Goal-Directed E-Unification

Christopher Lynch and Barbara Morawska

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

Abstract. We give a general goal directed method for solving the E-unification problem. Our inference system is a generalization of the inference rules for Syntactic Theories, except that our inference system is proved complete for any equational theory. We also show how to easily modify our inference system into a more restricted inference system for Syntactic Theories, and show that our completeness techniques prove completeness there also.

1 Introduction

E-unification [1] is a problem that arises in several areas of computer science, including automated deduction, formal verification and type inference. The problem is, given an equational theory E and a goal equation $u \approx v$, to find the set of all substitutions θ such that $u\theta$ and $v\theta$ are identical modulo E. In practice, it is not necessary to find all such substitution. We only need to find a set from which all such substitutions can be generated, called a *complete set of E-unifiers.*

The decision version of E-unification (Does an E-unifier exist?) is an undecidable problem, even for the simpler *word problem* which asks if all substitutions θ will make $u\theta$ and $v\theta$ equivalent modulo E. However there are procedures which are *complete* for the problem. Complete, in this sense, means that each E-unifier will be generated eventually. However, because of the undecidability, the procedure may continue to search for an E-unifier forever, when no E-unifier exists.

One of the most successful methods for solving the E-unification problem has been Knuth-Bendix Completion[8]. This procedure deduces new equalities from E. If the procedure ever halts, then it can solve the word problem. However, because of the undecidability, Knuth-Bendix Completion cannot always halt.

Our goal in this paper is to develop an alternative E-unification procedure. Why do we want an alternative to Knuth-Bendix Completion? There are several reasons. First, there are simple equational theories for which Completion does not halt. An example is the equational theory $E = \{f(f(x)) \approx g(f(x))\}$. So then it is impossible to decide any word problem in this theory, even a simple example like $a \approx b$, which is obviously not true. Using our method, examples like this will quickly halt and say there is no solution.

[**] This work was supported by NSF grant number CCR-9712388 .

A related deficiency of Completion is that it is difficult to identify classes of equational theories where the procedure halts, and to analyze the complexity of solving those classes. That is our main motivation for this line of research. We do not pursue that subject in this paper, since we first need to develop a complete inference system. That subject will be addressed soon in a follow-up paper.

Another aspect of Completion is that it is insensitive to the goal. It is possible to develop heuristics based on the goal, but problems like the example above still exist, because of the insensitivity to the goal. The method we develop in this paper is goal directed, in the sense that every inference step is a step backwards from the goal, breaking the given goal into separate subgoals. Therefore we call our method a goal directed inference system for equational reasoning. This quality of goal-directedness is especially important when combining an equational inference system with another inference system. Most of the higher order inference systems used for formal verification have been goal directed inference systems. Even most inference systems for first order logic, like OTTER, are often run with a set of support strategy. For things like formal verification, we need equality inference systems that can be added as submodules of previously existing inference systems. We believe that the best method for achieving this is to have a goal directed equality inference system.

We do not claim that our procedure is the first goal directed equational inference system. Our inference system is similar to the inference system Syntactic Mutation first developed by Claude Kirchner [4, 6]. That inference system applies to a special class of equational theories called Syntactic Theories. In such theories, any true equation has an equational proof with at most one step at the root. The problem of determining if an equational theory is syntactic is undecidable[7]. In the Syntactic Mutation inference system, it is possible to determine which inference rule to apply next by looking at the root symbols on the two sides of a goal equation. This restricts which inference rules can be applied at each point, and makes the inference system more efficient than a blind search.

Our inference system applies to every equational theory, rather than just Syntactic Theories. Therefore, we cannot examine the root symbol at both sides of a goal equation. However, we do prove that we may examine the root symbol of one side of an equation to decide which inference rule to apply. Other than that, our inference system is similar to Syntactic Mutation. We prove that our inference system is complete. The Syntactic Mutation rules were never proved to be complete. In [5], it is stated that there is a problem proving completeness because the Variable Elimination rule (called "Replacement" there) does not preserve the form of the proof. We think we effectively deal with that problem.

There is still an open problem of whether the Variable Elimination rule can be applied eagerly. We have not solved that problem. But we have avoided those problems as much as possible. The inefficiency of the procedure comes from cases where one side of a goal equation is a variable. We prove that any equation where both sides are variables may be ignored without losing completeness. We also orient equations so that inference rules are applied to the nonvariable side of an equation. This gives some of the advantages of Eager Variable Elimination.

Another goal directed equational inference procedure is the General Unification Procedure developed by Gallier and Snyder[2, 3]. Their method differs in that the inference rules do not apply to the root of the terms in the goal equation. It can apply underneath, and the unification is performed in a lazy way. We think our method is easier to implement and may be more efficient. We also think that in our method it will be simpler to find decidable classes of equational theories. The Eager Variable Elimination problem was first presented in this setting, and it has not been solved there either.

The format of the paper is to first give some preliminary definitions. Then present our inference system. After a discussion of normal form, we present soundness results. In order to prove completeness, we first give a bottom-up method for deducing ground equations, then use that method to prove completeness of our goal-directed method. After that we show how our completeness technique can be applied to Syntactic Theories to show completeness of a procedure similar to Syntactic Mutation. Finally, we conclude the paper.

2 Preliminaries

We assume we are given a set of variables and a set of uninterpreted function symbols of various arities. An arity is a non-negative integer. *Terms* are defined recursively in the following way: each variable is a term, and if $t_1, \cdots, t_n$ are terms, and f is of arity $n \geq 0$, then $f(t_1, \cdots, t_n)$ is a term, and f is the symbol at the *root* of $f(t_1, \cdots, t_n)$. A term (or any object) without variables is called *ground*. We consider equations of the form $s \approx t$, where s and t are terms. Let E be a set of equations, and $u \approx v$ be an equation, then we write $E \models u \approx v$ (or $u =_E v$) if $u \approx v$ is true in any model containing E. If G is a set of equations, then $E \models G$ means that $E \models e$ for all e in G.

A *substitution* is a mapping from the set of variables to the set of terms, such that it is almost everywhere the identity. We identity a substitution with its homomorphic extension. If θ is a substitution then $Dom(\theta) = \{x \mid x\theta \neq x\}$. A substitution θ is an *E-unifier* of an equation $u \approx v$ if $E \models u\theta \approx v\theta$. θ is an *E-unifier* of a set of equations G if θ is an E-unifier of all equations in G.

If σ and θ are substitutions, then we write $\sigma \leq_E \theta[Var(G)]$ if there is a substitution ρ such that $E \models x\sigma\rho \approx x\theta$ for all x appearing in G. If G is a set of equations, then a substitution θ is a *most general unifier of* G, written $\theta = mgu(G)$ if θ is an E unifier of G, and for all E unifiers σ of G, $\theta \leq_E G\theta[VAr(G)]$. A complete set of E-unifiers of G, is a set of E-unifiers Θ of G such that for all E-unifiers σ of G, there is a θ in Θ such that $\theta \leq_E \sigma[Var(G)]$.

3 The Goal Directed Inference Rules

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

We define two special kinds of equations appearing in the goal G. An equation of the form $x \approx y$ where x and y are both variables is called a *variable-variable* equation. An equation $x \approx t$ appearing in G where x only appears once in G is called *solved*.

As in Logic Programming, we can have a selection rule for goals. For each goal G, we don't-care nondeterministically select an equation $u \approx v$ from G, such that $u \approx v$ is not a variable-variable equation and $u \approx v$ is not solved. We say that $u \approx v$ is *selected in* G. If there is no such equation $u \approx v$ in the goal, then nothing is selected. We will prove that if nothing is selected, then the goal is in normal form and a most general E unifier can be easily determined.

There is a Decomposition rule.

Decomposition

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

where $f(s_1,\cdots,s_n) \approx f(t_1,\cdots,t_n)$ is selected in the goal.

This is just an application of the Congruence Axiom, in a goal-directed way. If f is of arity 0 (a constant) then this is a goal-directed application of Reflexivity.

