数理解析研究所講究録
1041 巻 1998 年 211-218

A Global Ordering Constraint for a Top-Down Transformation system of General E-Unification
— A Preliminary Report — *

Koji Iwanuma (岩沼 宏治)

Department of Electrical Engineering and Computer Science
Yamanashi University
Takeda 4-3-11, Kofu, Yamanashi, 400, Japan.
E-mail: iwanuma@esi.yamanashi.ac.jp

Abstract. In this paper, we give a brief study on an ordering constraint for a top-down transformation system of general E-unification. It is well known that the completeness of top-down E-unification transformation can be achieved only if we sacrifice an ordinary ordering constraint for a top-down system. More precisely, any *local* ordering constraint can not be imposed on each equality inference step. In this paper, we show that a weaker, i.e., *global* ordering restriction can be imposed for a top-down transformation system for general E-unification, without losing the completeness.

1 Introduction

Equality reasoning is indeed one of central problems in the research field of automated theorem proving. General E-unification problem [11] is one of the most important subproblems of equality reasoning. So far, several sophisticated methods, such as ordered paramodulation [9], superposition [3], basic paramodulation [4] and completion [2] etc., were proposed mainly in a saturation framework, i.e., in the framework of bottom-up computation

Compared with these bottom-up methods, top-down (or goal-oriented, in other words) proving methods usually have a great advantage, i.e., their goal-oriented behavior, for general first-order theorem proving. Moreover almost all top-down provers [10, 7, 12, 14] can be favored by PTTP (Prolog Technology Theorem Prover) technology [17, 18], which supplies a brute-force, but extremely fast inference engine to tableau-based top-down theorem provers.

However, unfortunately, top-down proving methods is slightly weak to equality reasoning problems. It is well known that the complete top-down computation can be achieved only if it sacrifices ordinary ordering constraints for the computations. Furthermore if paramodulation into variables is forbidden, we must introduce a sort of lazy application mechanism of equations (see [16]).

A remarkable top-down transformation system for general E-unification problem was proposed by Gallier and Snyder [8]. Dougherty and Johann [6] introduced the *top-unify*

* This research was supported in part by TAO (Telecommunications Advancement Organization of Japan).

mechanism and succeeded in reducing non-determinism involved in Gallier and Synder's system. Moser [15] independently reconstructed Gallier and Synder's one with simple two inference rules, i.e., ordinary unification and lazy paramodulation, and moreover integrated "basic" restriction [3, 4]. Furthermore Moser gave the independence of the selection rule of equational subgoals at each inference stage. However Moser's system $\mathcal{T}_{\mathrm{BP}}$ still does not have any ordering constraint at all.

In this paper, we give a short study on a weak form of ordering constraint for Moser's $\mathcal{T}_{\mathrm{BP}}$. Ordinary ordering constraints in the literature can be recognized as a *local* conditions. Every inference rule such as paramodulation or superposition must satisfy some ordering conditions whenever it is invoked. This paper is concerned with a weaker form of ordering, i.e., a *global* constraint. This global ordering constrain is imposed on entire (sub-)transformation sequences of $\mathcal{T}_{\mathrm{BP}}$, but not on each inference step. This refined top-down system denoted as $\mathcal{T}_{\mathrm{OBP}}$, temporally allows some equational goals to be rewritten into larger ones in a given order, at some points of a transformation. Eventually these inappropriate rewritten goals are forced to be modified into smaller ones in some succeeding steps.

This paper is organized as follows: Section 2 is preliminaries. Section 3 shows a globally ordered transformation system for general E-unification.

2 Preliminaries

Following [11, 15], we shall provide a brief sketch of basic concepts and definitions used in this paper. See [4, 8, 11, 15] for a more detailed description.

A *position* p in a term t is presented by a sequence of positive integers. $\mathcal{P}\mathrm{os}(t)$ denotes the set of all positions in a term t. $\mathcal{FP}\mathrm{os}(t)$ is the set of all non-variable positions. Λ represents the top position in a term. $t|_p$ expresses the *subterm* of t at the position p and $t[s]_p$ the result of replacing the subterm in t at the position p by the term s. For a term t, $\mathcal{H}(t)$ denotes the function symbol heading it.

