformations preserve the logically invariant properties of a unification problem is shown by

Lemma 3.7 If $S \Longrightarrow S'$ using any transformation from $\mathcal{ST}$, then $U(S) = U(S')$.

Proof. The only difficulty concerns Variable Elimination. Suppose $\{\langle x, v\rangle\} \cup S \Longrightarrow_{\text{vel}} \{\langle x, v\rangle\} \cup \sigma(S)$ with $\sigma = [v/x]$. For any substitution θ, if $\theta(x) = \theta(v)$, then $\theta = \sigma \circ \theta$, since $\sigma \circ \theta$ differs from θ only at x, but $\theta(x) = \theta(v) = \sigma \circ \theta(x)$. Thus,

$$\begin{aligned}
&\theta \in U(\{\langle x, v\rangle\} \cup S) \\
\text{iff}\ \ &\theta(x) = \theta(v)\ \text{ and }\ \theta \in U(S) \\
\text{iff}\ \ &\theta(x) = \theta(v)\ \text{ and }\ \sigma \circ \theta \in U(S) \\
\text{iff}\ \ &\theta(x) = \theta(v)\ \text{ and }\ \theta \in U(\sigma(S)) \\
\text{iff}\ \ &\theta \in U(\{\langle x, v\rangle\} \cup \sigma(S)).
\end{aligned}$$

□

The point here is that the most important feature of a unification problem—its set of solutions—is preserved under these transformations, and hence we are justified in our method of attempting to transform such problems into a trivial (solved) form in which the existence of an *mgu* is evident.

We may now show the soundness and completeness of these transformations following [30].

Theorem 3.8 (Soundness) If $S \overset{*}{\Longrightarrow} S'$ with S' in solved form, then $\sigma_{S'} \in U(S)$.

Proof. Using the previous lemma and a trivial induction on the length of transformation sequences, we see that $U(S) = U(S')$, and so clearly $\sigma_{S'} \in U(S)$. □

Theorem 3.9 (Completeness) Suppose that $\theta \in U(S)$. Then any sequence of transformations

$$S = S_0 \Longrightarrow S_1 \Longrightarrow S_2 \Longrightarrow \ldots$$

must eventually terminate in a solved form S' such that $\sigma_{S'} \leq \theta$.

Proof. We first show that every transformation sequence terminates. For any system S, let us define a complexity measure $\mu(S) =< n, m >$, where n is the number of *unsolved* variables in the system, and m is the sum of the sizes of all the terms in the system. Then the lexicographic ordering on $< n, m >$ is well-founded, and each transformation produces a new system with a measure strictly smaller under this ordering: Trivial and Term Decomposition must decrease m and can not increase n, and Variable Elimination must decrease n.

Therefore the relation $\Longrightarrow$ is well-founded, and every transformation sequence must end in some system to which no transformation applies. Suppose a given sequence ends in a system S'. Now $\theta \in U(S)$ implies by lemma 3.7 that $\theta \in U(S')$, and so S' can contain no pairs of the form $\langle f(t_1, \ldots, t_n), g(t'_1, \ldots, t'_m)\rangle$ or of the form $\langle x, t\rangle$ with $x \in Var(t)$. But since no transformation applies, all pairs in S' must be in solved form. Finally, since $\theta \in U(S')$, by lemma 3.4 we must have $\sigma_{S'} \leq \theta$. □

Putting these two theorems together, we have that the set $\mathcal{ST}$ can always find an *mgu* for a unifiable system of terms; as remarked in [30], this abstract formulation can be used to model many different unification algorithms, by simply specifying data structures and a control strategy.

In fact, we have proved something stronger than necessary in Theorem 3.9: it has been shown that all transformation sequences terminate and that *any* sequence of transformations issuing from a unifiable system must eventually result in a solved form. This is possible because the problem is decidable. Strictly speaking, it would have been sufficient for completeness to show that if S is unifiable then there exists *some* sequence of transformations which results in a solved form, since then a complete search strategy, such as breadth-first search, could find the solved form. This form of completeness, which might be termed *non-deterministic completeness*, will be used in finding results on E-unification, where the general problem is undecidable.

In some contexts it may be useful to deal with idempotent unifiers which are renamed away from some set of 'protected' variables but which are most general over the set of variables in the original system. The next definition makes this precise. (In the next section we shall offer a variation of this notion for E-unification.)

Definition 3.10 Given a system S and a finite set V of 'protected' variables, a substitution σ is a *most general unifier of S away from V* (abbreviated $mgu(S)[V]$) iff

(i) $D(\sigma) \subseteq Var(S)$ and $I(\sigma) \cap (V \cup D(\sigma)) = \emptyset$;

(ii) $\sigma \in U(S)$;

(iii) For every $\theta \in U(S)$, $\sigma \leq \theta[Var(S)]$.

That such substitutions may always be found for unifiable systems is shown by

Lemma 3.11 If S is a unifiable system and V a protected set of variables, then there exists a substitution σ which is a $mgu(S)[V]$.

Proof. Let $\theta = \text{Unify}(S)$, as in Definition 3.5, so that θ is an idempotent mgu of S such that $D(\theta) \cup I(\theta) \subseteq Var(S)$. If $V \cap I(\theta) = \emptyset$, then $\sigma = \theta$ is a $mgu(S)[V]$. Otherwise, let ρ be a renaming substitution away from $V \cup Var(S)$ such that $D(\rho) = I(\theta)$, and let $\sigma = \theta \circ \rho$. Clearly $D(\sigma) = D(\theta) \cup I(\theta) \subseteq Var(S)$. Since $I(\sigma) = I(\rho)$, by the definition of ρ, σ is idempotent and also $I(\sigma) \cap V = \emptyset$, and hence condition (i) is satisfied. Condition (ii) is satisfied also, since for any pair $\langle u, v \rangle$ in S, we have that $\theta(u) = \theta(v)$, and thus $\sigma(u) = \rho(\theta(u)) = \rho(\theta(v)) = \sigma(v)$, so that $\sigma \in U(S)$. To show the last condition, we first observe that from the definition of a renaming there must exist an inverse ρ^{-1} such that $\rho \circ \rho^{-1} = Id[I(\theta)]$ (since $I(\theta) = D(\rho)$). Now, for every $x \in D(\sigma)$, $\sigma(x) = \rho(\theta(x))$, and so $\rho^{-1}(\sigma(x)) = \rho \circ \rho^{-1}(\theta(x)) = \theta(x)$, with the result that $\theta = \sigma \circ \rho^{-1}[D(\sigma)]$. But since $D(\rho^{-1}) \cap Var(S) = \emptyset$, then also $\theta = \sigma \circ \rho^{-1}[Var(S)]$. Now suppose $\theta' \in U(S)$, so that $\theta' = \theta \circ \eta$ for some η. Then $\theta' = \sigma \circ \rho^{-1} \circ \eta[Var(S)]$ and finally $\sigma \leq \theta'[Var(S)]$. □

The following corollary will be used in a later result.

Corollary 3.12 If σ is a $mgu(S)[V]$ for some S and some V, then for every $\theta' \in U(S)$ we have $\sigma \leq \theta'[Var(S) \cup V]$.

Proof. By examining the details of the previous proof, we see that in fact $\theta = \sigma \circ \rho^{-1}[Var(S) \cup V]$, since $D(\rho^{-1}) \cap V = \emptyset$, and so $\theta' = \sigma \circ \rho^{-1} \circ \eta[Var(S) \cup V]$ and finally $\sigma \leq \theta'[Var(S) \cup V]$. □

4 *E*-Unification via Transformations

First we define the notion of E-unification and of a complete set of E-unifiers.

Definition 4.1 Let E be a finite set of equations. We say that a substitution θ is a unifier of a pair $\langle s, t \rangle$ *modulo E*, or an *E-unifier of s and t*, iff $\theta(s) \overset{*}{\longleftrightarrow}_E \theta(t)$. A substitution θ is

an E-unifier of a system S if it E-unifies every pair in S, and the set of all such E-unifiers will be denoted $U_E(S)$. If $S = \{\langle s, t\rangle\}$, then this will be denoted by $U_E(s, t)$.

It is well known that for any S the set $U_E(S)$ is only semi-decidable, and that even if a system is E-unifiable, there is in general no mgu unique up to renaming, but instead a possibly infinite set (see [7]). We now discuss some notions needed to deal with this more complex situation.

Definition 4.2 Given a finite set E of equations and any set V of variables, we say that two substitutions σ and θ are *equal modulo E over V*, denoted by $\sigma =_E \theta[V]$, iff $\forall x \in V$, $\sigma(x) \stackrel{*}{\longleftrightarrow}_E \theta(x)$. We say that σ is *more general modulo E than* θ *over V*, denoted by $\sigma \leq_E \theta[V]$, iff there exists some substitution η such that $\theta =_E \sigma \circ \eta[V]$. When V is the set of all variables, we drop the notation $[V]$, and similarly we drop the subscript E when $E = \emptyset$.

An important property of the relation $=_E$ which will be needed later is given by

Lemma 4.3 If $\theta =_E \sigma$ then for any system S, $\theta \in U_E(S)$ iff $\sigma \in U_E(S)$.

Proof. For any pair $\langle u, v\rangle$ in S, a simple induction on the structure of u and v suffices to show that $\theta(u) \stackrel{*}{\longleftrightarrow}_E \theta(v)$ iff $\sigma(u) \stackrel{*}{\longleftrightarrow}_E \sigma(v)$. □

From this lemma and the stability of E-congruence we can show

Corollary 4.4 If $\sigma \in U_E(S)$ and $\sigma \leq_E \theta[Var(S)]$ then $\theta \in U_E(S)$.

Note that this result is true in particular when $E = \emptyset$. Next we generalize the concept of a $mgu(S)[V]$ to E-unifiers; this formulation of a generating set for a set of E-unifiers is due to [34]; we present a modification of the definition from [7] for term systems.[6]

Definition 4.5 Given a finite set E of equations, a system S, and a finite set V of 'protected' variables, a set U of substitutions is a *complete set of E-unifiers for S away from V* (which we shall abbreviate by $CSU_E(S)[V]$) iff

(i) For all $\sigma \in U$, $D(\sigma) \subseteq Var(S)$ and $I(\sigma) \cap (V \cup D(\sigma)) = \emptyset$;

(ii) $U \subseteq U_E(S)$;

(iii) For every $\theta \in U_E(S)$, there exists some $\sigma \in U$ such that $\sigma \leq_E \theta[Var(S)]$.

The first condition is called the *purity condition*, the second the *coherence condition*, and the last the *completeness condition*. If S consists of a single pair $\langle u, v\rangle$ then we use the

[6] We also generalize slightly the Fages and Huet definition by allowing the protected set of variables to be arbitrary. The original definition imposed the restriction that $V \cap Var(S) = \emptyset$ in order that variable renaming not be necessary. We relax this restriction so that we have a true generalization of a $mgu(S)[V]$ to E-unifiers, and allow renaming to be imposed or not, by setting V appropriately.

abbreviation $CSU_E(u, v)[V]$. When the use of V is not relevant to our discussion we shall drop the notation $[V]$.

We now justify the purity condition and show the generality of idempotent E-unifiers.

Lemma 4.6 For any system S, substitution θ, and set of protected variables W, if $\theta \in U_E(S)$ then there exists some substitution σ such that

(i) $D(\sigma) \subseteq Var(S)$ and $I(\sigma) \cap (W \cup D(\sigma)) = \emptyset$;
(ii) $\sigma \in U_E(S)$;
(iii) $\sigma \leq \theta[Var(S)]$ and $\theta \leq \sigma[Var(S)]$.

Proof. If $\sigma = \theta|_{Var(S)}$ satisfies condition (i), then we have our result trivially. Otherwise, if $I(\theta) = \{x_1, \ldots, x_n\}$ then let $\{y_1, \ldots, y_n\}$ be a set of new variables disjoint from the variables in W, $D(\theta)$, $I(\theta)$, and $Var(S)$. Now define the renaming substitutions $\rho_1 = [y_1/x_1, \ldots, y_n/x_n]$ and $\rho_2 = [x_1/y_1, \ldots, x_n/y_n]$, and then let $\sigma = \theta \circ \rho_1|_{Var(S)}$. Clearly σ satisfies (i), and since $\sigma = \theta \circ \rho_1[Var(S)]$, we have the second part of (iii). Now since $\rho_1 \circ \rho_2 = Id[Var(S) \cup I(\theta)]$, we must have $\theta = \theta \circ \rho_1 \circ \rho_2[Var(S)]$. But then by the fact that $\sigma = \theta \circ \rho_1[Var(S)]$ we have $\theta = \sigma \circ \rho_2[Var(S)]$, proving the first part of (iii). To show (ii), observe that for any $\langle u, v\rangle \in S$ we have $\theta(u) \stackrel{*}{\longleftrightarrow}_E \theta(v)$, and so by the stability of E-congruence we have

$$\sigma(u) = \rho_1(\theta(u)) \stackrel{*}{\longleftrightarrow}_E \rho_1(\theta(v)) = \sigma(v),$$

which shows that $\sigma \in U_E(S)$. □

This proves that for any S and W, the set of all unifiers satisfying condition (i) and (ii) of Definition 4.5 is a $CSU(S)[W]$, and so in particular there is no loss of generality in considering only idempotent E-unifiers in what follows. This will simplify several of the definitions and proofs.

We now show how to extend the previous set of transformations to perform E-unification of a system under some arbitrary E, and develop the non-deterministic completeness of the method using a new formalism for 'proofs' that two terms are E-unifiable, known as *equational proof trees*. The new set of transformations is fully general in that it is capable of enumerating a $CSU_E(S)$ for any system S and set of equations E, and we intend this section to provide a paradigm for the abstract study of complete methods for general E-unification. The set of E-unifiers found by this method is highly redundant, however, and in the next section, we show how to restrict this method to avoid rewriting at variable occurrences while still retaining the ability to enumerate a $CSU_E(S)$.

We shall follow for the most part the plan of the previous section, in order to highlight the essential similarities and differences between standard unification and E-unification.

4.1 Transformations for E-Unification

First we examine the significance of solved form systems in this new context.

Lemma 4.7 If $S' = \{\langle x_1, t_1\rangle, \ldots, \langle x_n, t_n\rangle\}$ is a system in solved form, then $\{\sigma_{S'}\}$ is a $CSU_E(S')[V]$ for any V such that $V \cap Var(S') = \emptyset$.

Proof. The first two conditions in Definition 4.5 are satisfied, since $\sigma_{S'}$ is an idempotent mgu of S', $V \cap Var(S') = \emptyset$, and $I(\sigma_{S'}) \subseteq Var(S')$. Now, if $\theta \in U_E(S')$, then $\theta =_E \sigma_{S'} \circ \theta$, since $\theta(x_i) \stackrel{*}{\longleftrightarrow}_E \theta(t_i) = \theta(\sigma_{S'}(x_i))$ for $1 \leq i \leq n$, and $\theta(x) = \theta(\sigma_{S'}(x))$ otherwise. Thus $\sigma_{S'} \leq_E \theta$ and so obviously $\sigma_{S'} \leq_E \theta[Var(S')]$. □

This allows us to effectively ignore any E-unifiers which use rewrite steps between pairs in solved systems, if we are just interested in complete sets of unifiers.

We may analyse the process of finding a $CSU_E(u, v)$ for two terms u and v as follows. If $\theta \in U_E(u, v)$ then there must exist some sequence

$$\theta(u) = u_0 \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1]} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \rho_2]} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \rho_m]} u_m = \theta(v),$$

with m minimal (so that there are no redundant steps), $D(\rho_i) \subseteq Var(l_i, r_i)$ for $1 \leq i \leq m$. Since all the equations are variants, then we can assume that $D(\theta), D(\rho_1), \ldots, D(\rho_m)$ are pairwise disjoint, and we can form an *extended E-unifier* $\theta' = \theta \cup \rho_1 \cup \ldots \cup \rho_m$, so that we have

$$\theta'(u) = u_0 \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \theta']} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \theta']} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \theta']} u_m = \theta'(v).$$

Given any such rewrite sequence and extended E-unifier, we have several cases.

(1) $m = 0$ and $\theta' = \theta \in U(u, v)$. Then the analysis for standard unification is sufficient.

(2) $m \neq 0$ and some rewrite step occurs at the root of some u_i. Assume that if one of u, v is not a variable, it is u, and pick the left-most rewrite step; then

$$\theta'(u) \stackrel{*}{\longleftrightarrow}_E \theta'(l_i) \longleftrightarrow_{[\epsilon, l_i \doteq r_i, \theta']} \theta'(r_i) \stackrel{*}{\longleftrightarrow}_E \theta'(v),$$

for some i, $1 \leq i \leq m$, where there is no rewrite at the root between $\theta'(u)$ and $\theta'(l_i)$.

(3) $m \neq 0$ and no rewrite step occurs at the root of any u_i.

(a) $u = f(u_1, \ldots, u_n)$, $v = f(v_1, \ldots, v_n)$ for some $f \in \Sigma_n$ with $n > 0$, and therefore $\theta'(u_i) \stackrel{*}{\longleftrightarrow}_E \theta'(v_i)$ for $1 \leq i \leq n$.

(b) Either u or v is a variable; assume u is a variable.

(i) $v = f(v_1, \ldots, v_n)$ for some $f \in \Sigma_n$ with $n > 0$, $\theta'(u) = f(t_1, \ldots, t_n)$ for some terms $t_1, \ldots, t_n$, and thus $t_i \stackrel{*}{\longleftrightarrow}_E \theta'(v_i)$ for $1 \leq i \leq n$.

(ii) v is a variable. Then $\theta'(u) = f(t_1, \ldots, t_n)$ and $\theta'(v) = f(t'_1, \ldots, t'_n)$ for some terms $t_1, \ldots, t_n, t'_1, \ldots, t'_n$, where $t_i \overset{*}{\longleftrightarrow}_E t'_i$ for $1 \leq i \leq n$.

By recursively applying this analysis to the subsequences found in each case, every rewrite step in the original sequence can be accounted for. We use cases (2) and (3) to define two new transformation rules to account for the presence of rewrite steps in a unification problem.

Definition 4.8 (The set of transformation rules $\mathcal{BT}$) To the transformations $\mathcal{ST}$ we add two more to deal with equations.

Root Rewriting: Let $\langle u, v\rangle$ be a pair and if one of u or v is not a variable, assume that it is u. Then

$$\{\langle u, v\rangle\} \cup S \implies_{\text{rrw}} \{\langle u, l\rangle, \langle r, v\rangle\} \cup S,$$

where $l \doteq r$ is an alphabetic variant of an equation in $E \cup E^{-1}$ such that $Var(l, r) \cap (Var(S) \cup Var(u, v)) = \emptyset$, and if neither u nor l is a variable, then $Root(u) = Root(l)$. Root Rewriting may not be applied hereafter to the pair $\langle u, l\rangle$. This transformation represents a *leftmost* rewrite step at the root, and avoids rewriting a variable occurrence if possible.[7]

Root Imitation: If x is a variable and $f \in \Sigma_n$ with $n > 0$, then we have

$$\{\langle x, v\rangle\} \cup S \implies_{\text{imit}} \{\langle x, f(y_1, \ldots, y_n)\rangle, \langle x, v\rangle\} \cup S,$$

where the $y_1, \ldots, y_n$ are *new* variables and if v is not a variable, then $f = \text{Root}(v)$. Also, we immediately apply Variable Elimination to the new pair $\langle x, f(y_1, \ldots, y_n)\rangle$.

As in the transformations in $\mathcal{ST}$, recall that systems are multisets, and the unions above are *multiset unions*. Unless specified otherwise the symbol $\implies$ will be used in the rest of this section for an arbitrary transformation from the set $\mathcal{BT}$.

Thus, given a set of equations E and a system S to be E-unified, we say that $\theta \in E{-}\text{Unify}(S)$ iff there exists a sequence of transformations from the set $\mathcal{BT}$

$$S \implies S_1 \implies \ldots \implies S',$$

with S' in solved form and $\theta = \sigma_{S'}|_{Var(S)}$.

---

[7] Strictly speaking this transformation is something like a paramodulation step at the root, except that the terms u and l are not unified. The point is that the juxtaposition of an equation between the terms u and v imitates the way a rewrite step occurs in the proof that two terms E-unify, and is not just paramodulation, since further rewrites can take place below the root of u and l.