We additionally add a second inference rule that is applied when one side of an equation is a variable.

Variable Decomposition

$$\frac{\{x \approx f(t_1,\cdots,t_n)\} \cup G}{\{x \approx f(x_1,\cdots,x_n)\} \cup (\{x_1 \approx t_1,\cdots,x_n \approx t_n\} \cup G)[x \mapsto f(x_1,\cdots,x_n)]}$$

where x is a variable, and $x \approx f(t_1,\cdots,t_n)$ is selected in the goal.

This is similar to the Variable Elimination rule for syntactic equalities. It can be considered a gradual form of Variable Elimination, since it is done one step at a time.

Now we add a rule called Mutate. We call it Mutate, because it is very similar to the inference rule Mutate that is used in the inference procedure for syntactic theories. Mutate is a kind of goal-directed application of Transitivity, but only transitivity steps involving equations from the theory.

Mutate

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

where $u \approx f(v_1,\cdots,v_n)$ is selected in the goal, and $s \approx f(t_1,\cdots,t_n) \in E$. [1]

This rule assumes that there is an equational proof of the goal equation at the root of the equation. If one of the equations in this proof is $s \approx t$ then that breaks up the proof at the root into two separate parts. We have performed a

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

Decomposition on one of the two equations that is created. Contrast this with the procedure for Syntactic Theories[4] which allows a Decomposition on both of the newly created equations. However, that procedure only works for Syntactic Theories, whereas our procedure is complete for any equational theory.

Next we give a Mutate rule for the case when one side of the equation from E is a variable.

Variable Mutate

$$\frac{\{u \approx f(v_1, \cdots, v_n)\} \cup G}{\{u \approx s\}[x \mapsto f(x_1, \cdots, x_n)] \cup \{x_1 \approx v_1, \cdots, x_n \approx v_n\} \cup G}$$

where $s \approx x \in E$, x is a variable, and $u \approx f(v_1, \cdots, v_n)$ is selected in the goal.

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

When an inference is performed, we may eagerly reorient any new equations in the goal. The way they are reoriented is don't-care nondeterministic, except that any equation of the form $t \approx x$, where t is not a variable and x is a variable, must be reoriented to $x \approx t$.

We will prove that the above inference rules solve a goal G by transforming it into normal forms representing a complete set of E-unifiers of G. There are two sources of non-determinism involved in the procedure defined by the inference rules. The first is "don't-care" non-determinism in deciding which equation to select, and in deciding which way to orient equations with non-variable terms on both sides. The second is "don't-know" non-determinism in deciding which rule to apply. Not all paths of inference steps will lead us to the normal form, and we do not know beforehand which ones do.

4 Normal Form

Notice that there are no inference rules that apply to an equation $x \approx y$, where x and y are both variables. In fact, such an equation can never be selected. The reason is that so many inferences could possibly apply to variable-variable pairs that we have designed the system to avoid them. That changes the usual definition of normal form, and shows that inferences with variable-variable pairs are unnecessary.

Let G be a goal of the form $\{x_1 \approx t_1, \cdots, x_n \approx t_n, y_1 \approx z_1, \cdots, y_m \approx z_m\}$, where all x_i, y_i and z_i are variables, the t_i are not variables, and for all i and j,

1. $x_i \notin t_j$,
2. $x_i \neq y_j$ and
3. $x_i \neq z_j$.

Then G is said to be in *normal form.* Let τ_G be the most general (syntactic) unifier of $y_1 = z_1, \cdots, y_m = z_m$ [2] and σ_G be the substitution $[x_1 \mapsto t_1, \cdots x_n \mapsto t_n]$. Then we will define θ_G to be the substitution $\sigma_G \tau_G$.

[2] Note that an mgu must exist, since, $y_1 = z_1, \cdots, y_m = z_m$ are unifiable. Also note that an mgu can be easily calculated using the syntactic unification procedure.

Proposition 1. *A goal with nothing selected is in normal form.*

Proof. Let G be a goal with nothing selected. Then all equations in G have a variable on the left hand side. So G is of the form $x_1 \approx t_1, \cdots, x_n \approx t_n, y_1 \approx z_1, \cdots, y_m \approx z_m$. Since nothing is selected, each equation $x_1 \approx t_1$ must be solved. So each x_i appears only once in G. Therefore the three conditions of normal form are satisfied. □

Now we will prove that the substitution represented by a goal in normal form is a most general E-unifier of that goal.

Lemma 1. *Let G be a set of equations in normal form. Then θ_G is a most general E-unifier of G.*

Proof. Let G be the goal $\{x_1 \approx t_1, \cdots, x_n \approx t_n, y_1 \approx z_1, \cdots, y_m \approx z_m\}$, such that for all i and j, $x_i \notin t_j$, $x_i \neq y_j$ and $x_i \neq z_j$. Let $\sigma_G = [x_1 \mapsto t_1, \cdots, x_n \mapsto t_n]$. Let $\tau_G = mgu(y_1 = z_1, \cdots, y_m = z_m)$. Let $\theta_G = \sigma_G \tau_G$. We will prove that θ_G is a most general E unifier of G.

Let i and j be integers such that $1 \leq i \leq n$ and $1 \leq j \leq n$. First we need to show that θ_G is a unifier of G, i.e. that $x_i\theta_G = t_i\theta_G$ and $y_j\theta_G =_E z_j\theta_G$. In other words, prove that $x_i\sigma_G\tau_G = t_i\sigma_G\tau_G$ and $y_j\sigma_G\tau_G =_E z_j\sigma_G\tau_G$. Since t_i, y_j and z_j are not in the domain of σ, this is equivalent to $t_i\tau_G = t_i\tau_G$ and $y_j\tau_G =_E z_j\tau_G$, which is trivially true, since τ_G is mgu of $\{y_1 \approx z_1, \cdots, y_m \approx z_m\}$.

Next we need to show that θ_G is more general than all other unifiers of G. So let θ be an E-unifier of G. In other words, $x_i\theta =_E t_i\theta$ and $y_j\theta =_E z_j\theta$. We need to show that $\theta_G \leq_E \theta[Var(G)]$. In particular, we will show that $G\theta_G\theta =_E G\theta$.

Then $x_i\theta_G\theta = x_i\sigma_G\tau_G\theta = t_i\tau_G\theta =_E t_i\theta =_E x_i\theta$. The only step that needs justification is the fact that $t_i\tau_G\theta =_E t_i\theta$. This can be verified by examining the variables of t_i. So let w be a variable in t_i. If $w \notin Dom(\tau_G)$ then obviously $w\tau_G\theta = w\theta$. If $w \in Dom(\tau_G)$ then w is some y_k. Note that $y_k\tau_G\theta = z_k\theta =_E y_k\theta$. So $t_i\tau_G\theta =_E t_i\theta$.

Also, $y_j\theta_G\theta = y_j\sigma_G\tau_G\theta = y_j\tau_G\theta = z_j\theta =_E y_j\theta$. Similaarly $z_j\theta_G\theta = z_j\sigma_G\tau_G\theta = z_j\tau_G\theta = z_j\theta$. □

5 An Example

Here is an example of the procedure. (The selected equations are underlined.)

Example 1. Let $E = E_0 = \{ffx \approx gfx\}$, $G = G_0 = \{\underline{fgfy \approx ggfz}\}$.
By rule Mutate applied to G_0 we have
$G_1 = \{\underline{fgfy \approx ffx_1}, fx_1 \approx gfz\}$.
After Decomposition,
$G_2 = \{\underline{gfy \approx fx_1}, fx_1 \approx gfz\}$.
After Mutate,
$G_3 = \{\underline{gfy \approx gfx_2}, x_1 \approx fx_2, fx_1 \approx gfz\}$
After Decomposition is used 2 times on G_3,

$G_4 = \{y \approx x_2, \underline{x_1 \approx fx_2}, fx_1 \approx gfz\}$.
Variable Decomposition:
$G_5 = \{y \approx x_2, x_1 \approx fx_3, x_3 \approx x_2, \underline{ffx_3 \approx gfz}\}$.
Mutate:
$G_6 = \{y \approx x_2, x_1 \approx fx_3, x_3 \approx x_2, \underline{ffx_3 \approx ffx_4}, fx_4 \approx fz\}$.
$2\times$ Decomposition:
$G_7 = \{y \approx x_2, x_1 \approx fx_3, x_3 \approx x_2, x_3 \approx x_4, \underline{fx_4 \approx fz}\}$.
Decomposition:
$G_8 = \{y \approx x_2, x_1 \approx fx_3, x_3 \approx x_2, x_3 \approx x_4, x_4 \approx z\}$.