An *equation* is a pair of terms s and t and written in the form $s \simeq t$. Given a set E of equations called *equational axioms*, we write $s \leftrightarrow^{p}_{l\simeq r} t$ or simply $s \leftrightarrow_E t$ if $s|_p = l\theta$ and $t = s[r\theta]_p$ for some substitution θ and equation $l \simeq r$ (or $r \simeq l$) in E. The *equational theory* of E is the transitive closure of the above relation and is denoted $=_E$. Note that $=_E$ is a congruence relation. Throughout this paper, we assume that E is *consistent*, that is, there is no pair $\langle s, t\rangle$ in $=_E$ such that s is a variable and does not occur in a term t. For substitutions θ and σ, we write $\theta \leq_E \sigma$ if there is a substitution δ such that for every variable x, $(x\theta)\delta =_E (x\sigma)$.

Given an equational axiom set E and two terms s and t, a substitution θ is an *E-unifier* of s and t if $s\theta =_E t\theta$. The problem of finding E-unifiers between s and t is called the *E-unification problem*, and denoted as $s =^?_E t$. This paper adopts a refutational setting as in [15], so $\neg(s \simeq t)$ (or for convenience $s \not\simeq t$) is called an *E-unification goal.* A *complete set of E-unifiers* $\mathcal{CSU}_E(s,t)$ for terms s and t is a set of E-unifiers of s and t such that, for any E-unifier θ of s and t, there is a $\sigma \in \mathcal{CSU}_E(s,t)$ and $\sigma \leq_E \theta$.

A *closure* $e \cdot \sigma$ is a pair consisting of a *skeleton* e and a substitution σ. The skeleton can be an arbitrary structure, e.g., an equation or multiset of equations. Closures will be used to distinguish terms occuring in the original expression from terms introduced by substitutions. With closures, we shall forbid any applications of an equation to a subterm introduced by a substitution. This restriction is called the *basic* condition.

An *transformation system* $\mathcal{T}$ is a procedure which takes an equational theory E and a goal $s \not\simeq t$ and generates a set of E-unifiers. A system $\mathcal{T}$ is *complete* if $\mathcal{T}$ generates a complete set $\mathcal{CSU}_E(s,t)$ for any theories E and any pairs of terms s and t.

An order $\succ$ over a set of first-order terms is called a *simplification ordering* if it posses the following three properties:

1. if $s \succeq t$, then $s\theta \succeq t\theta$ for any substitutions θ.
2. if $s \succeq t$, then $C[s] \succeq C[t]$ for any contexts C[].
3. if t is a proper subterm of s, then $s \succ t$.

A simplification ordering $\succ$ over terms is called *strong* if $\succ$ is a total ordering over the set of ground terms.

3 Globally Ordered Transformation System

In this section, we present a new ordered transformation system $\mathcal{T}_{\mathrm{OBP}}$ for E-unification problems. $\mathcal{T}_{\mathrm{OBP}}$ is a refinement of Moser's $\mathcal{T}_{\mathrm{BP}}$ [15] by integrating a global ordering constraint with $\mathcal{T}_{\mathrm{BP}}$.

At first, we show $\mathcal{T}_{\mathrm{BP}}$, which consists of two inference rules: *unification* and *lazy-paramodulation.*

1. **Unification** ($\Downarrow$unify)

$$\frac{(R \cup \{s \not\simeq t\}) \cdot \sigma}{R \cdot \sigma\theta} \left\langle \theta = mgu(s\sigma, t\sigma) \right\rangle$$

2. **Lazy Basic Paramodulation** ($\Downarrow$ lazy-param)

$$\frac{(R \cup \{s \not\simeq t\}) \cdot \sigma}{(R \cup \{s|_p \not\simeq l,\ s[r]_p \not\simeq t\}) \cdot \sigma} \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ (l \simeq r) \in E, \\ \text{if } l \notin \mathcal{V}\mathrm{ar} \text{ then } \mathcal{H}(s|_p) = \mathcal{H}(l) \end{array} \right\rangle$$

