Lazy Narrowing: Strong Completeness and Eager Variable Elimination

Aart Middeldorp ††
Satoshi Okui †
Tetsuo Ida ‡

University of Tsukuba
Tsukuba 305 Japan

1995

ISE-TR-95-119

†† Institute of Information Sciences and Electronics, e-mail: ami@softlab.is.tsukuba.ac.jp
† Doctoral Program in Engineering, e-mail: okui@softlab.is.tsukuba.ac.jp
‡ Institute of Information Sciences and Electronics, e-mail: ida@softlab.is.tsukuba.ac.jp

Lazy Narrowing: Strong Completeness and Eager Variable Elimination

Aart Middeldorp Satoshi Okui Tetsuo Ida
Institute of Information Sciences and Electronics
University of Tsukuba, Tsukuba 305, Japan
{ami,okui,ida}@score.is.tsukuba.ac.jp

February 23, 1995

ABSTRACT

Narrowing is an important method for solving unification problems in equational theories that are presented by confluent term rewriting systems. Because narrowing is a rather complicated operation, several authors studied calculi in which narrowing is replaced by more simple inference rules. This paper is concerned with one such calculus. Contrary to what has been stated in the literature, we show that the calculus lacks strong completeness, so selection functions to cut down the search space are not applicable. We prove completeness of the calculus and we establish an interesting connection between its strong completeness and the completeness of basic narrowing. We also address the eager variable elimination problem. It is known that many redundant derivations can be avoided if the variable elimination rule, one of the inference rules of our calculus, is given precedence over the other inference rules. We prove the completeness of a restricted variant of eager variable elimination in the case of orthogonal term rewriting systems.

1. Introduction

E-unification—solving equations modulo some equational theory E—is a fundamental technique in automated reasoning. Narrowing ([20, 4, 11]) is a general E-unification procedure for equational theories that are presented by confluent term rewriting systems (TRSs for short). Narrowing is the computational mechanism of many functional-logic programming languages (see Hanus [7] for a recent survey on the integration of functional and logic programming). It is well-known that narrowing is complete with respect to normalizable solutions. Completeness means that for every solution to a given equation, a more general solution can be found by narrowing. If we extend narrowing to goals consisting of several equations, we obtain *strong* completeness. This means that we don't lose completeness when we restrict applications of the narrowing rule to a single equation in each goal.

An extended abstract of this paper will appear in the Proceedings of the 20th Colloquium on Trees in Algebra and Programming, Aarhus, Lecture Notes in Computer Science, 1995.

Since narrowing is not easily implemented, several authors studied calculi consisting of a small number of more elementary inference rules that simulate narrowing (e.g. [16, 8, 9, 14, 21, 6]). In this paper we are concerned with a subset (actually the specialization to confluent TRSs) of the calculus TRANS proposed by Hölldobler [9]. We call this calculus *lazy narrowing calculus* (LNC for short). Because the purpose of LNC is to simulate narrowing by more elementary inference rules, it is natural to expect that LNC inherits strong completeness from narrowing, and indeed this is stated by Hölldobler (Corollary 7.3.9 in [9]). We show however that LNC lacks strong completeness.

An important improvement over narrowing is *basic* narrowing (Hullot [11]). In basic narrowing narrowing steps are never applied to (sub)terms introduced by previous narrowing substitutions, resulting in a significant reduction of the search space. In this paper we establish a surprising connection between LNC and basic narrowing: we show that LNC is strongly complete whenever basic narrowing is complete. The latter is known for complete (i.e., confluent and terminating) TRSs (Hullot [11]). Other sufficient conditions are right-linearity and orthogonality (Middeldorp and Hamoen [17]). So LNC is strongly complete for these three classes of TRSs. We prove completeness of LNC for the general case of confluent TRSs. In the literature completeness of LNC-like calculi is proved under the additional termination assumption. Without this assumption the completeness proof is significantly more involved.

It is known that LNC-like calculi generate many derivations which produce the same solutions (up to subsumption). Martelli *et al.* [16, 14] and Hölldobler [9], among others, pointed out that many of these redundant derivations can be avoided by giving the variable elimination rule, one of the inference rules of LNC-like calculi, precedence over the other inference rules. The problem whether this strategy is complete or not is called the *eager variable elimination* problem in [9, 21]. Martelli *et al.* stated in [14] that this is easily shown in the case of terminating (and confluent) TRSs, but Snyder questions the validity of this claim in his monograph [21] on E-unification. We address the eager variable elimination problem for non-terminating TRSs. We prove completeness of a slightly restricted version of eager variable elimination in the case of orthogonal TRSs. To this end we simplify and extend the main result of You [23] concerning the completeness of *outer* narrowing for orthogonal *constructor-based* TRSs.

The remainder of the paper is organized as follows. In a preliminary section we introduce narrowing and basic narrowing, and we state the relevant completeness results. The narrowing calculus that we are interested in—LNC—is defined in Section 3. In that section we also show that LNC is not strongly complete. In Section 4 we establish the connection between the strong completeness of LNC and the completeness of basic narrowing. We prove the completeness of LNC for general confluent systems in Section 5. Section 6 is concerned with the eager variable elimination problem. In the final section we give suggestions for further research. The appendix contains proofs of a few technical results.

2. Preliminaries

In this preliminary section we review the basic notions of term rewriting and narrowing. We refer to Dershowitz and Jouannaud [2] and Klop [12] for extensive surveys.

A *signature* is a set $\mathcal{F}$ of *function symbols*. Associated with every $f \in \mathcal{F}$ is a natural number denoting its arity. Function symbols of arity 0 are called *constants*. The set $\mathcal{T}(\mathcal{F}, \mathcal{V})$ of *terms* built from a signature $\mathcal{F}$ and a countably infinite set of *variables* $\mathcal{V}$ with $\mathcal{F} \cap \mathcal{V} = \varnothing$ is the smallest set containing $\mathcal{V}$ such that $f(t_1, \dots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ whenever $f \in \mathcal{F}$ has arity n and

$t_1, \ldots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. We write c instead of $c()$ whenever c is a constant. The set of variables occurring in a term t is denoted by $\mathcal{V}ar(t)$.

A *position* is a sequence of natural numbers identifying a subterm in a term. The set $\mathcal{P}os(t)$ of positions in a term t is inductively defined as follows: $\mathcal{P}os(t) = \{\varepsilon\}$ if t is a variable and $\mathcal{P}os(t) = \{\varepsilon\} \cup \{i{\cdot}p \mid 1 \leqslant i \leqslant n \text{ and } p \in \mathcal{P}os(t_i)\}$ if $t = f(t_1, \ldots, t_n)$. Here ε, the *root* position, denotes the empty sequence. If $p \in \mathcal{P}os(t)$ then $t_{|p}$ denotes the *subterm* of t at position t and $t[s]_p$ denotes the term that is obtained from t by replacing the subterm at position p by the term s. Formally, $t_{|p} = t$ and $t[s]_p = s$ if $p = \varepsilon$ and $t_{|p} = (t_i)_{|q}$ and $t[s]_p = f(t_1, \ldots, t_i[s]_q, \ldots, t_n)$ if $p = i{\cdot}q$ and $t = f(t_1, \ldots, t_n)$. The set $\mathcal{P}os(t)$ is partitioned into $\mathcal{P}os_{\mathcal{V}}(t)$ and $\mathcal{P}os_{\mathcal{F}}(t)$ as follows: $\mathcal{P}os_{\mathcal{V}}(t) = \{p \in \mathcal{P}os(t) \mid t_{|p} \in \mathcal{V}\}$ and $\mathcal{P}os_{\mathcal{F}}(t) = \mathcal{P}os(t) - \mathcal{P}os_{\mathcal{V}}(t)$. Elements of $\mathcal{P}os_{\mathcal{V}}(t)$ are called *variable positions*. Positions are partially ordered by the *prefix order* $\leqslant$, i.e., $p \leqslant q$ if there exists a (necessarily unique) r such that $p{\cdot}r = q$. In that case we define $q \backslash p$ as the position r. We write $p < q$ if $p \leqslant q$ and $p \neq q$. If neither $p \leqslant q$ nor $q \leqslant p$, we write $p \perp q$. The *size* $|t|$ of a term t is the cardinality of the set $\mathcal{P}os(t)$.

A *substitution* is a map θ from $\mathcal{V}$ to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ with the property that the set $\{x \in \mathcal{V} \mid \theta(x) \neq x\}$ is finite. This set is called the *domain* of θ and denoted by $\mathcal{D}(\theta)$. We frequently identify a substitution θ with the set $\{x \mapsto \theta x \mid x \in \mathcal{D}(\theta)\}$ of *variable bindings*. The *empty* substitution will be denoted by ϵ. So $\epsilon = \varnothing$ by abuse of notation. Substitutions are extended to homomorphisms from $\mathcal{T}(\mathcal{F}, \mathcal{V})$ to $\mathcal{T}(\mathcal{F}, \mathcal{V})$, i.e., $\theta(f(t_1, \ldots, t_n)) = f(\theta(t_1), \ldots, \theta(t_n))$ for every n-ary function symbol $f \in \mathcal{F}$ and terms $t_1, \ldots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. In the following we write $t\theta$ instead of $\theta(t)$. We denote the set $\bigcup_{x \in \mathcal{D}(\theta)} \mathcal{V}(x\theta)$ of variables *introduced* by θ by $\mathcal{I}(\theta)$. The *composition* $\theta_1\theta_2$ of two substitutions θ_1 and θ_2 is defined by $x(\theta_1\theta_2) = (x\theta_1)\theta_2$ for all $x \in \mathcal{V}$. A substitution θ_1 is *at least as general* as a substitution θ_2, denoted by $\theta_1 \leqslant \theta_2$, if there exists a substitution θ such that $\theta_1\theta = \theta_2$. The *restriction* $\theta\lceil_V$ of a substitution θ to a set $V (\subseteq \mathcal{V})$ of variables is defined as follows: $\theta\lceil_V(x) = \theta(x)$ if $x \in \mathcal{V}$ and $\theta\lceil_V(x) = x$ if $x \notin \mathcal{V}$. A *variable renaming* is a bijective substitution from $\mathcal{V}$ to $\mathcal{V}$. We write $\theta_1 = \theta_2\ [V]$ if $\theta_1\lceil_V = \theta_2\lceil_V$. We write $\theta_1 \leqslant \theta_2\ [V]$ if there exists a substitution θ such that $\theta_1\theta = \theta_2\ [V]$. A substitution θ is called *idempotent* if $\theta\theta = \theta$. A substitution θ is idempotent if and only if $\mathcal{D}(\theta) \cap \mathcal{I}(\theta) = \varnothing$. Terms s and t are *unifiable* if there exists a substitution θ, a so-called *unifier* of s and t, such that $s\theta = t\theta$. A *most general* unifier θ has the property that $\theta \leqslant \theta'$ for every other unifier θ' of s and t. Most general unifiers are unique up to variable renaming. Given two unifiable terms s and t, the unification algorithms of Robinson [19] and Martelli and Montanari [15] produce an idempotent most general unifier θ that satisfies $\mathcal{D}(\theta) \cup \mathcal{I}(\theta) \subseteq \mathcal{V}(s) \cup \mathcal{V}(t)$.

A *rewrite rule* is a directed equation $l \to r$ that satisfies $l \notin \mathcal{V}$ and $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$. A *term rewriting system* (TRS for short) is a set of rewrite rules. The *rewrite relation* $\to_{\mathcal{R}}$ associated with a TRS $\mathcal{R}$ is defined as follows: $s \to_{\mathcal{R}} t$ if there exists a rewrite rule $l \to r \in \mathcal{R}$, a substitution θ, and a position $p \in \mathcal{P}os(s)$ such that $s_{|p} = l\theta$ and $t = s[r\theta]_p$. The subterm $l\theta$ of s is called a *redex* and we say that s rewrites to t by *contracting* redex $l\sigma$. Occasionally we write $s \to_{p,l \to r,\theta} t$ or $s \to_{p,l \to r} t$. The transitive-reflexive closure of $\to_{\mathcal{R}}$ is denoted by $\to^*_{\mathcal{R}}$. If $s \to^*_{\mathcal{R}} t$ we say that s *rewrites* to t. The transitive closure of $\to_{\mathcal{R}}$ is denoted by $\to^+_{\mathcal{R}}$. We usually omit the subscript $\mathcal{R}$. A term without redexes is called a *normal form*. We say that a term t has a normal form if there exists a rewrite sequence starting from t that ends in a normal form. A substitution θ is called *normalized* (*normalizable*) if $x\theta$ is (has) a normal form for every $x \in \mathcal{D}(\theta)$. The routine proofs of the following lemmata are omitted.

LEMMA 2.1. *Let θ, θ_1, θ_2 be substitutions and V a set of variables.*
(1) *If $\theta\theta_1 = \theta\theta_2$ $[V]$ then $\theta_1 = \theta_2$ $[(V - \mathcal{D}(\theta)) \cup \mathcal{I}(\theta\restriction_V)]$.*
(2) *If $\theta_1 = \theta_2$ $[V]$ then $\theta\theta_1 = \theta\theta_2$ $[V']$ for any V' such that $(V' - \mathcal{D}(\theta)) \cup \mathcal{I}(\theta\restriction_{V'}) = V$.*
□

LEMMA 2.2. *Let θ, θ', θ_1, θ_2 be substitutions and V, V' sets of variables.*
(1) *If $\theta_1 \leqslant \theta_2$ $[V]$ and $(V' - \mathcal{D}(\theta)) \cup \mathcal{I}(\theta\restriction_{V'}) = V$ then $\theta\theta_1 \leqslant \theta\theta_2$ $[V']$.*
(2) *If $\theta \leqslant \theta'$ $[V]$, $\theta_1\theta' \leqslant \theta_2$ $[V']$, and $(V' - \mathcal{D}(\theta_1)) \cup \mathcal{I}(\theta_1\restriction_{V'}) \subseteq V$ then $\theta_1\theta \leqslant \theta_2$ $[V']$.*
□

LEMMA 2.3. *Let θ_1, θ_2 be substitutions and V a set of variables. If $\theta_1\theta_2\restriction_V$ is normalized then $\theta_2\restriction_{V'}$ is normalized for any V' such that $V' \subseteq (V - \mathcal{D}(\theta)) \cup \mathcal{I}(\theta_1\restriction_V)$.* □

A TRS is *terminating* if it doesn't admit infinite rewrite sequences. A TRS is *confluent* if for all terms t_1, t_2, t_3 with $t_1 \rightarrow^* t_2$ and $t_1 \rightarrow^* t_3$ there exists a term t_4 such that $t_2 \rightarrow^* t_4$ and $t_3 \rightarrow^* t_4$. If $l \rightarrow r$ is a rewrite rule and θ a variable renaming then the rewrite rule $l\theta \rightarrow r\theta$ is called a *variant* of $l \rightarrow r$. A rewrite rule $l \rightarrow r$ is *left-linear* (*right-linear*) if l (r) does not contain multiple occurrences of the same variable. A left-linear (right-linear) TRS only contains left-linear (right-linear) rewrite rules. Let $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ be variants of rewrite rules of a TRS $\mathcal{R}$ such that they have no variables in common. Suppose $l_{1|p}$, for some $p \in \mathcal{P}os_{\mathcal{F}}(l_1)$, and l_2 are unifiable with most general unifier θ. The pair of terms $(l_1[r_2]_p\theta, r_1\theta)$ is called a *critical pair* of $\mathcal{R}$, except in the case that $l_1 \rightarrow r_1$ and $l_2 \rightarrow r_2$ are renamed versions of the same rewrite rule and $p = \varepsilon$. A TRS without critical pairs is called *non-ambiguous*. An *orthogonal* TRS is left-linear and non-ambiguous. For orthogonal TRSs a considerable amount of theory has been developed, see Klop [12] for a comprehensive survey. The most prominent fact is that orthogonal TRSs are confluent. In Section 6 we make use of the work of Huet and Lévy [10] on standardization.