The extended θ' that unifies the goal G_0 is equal to: $[x_1 \mapsto fx_3][y \mapsto z, x_3 \mapsto z, x_2 \mapsto z, x_4 \mapsto z]$. θ' is equivalent on the variables of G to θ equal to: $[y \mapsto z]$.

Example 2. Let $E = \{ffx \approx gfx\}$, $G = G_0 = \{\underline{fgfa \approx ggfa}\}$.
By rule Mutate applied to G_0 we have
$G_1 = \{\underline{fgfa \approx ffx_1}, fx_1 \approx gfa\}$.
After Decomposition,
$G_2 = \{\underline{gfa \approx fx_1}, fx_1 \approx gfa\}$.
After Mutate,
$G_3 = \{\underline{gfa \approx gfx_2}, x_1 \approx fx_2, fx_1 \approx gfa\}$
After Decomposition is used 2 times on G_3,
$G_5 = \{\underline{x_2 \approx a}, x_1 \approx fx_2, fx_1 \approx gfa\}$.
Variable Decomposition:
$G_6 = \{x_2 \approx a, \underline{x_1 \approx fa}, fx_1 \approx gfa\}$.
Variable Decomposition:
$G_7 = \{x_2 \approx a, x_1 \approx fx_3, \underline{x_3 \approx a}, ffx_3 \approx gfa\}$.
Variable Decomposition:
$G_8 = \{x_2 \approx a, x_1 \approx fa, x_3 \approx a, \underline{ffa \approx gfa}\}$.
Mutate:
$G_9 = \{x_2 \approx a, x_1 \approx fa, x_3 \approx a, \underline{ffa \approx ffx_4}, fx_4 \approx fa\}$.
$2\times$ Decomposition:
$G_{11} = \{x_2 \approx a, x_1 \approx fa, x_3 \approx a, \underline{x_4 \approx a}, fx_4 \approx fa\}$.
Variable Decomposition:
$G_{12} = \{x_2 \approx a, x_1 \approx fa, x_3 \approx a, x_4 \approx a, \underline{fa \approx fa}\}$.
$2\times$ decomposition deletes the last equation:
$G_{14} = \{x_2 \approx a, x_1 \approx fa, x_3 \approx a, x_4 \approx a\}$.

Here is another example, when the equational theory E is not regular. It also ilustrates the use of Variable Mutate rule.

Example 3. Let $E_0 = \{x \approx gfx\}$ and $G_0 = \{\underline{x_1 \approx fgx_1}\}$.
By Variable Decomposition:
$G_1 \approx \{x_1 \approx fx_2, \underline{x_2 \approx gfx_2}\}$.
By Variable Mutate:
$G_2 \approx \{x_1 \approx fx_2, x_2 \approx x_3, \underline{fx_3 \approx fx_2}\}$.
By Decomposition:
$G_3 \approx \{x_1 \approx fx_2, x_2 \approx x_3, x_3 \approx x_2\}$.

6 Soundness

Theorem 1. *The above procedure is sound, i.e. if* $G' \xrightarrow{*} G$ *and* G *is in normal form, then* $E \models G'\theta_G$.

Proof. Assume that G' is in normal form. Then $\theta_{G'}$ unifies equations in G', as shown in Lemma 1.

Now assume that $E \models G_{i+1}\theta_G$, and prove that $E \models G_{i+1}\theta_G$.

Case 1. G_{i+1} was obtained by Decomposition from G_i. We know that $E \models G_{i+1}\theta_G$, but then since E must be closed under congruence, $E \models G_i\theta_G$.

Case 2. $G_{i+1} = \{x \approx f(t_1, \cdots, t_n\} \cup H$ and was obtained from G_i by Variable Decomposition. First we prove that $[x \mapsto f(x_1, \cdots, x_n)]\theta_G$ and θ_G, are E-equivalent. We justify this claim by considering a variable y If $y \neq x$, then $y[x \mapsto f(x_1, \cdots, x_n)]\theta_G = y\theta_G$. If $y = x$, then $y[x \mapsto f(x_1, \cdots, x_n)]\theta_G = f(x_1, \cdots, x_n)\theta_G =_E x\theta_G$, since $E \models x\theta_G \approx f(x_1, \cdots, x_n)\theta_G$. Therefore $E \models H\theta_G$, since $E \models H[x \mapsto f(x_1, \cdots, x_n)\theta_G$. We also need to show that $E \models x\theta_G \approx f(t_1, \cdots, t_n)\theta_G$. This is true, since $x\theta_G =_E f(x_1, \cdots, x_n)\theta_G =_E f(x_1 \cdots, x_n)[x \mapsto f(x_1, \cdots, x_n)]\theta_G =_E f(t_1 \cdots, t_n)[x \mapsto f(x_1, \cdots, x_n)]\theta_G =_E f(t_1 \cdots, t_n)\theta_G$.

Case 3. G_{i+1} was obtained from G_i by Mutate. In this case, $E \models u\theta_G \approx s\theta_G$ and $s \approx f(t_1, \cdots, t_n) \in E$. So, $E \models u\theta_G \approx f(t_1, \cdots, t_n)\theta_G$. We assume that $E \models t_i\theta_G \approx v_i\theta_G$, for all $i \in \{1, \cdots, n\}$ and thus by congruence, $E \models f(t_1, \cdots, t_n)\theta_G \approx f(v_1, \cdots, v_n)\theta_G$, hence by transitivity, $E \models u\theta_G \approx f(v_1, \cdots, v_n)\theta_G$.

Case 4. G_{i+1} was obtained from G_i by rule Variable Mutate. We know that $E \models (u \approx s)[x \mapsto f(x_1, \cdots, x_n)]\theta_G$. On the other hand $s \approx x$ belongs to the axiom schemas of E, and hence $E \models (s \approx x)[x \mapsto f(x_1, \cdots, x_n)]\theta_G$, i.e. $E \models (s \approx f(x_1, \cdots, x_n))[x \mapsto f(x_1, \cdots, x_n)]\theta_G$, thus by transitivity, $E \models (u \approx f(x_1, \cdots, x_n))[x \mapsto f(x_1, \cdots, x_n)]\theta_G$. This is equivalent to $E \models (u \approx f(x_1, \cdots, x_n))\theta_G$, because there is no x in u nor in $f(x_1, \cdots, x_n)$. We know that $E \models x_i \approx v_i$ for all $i \in \{1, \cdots, n\}$. Hence $E \models f(x_1, \cdots, x_n) \approx f(v_1, \cdots, v_n)$, therefore by transitivity $E \models (u \approx f(v_1, \cdots, v_n))\theta_G$.

□

7 A Bottom Up Inference System

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

We will define completeness with respect to any equational theory obtained by the following rules of inference from a set of equations closed under symmetry:

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

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

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

We define $E \vdash u \approx v$ if there is a proof of $u \approx v$ using the Congruence and Equation Application rules. If π is a proof, then $|\pi|$ is the number of steps in the proof. $|u \approx v|_E$ is the number of steps in the shortest proof of $u \approx v$.

We need to prove that $\{u \approx v \,|\, E \vdash u \approx v\}$ is closed under Reflexivity, Symmetry and Transitivity. First we prove Reflexivity.