Example 4.9 Let $E = \{ f(g(z)) \doteq z \}$ and $S = \{ \langle h(x), h(g(f(x))) \rangle \}$. Then we have the following sequence of transformations:

$$\begin{aligned}
\langle h(x), h(g(f(x))) \rangle &\Longrightarrow_{\text{dec}} \langle x, g(f(x)) \rangle \\
&\Longrightarrow_{\text{imit,vel}} \langle x, g(y_1) \rangle, \langle g(y_1), g(f(g(y_1))) \rangle \\
&\Longrightarrow_{\text{dec}} \langle x, g(y_1) \rangle, \langle y_1, f(g(y_1)) \rangle \\
&\Longrightarrow_{\text{rrw}} \langle x, g(y_1) \rangle, \langle y_1, z' \rangle, \langle f(g(z')), f(g(y_1)) \rangle \\
&\Longrightarrow_{\text{vel}} \langle x, g(y_1) \rangle, \langle y_1, z' \rangle, \langle f(g(y_1)), f(g(y_1)) \rangle \\
&\Longrightarrow_{\text{triv}} \langle x, g(y_1) \rangle, \langle y_1, z' \rangle
\end{aligned}$$

Therefore, $[g(y_1)/x] = \theta \in E{-}\text{Unify}(S)$ is an E-unifier of $h(x)$ and $h(g(f(x)))$, as shown by the rewrite sequence

$$\theta(h(x)) = h(g(y_1)) \longleftrightarrow_{[11, z' \doteq f(g(z')), y_1/z']} h(g(f(g(y_1)))) = \theta(h(g(f(x)))).$$

The general idea here is that given some $\theta \in U_E(S)$, we wish to show that it is always possible to find some $\sigma \in U_E(S)$ such that $\sigma \leq_E \theta[Var(S)]$; in particular, this will be accomplished if we can find a substitution $\sigma \in U_E(S)$ such that $\theta =_E \sigma \circ \theta[Var(S)]$. The basic method of the transformations is to find solved pairs $\langle x, t \rangle$ such that $\theta(x) \overset{*}{\longleftrightarrow}_E \theta(t)$, so that, by an argument similar to that used in lemma 3.4, we have $\theta =_E [t/x] \circ \theta$. The sequence of solved pairs found may be thought of as 'pieces' of the substitution θ, and the set of solved pairs collected constitute successive approximations of the substitution θ, namely, $\sigma_1 = [t_1/x_1], \sigma_2 = [t_1/x_1] \circ [t_2/x_2], \ldots$. When we have approximated θ sufficiently to E-unify the system, we may stop. Along the way, we shall also build up the various matching substitutions as we solve for variables from the variants of equations inserted by Root Rewriting. This is the reason for restricting the substitution extracted from the final solved form to just those variables occurring in the original system S.

In this context, Root Imitation represents a 'minimal approximation' of a substitution. This corresponds to case 3.b in our previous analysis of E-unification, where some rewrite steps occur, but *not* at the root, and one of the terms is a variable. We assume u is some variable x, and then either (i) v is a compound term $f(v_1, \ldots, v_n)$, where $n \neq 0$, or (ii) v is a variable. In case (i), we know that $\theta'(u) = f(t_1, \ldots, t_n)$ for some terms $t_1, \ldots, t_n$, and $t_i \overset{*}{\longleftrightarrow}_E \theta'(v_i)$ for $1 \leq i \leq n$. But we can not yet tell the exact identity of the terms $t_1, \ldots, t_n$; we know only that $Root(\theta(x)) = f$. Thus we assume that $\theta'(x) = f(y_1, \ldots, y_n)$, where the new variables $y_1, \ldots, y_n$ are *placeholders* for the rest of the binding, and will be found at some later point. Such a binding for x may be called a *general binding for* x. We

may roughly think of this as extending the substitution $\theta' = [f(t_1, \ldots, t_n)/x] \cup \theta''$ into a substitution

$$\widehat{\theta'} = [f(y_1, \ldots, y_n)/x] \circ [t_1/y_1, \ldots, t_n/y_n] \cup \theta'',$$

where clearly $\widehat{\theta'} = \theta'[D(\theta')]$. By solving the pair $\langle x, f(y_1, \ldots, y_n)\rangle$, we have found a piece of this extended substitution. The bindings for the new variables will be found later and substituted in using Variable Elimination. In case (ii), where both u and v are variables, we know that $\theta'(u) = f(t_1, \ldots, t_n)$ and $\theta'(v) = f(t'_1, \ldots, t'_n)$ for some terms $t_1, \ldots, t_n, t'_1, \ldots, t'_n$ where $t_i \overset{*}{\longleftrightarrow}_E t'_i$ for $1 \leq i \leq n$. In this case we 'guess' a general binding for u, and then this case is reduced to the previous one. Thus we must guess the root symbol of the binding; this 'don't know' non-determinism clearly presents implementation problems, but for the present we are only concerned with demonstrating the completeness of a very general set of transformations; in §6 we show how this can be avoided.

One interesting special case where Root Imitation is applicable is in E-unifying a pair of the form $\langle x, t\rangle$, where $x \in Var(t)$, i.e., when the *occur check* fails for x. Although such a pair cannot have a mgu, it is potentially E-unifiable by rewriting at the root (e.g., $[a/x] \in U_E(x, f(x))$ for $E = \{a \doteq f(a)\}$) or by rewriting below the root, as shown in Example 4.9 for the pair $\langle x, g(f(x))\rangle$. To E-unify a pair $\langle x, f(v_1, \ldots, v_n)\rangle$ where the occur check fails for x and no rewrite occurs at the root, we simulate rewriting below the root by the use of Root Imitation and Term Decomposition, imitating the root f with a general binding for x, and decomposing, thus distributing the occur check into at least one of the pairs $\langle y_1, v_1\rangle, \ldots, \langle y_n, v_n\rangle$, whereupon we may apply Root Rewriting or Root Imitation again to that pair. At some point we must find an application of Root Rewriting if we are to eliminate the occur check. Unfortunately, it is possible to create an infinite series of pairs isomorphic up to renaming by repeatedly applying Root Imitation and Term Decomposition:

$$\begin{aligned}\langle x, f(x)\rangle &\Longrightarrow_{\text{imit,vel,dec}} \langle x, f(y_1)\rangle, \langle y_1, f(y_1)\rangle \\ &\Longrightarrow_{\text{imit,vel,dec}} \langle x, f(f(y_2))\rangle, \langle y_1, f(y_2)\rangle, \langle y_2, f(y_2)\rangle \ldots .\end{aligned}$$

Obviously this problem can not arise unless the occur check fails. In §6 we shall show that we can eliminate such redundant sequences without affecting the completeness of the procedure.

4.2 Soundness of the Transformations

The following lemmas will be used to show that our procedure is sound. The first is a straightforward adaptation of lemma 3.7.

Lemma 4.10 If $S \Longrightarrow S'$ using Trivial or Variable Elimination, then $U_E(S) = U_E(S')$.

Proof. As with standard unification, the only difficulty is with Variable Elimination. We must show that $U_E(\{\langle x, v\rangle\} \cup S) = U_E(\{\langle x, v\rangle\} \cup \sigma(S))$ where $\sigma = [v/x]$ and $x \notin Var(v)$. For any substitution θ, if $\theta(x) \stackrel{*}{\longleftrightarrow}_E \theta(v)$, then $\theta =_E \sigma \circ \theta$, since $\sigma \circ \theta$ differs from θ only at x, but $\theta(x) \stackrel{*}{\longleftrightarrow}_E \theta(v) = \sigma \circ \theta(x)$. Thus,

$$\begin{aligned}
&\theta \in U_E(\{\langle x, v\rangle\} \cup S) \\
\text{iff } &\theta(x) \stackrel{*}{\longleftrightarrow}_E \theta(v) \text{ and } \theta \in U_E(S) \\
\text{iff } &\theta(x) \stackrel{*}{\longleftrightarrow}_E \theta(v) \text{ and } \sigma \circ \theta \in U_E(S); \text{ by lemma 4.3} \\
\text{iff } &\theta(x) \stackrel{*}{\longleftrightarrow}_E \theta(v) \text{ and } \theta \in U_E(\sigma(S)) \\
\text{iff } &\theta \in U_E(\{\langle x, v\rangle\} \cup \sigma(S)).
\end{aligned}$$

□

Lemma 4.11 If $S \Longrightarrow S'$ using one of Term Decomposition, Root Rewriting, or Root Imitation, then $U_E(S') \subseteq U_E(S)$.

Proof. The basic idea here is that these transformations do not preserve those E-unifiers which require a rewrite step or an application of root imitation, but do not introduce the possibility of new E-unifiers. There are three cases.

(*i*) *Term Decomposition*: If we have $\theta(s_i) \stackrel{*}{\longleftrightarrow}_E \theta(t_i)$, for $1 \leq i \leq n$, then $\theta\big(f(s_1, \ldots, s_n)\big) \stackrel{*}{\longleftrightarrow}_E \theta\big(f(t_1, \ldots, t_n)\big)$, so clearly $S \Longrightarrow_{\text{dec}} S'$ and $\theta \in U_E(S')$ implies that $\theta \in U_E(S)$.

(*ii*) *Root Rewriting*: If $\theta(u) \stackrel{*}{\longleftrightarrow}_E \theta(l)$, $\theta(r) \stackrel{*}{\longleftrightarrow}_E \theta(v)$ for some variant $l \doteq r$ of an equation from $E \cup E^{-1}$, then

$$\theta(u) \stackrel{*}{\longleftrightarrow}_E \theta(l) \longleftrightarrow_{[\epsilon, l \doteq r, \theta]} \theta(r) \stackrel{*}{\longleftrightarrow}_E \theta(v).$$

Thus $S \Longrightarrow_{\text{rrw}} S'$ and $\theta \in U_E(S')$ implies that $\theta \in U_E(S)$.

(*iii*) *Root Imitation*: This rule is in two parts. First we add a pair $\langle x, f(y_1, \ldots, y_n)\rangle$ to the system, and then we apply Variable Elimination. Since we showed the soundness of Variable Elimination, we simply observe that if $S \Longrightarrow_{\text{imit}} S'$ then $S \subseteq S'$, so clearly $\theta \in U_E(S')$ implies that $\theta \in U_E(S)$.

In the case of Root Rewriting, the inclusion is always proper if the equation is not ground, since E-unifiers of the new system must account for the variables in the equation used in the rewrite step. The inclusion is also proper with Root Imitation, since new variables are introduced again. □

Using these lemmas, we have the major result of this subsection.

Theorem 4.12 (Soundness) If $S \stackrel{*}{\Longrightarrow} S'$, with S' in solved form, then $\sigma_{S'}|_{Var(S)} \in U_E(S)$.

Proof. Using the previous two lemmas and a trivial induction on the length of transformation sequences, we have that $\sigma_{S'} \in U_E(S)$. But since the restriction has no effect as regards the terms in S, we must have also that $\sigma_{S'}|_{Var(S)} \in U_E(S)$. □

4.3 Completeness

It is a testament to the power and elegance of the technique of unification by transforming systems of terms that it can be adapted to E-unification by adding only two additional transformations, and that this method, as we prove in this section, can non-deterministically find a $CSU_E(S)$ for *arbitrary* E and S.

In order to prove the completeness of the set $\mathcal{BT}$, we must show that if $\theta \in U_E(S)$, then there exists some sequence of transformations resulting in a solved form S' such that $\sigma_{S'} \leq_E \theta[Var(S)]$. The strategy we adopt is to take a representation for the fact that $\theta \in U_E(S)$, and let its structure determine the sequence of transformations. In particular, we shall proceed as follows. First, we observe that for any system $S = \{\langle u_1, v_1\rangle, \ldots, \langle u_n, v_n\rangle\}$ there must exist sequences of rewrite steps $\theta(u_1) \stackrel{*}{\longleftrightarrow}_E \theta(v_1), \ldots, \theta(u_n) \stackrel{*}{\longleftrightarrow}_E \theta(v_n)$ proving that $\theta \in U_E(S)$, and we form an E-unifier θ' similar to the extension of θ as defined above in section §4.1. Then we define an extension $\widehat{\theta'}$ of θ' and a system of pairs $B_{\theta'}$ which account for all the potential uses of general bindings by Root Imitation used in building up parts of the substitution θ'. The next step is to show how, for every sequence of rewrite steps $\theta(u_i) \stackrel{*}{\longleftrightarrow}_E \theta(v_i)$ there corresponds an *equational proof tree* which represents the sequence of rewrite steps in a more convenient form, and then define a *proof system* $< \widehat{\theta'}, B_{\theta'}, P >$, where P is a set of equational proof trees corresponding to all the pairs in S. This proof system is essentially a 'preprocessing' of the original θ, S, and the sequences of rewrite steps showing that $\theta \in U_E(S)$, in which all the syntactic materials possibly used by the transformation rules have been collected together in a fashion which makes the completeness of the set $\mathcal{BT}$ more evident. We then define a set of proof transformation rules analogous to the set of transformations for systems which decompose the set of proof trees to a trivial form; this sequence of proof transformations corresponds in a natural way to a sequence of transformations on systems of pairs which, when applied to the original system S, finds a system S' in solved form such that $\sigma_{S'} \leq_E \theta[Var(S)]$. This is the essence of the method of proving non-deterministic completeness: we show that for any $\theta \in U_E(S)$, with E and S arbitrary, there always exists *some* sequence of transformations which finds a E-unifier more general than θ.

We showed in section §4.1 how for any $\theta \in U_E(S)$, there corresponds a set of rewrite sequences and an extension θ' of θ incorporating all the matching substitutions. We

provide a more rigorous formulation of this as follows. We need one preliminary lemma.

Lemma 4.13 If

$$u = u_0 \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1]} u_1 \ldots \longleftrightarrow_{[\alpha_n, l_n \doteq r_n, \rho_n]} u_n = v,$$

for some sequence of equations from $E \cup E^{-1}$, then for any σ we have

$$\sigma(u_0) \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1 \circ \sigma]} \sigma(u_1) \ldots \longleftrightarrow_{[\alpha_n, l_n \doteq r_n, \rho_n \circ \sigma]} \sigma(u_n). \qquad (*)$$

Proof. We proceed by induction on n. If $n = 0$ then the result holds trivially. Now assume the hypothesis for all such sequences of length less than n for $n > 0$. For a sequence of length n we have

$$\sigma(u_0) \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1 \circ \sigma]} \sigma(u_1) \ldots \longleftrightarrow_{[\alpha_{n-1}, l_{n-1} \doteq r_{n-1}, \rho_{n-1} \circ \sigma]} \sigma(u_{n-1})$$

and $u_{n-1} \longleftrightarrow_{[\alpha_n, l_n \doteq r_n, \rho_n]} u_n$, that is, $u_{n-1}/\alpha_n = \rho_n(l_n)$ and $u_n = u_{n-1}[\alpha_n \leftarrow \rho_n(r_n)]$. But then, since $\alpha_n \in Dom(u_{n-1})$ we have $\sigma(u_{n-1})/\alpha_n = \sigma(u_{n-1}/\alpha_n) = \sigma(\rho_n(l_n))$ and

$$\sigma(u_n) = \sigma(u_{n-1}[\alpha_n \leftarrow \rho_n(r_n)]) = \sigma(u_{n-1})[\alpha_n \leftarrow \sigma(\rho_n(r_n))],$$

and so therefore $\sigma(u_{n-1}) \longleftrightarrow_{[\alpha_n, l_n \doteq r_n, \rho_n \circ \sigma]} \sigma(u_n)$, from which $(*)$ follows. □

Lemma 4.14 For any system $S = \{\langle u_1, v_1\rangle, \ldots, \langle u_n, v_n\rangle\}$, if $\theta \in U_E(S)$ then there exists some idempotent $\theta' \in U_E(S)$ such that $\theta' \leq \theta[Var(S)]$ and some set of rewrite sequences $R = \{\Pi_1, \ldots, \Pi_n\}$ proving[8] that θ' E-unifies each pair in S, where each such sequence has the form

$$\theta'(u) = u_0 \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \theta']} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \theta']} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \theta']} u_m = \theta'(v). \qquad (1)$$

Proof. Let $\{\rho_1, \ldots, \rho_m\}$ be the set of all matching substitutions used in all the n rewrite sequences in R; as in the beginning of section §4.1 we may create an extension incorporating all the matching substitutions used in a rewrite sequence, since all occurrences of equations in all rewrite sequences are assumed to be renamed away from each other and from $Var(S)$. Thus, let $\theta'' = \theta \cup \rho_1 \cup \ldots \cup \rho_m$, so that we have

$$\theta''(u) = u_0 \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \theta'']} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \theta'']} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \theta'']} u_m = \theta''(v). \quad (2)$$

[8] R is a set of *specific* sequences of rewrite steps, denoted by Π_i; see Definition 5.1.

Now, because all equations in R are variants, we have $\theta'' = \theta[Var(S)]$. If θ'' is not idempotent then there exists by lemma 2.8 a renaming substitution ρ' and an idempotent $\theta' = \theta'' \circ \rho'$ such that $\theta' \leq \theta''[W]$ where W is the set of all variables in S, in the set of variants of equations used in R, and in $D(\theta')$. Clearly we have $\theta' \leq \theta'' = \theta[Var(S)]$, and finally, by our preceeding lemma, we may apply the substitution ρ' to the entire sequence (2) to obtain the sequence (1). □

Let us assume in what follows that such a set of rewrite sequences and such a θ' is fixed. We now proceed to define the set $B_{\theta'}$ and the extension $\widehat{\theta'}$ which account for the general bindings used by root imitation.

Definition 4.15 For a given substitution θ', let us define a *general expansion of* θ', denoted $\widehat{\theta'}$, and the corresponding *system of general bindings for* θ', denoted $B_{\theta'}$, as follows. For each $x \in D(\theta')$, let $\theta'_x = \theta'|_{\{x\}}$. For each such θ'_x, define inductively the substitution $\widehat{\theta'_x}$ and the set $B_{\theta'_x}$ as follows. If $\theta'_x = [t/x]$ with $|t| = 0$, i.e., t is either a constant or a variable, then let $\widehat{\theta'_x} = \theta'_x$ and $B_{\theta'_x} = \emptyset$. Otherwise, if $\theta'_x = [f(t_1, \ldots, t_n)/x]$, then for some new variables $y_1, \ldots, y_n$, let $\theta'_{y_i} = [t_i/y_i]$ for $1 \leq i \leq n$, let

$$\widehat{\theta'_x} = \theta'_x \cup \widehat{\theta'_{y_1}} \cup \ldots \cup \widehat{\theta'_{y_n}},$$

and let

$$B_{\theta'_x} = \{\langle x, f(y_1, \ldots, y_n)\rangle\} \cup B_{\theta'_{y_1}} \cup \ldots \cup B_{\theta'_{y_n}}.$$

Finally, let $\widehat{\theta'} = \bigcup_{x \in D(\theta')} \widehat{\theta'_x}$ and $B_{\theta'} = \bigcup_{x \in D(\theta')} B_{\theta'_x}$.

For example, if $\theta' = [g(f(a), b)/x, z/y]$, then

$$\widehat{\theta'} = [g(f(a), b)/x, f(a)/y_1, a/y_2, b/y_3, z/y],$$

and

$$B_{\theta'} = \{\langle x, g(y_1, y_3)\rangle, \langle y_1, f(y_2)\rangle\}.$$

The following lemma demonstrates the essential properties of $\widehat{\theta'}$ and and $B_{\theta'}$ needed in our completeness proof.

Lemma 4.16 For any substitution $\theta' \in U_E(S)$ for some S, there exists some $\widehat{\theta'}$ and $B_{\theta'}$ such that

(i) $\widehat{\theta'}$ and $B_{\theta'}$ are unique up to the choice of new variables in $D(\widehat{\theta'}) - D(\theta')$;
(ii) θ' is idempotent iff $\widehat{\theta'}$ is idempotent;
(iii) $\widehat{\theta'} = \theta'[D(\theta') \cup Var(S)]$, with the result that $\widehat{\theta'} \in U_E(S)$;
(iv) $\widehat{\theta'} \in U(B_{\theta'})$.

Proof. By a simple induction on $|t|$ we can show that $\widehat{\theta'}_x$ exists for any $\theta' = [t/x]$, and so clearly $\widehat{\theta'}$ and $B_{\theta'}$ exist, and since the only place in the construction for non-uniqueness is in picking the new variables, the result is always unique up to this choice, showing (i). By an induction which follows the construction of $\widehat{\theta'}$ we can show that $I(\widehat{\theta'}) = I(\theta')$ and $D(\widehat{\theta'}) = D(\theta') \cup Y$, where Y is the set of new variables chosen. Now, since Y consists of new variables, we must have $Y \cap I(\widehat{\theta'}) = \emptyset$, so that $D(\widehat{\theta'}) \cap I(\widehat{\theta'}) = \emptyset$ iff $D(\theta') \cap I(\theta') = \emptyset$. But then by lemma 2.7, we have (ii). Again, as a consequence of the set Y being new variables, (iii) must hold. Finally, note that by our definition, for any single binding t/x in $\widehat{\theta'}$, either $|t| = 0$ or t is some compound term $f(t_1, \ldots, t_n)$ such that there exists a pair $\langle x, f(y_1, \ldots, y_n)\rangle$ in $B_{\theta'}$ and some bindings $t_1/y_1, \ldots, t_n/y_n$ in $\widehat{\theta'}$. Thus by a simple induction on the construction of $B_{\theta'}$ we see that (iv) holds. □

The idea here is that we wish to preprocess the substitution θ' in order to determine the set of general bindings which *might* be used in a transformation by Root Imitation. Thus we determine in advance the set of pairs potentially introduced by Root Imitation and also the extensions to the substitution which 'fill in' these general bindings.

Now we define our formalism for the fact that such a substitution E-unifies a pair of terms.

Definition 4.17 Let θ' be some idempotent substitution, and let $\widehat{\theta'}$ and $B_{\theta'}$ be as above. The set of *proof trees associated with* $\widehat{\theta'}$ is defined inductively as follows. For simplicity we use $*$ as a syntactic variable for one of the symbols $\approx$, $\sim$, or $=$.