A $\mathcal{T}_{\mathrm{BP}}$*-sequence* is a sequence of multiset closures where each member of the sequence can be obtained by applying an inference rule of $\mathcal{T}_{\mathrm{BP}}$ to a preceding member. A $\mathcal{T}_{\mathrm{BP}}$-sequence for $s =^?_E t$ *starts with* $\{s \not\simeq t\} \cdot \varepsilon$, where ε, is the empty substitution. It is called *terminating* if the empty closure $\{\} \cdot \sigma$ is derived. Then σ is the E-unifier resulting from the $\mathcal{T}_{\mathrm{BP}}$-sequence.

$\mathcal{T}_{\mathrm{BP}}$ is a complete transformation system [15]. However $\mathcal{T}_{\mathrm{BP}}$ does not adopt any ordering constraints at all. We shall introduce, without losing the completeness, a sort of global ordering constraint into $\mathcal{T}_{\mathrm{BP}}$.

We use an *annotated equation* for defining a global ordering condition. The annotation serves to show of the initial form of each equational goal, i.e., the form at the birth of the goal. More formally, an expression $\langle s : s' \rangle \simeq \langle t : t' \rangle$ is an *annotated equation* of terms s and t, where s and t are respectively annotated with terms s' and t'.

A globally ordered transformation system $\mathcal{T}_{\mathrm{OBP}}$ has two inference rules: *(globally) ordered unification* and *annotated lazy-paramodulation.*

1. **Ordered Unification** ($\Downarrow$o-unify)

$$\frac{(R \cup \{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\}) \cdot \sigma}{R \cdot \sigma\theta} \quad \left\langle \begin{array}{l} \theta = mgu(s\sigma, t\sigma) \\ s\sigma\theta \not\succeq s'\sigma\theta \text{ and } t\sigma\theta \not\succeq t'\sigma\theta \end{array} \right\rangle$$

2. **Annotated Lazy Basic Paramodulation** ($\Downarrow$a-lazy-param)

$$\frac{(R \cup \{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\}) \cdot \sigma}{\left(R \cup \left\{ \begin{array}{l} \langle s|_p : s|_p \rangle \not\simeq \langle l : l \rangle, \\ \langle s[r]_p : s' \rangle \not\simeq \langle t : t' \rangle \end{array} \right\}\right) \cdot \sigma} \quad \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ (l \simeq r) \in E, \\ \text{if } l \notin \mathcal{V}\mathrm{ar} \text{ then } \mathcal{H}(s|_p) = \mathcal{H}(l) \end{array} \right\rangle$$

The equational goal $\langle s|_p : s|_p \rangle \not\simeq \langle l : l \rangle$ as the result of $\Downarrow$a-lazy-param step is called *witness pair*, and the goal $\langle s[r]_p : s' \rangle \not\simeq \langle t : t' \rangle$ is called *result pair*.

The $\Downarrow$o-unify rule for terms s and t is applicable only if s and t are not greater than or equal to their initial forms s' and t', respectively. This ordering constraint is a global one, and is weaker than the ordinary one used in the saturation-based framework, where the ordering constraint is locally imposed on every rewriting step.

The $\Downarrow$a-lazy-param rule is essentially same to Moser's $\Downarrow$lazy-param. The additional work for $\Downarrow$a-lazy-param is to generate a new annotation for a witness pair and to inherit the annotation for the result pair. The newly generated annotation is the record of the initial form of the witness pair. This record will be necessary for the ordering constraint check performed at the succeeding $\Downarrow$o-unify steps.

A $\mathcal{T}_{\mathrm{OBP}}$*-sequence* is a sequence of multiset closures where each member of the sequence can be obtained by applying one of the above inference rules to a preceding one. A $\mathcal{T}_{\mathrm{OBP}}$-sequence for $s =^?_E t$ *starts with* $\{\langle s : s \rangle \not\simeq \langle t : t \rangle\} \cdot \varepsilon$. It is called *terminating* if the empty closure $\{\} \cdot \sigma$ is derived. Then σ is the E-unifier resulting from the $\mathcal{T}_{\mathrm{OBP}}$-sequence.