Lemma 2. *Let E be an equational theory. Then $E \vdash u \approx u$ for all u.*

Proof. The prove is by induction on the size of u. If the size of $u \geq 1$, then $u = f(s_1 \cdots s_n)$ and we assume that for each $j \in \{1, 2, \cdots, n\}$ $\exists k_j.|s_j \approx s_j|_E = k_j$. Then by applying Congruence we have $|u \approx u|_E = \Sigma_{i=1}^{n} k_i + 1$. If the size of $u = 1$, we apply Congruence over the empty set of assumptions and get the desired reflexivity, $|u \approx u|_E = 1$. □

Next we prove closure under symmetry.

Lemma 3. *Let E be an equational theory such that $E \vdash u \approx v$ and $|u \approx v|_E = n$. Then $E \vdash v \approx u$, and $|v \approx u|_E = n$.*

Proof. The argument is by induction on $|u \approx v|_E$. There are 2 subcases, depending on whether $u \approx v$ was obtained by Congruence or Equation Application:

1. $u \approx v$ was obtained by Congruence, i.e. $u = f(s_1, \cdots, s_n)$ and $v = f(t_1, \cdots, t_n)$, and $|s_1 \approx t_1|_E + \cdots + |s_n \approx t_n|_E = i - 1$. But then, by the inductive argument $|t_1 \approx s_1|_E + \cdots + |t_n \approx s_n|_E = i - 1$ also, and by Congruence $|f(t_1, \cdots, t_n) \approx f(s_1, \cdots, s_n)|_E = i$, i.e. $|v \approx u|_E = i$.
2. $u \approx v$ was obtained by Equation Application, i.e. there is $s \approx t \in E$ and $|u \approx s|_E + |t \approx v|_E = i - 1$. By induction also $|v \approx t|_E + |s \approx u|_E = i - 1$. Because E is closed under symmetry, by Equation Application we have $|v \approx u|_E = i$. □

Next we show closure under Transitivity.

Lemma 4. *Let E be an equational theory such that $E \vdash s \approx t$ and $E \vdash t \approx u$. Suppose that $|s \approx t|_E = m$ and $|t \approx u|_E = n$. Then $E \vdash s \approx u$, and $|s \approx u|_E \leq m + n$.*

Proof. The proof will be by induction on $m + n$, where m and n are the sizes of the derivations for the assumptions of the desired transitivity step. We shall divide the proof into cases:

1. Assume that both equations were obtained by Congruence. Then $s \approx f(s_1, \cdots, s_n)$, $t \approx f(t_1, \cdots, t_n)$ and $u \approx f(u_1, \cdots, u_n)$. Then there are equations, such that: $|s_1 \approx t_1|_E + ... + |s_k \approx t_k|_E = m - 1$, and $|t_1 \approx u_1|_E + ... + |t_k \approx u_k|_E = n - 1$.

$$\dfrac{\dfrac{\overset{\vdots}{s_1 \approx t_1} \quad \cdots \quad \overset{\vdots}{s_k \approx t_k}}{f(s_1,\cdots,s_k)\approx f(t_1,\cdots,t_k)}\,\text{Congr.} \qquad \dfrac{\overset{\vdots}{t_1\approx u_1} \quad\cdots\quad \overset{\vdots}{t_k\approx u_k}}{f(t_1,\cdots,t_k)\approx f(u_1,\cdots,u_k)}\,\text{Congr.}}{f(s_1,\cdots,s_k)\approx f(u_1,\cdots,u_k)}\,\text{Trans.}$$

By induction, $|s_i \approx u_i|_E \leq |s_i \approx t_i|_E + |t_i \approx u_i|_E$ for each $i \in \{1,\cdots,k\}$. Hence, $|s_1 \approx u_1|_E + \cdots + |s_k \approx u_k|_E \leq (m-1)+(n-1)$. By Congruence we have $|f(s_1,\cdots,s_n) \approx f(u_1,\cdots u_n)|_E \leq m+n-1$.

2. Assume that one of the equations was not obtained by Congruence, e.g. the first one. Then it had to appear due to the rule Equation Application.

$$\dfrac{\dfrac{\overset{\vdots}{s \approx v} \qquad \overset{\vdots}{w \approx t}}{s \approx t}\,\text{Eq.App.} \qquad \overset{\vdots}{t \approx u}}{s \approx u}\,\text{Trans.}$$

where $v \approx w$ is a ground instance of an equation in E and the final equation is the desired effect of transitivity. Now we can remove this transitivity step by moving it up in the derivation and then apply the inductive hypothesis.

$$\dfrac{\overset{\vdots}{s \approx v} \qquad \dfrac{\overset{\vdots}{w \approx t} \qquad \overset{\vdots}{t \approx u}}{w \approx u}\,\text{Trans.}}{s \approx u}\,\text{Eq.App.}$$

If originally transitivity occurred at the step $|s \approx t|_E + |t \approx u|_E$ in the derivation, now transitivity occurs at the step $|w \approx t|_E + |t \approx u|_E$, which is smaller, hence we can apply the inductive hypothesis.

□

Closure under Congruence is trivial. Now we put these lemmas together to show that anything true under the semantic definition of Equality is also true under the syntactic definition given here.

Theorem 2. *If $E \models u \approx v$, then $E \vdash u \approx v$.*

Proof. If $s \approx t \in E$, then by Lemma 2, $E \vdash s \approx s$ and $E \vdash t \approx t$. Applying Equational Application says that $E \vdash s \approx t$. Since we proved that $\{u \approx v \mid E \vdash u \approx v\}$ is closed under reflexivity, symmetry, transitivity and congruence, it must contain all the consequences of E. □

We can restrict our proofs to only certain kinds of proofs. In particular. If the root step of a proof is an Equation Application, then we can show there is a proof such that the proof step of the right child is not an Equation Application.

Lemma 5. *Let π be a proof of $u \approx v$ in E, whose proof step at the root is Equation Application, and whose proof step of the right child is also Equation Application. Then there is a proof π' of $u \approx v$ in E such that the root of π' is Equation Application but the proof step of the right child is Congruence, and $|\pi'| = |\pi|$.*

Proof. Let π be a proof of $u \approx v$ in E such that the step at the top is Equation Application, and the step at the right child is also Equation Application. We will show that there is another proof π' of $u \approx v$ in E such that $|\pi'| = |\pi|$, and the size of the right subtree of π' is smaller than the size of the right subtree of π. So this proof is an induction on the size of the right subtree of the proof.

Suppose $u \approx v$ is at the root of π and $u \approx s$ labels the left child n_1. Suppose the right child n_2 is labeled with $t \approx v$. Further suppose that the left child of n_2 is labeled with $t \approx w_1$ and the right child of n_2 is labeled with $w_2 \approx v$. Then $s \approx t$ and $w_1 \approx w_2$ must be members of E.

$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ n_1\colon u \approx s\end{array} \qquad \dfrac{\begin{array}{c}\pi_2\\ \vdots\\ t \approx w_1\end{array} \qquad \begin{array}{c}\pi_3\\ \vdots\\ w_2 \approx v\end{array}}{n_2\colon t \approx v}\ \text{Eq. App.}}{u \approx v}\ \text{Eq. App.}$$

Then we can let π' be the proof whose root is labeled with $u \approx v$, whose left child n_3 is labeled with $u \approx w_1$. Let the left child of n_3 be labeled with $u \approx s$ and the right child of n_3 be labeled with $t \approx w_1$. Also let the right child of the root n_4 be labeled with $w_2 \approx v$.

$$\text{Eq. App.}\ \dfrac{\text{Eq. App.}\ \dfrac{\begin{array}{c}\pi_1\\ \vdots\\ u \approx s\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ t \approx w_1\end{array}}{n_3\colon u \approx w_1} \qquad \begin{array}{c}\pi_3\\ \vdots\\ n_4\colon w_1 \approx v\end{array}}{u \approx v}$$

□

8 Completeness of the Goal-Directed Inference System

Now we finally get to the main theorem of this paper, which is the completeness of the inference rules given in section 3. But first we need to define a measure on the equations in the goal.