(i) (Axioms) For every term u, the one node tree labeled with $u = u$ is a proof tree associated with $\widehat{\theta'}$. For every two terms $u \neq v$, at least one of which is a variable and the other a constant or a variable, such that $\widehat{\theta'}(u) = \widehat{\theta'}(v)$, the one node tree labeled with $u = v$ is a proof tree associated with $\widehat{\theta'}$. Thus, axioms are trivial proofs that identical terms are E-unifiable or that a variable in the domain of the substitution associated with the proof trivially E-unifies with some term. Note that in the latter case, the axiom will be formed from two terms x and t, where $x \notin Var(t)$, and that it is not necessary that $\widehat{\theta'}(x) = t$.

(ii) (Term Decomposition) Let u and v be a pair of terms, $f \in \Sigma_n$, and $u_1, \ldots, u_n$, $v_1, \ldots, v_n$ be terms such that

(a) If u is a variable, then $\widehat{\theta'}(u) = f(u_1, \ldots, u_n)$, otherwise $u = f(u_1, \ldots, u_n)$, and

(b) If v is a variable, then $\widehat{\theta'}(v) = f(v_1, \ldots, v_n)$, otherwise $v = f(v_1, \ldots, v_n)$.

Given any n proof trees $T_1, \ldots, T_n$ associated with $\widehat{\theta'}$, where each T_i is a proof tree whose root is labeled with $u_i * v_i$, the tree T whose root is labeled with $u \sim v$ and such that $T/i = T_i$ for $1 \leq i \leq n$ is a proof tree associated with $\widehat{\theta'}$. Thus, a proof tree whose root is labeled with $u \sim v$ represents the fact that $\widehat{\theta'}(u) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$, where no rewrite steps occur

at the root. Note that if either of the terms u or v is a variable, then we must instantiate it before decomposing it in the proof tree; if a term is compound it is simply decomposed, without the substitution being applied.

(iii) (Root Rewriting) Let u and v be a pair of terms and $l_i \doteq r_i$ for $1 \leq i \leq m$ be variants of equations from $E \cup E^{-1}$. Furthermore, let $T_1, \ldots, T_{m+1}$ be proof trees associated with $\widehat{\theta'}$, where T_1 is a proof tree whose root is labeled with either $u = l_1$ or $u \sim l_1$, and for $2 \leq i \leq m$, T_i is a proof tree whose root is labeled with either $r_{i-1} = l_i$ or $r_{i-1} \sim l_i$, and T_{m+1} is a proof tree whose root is labeled with either $r_m = v$ or $r_m \sim v$. Then the tree T whose root is labeled with $u \approx v$ and such that $T/i = T_i$ for $1 \leq i \leq m+1$ is a proof tree associated with $\widehat{\theta'}$. This shows the effect of all the rewrites occurring at the root in $\widehat{\theta'}(u) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$.

In general, we regard the nodes of a proof tree as *unordered* pairs of terms, in accordance with the unordered nature of term pairs. A proof tree associated with $\widehat{\theta'}$ whose root is labeled with $u * v$ will be denoted by the pair $\langle \widehat{\theta'}, (u * v) \rangle$, or simply $(u * v)$ if the substitution is available from context.[9] It should be obvious that with any set of proof trees P we may associate a system of pairs S, namely, the set of pairs of terms occurring in the roots of the proof trees in the set P; this is called the *root system* of P.

Finally, a triple $\langle \widehat{\theta'}, B_{\theta'}, P \rangle$ is a *proof system for* θ *and* S if θ' is an idempotent substitution such that $\theta' \leq \theta[Var(S)]$, $\widehat{\theta'}$ is the general expansion of θ', $B_{\theta'}$ is the set of general bindings for θ', and finally if P is a set of proof trees associated with $\widehat{\theta'}$ with a root system S. (The point here is that although θ' must be idempotent, θ need not be.) Note that as a consequence of these definitions, for each subproof $(x \sim v)$ occurring somewhere in a proof in P, there exists some pair $\langle x, t \rangle$ in $B_{\theta'}$; this corresponds to the pair possibly added to the system by some application of Root Imitation to the pair $\langle x, v \rangle$.

We shall prove that these proof systems are sound and complete with respect to the definition of E-unification after presenting an illustration based on a variation of Example 4.9.

Example 4.18 Let $E = \{ f(g(z)) \doteq z \}$. The rewrite sequence which proves that $\theta = [g(a)/x]$ is an E-unifier of $S = \{\langle h(x), h(g(f(x))) \rangle\}$ is

$$\theta(h(x)) = h(g(a)) \longleftrightarrow_{[11, z' \doteq f(g(z')), a/z']} h(g(f(g(a)))) = \theta(h(g(f(x)))),$$

and so we may form the E-unifier $\theta' = [g(a)/x, a/z']$ and then the general expansion $\widehat{\theta'} = [g(a)/x, a/y_1, a/z']$ and the set of general bindings $B_{\theta'} = \{\langle x, g(y_1) \rangle\}$. The proof

[9] Note carefully that $u * v$ is the *label* of a proof tree node, and $(u * v)$ is a proof tree whose root node is labeled with $u * v$.

system for θ and S is thus $\langle\widehat{\theta'}, B_{\theta'}, P\rangle$, where P is the set consisting of the single proof tree

$$h(x) \sim h(g(f(x)))$$

$$x \sim g(f(x))$$

$$a \approx f(x)$$

$$a = z' \qquad\qquad f(g(z')) \sim f(x)$$

$$g(z') \sim x$$

$$z' = a$$

The root system of P is $\{\langle h(x), h(g(f(x)))\rangle\}$. (Compare with Example 4.9.)

When convenient, we shall represent the (partial) structure of a proof tree with root node $u * v$ and subtrees $P_1, \ldots, P_n$ in the prefix form $u * v[P_1, \ldots, P_n]$, e.g., variously representing the subtree with root node $a \approx f(x)$ above in any of the forms

$$\big(a \approx f(x)\big), \qquad a \approx f(x)\big[a = z', (f(g(z')) \sim f(x))\big],$$

or

$$a \approx f(x)\big[a = z', f(g(z')) \sim f(x)[g(z') \sim x[z' = a]]\big].$$

This linear notation will make it somewhat easier to manipulate proof trees.

Our next two theorems show that our proof representation is sound and complete with respect to the definition of E-unification.

Theorem 4.19 For some given substitution θ, system S, and set of equations E, if $\langle\widehat{\theta'}, B_{\theta'}, P\rangle$ is a proof system for θ and S, then $\theta \in U_E(S)$.

Proof. By the previous definition and lemma 4.16, we have $\widehat{\theta'} = \theta' \leq \theta[Var(S)]$, and so if we can show that for each proof tree $(u * v)$ in P, we have $\widehat{\theta'}(u) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$, then by Corollary 4.4 we shall have our result. Thus let $T = (u*v)$ be an arbitrary proof tree in P. We proceed by induction on the number n of tree nodes in T. If $n = 1$, then $\widehat{\theta'}(u) = \widehat{\theta'}(v)$ by definition. Now assume that the result holds for all proof trees with less than n nodes, with $n > 1$, and suppose T contains n nodes. There are two cases.

(i) If the root node of T is labeled with $u \sim v$, then as above we suppose f is the root of $\widehat{\theta'}(u)$ and let $u_1, \ldots, u_m, v_1, \ldots, v_m$ be terms such that

(a) If u is a variable, then $\widehat{\theta'}(u) = f(u_1, \ldots, u_m)$, otherwise $u = f(u_1, \ldots, u_m)$,

(b) If v is a variable, then $\widehat{\theta'}(v) = f(v_1, \ldots, v_m)$, otherwise $v = f(v_1, \ldots, v_m)$.

There are thus proof trees

$$T/1 = (u_1 * v_1), \ \ldots, \ T/m = (u_m * v_m)$$

and by the hypothesis, $\widehat{\theta'}(u_i) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v_i)$ for $1 \leq i \leq m$. By changing the rewrite addresses $\alpha_1, \alpha_2, \ldots$ in the i^{th} such sequence to $i\alpha_1, i\alpha_2, \ldots$, and concatenating these m new rewrite sequences, we see that $\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$. (Note how the idempotency of $\widehat{\theta'}$ is used here.)

(ii) If the root node of T is labeled with $u \approx v$ then there are proof trees

$$T/1 = (u * l_1), \ T/2 = (r_1 * l_2), \ \ldots, \ T/k+1 = (r_k * v),$$

where the $l_i \doteq r_i$ are variants of equations from $E \cup E^{-1}$, and, by hypothesis,

$$\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(l_1), \ \ldots, \ \widehat{\theta'}(r_k) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v),$$

and so

$$\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(l_1) \ \longleftrightarrow_{[\epsilon, l_1 \doteq r_1, \widehat{\theta'}]} \ \widehat{\theta'}(r_1) \overset{*}{\longleftrightarrow}_E \ \ldots \ \widehat{\theta'}(r_k) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v),$$

with the result that again $\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$. □

Theorem 4.20 If $\theta \in U_E(S)$, then there exists a proof system $\langle \widehat{\theta'}, B_{\theta'}, P \rangle$ associated with θ and S.

Proof. As shown in lemma 4.14, if $\theta \in U_E(S)$ then there must exist some particular sequence of rewrites proving this fact, and an idempotent E-unifier θ' incorporating all the matching substitutions used in rewrite steps. Then by lemma 4.16 we know that $\widehat{\theta'}$ and $B_{\theta'}$ must exist, so if we can show that for any $\langle u, v \rangle \in S$ there exists an equational proof tree $(u * v)$ associated with $\widehat{\theta'}$, then we can simply collect all these trees together to form P and we have our result.

Thus we shall show by induction that for any particular sequence

$$\widehat{\theta'}(u) = u_0 \ \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \widehat{\theta'}]} \ u_1 \ \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \widehat{\theta'}]} \ \ldots u_{n-1} \ \longleftrightarrow_{[\alpha_n, l_n \doteq r_n, \widehat{\theta'}]} \ u_n = \widehat{\theta'}(v),$$

we have a proof tree $(u * v)$ associated with $\widehat{\theta'}$. With any such rewrite sequence, we associate a complexity measure

$$\mu = \{|u_0|, |u_1|, \ldots, |u_n|\},$$

that is, a multiset of the depths of the terms $u_0, \ldots, u_n$. Our proof proceeds by induction on μ, using the standard multiset ordering.

Basis. $\mu = \{k\}$ and either $u = v$ or one of u, v is a variable. Then by Definition 4.17 $(u = v)$ is a proof tree associated with $\widehat{\theta'}$. (This constitutes a sufficient basis since it includes the case $k = 0$.)

Induction. Assume there exists a corresponding proof tree for all such rewrite sequences with complexity strictly less than μ, and consider a sequence with complexity μ, as above. There are three cases.

(*i*) $\mu = \{k\}$ where $u \neq v$ and neither of u, v is a variable. Now we must have $Root(u) = Root(v)$, and since $u \neq v$, both are compound terms, i.e., $k > 0$. Thus $\widehat{\theta'}(u) = u_0 = \widehat{\theta'}(v)$ and $u = f(s_1, \ldots, s_m)$ and $v = f(t_1, \ldots, t_m)$ for some terms $s_1, \ldots, s_m, t_1, \ldots, t_m$. Then $\widehat{\theta'}(s_i) = u_0/i = \widehat{\theta'}(t_i)$ with $|u_0/i| < |u_0|$ for $1 \leq i \leq m$, and by hypothesis, there are proof trees $(s_1 * t_1), \ldots, (s_m * t_m)$ associated with $\widehat{\theta'}$, and so by definition there must exist a proof tree $u \sim v[(s_1 * t_1), \ldots, (s_m * t_m)]$ associated with $\widehat{\theta'}$. (This proof tree will naturally contain no rewrite nodes.)

(*ii*) $\mu = \{k_0, k_1, \ldots, k_n\}$ for $n > 0$, and there is no rewrite at the root of any u_i. In this case, $Root(\widehat{\theta'}(u)) = Root(\widehat{\theta'}(v))$, and the subterms are pairwise E-congruent. More precisely, let $f = Root(\widehat{\theta'}(u))$ be of arity m, and $s_1, \ldots, s_m, t_1, \ldots, t_m$ be terms such that

(a) If u is a variable, then $\widehat{\theta'}(u) = f(s_1, \ldots, s_m)$, otherwise $u = f(s_1, \ldots, s_m)$, and

(b) If v is a variable, then $\widehat{\theta'}(v) = f(t_1, \ldots, t_m)$, otherwise $v = f(t_1, \ldots, t_m)$.

Then for each $1 \leq i \leq m$ we have that

$$\widehat{\theta'}(s_i) = u_0/i \longleftrightarrow_E u_1/i \longleftrightarrow_E \ldots \longleftrightarrow_E u_n/i = \widehat{\theta'}(t_i),$$

with a complexity strictly less than μ. By the induction hypothesis, there exist proof trees $(s_1 * t_1), \ldots, (s_m * t_m)$ associated with $\widehat{\theta'}$, and thus by definition a proof tree

$$u \sim v[(s_1 * t_1), \ldots, (s_m * t_m)]$$

associated with $\widehat{\theta'}$. (Note that the idempotency of $\widehat{\theta'}$ is necessary in case one of u, v is a variable.)

(*iii*) $\mu = \{k_0, k_1, \ldots, k_n\}$ for $n > 0$, and there is a rewrite at the root of some u_i. Then we may represent the sequence as

$$\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(l'_1) \longleftrightarrow_{[\epsilon, l'_1 \doteq r'_1, \widehat{\theta'}]} \widehat{\theta'}(r'_1) \overset{*}{\longleftrightarrow}_E \ldots \widehat{\theta'}(l'_p) \longleftrightarrow_{[\epsilon, l'_p \doteq r'_p, \widehat{\theta'}]} \widehat{\theta'}(r'_p) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$$

for some subset $\{l'_1 \doteq r'_1, \ldots, l'_p \doteq r'_p\}$ of the equations used in the original sequence. But then the complexity of each of the sequences

$$\widehat{\theta'}(u) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(l'_1), \quad \widehat{\theta'}(r'_1) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(l'_2), \quad \ldots, \quad \widehat{\theta'}(r'_p) \overset{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$$

is strictly less than μ, and by hypothesis, there are proof trees $(u*l'_1), (r'_1*l'_2), \ldots, (r'_p*v)$ associated with $\widehat{\theta'}$. Finally, by definition there must exist a proof tree

$$u \approx v[(u*l'_1), \ldots, (r'_p*v)]$$

associated with $\widehat{\theta'}$. $\square$

One interesting point about this completeness proof is that it gives us a canonical way of constructing a proof tree for any particular sequence of rewrite steps proving that two terms are E-unifiable by the substitution $\widehat{\theta'}$. This is particularly useful in eliminating variables by applying substitutions to proof trees.

Lemma 4.21 If x is a variable, t a term, and $\widehat{\theta'}$ an idempotent general expansion such that $\widehat{\theta'}(x) = \widehat{\theta'}(t)$, and if u and v are two arbitrary terms, then there exists a proof tree $(u*v)$ associated with $\widehat{\theta'}$ iff there exists a proof tree $(u[t/x]*v[t/x])$ associated with $\widehat{\theta'}$. Furthermore, if such proof trees exist, there always exist two with the same number of $\approx$-nodes.

Proof. Since $\widehat{\theta'}(x) = \widehat{\theta'}(t)$ we must have $\widehat{\theta'} = [t/x] \circ \widehat{\theta'}$, so that by lemma 4.3 we have $\widehat{\theta'}(u) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(v)$ iff $[t/x] \circ \widehat{\theta'}(u) \stackrel{*}{\longleftrightarrow}_E [t/x] \circ \widehat{\theta'}(v)$ iff $\widehat{\theta'}(u[t/x]) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(v[t/x])$, and so, by our previous two results, there exists a proof tree $(u*v)$ associated with $\widehat{\theta'}$ iff there exists a proof tree $(u[t/x]*v[t/x])$ associated with $\widehat{\theta'}$. Now by structural induction, it is easy to show that for any *particular sequence* of m rewrite steps we have

$$\widehat{\theta'}(u[t/x]) \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1]} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \rho_2]} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \rho_m]} \widehat{\theta'}(v[t/x])$$

if and only if

$$\widehat{\theta'}(u) \longleftrightarrow_{[\alpha_1, l_1 \doteq r_1, \rho_1]} u_1 \longleftrightarrow_{[\alpha_2, l_2 \doteq r_2, \rho_2]} u_2 \ldots \longleftrightarrow_{[\alpha_m, l_m \doteq r_m, \rho_m]} \widehat{\theta'}(v).$$

But then by multiset induction on this sequence, following the proof of Theorem 4.20, it is easy to show that if such terms are E-congruent using this particular sequence, then proof trees exist for each pair, and that the creation of $\approx$-nodes corresponds to the structure of this particular sequence, and hence the number of such nodes is the same in both trees. $\square$

We remark that, depending on the set E, there may exist many equivalent sequences of rewrite steps, so that we can not enforce that the number of $\approx$-nodes *always* be the same for any two trees; we simply prove that there always exist two such similar trees. Also, note that it would be possible to be more precise about the structural similarity of trees created canonically from the same rewrite sequence, in the sense that their $\approx$-nodes occur in the same tree addresses, but this formality is unnecessary for our purposes, so we omit

it. Finally, we remark that it would not in general be possible to define a similar lemma for the case of two terms x and t such that $\widehat{\theta'}(x) \stackrel{*}{\longleftrightarrow}_E \widehat{\theta'}(t)$ without extending the substitution $\widehat{\theta'}$. The reason is that we can not use the same rewrite sequence $\stackrel{*}{\longleftrightarrow}_E$ in both cases, since there may be more rewrite steps in one than the other, and since the rewrites between $\widehat{\theta'}(x)$ and $\widehat{\theta'}(t)$ may be used many times, by our assumption that all rewrite sequences contain distinct variants of equations, these would be additional instances of equations, and the extension θ' would no longer be sufficient. This problem turns out to have serious consequences in proving the completeness of the strategy of eager variable elimination (see Section §10).

Now we show that the transformations on systems $\mathcal{BT}$ correspond to a certain set of transformations on proof systems.

Definition 4.22 Let P' be a set of proof trees (possibly empty). We have the following five proof transformations.

$$\{(u * u)\} \cup P' \Longrightarrow P' \tag{A}$$

$$\{u \sim v[T_1, \ldots, T_n]\} \cup P' \Longrightarrow \{T_1, \ldots, T_n\} \cup P', \tag{B}$$

where u and v are both compound terms.

$$\{(x * t)\} \cup P' \Longrightarrow \{(x = t)\} \cup P'[t/x], \tag{C}$$

where there are no $\approx$-nodes in the tree $(x * t)$ (i.e., no rewrite steps), x occurs in some tree in P' and where $P'[t/x]$ denotes the result of replacing each proof tree $(u * v)$ in P by a proof tree $(u[t/x] * v[t/x])$ (the existence of such a proof tree was shown in the previous lemma).

$$\{u \approx v[T_1, \ldots, T_n]\} \cup P' \Longrightarrow \{T_1, \ldots, T_n\} \cup P' \tag{D}$$

$$\{(x \sim v)\} \cup P' \Longrightarrow \{(x = t), (x \sim v)\} \cup P', \tag{E}$$

where $\langle x, t\rangle \in B_{\theta'}$ and where transformation (C) is immediately applied to the axiom $(x = t)$.

These proof transformations are extended from trees to systems, so that we say $\langle \widehat{\theta'}, B_{\theta'}, P\rangle \Longrightarrow \langle \widehat{\theta'}, B_{\theta'}, P'\rangle$ iff $P \Longrightarrow P'$.

It should be obvious that we have taken pains to define these proof transformations by analogy with our transformations on term systems. In particular, for some proof trees

P and P' with root systems S and S' respectively, if $P \Longrightarrow P'$ using proof transformations (A), (B), (C), or (E), then there is a corresponding transformation on the root system $S \Longrightarrow S'$ using Trivial , Term Decomposition, Variable Elimination, or Root Imitation respectively. Similarly, if $P \Longrightarrow_{(D)} P'$, then we have a sequence $S \stackrel{*}{\Longrightarrow}_{\mathrm{rrw}} S'$, with one transformation step for each rewrite step left to right in the proof tree transformed in P.

Now we may prove the correctness of these proof transformations, after which we shall give an example of their use.

Lemma 4.23 If $\langle \widehat{\theta'}, B_{\theta'}, P \rangle$ is a proof system and $P \Longrightarrow P'$ using one of the transformations (A)–(E), then $\langle \widehat{\theta'}, B_{\theta'}, P' \rangle$ is a proof system.