Theorem 1. *If the ordering $\succeq$ used in $\mathcal{T}_{\mathrm{OBP}}$ is a strong simplification ordering, then $\mathcal{T}_{\mathrm{OBP}}$ is a complete transformation system for any E-unification problems.*

Basic superposition proposed by Bachmair et al. [3, 4] is an excellent saturation-based calculus for general equational reasoning. Moser [15] showed that a restrict calculus, denoted as $\mathcal{S}$, of basic superposition for equational unit theories E is complete for enumerating elements of $\mathcal{CSU}_E(s, t)$ for any terms s and t.

In this paper, as a proof for the completeness of $\mathcal{T}_{\mathrm{OBP}}$, we shall show that, for any E-unification problem, $\mathcal{T}_{\mathrm{OBP}}$ can simulate any refutations made up over the restricted basic superposition $\mathcal{S}$ whenever an ordering used in $\mathcal{S}$ is a strong simplification ordering. The restrict calculus $\mathcal{S}$ consists of the following three rules:

1. **Equality Resolution**

$$\frac{(s \not\simeq t) \cdot \sigma}{\bot \cdot \sigma\theta} \left\langle \theta = mgu(s\sigma, t\sigma) \right\rangle$$

2. **Basic Left Superposition**

$$\frac{(s \not\simeq t) \cdot \sigma \quad (u \simeq v) \cdot \sigma}{(s[v]_p \not\simeq t) \cdot \sigma\theta} \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ \theta = mgu(s|_p\sigma, u\sigma) \\ t\sigma\theta \not\succeq s\sigma\theta \text{ and } v\sigma\theta \not\succeq u\sigma\theta \end{array} \right\rangle$$

3. **Basic Right Superposition**

$$\frac{(s \simeq t) \cdot \sigma \quad (u \simeq v) \cdot \sigma}{(s[v]_p \simeq t) \cdot \sigma\theta} \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ \theta = mgu(s|_p\sigma, u\sigma) \\ t\sigma\theta \not\succeq s\sigma\theta \text{ and } v\sigma\theta \not\succeq u\sigma\theta \end{array} \right\rangle$$

An $\mathcal{S}$*-sequence* for $s =^?_E t$ is a sequence $C_0, C_1, \cdots$, of closures such that

- C_0 is $(s \not\simeq t) \cdot \varepsilon$.
- each C_i for $i = 1, 2, \cdots$ is either
  - a closure $(l \simeq r) \cdot \varepsilon$ for an axiom $l \simeq r \in E$, or
  - a closure obtained by applying basic left (or right) superposition to preceding closure C_j and C_k $(0 \leq j, k < i)$.

An $\mathcal{S}$–sequence is a *refutation* if $\bot \cdot \sigma$ is derived. Then σ is the E-unifier resulting from the $\mathcal{S}$–sequence.

Theorem 2 (Completeness of $\mathcal{S}$ [15]). *The calculus $\mathcal{S}$ is complete for any E-unification problems.*

At first, we study how to simulate a refutation in $\mathcal{S}$ with a stronger calculus of $\mathcal{T}_{\mathrm{OBP}}$, where we shall ignore the ordering constraint for $\Downarrow$o-unify and the annotations affixed with equational goals in $\mathcal{T}_{\mathrm{OBP}}$. Note that this stronger calculus is essentially identical to $\mathcal{T}_{\mathrm{BP}}$. Next we shall verify that this simulation code also satisfies the global ordering constraint, which is given in the original $\mathcal{T}_{\mathrm{OBP}}$, therefore, we can conclude $\mathcal{T}_{\mathrm{OBP}}$ can simulate any refutations in $\mathcal{S}$, and thus $\mathcal{T}_{\mathrm{OBP}}$ is complete for E-unification problems.