Definition 1. *Let E be an equational theory and G be a goal. Let θ be a substitution such that $E \models G\theta$. We will define a measure μ, parameterized by θ and G. Define $\mu(G, \theta)$ as the multiset $\{|u\theta \approx v\theta|_E \mid u \approx v$ is an unsolved equation in $G\}$.*

The intension of the definition is that the measure of an equation in a goal is the number of steps it takes to prove that equation. However, solved equations are ignored.

Now, finally, the completeness theorem:

Theorem 3. *Suppose that E is an equational theory, G is a set of goal equations, and θ is a ground substitution. If $E \models G\theta$ then there exists a goal H such that $G \xrightarrow{*} H$ and $\theta_H \leq_E \theta[Var(G)]$.*

Proof. Let G be a set of goal equations, and θ a ground substitution such that $E \models G\theta$. Let $\mu(< G, \theta >) = M$. We will prove by induction on M that there exists a goal H such that $G \xrightarrow{*} H$ and $\theta_H \leq_E \theta[Var(G)]$.

If nothing is selected in G, then G must be in normal form, by Proposition 1. By Lemma 1, θ_G is the most general unifier of G, so $\theta_G \leq_E \theta[Var(G)]$.

If some equation is selected in G, we will prove that there is a goal G' and a substitution θ' such that $G \longrightarrow G'$, $\theta' \leq_E \theta[Var(G)]$, and $\mu(G', \theta') \leq \mu(G, \theta)$.

So assume that some equation $u \approx v$ is selected in G. Then G is of the form $\{u \approx v\} \cup G_1$. We assume that v is not a variable, because any term-variable equation $t \approx x$ is immediately reoriented to $x \approx t$. By Lemma 3, $|v\theta \approx u\theta|_E = |u\theta \approx v\theta|_E$. Also, according to our selection rule, a variable-variable equation is never selected. Since v is not a variable, it is in the form $f(v_1, \cdots, v_n)$. Let $|u\theta \approx v\theta|_E = m$.

Consider the rule used at the root of the proof tree that $E \vdash u\theta \approx v\theta$. This was either an application of Congruence or Equation Application.

Case 1: Suppose the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an Equation Application. Then there exists a ground instance $s\theta' \approx t\theta'$ of an equation $s \approx t$ in E, such that $E \vdash u\theta' \approx s\theta'$ and $E \vdash t\theta' \approx v\theta'$, where θ is an extension of θ such that $u\theta' = u\theta$ and $v\theta' = v\theta$. Let $|u\theta' \approx s\theta'|_E = p$. Let $|t\theta' \approx v\theta'|_E = q$. Then $m = p + q + 1$. We now consider two subcases, depending on whether or not t is a variable.

Case 1A: Suppose that t is a not a variable. Then, by Lemma 5, we can assume that the rule at the root of the proof tree of $E \vdash t\theta' \approx v\theta'$, is Congruence. So then t is of the form $f(t_1, \cdots, t_n)$, and the previous nodes of the proof tree are labeled with $t_1\theta' \approx v_1\theta', \cdots, t_n\theta' \approx v_n\theta'$. And, for each i, $|t_i\theta' \approx v_i\theta'|_E = q_i$ such that $1 + \Sigma_{1 \leq i \leq n} q_i = q$.

The proof tree of $E \vdash u\theta' \approx v\theta'$ in this case:

$$\dfrac{\begin{matrix}\vdots \\ u\theta' \approx s\theta'\end{matrix} \qquad \dfrac{\begin{matrix}\vdots \\ t_1\theta' \approx v_1\theta'\end{matrix} \quad \cdots \quad \begin{matrix}\vdots \\ t_n\theta' \approx v_n\theta'\end{matrix}}{f(t_1, \cdots, t_n)\theta' \approx f(v_1, \cdots, v_n)\theta'}\ \text{Congr.}}{u\theta' \approx f(v_1, \cdots, v_n)\theta'}\ \text{Eq. App.}$$

Therefore, there is an application of Mutate that can be applied to $u \approx v$, resulting in the new goal $G' = \{u \approx s, t_1 \approx v_1, \cdots, t_n \approx v_n\} \cup G_1$. Then $|u\theta \approx s\theta|_E = p$, and $|t_i\theta \approx v_i\theta|_E = q_i$ for all i, so $\mu(G'\theta') < \mu(G, \theta)$.

$$\text{Mutate} \quad \frac{u \approx f(v_1, \cdots, v_n)}{u \approx s, \quad t_1 \approx v_1, \quad \cdots, \quad t_n \approx v_n}$$

By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $\theta_H \leq_E \theta'[Var(G')]$. This implies that $G \xrightarrow{*} H$. Also, $\theta_H \leq_E G\theta'[Var(G)]$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $\theta_H \leq_E \theta[Var(G)]$.

Case 1B: Suppose that t is a variable. Then, by Lemma 5, we can assume that the rule at the root of the proof tree of $E \vdash t\theta' \approx v\theta'$ is Congruence. So then $t\theta'$ is of the form $f(t_1, \cdots, t_n)$, and the previous nodes of the proof tree are labeled with $t_1 \approx v_1\theta', \cdots, t_n \approx v_n\theta'$. And, for each i, $|t_i \approx v_i\theta'|_E = q_i$ such that $1 + \Sigma_{1 \leq i \leq n} q_i = q$.

$$\cfrac{\begin{matrix}\vdots \\ u\theta' \approx s\theta'\end{matrix} \qquad \cfrac{\begin{matrix}\vdots & & \vdots \\ t_1 \approx v_1\theta' & \cdots & t_n \approx v_n\theta'\end{matrix}}{f(t_1, \cdots, t_n) \approx f(v_1, \cdots, v_n)\theta'}\ \text{Congr.}}{u\theta' \approx f(v_1, \cdots, v_n)\theta'}\ \text{Eq. App.}$$

Therefore, there is an application of Variable Mutate that can be applied to $u \approx v$, resulting in the new goal $G' = \{u \approx s[t \mapsto f(x_1, \cdots, x_n)], x_1 \approx v_1, \cdots, x_n \approx v_n\} \cup G_1\}$. We will extend θ' so that $x_i\theta' = t_i$ for all i. Then $|u\theta' \approx s\theta'|_E = p$, and $|x_i\theta' \approx v_i\theta'|_E = q_i$ for all i, so $\mu(G', \theta') < \mu(G, \theta)$.

$$\text{Var. Mut.} \quad \frac{u \approx f(v_1, \cdots, v_n)}{u \approx s[t \mapsto f(x_1, \cdots, x_n)], \quad x_1 \approx v_1, \quad \cdots, \quad x_n \approx v_n}$$

where $s \approx t \in E$ and t is a variable.

By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $\theta_H \leq_E \theta'[Var(G')]$. This implies that $G \xrightarrow{*} H$. Also, $\theta_H \leq_E \theta'[Var(G)]$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $\theta_H \leq_E \theta[Var(G)]$.

Case 2 Now suppose that the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an application of Congruence. There are two cases here: u is a variable or u is not a variable.

Case 2A First we will consider the case where u is not a variable. Then $u = f(u_1, \cdots, u_n)$, $v = f(v_1, \cdots, v_n)$ and $E \vdash u_i\theta \approx v_i\theta$ for all i.
Then $u = f(u_1, \cdots, u_n)$, $v = f(v_1, \cdots, v_n)$ and $E \vdash u_i\theta \approx v_i\theta$ for all i.

$$\text{Congr.} \quad \frac{u_1\theta \approx v_1\theta, \quad \cdots \quad , u_n\theta \approx v_n\theta}{f(u_1, \cdots, u_n)\theta \approx f(v_1, \cdots, v_n)\theta}$$

There is an application of Decomposition that can be applied to $u \approx v$, resulting in the new goal $G' = \{u_1 \approx v_1, \cdots, u_n \approx v_n\} \cup G_1$. Then $|u_i\theta \approx v_i\theta|_E < |u\theta \approx v\theta|$ for all i, so $\mu(G', \theta) < \mu(G, \theta)$.

$$\text{Decomp.} \quad \frac{f(u_1, \cdots, u_n) \approx f(v_1, \cdots, v_n), G_1}{u_1 \approx v_1, \quad \cdots, \quad u_n \approx v_n, \quad G_1}$$

By the induction assumption there is an H such that $G' \stackrel{*}{\longrightarrow} H$ with $\theta_H \leq_E \theta[Var(G')]$. This implies that $G \stackrel{*}{\longrightarrow} H$ and $\theta_H \leq_E \theta[Var(G)]$.