We distinguish a nullary function symbol true and a binary function symbol $\approx$, written in infix notation. A term of the form $s \approx t$, where neither s nor t contains any occurrences of $\approx$ and true, is called an *equation*. The term true is also considered as an equation. The extension of a TRS $\mathcal{R}$ with the rewrite rule $x \approx x \rightarrow \mathtt{true}$ is denoted by $\mathcal{R}_+$. Let e be an equation and θ a substitution. If there exists a rewrite sequence $e\theta \rightarrow^*_{\mathcal{R}_+} \mathtt{true}$, we write $\mathcal{R} \vdash e\theta$ and we say that θ is an ($\mathcal{R}$-)*solution* of e. Narrowing is formulated as the following inference rule:

$$\frac{e}{(e[r]_p)\theta} \quad \text{if there exist a fresh variant } l \rightarrow r \text{ of a rewrite rule in } \mathcal{R}_+, \text{ a position } p \in \mathcal{P}os_{\mathcal{F}}(e), \text{ and a most general unifier } \theta \text{ of } e_{|p} \text{ and } l.$$

In the above situation we write $e \leadsto_{\theta,p,l\rightarrow r} e'$. This is called an NC-*step* (NC stands for narrowing calculus). Subscripts will be omitted when they are clear from the context or irrelevant. A (finite) NC-*derivation* is a sequence

$$e_1 \leadsto_{\theta_1,p_1,l_1\rightarrow r_1} \cdots \leadsto_{\theta_{n-1},p_{n-1},l_{n-1}\rightarrow r_{n-1}} e_n$$

of NC-steps and abbreviated to $e_1 \leadsto^*_\theta e_n$ where $\theta = \theta_1 \cdots \theta_{n-1}$. An NC-derivation which ends in true is called an NC-*refutation*. The following completeness result is due to Hullot [11].

THEOREM 2.4. *Let $\mathcal{R}$ be a confluent TRS. If $\mathcal{R} \vdash e\theta$ and $\theta\lceil_{\mathcal{V}ar(e)}$ is normalized*[1] *then there exists an* NC*-refutation $e \leadsto^*_{\theta'}$ true such that $\theta' \leqslant \theta$ $[\mathcal{V}ar(e)]$.* □

The narrowing calculus that we are interested in (LNC—to be defined in the next section) operates on sequences of equations, the so-called *goals*. A substitution θ is a solution of a goal $G = e_1, \ldots, e_n$, denoted by $\mathcal{R} \vdash G\theta$, if $\mathcal{R} \vdash e_i\theta$ for all $i \in \{1, \ldots, n\}$. We use $\top$ as a generic notation for goals containing only equations true. So $\mathcal{R} \vdash G\theta$ if and only if $G\theta \to^*_{\mathcal{R}_+} \top$. The calculus NC is extended to goals as follows:

$$\frac{G_1, e, G_2}{(G_1, e[r]_p, G_2)\theta}$$ if there exist a fresh variant $l \to r$ of a rewrite rule in $\mathcal{R}_+$, a position $p \in \mathcal{P}os_{\mathcal{F}}(e)$, and a most general unifier θ of $e_{|p}$ and l.

Notions like NC-step, NC-derivation, and NC-refutation are defined as in the single equation case. We use the symbol Π (and its derivatives) to denote NC-derivations over goals. There are three sources of non-determinism in NC: the choice of the equation e, the choice of the subterm $e_{|p}$, and the choice of the rewrite rule $l \to r$. The last two choices are *don't know* non-deterministic, meaning that in general all possible choices have to be considered in order to guarantee completeness. The choice of the equations e is *don't care* non-deterministic, because of the *strong completeness* of NC. Strong completeness means completeness independent of selection functions. A *selection function* is mapping that assigns to every goal G different from $\top$ an equation $e \in G$ different from true. An example of a selection function is $\mathcal{S}_{left}$ which always returns the leftmost equation different from true. We say that an NC-derivation Π *respects* a selection function $\mathcal{S}$ if the selected equation in every step $G_1 \leadsto G_2$ of Π coincides with $\mathcal{S}(G_1)$. Now strong completeness of NC is formulated as follows.

THEOREM 2.5. *Let $\mathcal{R}$ be a confluent TRS and $\mathcal{S}$ a selection function. If $\mathcal{R} \vdash G\theta$ and $\theta\lceil_{\mathcal{V}ar(G)}$ is normalized then there exists an* NC*-refutation $G \leadsto^*_{\theta'} \top$ respecting $\mathcal{S}$ such that $\theta' \leqslant \theta$ $[\mathcal{V}ar(G)]$.* □

In the last part of this preliminary section we introduce basic narrowing. Hullot [11] defined basic narrowing for the single equation case. The extension to goals presented below follows Middeldorp and Hamoen [17].

DEFINITION 2.6. A *position constraint* for a goal G is a mapping that assigns to every equation $e \in G$ a subset of $\mathcal{P}os_{\mathcal{F}}(e)$. The position constraint that assigns to every $e \in G$ the set $\mathcal{P}os_{\mathcal{F}}(e)$ is denoted by $\bar{G}$.

DEFINITION 2.7. An NC-derivation

$$G_1 \leadsto_{p_1, \theta_1, l_1 \to r_1, e_1} \cdots \leadsto_{p_{n-1}, \theta_{n-1}, l_{n-1} \to r_{n-1}, e_{n-1}} G_n$$

is *based* on a position constraint B_1 for G_1 if $p_i \in B_i(e_i)$ for $1 \leqslant i \leqslant n-1$. Here the position constraints $B_2, \ldots, B_{n-1}$ for the goals $G_2, \ldots, G_{n-1}$ are inductively defined by

$$B_{i+1}(e) = \begin{cases} B_i(e') & \text{if } e' \in G_i - \{e_i\} \\ \mathcal{B}(B_i(e_i), p_i, r_i) & \text{if } e' = e_i[r_i]_{p_i} \end{cases}$$

[1] Often completeness is stated with respect to normaliz*able* solutions: if $\mathcal{R} \vdash e\theta$ and $\theta\lceil_{\mathcal{V}ar(e)}$ is normalizable then there exists an nc-refutation $e \leadsto^*_{\theta'}$ true such that $\theta' \leqslant_{\mathcal{R}} \theta$ $[\mathcal{V}ar(e)]$. Notwithstanding the fact that completeness with respect to normalized solutions implies completeness with respect to normalizable solutions *but not vice-versa*, to all intents and purposes normalization and normalizability are interchangeable.

for all $1 \leqslant i < n-1$ and $e = e'\theta_i \in G_{i+1}$, with $\mathcal{B}(B_i(e_i), p_i, r_i)$ abbreviating the set of positions

$$B_i(e_i) - \{q \in B_i(e_i) \mid q \geqslant p_i\} \cup \{p_i{\cdot}q \in \mathcal{P}os_{\mathcal{F}}(e) \mid q \in \mathcal{P}os_{\mathcal{F}}(r_i)\}.$$

An NC-derivation issued from a goal G is called *basic* if it is based on $\bar{G}$.

The following statement summarizes the known completeness results for basic narrowing. Part (1) is due to Hullot [11]. Parts (2) and (3) are due to Middeldorp and Hamoen [17].

THEOREM 2.8. *Let $\mathcal{R}$ be a confluent TRS. If $\mathcal{R} \vdash G\theta$ and $\theta\restriction_{\mathcal{V}ar(G)}$ is normalized then there exists a basic* NC*-refutation $G \leadsto^*_{\theta'} \top$ such that $\theta' \leqslant \theta$ $[\mathcal{V}ar(G)]$, provided one of the following conditions is satisfied:*
(1) *$\mathcal{R}$ is terminating,*
(2) *$\mathcal{R}$ is orthogonal and $G\theta$ has an $\mathcal{R}$-normal form, or*
(3) *$\mathcal{R}$ is right-linear.*
□

3. Lazy Narrowing Calculus

Calculi in which the narrowing inference rule is replaced by a small number of more primitive operations are comprehensively examined by Hölldobler in his thesis [9] and Snyder in his monograph [21]. The calculus that we investigate in this paper is the specialization of Hölldobler's calculus TRANS, which is defined for general equational systems and based on paramodulation, to (confluent) TRSs and narrowing.

DEFINITION 3.1. Let $\mathcal{R}$ be a TRS. The *lazy narrowing calculus*, LNC for short, consists of the following five inference rules:

[*on*] *outermost narrowing*

$$\frac{G_1, f(s_1,\ldots,s_n) \simeq t, G_2}{G_1, s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, G_2}$$

if there exists a fresh variant $f(l_1,\ldots,l_n) \to r$ of a rewrite rule in $\mathcal{R}$

[*im*] *imitation*

$$\frac{G_1, f(s_1,\ldots,s_n) \simeq x, G_2}{(G_1, s_1 \approx x_1, \ldots, s_n \approx x_n, G_2)\theta}$$

if $\theta = \{x \mapsto f(x_1,\ldots,x_n)\}$ with $x_1,\ldots,x_n$ fresh variables

[*d*] *decomposition*

$$\frac{G_1, f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n), G_2}{G_1, s_1 \approx t_1, \ldots, s_n \approx t_n, G_2}$$

[*v*] *variable elimination*

$$\frac{G_1, x \simeq t, G_2}{(G_1, G_2)\theta}$$

if $x \notin \mathcal{V}ar(t)$ and $\theta = \{x \mapsto t\}$

[*t*] *removal of trivial equations*

$$\frac{G_1, x \approx x, G_2}{G_1, G_2}$$

Here $s \simeq t$ stands for $s \approx t$ or $t \approx s$.

The variable elimination rule $[v]$ is different from the one of Martelli *et al.* [16, 14] in that we don't keep the solved equation $x \simeq t$ around. The rules $[v]$, $[d]$, and $[t]$ constitute the syntactic unification algorithm of Martelli and Montanari [15]. We refer to these three rules as UC, which stands for *unification calculus.* Because syntactic unification is performed by UC, the rewrite rule $x \approx x \to \mathtt{true}$ is no longer used in LNC. As a consequence, we may assume that the symbol `true` doesn't occur in LNC-goals.

Contrary to usual narrowing, the outermost narrowing rule $[on]$ generates new *parameter-passing* equations $s_1 \approx l_1, \ldots, s_n \approx l_n$ besides the *body* equation $r \approx t$. These parameter-passing equations must eventually be solved in order to obtain a refutation, but we don't require that they are solved right away. That is the reason why we call the calculus *lazy.* We introduce some useful notations relating to the calculus LNC. If G and G' are the upper and lower goal in the inference rule $[\alpha]$ ($\alpha \in \{on, im, d, v, t\}$), we write $G \Rightarrow_{[\alpha]} G'$. This is called an LNC-step. The applied rewrite rule or substitution may be supplied as subscript, that is, we will write things like $G \Rightarrow_{[on], l \to r} G'$ and $G \Rightarrow_{[im], \theta} G'$. LNC-derivations are defined as in the case of NC. An LNC-refutation is an LNC-derivation ending in the *empty goal* $\square$.

Because the purpose of LNC is to simulate narrowing, it is natural to expect that LNC inherits strong completeness from NC. Indeed, Hölldobler [9, Corollary 7.3.9] states the strong completeness of LNC for confluent TRSs with respect to normalizable solutions. However, this does not hold.

COUNTEREXAMPLE 3.2. Consider the TRS

$$\mathcal{R} = \begin{cases} f(x) & \to & g(h(x), x) \\ g(x, x) & \to & a \\ b & \to & h(b) \end{cases}$$

and the goal $G = f(b) \approx a$. Confluence of $\mathcal{R}$ can be proved by a routine induction argument on the structure of terms. The (normalized) empty substitution ε is a solution of G because

$$f(b) \approx a \;\to_{\mathcal{R}}\; g(h(b), b) \approx a \;\to_{\mathcal{R}}\; g(h(b), h(b)) \approx a \;\to_{\mathcal{R}}\; a \approx a \;\to_{\mathcal{R}_+}\; \mathtt{true}.$$

Consider the selection function $\mathcal{S}_{right}$ that selects the rightmost equation in every goal. There is essentially only one LNC-derivation issued from G respecting $\mathcal{S}_{right}$:

$$\begin{array}{lll} f(b) \approx a & \Rightarrow_{[on], f(x) \to g(h(x), x)} & b \approx x, g(h(x), x) \approx a \\ & \Rightarrow_{[on], g(x_1, x_1) \to a} & b \approx x, h(x) \approx x_1, x \approx x_1, a \approx a \\ & \Rightarrow_{[d]} & b \approx x, h(x) \approx x_1, x \approx x_1 \\ & \Rightarrow_{[v], \{x_1 \mapsto x\}} & b \approx x, h(x) \approx x \\ & \Rightarrow_{[im], \{x \mapsto h(x_2)\}} & b \approx h(x_2), h(x_2) \approx x_2 \\ & \Rightarrow_{[im], \{x_2 \mapsto h(x_3)\}} & \cdots \end{array}$$

This is clearly not a refutation. (The alternative binding $\{x \mapsto x_1\}$ in the $\Rightarrow_{[v]}$-step results in a variable renaming of the above LNC-derivations.) Hence LNC is *not* strongly complete.

This counterexample doesn't refute the completeness of LNC. The goal $f(b) \approx a$ can be solved, for instance, by adopting the selection function $\mathcal{S}_{left}$:

$$\begin{array}{lll} f(b) \approx a & \Rightarrow_{[on],\, f(x) \to g(h(x),x)} & b \approx x, g(h(x),x) \approx a \\ & \Rightarrow_{[v],\, \{x \mapsto b\}} & g(h(b),b) \approx a \\ & \Rightarrow_{[on],\, g(x_1,x_1) \to a} & h(b) \approx x_1, b \approx x_1, a \approx a \\ & \Rightarrow_{[v],\, \{x_1 \mapsto h(b)\}} & b \approx h(b), a \approx a \\ & \Rightarrow_{[on],\, b \to h(b)} & h(b) \approx h(b), a \approx a \\ & \Rightarrow_{[d]} & b \approx b, a \approx a \\ & \Rightarrow_{[d]} & a \approx a \\ & \Rightarrow_{[d]} & \square. \end{array}$$

In Section 5 we show that LNC is complete in the general case of confluent TRSs and normalized solutions. In the next section we present sufficient conditions for the strong completeness of LNC, which turns out to be a simpler than proving completeness.

4. Restoring Strong Completeness

Observe that the TRS $\mathcal{R}$ of Counterexample 3.2 satisfies none of the sufficient conditions for the completeness of basic narrowing stated in Theorem 2.8. As a matter of fact, basic narrowing is not able to solve the goal $f(b) \approx a$, see Middeldorp and Hamoen [17]. This suggests a surprising connection between strong completeness of LNC and completeness of basic NC. In this section we prove that LNC is strongly complete whenever basic NC is complete.

The basis of our proof is the specialization of the transformation process used by Hölldobler in his proof of the (strong) completeness of TRANS. First we formalize the intuitively clear propagation of equations along NC-derivations.

DEFINITION 4.1. Let $G \leadsto_{\theta,p,l\to r} G'$ be an NC-step and e an equation in G. If e is the selected equation in this step, then e is narrowed into the equation $e[r]_p\theta$ in G'. In this case we say that $e[r]_p\theta$ is the *descendant* of e in G'. Otherwise, e is simply instantiated to the equation $e\theta$ in G' and we call $e\theta$ the descendant of e. The notion of descendant extends to NC-derivations in the obvious way.

Observe that in an NC-refutation $G \leadsto^* \top$ every equation $e \in G$ has exactly one descendant $\mathtt{true}$ in $\top$. We now introduce five transformation steps on NC-refutations. The first one states that non-empty NC-refutations are closed under renaming.

LEMMA 4.2. *Let δ be a variable renaming. For every NC-refutation*

$$\Pi\colon\ G \ \leadsto^+_\theta\ \top$$

there exists an NC-refutation

$$\phi_\delta(\Pi)\colon\ G\delta \ \leadsto^+_{\delta^{-1}\theta}\ \top.$$

PROOF. Let $G \leadsto_{\sigma,p,l\to r,e} G'$ be the first step of Π and let $\Pi_1\colon G' \leadsto^*_{\theta_1} \top$ be the remainder of Π. We have $\sigma\theta_1 = \theta$. We show the existence of an NC-step $G\delta \leadsto_{\delta^{-1}\sigma,p,l\delta\to r\delta,e\delta} G'$. First we show that $\delta^{-1}\sigma$ is a most general unifier of $(e\delta)_{|p}$ and $l\delta$. We have $(e\delta)_{|p}\delta^{-1}\sigma = e_{|p}\sigma = l\sigma = (l\delta)\delta^{-1}\sigma$, so $\delta^{-1}\sigma$ is a unifier of $(e\delta)_{|p}$ and $l\delta$. Let θ be an arbitrary unifier of $(e\delta)_{|p}$ and $l\delta$. Because $\delta\theta$ is a unifier of $e_{|p}$ and l, and σ is a most general unifier of these two terms, it follows that $\sigma \leqslant \delta\theta$ and thus $\delta^{-1}\sigma \leqslant \theta$. We conclude that $\delta^{-1}\sigma$ is a most general unifier of $(e\delta)_{|p}$ and $l\delta$. Write G as G_1, e, G_2. We obtain

$$G\delta \leadsto_{\delta^{-1}\sigma, p, l\delta \to r\delta, e\delta} (G_1\delta, (e\delta)[r\delta]_p, G_2\delta)\delta^{-1}\theta = (G_1, e[r]_p, G_2)\theta = G'.$$

Concatenating this NC-step with the NC-refutation Π_1 yields the desired NC-refutation $\phi_\delta(\Pi)$:

$$G\delta \leadsto^{+}_{\delta^{-1}\theta} \top.$$

□

Observe that Lemma 4.2 doesn't hold for the empty NC-refutation Π. The second transformation corresponds to Proposition 7.3.4 in Hölldobler [9].