Proof. Clearly, the only point at issue is whether the new set P' is a set of proof trees associated with $\widehat{\theta'}$. In case (A), P' differs from P only in having one less proof tree, so clearly if P is a set of proof trees associated with $\widehat{\theta'}$, so is P'. In the case of transformations (B) and (D), since proof trees were defined inductively, for any proof tree T associated with $\widehat{\theta'}$, where T is not an axiom, the subtrees $T/1, \ldots, T/n$ for some n must still be proof trees associated with $\widehat{\theta'}$, and thus the result of either of these transformations must still be a set of proof trees associated with $\widehat{\theta'}$. If $P \Longrightarrow_{(C)} P'$, then since no rewrites occur in $(x * t)$, we must have $\widehat{\theta'}(x) = \widehat{\theta'}(t)$, and so $(x = t)$ is a proof tree associated with $\widehat{\theta'}$, and by lemma 4.21, there exists a proof tree $(u[t/x] * v[t/x])$ associated with $\widehat{\theta'}$. Finally, if $P \Longrightarrow_{(E)} P'$, then we have simply converted a pair $\langle x, t \rangle$ from $B_{\theta'}$ into a proof tree $(x = t)$, and since, by lemma 4.16, $\widehat{\theta'}(x) = \widehat{\theta'}(t)$, this is an axiom tree associated with $\widehat{\theta'}$. But then $\{(x = t)\} \cup P$ is a set of proof trees associated with $\widehat{\theta'}$, and we have already shown that the subsequent application of (C) is correct. □

Example 4.24 The transformations on the single proof tree in the proof system from Example 4.18 corresponding to the transformations in Example 4.9 are as follows.

$$h(x) \sim h(g(f(x)))$$

$$x \sim g(f(x))$$

$$a \approx f(x)$$

$$a = z' \qquad\qquad f(g(z')) \sim f(x)$$

$$g(z') \sim x$$

$$z' = a$$

$$\Downarrow_{(B)}$$

$$x \sim g(f(x))$$

$$a \approx f(x)$$

$$a = z' \qquad\qquad f(g(z')) \sim f(x)$$

$$g(z') \sim x$$

$$z' = a$$

$$\Downarrow_{(E)}$$

$$x = g(y_1) \qquad\qquad g(y_1) \sim g(f(g(y_1)))$$

$$y_1 \approx f(g(y_1))$$

$$y_1 = z' \qquad\qquad f(g(z')) \sim f(g(y_1))$$

$$g(z') \sim g(y_1)$$

$$z' = y_1$$

$$\Downarrow_{(B)}$$

$$x = g(y_1) \qquad y_1 \approx f(g(y_1))$$

$$y_1 = z' \qquad f(g(z')) \sim f(g(y_1))$$

$$g(z') \sim g(y_1)$$

$$z' = y_1$$

$$\Downarrow_{(D)}$$

$$x = g(y_1) \quad y_1 = z' \qquad f(g(z')) \sim f(g(y_1))$$

$$g(z') \sim g(y_1)$$

$$z' = y_1$$

$$\Downarrow_{(C)}$$

$$x = g(y_1) \quad y_1 = z' \qquad f(g(y_1)) = f(g(y_1))$$

$$\Downarrow_{(A)}$$

$$x = g(y_1) \quad y_1 = z'$$

Note that this corresponds to the solved form system $S' = \{\langle x, g(y_1)\rangle, \langle y_1, z'\rangle\}$ found in Example 4.9, and that for $\theta = [g(a)/x]$ as in Example 4.18 we have $\sigma_{S'} \leq \theta[Var(S)]$. Our next result formalizes this by showing that the proof transformations always result in trivial proofs corresponding to solved form systems.

Lemma 4.25 Let $\langle \widehat{\theta'}, B_{\theta'}, P\rangle$ be a proof system. Then any sequence of proof transformations

$$P = P_0 \Longrightarrow P_1 \Longrightarrow \ldots$$

must terminate in a system $P' = \{(x_1 = t_1), \ldots, (x_n = t_n)\}$ associated with $\widehat{\theta'}$ where no transformation applies, and the root system of P' is a system in solved form.

Proof. First we show that every sequence of proof transformations must terminate. Let us define a measure of complexity for a set P of proof trees as $\mu(P) = \langle n, m\rangle$, where n is the number of variables in $D(\widehat{\theta'})$ which are not solved in the root system of P, and m is the number of nodes in all the proof trees in P. Then the lexicographic ordering on $\langle n, m\rangle$ is well-founded, and each proof transformation produces a new proof system whose measure is strictly smaller under this ordering: (A), (B), and (D) must decrease m and can not increase n; and (C) and (E) must decrease n.

Therefore the relation $\Longrightarrow$ on proof systems is well-founded, and there must exist some sequence $P \stackrel{*}{\Longrightarrow} P'$ where no transformation applies to P'. But then P' must consist solely of axioms of the form $(x_i = t_i)$ with x_i not identical with t_i, since otherwise either (A), (B), (D), or (E) would apply, no x_i occurs in a t_i, since the two are unifiable, and furthermore each variable x_i may not occur elsewhere in the proof system, or else (C) would apply. Clearly the root system $\{\langle x_1, t_1\rangle, \ldots, \langle x_n, t_n\rangle\}$ is a system in solved form.

By a simple induction on the length of the proof transformation sequence, and using lemma 4.23 in the induction step, we see that P' is a proof system associated with $\widehat{\theta'}$. □

Now we are ready to state the major result of this section. The completeness of our method is shown in the following theorem.

Theorem 4.26 (Completeness) For every $\theta \in U_E(S)$, there exists a sequence of transformations $S \stackrel{*}{\Longrightarrow} S'$ such that S' is in solved form, and $\sigma_{S'} \leq \theta[Var(S)]$.

Proof. Suppose $\theta \in U_E(S)$. Then by Theorem 4.20 there must exist an equational proof system $\langle\widehat{\theta'}, B_{\theta'}, P\rangle$, where by lemmas 4.14 and 4.16, we have $\widehat{\theta'} = \theta' \leq \theta[Var(S)]$. By lemma 4.25 we see that there must exist some sequence of proof transformations $P \stackrel{*}{\Longrightarrow} P'$ with $P' = \{(x_1 = t_1), \ldots, (x_k = t_k)\}$ a set of proof trees associated with $\widehat{\theta'}$ to which no transformation applies, and whose root system S' is a system in solved form. By a simple induction on the length of the proof transformation sequence, we may show that there is a corresponding sequence of transformations on the root system $S \stackrel{*}{\Longrightarrow} S'$ with $S' = \{\langle x_1, t_1\rangle, \ldots, \langle x_k, t_k\rangle\}$ in solved form, and since P' is a set of proof trees associated with $\widehat{\theta'}$, we have $\widehat{\theta'} \in U(S')$, so that by lemma 3.4 we see that $\sigma_{S'} \leq \widehat{\theta'}$, with the result that $\sigma_{S'} \leq \widehat{\theta'} = \theta' \leq \theta[Var(S)]$. □

By the soundness of the transformations, clearly any such $\sigma_{S'} \in U_E(S)$. Note that this theorem implies that $\sigma_{S'} \leq_E \theta[Var(S)]$, but is in fact a stronger result. The reason that we find more general substitutions under $\leq$ and not just $\leq_E$ is that we only perform a generalization step at the last stage, when we take the mgu of a solved form.

Finally, we may characterize the set of substitutions non-deterministically found by the set of transformations $\mathcal{BT}$ as follows.

Theorem 4.27 For any system S and any set of equations E, the set

$$\{\sigma_{S'}|_{Var(S)} \mid S \stackrel{*}{\Longrightarrow} S', \text{ and } S' \text{ is in solved form}\}$$

is a $CSU_E(S)$. By application of the appropriate renaming substitution away from V, this set is a $CSU_E(S)[V]$ for any V.

Proof. We must simply verify the conditions in Definition 4.5. Coherence was shown in Theorem 4.12 and our previous result demonstrated completeness. By restricting the idempotent substitution $\sigma_{S'}$ to $Var(S)$ we satisfy purity for V empty. If V is not empty, we may suitably rename the variables introduced by each of the substitutions $\sigma_{S'}$ away from V, as shown in lemma 3.11. □

Using these results, it would be possible to implement a general procedure for E-unification in arbitrary theories by using a complete search strategy over all possible transformation sequences. In [11], a pseudo-code procedure based on Robinson's original algorithm for standard unification [36] is given for a different set of transformations for E-unification, using depth-first iterative deepening to simulate breadth-first search without excessive storage overhead. However, basing such a method on the set $\mathcal{BT}$ would be very inefficient, due to the possibility of rewriting variables in Root Rewriting. This creates many extraneous rewrite sequences, since *any* rule can unify with a variable. In addition, we must guess general bindings in the two variable case in Root Imitation to uncover potential rewrites below such pairs, and, finally, we admit the potential for infinite recursion in the same rule, as remarked in section §4.1. In the following sections we present a new set of transformations which rectify this problem, and a proof of its completeness.

5 Ground Church-Rosser Systems

In this section, we shall develop techniques that will allow us to overcome the problem of possible nonterminating sequences of applications of Root Imitation. The key point is that if the equations in E were orientable and formed a canonical system R, then we could work with normalized substitutions, that is, substitutions such that $\theta(x)$ is irreducible for every $x \in D(\theta)$. If R is canonical, for every pair $\langle x, v\rangle$ where x is a variable, there is a proof of the form $\theta(v) \stackrel{*}{\longrightarrow}_R w \stackrel{*}{\longleftarrow}_R \theta(x)$ for some irreducible w, and if θ is normalized, then the proof is in fact of the form $\theta(v) \stackrel{*}{\longrightarrow}_R \theta(x)$, where every rule $\rho(l) \to \rho(r)$ used in this sequence applies at some *nonvariable* address β in v. Hence, for any rule in this sequence applied at

a topmost level, $\theta(v/\beta)$ and $\rho(l)$ must be E-congruent. This is the motivation for a new rule, called *Lazy Paramodulation*, to replace Root Rewriting and Root Imitation:

$$\{\langle u, v\rangle\} \cup S \Longrightarrow \{\langle u/\beta, l\rangle, \langle u[\beta \leftarrow r], v\rangle\} \cup S,$$

where β is a nonvariable occurrence in u. A formal definition of this transformation will be given in section 6, and the set of transformations $\mathcal{T}$ obtained by adding this new rule to $\mathcal{ST}$ will be given in definition 6.1.

However, not every set of equations is equivalent to a canonical system of rewrite rules, and even if it is orientable with respect to some reduction ordering (thus forming a noetherian set of rules), it may not be confluent. Three crucial observations allow us to overcome these difficulties:

(1) There is no loss of generality in considering only ground substitutions;

(2) There are simplification orderings $\succ$ that are total on ground terms;

(3) Ground confluence (or equivalently, being ground Church-Rosser) is all that is needed.

These ingredients make possible the existence of *unfailing completion* procedures (Bachmair, Dershowitz, Hsiang, and Plaisted [1,2,3]). The main trick is that one can use *orientable ground instances of equations*, that is, ground equations of the form $\rho(l) \doteq \rho(r)$ with $\rho(l) \succ \rho(r)$, where $l \doteq r$ is a variant of an equation in $E \cup E^{-1}$. Even if $l \doteq r$ is not orientable, $\rho(l) \doteq \rho(r)$ always is if $\succ$ is total on ground terms. The last ingredient is that given a set E of equations and a reduction ordering $\succ$ total on ground terms, we can show that E can be extended to a set E^ω equivalent to E such that the set $R(E^\omega)$ of orientable instances of E^ω is ground Church-Rosser. Furthermore, E^ω is obtained from E by computing critical pairs (in a hereditary fashion), treating the equations in E as two-way rules.[10]

Our "plan of attack" for the completeness proof of the new set of transformations $\mathcal{T}$ (given in definition 6.1) is the following.

(1) Show the existence of the ground Church-Rosser completion E^ω of E (theorem 5.7).

(2) Assuming that E is ground Church-Rosser, show that the $\mathcal{T}$-transformations are complete by examining closely the completeness proof in the basic case discussed in the previous section.

---

[10] Although a consequence of the existence of fair unfailing completion procedures proved by Bachmair, Dershowitz, Hsiang, and Plaisted [1,2,3], this result can be proved more directly and with less machinery.

(3) For an arbitrary E, show that the $\mathcal{T}$-transformations are complete using theorem 5.7 and a lemma which shows that the computation of critical pairs can be simulated by Lazy Paramodulation.

In (2), we shall also show that given any E-unifier θ, there is another normalized E-unifier σ such that $\sigma =_E \theta$.

It is actually more general (and more flexible) but no more complicated to deal with pairs (E, R) where E is a set of equations and R a set of rewrite rules contained in some given reduction ordering $\succ$. The set E represents the nonorientable part (w.r.t. $\succ$) of the system. Thus, as in Bachmair, Dershowitz, Hsiang, and Plaisted [1,2,3], we present our results for such systems. First, we generalize the notion of equational proof. Given a set E of equations and a rewrite system R, we define the notion of proof and rewrite proof for the pair (E, R).

Definition 5.1 Let E be a set of equations and R a rewrite system. For any two terms u, v, a *proof step* from u to v is a tuple $\langle u, \alpha, l, r, \sigma, v\rangle$, where α is a tree address in u, σ is a substitution, and either

$$u \longleftrightarrow_{[\alpha, l \dot{=} r, \sigma]} v$$

where $l \doteq r$ is a variant of an equation in $E \cup E^{-1}$, or

$$u \longrightarrow_{[\alpha, l \to r, \sigma]} v$$

where $l \to r$ is a variant of a rewrite rule in R, or

$$v \longrightarrow_{[\alpha, l \to r, \sigma]} u$$

where $l \to r$ is a variant of a rewrite rule in R.

A proof step may be (partially) described as either an *equality step* $u \longleftrightarrow_E v$, or a *rewrite step* $u \longrightarrow_R v$ or $u \longleftarrow_R v$. A *proof* that $u \overset{*}{\longleftrightarrow}_{E \cup R} v$ is a sequence

$$\langle\langle u_0, \alpha_1, l_1, r_1, \sigma_1, u_1\rangle, \langle u_1, \alpha_2, l_2, r_2, \sigma_2, u_2\rangle, \ldots, \langle u_{n-1}, \alpha_n, l_n, r_n, \sigma_n, u_n\rangle\rangle$$

obtained by concatenating proof steps, with $u = u_0$ and $v = u_n$. It is obvious that proofs can be concatenated. A proof consisting only of rewrite steps involving rules in R used from left to right is denoted as $u_0 \longrightarrow_R u_1 \ldots u_{n-1} \longrightarrow_R u_n$ or $u_0 \overset{*}{\longrightarrow}_R u_n$. A proof consisting only of rewrite steps involving rules in R used from right to left is denoted as $u_0 \longleftarrow_R u_1 \ldots u_{n-1} \longleftarrow_R u_n$ or $u_0 \overset{*}{\longleftarrow}_R u_n$. A proof of the form $u \overset{*}{\longrightarrow}_R w \overset{*}{\longleftarrow}_R v$ is called a *rewrite proof*. A proof of the form $u \longleftarrow_R w \longrightarrow_R v$ is called a *peak*. Clearly, a proof is a rewrite proof iff it is a proof without peaks.

We also need the concepts of orientable instance, ground Church-Rosser, and critical pair.

Definition 5.2 Let E be a set of equations and $\succ$ a reduction ordering. Given a variant $l \doteq r$ of an equation in $E \cup E^{-1}$, an equation $\sigma(l) \doteq \sigma(r)$ is an *orientable instance* (w.r.t. $\succ$) of $l \doteq r$ iff $\sigma(l) \succ \sigma(r)$ for some substitution σ.[11] Given a reduction ordering $\succ$, the set of all orientable instances of equations in $E \cup E^{-1}$ is denoted by $R(E)$. Note that if $u \longrightarrow_{R(E)} v$, then $u \longrightarrow_{[\alpha, \sigma(l) \doteq \sigma(r)]} v$ for some variant of an equation $l \doteq r$ in $E \cup E^{-1}$ such that $\sigma(l) \succ \sigma(r)$, and since $\succ$ is a reduction ordering, $u \succ v$.

Definition 5.3 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering. The pair (E, R) is *ground Church-Rosser relative to* $\succ$ iff (a) $R \subseteq \succ$ and (b) for any two *ground* terms u, v, if $u \overset{*}{\longleftrightarrow}_{E \cup R} v$, then there is a rewrite proof $u \overset{*}{\longrightarrow}_{R(E) \cup R} w \overset{*}{\longleftarrow}_{R(E) \cup R} v$ for some w. A reduction ordering $\succ$ is *total on E-equivalent ground terms* iff for any two distinct ground terms u, v, if $u \overset{*}{\longleftrightarrow}_{E} v$, then either $u \succ v$ or $v \succ u$. A reduction ordering $\succ$ that is total on E-equivalent ground terms is called a *ground reduction ordering for E*.

It is important to note that for every set R of rewrite rules which is noetherian with respect to a given reduction ordering $\succ$, if R is Church-Rosser, then it is ground Church-Rosser relative to $\succ$, but in general the converse is not true. For example, consider the set of rewrite rules

$$\begin{aligned} R \;=\; \{\; & fx \to gx \\ & fx \to hx \\ & fa \to a \\ & ga \to a \\ & ha \to a \;\}, \end{aligned}$$

where $\Sigma = \{f, g, h, a\}$. It is easy to show that R is noetherian with respect to the recursive path ordering generated by the precedence $f \succ g \succ h \succ a$, and, since every ground term reduces to a, it is ground Church-Rosser relative to $\succ$. But R is *not* Church-Rosser, since $hy \longleftarrow_R fy \longrightarrow_R gy$, and hy and gy are irreducible. In general, being Church-Rosser is a stronger condition than being ground Church-Rosser.

Using lemma 4.13, it is easy to show that for any two ground terms u, v, if $u \overset{*}{\longleftrightarrow}_{E \cup R} v$, then there is also a proof Π with sequence of terms $\langle u_0, \ldots, u_n \rangle$ where all the u_i are ground. If $\succ$ is a ground reduction ordering for E, then each equality step $u_{i-1} \longleftrightarrow_E u_i$ in the proof Π must be either of the form $u_{i-1} \longrightarrow_{R(E)} u_i$ or $u_{i-1} \longleftarrow_{R(E)} u_i$.

---

[11] The interested reader might convince himself that because $\succ$ is stable and has the subterm property, for any two terms u and v, $u \succ v$ implies that $Var(v) \subseteq Var(u)$. This fact is sometimes glossed over. In the present case thus $Var(\sigma(r)) \subseteq Var(\sigma(l))$.

Definition 5.4 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. Let $l_1 \to r_1$ and $l_2 \to r_2$ be variants of rules in $E \cup E^{-1} \cup R$ with no variables in common (viewing an equation $l \doteq r \in E \cup E^{-1}$ as the rule $l \to r$). Suppose that for some address β in l_1, l_1/β is *not* a variable and l_1/β and l_2 are unifiable, and let σ be the mgu of l_1/β and l_2. If $\sigma(r_1) \not\succeq \sigma(l_1)$ and $\sigma(r_2) \not\succeq \sigma(l_2)$, the *superposition* of $l_1 \to r_1$ on $l_2 \to r_2$ at β determines a *critical pair* $\langle g, d\rangle$ *of* (E, R), with $g = \sigma(r_1)$ and $d = \sigma(l_1[\beta \leftarrow r_2])$. The term $\sigma(l_1)$ is called the *overlapped term*, and β the *critical pair position*.

The importance of critical pairs lies in the fact that they can be used to eliminate peaks in proofs.

Lemma 5.5 (Critical pair lemma, Knuth and Bendix, [25], Huet [16]) Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. For every peak $s \longleftarrow_{R(E)\cup R} u \longrightarrow_{R(E)\cup R} t$, either there exists some term v such that $s \xrightarrow{*}_{R(E)\cup R} v \xleftarrow{*}_{R(E)\cup R} t$, or there exists a critical pair $\langle g, d\rangle$ of $E \cup R$, an address α in u (s.t. u/α is not a variable) and a substitution η such that, $s = u[\alpha \leftarrow \eta(g)]$ and $t = u[\alpha \leftarrow \eta(d)]$.

We shall now prove that given a pair (E, R) and a reduction ordering $\succnsim$ containing R that is a ground reduction ordering for $E \cup R$, there is a pair (E^ω, R^ω) containing (E, R) that is equivalent to (E, R) and is ground Church-Rosser relative to $\succnsim$. The pair (E^ω, R^ω) can be viewed as an abstract completion of (E, R) (not produced by any specific algorithm). The existence of (E^ω, R^ω) follows from the existence of fair unfailing completion procedures proved by Bachmair, Dershowitz, Hsiang, and Plaisted [1,2,3]. However, this proof requires more machinery than we need for our purposes. We give a more direct and simpler proof (inspired by their proof) that isolates clearly the role played by critical pairs. (In this proof, one will not be distracted by features of completion procedures that have to do with efficiency, like simplification of equations or rules by other rules.) The following definition is needed.

Definition 5.6 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. Let $CR(E, R)$ denote the set of all critical pairs of (E, R) (w.r.t. $\succ$). The sets E^n and R^n are defined inductively as follows: $E^0 = E$, $R^0 = R$, and for every $n \geq 0$,

$$R^{n+1} = R^n \cup \{g \to d \mid \langle g, d\rangle \in CR(E^n, R^n) \text{ and } g \succ d\} \cup \{d \to g \mid \langle g, d\rangle \in CR(E^n, R^n) \text{ and } d \succ g\},$$

and

$$E^{n+1} = E^n \cup \{g \doteq d \mid \langle g, d\rangle \in CR(E^n, R^n),\ g \not\succeq d \text{ and } d \not\succeq g\}.$$

We also let

$$E^\omega = \bigcup_{n\geq 0} E^n \quad \text{and} \quad R^\omega = \bigcup_{n\geq 0} R^n.$$

Thus, R^ω consists of orientable critical pairs obtained from (E, R) (hereditarily), and E^ω consists of nonorientable critical pairs obtained from (E, R) (hereditarily). As the next theorem shows, (E^ω, R^ω) is a kind of abstract completion of (E, R).