Case 2B: Now we consider the final case, where u is a variable and the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an application of Congruence. Let $u\theta = f(u_1, \cdots, u_n)$. Then, for each i, $E \models u_i \approx v_i\theta$, and $|u_i \approx v_i\theta|_E < |u\theta \approx v\theta|_E$.

$$\text{Congr.} \quad \frac{u_1 \approx v_1\theta, \quad \cdots \quad , u_n \approx v_n\theta}{f(u_1, \cdots, u_n) \approx f(v_1, \cdots, v_n)\theta}$$

There is an application of Variable Decomposition that can be applied to $u \approx v$, resulting in the new goal $G' = \{u \approx f(x_1, \cdots, x_n)\} \cup (\{x_1 \approx v_1, \cdots, x_n \approx v_n\} \cup G_1)[u \mapsto f(x_1, \cdots, x_n)]$.

$$\frac{u \approx f(v_1, \cdots, v_n), G_1}{u \approx f(x_1, \cdots, x_n), \quad (x_1 \approx v_1, \cdots, x_n \approx v_n, G_1)[u \mapsto f(x_1, \cdots, x_n)]}$$

Let θ' be the substitution $\theta \cup [x_1 \mapsto u_1, \cdots, x_n \mapsto u_n]$. Then $u \approx f(x_1, \cdots, x_n)$ is solved in G'. Also $|x_i\theta' \approx v_i\theta'|_E < |u\theta \approx v\theta|_E$ for all i. Therefore $\mu(G, \theta) < \mu(G', \theta')$. By the induction assumption there is an H such that $G' \stackrel{*}{\longrightarrow} H$ with $\theta_H \leq_E \theta'[Var(G')]$. This implies that $G \stackrel{*}{\longrightarrow} H$. Also, $\theta_H \leq_E \theta'[Var(G)]$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $\theta_H \leq_E \theta[Var(G)]$.

□

9 *E*-Unification for Syntactic Theories

In this section we will show how we can restrict our inference rules further to get a set of inference rules that resembles the Syntactic Mutation rules of Kirchner. Then we prove that that set of inference rules is complete for syntactic theories.

The definition of a syntactic theory is in terms of equational proofs. The definition of a proof is as follows.

Definition 2. *An* equational proof of $u \approx v$ from E *is a sequence* $u_0 \approx u_1 \approx u_w \approx \cdots \approx u_n$, *for* $n \geq 0$ *such that* $u_0 = u$, $u_n = v$ *and for all* $i \geq 0$, $u_i = u_i[s]$ *and* $u_{i+1} = u_i[t]$ *for some* $s \approx t \in E$.

Now we give Kirchner's definition of *syntactic theory.*

Definition 3. *An equational theory* E *is* resolvent *if every equation* $u \approx v$ *with* $E \models u \approx v$ *has an equational proof such that there is at most one step at the root. A theory is* syntactic *if it has an equivalent resolvent presentation.*

From now on, when we discuss a Syntactic Theory E, we will assume that E is the resolvent presentation of that theory.

In this paper, we are considering bottom-up proofs instead of equational replacement proofs. We will call a bottom-up proof *resolvent* if whenever an equation appears as a result of Equation Application, then its left and right children must have appeared as a result of an application of Congruence at the root. We will call E *bottom-up resolvent* if every ground equation $u \approx v$ implied by E has a bottom-up resolvent proof. Now we show that the definition of resolvent for equational proofs is equivalent to the definition of resolvent for bottom-up proofs.

Theorem 4. *E is a resolvent presentation of an equational theory E if and only if E is a bottom-up resolvent presentation of E.*

Proof. We need to show how to transform a resolvent equational proof into a resolvent bottom-up proof and vice versa.

Case 1: First consider transforming a resolvent equational proof into a resolvent bottom-up proof. We will prove this can be done by induction on the number of symbols appearing in the equation.

Case 1A: Suppose $u \approx v$ has an equational proof with no steps at the root. Then $u \approx v$ is of the form $f(u_1, \cdots, u_n) \approx f(v_1, \cdots, v_n)$, and there are equational proofs of $u_i \approx v_i$ for all i. Since each equation $u_i \approx v_i$ has fewer symbols than $u \approx v$, then, by the induction argument there is a resolvent bottom-up proof of each $u_i \approx v_i$, and by adding one more congruence step to all the $u_i \approx v_i$, we get a resolvent bottom-up proof of $u \approx v$.

Case 1B: Now suppose $u \approx v$ has an equational proof with one step at the root. Then there is some ground instance of $s \approx t$ in E such that the proof of $u \approx v$ is a proof of $u \approx s$ with no steps at the top, followed by a replacement of s with t, followed by a proof of $t \approx v$ with no steps at the root. By induction, each child in the proof of $u \approx s$ has a resolvent bottom-up proof. Therefore $u \approx s$ has a resolvent bottom-up proof with a Congruence step at the root. Similarly, $t \approx v$ has a resolvent bottom-up proof with a Congruence step at the root. If we apply Equation Application to those two proofs, we get a bottom-up resolvent proof of $u \approx v$.

Case 2: Now we will transform a resolvent bottom-up proof of $u \approx v$ to an equational proof of $u \approx v$, by induction on $|u \approx v|_E$.

Case 2A: Suppose $u \approx v$ has a bottom-up resolvent proof with an application of Congruence at the root. Then $u \approx v$ is of the form $f(u_1, \cdots, u_n) \approx f(v_1, \cdots, v_n)$, and there are bottom-up resolvent proofs of $u_i \approx v_i$ for all i. Since each equational proof of $u_i \approx v_i$ is shorter than the proof of $u \approx v$, then, by the induction argument there is a resolvent equational proof of each $u_i \approx v_i$, and they can be combined to give a resolvent equational proof of $u \approx v$.

Case 2B: Now suppose $u \approx v$ has a resolvent bottom-up proof with one Equation Application step at the root. Then there is some $s \approx t$ in E such that the proof of $u \approx v$ is a proof of $u \approx s$ with a Congruence step at the root, and a proof of $t \approx v$ with a Congruence step at the root. Then an Equation Application using the equation $s \approx t$ from E. By induction, the corresponding equalities of subterms of $u \approx s$ have resolvent equational proofs. So $u \approx s$ has a resolvent equational proof with no steps at the root. Similarly, $t \approx v$

also has a resolvent equational proof with no steps at the root. So $u \approx v$ has a resolvent equational proof with one step at the root.

□

Now we give the inference rules for solving E-unification problems in Syntactic Theories. The rules for Decomposition and Variable Decomposition remain the same, but Mutate becomes more restrictive. We replace Mutate and Variable Mutate with one rule that covers several cases.