LEMMA 4.3. *Let*

$$\Pi: \; G_1, s \approx t, G_2 \leadsto^{*}_{\theta} \top$$

be an NC*-refutation with the property that narrowing is applied to a descendant of $s \approx t$ at position 1. Let V be a finite set of variables such that $\mathcal{V}ar(G_1, s \approx t, G_2) \subseteq V$. If $l \to r$ is the applied rewrite rule in the first such step then there exists an* NC*-refutation*

$$\phi_{[on]}(\Pi): \; G_1, s \approx l, r \approx t, G_2 \leadsto^{*}_{\theta'} \top$$

such that $\theta' = \theta \; [V]$.

PROOF. Write $l = f(l_1, \ldots, l_n)$. The given refutation Π is of the form

$$\begin{aligned} G_1, s \approx t, G_2 &\leadsto^{*}_{\theta_1} G'_1, f(u_1, \ldots, u_n) \approx t', G'_2 \\ &\leadsto_{\theta_2, 1, l \to r} (G'_1, r \approx t', G'_2)\theta_2 \\ &\leadsto^{*}_{\theta_3} \top \end{aligned}$$

with $\theta_1\theta_2\theta_3 = \theta$. Let x be a fresh variable (so $x \notin V$) and define the substitution θ'_2 as the (disjoint) union of θ_2 and $\{x \mapsto l\theta_2\}$. Because θ'_2 is a most general unifier of $f(u_1, \ldots, u_n) \approx l$ and $x \approx x$, Π can be transformed into the refutation $\phi_{[on]}(\Pi)$:

$$\begin{aligned} G_1, s \approx l, r \approx t, G_2 &\leadsto^{*}_{\theta_1} G'_1, f(u_1, \ldots, u_n) \approx l, r \approx t', G'_2 \\ &\leadsto_{\theta'_2, x \approx x \to \mathtt{true}} (G'_1, \mathtt{true}, r \approx t', G'_2)\theta'_2 = (G'_1, \mathtt{true}, r \approx t', G'_2)\theta_2 \\ &\leadsto^{*}_{\theta_3} \top. \end{aligned}$$

Since $\theta_1\theta'_2\theta_3 = \theta \cup \{x \mapsto l\theta_2\theta_3\}$ and $x \notin \mathcal{V}$ we obtain $\theta = \theta_1\theta'_2\theta_3 \; [\mathcal{V}]$. □

The third transformation step corresponds to Proposition 7.3.3 in Hölldobler [9].

LEMMA 4.4. *Let*

$$\Pi: \; G_1, f(s_1, \ldots, s_n) \approx f(t_1, \ldots, t_n), G_2 \leadsto^{*}_{\theta} \top$$

be an NC*-refutation with the property that narrowing is never applied to a descendant of* $f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n)$ *at position 1 or 2. Let* V *be a finite set of variables such that* $\mathcal{V}ar(G_1, f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n), G_2) \subseteq V$. *There exists an* NC*-refutation*

$$\phi_{[d]}(\Pi)\colon\ \ G_1, s_1 \approx t_1, \ldots, s_n \approx t_n, G_2 \ \ \leadsto^*_{\theta'} \ \ \top$$

such that $\theta' \leqslant \theta\ [V]$.

PROOF. The given refutation Π must be of the form

$$\begin{aligned} G_1, f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n), G_2 \ \ &\leadsto^*_{\theta_1} \ \ G_1', f(s_1',\ldots,s_n') \approx f(t_1',\ldots,t_n'), G_2' \\ &\leadsto_{\theta_2,\epsilon} \ \ (G_1', \mathtt{true}, G_2')\theta_2 \\ &\leadsto^*_{\theta_3} \ \ \top \end{aligned}$$

with $\theta_1\theta_2\theta_3 = \theta$. The first part of Π can be transformed into

$$\Pi_1\colon\ \ G_1, s_1 \approx t_1, \ldots, s_n \approx t_n, G_2 \ \ \leadsto^*_{\theta_1} \ \ G_1', s_1' \approx t_1', \ldots, s_n' \approx t_n', G_2'.$$

Consider the step from $G_1', f(s_1',\ldots,s_n') \approx f(t_1',\ldots,t_n'), G_2'$ to $(G_1', \mathtt{true}, G_2')\theta_2$. Let $x \approx x \to \mathtt{true}$ be the employed rewrite rule, so θ_2 is a most general unifier of $x \approx x$ and $f(s_1',\ldots,s_n') \approx f(t_1',\ldots,t_n')$. There clearly exists a rewrite sequence

$$(G_1', s_1' \approx t_1', \ldots, s_n' \approx t_n', G_2')\theta_2 \ \ \to^*_\epsilon \ \ (G_1', \top, G_2')\theta_2.$$

Lifting[2] results in an NC-derivation

$$\Pi_2\colon\ \ G_1', s_1' \approx t_1', \ldots, s_n' \approx t_n', G_2' \ \ \leadsto^*_{\theta_2',\epsilon} \ \ (G_1', \top, G_2')\theta_2'$$

such that $\theta_2' \leqslant \theta_2\ [V \cup \mathcal{I}(\theta_1)]$. We distinguish two cases.

(1) Suppose $G_1', G_2' = \top$. In this case $\theta_3 = \varepsilon$. We simply define $\phi_{[d]}(\Pi) = \Pi_1; \Pi_2$. From $\theta_2' \leqslant \theta_2\ [V \cup \mathcal{I}(\theta_1)]$ we infer that $\theta' = \theta_1\theta_2' \leqslant \theta_1\theta_2 = \theta\ [V]$.

(2) The case $G_1', G_2' \neq \top$ is more involved. First observe that θ_2' is a unifier of $f(s_1',\ldots,s_n')$ and $f(t_1',\ldots,t_n')$. Using the fact that θ_2 is a most general unifier of $f(s_1',\ldots,s_n') \approx f(t_1',\ldots,t_n')$ and $x \approx x$, it is not difficult to show that $\theta_2 \leqslant \theta_2'\ [\mathcal{V} - \{x\}]$. Since $x \notin V \cup \mathcal{I}(\theta_1)$ we have in particular $\theta_2 \leqslant \theta_2'\ [V \cup \mathcal{I}(\theta_1)]$. It follows that there exists a variable renaming δ such that $\theta_2' = \theta_2\delta\ [V \cup \mathcal{I}(\theta_1)]$. Clearly $\mathcal{V}ar(G_1', G_2') \subseteq V \cup \mathcal{I}(\theta_1)$. The last part of Π can be trivially transformed (by changing the number of occurrences of true in each goal) into

$$\Pi_3\colon\ \ (G_1', \top, G_2')\theta_2 \ \ \leadsto^+_{\theta_3} \ \ \top$$

An application of Lemma 4.2 results in the NC-refutation

$$\phi_\delta(\Pi_3)\colon\ \ (G_1', \top, G_2')\theta_2' \ \ \leadsto^+_{\delta^{-1}\theta_3} \ \ \top.$$

Define $\phi_{[d]}(\Pi) = \Pi_1; \Pi_2; \phi_\delta(\Pi_3)$. We have $\theta' = \theta_1\theta_2'\delta^{-1}\theta_3 = \theta_1\theta_2\theta_3 = \theta\ [V]$.

□

[2] The lifting lemma for NC requires the normalization of the substitution θ_2, which is not necessarily the case here. The reason for requiring normalization is to avoid rewrite sequences in which a term introduced by θ_2 is rewritten, because such sequences cannot be lifted. In the present situation there is no problem since we know that all steps in the rewrite sequence take place at root positions.

It should be noted that in general we don't have $\theta' = \theta\ [V]$ in Lemma 4.4. Consider for example the NC-refutation

$$\Pi:\quad a \approx a \quad \rightsquigarrow_{\theta, x\approx x\to \mathtt{true}} \quad \mathtt{true}$$

where we used the (non-idempotent) most general unifier $\theta = \{x \mapsto a, y \mapsto z, z \mapsto y\}$. Decomposition results in the empty goal, so

$$\phi_{[d]}(\Pi):\quad \square$$

produces the empty substitution $\theta' = \varepsilon$. Clearly $\theta' \neq \theta\ [V]$ if V contains y or z.

The fourth transformation step corresponds to Corollary 7.3.5 in Hölldobler [9]. This corollary is an immediate consequence of Hölldobler's lifting lemma for reflection, instantiation, and paramodulation (Lemma 6.2.6 in [9]). This easy proof does not work in our case since narrowing, unlike paramodulation, cannot be applied at variable positions. Nevertheless, we can adapt the proof of the lifting lemma (for NC) to obtain the following result.

LEMMA 4.5. *Let $G \rightsquigarrow^*_\theta \top$ be an NC-refutation, V a finite set of variables, and γ a substitution such that $\mathcal{V}ar(G) \subseteq V$, $\gamma \leqslant \theta\ [V]$, and the variables in $\mathcal{D}(\gamma) \cup \mathcal{I}(\gamma)$ are different from the variables in the employed rewrite rules. There exists an NC-refutation $G\gamma \rightsquigarrow^*_{\theta'} \top$ which employs the same rewrite rules at the same positions in the corresponding equations of the goals in $G \rightsquigarrow^*_\theta \top$ such that $\gamma\theta' = \theta\ [V]$.* $\square$

The proof can be found in Appendix. The validity of the fourth transformation step is an easy consequence of this lemma.

LEMMA 4.6. *Let*

$$\Pi:\quad G \quad \rightsquigarrow^*_\theta \quad \top$$

be an NC-refutation with the property that $x\theta = f(t_1, \ldots, t_n)$ for some $x \in \mathcal{V}ar(G)$ and let V be a finite set of variables such that $\mathcal{V}ar(G) \subseteq V$. Let $\gamma = \{x \mapsto f(x_1, \ldots, x_n)\}$ with $x_1, \ldots, x_n \notin V$. There exists an NC-refutation

$$\phi_{[im]}(\Pi):\quad G\gamma \quad \rightsquigarrow^*_{\theta'} \quad \top$$

which employs the same rewrite rules at the same positions in the corresponding equations of the goals in Π such that $\gamma\theta' = \theta\ [V]$.

PROOF. Define the substitution δ as the (disjoint) union of $\theta\lceil_{V-\{x\}}$ and $\{x_i \mapsto t_i \mid 1 \leqslant i \leqslant n\}$. We clearly have $\gamma\delta = \theta\ [V]$ and thus $\gamma \leqslant \theta\ [V]$. An application of Lemma 4.5 yields the desired refutation $\phi_{[im]}(\Pi)$. $\square$

The fifth and final transformation step is presented in the following lemma.

LEMMA 4.7. *For every NC-refutation*

$$\Pi:\quad G_1, s \approx t, G_2 \quad \rightsquigarrow_{\theta_1, \epsilon} \quad (G_1, \mathtt{true}, G_2)\theta_1 \quad \rightsquigarrow^*_{\theta_2} \quad \top$$

there exists an NC-refutation

$$\phi_{\mathrm{UC}}(\Pi):\quad (G_1, G_2)\theta_1 \quad \rightsquigarrow^*_{\theta_2} \quad \top.$$

PROOF. Consider the subderivation $\Pi'\colon (G_1, \mathtt{true}, G_2)\theta_1 \rightsquigarrow^*_{\theta_2} \top$ of Π. Simply dropping a single occurrence of $\mathtt{true}$ in every goal of Π' yields the desired NC-refutation $\phi_{\mathrm{UC}}(\Pi)$. $\square$

The idea now is to repeatedly apply the above transformation steps to a given NC-refutation, connecting the initial goals of (some of) the resulting NC-refutations by LNC-steps, until we reach the empty goal. In order to guarantee termination of this process, we need a well-founded order on NC-refutations that is compatible with the (last four) transformation steps. One of the components of our well-founded order is a *multiset order*. A *multiset* over a set A is an unordered collection of elements of A in which elements may have multiple occurrences. Every (strict) partial order $\succ$ on A can be extended to a partial order $\succ_{mul}$ on the set of finite multisets over A as follows: $M \succ_{mul} N$ if there exist multisets X and Y such that $\varnothing \neq X \subseteq M$, $N = (M - X) \cup Y$, and for every $y \in Y$ there exists an $x \in X$ such that $x \succ y$. Dershowitz and Manna [3] showed that multiset extension preserves well-foundedness.

DEFINITION 4.8. The *depth* $|t|$ of a term t is inductively defined as follows:

$$|t| = \begin{cases} 1 & \text{if } t \text{ is a variable,} \\ 1 + \max\{|t_1|, \ldots, |t_n| \mid 1 \leqslant i \leqslant n\} & \text{if } t = f(t_1, \ldots, t_n). \end{cases}$$

The *complexity* $|\Pi|$ of an NC-refutation $\Pi\colon G \rightsquigarrow_\theta^* \top$ is defined as the triple (n, M, s) where n is the number of applications of narrowing in Π at non-root positions (so the number of steps that do not use the rewrite rule $x \approx x \to \mathtt{true}$), M is the multiset

$$\{|x_1\theta|, \ldots, |x_m\theta| \mid \{x_1, \ldots, x_m\} \text{ is the } \textit{multiset} \text{ of variables occurring in } G\},$$

and s is the number of occurrences of symbols different from $\approx$ and $\mathtt{true}$ in G. We define a partial order $\gg$ on NC-refutations as follows: $\Pi_1 \gg \Pi_2$ if

$$|\Pi_1| \;\; lex(>, >_{mul}, >) \;\; |\Pi_2|.$$

Here $lex(>, >_{mul}, >)$ denotes the lexicographic product of $>$ (the standard order on $\mathbb{N}$), $>_{mul}$, and $>$.

Let M be a multiset $\{t_1, \ldots, t_n\}$ of terms. We abbreviate the multiset $\{t_1\sigma, \ldots, t_n\sigma\}$ to $M\sigma$ and $\{|t_1|, \ldots, |t_n|\}$ to $|M|$.

LEMMA 4.9. *The partial order $\gg$ is a well-founded order on NC-refutations.*

PROOF. Both lexicographic product and multiset extension preserve well-foundedness. □

Our complexity measure on NC-refutations is different from the one in Hölldobler [9, p. 188]. Since we are concerned with one-directional term rewriting and narrowing (as opposed to bi-directional equational reasoning and paramodulation in [9]), our simpler definition suffices. The next lemma states that $\gg$ is compatible with the transformation steps defined above.

LEMMA 4.10. *Let Π be an NC-refutation.*
(1) *If Π is non-empty and δ a variable renaming then $|\Pi| = |\phi_\delta(\Pi)|$.*
(2) *Let $\alpha \in \{[on], [d], [im], \text{UC}\}$. If $\phi_\alpha(\Pi)$ is defined then $\Pi \gg \phi_\alpha(\Pi)$.*
PROOF.

(1) Consider the proof of Lemma 4.2. Clearly Π and $\phi_\delta(\Pi)$ have the same number of narrowing steps at non-root positions. Let X and Y be the multiset of all variables occurring in G and $G\delta$ respectively. Because δ is a variable renaming we have $Y = \{x\delta \mid x \in X\}$. Hence

$$\begin{aligned} |Y\delta^{-1}\theta| &= \{|y\delta^{-1}\theta| \mid y \in Y\} \\ &= \{|y\delta^{-1}\theta| \mid y \in \{x\delta \mid x \in X\}\} \\ &= \{|x\delta\delta^{-1}\theta| \mid x \in X\} \\ &= \{|x\theta| \mid x \in X\} \\ &= |X\theta|, \end{aligned}$$

so also the second components of the triples $|\Pi|$ and $|\phi_\delta(\Pi)|$ are the same. Clearly G and $G\delta$ have the same number of symbols different from $\approx$ and $\mathtt{true}$. We conclude that $|\Pi| = |\phi_\delta(\Pi)|$.