Theorem 5.7 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R that can be extended to a ground reduction ordering $\succcurlyeq$ for $E \cup R$. Then, (E^ω, R^ω) is equivalent to (E, R) and is ground Church-Rosser relative to $\succcurlyeq$.

Proof. That (E^ω, R^ω) is equivalent to (E, R) follows easily from the fact that (E^ω, R^ω) contains (E, R) and that critical pairs in $CR(E^n, R^n)$ are provably equal from (E^n, R^n). We need to prove that for any two *ground* terms u, v, if $u \overset{*}{\longleftrightarrow}_{E^\omega \cup R^\omega} v$, then there is a rewrite proof $u \overset{*}{\longrightarrow}_{R(E^\omega)\cup R^\omega} w \overset{*}{\longleftarrow}_{R(E^\omega)\cup R^\omega} v$ for some w. Let

$$\Pi = \langle\langle u_0, \alpha_1, l_1, r_1, \sigma_1, u_1\rangle, \langle u_1, \alpha_2, l_2, r_2, \sigma_2, u_2\rangle, \ldots, \langle u_{n-1}, \alpha_n, l_n, r_n, \sigma_n, u_n\rangle\rangle$$

be a proof that $u \overset{*}{\longleftrightarrow}_{E^\omega \cup R^\omega} v$ (where n is minimal), with $u = u_0$, $v = u_n$, and where u and v are ground. Because $\succcurlyeq$ is a ground reduction ordering for $E \cup R$, as observed earlier, we can always assume that the terms u_i are all ground, and we have in fact a proof $u \overset{*}{\longleftrightarrow}_{R(E^\omega)\cup R^\omega} v$. We show that for every proof Π of the form $u \overset{*}{\longleftrightarrow}_{R(E^\omega)\cup R^\omega} v$, there is a rewrite proof $u \overset{*}{\longrightarrow}_{R(E^\omega)\cup R^\omega} w \overset{*}{\longleftarrow}_{R(E^\omega)\cup R^\omega} v$, by induction on the multiset $\{u_0, \ldots, u_n\}$, using the multiset ordering $\succcurlyeq_m$. For the base case, if the rewrite sequence is either trivial (i.e. $u = v$, corresponding to the multiset $\{u\}$) or consists of a single step (corresponding to the multiset $\{u, v\}$), then clearly the proof has no peaks and so is a rewrite proof. For the induction step, suppose Π is a proof with corresponding multiset $\{u_0, \ldots, u_n\}$ with $n \geq 2$. If Π has no peaks, then it is a rewrite proof and we are done. Otherwise, let $u_{i-1} \longleftarrow_{R(E^\omega)\cup R^\omega} u_i \longrightarrow_{R(E^\omega)\cup R^\omega} u_{i+1}$ be a peak in Π. Note that $u_i \succcurlyeq u_{i-1}$ and $u_i \succcurlyeq u_{i+1}$ since $R(E^\omega)$ is the set of orientable instances w.r.t. $\succcurlyeq$ of $E^\omega \cup (E^\omega)^{-1}$, and since R^ω is contained in $\succ$ by its definition. By the critical pair lemma 5.5, either there is some term v such that $u_{i-1} \overset{*}{\longrightarrow}_{R(E^\omega)\cup R^\omega} v \overset{*}{\longleftarrow}_{R(E^\omega)\cup R^\omega} u_{i+1}$, or $u_{i-1} \longleftrightarrow_{[\eta(g)\doteq\eta(d)]} u_{i+1}$, where $\eta(g) \doteq \eta(d)$ is a ground instance of a critical pair $\langle g, d\rangle$ of $E^\omega \cup R^\omega$. In the first case, we can replace the peak by a rewrite proof relative to $\succcurlyeq$ and we obtain a proof Π' with associated sequence $\langle u_0, \ldots, u_{i-1}, v_1, \ldots, v_k, u_{i+1}, \ldots, u_n\rangle$ such that $u_i \succcurlyeq v_j$ for all j, $1 \leq j \leq k$. Hence

$$\{u_0, \ldots, u_n\} \succcurlyeq_m \{u_0, \ldots, u_{i-1}, v_1, \ldots, v_k, u_{i+1}, \ldots, u_n\},$$

and we conclude by applying the induction hypothesis. In the second case, observe that $E^\omega \cup R^\omega$ is closed under the formation of critical pairs, and so, $g \doteq d \in E^\omega \cup R^\omega$. Thus,

$\eta(g) \doteq \eta(d)$ is orientable either because $g \doteq d \in R^\omega$, or because $g \doteq d \in E^\omega$ and $\succcurlyeq$ is a ground reduction ordering relative to $E \cup R$. Hence, the peak can be replaced by a proof step $u_{i-1} \longleftrightarrow_{R(E^\omega) \cup R^\omega} u_{i+1}$, obtaining a proof Π' with associated sequence $\langle u_0, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n \rangle$. Since

$$\{u_0, \ldots, u_n\} \succcurlyeq_m \{u_0, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n\},$$

we conclude by applying the induction hypothesis. This concludes the proof. □

Note that since a proof is finite, for any proof $u \overset{*}{\longleftrightarrow}_{E^\omega \cup R^\omega} v$, there is a rewrite proof $u \overset{*}{\longrightarrow}_{R(E^k) \cup R^k} w \overset{*}{\longleftarrow}_{R(E^k) \cup R^k} v$ for some natural number k. Thus, only finitely many critical pairs need to be computed. In some sense, the number of critical pairs to be computed shows how "nonground Church-Rosser" (E, R) is. Also, a sufficient condition for theorem 5.7 to apply is that the reduction ordering $\succ$ containing R be also a total reduction ordering on ground terms. In particular, the theorem applies when $R = \emptyset$, in which case only a total simplification ordering on ground terms is needed. As mentioned earlier, such orderings always exist. On the other hand, given a set R of rewrite rules, there may not be any simplification ordering containing R that is also total on ground terms. Such behavior is illustrated by the set $R = \{f(a) \to f(b),\ g(b) \to g(a)\}$.

The fact that a system (E, R) is ground Church-Rosser has the important consequence that $R(E) \cup R$ is canonical on ground terms. This is shown as follows. First, note that $R(E) \cup R$ is Noetherian on ground terms, since R is contained in the reduction ordering $\succ$ by hypothesis and $R(E)$ is also contained in $\succ$ since it is the set of orientable instances of E relative to $\succ$ (which is total on ground terms). To show confluence, note that for any *ground* terms u, v_1, v_2, if

$$v_1 \overset{*}{\longleftarrow}_{R(E) \cup R} u \overset{*}{\longrightarrow}_{R(E) \cup R} v_2,$$

then $v_1 \overset{*}{\longleftrightarrow}_{R(E) \cup R} v_2$, and since (E, R) is ground Church-Rosser, there is a rewrite proof

$$v_1 \overset{*}{\longrightarrow}_{R(E) \cup R} w \overset{*}{\longleftarrow}_{R(E) \cup R} v_2$$

for some w. Hence, every ground term u can be reduced to a unique irreducible term $u{\downarrow}$ (w.r.t. $R(E) \cup R$), its *normal form*.

Definition 5.8 Given a rewrite system R, we say that a substitution σ is *reduced* w.r.t. R iff every term of the form $\sigma(x)$ is irreducible w.r.t. R, where $x \in D(\sigma)$.

It is very useful to observe that if a procedure P for finding sets of E-unifiers satisfies the property stated in the next definition, then in order to show that this procedure yields complete sets, there is no loss of generality in showing completeness with respect to *ground* E-unifiers whose domains contain $Var(S)$ (that is, in clause (iii) of definition 4.5, $\theta(x)$ is a ground term for every $x \in D(\theta)$, and $Var(S) \subseteq D(\theta)$).

Definition 5.9 We call an E-unification procedure P *pure* if for every ranked alphabet Σ, every finite set E of equations over $T_\Sigma(X)$ and every term system S over $T_\Sigma(X)$, if $U = P(E,S)$ is the set of E-unifiers for S given by procedure P, then for every $\sigma \in U$, for every $x \in D(\sigma)$, every constant or function symbol occurring in $\sigma(x)$ occurs either in some equation in E or some pair in S.

In other words, $P(E,S)$ does not contain constant or function symbols that do not already occur in the input (E,S). (For example, it is easy to prove that all the sets of transformations presented in this paper are pure.) To prove our previous claim, we proceed as follows. We add countably infinitely many new (distinct) constants c_x to Σ, each constant c_x being associated with the variable x. The resulting alphabet is denoted by Σ_{SK}. If θ is not ground, we create the Skolemized version of θ, that is, the substitution $\widehat{\theta}$ obtained by replacing the variables in the terms $\theta(x)$ by new (distinct) constants.[12]

Lemma 5.10 Given a pure E-unification procedure P, assume that for every ranked alphabet Σ, every finite set E of equations over $T_\Sigma(X)$ and every S over $T_\Sigma(X)$, the set $U = P(E,S)$ of E-unifiers of S given by P satisfies conditions (i) and (ii) of definition 4.5, and for every E-unifier θ of S such that $Var(S) \subseteq D(\theta)$ and $\theta(x) \in T_\Sigma$ for every $x \in D(\theta)$, there is some $\sigma \in U$ such that $\sigma \leq_E \theta[Var(S)]$ (c.f. condition (iii) of definition 4.5). Then every set $U = P(E,S)$ is a complete set of E-unifiers for S.

Proof. Let θ be any E-unifier of S over $T_\Sigma(X)$. If $D(\theta)$ does not contain $Var(S)$, extend θ such that $\theta(y) = c_y$ for every $y \in Var(S) - D(\theta)$, and let $\widehat{\theta}$ be the Skolemized version of this extension of θ. We are now considering the extended alphabet Σ_{SK}. It is immediately verified that $\widehat{\theta}$ is also an E-unifier of S such that $Var(S) \subseteq D(\widehat{\theta})$ and $\widehat{\theta}(x) \in T_{\Sigma_{SK}}$ for all $x \in D(\widehat{\theta})$. Then, there is some $\sigma \in U$ such that $\sigma \leq_E \widehat{\theta}[Var(S)]$, which means that there is some substitution η (over $T_{\Sigma_{SK}}(X)$) such that $\sigma \circ \eta =_E \widehat{\theta}[Var(S)]$. Note that by the purity of P, since E and S do not contain Skolem constants, σ does not contain Skolem constants. Let η' be obtained from η by changing each Skolem constant back to the corresponding variable. Since σ does not contain Skolem constants, it is immediately verified that $\sigma \circ \eta' =_E \theta$. Thus, the set U is a complete set of E-unifiers for S over $T_\Sigma(X)$. □

The following result is also useful.

Lemma 5.11 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R, and assume that (E,R) is ground Church-Rosser relative to $\succ$. If θ is a ground

[12] More precisely, $\widehat{\theta}$ is obtained from θ by replacing every variable y in each term $\theta(x)$ by the corresponding Skolem constant c_y, for each $x \in D(\theta)$.

(E, R)-unifier of u and v and $Var(u, v) \subseteq D(\theta)$, then there is a ground substitution σ that is reduced w.r.t. $R(E) \cup R$ such that $\sigma =_{E \cup R} \theta$, σ is an (E, R)-unifier of u and v, and $Var(u, v) \subseteq D(\sigma)$.

Proof. Since (E, R) is ground Church-Rosser relative to $\succ$, $R(E) \cup R$ is canonical on ground terms. Thus, if $\theta(u) \overset{*}{\longleftrightarrow}_{E \cup R} \theta(v)$, since θ is ground and $Var(u, v) \subseteq D(\theta)$, then there is a rewrite proof

$$\theta(u) \overset{*}{\longrightarrow}_{R(E) \cup R} u' \overset{*}{\longrightarrow}_{R(E) \cup R} w \overset{*}{\longleftarrow}_{R(E) \cup R} v' \overset{*}{\longleftarrow}_{R(E) \cup R} \theta(v)$$

where w is ground and in normal form (w.r.t. $R(E) \cup R$), and where the reductions $\theta(u) \overset{*}{\longrightarrow}_{R(E) \cup R} u'$ and $v' \overset{*}{\longleftarrow}_{R(E) \cup R} \theta(v)$ reduce each $\theta(x)$ $(x \in D(\theta))$ to its normal form $\theta(x)\downarrow$ (w.r.t. $R(E) \cup R$). Thus, defining the reduced substitution σ such that $\sigma(x) = \theta(x)\downarrow$ for each $x \in D(\theta)$, we have $u' = \sigma(u)$, $v' = \sigma(v)$, σ is a ground (E, R)-unifier of u and v, and $\sigma =_{E \cup R} \theta$. □

For our next result, we need the following definition.

Definition 5.12 Given a rewrite system R, a rewrite step $u \longrightarrow_{[\beta, l \doteq r, \rho]} v$ is *innermost* (w.r.t. R) iff every proper subterm of $u/\beta = \rho(l)$ is irreducible w.r.t. R.

The next lemma shows that in ground Church-Rosser systems, normal forms can always be reached via certain canonical innermost rewrite sequences. The proof is not trivial because $Var(r) - Var(l)$ may be nonempty for an equation $l \doteq r \in E \cup E^{-1}$.

Lemma 5.13 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R, and assume that (E, R) is ground Church-Rosser relative to $\succ$. Every ground term u reduces to its normal form $u\downarrow$ (w.r.t. $R(E) \cup R$) in a sequence of innermost reductions $u \overset{*}{\longrightarrow}_{R(E) \cup R} u\downarrow$, such that for every rule $\rho(l) \to \rho(r)$ used in the sequence, ρ is reduced (w.r.t. $R(E) \cup R$).

Proof. Since (E, R) is ground Church-Rosser relative to $\succ$, $R(E) \cup R$ is canonical on ground terms. We proceed by induction on the well founded ordering $\succ$. If u is in normal form, we are done. Otherwise, there is a sequence of reduction steps $u \overset{*}{\longrightarrow}_{R(E) \cup R} u\downarrow$, and and because u is ground, we can assume that every rule $\rho(l) \to \rho(r)$ used in such a proof is ground. Note that $\rho(l) \succ \rho(r)$ whenever either $l \to r \in R$ or $\rho(l) \to \rho(r) \in R(E)$, and $Var(l) \cup Var(r) = D(\rho)$ since $\rho(l)$ and $\rho(r)$ are ground.[13] If u is not in normal form, there must be some innermost step

$$u \longrightarrow_{[\beta, l \doteq r, \rho]} u[\beta \leftarrow \rho(r)].$$

[13] Certainly, $\rho(l)$ and $\rho(r)$ ground implies that $Var(l) \cup Var(r) \subseteq D(\rho)$, but the fact that ρ may be defined outside of $Var(l) \cup Var(r)$ is not used anywhere, so we might as well assume that $Var(l) \cup Var(r) = D(\rho)$.

For every $x \in Var(l)$, $\rho(x)$ must be in normal form (w.r.t. $R(E) \cup R$), since otherwise some proper subterm of $\rho(l) = u/\beta$ would be reducible, contradicting the fact that we have an innermost step. For each $x \in (Var(r) - Var(l))$, let $\rho(x)\downarrow$ be the normal form of $\rho(x)$ (w.r.t. $R(E) \cup R$), and let ρ' be the reduced substitution such that $\rho'(x) = \rho(x)\downarrow$ for each $x \in (Var(r) - Var(l))$, and $\rho'(x) = \rho(x)$ for each $x \in Var(l)$. The definition of ρ' implies that $\rho'(l) = \rho(l)$ and $\rho(x) \succeq \rho'(x)$ for every $x \in D(\rho)$. Thus, $\rho(l) \succ \rho(r)$ implies that $\rho'(l) \succ \rho'(r)$. Since $R(E) \cup R$ is canonical on ground terms, $\rho'(l) = \rho(l)$, and $u = u[\beta \leftarrow \rho(l)]$, using the rule $\rho'(l) \to \rho'(r)$, we have a proof

$$u = u[\beta \leftarrow \rho'(l)] \ \longrightarrow_{R(E)\cup R} \ u[\beta \leftarrow \rho'(r)] \ \stackrel{*}{\longrightarrow}_{R(E)\cup R} \ u\downarrow$$

where the first reduction step is innermost and ρ' is reduced (w.r.t. $R(E) \cup R$). Letting $u' = u[\beta \leftarrow \rho'(r)]$, we have $u \succ u'$ since $\rho'(l) \succ \rho'(r)$. We conclude by applying the induction hypothesis to u'. □

We are now ready to apply the results of this section to prove the completeness of an improved set of transformations.

6 Completeness of an Improved Set of Transformations

Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R.

Definition 6.1 (The set of transformation rules $\mathcal{T}$) The set $\mathcal{T}$ consists of the transformations Trivial, Term Decomposition, and Variable Elimination from the set $\mathcal{ST}$ plus one more transformation defined as follows:

Lazy Paramodulation: Given a multiset of pairs $\{\langle u, v\rangle\} \cup S$, then

$$\{\langle u, v\rangle\} \cup S \Longrightarrow \{\langle u/\beta, l\rangle, \langle u[\beta \leftarrow r], v\rangle\} \cup S,$$

where β is a nonvariable occurrence in u (i.e., u/β is *not* a variable) and $l \doteq r$ is a variant (whose variables do not occur in $\{\langle u, v\rangle\} \cup S$) of some equation in $E \cup E^{-1} \cup R \cup R^{-1}$. Furthermore, if l is not a variable, then $Root(u/\beta) = Root(l)$ and Term Decomposition is immediately applied to $\langle u/\beta, l\rangle$ (this corresponds to a leftmost rewrite at address β).[14] Thus, if l is not a variable, letting $l = f(l_1, \ldots, l_k)$ and $u/\beta = f(t_1, \ldots, t_k)$, Lazy Paramodulation can be specialized to:

$$\{\langle u, v\rangle\} \cup S \Longrightarrow \{\langle t_1, l_1\rangle, \ldots, \langle t_k, l_k\rangle, \langle u[\beta \leftarrow r], v\rangle\} \cup S.$$

[14] As with Root Rewriting, note that this is *not* simply a paramodulation step, nor simply a paramodulation step where the unification of u/β and l is delayed; it allows further rewrite steps to occur below (but *not* at) the roots of u/β and l, hence the name *Lazy* Paramodulation.

Recall that a pair $\langle u, v\rangle$ is in fact a multiset, and so Lazy Paramodulation also applies from v to u, as in

$$\{\langle u, v\rangle\} \cup S \Longrightarrow \{\langle v/\beta, l\rangle, \langle u, v[\beta \leftarrow r]\rangle\} \cup S,$$

where β is a nonvariable occurrence in v. As in our previous set of transformations, we note that systems are multisets and the unions in this rule are multiset unions.

In order to distinguish between the set $\mathcal{BT}$ and the set $\mathcal{T}$, the former will be called $\mathcal{BT}$-transformations and the latter $\mathcal{T}$-transformations. The soundness of the $\mathcal{T}$-transformations is given by

Theorem 6.2 (Soundness of $\mathcal{T}$) If $S \stackrel{*}{\Longrightarrow} S'$, using transformations from the set $\mathcal{T}$, with S' in solved form, then $\sigma_{S'}|_{Var(S)} \in U_E(S)$.

Proof. The only difference from theorem 4.12 is that we must prove the soundness of Lazy Paramodulation, i.e., that if $S \Longrightarrow S'$ using this transformation, then $U_E(S') \subseteq U_E(S)$. But clearly if $\theta(u/\beta) \stackrel{*}{\longleftrightarrow}_E \theta(l)$ and $\theta(u[\beta \leftarrow r]) \stackrel{*}{\longleftrightarrow}_E \theta(v)$ then we have

$$\theta(u) = \theta(u[\beta \leftarrow u/\beta]) \stackrel{*}{\longleftrightarrow}_E \theta(u[\beta \leftarrow l]) \longrightarrow_{[\beta, l \dot{=} r, \theta]} \theta(u[\beta \leftarrow r]) \stackrel{*}{\longleftrightarrow}_E \theta(v),$$

from which the result follows. □

The completeness of the set of $\mathcal{T}$-transformations is shown in two steps. First, we assume that (E, R) is ground Church-Rosser and we show that the $\mathcal{T}$-transformations are complete, even when Lazy Paramodulation is restricted so that it applies only when either $\beta = \epsilon$ (i.e. at the root) or when one of u, v is a variable (but not both). Then, we use theorem 5.7 and a lemma that shows that the computation of critical pairs can be simulated by Lazy Paramodulation unrestricted.