Mutate

$$\frac{\{u \approx v\} \cup G}{\{Dec(u \approx s), Dec(v \approx t)\} \cup G}$$

where $u \approx v$ is selected in the goal, $s \approx t \in E$, if both u and s are not variables then they have the same root symbol, and if both v and t are not variables then they have the same root symbol. We also introduce a function Dec, which when applied to an equation indicates that the equation should be decomposed further according to the following rules:

$$\frac{\{Dec(f(u_1, \cdots, u_n) \approx f(s_1, \cdots, s_n))\} \cup G}{\{u_1 \approx s_1, \cdots, u_n \approx s_n\} \cup G}$$

$$\frac{\{Dec(x \approx f(s_1, \cdots, s_n)\} \cup G}{\{x \approx f(s_1, \cdots, s_n)\} \cup G[x \mapsto f(s_1, \cdots, s_n)]}$$

$$\frac{\{Dec(x \approx y)\} \cup G}{\{x \approx y\} \cup G}$$

$$\frac{\{Dec(f(s_1, \cdots, s_n) \approx x)\} \cup G}{G[x \mapsto f(s_1, \cdots, s_n)]}$$

Now we prove a completeness theorem for this new set of inference rules, which is Decomposition, Variable Decomposition, and the Mutate rule given above.

Theorem 5. *Suppose that E is a resolvent presentation of an equational theory, G is a set of goal equations, and θ is a ground substitution. If $E \models G\theta$ then there exists a goal H such that $G \xrightarrow{*} H$ and $\theta_H \leq_E \theta[Var(G)]$.*

Proof. The proof is the same as the proof of Theorem 3, except for Case 1. In this case, we can show that one of the cases of the Mutate rules from this section is applicable. Here, instead of using Lemma 5 to say that an Equation Application must have a Congruence as a right child, we instead use the definition of bottom-resolvent to say that an Equation Application has a Congruence as both children.

If nothing is selected, i.e. G is in normal form, then θ_G is the most general unifier of G, therefore $\theta_G \leq_E \theta[Var(G)]$.

Assume that $u \approx v$ is selected in G. We can assume as before that v is not a variable (because otherwise it would be at once oriented, or if u is also a variable, the equation could not have been selected at all). Let $|u\theta \approx v\theta|_E = m$.

Case 1: Suppose the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an Equation Application. Then there exists a ground instance $s\theta' \approx t\theta'$ of an equation $s \approx t$ in E, such that $E \vdash u\theta' \approx s\theta'$ and $E \vdash t\theta' \approx v\theta'$, where θ is an extension of θ such that $u\theta' = u\theta$ and $v\theta' = v\theta$. Let $|u\theta \approx s\theta|_E = p$. Let $|t\theta \approx v\theta|_E = q$. Then $m = p + q + 1$. We know also that $E \vdash u\theta' \approx s\theta'$ and $E \vdash t\theta' \approx v\theta'$, were both obtained by Congruence (because we are considering only resolvent ground proofs). We now consider two subcases, depending on whether or not t is a variable.

Case 1A: Suppose that u, s, t are not variables. So then u is of the form $f(u_1, \cdots, u_n)$, s is of the form $f(s_1, \cdots, s_n)$ and t is of the form $f(t_1, \cdots, t_n)$, and the fragment of the derivation looks like the following:

$$\dfrac{\dfrac{u_i\theta' \approx s_i\theta'}{f(u_1,\cdots,u_n)\theta' \approx f(s_1,\cdots,s_n)\theta'} \qquad \dfrac{t_i\theta' \approx v_i\theta'}{f(t_1,\cdots,t_n)\theta' \approx f(v_1,\cdots,v_n)\theta'}}{f(u_1,\cdots,u_n)\theta' \approx f(v_1,\cdots,v_n)\theta'}$$

Therefore there is an application of Mutate resulting in new goal $G' = \{u_1 \approx s_1, \cdots u_n \approx s_n, t_1 \approx v_1, \cdots, t_n \approx v_n\} \cup G_1$.

$$\dfrac{\dfrac{f(u_1,\cdots,u_n) \approx f(v_1,\cdots,v_n) \quad (f(s_1,\cdots,s_n) \approx f(t_1,\cdots,t_n) \in E)}{\{Dec(f(u_1,\cdots,u_n) \approx f(s_1,\cdots,s_n)), Dec(f(v_1,\cdots,v_n) \approx f(t_1,\cdots,t_n))\} \cup G}}{\{u_1 \approx s_1, \cdots u_n \approx s_n, t_1 \approx v_1, \cdots, t_n \approx v_n\} \cup G}$$

Then $\Sigma_{1 \le i \le n} |u_i\theta' \approx s_i\theta'|_E = p$ and $\Sigma_{1 \le i \le n} |t_i\theta' \approx v_i\theta'|_E = q$, so $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \le_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \le_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \le G\theta$.

Case 1B: Suppose t is a variable, and u and s are as in the previous case. The root of the proof tree of $E \vdash t\theta' \approx v\theta'$ is Congruence. So then $t\theta'$ is of the form $f(t_1, \cdots, t_n)$.

$$\dfrac{\dfrac{u_i\theta' \approx s_i\theta'}{f(u_1,\cdots,u_n)\theta' \approx f(s_1,\cdots,s_n)\theta'} \qquad \dfrac{t_i \approx v_i\theta'}{f(t_1,\cdots,t_n) \approx f(v_1,\cdots,v_n)\theta'}}{f(u_1,\cdots,u_n)\theta' \approx f(v_1,\cdots,v_n)\theta'}$$

There is an application of Mutate resulting in the new goal, $G' = \{(u_1 \approx s_1, \cdots, u_n \approx s_n)[t \mapsto f(v_1, \cdots, v_n)]\} \cup G$.

$$\dfrac{\dfrac{f(u_1,\cdots,u_n) \approx f(v_1,\cdots,v_n) \quad (f(s_1,\cdots,s_n) \approx t \in E)}{\{Dec(f(u_1,\cdots,u_n) \approx f(s_1,\cdots,s_n)), Dec(f(v_1,\cdots,v_n) \approx t)\} \cup G}}{(\{u_1 \approx s_1, \cdots u_n \approx s_n\} \cup G)[t \mapsto f(v_1,\cdots,v_n)]}$$

For each $u_i \approx s_i$, $|(u_i \approx s_i)[t \mapsto f(v_1, \cdots, v_n)]\theta'|_E$ is equal to $|u_i\theta' \approx s_i\theta'|_E$, because $t = f(v_1, \cdots, v_n)\theta'$ and thus $[t \mapsto f(v_1, \cdots, v_n)]\theta'$ is the same as θ'. Hence, $\Sigma_{1 \le i \le n} |u_i\theta' \approx s_i[t \mapsto f(v_1, \cdots, v_n)]\theta'|_E = p - 1$, so $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption there is an H such that $G' \xrightarrow{*} H$

with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 1C: Assume s is a variable and u, t are not variables. This case is symmetrical to the previous one.

Case 1D: Assume that u is variable and s, t are not. Then $u\theta' \approx f(u_1, \cdots, u_n)$, and the situation in the ground derivation is as in the following diagram:

$$\cfrac{\cfrac{u_i \approx s_i\theta'}{f(u_1,\cdots,u_n) \approx f(s_1,\cdots,s_n)\theta'} \qquad \cfrac{t_i\theta' \approx v_i\theta'}{f(t_1,\cdots,t_n)\theta' \approx f(v_1,\cdots,v_n)\theta'}}{f(u_1,\cdots,u_n) \approx f(v_1,\cdots,v_n)\theta'}$$

Then there is an application of Mutate to $u \approx v$ resulting in the goal $G' = \{u \approx f(s_1, \cdots, s_n)\} \cup (\{v_1 \approx t_1, \cdots, v_n \approx t_n\} \cup G)[u \mapsto f(s_1, \cdots, s_n)]$.

$$\cfrac{\cfrac{u \approx f(v_1,\cdots,v_n) \quad (f(s_1,\cdots,s_n) \approx f(t_1,\cdots,t_n) \in E)}{\{Dec(u \approx f(s_1,\cdots,s_n)), Dec(f(v_1,\cdots,v_n) \approx f(t_1,\cdots,t_n))\} \cup G}}{\{u \approx f(s_1,\cdots,s_n)\} \cup (\{v_1 \approx t_1,\cdots,v_n \approx t_n\} \cup G)[u \mapsto f(s_1,\cdots,s_n)]}$$