The simulation of $\mathcal{S}$ by the stronger calculus of $\mathcal{T}_{\mathrm{OBP}}$(i.e., $\mathcal{T}_{\mathrm{BP}}$) is similar to the one in Moser [15]. In this paper, we directly do a simulation of the calculus $\mathcal{S}$. This is allowed by restricting the ordering used in $\mathcal{S}$ to a strong simplification ordering. As is well known, simplification ordering is much enough for practical application of $\mathcal{S}$ to arbitrary equational theory.

The direct simulation of $\mathcal{S}$ by $\mathcal{T}_{\mathrm{OBP}}$is achieved in two phases. The first is the simulation of basic left superposition. The second is for basic right superposition.

Lemma 3 (Simulation of Basic Left Superposition). *Let ρ be an E-unifier generated by an $\mathcal{S}$–sequence for $s =^?_E t$. Let $\widehat{E}$ be the set of equations used for basic left superposition. Then there is also a sequence of transformation steps in $\mathcal{T}_{\mathrm{OBP}}$ based on $\widehat{E}$ terminating with ρ.*

Proof. Each basic left superposition is of the form

$$\frac{(s \not\simeq t) \cdot \sigma \quad (u \simeq v) \cdot \sigma}{(s[v]_p \not\simeq t) \cdot \sigma\theta} \quad \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ \theta = mgu(s|_p\sigma, u\sigma) \\ t\sigma\theta \not\succeq s\sigma\theta \text{ and } v\sigma\theta \not\succeq u\sigma\theta \end{array} \right\rangle$$

This superposition can immediately be simulated in $\mathcal{T}_{\mathrm{OBP}}$ as follows:

$$\begin{array}{c} \{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma \\ \hline \{\langle s|_p : l \rangle \not\simeq \langle u : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma \\ \hline \{\langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta \end{array} \quad \begin{array}{l} \Downarrow \text{a-lazy-param}(u \simeq v) \\ \Downarrow \text{o-unify} \end{array}$$

Notice that the introduced annotation terms l and r in the intermediate step shown above are exactly same to $s|_p$ and u, respectively, if $u \simeq v \in E$. Otherwise l and r are appropriate terms which are introduced at some base step in a recursively constructed simulation of the basic right superposition with $u \simeq v$. □

Lemma 4 (Simulation of Basic Right Superposition). *Let ρ be an E-unifier generated by an $\mathcal{S}$-sequence for $s =^?_E t$ using a simplification ordering $\succ$. Let $\widehat{E}$ be the set of equations (and which are either from E or are generated by basic right superposition) used for basic left superposition and Q a corresponding sequence of transformation steps in $\mathcal{T}_{\mathrm{OBP}}$ based on $\widehat{E}$. Then there is also a sequence Q' where just equations from E are used and which terminates with a variant of ρ.*

Proof. We define the level of an equation to be the number of basic right superpositions which were necessary to obtain it. We proof the lemma by multiset induction on μ, where μ is a multiset of the level k of the equations in a $\mathcal{T}_{\mathrm{OBP}}$-transformation sequence based on $\widehat{E}$. For an equation in $\widehat{E}$ with level $k > 0$, there must be some basic right superposition

$$\frac{(u \simeq v) \cdot \sigma \quad (u' \simeq v') \cdot \sigma}{(u[v']_q \simeq v) \cdot \sigma\theta} \quad \left\langle \begin{array}{l} q \in \mathcal{FP}\mathrm{os}(u), \\ \theta = mgu(u|_q\sigma, u'\sigma) \\ v\sigma\theta \not\succeq u\sigma\theta \text{ and } v'\sigma\theta \not\succeq u'\sigma\theta \end{array} \right\rangle$$

where the input equations are levels $k', k'' < k$.

For the induction step, we have to distinguish among the direction of the application of the equation $(u[v']_q \simeq v) \cdot \sigma\theta$.

Case (a): left to right: In this case we furthermore have to among whether $q = \Lambda$ or not.