(2) According to the proof of Lemma 4.3 the NC-refutation $\phi_{[on]}(\Pi)$ has one less narrowing step at non-root positions than Π. According to the proof of Lemma 4.4 the number of narrowing steps at non-root positions in $\phi_{[d]}(\Pi)$ is the same in Π. Because the substitution produced in $\phi_{[d]}(\Pi)$ subsumes the substitution produced in Π for the initial variables, the second component of $|\phi_{[d]}(\Pi)|$ doesn't exceed the second component of $|\Pi|$. Finally, the initial goal of $\phi_{[d]}(\Pi)$ has less symbols different from $\approx$ and $\mathtt{true}$ than the initial goal of Π (viz. two occurrences of the function symbol f). Next we consider the case $\phi_{[im]}$. Let Π be the NC-refutation $G \leadsto_\theta^* \top$ and $\phi_{[im]}(\Pi)$ the NC-refutation $G\gamma \leadsto_{\theta'}^* \top$. Since both refutations have the same number of narrowing steps at non-root positions, it suffices to show that $M_\Pi >_{mul} M_{\phi_{[im]}(\Pi)}$. Here M_Π ($M_{\phi_{[im]}(\Pi)}$) denotes the second component of the triple $|\Pi|$ ($|\phi_{[im]}(\Pi)|$). Let δ be the disjoint union of $\theta\restriction_{\mathcal{V}ar(G)-\{x\}}$ and $\{x_i \mapsto t_i \mid 1 \leqslant i \leqslant n\}$. From (the proof of) Lemma 4.6 we learn that $\gamma\theta' = \theta = \gamma\delta\ [\mathcal{V}ar(G)]$. Let X be the multiset of all occurrences of the variable x in G, Y the multiset of the all other variable occurrences in G, and X_i for $1 \leqslant i \leqslant m$ the multiset of all occurrences of the variable x_i in $G\gamma$. We have

$$M_\Pi = |X\theta| \cup |Y\theta|$$

and

$$M_{\phi_{[im]}(\Pi)} = |X_1\theta'| \cup \ldots \cup |X_m\theta'| \cup |Y\theta'|.$$

For all $i \in \{1, \ldots, m\}$ we have $x_i\theta' = x_i\gamma\theta' = x_i\gamma\delta = x_i\delta = t_i$ and hence $|x_i\theta'| = |t_i| < |f(t_1, \ldots, t_m)| = |x\theta|$. Therefore $|X\theta| >_{mul} |X_1\theta'| \cup \ldots \cup |X_m\theta'|$. For any $y \in Y$ we have $y\theta' = y\gamma\theta' = y\gamma\delta = y\theta$ and thus $|y\theta'| = |y\theta|$. Hence $|Y\theta| = |Y\theta'|$. We conclude that $M_\Pi >_{mul} M_{\phi_{[im]}(\Pi)}$ and therefore $\Pi \gg \phi_{[im]}(\Pi)$. The final case is ϕ_{UC}. Let Π be the NC-refutation

$$G_1, e, G_2 \leadsto_{\theta_1} (G_1, \mathtt{true}, G_2)\theta_1 \leadsto_{\theta_2}^* \top.$$

We partition the variable occurrences in the initial goal G_1, e, G_2 into the following three multisets: X the multiset of all occurrences of variables in $\mathcal{V}ar(G_1, G_2)$ that belong to $\mathcal{D}(\theta_1)$, Y the multiset of all occurrences of variables in $\mathcal{V}ar(G_1, G_2)$ that do not belong to $\mathcal{D}(\theta_1)$, and Z the multiset of all variable occurrences in e. We have

$$M_\Pi = |X\theta_1\theta_2| \cup |Y\theta_2| \cup |Z\theta_1\theta_2|.$$

Let X' be the multiset of all variable occurrences in the initial goal $(G_1, G_2)\theta_1$ of $\phi_{\mathrm{UC}}(\Pi)$ that are introduced by θ_1, so $X' \cup Y$ is the multiset of all variable occurrences in $(G_1, G_2)\theta_1$.

We have

$$M_{\phi_{\mathrm{UC}}(\Pi)} = |X'\theta_2| \cup |Y\theta_2|.$$

It is not difficult to see that $|X\theta_1\theta_2| >_{mul}^{=} |X'\theta_2|$. Hence $M_\Pi >_{mul}^{=} M_{\phi_{\mathrm{UC}}(\Pi)}$. Equality is only possible if $\theta_1\restriction_{\mathcal{V}ar(G_1,G_2)}$ is a variable substitution and $\mathcal{V}ar(e) = \varnothing$. (A *variable substitution* maps variables to variables.) Hence the number of symbols different from $\approx$ and true is exactly the same in the goals $(G_1, G_2)\sigma$ and G_1, G_2, which is clearly less than the number of such symbols in the goal G_1, e, G_2. Hence we always have $\Pi \gg \phi_{\mathrm{UC}}(\Pi)$.

□

The following example illustrates how the above results are used to transform NC-refutations into LNC-refutations.

EXAMPLE 4.11. Consider the TRS $\mathcal{R} = \{f(g(y)) \to y\}$ and the NC-refutation

$$\Pi_1\colon\ g(f(x)) \approx x \ \leadsto_{\{x \mapsto g(y)\}}\ g(y) \approx g(y) \ \leadsto_\epsilon\ \mathtt{true}.$$

In Π_1 the variable x is bound to $g(y)$, so the complexity of Π_1 is $(1, \{2, 2\}, 4)$. Transformation steps $\phi_{[on]}$, $\phi_{[d]}$, and ϕ_{UC} are not applicable to Π_1. Hence we try $\phi_{[im]}$. This yields the NC-refutation

$$\Pi_2 = \phi_{[im]}(\Pi_1)\colon\ g(f(g(x_1))) \approx g(x_1) \ \leadsto_{\{x_1 \mapsto y\}}\ g(y) \approx g(y) \ \leadsto_\epsilon\ \mathtt{true}$$

which has complexity $(1, \{1, 1\}, 6)$. Next we apply $\phi_{[d]}$. This gives the NC-refutation

$$\Pi_3 = \phi_{[d]}(\Pi_2)\colon\ f(g(x_1)) \approx x_1 \ \leadsto_{\{x_1 \mapsto y\}}\ y \approx y \ \leadsto_\epsilon\ \mathtt{true}$$

with complexity $(1, \{1, 1\}, 4)$. Observe that the initial goal of Π_1 is transformed into the initial goal of Π_3 by the single LNC-step $g(f(x)) \approx x \Rightarrow_{[im], \{x \mapsto g(x_1)\}} f(g(x_1)) \approx x_1$. In Π_3 narrowing is applied to the initial equation at position 1. This calls for the transformation step $\phi_{[on]}$:

$$\Pi_4 = \phi_{[on]}(\Pi_3)\colon\ f(g(x_1)) \approx f(g(y)), y \approx x_1 \ \leadsto_{\{x_1 \mapsto y\}}\ \mathtt{true}, y \approx y \ \leadsto_\epsilon\ \top.$$

NC-refutation Π_4 has complexity $(0, \{1, 1, 1, 1\}, 8)$. If we apply $\phi_{[d]}$ to Π_4, we obtain the NC-refutation

$$\Pi_5 = \phi_{[d]}(\Pi_4)\colon\ g(x_1) \approx g(y), y \approx x_1 \ \leadsto_{\{x_1 \mapsto y\}}\ \mathtt{true}, y \approx y \ \leadsto_\epsilon\ \top$$

with complexity $(0, \{1, 1, 1, 1\}, 6)$. The initial goals of Π_3 and Π_5 are connected by an $\Rightarrow_{[on]}$-step: $f(g(x_1)) \approx x_1 \Rightarrow_{[on]} g(x_1) \approx g(y), y \approx x_1$. In the first step of Π_5 narrowing is applied at the root position of the selected equation $g(x_1) \approx g(y)$, so the terms $g(x_1)$ and $g(y)$ are unifiable. A most general unifier is obtained by an application of $\Rightarrow_{[d]}$ followed by an application of $\Rightarrow_{[v]}$. So first we use $\phi_{[d]}$, yielding the NC-refutation

$$\Pi_6 = \phi_{[d]}(\Pi_5)\colon\ x_1 \approx y, y \approx x_1 \ \leadsto_{\{x_1 \mapsto y\}}\ \mathtt{true}, y \approx y \ \leadsto_\epsilon\ \top$$

with complexity $(0, \{1, 1, 1, 1\}, 4)$. Next we use ϕ_{UC}, yielding the NC-refutation

$$\Pi_7 = \phi_{\mathrm{UC}}(\Pi_6)\colon\ y \approx y \ \leadsto_\epsilon\ \mathtt{true}$$

with complexity $(0, \{1,1\}, 2)$. The initial goals of Π_5, Π_6, and Π_7 are connected by the UC-derivation $g(x_1) \approx g(y), y \approx x_1 \Rightarrow_{[d]} x_1 \approx y, y \approx x_1 \Rightarrow_{[v],\{x_1 \mapsto y\}} y \approx y$. Another application of ϕ_{UC} results in the empty NC-refutation

$$\Pi_8 = \phi_{\mathrm{UC}}(\Pi_7): \quad \square$$

which has complexity $(0, \varnothing, 0)$. Clearly $y \approx y \Rightarrow_{[t]} \square$. Concatenating the various LNC-sequences yields an LNC-refutation $g(f(x)) \approx x \Rightarrow^*_\theta \square$ whose substitution θ satisfies $x\theta = g(y)$.

Unfortunately, the simulation of NC by LNC illustrated above doesn't always work, as shown in the following example.

EXAMPLE 4.12. Consider the TRS

$$\mathcal{R} = \begin{cases} f(x) & \rightarrow & x \\ a & \rightarrow & b \\ b & \rightarrow & g(b) \end{cases}$$

and the NC-refutation

$$\Pi_{fail}: \; f(a) \approx g(a) \;\rightsquigarrow\; f(a) \approx g(b) \;\rightsquigarrow\; a \approx g(b) \;\rightsquigarrow\; b \approx g(b) \;\rightsquigarrow\; g(b) \approx g(b) \;\rightsquigarrow\; \mathtt{true}.$$

Because we apply narrowing at position 1 in the descendant $f(a) \approx g(b)$ of the initial equation $f(a) \approx g(a)$, using the rewrite rule $f(x) \rightarrow x$, we transform Π_{fail} using $\phi_{[on]}$ and $\phi_{[d]}$. This yields the NC-refutation $\phi_{[d]}(\phi_{[on]}(\Pi_{fail}))$:

$$a \approx x, x \approx g(a) \;\rightsquigarrow\; a \approx x, x \approx g(b) \;\rightsquigarrow\; \mathtt{true}, a \approx g(b) \;\rightsquigarrow\; \mathtt{true}, b \approx g(b) \;\rightsquigarrow\; \mathtt{true}, g(b) \approx g(b) \;\rightsquigarrow\; \top.$$

Observe that the initial goals of Π_{fail} and $\phi_{[d]}(\phi_{[on]}(\Pi_{fail}))$ are connected by $\Rightarrow_{[on]}$. Since in the refutation $\phi_{[d]}(\phi_{[on]}(\Pi_{fail}))$ narrowing is applied at position 1 in the descendant $a \approx g(b)$ of the selected equation $x \approx g(a)$ in the initial goal $a \approx x, x \approx g(a)$, we would like to use once more the transformation steps $\phi_{[on]}$ and $\phi_{[d]}$. This is however impossible since the subterm of $x \approx g(a)$ at position 1 is a variable.

The reason why Π_{fail} cannot be transformed to an LNC-refutation by the transformation steps in this section is that in $\phi_{[d]}(\phi_{[on]}(\Pi_{fail}))$ narrowing is applied to a subterm introduced by a previous narrowing substitution. One might be tempted to think that this problem cannot occur if we restrict ourselves to normalized solutions. This is not true, however, because Π_{fail} computes the empty substitution ε, which is clearly normalized, but $\phi_{[d]}(\phi_{[on]}(\Pi_{fail}))$ computes the non-normalized solution $\{x \mapsto a\}$. So the transformation steps do not preserve *normalization* of the computed NC-solutions (restricted to the variables in the initial goal). However, it turns out that *basicness* (cf. Definition 2.7) is preserved. This is one of the two key observations to the connection between strong completeness of LNC and completeness of basic NC.

LEMMA 4.13. *Let Π be a basic* NC*-refutation. The* NC*-refutations $\phi_{[on]}(\Pi)$, $\phi_{[d]}(\Pi)$, $\phi_{[im]}(\Pi)$, and $\phi_{\mathrm{UC}}(\Pi)$ are basic whenever they are defined. If Π is non-empty and δ a variable renaming then $\phi_\delta(\Pi)$ is basic.*

PROOF. The transformation ϕ_{UC} trivially preserves basicness. It is not difficult to see that narrowing is never applied to subterms introduced by previous narrowing substitutions in $\phi_{[d]}(\Pi)$ and $\phi_{[on]}(\Pi)$ whenever this is true for Π. Hence $\phi_{[d]}(\Pi)$ and $\phi_{[on]}(\Pi)$ are basic provided that Π is basic. Next we consider $\phi_{[im]}$. From Lemma 4.5 we learn that $\phi_{[im]}(\Pi)$ and Π employ the same rewrite rules at the same positions in the corresponding equations of the goals. Hence $\phi_{[im]}(\Pi)$ inherits basicness from Π. This reasoning also applies to $\phi_{\delta}(\Pi)$. □

The other key observation is that for basic NC, strong completeness and completeness coincide. This is an easy consequence of the following switching lemma, whose proof can be found in the appendix.

LEMMA 4.14. *For every* NC*-derivation*

$$\begin{aligned} G_1, e_1, G_2, e_2, G_3 &\leadsto_{p_1,\sigma_1,l_1\to r_1,e_1} (G_1, e_1[r_1]_{p_1}, G_2, e_2, G_3)\sigma_1 \\ &\leadsto_{p_2,\sigma_2,l_2\to r_2,e_2\sigma_1} (G_1, e_1[r_1]_{p_1}, G_2, e_2[r_2]_{p_2}, G_3)\sigma_1\sigma_2 \end{aligned}$$

with $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$ *there exists an* NC*-derivation*

$$\begin{aligned} G_1, e_1, G_2, e_2, G_3 &\leadsto_{p_2,\sigma_2',l_2\to r_2,e_2} (G_1, e_1, G_2, e_2[r_2]_{p_2}, G_3)\sigma_2' \\ &\leadsto_{p_1,\sigma_1',l_1\to r_1,e_1\sigma_2'} (G_1, e_1[r_1]_{p_1}, G_2, e_2[r_2]_{p_2}, G_3)\sigma_2'\sigma_1' \end{aligned}$$

such that $\sigma_1\sigma_2 = \sigma_2'\sigma_1'$. □

Observe that the requirement $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$ in Lemma 4.14 is always satisfied if the two steps are part of a basic narrowing derivation. Moreover, the exchange of the two steps preserves basicness. This is used in the proof below.

LEMMA 4.15. *Let $\mathcal{S}$ be a selection function. For every basic* NC*-refutation* $\Pi\colon G \leadsto_{\theta}^{*} \top$ *there exists a basic* NC*-refutation* $\Pi_{\mathcal{S}}\colon G \leadsto_{\theta}^{*} \top$ *respecting $\mathcal{S}$ with the same complexity.*

PROOF. Using the basicness of the given NC-refutation Π, we can transform Π into a basic refutation $\Pi_{\mathcal{S}}\colon G \leadsto_{\theta}^{*} \top$ that respect $\mathcal{S}$ by a finite number of applications of Lemma 4.14. Since the transformation in Lemma 4.14 preserves the number of narrowing steps at non-root positions, it follows that the complexities of Π and $\Pi_{\mathcal{S}}$ are the same. □

Now we can state and prove the main result of this section.