The quickest way to prove the completeness of the set $\mathcal{T}$ in the case where (E, R) is ground Church-Rosser w.r.t. $\succ$ is to adapt the definition of proof trees. Another proof consists in showing that applications of Root Imitation can be bounded and simulated by Lazy Paramodulation, but this proof is more cumbersome. Suppose that θ is an (E, R)-unifier of a system S. First, observe that any procedure using the transformations in $\mathcal{T}$ satisfies the purity condition of definition 5.9, and by lemma 5.11 and lemma 5.10, we can assume that θ is reduced w.r.t. $R(E) \cup R$, ground, and that $Var(S) \subseteq D(\theta)$. Since (E, R) is ground Church-Rosser relative to $\succ$, there is a rewrite proof

$$\theta(u) \stackrel{*}{\longrightarrow}_{R(E)\cup R} w \stackrel{*}{\longleftarrow}_{R(E)\cup R} \theta(v)$$

for every pair $\langle u, v\rangle \in S$, where w is irreducible (w.r.t. $R(E) \cup R$). By lemma 5.13 (and because θ is reduced), we can assume that for every rule $\rho(l) \rightarrow \rho(r)$ used in each of these

rewrite proofs, ρ is reduced (w.r.t. $R(E) \cup R$). Now, since θ and all the matching substitutions ρ are ground, and by our assumption that all equations used are variants, it is immediate that we can form a ground substitution extending θ incorporating all the matching substitutions. For simplicity of notation we shall also call this extension θ. Observe that the extended substitution θ is still reduced. The crucial observation is the following. If v is a variable, say y, because θ is reduced we must have $w = \theta(y)$ and $\theta(u) \xrightarrow{*}_{R(E)\cup R} \theta(y)$. If u is also a variable, say x, we must have $\theta(x) = w = \theta(y)$. Thus, when $\langle u, v\rangle$ is a pair of variables, Variable Elimination always applies. Also, in the case of a pair $\langle u, y\rangle \in S$ where u is a compound term, y is a variable, and there is a sequence of rewrite steps $\theta(u) \xrightarrow{*}_{R(E)\cup R} \theta(y)$ but no step takes place at the root, some rewrite step must take place at some address β in u such that u/β is not a variable. More specifically, let $\{\beta_1, \ldots, \beta_m\}$ be the set of independent addresses (of nonvariable occurrences) in v at which topmost rewrite steps take place in $\theta(u) \xrightarrow{*}_{R(E)\cup R} \theta(y)$. Then, for each i, $1 \leq i \leq m$, there is a finite set $\{l^i_1 \doteq r^i_1, \ldots, l^i_{n_i} \doteq r^i_{n_i}\}$ of variants of equations in $E \cup E^{-1} \cup R$ such that

$$\theta(u/\beta_i) \xrightarrow{*}_{R(E)\cup R} \theta(l^i_1) \xrightarrow{*}_{R(E)\cup R} \theta(r^i_1) \xrightarrow{*}_{R(E)\cup R} \ \ldots\ \theta(l^i_{n_i}) \xrightarrow{*}_{R(E)\cup R} \theta(r^i_{n_i}),$$

and we also have

$$\theta(u[\beta_1 \leftarrow r^1_{n_1}, \ldots, \beta_m \leftarrow r^m_{n_m}]) \xrightarrow{*}_{R(E)\cup R} \theta(y).$$

This suggests modifying the definition of proof trees to allow rewrite rule insertion not just at address ϵ, but more generally at topmost addresses where rewrites take place. Furthermore, this generalization is only necessary in the case of pairs $\langle u, v\rangle$ where v (or u, but not both) is a variable. Now, decomposition only applies to pairs $\langle u, v\rangle$ where both u and v are compound terms whose root symbol is identical. For a pair $\langle u, v\rangle$ where v (or u, but not both) is a variable, we have either an axiom, or rewrite rule insertion. The new definition is as follows.

Definition 6.3 Let θ be some (idempotent) substitution. The set of *proof trees associated with* θ is defined inductively as follows. For simplicity we use $*$ as a syntactic variable for one of the symbols $\approx$, $\sim$, or $=$.

(i) (Axioms) For every term u, the one node tree labeled with $u = u$ is a proof tree associated with θ. For every two terms $u \neq v$, at least one of which is a variable, such that $\theta(u) = \theta(v)$, the one node tree labeled with $u = v$ is a proof tree associated with θ.

(ii) (Term Decomposition) Let u and v be a pair of compound terms of the form $f(u_1, \ldots, u_n)$ and $f(v_1, \ldots, v_n)$. Given any n proof trees $T_1, \ldots, T_n$ associated with θ, where each T_i is a proof tree whose root is labeled with $u_i * v_i$, the tree T whose root is labeled with $u \sim v$ and such that $T/i = T_i$ for $1 \leq i \leq n$ is a proof tree associated with θ.

(iii) (Rewrite Rule Insertion) Let u and v be a pair of terms. If both u, v are compound terms, let $l_i \doteq r_i$ for $1 \leq i \leq m$ be variants of equations from $E \cup E^{-1} \cup R$. Furthermore, let $T_1, \ldots, T_{m+1}$ be proof trees associated with θ, where T_1 is a proof tree whose root is labeled with either $u = l_1$ or $u \sim l_1$, and for $2 \leq i \leq m$, T_i is a proof tree whose root is labeled with either $r_{i-1} = l_i$ or $r_{i-1} \sim l_i$, and T_{m+1} is a proof tree whose root is labeled with either $r_m = v$ or $r_m \sim v$. Then the tree T whose root is labeled with $u \approx v$ and such that $T/i = T_i$ for $1 \leq i \leq m+1$ is a proof tree associated with θ.

If one of u, v (but not both) is a variable, say $v = y$, let $\{\beta_1, \ldots, \beta_m\}$ be a set of independent addresses (of nonvariable occurrences) in v, and for each i, $1 \leq i \leq m$, let $\{l_1^i \doteq r_1^i, \ldots, l_{n_i}^i \doteq r_{n_i}^i\}$ be a set of variants of equations in $E \cup E^{-1} \cup R$. For each i, $1 \leq i \leq m$, let $T_1^i, \ldots, T_{n_i}^i$ be proof trees associated with θ, where T_1^i is a proof tree whose root is labeled with either $u/\beta_i = l_1^i$ or $u/\beta_i \sim l_1^i$, T_j^i $(2 \leq j \leq n_i)$ is a proof tree whose root is labeled with either $r_{j-1}^i = l_j^i$ or $r_{j-1}^i \sim l_j^i$, and T^{m+1} a proof tree whose root is labeled with either $u[\beta_1 \leftarrow r_{n_1}^1, \ldots, \beta_m \leftarrow r_{n_m}^m] = y$ or $u[\beta_1 \leftarrow r_{n_1}^1, \ldots, \beta_m \leftarrow r_{n_m}^m] \approx y$. Then, the tree T whose root is labeled with $u \approx y$ and having $n_1 + \ldots + n_m + 1$ sons defined such that $T/j = T_j^1$, for $1 \leq j \leq n_1$, $T/(n_1 + \ldots + n_{k-1} + j) = T_j^k$, for $2 \leq k \leq m$, $1 \leq j \leq n_k$, and $T/(n_1 + \ldots + n_m + 1) = T^{m+1}$, is a proof tree associated with θ. We also assume that all edges from $u \approx y$ to the root nodes of the trees $T_1^i, \ldots, T_{n_i}^i$ are labeled with the address β_i. When $\beta_1 = \ldots = \beta_m = \epsilon$, this label is omitted.

A proof system is now defined as a pair $\langle \theta, P \rangle$ where θ is a substitution and P is a set of proof trees associated with θ. It is now easy to adapt the proofs of theorem 4.19 and theorem 4.20 to the new definition of proof trees, in the case where (E, R) is ground Church-Rosser.

Theorem 6.4 For some given substitution θ, system S, and pair (E, R), if $\langle \theta, P \rangle$ is a proof system for θ and S, then θ is an (E, R)-unifier of S.

Note that the result actually holds for any substitution, not necessarily ground or idempotent, and does not require (E, R) to be ground Church-Rosser. On the other hand, the fact that (E, R) is ground Church-Rosser is crucial to the completeness of proof trees.

Theorem 6.5 If (E, R) is ground Church-Rosser and θ is a ground reduced (E, R)-unifier of S such that $Var(S) \subseteq D(\theta)$, then there exists a proof system $\langle \theta, P \rangle$ associated with θ and S.

Proof. It is similar to that of theorem 4.20 and proceeds by multiset induction. The only changes occur in the case of a pair $\langle u, v \rangle$ where u or v is a variable. Instead of decomposition, we either have an axiom or rewrite rule insertion as discussed earlier. The details are straightforward. □

We are now in the position to prove the completeness of the set $\mathcal{T}$ when (E, R) is ground Church-Rosser.

Lemma 6.6 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R, and assume that (E, R) is ground Church-Rosser relative to $\succ$. Given any system S if θ is an (E, R)-unifier of S, then there is a sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}$ (using variants of equations in $E \cup E^{-1} \cup R$) yielding a solved system $\widehat{S}$ such that if $\sigma_{\widehat{S}}$ is the substitution associated with $\widehat{S}$, then $\sigma_{\widehat{S}} \leq_{E \cup R} \theta[Var(S)]$. Furthermore, Lazy Paramodulation can be restricted so that it is applied only when either $\beta = \epsilon$ or one of u, v is a variable (but not both).

Proof. The proof is similar to the proof of theorem 4.26. The only significant difference is that we need to use theorem 6.5 instead of theorem 4.20. In the present situation, the proof transformation (E) is never used, and clearly lemma 4.23 and lemma 4.25 still hold. The only thing to verify to make sure that theorem 4.26 goes through is to check that for every sequence of proof tree transformations, there is a corresponding sequence of transformations on root systems. The only new case is that of a proof tree $(u \approx y)$ where $u \approx y$ has $n_1 + \ldots + n_m + 1$ sons corresponding to rewrite rule insertions at independent addresses $\{\beta_1, \ldots, \beta_m\}$. For each i, $1 \leq i \leq m$, we have a set $\{l_1^i \doteq r_1^i, \ldots, l_{n_i}^i \doteq r_{n_i}^i\}$ of variants of equations in $E \cup E^{-1} \cup R$. We need to show that

$$\begin{aligned}\langle u, y\rangle \stackrel{+}{\Longrightarrow} & \langle u/\beta_1, l_1^1\rangle, \langle r_1^1, l_2^1\rangle, \ldots, \langle r_{n_1-1}^1, l_{n_1}^1\rangle, \\ & \ldots \\ & \langle u/\beta_m, l_1^m\rangle, \langle r_1^m, l_2^m\rangle, \ldots, \langle r_{n_m-1}^m, l_{n_m}^m\rangle, \\ & \langle u[\beta_1 \leftarrow r_{n_1}^1, \ldots, \beta_m \leftarrow r_{n_m}^m], y\rangle.\end{aligned}$$

This is easily shown by repeated use of Lazy Paramodulation, first at address β_1, then β_2, ..., and finally at β_m. This sequence starts as follows:

$$\begin{aligned}\langle u, y\rangle &\Longrightarrow \langle u/\beta_1, l_1^1\rangle, \langle u[\beta_1 \leftarrow r_1^1], y\rangle \\ &\Longrightarrow \langle u/\beta_1, l_1^1\rangle, \langle r_1^1, l_2^1\rangle, \langle u[\beta_1 \leftarrow r_2^1], y\rangle \\ &\stackrel{*}{\Longrightarrow} \ldots \\ &\Longrightarrow \langle u/\beta_1, l_1^1\rangle, \langle r_1^1, l_2^1\rangle, \ldots, \langle r_{n_1-1}^1, l_{n_1}^1\rangle, \langle u[\beta_1 \leftarrow r_{n_1}^1], y\rangle \\ &.\end{aligned}$$

The details are straightforward and left to the reader. □

In order to prove the completeness of the $\mathcal{T}$-transformations in the general case, the following lemma showing that the computation of critical pairs can be simulated by Lazy Paramodulation is needed.

Lemma 6.7 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. For every finite system S, every sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}$ using equations in $E^\omega \cup (E^\omega)^{-1} \cup R^\omega$ can be converted to a sequence $S \stackrel{*}{\Longrightarrow} \widehat{S}'$ using equations only in $E \cup E^{-1} \cup R \cup R^{-1}$, such that $\widehat{S}$ and $\widehat{S}'$ are in solved form and $\sigma_{\widehat{S}}|_{Var(S)} = \sigma_{\widehat{S}'}|_{Var(S)}$.

Proof. The lemma is established by proving by induction on k that every sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}$ using equations in $E^k \cup (E^k)^{-1} \cup R^k$ can be converted to a sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}'$ using equations only in $E \cup E^{-1} \cup R \cup R^{-1}$, such that $\widehat{S}$ and $\widehat{S}'$ are in solved form and $\sigma_{\widehat{S}}|_{Var(S)} = \sigma_{\widehat{S}'}|_{Var(S)}$. The base case is trivial. For the induction step, let $\sigma(r_1) \doteq \sigma(l_1[\beta \leftarrow r_2])$ be an equation obtained by forming a critical pair from $l_1 \to r_1$ and $l_2 \to r_2$ at β in l_1, with mgu σ of l_1/β and l_2. It is sufficient to show that whenever such a critical pair is used in one step of Lazy Paramodulation, say

$$\langle u, v\rangle \Longrightarrow \langle u/\alpha, \sigma(r_1)\rangle, \langle u[\alpha \leftarrow \sigma(l_1[\beta \leftarrow r_2])], v\rangle \tag{1}$$

or

$$\langle u, v\rangle \Longrightarrow \langle u/\alpha, \sigma(l_1[\beta \leftarrow r_2])\rangle, \langle u[\alpha \leftarrow \sigma(r_1)], v\rangle, \tag{2}$$

where α is some nonvariable occurrence in u, then there is another sequence of transformations using only the equations $l_1 \doteq r_1$ and $l_2 \doteq r_2$. Such a sequence for (1) is as follows:

$$\langle u, v\rangle \Longrightarrow \langle u/\alpha, r_1\rangle, \langle u[\alpha \leftarrow l_1], v\rangle,$$

using the equation $r_1 \doteq l_1$ at α in u. Note that equation $l_1 \doteq r_1$ is used backwards. Next,

$$\langle u/\alpha, r_1\rangle, \langle u[\alpha \leftarrow l_1], v\rangle \Longrightarrow \langle u/\alpha, r_1\rangle, \langle l_1/\beta, l_2\rangle, \langle u[\alpha \leftarrow l_1[\beta \leftarrow r_2]], v\rangle,$$

using the equation $l_2 \doteq r_2$ at $\alpha\beta$ in $u[\alpha \leftarrow l_1]$ and the fact that

$$u[\alpha \leftarrow l_1]/\alpha\beta = l_1/\beta$$

and

$$u[\alpha \leftarrow l_1][\alpha\beta \leftarrow r_2] = u[\alpha \leftarrow l_1[\beta \leftarrow r_2]];$$

Finally, use any sequence of transformations from the set $\mathcal{ST}$ that computes the mgu σ of l_1/β and l_2 with associated solved system S_1:

$$\langle u/\alpha, r_1\rangle, \langle l_1/\beta, l_2\rangle, \langle u[\alpha \leftarrow l_1[\beta \leftarrow r_2]], v\rangle \stackrel{*}{\Longrightarrow} S_1 \cup \langle u/\alpha, \sigma(r_1)\rangle, \langle u[\alpha \leftarrow \sigma(l_1[\beta \leftarrow r_2])], v\rangle.$$

In these last steps, we used the fact that $D(\sigma)$ is disjoint from the set of variables $Var(u) \cup Var(v)$. A sequence for (2) is as follows:

$$\langle u, v\rangle \Longrightarrow \langle u/\alpha, l_1\rangle, \langle u[\alpha \leftarrow r_1], v\rangle,$$

using equation $l_1 \doteq r_1$ at α in u;

$$\langle u/\alpha, l_1\rangle, \langle u[\alpha \leftarrow r_1], v\rangle \Longrightarrow \langle u/\alpha, l_1[\beta \leftarrow r_2]\rangle, \langle l_1/\beta, l_2\rangle, \langle u[\alpha \leftarrow r_1], v\rangle,$$

using equation $l_2 \doteq r_2$ at β in l_1. Finally, use any sequence of transformations from the set $\mathcal{ST}$ that computes the mgu σ of l_1/β and l_2 with associated solved system S_1:

$$\langle u/\alpha, l_1[\beta \leftarrow r_2]\rangle, \langle l_1/\beta, l_2\rangle, \langle u[\alpha \leftarrow r_1], v\rangle \stackrel{*}{\Longrightarrow} S_1 \cup \langle u/\alpha, \sigma(l_1[\beta \leftarrow r_2])\rangle, \langle u[\alpha \leftarrow \sigma(r_1)], v\rangle.$$

(we also used the fact that $D(\sigma)$ is disjoint from $Var(u) \cup Var(v)$.) $\square$

Finally, we can prove the completeness of the $\mathcal{T}$-transformations in the general case.

Theorem 6.8 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R total on ground terms. Given any finite system S, if θ is an (E, R)-unifier of S, then there is a sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}$ (using variants of equations in $E \cup E^{-1} \cup R \cup R^{-1}$) yielding a solved system $\widehat{S}$ such that if $\sigma_{\widehat{S}}$ is the substitution associated with $\widehat{S}$, then $\sigma_{\widehat{S}} \leq_{E \cup R} \theta[Var(S)]$.

Proof. By theorem 5.7, $E^\omega \cup R^\omega$ is equivalent to (E, R) and is ground Church-Rosser relative to $\succ$. By lemma 6.6, there is a sequence of $\mathcal{T}$-transformations $S \stackrel{*}{\Longrightarrow} \widehat{S}$ using variants of equations in $E^\omega \cup (E^\omega)^{-1} \cup R^\omega$ yielding a solved system $\widehat{S}$ such that if $\sigma_{\widehat{S}}$ is the substitution associated with $\widehat{S}$, then $\sigma_{\widehat{S}} \leq_{E \cup R} \theta[Var(S)]$. Finally, we use lemma 6.7 to eliminate uses of critical pairs, obtaining a sequence where all equations are in $E \cup E^{-1} \cup R \cup R^{-1}$. $\square$

Note that when (E, R) is ground Church-Rosser, equations in E are used as two-way rules in Lazy Paramodulation, but rules in R can be used oriented. This means that in a step

$$\langle u, v\rangle \Longrightarrow \langle u/\beta, l\rangle, \langle u[\beta \leftarrow r], v\rangle,$$

where β is a nonvariable occurrence in u, then $l \doteq r \in E \cup E^{-1}$ if $l \doteq r$ is not in R, but $r \to l$ is not tried if $l \to r$ is in R, and similarly for a step

$$\langle u, v\rangle \Longrightarrow \langle u, v[\beta \leftarrow r]\rangle, \langle l, v/\beta\rangle,$$

where β is a nonvariable occurrence in v. Furthermore, Lazy Paramodulation can be restricted so that it applies only when either $\beta = \epsilon$ or one of u, v is a variable (but not both). This is in contrast to the general case where even rules in R may have to be used as two-way rules due to the computation of critical pairs. Also, Lazy Paramodulation may have to be applied with $\beta \neq \epsilon$ even when both u and v are not variables. This case only seems necessary to compute critical pairs. So far, we have failed to produce an example where Lazy Paramodulation needs to be applied in its full generality (that is, when neither u nor v is a variable and $\beta \neq \epsilon$). We conjecture that $\mathcal{T}$ is still complete if Lazy Paramodulation is restricted so that it applies only when either $\beta = \epsilon$ or one of u, v is a variable (but not both). The following example might help the reader's intuition.

Example 6.9 Let $E = \{f(g(x)) \doteq x\ [1],\ g(h(y)) \doteq g(k(y))\ [2],\ g(k(f(z))) \doteq z\ [3]\}$, and consider finding E-unifiers for the pair $\langle h(u), u\rangle$. Equations [1] and [2] overlap at 1 in $f(g(x))$, and we get the critical pair $h(v) \doteq f(g(k(v)))$ [4]. We have the sequence of transformations (using equation [4]):

$$\begin{aligned}
\langle h(u), u\rangle &\Longrightarrow_{\text{para}} \langle h(u), h(v)\rangle, \langle f(g(k(v))), u\rangle && \text{using [4]}\\
&\Longrightarrow_{\text{para}} \langle h(u), h(v)\rangle, \langle g(k(v)), g(k(f(z)))\rangle, \langle f(z), u\rangle && \text{using [3]}\\
&\overset{*}{\Longrightarrow}_{\text{dec}} \langle u, v\rangle, \langle v, f(z)\rangle, \langle f(z), u\rangle &&\\
&\Longrightarrow_{\text{vel}} \langle u, v\rangle, \langle u, f(z)\rangle. && \text{applied to } v\\
&\Longrightarrow_{\text{vel}} \langle f(z), v\rangle, \langle u, f(z)\rangle. && \text{applied to } u
\end{aligned}$$

Thus, $[f(z)/u, f(z)/v]$ is an E-unifier of $\langle h(u), u\rangle$, and $[f(z)/u]$ belongs to a complete set of E-unifiers for $\langle h(u), u\rangle$. Interestingly, $[f(z)/u]$ can also be found using the original equations [1], [2], [3].

$$\begin{aligned}
\langle h(u), u\rangle &\Longrightarrow_{\text{para}} \langle h(u), x\rangle, \langle f(g(x)), u\rangle && \text{using [1]}\\
&\Longrightarrow_{\text{para}} \langle h(u), x\rangle, \langle g(x), g(k(f(z)))\rangle, \langle f(z), u\rangle && \text{using [3]}\\
&\Longrightarrow_{\text{vel}} \langle h(u), x\rangle, \langle g(h(u)), g(k(f(z)))\rangle, \langle f(z), u\rangle && \text{applied to } x\\
&\Longrightarrow_{\text{para}} \langle h(u), x\rangle, \langle g(h(u)), g(h(y))\rangle, \langle g(k(y)), g(k(f(z)))\rangle, \langle f(z), u\rangle && \text{using [2]}\\
&\overset{*}{\Longrightarrow}_{\text{dec}} \langle h(u), x\rangle, \langle u, y\rangle, \langle y, f(z)\rangle, \langle f(z), u\rangle &&\\
&\Longrightarrow_{\text{vel}} \langle h(u), x\rangle, \langle u, y\rangle, \langle f(z), u\rangle && \text{applied to } y\\
&\Longrightarrow_{\text{vel}} \langle h(f(z)), x\rangle, \langle f(z), y\rangle, \langle f(z), u\rangle. && \text{applied to } u
\end{aligned}$$

Thus, $[f(z)/u, h(f(z))/x, f(z)/y]$ is an E-unifier of $\langle h(u), u\rangle$.