Case (a.1) left to right and $q \neq \Lambda$: We replace the transformation step

$$\frac{\{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta}{\{\langle s|_p : l \rangle \not\simeq \langle u[v']_q : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta} \Downarrow \text{a-lazy-param}(u[v'] \simeq v)$$

by two successive applications of $(u \simeq v) \cdot \sigma\theta$ and $(u' \simeq v') \cdot \sigma\theta$.

$$\begin{array}{c} \{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta \\ \hline \{\langle s|_p : l \rangle \not\simeq \langle u : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta \\ \hline \{\langle u|_q : m \rangle \not\simeq \langle u' : n \rangle,\ \langle s|_p : l \rangle \not\simeq \langle u[v']_q : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta \\ \hline \{\langle s|_p : l \rangle \not\simeq \langle u[v']_q : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta \end{array} \quad \begin{array}{l} \Downarrow \text{a-lazy-param}(u \simeq v) \\ \Downarrow \text{a-lazy-param}(u' \simeq v') \\ \Downarrow \text{o-unify} \end{array}$$

Notice that the introduced annotation terms l and r in the first intermediate step are exactly same to $s|_p$ and u, respectively, if $u \simeq v \in E$. Otherwise l and r are appropriate terms which should be introduced at some base step in a recursively constructed simulation of the basic right superposition with $u \simeq v$. The These are similar for the other annotation m and n.

Case (a.2) left to right and $q = \Lambda$: In this case the respective equation is $(v' \simeq v) \cdot \sigma\theta$ and we replace the transformation step

$$\frac{\{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta}{\{\langle s|_p : l \rangle \not\simeq \langle v' : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta} \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ (v' \simeq v) \in \widehat{E}, \\ \text{if } v' \notin \mathcal{V}\mathrm{ar} \text{ then } \mathcal{H}(s|_p) = \mathcal{H}(v') \end{array} \right\rangle$$

by two successive applications of $(v' \simeq u') \cdot \sigma\theta$ and $(u \simeq v) \cdot \sigma\theta$. Note that the simulation in this case is possible only if the term u' of $u' \simeq v'$is not a variable. Recall that we assumed E is consistent throughout this paper. Thus if the term u' is a variable, then u' must also occur in the term v' as its proper subterm. Hence we have $v'\delta \succeq u'\delta$ to such a pair u' and v' for any substitution δ, because $\succ$ is a simplification ordering. This is a contradiction to the condition $v\sigma\theta \not\preceq u'\sigma\theta$ in the hypothesis. This is the reason why u' is not a variable.

$$\begin{array}{cl}
\{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v' : r \rangle,\ \langle s[u']_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{a-lazy-param}(v' \simeq u') \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v' : r \rangle,\ \langle u' : m \rangle \not\simeq \langle u : n \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{a-lazy-param}(u \simeq v) \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v' : r \rangle,\ \langle s[v]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{o-unify}
\end{array}$$

Case (b): right to left: We replace the transformation step

$$\frac{\{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta}{\{\langle s|_p : l \rangle \not\simeq \langle v : r \rangle,\ \langle s[(u[v']_q)]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta} \left\langle \begin{array}{l} p \in \mathcal{FP}\mathrm{os}(s), \\ (v \simeq u[v']) \in \widehat{E}, \\ \text{if } v \notin \mathcal{V}\mathrm{ar} \text{ then } \mathcal{H}(s|_p) = \mathcal{H}(v) \end{array} \right\rangle$$

by two successive applications of $(v \simeq u) \cdot \sigma\theta$ and $(u' \simeq v') \cdot \sigma\theta$.

$$\begin{array}{cl}
\{\langle s : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v : r \rangle,\ \langle s[u]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{a-lazy-param}(v \simeq u) \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v : r \rangle,\ \langle u|_q : m \rangle \not\simeq \langle u' : n \rangle,\ \langle s[(u[v']_q)]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{a-lazy-param}(u' \simeq v') \\ \hline
\{\langle s|_p : l \rangle \not\simeq \langle v : r \rangle,\ \langle s[(u[v']_q)]_p : s' \rangle \not\simeq \langle t : t' \rangle\} \cdot \sigma\theta & \Downarrow \textsf{o-unify}
\end{array}$$

Now we have completed a direct simulation of $\mathcal{S}$ by $\mathcal{T}_{\mathrm{OBP}}$. The remaining for establishing the completeness of $\mathcal{T}_{\mathrm{OBP}}$ is to verify whether the simulation code shown above satisfy the global ordering constraints for $\Downarrow$o-unify steps. This can be achieved by investigating the proof of the completeness of basic superposition calculus S shown in [?]. There the strongness of simplification ordering plays an important role. The space allowed to us is limited, so we shall omit this task here.