THEOREM 4.16. *Let $\mathcal{R}$ be a TRS and $G \leadsto_{\theta}^{*} \top$ a basic* NC*-refutation. For every selection function $\mathcal{S}$ there exists an* LNC*-refutation $G \Rightarrow_{\theta'}^{*} \Box$ respecting $\mathcal{S}$ such that $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$.*

PROOF. We use well-founded induction on the complexity of the given basic NC-refutation $\Pi\colon G \leadsto_{\theta}^{*} \top$, which is possible because of Lemma 4.9. In order to make the induction work we prove $\theta' \leqslant \theta\ [V]$ for a finite set of variables V that includes $\mathcal{V}ar(G)$ instead of $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$. The base case is trivial: G must be the empty goal. For the induction step we proceed as follows. First we use Lemma 4.15 to transform Π into a basic NC-refutation $\Pi_{\mathcal{S}}\colon G \leadsto_{\theta}^{*} \top$ respecting $\mathcal{S}$ with equal complexity. Next we show the existence of an LNC-step $\Pi'\colon G \Rightarrow_{\theta_2} G'$, an NC-refutation $\Pi_1\colon G' \leadsto_{\theta_1}^{*} \top$ of smaller complexity than $\Pi_{\mathcal{S}}$, and a finite set of variables V' such that $(V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\restriction_V) \subseteq V'$, $\mathcal{V}ar(G') \subseteq V'$, and $\theta_2\theta_1 \leqslant \theta\ [V]$. We distinguish the following cases, depending on what happens to the selected equation $e = \mathcal{S}(G)$ in the first step of $\Pi_{\mathcal{S}}$. Let $G = G_1, e, G_2$ and $e = s \approx t$.

(1) Suppose narrowing is applied to e at root position. We may write

$$\Pi_S\colon\ G \ \rightsquigarrow_{\theta_0,\epsilon,x\approx x\to \mathtt{true}} \ (G_1,\mathtt{true},G_2)\theta_0 \ \rightsquigarrow^*_{\theta_1'} \ \top$$

with $\theta_0\theta_1' = \theta$. We may assume that x is a fresh variable, so $x \notin V$. We distinguish the following two cases.

(a) Suppose $s \in \mathcal{V}$ or $t \in \mathcal{V}$. We distinguish two further cases, depending on whether or not s and t are equal.

(i) Suppose s and t are the same variable, say y. Let G' be the goal G_1, G_2 and G'' the goal $(G_1,G_2)\theta_0$. We clearly have $G \Rightarrow_{[t]} G'$. If G' is the empty goal $\square$ then we let Π_1 be the empty NC-refutation (and thus $\theta_1 = \varepsilon$) and we define $\theta_2 = \varepsilon$ and $V' = V$. Otherwise we proceed as follows. From Lemma 4.7 we obtain the (non-empty) NC-refutation

$$\Pi_1' = \phi_{\mathrm{UC}}(\Pi_S)\colon\ G'' \ \rightsquigarrow^+_{\theta_1'} \ \top.$$

Define $\theta_2' = \{x \mapsto y\}$. Clearly θ_2' is a most general unifier of the equations $x \approx x$ and $y \approx y$. Since also θ_0 is a most general unifier of these equations, there exists a variable renaming δ such that $\theta_0\delta = \theta_2'$. Let $\theta_1 = \delta^{-1}\theta_1'$. From Lemma 4.2 we obtain an NC-refutation

$$\phi_\delta(\Pi_1')\colon\ (G_1,G_2)\theta_0\delta \ \rightsquigarrow^+_{\theta_1} \ \top.$$

Because $x \notin V$ we have $\theta_1 = \theta_2'\theta_1 = \theta_0\delta\delta^{-1}\theta_1' = \theta_0\theta_1' = \theta\ [V]$. Moreover, $(G_1,G_2)\theta_0\delta = (G_1,G_2)\theta_2' = G'$. Hence we can take $\Pi_1 = \phi_\delta(\Pi_1')$, $\theta_2 = \varepsilon$, and $V' = V$.

(ii) Suppose $s \neq t$. We assume that $s \in \mathcal{V}$, say y. (The case $t \in \mathcal{V}$ is similar.) Let θ_2 be the substitution $\{y \mapsto t\}$ and G' the goal $(G_1,G_2)\theta_2$. We clearly have $G \Rightarrow_{[v],\theta_2} G'$. Since $x \notin \mathcal{D}(\theta_2)$, $\theta_2' = \theta_2 \cup \{x \mapsto t\}$ is a well-defined substitution. Clearly θ_2' is a unifier of the equations e and $x \approx x$. It is not too difficult to show that θ_2' is a most general unifier of these two equations. Since also θ_0 is a most general unifier of e and $x \approx x$, there exists a variable renaming δ such that $\theta_2' = \theta_0\delta$. If G' is the empty goal $\square$ then we let Π_1 be the empty NC-refutation (and thus $\theta_1 = \varepsilon$) and we define $V' = (V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\restriction_V)$. In this case we have $\theta_2\theta_1 = \theta_2 = \theta_2' \leqslant \theta_0 = \theta\ [V]$. If G' is not the empty goal, we reason as follows. Let $G'' = (G_1,G_2)\theta_0$. From Lemma 4.7 we obtain the NC-refutation

$$\Pi_1' = \phi_{\mathrm{UC}}(\Pi_S)\colon\ G'' \ \rightsquigarrow^+_{\theta_1'} \ \top.$$

According to Lemma 4.2 Π_1' can be transformed into the NC-refutation

$$\Pi_1 = \phi_\delta(\Pi_1')\colon\ G''\delta \ \rightsquigarrow^+_{\theta_1} \ \top$$

with $\theta_1 = \delta^{-1}\theta_1'$. Observe that $G''\delta = (G_1,G_2)\theta_2' = (G_1,G_2)\theta_2$ because x does not occur in G_1, G_2. We have $\theta_2'\theta_1 = \theta_0\delta\delta^{-1}\theta_1' = \theta_0\theta_1' = \theta$. Since $\theta_2 = \theta_2'\ [V]$ we obtain $\theta_2\theta_1 = \theta\ [V]$. Define $V' = (V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\restriction_V)$. Clearly $\mathcal{V}ar(G') \subseteq V'$.

(b) Suppose neither s nor t is a variable. We may write $s = f(s_1,\ldots,s_n)$ and $t = f(t_1,\ldots,t_n)$. Let G' be the goal $G_1, s_1 \approx t_1, \ldots, s_n \approx t_n, G_2$ We clearly have $G \Rightarrow_{[d]} G'$. According to Lemma 4.4 there exists an NC-refutation

$$\Pi_1 = \phi_{[d]}(\Pi_S)\colon\ G' \ \rightsquigarrow^*_{\theta_1} \ \top$$

such that $\theta_1 \leqslant \theta\ [V]$. Define $\theta_2 = \varepsilon$ and $V' = V$.

(2) Suppose narrowing is not applied to e at root position. We distinguish the following three cases.

(a) Suppose narrowing is applied to a descendant of $s \approx t$ at position 1. Let $f(l_1,\ldots,l_n) \to r$

be the used rewrite rule the first time this happens. Since $\Pi_{\mathcal{S}}$ is basic s cannot be variable for otherwise narrowing would be applied to a subterm introduced by previous narrowing substitutions. Hence we may write $s = f(s_1, \ldots, s_n)$. Let G'' be the goal

$$G_1, f(s_1, \ldots, s_n) \approx f(l_1, \ldots, l_n), r \approx t, G_2$$

and let G' be $G_1, s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, G_2$. We have the LNC-step $G \Rightarrow_{[on]} G'$. Applying Lemma 4.3 to $\Pi_{\mathcal{S}}$ yields an NC-refutation

$$\Pi_2 = \phi_{[on]}(\Pi)\colon\ G'' \ \leadsto^*_{\theta'}\ \top$$

such that $\theta' = \theta\ [V]$. An application of Lemma 4.4 to Π_2 results in an NC-refutation

$$\Pi_1 = \phi_{[d]}(\Pi_2)\colon\ G' \ \leadsto^*_{\theta_1}\ \top$$

such that $\theta_1 \leqslant \theta'\ [V]$. Define $\theta_2 = \varepsilon$ and $V' = V \cup \mathcal{V}ar(f(l_1, \ldots, l_n))$.

(b) Suppose narrowing is applied to some descendant of $s \approx t$ at position 2. The basic NC-refutation $\Pi_{\mathcal{S}}$ can be transformed into a basic NC-refutation

$$\Pi'_{\mathcal{S}}\colon\ G_1, t \approx s, G_2 \ \leadsto^*_{\theta}\ \top$$

by simply swapping the two sides in every descendant of e. This simple transformation doesn't affect the complexity. Now we apply case (2)(a) to $\Pi'_{\mathcal{S}}$.

(c) Suppose narrowing is never applied to a descendant of $s \approx t$ at position 1 or 2. We furthermore distinguish the following three cases.

(i) Suppose $s, t \notin \mathcal{V}$. We must have $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$, hence we can repeat case (1)(b).

(ii) Suppose $s \notin \mathcal{V}$ and $t \in \mathcal{V}$. Write $s = f(s_1, \ldots, s_n)$ and let t be the variable x. Let θ_2 be the substitution $\{x \mapsto f(x_1, \ldots, x_n)\}$ where $x_1, \ldots, x_n$ are fresh variables. Define the goal G'' as $(G_1, f(s_1, \ldots, s_n) \approx f(x_1, \ldots, x_n), G_2)\theta_2$ and let G' be the goal $(G_1, s_1 \approx x_1, \ldots, s_n \approx x_n, G_2)\theta_2$. We have $G \Rightarrow_{[im], \theta_2} G'$. Since $x\theta$ is of the form $f(t_1, \ldots, t_n)$ we can apply Lemma 4.6 to $\Pi_{\mathcal{S}}$, resulting in the NC-refutation

$$\Pi_2 = \phi_{[im]}(\Pi_{\mathcal{S}})\colon\ G'' \ \leadsto^*_{\theta_0}\ \top$$

such that $\theta_2\theta_0 = \theta\ [V]$. Define $V' = (V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\restriction_V)$. Clearly $\mathcal{V}ar(G'') = \mathcal{V}ar(G') \subseteq V'$. Next we apply Lemma 4.4 to Π_2. This yields the NC-refutation

$$\Pi_1 = \phi_{[d]}(\Pi_2)\colon\ G' \ \leadsto^*_{\theta_1}\ \top$$

with $\theta_1 \leqslant \theta_0\ [V']$. From Lemma 2.2(1) we obtain $\theta_2\theta_0 \leqslant \theta_2\theta_1\ [V]$. Combining this with $\theta_2\theta_0 = \theta\ [V]$ yields $\theta_2\theta_1 \leqslant \theta\ [V]$.

(iii) Suppose $s \in \mathcal{V}$ and $t \notin \mathcal{V}$. In this case we transform $\Pi_{\mathcal{S}}$ into the basic NC-refutation

$$\Pi'_{\mathcal{S}}\colon\ G_1, t \approx s, G_2 \ \leadsto^*_{\theta}\ \top$$

by simply swapping the two sides in every descendant of e. This simple transformation doesn't affect the complexity. Now we apply case (2)(c)(ii) to $\Pi'_{\mathcal{S}}$.

It is not possible that both s and t are variables, due to the basicness of $\Pi_{\mathcal{S}}$. (The case $s, t \in \mathcal{V}$ is covered in (1)(a).)

In all cases we obtained Π_1 from $\Pi_{\mathcal{S}}$ by applying one or two transformation steps $\phi_{[on]}$, $\phi_{[d]}$, $\phi_{[im]}$, ϕ_{UC} together with an additional application of ϕ_{δ} in case (1)(a)(i) and (1)(a)(ii). According to Lemma 4.10 Π_1 has smaller complexity than $\Pi_{\mathcal{S}}$. According to Lemmata 4.13 Π_1 is basic. Hence we can apply the induction hypothesis. This yields an LNC-refutation $\Pi''\colon G' \Rightarrow^*_{\theta_3} \square$ respecting $\mathcal{S}$ such that $\theta_3 \leqslant \theta_1\ [V']$. Now define $\theta' = \theta_2\theta_3$. From $\theta_2\theta_1 \leqslant \theta\ [V]$, $\theta_3 \leqslant \theta_1\ [V']$, and $(V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\restriction_V) \subseteq V'$, we infer—using Lemma 2.2(2)—that $\theta' \leqslant \theta\ [V]$. Concatenating the LNC-step Π' and the LNC-refutation Π'' yields the desired LNC-refutation. □

A related result for lazy paramodulation calculi is given by Moser [18]. He showed the completeness of his calculus $\mathcal{T}_{BP}$, a refined version of the calculus $\mathcal{T}$ of Gallier and Snyder [5], by a reduction to the basic superposition calculus $\mathcal{S}$ of [1]. Strong completeness (of $\mathcal{T}_{BP}$) follows because $\mathcal{T}_{BP}$ satisfies the so-called "switching lemma" ([13]). Since from every $\mathcal{T}_{BP}$-refutation one easily extracts a $\mathcal{T}$-refutation respecting the same selection function, strong completeness of $\mathcal{T}$ is an immediate consequence.

Combining Theorem 4.16 with Theorem 2.8 yields the following result.

COROLLARY 4.17. *Let $\mathcal{R}$ be a confluent TRS and $\mathcal{S}$ a selection function. If $\mathcal{R} \vdash G\theta$ and $\theta\restriction_{\mathcal{V}ar(G)}$ is normalized then there exists an* LNC*-refutation $G \Rightarrow^*_{\theta'} \Box$ respecting $\mathcal{S}$ such that $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$, provided one of the following conditions is satisfied:*
(1) *$\mathcal{R}$ is terminating,*
(2) *$\mathcal{R}$ is orthogonal and $G\theta$ has an $\mathcal{R}$-normal form, or*
(3) *$\mathcal{R}$ is right-linear.*
□

The converse of Theorem 4.16 does not hold, as witnessed by the confluent TRS

$$\mathcal{R} = \begin{cases} f(x) & \to & g(x,x) \\ a & \to & b \\ g(a,b) & \to & c \\ g(b,b) & \to & f(a) \end{cases}$$

from Middeldorp and Hamoen [17]. They show that the goal $f(a) \approx c$ cannot be solved by basic narrowing. Straightforward calculations reveal that for any selection function $\mathcal{S}$ there exists an LNC-refutation $f(a) \approx c \Rightarrow^* \Box$ respecting $\mathcal{S}$.

5. Completeness

In this section we show the completeness of LNC for confluent TRSs with respect to normalized solutions. Actually we show a stronger result: all normalized solutions are subsumed by substitutions produced by LNC-refutations that respect $\mathcal{S}_{left}$. Basic narrowing is of no help because of its incompleteness [17] for this general case. If we are able to define a class of NC-refutations respecting $\mathcal{S}_{left}$ that
(1) includes all NC-refutations respecting $\mathcal{S}_{left}$ that produce normalized solutions and
(2) which is closed under the transformation steps ϕ_δ, $\phi_{[on]}$, $\phi_{[d]}$, $\phi_{[im]}$, and ϕ_{UC},
then completeness with respect to $\mathcal{S}_{left}$ follows along the lines of the proof of Theorem 4.16. We didn't succeed in defining such a class, the main problem being the fact that an application of $\phi_{[on]}$ or $\phi_{[d]}$ to an NC-refutation that respects $\mathcal{S}_{left}$ may result in an NC-refutation that doesn't respect $\mathcal{S}_{left}$. We found however a class of NC-refutations respecting $\mathcal{S}_{left}$ that satisfies the first property and which is closed under ϕ_δ, $\phi_{[on]} \circ \phi_1$, $\phi_{[d]} \circ \phi_2$, $\phi_{[im]}$, and ϕ_{UC}. Here ϕ_1 and ϕ_2 are transformations that preprocess a given NC-refutation in such a way that a subsequent application of $\phi_{[on]}$ and $\phi_{[d]}$ results in an NC-refutation respecting $\mathcal{S}_{left}$. The following definition introduces our class of NC-refutations.

DEFINITION 5.1. An NC-refutation $\Pi\colon G \leadsto^*_\theta \top$ respecting $\mathcal{S}_{left}$ is called *normal* if it satisfies the following property: if narrowing is applied to the left-hand side (right-hand side) of a descendant of an equation $s \approx t$ in G then $\theta_2\lceil_{\mathcal{V}ar(s\theta_1)}$ $(\theta_2\lceil_{\mathcal{V}ar(t\theta_1)})$ is normalized. Here θ_1 and θ_2 are defined by writing Π as

$$G = G_1, s \approx t, G_2 \quad \rightsquigarrow^*_{\theta_1} \quad \top, (s \approx t, G_2)\theta_1 \quad \rightsquigarrow^*_{\theta_2} \quad \top.$$

The following result states that the class of normal NC-refutations satisfies property (1) mentioned above.

LEMMA 5.2. *Every* NC*-refutation respecting* $\mathcal{S}_{left}$ *that produces a normalized solution is normal.*
PROOF. Straightforward. □

The converse of this lemma is not true, see Example 5.7 below. Before introducing the transformations ϕ_1 and ϕ_2 we present a switching lemma which is used in the existence proofs. For the proof of this switching lemma we refer to the appendix.