Lemma 6.6 also provides a rigorous proof of the correctness of the transformations of Martelli, Moiso, and Rossi [31] in the case where $E = \emptyset$ and R is canonical. In fact, we have shown the more general case where R is ground Church-Rosser w.r.t. $\succ$.

7 Surreduction

In this section, an alternate proof of the completeness of the $\mathcal{T}$-transformations is established by showing that the rewrite steps occurring in a rewrite proof of $\sigma(u) \overset{*}{\longleftrightarrow}_E \sigma(v)$ can be simulated by certain generalizations of rewrite steps called surreduction steps (or narrowing steps). It should be noted that this completeness result is weaker than the completeness results given by lemma 6.6 and theorem 6.8. This point will be clarified in the next section.

Definition 7.1 Let E be a set of equations (or a rewrite system) and let W be a set of protected variables. Given any two terms u, v, we say that there is a *surreduction step* (or *narrowing step*) from u to v away from W iff there is some address β in u where u/β is not a variable, a variant $l \doteq r$ of an equation in $E \cup E^{-1}$ (or E if E is a rewrite system) such that u/β and l are unifiable and the variables in $Var(l, r)$ are *new* and occur *only* in l and r (so that $Var(l, r) \cap (Var(u) \cup W) = \emptyset$) and if $\sigma = mgu(u/\beta, l)[W]$, then $v = \sigma(u[\beta \leftarrow r])$. A surreduction step is denoted as

$$u \rightarrowtail_{[\beta, l \dot{=} r, \sigma, W]} v.$$

(some arguments may be omitted). The substitution σ is called the *surreducing substitution*. A surreduction sequence (or narrowing sequence) is defined in the obvious way. Thus, a surreduction step

$$u \rightarrowtail_{[\beta, l \dot{=} r, \sigma]} v$$

corresponds to the rewrite step

$$\sigma(u) \longrightarrow_{[\beta, l \dot{=} r, \sigma]} v.$$

The crucial lemma in proving the completeness result of this section is a version of the "lifting lemma" that establishes the precise relationship between a rewrite step $\theta(u) \longrightarrow_{[\rho(l) \dot{=} \rho(r)]} v$ and the corresponding surreduction step $u \rightarrowtail_{[l \dot{=} r]} v'$, a result of Hullot [18] shown in detail in Kirchner and Kirchner [24] in the case of canonical systems of rewrite rules. Since we are not necessarily dealing with rewrite rules ($Var(r)$ is not necessarily a subset of $Var(l)$ for an equation $l \doteq r$), we give a detailed proof of our extension of this result.

Lemma 7.2 Let E be a set of equations, R a rewrite system, $\succ$ a reduction ordering containing R, u a term, W a set of 'protected variables' containing $Var(u)$, θ a ground substitution reduced w.r.t. $R(E) \cup R$ such that $D(\theta) \subseteq W$, and $\rho(l) \rightarrow \rho(r)$ a ground rule such that either $l \rightarrow r$ is a variant of a rule in R or a variant of an equation in E such that $\rho(l) \rightarrow \rho(r) \in R(E)$, $D(\rho) = Var(l, r)$ and by the variant assumption, the variables in $Var(l, r)$ are *new* and occur only in this rule. For any ground term v, if

$$\theta(u) \longrightarrow_{[\beta, l \dot{=} r, \rho]} v,$$

for some address $\beta \in \theta(u)$, then there are two substitutions θ' and σ, a new set of protected variables W', and a term v' such that:

(1) u/β is not a variable and σ is the mgu of u/β and l away from $W \cup Var(l, r)$

(2) $v' = \sigma(u)[\beta \leftarrow \sigma(r)]$ and $\sigma(u) \longrightarrow_{[\beta, l \dot{=} r, \sigma]} v'$

(3) $\theta = \sigma \circ \theta'[W]$ and $\theta'|_{W \cup I(\sigma)}$ is reduced w.r.t. $R(E) \cup R$

(4) $v = \theta'(v')$ and

(5) $Var(v') \subseteq W'$ and $D(\theta') \subseteq W'$.

This may be illustrated as follows:

$$\begin{array}{ccc} \theta(u) & \longrightarrow_{[\beta, l \dot{=} r, \rho]} & v = \theta'(v') \\ {\scriptstyle\theta}\uparrow & & \uparrow{\scriptstyle\theta'} \\ u & \rightarrowtail_{[\beta, l \dot{=} r, \sigma, W]} & v' \end{array}$$

Proof. Obviously, $\theta(u)/\beta = \rho(l)$. Since θ is reduced w.r.t. $R(E) \cup R$, β must be the address of a nonvariable symbol in u, and $\theta(u)/\beta = \theta(u/\beta)$. Let $t = u/\beta$. Since $D(\theta) \cap D(\rho) = \emptyset$, we can form the union $\varphi = \theta \cup \rho$ of the substitutions θ and ρ, and we have $\varphi(t) = \varphi(l)$, i.e., φ is a unifier of t and l. By lemma 3.11 we have an mgu σ of t and l away from $W \cup Var(l, r)$, proving (1). Also, by corollary 3.12 there is some substitution η such that $\varphi = \theta \cup \rho = \sigma \circ \eta[W \cup Var(l)]$, where w.l.g., since σ is idempotent, we can assume that $D(\eta) \cap D(\sigma) = \emptyset$. Also note that since $Var(l)$ and $Var(u)$ are disjoint, then $D(\sigma) = Var(t) \cup Var(l)$. Let $v' = \sigma(u)[\beta \leftarrow \sigma(r)]$. Observe that the variables in v' are contained in the union of the three disjoint sets W, $I(\sigma)$, and $(Var(r) - Var(l))$. This last set is nonempty when $Var(r)$ is not a subset of $Var(l)$, which is possible when $\rho(l) \doteq \rho(r)$ is an orientable instance. We define $W' = W \cup I(\sigma) \cup (Var(r) - Var(l))$ (proving the first part of (5)), and we define the substitution θ' as follows:

$$\theta'(y) = \begin{cases} \eta(y), & \text{if } y \in W \cup I(\sigma); \\ \rho(y), & \text{if } y \in (Var(r) - Var(l)). \end{cases}$$

Clearly, the first part of (5) holds. Since $v' = \sigma(u)[\beta \leftarrow \sigma(r)]$ and $\sigma(u)/\beta = \sigma(t) = \sigma(l)$ (because σ is a unifier of t and l), we have

$$\sigma(u) \longrightarrow_{[\beta, l \dot{=} r, \sigma]} v'$$

and (2) holds. Since

$$\theta(u) \longrightarrow_{[\beta, l \dot{=} r, \rho]} v,$$

we have $v = \theta(u)[\beta \leftarrow \rho(r)]$. We now show that $v = \theta'(v')$. Since $v' = \sigma(u)[\beta \leftarrow \sigma(r)]$, we have $\theta'(v') = \theta'(\sigma(u))[\beta \leftarrow \theta'(\sigma(r))]$. Hence, we need to show that

$$\theta'(\sigma(u))[\beta \leftarrow \theta'(\sigma(r))] = \theta(u)[\beta \leftarrow \rho(r)].$$

Since $\theta \cup \rho = \sigma \circ \eta[W \cup Var(l)]$ and $\theta' = \eta[W \cup I(\sigma)]$, then by the definition of θ' and the variant assumption we have $\theta = \sigma \circ \theta'[W]$ and $\theta'(\sigma(u)) = \theta(u)$. This also shows the first

part of (3). Since $\theta \cup \rho = \sigma \circ \eta[W \cup Var(l)]$ and $\theta' = \eta[W \cup I(\sigma)]$, if $y \in Var(l) \cap Var(r)$, then $\theta'(\sigma(y)) = \rho(y)$. If $y \in Var(r) - Var(l)$, since $\theta'(y) = \rho(y)$ and $\sigma(y) = y$ (because $D(\sigma) = Var(l) \cup Var(t)$), we also have $\theta'(\sigma(y)) = \rho(y)$. Hence, $\theta'(\sigma(r)) = \rho(r)$, and we have shown that $v = \theta'(v')$. Thus, (4) holds. It remains to show the second part of (3), that $\theta'|_{W \cup I(\sigma)}$ is reduced w.r.t. $R(E) \cup R$. Recall that $\theta' = \eta[W \cup I(\sigma)]$. Thus, we show that η is reduced w.r.t. $R(E) \cup R$ on $W \cup I(\sigma)$. For any $y \in D(\eta) \cap (W \cup I(\sigma))$, there are two cases. If $y \in W$, then, since $D(\theta') \cap D(\sigma) = \emptyset$, $\sigma(y) = y$, and since $\theta \cup \rho = \sigma \circ \eta[W \cup Var(l)]$, $\eta(y) = \eta(\sigma(y)) = \theta(y)$. Since $\theta(y)$ is reduced w.r.t. $R(E) \cup R$, so is $\eta(y)$. Now by the definition of σ and by the variant assumption, we have $I(\sigma) = Var(\sigma(t))$ and $Var(\sigma(t)) \cap Var(t) = \emptyset$. Also, since $\theta \cup \rho = \sigma \circ \eta[W \cup Var(l)]$, then for every variable z in $Var(t)$, $\theta(z) = \eta(\sigma(z))$. Hence, for every $y \in I(\sigma)$, $\eta(y) = \theta(z)/\alpha$ for some $z \in Var(t)$, where α is the address of y in $\sigma(z)$. Since $\theta(z)$ is reduced w.r.t. $R(E) \cup R$, so is its subterm $\eta(y)$. Thus (3) holds, and the proof is complete. □

We now have the following result showing the crucial role played by surreductions.

Lemma 7.3 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R, and assume that (E, R) is ground Church-Rosser relative to $\succ$. Let the symbol eq be a new binary function symbol not in Σ. Given any two terms u, v, if a ground substitution θ reduced w.r.t. $R(E) \cup R$ and such that $Var(u, v) \subseteq D(\theta)$ is an (E, R)-unifier of u and v, then for any set of protected variables W containing $D(\theta)$, there is a surreduction sequence

$$eq(u, v) \ \succ\!\!\rightarrow_{[l_1 \doteq r_1, \sigma_1]} \ eq(u_1, v_1) \ldots \ \succ\!\!\rightarrow_{[l_n \doteq r_n, \sigma_n]} \ eq(u_n, v_n)$$

(where each $l_i \doteq r_i$ is a variant of an equation in $E \cup E^{-1} \cup R$) and some mgu μ of u_n and v_n such that

$$\sigma_1 \circ \ldots \circ \sigma_n \circ \mu \leq \theta[W].$$

Furthermore, the substitution $\sigma_1 \circ \ldots \circ \sigma_n \circ \mu|_{Var(u,v)}$ is an (E, R)-unifier of u and v.

Proof. Since (E, R) is ground Church-Rosser relative to $\succ$, there is a rewrite proof

$$\theta(u) \xrightarrow{*}_{R(E) \cup R} N \xleftarrow{*}_{R(E) \cup R} \theta(v),$$

where N is irreducible (w.r.t. $R(E) \cup R$). Hence, there is a rewrite proof

$$\theta(eq(u, v)) \xrightarrow{*}_{R(E) \cup R} eq(N, N),$$

where $eq(N, N)$ is irreducible. We proceed by induction on the well-founded ordering $\succ$. If $\theta(eq(u, v))$ is irreducible, obviously $eq(\theta(u), \theta(v)) = eq(N, N)$, and θ is a unifier of u and

v. The lemma is satisfied by choosing μ as a $mgu(u,v)[W]$. Otherwise, there is a rewrite proof

$$\theta(eq(u,v)) \quad \longrightarrow_{[\beta,l\dot{=}r,\rho]} \quad w \xrightarrow{*}_{R(E)\cup R} eq(N,N),$$

where $\rho(l) \to \rho(r) \in R(E)$ or $l \to r \in R$, and $\rho(l) \succ \rho(r)$. If $\rho|_{Var(r)-Var(l)}$ is not reduced, since $R(E) \cup R$ is canonical on ground terms, we can reduce each $\rho(x)$ where $x \in Var(r) - Var(l)$ to its normal form $\rho(x)\downarrow$ (w.r.t. $R(E) \cup R$), obtaining a reduced substitution ρ_1. But then, using the rule $\rho_1(l) \to \rho_1(r)$ which also satisfies $\rho_1(l) \succ \rho_1(r)$, since $\rho|_{Var(l)} = \rho_1|_{Var(l)}$ and $\rho(y) \succ \rho_1(y)$ for each $y \in Var(r) - Var(l)$, we have a rewrite proof

$$\theta(eq(u,v)) \quad \longrightarrow_{[\beta,l\dot{=}r,\rho_1]} \quad w_1 \xrightarrow{*}_{R(E)\cup R} eq(N,N).$$

Then, by lemma 7.2, we have a surreduction step away from W

$$eq(u,v) \quad \rightarrowtail_{[\beta,l\dot{=}r,\sigma_1,W]} \quad w_1',$$

substitutions σ_1 and θ_1, and $W' = W \cup I(\sigma) \cup (Var(r) - Var(l))$ such that $\theta_1(w_1') = w_1$, $\theta = \sigma_1 \circ \theta_1[W]$, $D(\theta_1), Var(w_1') \subseteq W'$, and the substitution $\theta_1|_{W\cup I(\sigma)}$ is reduced w.r.t. $R(E) \cup R$. Since $\theta_1|_{Var(r)-Var(l)} = \rho_1|_{Var(r)-Var(l)}$ and ρ_1 is reduced (w.r.t. $R(E) \cup R$), θ_1 is reduced w.r.t. $R(E) \cup R$. But w_1' is of the form $eq(u_1, v_1)$ and $w_1 = \theta_1(eq(u_1, v_1))$. Also, since $\rho_1(l) \succ \rho_1(r)$ and

$$\theta(eq(u,v)) \quad \longrightarrow_{[\beta,l\dot{=}r,\rho_1]} \quad \theta_1(eq(u_1,v_1)),$$

we have $\theta(eq(u,v)) \succ \theta_1(eq(u_1,v_1))$. Since $w_1 \xrightarrow{*}_{R(E)\cup R} eq(N,N)$, we have

$$\theta_1(eq(u_1,v_1)) \xrightarrow{*}_{R(E)\cup R} eq(N,N).$$

Hence, the induction hypothesis applies using the new set of protected vars $W' = W \cup I(\sigma) \cup (Var(r) - Var(l))$, and there is some surreduction sequence

$$eq(u_1,v_1) \rightarrowtail_{[l_2\dot{=}r_2,\sigma_2]} eq(u_2,v_2) \ldots \rightarrowtail_{[l_n\dot{=}r_n,\sigma_n]} eq(u_n,v_n)$$

and some mgu μ of u_n and v_n such that

$$\sigma_2 \circ \ldots \circ \sigma_n \circ \mu \leq \theta_1[W'].$$

Since $\theta = \sigma_1 \circ \theta_1[W]$, we have

$$\sigma_1 \circ \ldots \circ \sigma_n \circ \mu \leq \theta[W].$$

The proof that $\sigma_1 \circ \ldots \circ \sigma_n \circ \mu|_{Var(u,v)}$ is an (E,R)-unifier of u and v is routine and left to the reader. □

The previous lemma implies the following important theorem.

Theorem 7.4 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R total on ground terms. Given any two terms u, v, if θ is an (E, R)-unifier of u and v, then for any set W containing $Var(u, v)$ and $D(\theta)$ there is a surreduction sequence

$$eq(u, v) \ \succ\!\!\rightarrow_{[l_1 \doteq r_1, \sigma_1]} \ eq(u_1, v_1) \ldots \ \succ\!\!\rightarrow_{[l_n \doteq r_n, \sigma_n]} \ eq(u_n, v_n)$$

(where each $l_i \doteq r_i$ is a variant of an equation in $E^\omega \cup (E^\omega)^{-1} \cup R^\omega$) and a mgu μ of u_n and v_n such that

$$\sigma_1 \circ \ldots \circ \sigma_n \circ \mu \leq_{E \cup R} \theta[W].$$

Furthermore, $\sigma_1 \circ \ldots \circ \sigma_n \circ \mu|_{Var(u,v)}$ is an (E, R)-unifier of u and v.

Proof. First, recall that by lemma 5.10 it can be assumed that θ is ground and that $Var(u, v) \subseteq D(\theta)$ without any loss of generality. Next, we use theorem 5.7 which shows that $E^\omega \cup R^\omega$ is equivalent to (E, R) and is ground Church-Rosser relative to $\succ$. Then, by lemma 5.11, we know that there is a ground substitution θ'' reduced w.r.t. $R(E^\omega) \cup R^\omega$ and such that $\theta'' =_{E \cup R} \theta'[Var(u, v)]$. Finally, we apply Lemma 7.3 to θ'' and $R(E^\omega) \cup R^\omega$. □

It is remarkable that theorem 7.4 shows the completeness of surreduction together with the computation of critical pairs. Note that rules in R^ω can be applied oriented, whereas equations in E^ω have to be used as two-way rules. This adds considerably to the nondeterminism of the method, and shows why oriented rules are preferred. We now show how a weaker version of the completeness of our $\mathcal{T}$-transformations can be obtained from theorem 7.4.

8 Completeness of the Improved Transformations Revisited

First, we show that the $\mathcal{T}$-transformations can simulate surreduction in the case of a pair (E, R) that is ground Church-Rosser (w.r.t. $\succ$).

Lemma 8.1 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. Assume that (E, R) is ground Church-Rosser (w.r.t. $\succ$). For every surreduction sequence

$$eq(u, v) \ \succ\!\!\rightarrow_{[l_1 \doteq r_1, \sigma_1]} \ eq(u_1, v_1) \ldots \ \succ\!\!\rightarrow_{[l_n \doteq r_n, \sigma_n]} \ eq(u_n, v_n)$$

where each $l_i \doteq r_i$ is a variant of an equation in $E \cup E^{-1} \cup R$ and μ is the mgu of u_n and v_n, there is a sequence of $\mathcal{T}$-transformations $\langle u, v \rangle \stackrel{*}{\Longrightarrow} S$ yielding a solved system S such that

$$\sigma_S = \sigma_1 \circ \ldots \circ \sigma_n \circ \mu[Var(u, v)].$$

Proof. The lemma is proved by induction on the length of surreduction sequences. If $n = 0$, then u and v are unifiable by μ, and by the completeness of the transformations for standard unification (without Lazy Paramodulation), the result holds. Otherwise, since $eq(u, v) \rightarrowtail_{[\sigma_1]} eq(u_1, v_1)$, either

$$u \rightarrowtail_{[\beta, l \dot{=} r, \sigma_1]} u_1$$

for some address β in u and $v_1 = \sigma_1(v)$, or $u_1 = \sigma_1(u)$ and

$$v \rightarrowtail_{[\beta, l \dot{=} r, \sigma_1]} v_1$$

for some address β in v. We consider the first case, the other being similar. By the induction hypothesis, $\langle u_1, v_1 \rangle \stackrel{*}{\Longrightarrow} S'$ by a sequence of $\mathcal{T}$-transformations, where S' is a solved system such that

$$\sigma_{S'} = \sigma_2 \circ \ldots \circ \sigma_n \circ \mu[Var(u, v)].$$

However, since $eq(u, v) \rightarrowtail_{[\beta, l \dot{=} r, \sigma_1]} eq(u_1, v_1)$, we have

$$\langle u, v \rangle \Longrightarrow \langle u/\beta, l \rangle, \langle u[\beta \leftarrow r], v \rangle$$

by Lazy Paramodulation, and

$$\langle u/\beta, l \rangle, \langle u[\beta \leftarrow r], v \rangle \stackrel{*}{\Longrightarrow} S_1 \cup \langle \sigma_1(u[\beta \leftarrow r]), \sigma_1(v) \rangle = S_1 \cup \langle u_1, v_1 \rangle,$$

by performing the sequence of transformations from the set $\mathcal{ST}$ that computes the mgu σ_1 of u/β and l and the corresponding solved system S_1. Thus,

$$\langle u, v \rangle \stackrel{*}{\Longrightarrow} S_1 \cup \langle u_1, v_1 \rangle.$$

Since by the induction hypothesis

$$\langle u_1, v_1 \rangle \stackrel{*}{\Longrightarrow} S',$$

it is easy to see (by induction on the length of the sequence) that

$$S_1 \cup \langle u_1, v_1 \rangle \stackrel{*}{\Longrightarrow} \sigma_{S'}(S_1) \cup S',$$

and so

$$\langle u, v \rangle \stackrel{*}{\Longrightarrow} \sigma_{S'}(S_1) \cup S',$$

and letting $S = \sigma_{S'}(S_1) \cup S'$, S is in solved form. Since S_1 is the system in solved form associated with σ_1, and since the substitutions σ_i and μ have pairwise disjoint domains, we have

$$\sigma_S = \sigma_1 \circ \ldots \circ \sigma_n \circ \mu[Var(u, v)].$$

□

We can now give another proof of the completeness of the set of transformations $\mathcal{T}$ when (E, R) is ground Church-Rosser.