References

1. P. Baumgartner and U. Furbach, Model elimination without contrapositives and its application to PTTP, *J. Automated Reasoning* **13** (1994) 339-360.
2. L. Bachmair and N. Dershowitz and D. Plaisted, Completion without failure, in: H. Ait-Kaci and M. Nivat eds., *Resolution of Equations in Algebraic Structures,* **2**, 1–30, (Academic Press, 1989).
3. L. Bachmair and H. Ganzinger, Rewrite-based equational theorem proving with selection and simplification, *Journal of Logic and Computation* **4** (1994) 217-247.
4. L. Bachmair, H. Ganzinger, C. Lynch and W. Synder, Basic paramodulation, *Information and Computation* **121** (1995) 172-192.
5. D. Brand, Proving theorems with the modification method, *SIAM Journal of Computing* **4** (4) (1975) 412-430.
6. D.J. Dougherty and P. Johann, An improved general E-unification method, in: *Proc. 10th Inter. Conf. on Automated Deduction (CADE-10), Lecture Notes in Artif. Intel.* **449** (1990) 261-275.
7. C. Goller, R. Letz, K. Mayr and J. Schumann, SETHEO V3.2: recent developments –system abstract–, *CADE-12, LNAI* **814** (1994) 778-782.
8. J.H. Gallier and W. Snyder, Complete sets of transformations for general E-unification, *Theoretical Computer Science* **67** (1989) 203–260.
9. J. Hsiang and M. Rusinowitch, Proving refutational completeness of theorem-proving strategies: the transfinite semantic tree method, *Journal of ACM* **38** (1991) 559-587.
10. K. Iwanuma, Lemma matching for a PTTP-based Top-down Theorem Prover, in: *Proc. 14th Inter. Conf. on Automated Deduction (CADE-14), Lecture Notes in Artif. Intel.* **1249** (1997) 146-160.
11. J.-P. Jouannaud and C. Kirchner, Solving equations in abstract Algebras: a rule-based survey of unification, in: *J.-L. Lassez and G. Plotkin eds., Computational Logic. Essays in Honor of Alan Robinson,* chapter 8, (MIT press, 1991) 257-321.
12. R. Letz, C. Goller and K. Mayr, Controlled integration of the cut rule into connection tableau calculi, *J. Automated Reasoning* **13** (1994) 297-338.
13. W.W. McCune, *OTTER 3.0 reference manual and guide*, Technical Report ANL-94/6, Argonne National Laboratory, 1994.
14. M. Moser, O. Ibens, R. Letz, J. Steinbach, C. Goller, J. Schumann and K. Mayr, SETHEO and E-SETHEO, *J. Automated Reasoning* **18** (1997) 237-246.
15. M. Moser Improving transformation systems for general E-unification *Proceedings 5th Inter. Conf. on Rewriting Techniques and Applications (RTA-5), LNAI* **690**, (1993) 92–105.
16. W. Snyder and C. Lynch, Goal-directed strategies for paramodulation, *Proceedings 4th Inter. Conf. on Rewriting Techniques and Applications (RTA'91), LNAI* **448**, (1991) 150-161.
17. M.E. Stickel, A prolog technology theorem prover: Implementation by an extended prolog compiler, *J. Automated Reasoning* **4** (1988) 353-380.
18. M.E. Stickel, A prolog technology theorem prover: a new exposition and implementation in prolog, *Theoret. Comput. Sci.* **104** (1992) 109-128.