LEMMA 5.3. *For every normal* NC*-refutation*

$$\begin{array}{lll} \Pi\colon\; e, G & \rightsquigarrow^*_{\theta_1} & e', G' \\ & \rightsquigarrow_{\sigma_1, p_1, l_1 \to r_1} & e'[r_1]_{p_1}\sigma_1, G'\sigma_1 \\ & \rightsquigarrow_{\sigma_2, p_2, l_2 \to r_2} & ((e'[r_1]_{p_1}\sigma_1)[r_2]_{p_2}, G'\sigma_1)\sigma_2 \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with $p_1 \perp p_2$ *there exists a normal* NC*-refutation*

$$\begin{array}{lll} \Pi'\colon\; e, G & \rightsquigarrow^*_{\theta_1} & e', G' \\ & \rightsquigarrow_{\sigma'_2, p_2, l_2 \to r_2} & (e'[r_2]_{p_2}, G')\sigma'_2 \\ & \rightsquigarrow_{\sigma'_1, p_1, l_1 \to r_1} & ((e'[r_2]_{p_2})[r_1]_{p_1}, G')\sigma'_2\sigma'_1 \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with the same complexity such that $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma'_2\sigma'_1\theta_2$. □

LEMMA 5.4. *For every normal* NC*-refutation*

$$\Pi\colon\; s \approx t, G \;\rightsquigarrow^*_{\theta_1}\; s' \approx t', G\theta_1 \;\rightsquigarrow_{\theta_2, 1, l \to r}\; (r \approx t', G\theta_1)\theta_2 \;\rightsquigarrow^*_{\theta_3}\; \top$$

with the property that narrowing is not applied to a descendant of $s \approx t$ *at position* 1 *in the subderivation that produces substitution* θ_1*, there exists a normal* NC*-refutation*

$$\phi_1(\Pi)\colon\; s \approx t, G \;\rightsquigarrow^*_{\theta'_1}\; s'' \approx t\theta'_1, G\theta'_1 \;\rightsquigarrow_{\theta'_2, 1, l \to r}\; (r \approx t\theta'_1, G\theta'_1)\theta'_2 \;\rightsquigarrow^*_{\theta'_3}\; \top$$

with the same complexity such that $\theta'_1\theta'_2\theta'_3 = \theta_1\theta_2\theta_3$ *and narrowing is neither applied at position* 1 *nor in the right-hand side of a descendant of* $s \approx t$ *in the subderivation that produces the substitution* θ'_1.
PROOF. Let Π' be the subderivation

$$s \approx t, G \;\rightsquigarrow^*_{\theta_1}\; s' \approx t', G\theta_1 \;\rightsquigarrow_{\theta_2, 1, l \to r}\; (r \approx t', G\theta_1)\theta_2$$

of Π. All steps in Π' take place in a descendant of $s \approx t$. If there are steps in Π' such that narrowing is applied to the right-hand side of the descendant of $s \approx t$ then there must be two consecutive steps in Π' such that the first one applies narrowing at the right-hand side and the second one at the left-hand side. The order of these two steps can be changed by an appeal to Lemma 5.3, resulting in a normal NC-refutation that has the same complexity and produces the same substitution as Π. This process is repeated until there are no more steps before the step in which position 1 is selected that apply narrowing at the right-hand side. Termination of this process is not difficult to see. We define $\phi_1(\Pi)$ as an outcome of this (non-deterministic) transformation process. □

LEMMA 5.5. *For every normal* NC*-refutation*

$$\Pi:\ f(s_1,\ldots,s_n)\approx f(t_1,\ldots,t_n),G \ \leadsto^*_{\theta_1}\ \mathtt{true},G\theta_1 \ \leadsto^*_{\theta_2}\ \top$$

with the property that narrowing is never applied to a descendant of $f(s_1,\ldots,s_n)\approx f(t_1,\ldots,t_n)$ *at position* 1 *or* 2*, there exists a normal* NC*-refutation*

$$\phi_2(\Pi):\ f(s_1,\ldots,s_n)\approx f(t_1,\ldots,t_n),G \ \leadsto^*_{\theta_1}\ \mathtt{true},G\theta_1 \ \leadsto^*_{\theta_2}\ \top$$

with the same complexity such that in the subderivation producing substitution θ_1 *narrowing is applied to the subterms* $s_1,\ldots,s_n,t_1,\ldots,t_n$ *in the order* $s_1,t_1,s_2,t_2,\ldots,s_n,t_n$.

PROOF. By a similar transformation process as in the proof of the preceding lemma. □

The next result states that the transformation steps ϕ_δ, $\phi_{[on]}\circ\phi_1$, $\phi_{[d]}\circ\phi_2$, $\phi_{[im]}$, and ϕ_{UC} preserve normality.

LEMMA 5.6. *Let* Π *be a normal* NC*-refutation. The* NC*-refutations* $\phi_{[on]}(\phi_1(\Pi))$, $\phi_{[d]}(\phi_2(\Pi))$, $\phi_{[im]}(\Pi)$, *and* $\phi_{\mathrm{UC}}(\Pi)$ *are normal whenever they are defined. If* Π *is non-empty and* δ *a variable renaming then* $\phi_\delta(\Pi)$ *is normal.*

PROOF. First we will show the normality of $\phi_{[on]}(\phi_1(\Pi))$. From Lemma 5.4 it follows that $\phi_1(\Pi)$, which we can write as

$$s\approx t,G \ \leadsto^*_{\theta_1}\ s'\approx t\theta_1,G\theta_1 \ \leadsto_{\theta_2,1,l\to r}\ (r\approx t\theta_1,G\theta_1)\theta_2 \ \leadsto^*_{\theta_3}\ \top,$$

is normal. This NC-refutation is transformed by $\phi_{[on]}$ into

$$\begin{array}{lll} s\approx l,r\approx t,G & \leadsto^*_{\theta_1} & s'\approx l,r\approx t\theta_1,G\theta_1 \\ & \leadsto_{\theta_2',x\approx x\to\mathtt{true}} & \mathtt{true},(r\approx t\theta_1,G\theta_1)\theta_2' = \mathtt{true},(r\approx t\theta_1,G\theta_1)\theta_2 \\ & \leadsto^*_{\theta_3} & \top. \end{array}$$

Here θ_2' is the substitution $\theta_2\cup\{x\mapsto l\theta_2\}$. We have to show that the condition of Definition 5.1 holds for every equation in the initial goal $s\approx l,r\approx t,G$ of the NC-refutation $\phi_{[on]}(\phi_1(\Pi))$. Consider the equation $s\approx l$. By construction $\phi_{[on]}(\phi_1(\Pi))$ doesn't contain steps in which narrowing is applied to l. Suppose there is a step in which narrowing is applied to the left-hand side of a descendant of s. (This is equivalent to saying that the derivation from $s\approx l$ to $s'\approx l$ is non-empty.) We have to show that $\theta_1\theta_2'\theta_3\restriction_{\mathcal{V}ar(s)}$ is normalized. Because in $\phi_1(\Pi)$ narrowing is applied to the left-hand side of a descendant of $s\approx t$, we obtain the normalization of $\theta_1\theta_2\theta_3\restriction_{\mathcal{V}ar(s)}$ from the normality of $\phi_1(\Pi)$. This implies that $\theta_1\theta_2'\theta_3\restriction_{\mathcal{V}ar(s)}$ is normalized since $\theta_1\theta_2\restriction_{\mathcal{V}ar(s)} = \theta_1\theta_2'\restriction_{\mathcal{V}ar(s)}$. By construction, the descendants of the equation $r\approx t$ and the equations in G are only selected in the common subrefutation $(r\approx t\theta_1,G\theta_1)\theta_2 \leadsto^*_{\theta_3} \top$ of $\phi_1(\Pi)$ and $\phi_{[on]}(\phi_1(\Pi))$. We conclude that $\phi_{[on]}(\phi_1(\Pi))$ is normal. Next we consider $\phi_{[d]}(\phi_2(\Pi))$.

According to Lemma 5.5 the transformation ϕ_2 preserves normality. It is easy to see that $\phi_{[d]}$ preserves normality and hence $\phi_{[d]}(\phi_2(\Pi))$ inherits normality from Π. Now consider $\phi_{[im]}$. Let $e = s \approx t$ be an equation in the initial goal of Π. We may write Π as

$$G_1, e, G_2 \quad \rightsquigarrow^*_{\theta_1} \quad \top, (e, G_2)\theta_1 \quad \rightsquigarrow^*_{\theta_2} \quad \top.$$

From the proof of Lemma 4.6 we learn that $\phi_{[im]}(\Pi)$ can be written as

$$(G_1, e, G_2)\gamma \quad \rightsquigarrow^*_{\theta'_1} \quad \top, (e, G_2)\theta_1\gamma' \quad \rightsquigarrow^*_{\theta'_2} \quad \top$$

with $\gamma'\theta'_2 = \theta_2\ [\mathcal{V}ar((e, G_2)\theta_1)]$. Suppose in $\phi_{[im]}(\Pi)$ narrowing is applied to the left-hand side of a descendant of $e\gamma$. In order to conclude normality, we have to show that $\theta'_2\lceil_{\mathcal{V}ar(s\theta_1\gamma')}$ is normalized. By construction, in Π narrowing is applied to the left-hand side of a descendant of e. The normalization of Π yields the normalization of $\theta_2\lceil_{\mathcal{V}ar(s\theta_1)}$. Because $s\theta_1$ occurs in $(e, G_2)\theta_1$ we obtain the normalization of $\gamma'\theta'_2\lceil_{\mathcal{V}ar(s\theta_1)}$. Lemma 2.3 yields the desired normalization of $\theta'_2\lceil_{\mathcal{V}ar(s\theta_1\gamma')}$. The transformation ϕ_{UC} trivially preserves normality. The remaining transformation $\phi_{\mathcal{S}}$ is easily seen to preserve normality. □

EXAMPLE 5.7. Consider again the NC-refutation Π_{fail} of Example 4.12. This refutation is easily seen to be normal. An application of $\phi_{[on]}$ results in the NC-refutation $\phi_{[on]}(\Pi_{fail})$:

$$\begin{array}{llllll} f(a) \approx f(x), x \approx g(a) & \rightsquigarrow & f(a) \approx f(x), x \approx g(b) & \rightsquigarrow & a \approx g(b) & \rightsquigarrow \ b \approx g(b) \\ & \rightsquigarrow & b \approx b & \rightsquigarrow & \mathtt{true} & \end{array}$$

which doesn't respect $\mathcal{S}_{left}$. If we first apply ϕ_1 we obtain the NC-refutation $\phi_1(\Pi_{fail})$:

$$f(a) \approx g(a) \ \rightsquigarrow \ a \approx g(a) \ \rightsquigarrow \ a \approx g(b) \ \rightsquigarrow \ b \approx g(b) \ \rightsquigarrow \ b \approx b \ \rightsquigarrow \ \mathtt{true}.$$

An application of $\phi_{[on]}$ to this normal NC-refutation yields $\phi_{[on]}(\phi_1(\Pi_{fail}))$:

$$f(a) \approx f(x), x \approx g(a) \ \rightsquigarrow \ a \approx g(a) \ \rightsquigarrow \ a \approx g(b) \ \rightsquigarrow \ b \approx g(b) \ \rightsquigarrow \ b \approx b \ \rightsquigarrow \ \mathtt{true}.$$

This NC-refutation is normal even though the produced substitution restricted to the variables in the initial goal is not normalized.

Lemma 5.9 below is the counterpart of Lemma 4.15 for normal NC-refutations. The proof is an easy consequence of the following switching lemma, whose proof can be found in the appendix.

LEMMA 5.8. *For every* NC-*refutation*

$$\begin{array}{lll} \Pi\colon & G_1, e_1, G_2, e_2, G_3 \ \rightsquigarrow^*_{\theta_1} & G'_1, e'_1, G'_2, e'_2, G'_3 \\ & \rightsquigarrow_{p_1, \sigma_1, l_1 \to r_1, e'_1} & (G'_1, e'_1[r_1]_{p_1}, G'_2)\sigma_1, e'_2\sigma_1, G'_3\sigma_1 \\ & \rightsquigarrow_{p_2, \sigma_2, l_2 \to r_2, e'_2\sigma_1} & ((G'_1, e'_1[r_1]_{p_1}, G'_2)\sigma_1, (e'_2\sigma_1)[r_2]_{p_2}, G'_3\sigma_1)\sigma_2 \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

that produces a normalized substitution there exists a NC-*refutation*

$$\begin{array}{lll} \Pi'\colon & G_1, e_1, G_2, e_2, G_3 \ \rightsquigarrow^*_{\theta_1} & G'_1, e'_1, G'_2, e'_2, G'_3 \\ & \rightsquigarrow_{p_2, \sigma'_2, l_2 \to r_2, e'_2} & (G'_1, e'_1, G'_2, e'_2[r_2]_{p_2}, G'_3)\sigma'_2 \\ & \rightsquigarrow_{p_1, \sigma'_1, l_1 \to r_1, e'_1\sigma'_2} & (G'_1, e'_1[r_1]_{p_1}, G'_2, e'_2[r_2]_{p_2}, G'_3)\sigma'_2\sigma'_1 \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with the same complexity such that $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma'_2\sigma'_1\theta_2$. □

LEMMA 5.9. *For every* NC*-refutation* $\Pi: G \leadsto_\theta^* \top$ *such that* $\theta\lceil_{\mathcal{V}ar(G)}$ *is normalized there exists a normal* NC*-refutation* $\Pi_{\mathcal{S}_{left}}: G \leadsto_\theta^* \top$ *with the same complexity.*

PROOF. Straightforward consequence of Lemma 5.8 and Lemma 5.2. □

Putting all pieces together, the following result can be proved along the lines of the proof of Theorem 4.16.

THEOREM 5.10. *Let* $\mathcal{R}$ *be a TRS. For every* NC*-refutation* $G \leadsto_\theta^* \top$ *with the property that* $\theta\lceil_{\mathcal{V}ar(G)}$ *is normalized there exists an* LNC*-refutation* $G \Rightarrow_{\theta'}^* \top$ *respecting* $\mathcal{S}_{left}$ *such that* $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$.

PROOF. First we apply Lemma 5.9 to the given NC-refutation. This results in a normal NC-refutation $\Pi: G \leadsto_\theta^* \top$. We use well-founded induction on the complexity of Π. In order to make the induction work we prove that for any finite set of variables V that includes $\mathcal{V}ar(G)$, there exists an LNC-refutation $G \Rightarrow_{\theta'}^* \square$ such that $\Pi': G \Rightarrow_{\theta_2}^* G'$, an NC-refutation $\theta' \leqslant \theta\ [V]$. The base case is trivial: G must be the empty goal. For the induction step we proceed as follows. We show the existence of an LNC-step $\Pi': G \Rightarrow_{\theta_2} G'$ respecting $\mathcal{S}_{left}$, a normal NC-refutation $\Pi_1: G' \leadsto_{\theta_1}^* \top$ of smaller complexity than Π, and a finite set of variables V' such that $(V - \mathcal{D}(\theta_2)) \cup \mathcal{I}(\theta_2\lceil_V) \subseteq V'$, $\mathcal{V}ar(G') \subseteq V'$, and $\theta_2\theta_1 \leqslant \theta\ [V]$. We distinguish the following cases, depending on what happens to the selected equation $e = \mathcal{S}_{left}(G)$ in the first step of Π. Let $G = e, G_1$ and $e = s \approx t$.

(1) Suppose narrowing is applied to e at root position. We may write

$$\Pi:\quad G \quad \leadsto_{\theta_0,\,\epsilon,\,x\approx x\to \mathtt{true}} \quad (\mathtt{true}, G_1)\theta_0 \quad \leadsto_{\theta_1'}^* \quad \top$$

with $\theta_0\theta_1' = \theta$. We may assume that x is a fresh variable, so $x \notin V$. We distinguish two cases.

   (a) Suppose $s \in \mathcal{V}$ or $t \in \mathcal{V}$. This case is only notationally different from case (1)(a) in the proof of Theorem 4.16.

   (b) Suppose neither s nor t is a variable. Apart from notation, the only difference with case (1)(b) in the proof of Theorem 4.16 is the replacement of $\phi_{[d]}$ by $\phi_{[d]} \circ \phi_2$.