Lemma 8.2 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R. The set of transformations $\mathcal{T}$ is complete for all ground Church-Rosser pairs (E, R).

Proof. We need to prove that given any two terms u, v, if θ is an (E, R)-unifier of u and v, then there is a sequence of $\mathcal{T}$-transformations $\langle u, v\rangle \stackrel{*}{\Longrightarrow} S$ (using variants of equations in $E \cup E^{-1} \cup R$) yielding a solved system S such that if σ_S is the substitution associated with S, then $\sigma_S \leq_{E\cup R} \theta[Var(u, v)]$. Without loss of generality, by lemma 5.10, it can be assumed that θ is ground and that $Var(u, v) \subseteq D(\theta)$. By lemma 5.11, there is a ground substitution θ' reduced w.r.t. $R(E) \cup R$ and such that $\theta' =_{E\cup R} \theta[Var(u, v)]$. By lemma 7.3, there is a surreduction sequence

$$eq(u, v) \ \succ\!\!\rightarrow_{[l_1 \doteq r_1, \sigma_1]} \ eq(u_1, v_1) \ldots \ \succ\!\!\rightarrow_{[l_n \doteq r_n, \sigma_n]} \ eq(u_n, v_n)$$

where each $l_i \doteq r_i$ is a variant of an equation in $E \cup E^{-1} \cup R$, u_n and v_n are unifiable, and if μ is the mgu of u_n and v_n, then

$$\sigma_1 \circ \ldots \circ \sigma_n \circ \mu \leq \theta'[Var(u, v)].$$

By lemma 8.1, there is a sequence of $\mathcal{T}$-transformations $\langle u, v\rangle \stackrel{*}{\Longrightarrow} S$ yielding a solved system S such that

$$\sigma_S = \sigma_1 \circ \ldots \circ \sigma_n \circ \mu[Var(u, v)].$$

Thus,

$$\sigma_S = \sigma_1 \circ \ldots \circ \sigma_n \circ \mu \leq \theta' =_{E\cup R} \theta[Var(u, v)],$$

and so $\sigma_S \leq_{E\cup R} \theta[Var(u, v)]$. □

It is worth noting that lemma 8.2 is weaker than lemma 6.6 in the following sense. Lemma 6.6 shows the completeness of the transformations $\mathcal{T}$ even when Lazy Paramodulation is restricted to apply either at the top ($\beta = \epsilon$) or when one of u, v is a variable (but not both). However, this is not the case for lemma 8.2. The simulation of surreduction steps requires Lazy Paramodulation unrestricted. This is not very surprising. In the proof of lemma 6.6, transformations are applied in a *top-down* and lazy fashion. By lazy, we mean that unification steps can be delayed. On the other hand, it is not clear that completeness is guaranteed if such a top-down strategy is applied in a sequence of surreduction steps. However, using lemma 5.13, it can be shown that surreduction steps can always be applied bottom-up, that is, using innermost steps, and it is easy to see that lemma 8.2 still holds under this strategy. This corresponds to a *bottom-up* strategy for applying the transformations, and the proof of lemma 6.6 does not yield the completeness of this strategy. Thus, it

appears that lemma 6.6 and lemma 8.2 correspond to different strategies for applying the transformations, and that they are complementary.

In a recent paper, Nutt, Réty, and Smolka [32] investigate complete sets of transformations for basic narrowing applied to ground confluent systems. It would be interesting to explore the relationship between our set of transformations $\mathcal{T}$ and the transformations presented in [32].

Finally, we give an alternate proof of the completeness of the $\mathcal{T}$-transformations in the general case. The above comments also apply to this theorem and to theorem 6.8.

Theorem 8.3 Let E be a set of equations, R a rewrite system, and $\succ$ a reduction ordering containing R total on ground terms. The set $\mathcal{T}$ is a complete set of transformations.

Proof. Without loss of generality, we can assume that θ is ground and that $Var(u, v) \subseteq D(\theta)$. By theorem 5.7, $E^\omega \cup R^\omega$ is equivalent to (E, R) and is ground Church-Rosser relative to $\succ$. Then, by lemma 8.2, there is a sequence of $\mathcal{T}$-transformations $\langle u, v\rangle \stackrel{*}{\Longrightarrow} S$ using equations in $E^\omega \cup (E^\omega)^{-1} \cup R^\omega$ yielding a solved system S such that $\sigma_S \leq_{E\cup R} \theta[Var(u, v)]$, where σ_S is the substitution associated with S. We conclude by applying lemma 6.7. □

9 Previous Work

Since the work of Plotkin [34], most of the energy of researchers in this field has been directed either toward (i) isolating and investigating the E-unification problem in specific theories such as commutativity, associativity, etc., and various combinations of such specific axioms, and (ii) investigating the E-unification problem in the presence of canonical rewrite systems. There has been some work as well on various extensions to the latter.

The first area of research will not concern us here, since we are interested only in more general forms of E-unification. The second area represents the most general form of E-unification which has been thoroughly investigated to date (but see also [14]).

Narrowing was first presented in [39] and [28], but the E-unification algorithm based on this technique first appeared in [8] and was refined by [18]. (A good presentation of the important results concerning the algorithm can be found in [24].) Since then the basic method has been developed by various researchers [22, 20, 19, 32,35]. Narrowing and its refinements represent a very clean and elegant solution to an important subclass of E-unification problems, and we do not claim to have improved upon these results. Instead we view our research as an attempt to place these results in a more general context, by showing in a very abstract way how the same proof techniques used in narrowing may be applied to our more general problem. We should in particular note that Martelli, Moiso,

and Rossi have presented an E-unification procedure using a set of transformations much the same as our set $\mathcal{T}$, but they attempted to prove completeness only in the context of canonical systems.

The work of Kirchner [22] attempts to extend the basic paradigm of E-unification in canonical theories by adapting the approach of Martelli and Montanari [30] to standard unification which uses the operations of merging and decomposition over multiequations to find mgu's in ordered form; by respecting the ordering of variable dependencies among the various terms, one may avoid explicit application of substitutions, and so Variable Elimination is not used. Kirchner expands this basic method by defining conditions under which decomposition may be done in the presence of equations, and by defining a new operation on multiequations, called mutation, which is dependent upon the theory under consideration. He extends the procedure for canonical theories by showing that if a theory permits the use of variable dependency orderings to avoid explicit substitution (such a theory is termed *strict*), and if a mutation operation can be deduced, then his procedure returns a complete set of E-unifiers. He then gives a general strategy for deriving the mutation operation via a critical pair computation, and hence a way of automating the creation of specialized E-unification procedures. As an example this strategy is applied to the class of *syntactic* theories, which basically allow complete sets of E-unifiers to be found by allowing at most one rewrite at the root between any two terms. Our approach to E-unification owes much to Kirchner's initial inspiration to adapt the method of transformations to E-unification, but our motivations are very different. We have used only the abstract notion of transformations on term systems, and not the technique of multiequations. Our research concerns not the derivation of specific procedures, but the abstract analysis of the general case. It is not surprizing, then, that we can subsume the methods of Kirchner in an abstract way. We could optimize our procedure for syntactic theories, for example, by simply allowing at most one root rewrite between any two terms. As in the case of narrowing, however, our general procedure is not likely to be as suitable for specific theories as specially designed procedures, although in an absolute sense it subsumes them.

Another form of more general E-unification has been investigated by Holldobler [14]. This is the problem of E-unification in the presence of a confluent set of rewrite rules. Holldobler's approach for showing the completeness of the transformations is to use the refutational completeness of SLD-resolution, an interesting idea. Given a confluent set R of rewrite rules, one views R as a set H_R of clauses of the form $eq(l, r) \leftarrow$ for every $l \to r \in R$, and adds to H_R the set H_E of equality axioms (for the set of function symbols in R) written as clauses. For example, there is a clause

$$eq(f(x_1, \ldots, x_n), f(y_1, \ldots, y_n)) \leftarrow eq(x_1, y_1), \ldots, eq(x_n, y_n)$$

for every function symbol f of rank n occurring in R (a congruence axiom). The pair $\langle u, v \rangle$

to be R-unified is converted to the goal clause $\leftarrow eq(u, v)$. It is easy to show that θ is an R-unifier of $\langle u, v\rangle$ iff there is some SLD-refutation for the logic program $\{\leftarrow eq(u, v)\} \cup H_R \cup H_E$ returning a substitution answer σ such that $\sigma \leq \theta$. Then, Holldobler shows how his transformations can simulate such SLD-refutations. However, it appears that in his completeness proof, the fact that a subgoal of the form $\leftarrow eq(f(x'_1, \ldots, x'_n), f(y'_1, \ldots, y'_n))$ could have been generated and that this subgoal will unify with the head of the equality axiom $eq(f(x_1, \ldots, x_n), f(y_1, \ldots, y_n)) \leftarrow eq(x_1, y_1), \ldots, eq(x_n, y_n)$ yielding the new subgoal $\leftarrow eq(x'_1, y'_1), \ldots, eq(x'_n, y'_n)$ seems to have been overlooked. This is a problem because a literal of the form $eq(x_1, y_1)$ will unify with the head of a congruence axiom, or with any rewrite rule from R (clauses $eq(l, r) \leftarrow$ where $l \rightarrow r \in R$). Thus, the proof does not prevent rewriting steps from being performed at or below variable positions. This is the same problem that we face with the system $\mathcal{BT}$, and solve in the later sections of our paper. Actually, we believe that using confluence alone is too weak, and that ground confluence with respect to some reduction ordering $\succ$ is needed if the transformations are to be applied oriented, as they are in Holldobler's paper.

In general, our approach to E-unification, although heavily indebted to many researchers in this field, is fundamentally different. Whereas the previous work in this field has concentrated on elucidating the structure of specific E-unification problems or in gradually expanding the class of theories for which complete E-unification procedures exist, our research has concentrated on finding a very general method for which a rigorous completeness proof was available, and then attempting to find techniques to prove the completeness of restricted versions of this method.

10 Eager Variable Elimination

We discuss in this section the primary open problem to be solved in our research on general E-unification. Notice that in our general discussion of E-unification in section 4, we prove the completeness of the method via a strategy which applies transformation (C) *only* to trivial proofs $(x = t)$ in which no rewrite steps occur. If the proof $(x * t)$ contains rewrite steps, we use transformation (D) or (E). This corresponds in the transformations on systems to non-deterministically allowing a pair $\langle x, t\rangle$ where $x \notin Var(t)$ to be transformed by either Variable Elimination, Root Rewriting, or Root Imitation in the set $\mathcal{BT}$ or, alternately, by either Variable Elimination or Lazy Paramodulation in the set $\mathcal{T}$. The strategy of *Eager Variable Elimination* is to always apply Variable Elimination to a pair (if possible) instead of Root Rewriting or Root Imitation (or Lazy Paramodulation in the case of $\mathcal{T}$). In other words, we never look for rewrites below the root of a pair $\langle x, t\rangle$ if $x \notin Var(t)$, and can immediately eliminate x via Variable Elimination. The question of whether such a set of

transformations is complete is still open.

In fact, our original formulation of E-unification via transformations used this strategy, but a difficulty arose in finding a measure on which to base our completeness proof. The problem is that—no matter what formalism is used for E-unification proofs—performing Variable Elimination on a pair which needs rewrite steps between $\theta(x)$ and $\theta(t)$ will have to incorporate these steps into the proof wherever x is replaced by t. The effect is that the same equation may end up being duplicated many times. Then, if variables are renamed in duplicated equations to avoid clashes, potentially not only the number of rewrite steps in the new system is increased, but also the number of unsolved variables; but if duplicate equations are not renamed, it must be ensured that no variable clashes will ever occur in any later sequence of transformations.

Actually, the notion of an equational proof tree was developed to clarify these issues, but we were not able to prove the correctness or termination of this new set of proof transformations, and so were led to the approach presented above in Chapter 5 to find useful restrictions on our transformations.

The literature has mostly overlooked this problem, and, as it is deceptively simple at first glance, it is generally assumed to be true. Martelli et al. [31] claim the completeness of such a strategy in the context of canonical rewrite systems. However, because their proof lacks many details, including a measure for a rigorous induction, we are unable to check the validity of their argument about Variable Elimination. Holldobler [14] claims the completeness of a set of transformations equivalent to our system $\mathcal{BT}$ with Eager Variable Elimination. As remarked above, his proof contains a gap, and no rigorous analysis of Variable Elimination is presented. Using the techniques developed in section 5, we believe that Holldobler's completeness proof can be partially patched, but we do not believe that the transformations are complete if Eager Variable Elimination is performed. We should remark that Kirchner has avoided this whole problem by examining only those theories in which Variable Elimination can be avoided by the use of variable dependency orderings.

Remark. (Added July 1988) At the Unification Workshop in Val D'Ajol, June 1988, Jieh Hsiang and Jean Pierre Jouannaud claimed to have found a proof of the completeness of eager variable elimination. They suggested that it is possible to give a bound on the number of new variables created in any sequence of transformations to account for new variants of equations. At the time of submission of this paper, we had not yet seen the details of this proof.

11 Conclusion

Although research in E-unification has grown tremendously in the past 15 years, for some reason the problem of general E-unification in arbitrary theories has been neglected. This is unfortunate, since progress in any area of science is often frustrated when fundamental issues of the basic paradigm are not well understood. In this paper we attempted to provide a rigorous paradigm for the study of complete procedures for general E-unification by adopting the method of transformations on systems of terms and showing how a basic set $\mathcal{BT}$ of very general transformations for E-unification corresponds to certain transformations on equational proof trees. In this context, the completeness of our method is easily shown, and highlights a number of features, such as the problem with Eager Variable Elimination discussed above, which are not obvious in completeness proofs using other techniques. In order to make this method efficient enough to be implemented, we then showed how restrictions may be placed on this basic set to obtain a set $\mathcal{T}$, thereby increasing its efficiency while retaining completeness for arbitrary equational theories. The method of proof here was adapted from unfailing completion, and showed that we need not ever rewrite at variable occurrences, which not only eliminates the guessing of functional reflexivity axioms and the potential for infinite recursion on Root Imitation, but also prunes out a large number of useless rewrite sequences. In addition, we showed how other more general forms of E-unification, such as narrowing, can be simulated by our method, by demonstrating that the set of $\mathcal{T}$-transformations is complete for a set R of ground Church-Rosser rewrite rules, and also that the strategy of surreduction plus the simulation of critical pair computation is complete.

In conclusion, it is our hope that this research, in addition to providing a theoretical foundation both for the study of complete methods of E-unification in the general case (or in various classes of theories), and for the study of equality in logic programming, will provide a unifying connection between the diverse approaches to E-unification currently being developed and the larger concerns of the proof theory of first order logic.

Acknowledgment: We thank the referees for some very helpful comments. We also wish to thank Leo Bachmair, Nachum Dershowitz, Jean Pierre Jouannaud, Claude Kirchner, Jieh Hsiang, and David Plaisted for many helpful discussions on the topics of E-unification and unfailing completion. Finally, special thanks to Stan Raatz for much helpful discussion concerning both the general content of this paper and its presentation.

12 References

[1] Bachmair, L., *Proof Methods for Equational Theories*, Ph.D thesis, University of Illinois, Urbana Champaign, Illinois (1987).

[2] Bachmair, L., Dershowitz, N., and Plaisted, D., "Completion without Failure," Proceedings of CREAS, Lakeway, Texas (May 1987), also submitted for publication.

[3] Bachmair, L., Dershowitz, N., and Hsiang, J., "Orderings for Equational Proofs," In *Proc. Symp. Logic in Computer Science*, Boston, Mass. (1986) 346-357.

[4] Bürckert, H., "Lazy E-Unification - A Method to Delay Alternative Solutions," Workshop on Unification, Val D'Ajol, France (1987).

[5] Bürckert, H., "Some relationships between unification, restricted unification and matching," In J. Siekmann, editor, *Proceedings 8th Conference on Automated Deduction, Oxford*, Springer Verlag, Oxford (England) (1986) 514-524.

[6] Dershowitz, N,. "Termination of Rewriting," Journal of Symbolic Computation 3 (1987) 69-116.

[7] Fages, F. and Huet, G., "Complete Sets of Unifiers and Matchers in Equational Theories," TCS 43 (1986) 189-200.

[8] Fay, M., "First-order Unification in an Equational Theory," Proceedings of the 4^{th} Workshop on Automated Deduction, Austin, Texas (1979).

[9] Gallier, J.H. *Logic for Computer Science: Foundations of Automatic Theorem Proving*, Harper and Row, New York (1986).

[10] Gallier, J.H., Raatz, S., "Extending SLD-Resolution to Equational Horn Clauses Using E-unification", to appear in *Journal of Logic Programming* (1987).

[11] Gallier, J.H., Snyder, W., "A General Complete E-Unification Procedure," Proceedings of the RTA, Bordeaux, France (1987).

[12] Goguen, J.A. and Meseguer, J., "Eqlog: Equality, Types, and Generic Modules for Logic Programming," in *Functional and Logic Programming*, Degroot, D. and Lindstrom, G., eds., Prentice-Hall (1985). Short version in Journal of Logic Programming 2 (1984) 179-210.

[13] Herbrand, J., "Sur la Théorie de la Démonstration," in *Logical Writings*, W. Goldfarb, ed., Cambridge (1971).

[14] Holldobler, S., "A Unification Algorithm for Confluent Theories," ICALP (1987).

[15] Huet, G., *Résolution d'Equations dans les Langages d'Ordre* $1, 2, \ldots, \omega$, Thèse d'Etat, Université de Paris VII (1976)

[16] Huet, G., "Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems," JACM 27:4 (1980) 797-821.

[17] Huet, G. and Oppen, D. C., "Equations and Rewrite Rules: A Survey," in *Formal Languages: Perspectives and Open Problems*, R.V.Book, ed., Academic Press, New York (1982).

[18] Hullot, J.-M., "Canonical Forms and Unification," CADE-5 (1980) 318-334.

[19] Hussmann, H., "Unification in Conditional Equational Theories," Proceedings of the EUROCAL 1985, Springer Lecture Notes in Computer Science 204, p. 543-553.

[20] Kaplan, S., "Fair Conditional Term Rewriting Systems: Unification, Termination, and Confluence," Technical Report 194, Universite de Paris-Sud, Centre D'Orsay, Laboratoire de Recherche en Informatique (1984).

[21] Kirchner, C., "A New Equational Unification Method: A Generalization of Martelli-Montanari's Algorithm," CADE-7, Napa Valley (1984).

[22] Kirchner, C., *Méthodes et Outils de Conception Systematique d'Algorithmes d'Unification dans les Theories Equationnelles*, Thèse d'Etat, Université de Nancy I (1985).

[23] Kirchner, C., "Computing Unification Algorithms," LICS'86, Cambridge, Mass. (1986).

[24] Kirchner, C. and Kirchner, H., *Contribution à la Resolution d'Equations dans les Algèbres Libres et les Varietés Equationnelles d'Algèbres*, Thèse de 3^e cycle, Université de Nancy I (1982).

[25] Knuth, D.E. and Bendix, P.B., "Simple Word Problems in Univeral Algebras," in *Computational Problems in Abstract Algebra*, Leech, J., ed., Pergamon Press (1970).

[26] Kozen, D., "Complexity of Finitely Presented Algebras," Technical Report TR 76-294, Department of Computer Science, Cornell University, Ithaca, New York (1976).

[27] Kozen, D., "Positive First-Order Logic is NP-Complete," IBM Journal of Research and Development, 25:4 (1981) 327-332.

[28] Lankford, D.S., "Canonical Inference," Report ATP-32, University of Texas (1975),

[29] Levy, A., *Basic Set Theory*, Springer-Verlag, New York (1979).

[30] Martelli, A., Montanari, U., "An Efficient Unification Algorithm," ACM Transactions on Programming Languages and Systems, 4:2 (1982) 258-282.

[31] Martelli, A., Moiso, C., Rossi, G.F., "An Algorithm for Unification in Equational Theories," Third IEEE Symposium on Logic Programming, Salt Lake City, Utah, (1986).

[32] Nutt, W., Réty, P., and Smolka, G., "Basic Narrowing Revisited," Technical Report SR-87-07, Kaiserslautern University, W. Germany (1987).

[33] Paterson, M.S., Wegman, M.N., "Linear Unification," Journal of Computer and System Sciences, 16 (1978) 158-167.

[34] Plotkin, G., "Building in Equational Theories," Machine Intelligence 7 (1972) 73-90.

[35] Rety, P., "Improving Basic Narrowing Techniques," Proceedings of the RTA, Bordeaux, France (1987).

[36] Robinson, J.A., "A Machine Oriented Logic Based on the Resolution Principle," JACM 12 (1965) 23 -41.

[37] Robinson, J.A., "A Review on Automatic Theorem Proving," Annual Symposia in Applied Mathematics, 1-18 (1967).

[38] Siekmann, J. H., "Universal Unification," CADE-7, Napa Valley (1984).

[39] Slagle, J.R., "Automated Theorem Proving for Theories with Simplifiers, Commutativity, and Associativity," JACM 21 (1974) 622-642.