Since $u\theta' = f(s_1, \cdots, s_n)\theta'$, the substitution $[u \mapsto f(s_1, \cdots, s_n)]$ does not change anything in the ground proof. We do not count p into the measure any more, because the equation is solved and $|(v_i \approx t_i)[u \mapsto f(s_1, \cdots, s_n)]\theta'|_E = |(v_i \approx t_i)\theta'|_E$, i.e. $\Sigma_{1 \leq i \leq n}|(v_i \approx t_i)\theta'|_E = q - 1$. Hence $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 1E: Now let us assume that u and s are variables and t is not. Then application of Mutation will result with the new goal $G' = \{u \approx s, v_1 \approx t_1, \cdots, v_n \approx t_n\} \cup G$.

$$\cfrac{\cfrac{u \approx f(v_1,\cdots,v_n) \quad (s \approx f(t_1,\cdots,t_n) \in E)}{\{Dec(u \approx s), Dec(f(v_1,\cdots,v_n) \approx f(t_1,\cdots,t_n))\} \cup G}}{\{u \approx s, v_1 \approx t_1,\cdots,v_n \approx t_n\} \cup G}$$

Obviously (because of the decomposition involved), $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption, there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 1F: Assume that s, t are variables and u is not. The application of Mutation to $u \approx v$ will result with a new goal G' that is equal to G with the quation $u \approx v$ erased.

$$\cfrac{\cfrac{\cfrac{f(u_1,\cdots,u_n) \approx f(v_1,\cdots,v_n) \quad (s \approx t \in E)}{\{Dec(f(u_1,\cdots,u_n) \approx s), Dec(f(v_1,\cdots,v_n) \approx t)\} \cup G}}{G[s \mapsto f(u_1,\cdots,u_n)][t \mapsto f(v_1,\cdots,v_n)]}}{G}$$

Obviously, $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption, there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also,

$G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 1G: Assume that u, t are variables and s is not. The application of Mutation to $u \approx v$ will result with the goal $G' \approx (\{u \approx f(s_1, \cdots, s_n)\} \cup G[u \mapsto f(s_1, \cdots, s_n)])[t \mapsto f(v_1, \cdots, v_n)]$.

$$\frac{\dfrac{u \approx f(v_1, \cdots, v_n) \quad (f(s_1, \cdots, s_n) \approx t \in E)}{\{Dec(u \approx f(s_1, \cdots, s_n)), Dec(f(v_1, \cdots, v_n) \approx t)\} \cup G}}{(\{u \approx f(s_1, \cdots, s_n)\} \cup G[u \mapsto f(s_1, \cdots, s_n)])[t \mapsto f(v_1, \cdots, v_n)]}$$

Both substitutions $[u \mapsto f(s_1, \cdots, s_n)]$ and $[t \mapsto f(v_1, \cdots, v_n)]$ will change nothing with respect to the grounding substitution θ', the equation $u \approx f(s_1, \cdots, s_n)$ will be solved, hence not taken into account in the measure, hence $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption, there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 1H: Finally, assume that all u, s, t are variables. Mutate will result with a new goal $G' = \{(\{u \approx s\} \cup G)[t \mapsto f(v_1, \cdots v_n)]\}$.

$$\frac{\dfrac{u \approx f(v_1, \cdots, v_n) \quad (s \approx t \in E)}{\{Dec(u \approx s), Dec(f(v_1, \cdots, v_n) \approx t)\} \cup G}}{(\{u \approx s\} \cup G)[t \mapsto f(v_1, \cdots v_n)]}$$

In fact, t is new in the inference, hence it can only be equal to s, and can be nowhere else in the new goal. Because $t\theta' = f(v_1, \cdots v_n)\theta'$ the substitution $[t \mapsto f(v_1, \cdots v_n)]$ cannot affect the ground proof, hence, obviously, $\mu(G'\theta') < \mu(G, \theta)$. By the induction assumption, there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

Case 2 Now suppose that the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an application of Congruence. There are two cases here: u is a variable or u is not a variable.

Case 2A First we will consider the case where u is not a variable. Then $u = f(u_1, \cdots, u_n)$, $v = f(v_1, \cdots, v_n)$ and $E \vdash u_i\theta \approx v_i\theta$ for all i.
There is an application of Decomposition that can be applied to $u \approx v$, resulting in the new goal $G' = \{u_1 \approx v_1, \cdots, u_n \approx v_n\} \cup G_1$. Then $|u_i\theta \approx v_i\theta|_E < |u\theta \approx v\theta|$ for all i, so $\mu(G', \theta) < \mu(G, \theta)$.
By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta$. This implies that $G \xrightarrow{*} H$ and $G\theta_H \leq_E G\theta$.

Case 2B: Now we consider the final case, where u is a variable and the rule at the root of the proof tree of $E \vdash u\theta \approx v\theta$ is an application of Congruence. Let $u\theta = f(u_1, \cdots, u_n)$. Then, for each i, $E \models u_i \approx v_i\theta$, and $|u_i \approx v_i\theta|_E < |u \approx v|_E$.
There is an application of Variable Decomposition that can be applied to $u \approx v$, resulting in the new goal $G' = \{u \approx f(x_1, \cdots, x_n)\} \cup (\{x_1 \approx v_1, \cdots, x_n \approx v_n\} \cup G_1)[u \mapsto f(x_1, \cdots, x_n)]$.

Let θ' be the substitution $\theta \cup [x_1 \mapsto u_1, \cdots, x_n \mapsto u_n]$. Then $u \approx f(x_1, \cdots, x_n)$ is solved in G'. Also $|x_i\theta' \approx v_i\theta'|_E < |u\theta \approx v\theta|_E$ for all i. Therefore $\mu(G, \theta) < \mu(G', \theta')$. By the induction assumption there is an H such that $G' \xrightarrow{*} H$ with $G'\theta_H \leq_E G'\theta'$. This implies that $G \xrightarrow{*} H$. Also, $G\theta_H \leq_E G\theta'$, since the variables of G are a subset of the variables of G'. Since $G\theta' = G\theta$, we know that $G\theta_H \leq G\theta$.

□

10 Conclusion

We have given a new goal-directed inference system for E-unification. We are interested in goal-directed E-unification for two reasons. One is that many other inferences systems for which E-unification would be useful are goal directed, and so a goal-directed inference system will be easier to combine with other inference systems. The second reason is that we believe this particular inference system is such that we can use it to find some decidable classes of equational theories for E-unification and analyze their complexity. We are writing a forthcoming paper on this topic.

The inference system we have given is similar to the Syntactic Mutation inference system of [5]. The difference is that our inference system can be applied to all equational theories, not just Syntactic Theories as in their case. Also, we give a completeness proof, even if E contains collapsing axioms.

References

1. F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge, 1998.
2. J. Gallier and W. Snyder. A general complete E-unification procedure. In *RTA 2*, ed. P. Lescanne, LNCS vol. 256, 216-227, 1987.
3. J. Gallier and W. Snyder. Complete sets of transformations for general E-unification. In *TCS*, vol. 67, 203-260, 1989.
4. C. Kirchner. Computing unification algorithms. In *Proceedings of the First Symposium on Logic in Computer Science*, Boston, 200-216, 1990.
5. C. Kirchner and H. Kirchner. *Rewriting, Solving, Proving.* `http://www.loria.fr/~ckirchne/` , 2000.
6. C. Kirchner and F. Klay. Syntactic Theories and Unification. In *LICS 5*, 270-277, 1990.
7. F. Klay. Undecidable Properties in Syntactic Theories. In *RTA 4*,ed. R. V. Book, LNCS vol. 488, 136-149, 1991.
8. D. E. Knuth and P. B. Bendix. Simple word problems in universal algebra. In *Computational Problems in Abstract Algebra*, ed. J. Leech, 263-297, Pergamon Press, 1970.
9. C. Lynch and B. Morawska. Goal Directed E-Unification. `http://www.clarkson.edu/~clynch/papers/goal_long.ps/`, 2000.