(2) Suppose narrowing is not applied to e at root position. We distinguish the following three cases.

   (a) Suppose narrowing is applied to a descendant of e at position 1. The essential difference with case (2)(a) in the proof of Theorem 4.16 is the replacement of $\phi_{[d]} \circ \phi_{[on]}$ by $\phi_{[d]} \circ \phi_2 \circ \phi_{[on]} \circ \phi_1$ and the use of normality rather than basicness to conclude that s is not a variable.

   (b) Suppose narrowing is applied to some descendant of $s \approx t$ at position 2. This case is reduced to the previous one as in the proof of Theorem 4.16.

   (c) Suppose narrowing is never applied to a descendant of $s \approx t$ at position 1 or 2. The essential difference with case (2)(c) in the proof of Theorem 4.16 is the replacement of $\phi_{[d]}$ by $\phi_{[d]} \circ \phi_2$ in subcase (2)(c)(ii).

The proof is concluded by an appeal to Lemmata 4.10, 5.6, 5.4, and 5.5, followed by an application of the induction hypothesis, similar to the final part of the proof of Theorem 4.16. □

COROLLARY 5.11. *Let* $\mathcal{R}$ *be a confluent TRS. If* $\mathcal{R} \vdash G\theta$ *and* $\theta\lceil_{\mathcal{V}ar(G)}$ *is normalized then there exists an* LNC*-refutation* $G \Rightarrow_{\theta'}^* \square$ *respecting* $\mathcal{S}_{left}$ *such that* $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$. □

6. Eager Variable Elimination

LNC has three sources of non-determinism: the choice of the equation in the given goal, the choice of the inference rule, and the choice of the rewrite rule (in the case of [*on*]). In Section 4 we were concerned with the first kind of non-determinism. In this section we address the second kind of non-determinism. The non-deterministic application of the various inference rules to selected equations causes LNC to generate many redundant derivations. Consider for example the (orthogonal hence confluent) TRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} f(g(x)) & \to & a \\ b & \to & g(b) \end{array} \right.$$

Figure 1 shows all LNC-refutations issued from the goal $f(b) \approx a$ that respect the selection function $\mathcal{S}_{left}$. There are infinitely many such refutations. Because the initial goal is ground,

$$\begin{array}{ccccccc}
f(b) \approx a & & & & & & \\
\Downarrow_{[on]} & & & & & & \\
b \approx g(x), a \approx a & & & & & & \\
\Downarrow_{[on]} & & & & & & \\
g(b) \approx g(x), a \approx a & & & & & & \\
\Downarrow_{[d]} & & & & & & \\
b \approx x, a \approx a & \Rightarrow_{[on]} & g(b) \approx x, a \approx a & \Rightarrow_{[im]} & b \approx x_1, a \approx a & \Rightarrow_{[on]} & \cdots \\
\Downarrow_{[v]} & & \Downarrow_{[v]} & & \Downarrow_{[v]} & & \\
a \approx a & & a \approx a & & a \approx a & & \\
\Downarrow_{[d]} & & \Downarrow_{[d]} & & \Downarrow_{[d]} & & \\
\Box & & \Box & & \Box & &
\end{array}$$

FIGURE 1.

one of them suffices for completeness. At several places in the literature it is mentioned that this type of redundancy can be greatly reduced by applying the variable elimination rule [*v*] prior to other applicable inference rules, although to the best of our knowledge there is no supporting proof of this so-called *eager variable elimination problem* for the general case of confluent systems.

In this section we show that a restricted version of the eager variable elimination strategy is complete with respect to $\mathcal{S}_{left}$ for orthogonal TRSs. Before we can define our strategy, we need to extend the concept of descendant to LNC-derivations. Descendants of non-selected equations are defined as in Definition 4.1. The selected equation $f(s_1, \ldots, s_n) \simeq t$ in the outermost narrowing rule [*on*] has the body equation $r \approx t$ as only (one-step) descendant. In the imitation rule [*im*], all equations $s_i\theta \approx x_i$ $(1 \leqslant i \leqslant n)$ are descendants of the selected equation $f(s_1, \ldots, s_n) \simeq x$. The selected equation $f(s_1, \ldots, s_n) \simeq f(t_1, \ldots, t_n)$ in the decomposition rule [*d*] has all equations $s_1 \approx t_1, \ldots, s_n \approx t_n$ as (one-step) descendants. Finally, the selected equations in [*v*] and [*t*] have no descendants.

DEFINITION 6.1. An equation of the form $x \simeq t$, with $x \notin \mathcal{V}ar(t)$, is called *solved*. An LNC-derivation Π is called *eager* if the variable elimination rule [*v*] is applied to all selected solved equations that are descendants of a parameter-passing equation in Π.

Of the infinitely many LNC-refutations in Figure 1 only the leftmost one is eager since all others apply the outermost narrowing rule [*on*] to the solved descendant $b \approx x$ of the parameter-passing equation $b \approx g(x)$ introduced in the first $\Rightarrow_{[on]}$-step.

In this section we prove that eager LNC is complete with respect to $\mathcal{S}_{left}$ for orthogonal TRSs (with respect to normalized solutions). The outline of our proof is as follows.

(1) We define *outside-in* NC-derivations. These are the narrowing counterpart to the outside-in rewrite sequences of Huet and Lévy [10].
(2) We show that the completeness of outside-in NC for orthogonal TRSs with respect to normalized solutions is an easy consequence of Huet and Lévy's *standardization* theorem.
(3) We show that the translation steps $\phi_{\mathcal{S}}$, ϕ_1, ϕ_2, $\phi_{[on]}$, $\phi_{[d]}$, $\phi_{[im]}$, and ϕ_{UC} preserve the outside-in property.
(4) We verify that the LNC-refutation obtained from an outside-in NC-refutation by means of the transformation described in the previous section is in fact eager.

Before defining outside-in NC-derivations, we introduce the concept of NC-*trace*. Let $\Pi : G \rightsquigarrow_\theta G'$ be an NC-step and e an equation in G different from true. Let e' be the (unique) descendant of e in G'. The construct $e \rightarrowtail_\theta e'$ is called a one-step NC-trace. NC-traces are obtained by concatenating one-step NC-traces. An NC-trace

$$e_1 \rightarrowtail_{\theta_1} \cdots \rightarrowtail_{\theta_{n-1}} e_n$$

may be rendered as $e_1 \rightarrowtail_\theta^* e_n$ where $\theta = \theta_1 \cdots \theta_{n-1}$. For every such NC-trace π there is a corresponding rewrite sequence $\mathcal{R}(\pi)$: $e_1\theta \rightarrow_{\mathcal{R}_+}^* e_n$. This rewrite sequence will be shorter than π if the latter contains one-step NC-traces of the form $e_i \rightarrowtail_{\theta_i} e_i\theta_i$—indicating that e_i was not selected in the underlying NC-step—which translate to identity at the rewrite level.

DEFINITION 6.2. Let $\mathcal{R}$ be an orthogonal TRS. An $\mathcal{R}_+$-rewrite sequence

$$e_1 \rightarrow_{p_1, l_1 \rightarrow r_1} \cdots \rightarrow_{p_{n-1}, l_{n-1} \rightarrow r_{n-1}} e_n$$

is called *outside-in* if the following condition is satisfied for all $1 \leqslant i < n-1$: if there exists a j with $i < j < n$ such that $\epsilon < p_j < p_i$ then $p_i \backslash p_j \in \mathcal{P}os_{\mathcal{F}}(l_j)$ for the least such j.

This definition is equivalent to the one given by Huet and Lévy in their seminal paper [10] on call-by-need computations in orthogonal TRSs. The following result is an immediate consequence of their standardization theorem (Theorem 3.19 in [10]).

THEOREM 6.3. *Let $\mathcal{R}$ be an orthogonal TRS and e an equation. For every rewrite sequence $e \rightarrow_{\mathcal{R}_+}^*$* true *there exists an outside-in rewrite sequence $e \rightarrow_{\mathcal{R}_+}^*$* true. □

DEFINITION 6.4. Let $\mathcal{R}$ be an orthogonal TRS. An NC-derivation Π issued from a G is called *outside-in* if $\mathcal{R}(\pi)$ is outside-in for all traces π of the equations $e \in G$.

EXAMPLE 6.5. Consider the orthogonal TRS

$$\mathcal{R} = \left\{ \begin{array}{rcl} f(x) & \rightarrow & x \\ a & \rightarrow & b \end{array} \right.$$

The NC-refutation

$$\Pi\colon\; f(a)\approx y, f(y)\approx b \;\rightsquigarrow_{f(x)\to x}\; a\approx y, f(y)\approx b \;\rightsquigarrow_{f(x_1)\to x_1}\; a\approx y, y\approx b \;\rightsquigarrow_{\{y\mapsto a\}}\; \mathtt{true}, a\approx b \;\rightsquigarrow_{a\to b}\; \mathtt{true}, b\approx b \;\rightsquigarrow\; \top$$

is outside-in, because the rewrite sequences

$$\mathcal{R}(\pi_1)\colon\; f(a)\approx a \;\to\; a\approx a \;\to\; \mathtt{true}$$

and

$$\mathcal{R}(\pi_2)\colon\; f(a)\approx b \;\to\; a\approx b \;\to\; b\approx b \;\to\; \mathtt{true}$$

corresponding to the maximal NC-traces

$$\pi_1\colon\; f(a)\approx y \;\rightarrowtail\; a\approx y \;\rightarrowtail\; a\approx y \;\rightarrowtail_{\{y\mapsto a\}}\; \mathtt{true}$$

and

$$\pi_2\colon\; f(y)\approx b \;\rightarrowtail\; f(y)\approx b \;\rightarrowtail\; y\approx b \;\rightarrowtail_{\{y\mapsto a\}}\; a\approx b \;\rightarrowtail\; b\approx b \;\rightarrowtail\; \mathtt{true}$$

are outside-in. The NC-refutation

$$\Pi'\colon\; f(a)\approx y, f(y)\approx b \;\rightsquigarrow_{f(x)\to x}\; a\approx y, f(y)\approx b \;\rightsquigarrow_{\{y\mapsto a\}}\; \mathtt{true}, f(a)\approx b \;\rightsquigarrow_{a\to b}\; \mathtt{true}, f(b)\approx b \;\rightsquigarrow_{f(x_1)\to x_1}\; \mathtt{true}, b\approx b \;\rightsquigarrow\; \top$$

is not outside-in because the rewrite sequence

$$\mathcal{R}(\pi)\colon\; f(a)\approx b \;\to\; f(b)\approx b \;\to\; b\approx b \;\to\; \mathtt{true}$$

corresponding to the trace

$$\pi\colon\; f(y)\approx b \;\rightarrowtail\; f(y)\approx b \;\rightarrowtail_{\{y\mapsto a\}}\; f(a)\approx b \;\rightarrowtail\; f(b)\approx b \;\rightarrowtail\; b\approx b \;\rightarrowtail\; \mathtt{true}$$

of the second equation in the initial goal is not outside-in.

THEOREM 6.6. *Let $\mathcal{R}$ be an orthogonal TRS. For every* NC*-refutation $G \rightsquigarrow_\theta^* \top$ with $\theta\restriction_{\mathcal{V}ar(G)}$ normalized there exists an outside-in* NC*-refutation $G \rightsquigarrow_{\theta'}^* \top$ such that $\theta' \leqslant \theta\ [\mathcal{V}ar(G)]$.*

PROOF. Let Π be the given NC-refutation $G \rightsquigarrow_\theta^* \top$. By instantiation we obtain the corresponding rewrite sequence $\mathcal{R}(\Pi)\colon G\theta \to_{\mathcal{R}_+}^* \top$. Let $G\theta = e_1, \ldots, e_n$. Clearly $\mathcal{R}(\Pi)$ can be partitioned into rewrite sequences from e_i to $\mathtt{true}$ for $1 \leqslant i \leqslant n$. To each of these n rewrite sequences we apply Theorem 6.3, yielding outside-in rewrite sequences from e_i to $\mathtt{true}$ ($1 \leqslant i \leqslant n$). Putting these n outside-in rewrite sequences together results in a outside-in rewrite sequence from $G\theta$ to $\top$. Let $\theta_1 = \theta\restriction_{\mathcal{V}ar(G)}$. Evidently, $G\theta_1 = G\theta$ and θ_1 is normalized. An application of the lifting lemma (for NC) to the outside-in rewrite sequence $G\theta_1 \to_{\mathcal{R}_+}^* \top$ results in an outside-in NC-refutation $G \rightsquigarrow_{\theta'}^* \top$ with $\theta' \leqslant \theta_1 = \theta\ [\mathcal{V}ar(G)]$. □

The above theorem extends and simplifies the main result of You [23]: the completeness of *outer* narrowing for orthogonal *constructor-based* TRSs with respect to *constructor-based* solutions. One easily verifies that outer narrowing coincides with outside-in narrowing in the case of orthogonal constructor-based TRSs and that constructor-based substitutions are a special case of normalized substitutions. Hence You's completeness result (Theorem 3.13 in [23]) is a consequence of Theorem 6.6. Since You doesn't use the powerful standardization theorem of Huet and Lévy, his completeness proof is (much) more complicated than the proof presented above, which covers a larger class of TRSs.

LEMMA 6.7. *The transformations ϕ_δ, ϕ_1, ϕ_2, $\phi_{[on]}$, $\phi_{[d]}$, $\phi_{[im]}$, and ϕ_{UC} preserve the outside-in property.*

PROOF. Straightforward by inspecting the various transformations. □

LEMMA 6.8. *If $\Pi: G \leadsto^*_\theta \top$ is an outside-in* NC*-refutation such that $\theta\restriction_{\mathcal{V}ar(G)}$ is normalized, then $\Pi_{\mathcal{S}_{left}}$ is outside-in.*

PROOF. Let e be an arbitrary equation in G and let π and π' be the respective traces of e in Π and $\Pi_{\mathcal{S}_{left}}$. The lemma is an immediate consequence of the equality $\mathcal{R}(\pi) = \mathcal{R}(\pi')$. □

We define a property of equations in the initial goal of NC-refutations. In Lemma 6.10 we show that the parameter-passing equations introduced in the proof of Theorem 5.10 satisfy this property, provided we start from a normal NC-refutation that is outside-in. In Lemma 6.11 we show that the property is preserved by LNC-descendants obtained during the transformations in Theorem 5.10. Finally, in Lemma 6.12, we show that the variable elimination rule is applied to selected solved equations that satisfy the property in the LNC-refutation obtained in the proof of Theorem 5.10.

DEFINITION 6.9. Let $\Pi: G \leadsto^* \top$ be an NC-refutation and $e \in G$. We write $\mathcal{P}_\Pi(e)$ if the following two properties are satisfied:

(1) narrowing is not applied to the right-hand side of a descendant of e in Π, and
(2) if narrowing is applied to the left-hand side of a descendant of e in Π and $1{\cdot}p$ is a narrowing position in a descendant of e such that later steps in the left-hand side of descendants of e do not take place above $1{\cdot}p$, then $2{\cdot}p \in \mathcal{P}os_\mathcal{F}(e)$.

A position $1{\cdot}p$ satisfying the condition in property (2) will be called *critical.*

LEMMA 6.10. *Let $\Pi: G \leadsto^* \top$ be an outside-in normal* NC*-refutation such that narrowing is applied to a descendant of the (selected) leftmost equation in G at position 1. Let $\Pi': G' \leadsto^* \top$ be the* NC*-refutation $\phi_{[d]}(\phi_2(\phi_{[on]}(\phi_1(\Pi))))$. We have $\mathcal{P}_{\Pi'}(e)$ for all parameter-passing equations $e \in G'$.*

PROOF. Let e be a parameter-passing equation in G'. The first condition of $\mathcal{P}_{\Pi'}(e)$ holds by construction. Suppose narrowing is applied to the left hand side of a descendant of e in Π'. Let $1{\cdot}p$ be a critical position. We have to show that $2{\cdot}p \in \mathcal{P}os_\mathcal{F}(e)$. Let s be the left-hand side of e. The initial goal of $\phi_1(\Pi)$ has the form $f(s_1,\ldots,s,\ldots,s_n) \approx t, G$. Suppose s is the i-th argument of $f(s_1,\ldots,s,\ldots,s_n)$. The NC-refutation $\phi_1(\Pi)$ can be written as

$$\begin{array}{lll} f(s_1,\ldots,s,\ldots,s_n) \approx t, G & \leadsto^* & f(s'_1,\ldots,s',\ldots,s'_n) \approx t', G' \\ & \leadsto_{1\cdot i\cdot p} & f(s''_1,\ldots,s'',\ldots,s''_n) \approx t'', G'' \\ & \leadsto^* & f(s'''_1,\ldots,s''',\ldots,s'''_n) \approx t''', G''' \\ & \leadsto_{1,\sigma,f(l_1,\ldots,l_n)\to r} & (r \approx t''', G''')\sigma \\ & \leadsto^* & \top \end{array}$$

where all narrowing steps in the subderivation

$$f(s''_1,\ldots,s'',\ldots,s''_n) \approx t'', G'' \leadsto^* f(s'''_1,\ldots,s''',\ldots,s'''_n) \approx t''', G'''$$

don't take place at positions above $1{\cdot}i{\cdot}p$. According to Lemma 6.7 $\phi_1(\Pi)$ is outside-in. Hence, by definition, $1{\cdot}i{\cdot}p\backslash 1 = i{\cdot}p \in \mathcal{P}os_\mathcal{F}(f(l_1,\ldots,l_n))$. Therefore $p \in \mathcal{P}os_\mathcal{F}(l_i)$ and thus $2{\cdot}p \in \mathcal{P}os_\mathcal{F}(s \approx l_i) = \mathcal{P}os_\mathcal{F}(e)$. □

In the following lemma, Π_1 and Π' refer to the NC-refutation and the LNC-step obtained from Π in the proof of Theorem 5.10.

LEMMA 6.11. *Suppose* $\Pi: G \rightsquigarrow^+ \top$ *is a normal* NC*-refutation and let* e *be an equation in* G *that satisfies* $\mathcal{P}_\Pi(e)$*. If* e' *is a descendant of* e *with respect to* Π' *then* $\mathcal{P}_{\Pi_1}(e')$.

PROOF. Let G' be the initial goal of Π_1 and let $e' \in G'$ be a Π'-descendant of e. We distinguish two cases.

(1) Suppose e is the selected (leftmost) equation in G. Consider the case analysis in the proof of Theorem 5.10.

 (a) In cases (1)(b) and (2)(c)(i) we have $\Pi_1 = \phi_{[d]}(\phi_2(\Pi))$. It is easy to see that $\mathcal{P}_{\phi_2(\Pi)}(e)$ holds. Write e' as $s \approx t$. The equation e must be of the form

 $$f(s_1, \dots, s, \dots, s_n) \approx f(t_1, \dots, t, \dots, t_n).$$

 Let s be the i-th argument of $f(s_1, \dots, s, \dots, s_n)$. The first part of $\mathcal{P}_{\Pi_1(e')}$ clearly holds. Suppose narrowing is applied to the left-hand side of a descendant of e' in Π_1. Let $1{\cdot}p$ be a critical position. By construction of $\phi_{[d]}$, $1{\cdot}i{\cdot}p$ is a critical position in $\phi_2(\Pi)$. Hence we obtain $2{\cdot}i{\cdot}p \in \mathcal{P}os_\mathcal{F}(e)$ from $\mathcal{P}_{\phi_2(\Pi)}(e)$. This implies $2{\cdot}p \in \mathcal{P}os_\mathcal{F}(e')$. We conclude that $\mathcal{P}_{\Pi_1}(e')$ holds.

 (b) In case (2)(a) we have $\Pi_1 = \phi_{[d]}(\phi_2(\phi_{[on]}(\phi_1(\Pi))))$. Let e'' be the (unique) descendant of e in $\phi_{[on]}(\phi_1(\Pi))$. It is not difficult to see that $\mathcal{P}_{\phi_1(\Pi)}(e)$ holds. From the construction of $\phi_{[on]}$ we learn that the trace of e'' in $\phi_{[on]}(\phi_1(\Pi))$ is essentially the same as the trace of e in $\phi_1(\Pi)$. Hence $\mathcal{P}_{\phi_{[on]}(\phi_1(\Pi))}(e'')$ is a consequence of $\mathcal{P}_{\phi_1(\Pi)}(e)$. The step from $\mathcal{P}_{\phi_{[on]}(\phi_1(\Pi))}(e'')$ to $\mathcal{P}_{\Pi_1}(e')$ is the same as above.

 (c) In case (2)(c)(ii) we have $\Pi_1 = \phi_{[d]}(\phi_2(\phi_{[im]}(\Pi)))$. Let γ be the substitution employed in $\phi_{[im]}$. Since $\phi_{[im]}(\Pi)$ uses the same rewrite rules at the same positions in the corresponding equations of the goals in Π, $\mathcal{P}_{\phi_{[im]}(\Pi)}(e\gamma)$ is an immediate consequence of $\mathcal{P}_\Pi(e)$. The desired $\mathcal{P}_{\Pi_1}(e')$ is obtained by repeating case (1)(a) in this proof.

 (d) In cases (1)(a), (1)(b), (2)(b), and (2)(c)(iii) we have nothing to show. (Either e has no Π'-descendants or the first part of $\mathcal{P}_\Pi(e)$ doesn't hold.)

(2) Suppose e is not the leftmost equation in G. Consider again the case analysis in the proof of Theorem 5.10. In most cases the unique Π'-descendant e' of e equals e and the trace of e in Π_1 differs at most a renaming from the trace of e in Π. Hence $\mathcal{P}_{\Pi_1}(e)$ follows from $\mathcal{P}_\Pi(e)$. In the case that Π_1 consists of a $\Rightarrow_{[v]}$-step, so in case (1)(a)(ii) of the proof of Theorem 5.10, e' is a Π-descendant of e and the trace of e' in Π_1 is contained (modulo variable renaming) in the trace of e in Π_1. Hence also in this case we have $\mathcal{P}_{\Pi_1}(e')$. In the remaining case—Π_1 consists of a $\Rightarrow_{[im]}$-step—the desired result is also easily obtained.

□

The following result is an easy consequence of the preceding Lemmata. Here Π^* denotes the LNC-refutation $G \Rightarrow^* \square$ constructed in the proof of Theorem 5.10.

LEMMA 6.12. *Suppose* $\Pi: G \rightsquigarrow^+ \top$ *is a normal* NC*-refutation and let* e *be an equation in* G *that satisfies* $\mathcal{P}_\Pi(e)$*. The variable elimination rule* $[v]$ *is applied to all selected solved descendants of* e *in* Π^*. □

THEOREM 6.13. *Let* $\mathcal{R}$ *be an orthogonal TRS. For every outside-in* NC*-refutation* $G \rightsquigarrow^*_\theta \top$ *with* $\theta\restriction_{\mathcal{V}ar(G)}$ *normalized there exists an eager* LNC*-refutation* $G \Rightarrow^*_{\theta'} \top$ *respecting* $\mathcal{S}_{left}$ *such that*

$\theta' \leqslant \theta$ $[\mathcal{V}ar(G)]$.

PROOF. Let Π be the given outside-in NC-refutation $G \rightsquigarrow_\theta^* \top$. From Theorem 5.10 we obtain an LNC-refutation $\Pi^*: G \Rightarrow_{\theta'}^* \top$ respecting $\mathcal{S}_{left}$ such that $\theta' \leqslant \theta$ $[\mathcal{V}ar(G)]$. From Lemmata 6.10 and 6.12 we learn that the variable elimination rule $[v]$ is applied to all selected solved descendants of parameter-passing equations in Π^*, i.e., Π^* is eager. □

The combination of Theorems 6.6 and 6.13 yields the final result of this paper.

THEOREM 6.14. *Let $\mathcal{R}$ be an orthogonal TRS. If $\mathcal{R} \vdash G\theta$ and $\theta\lceil_{\mathcal{V}ar(G)}$ is normalized then there exists an eager* LNC*-refutation $G \Rightarrow_{\theta'}^* \top$ respecting $\mathcal{S}_{left}$ such that $\theta' \leqslant \theta$ $[\mathcal{V}ar(G)]$.* □

7. Suggestions for Further Research

This paper leaves many questions unanswered. In the near future we would like to address the following problems.

- We have seen that LNC lacks strong completeness. This does not mean that *all* selection functions result in incompleteness. We already showed that LNC is complete (for confluent TRSs and normalized solutions) with respect to $\mathcal{S}_{left}$. Extending this to selection functions that never select descendants of a body equation before all descendants of the corresponding parameter passing equations have been selected shouldn't be too difficult.
- In Section 4 we have shown the strong completeness of LNC in the case of orthogonal TRSs, using the completeness of basic NC. In Section 6 we showed the completeness of eager LNC with respect to $\mathcal{S}_{left}$ for orthogonal TRSs, using the completeness of outside-in NC. A natural question is whether these two results can be combined, i.e., is eager LNC strongly complete for orthogonal TRSs. Consider the orthogonal TRS $\mathcal{R}$ of Example 6.5 and the goal $f(a) \approx b$. There are two different NC-refutations starting from this goal:

$$\Pi_1: \quad f(a) \approx b \quad \rightsquigarrow_1 \quad a \approx b \quad \rightsquigarrow_1 \quad b \approx b \quad \rightsquigarrow_\epsilon \quad \texttt{true}$$

and

$$\Pi_2: \quad f(a) \approx b \quad \rightsquigarrow_{1\cdot 1} \quad f(b) \approx b \quad \rightsquigarrow_1 \quad b \approx b \quad \rightsquigarrow_\epsilon \quad \texttt{true}.$$

Refutation Π_1 is not basic and refutation Π_2 is not outside-in. Hence basic outside-in NC is not complete for orthogonal TRSs. This suggests that it is not obvious whether or not eager LNC is strongly complete for orthogonal TRSs.
- The orthogonality assumption in our proof of the completeness of eager LNC is essential since we make use of Huet and Lévy's standardization theorem. We didn't succeed in finding a non-orthogonal TRS for which eager LNC is not complete. Hence it is an open problem whether our restricted variable elimination strategy is complete for arbitrary confluent TRSs with respect to normalized solutions. A more general question is of course whether the variable elimination rule can always be eagerly applied, i.e., is the restriction to solved descendants of parameter-passing equations essential? In a recent paper Socher-Ambrosius [22] reports that the eager variable elimination problem has a positive solution in case of lazy paramodulation for arbitrary equational theories. It remains to be seen whether his techniques can be lifted to the present setting.
- In Section 6 we addressed non-determinism between the variable elimination rule on the one hand and the outermost narrowing and imitation rules on the other hand. This is not the only non-determinism between the inference rules. For instance, there are conflicts among

the outermost narrowing, imitation, and decomposition rules. A question that arises here is whether it is possible to remove all non-determinism between the various inference rules. (This does not prohibit the generation of different solutions to a given goal, because the outermost rule is non-deterministic in itself due to the various rewrite rules that may be applied.) The very simple orthogonal constructor-based TRS $\{f(a) \to f(b)\}$ together with the goal $f(x) \approx f(b)$ show that the restrictions for ensuring the completeness of a truly deterministic subset of LNC have to be very strong. Observe that the solution $\{x \mapsto a\}$ can only be produced by outermost narrowing, whereas decomposition is needed for obtaining the unrelated solution $\{x \mapsto b\}$.

References

1. L. Bachmair, H. Ganzinger, C. Lynch, and W. Snyder, *Basic Paramodulation and Superposition*, Proceedings of the 11th Conference on Automated Deduction, Lecture Notes in Computer Science **607**, pp. 462–476, 1992.
2. N. Dershowitz and J.-P. Jouannaud, *Rewrite Systems*, in: Handbook of Theoretical Computer Science, Vol. B ed. J. van Leeuwen), North-Holland, pp. 243–320, 1990.
3. N. Dershowitz and Z. Manna, *Proving Termination with Multiset Orderings*, Communications of the ACM **22**(8), pp. 465–476, 1979.
4. M. Fay, *First-Order Unification in Equational Theories*, Proceedings of the 4th Conference on Automated Deduction, Austin, pp. 161–167, 1979.
5. J. Gallier and W. Snyder, *Complete Sets of Transformations for General E-Unification*, Theoretical Computer Science **67**, pp. 203–260, 1989.
6. M. Hanus, *Efficient Implementation of Narrowing and Rewriting*, Proceedings of the International Workshop on Processing Declarative Knowledge, Lecture Notes in Artificial Intelligence **567**, pp. 344–365, 1991.
7. M. Hanus, *The Integration of Functions into Logic Programming: From Theory to Practice*, Journal of Logic Programming **19** & **20**, pp. 583–628, 1994.
8. S. Hölldobler, *A Unification Algorithm for Confluent Theories*, Proceedings of the 14th International Colloquium on Automata, Languages and Programming, Karlsruhe, Lecture Notes in Computer Science **267**, pp. 31–41, 1987.
9. S. Hölldobler, *Foundations of Equational Logic Programming*, Lecture Notes in Artificial Intelligence **353**, 1989.
10. G. Huet and J.-J. Lévy, *Computations in Orthogonal Rewriting Systems, I and II*, in: Computational Logic, Essays in Honor of Alan Robinson (eds. J.-L. Lassez and G. Plotkin), The MIT Press, pp. 396–443, 1991.
11. J.-M. Hullot, *Canonical Forms and Unification*, Proceedings of the 5th Conference on Automated Deduction, Lecture Notes in Computer Science **87**, pp. 318–334, 1980.
12. J.W. Klop, *Term Rewriting Systems*, in: Handbook of Logic in Computer Science, Vol. II (eds. S. Abramsky, D. Gabbay and T. Maibaum), Oxford University Press, pp. 1–116, 1992.
13. J.W. Lloyd, *Foundations of Logic Programming*, Springer, 2nd edition, 1987.
14. A. Martelli, C. Moiso, and G.F. Rossi, *Lazy Unification Algorithms for Canonical Rewrite Systems*, in: Resolution of Equations in Algebraic Structures, Vol. II, Rewriting Techniques (eds. H. Aït-Kaci and M. Nivat), Academic Press, pp. 245–274, 1989.
15. A. Martelli and U. Montanari, *An Efficient Unification Algorithm*, ACM Transactions on Programming Languages and Systems 4(2), pp. 258–282, 1982.

16. A. Martelli, G.F. Rossi, and C. Moiso, *An Algorithm for Unification in Equational Theories*, Proceedings of the 1986 Symposium on Logic Programming, pp. 180–186, 1986.
17. A. Middeldorp and E. Hamoen, *Completeness Results for Basic Narrowing*, Applicable Algebra in Engineering, Communication and Computing **5**, pp. 213–253, 1994.
18. M. Moser, *Improving Transformation Systems for General E-Unification*, Proceedings of the 5th International Conference on Rewriting Techniques and Applications, Montreal, Lecture Notes in Computer Science **690**, pp. 92–105, 1993.
19. J.A. Robinson, *A Machine-Oriented Logic Based on the Resolution Principle*, Journal of the ACM **12**(1), pp. 23–41, 1965.
20. J.R. Slagle, *Automatic Theorem Proving in Theories with Simplifiers, Commutativity and Associativity*, Journal of the ACM **21**, pp. 622–642, 1974.
21. W. Snyder, *A Proof Theory for General Unification*, Birkhäuser, 1991.
22. R. Socher-Ambrosius, *A Refined Version of General E-Unification*, Proceedings of the 12th International Conference on Automated Deduction, Nancy, Lecture Notes in Artificial Intelligence **814**, pp. 665–677, 1994.
23. Y.H. You, *Enumerating Outer Narrowing Derivations for Constructor Based Term Rewriting Systems*, Journal of Symbolic Computation **7**, pp. 319–343, 1989.

Appendix

A.1. Proof of Lemma 4.5

The following lemma can be viewed as a kind of partial lifting. It is the key to prove Lemma 4.5.

LEMMA A.1. *Let* $\Pi: G \leadsto_\theta^* \top$ *be an* NC*-refutation. Let* W *be the set of variables in the employed rewrite rules and* V *a set of variables which includes both* $\mathcal{V}ar(G)$ *and* W*. For all substitutions* γ *with* $\gamma \leqslant \theta\ [V]$ *and* $(\mathcal{D}(\gamma) \cup \mathcal{I}(\gamma)) \cap W = \varnothing$ *there exits an* NC*-refutation* $\Pi': G\gamma \leadsto_{\theta'}^* \top$ *such that* $\gamma\theta' = \theta\ [V]$*. Moreover, we may assume that* Π *and* Π' *employ the same rewrite rules at the same positions in the corresponding equations of the goals.*

PROOF. The proof is by induction on the length n of Π. The case $n = 0$ is obvious. Suppose $n > 0$. Without loss of generality we assume $\mathcal{V}ar(G) \cap W = \varnothing$. For all $i \in \{1, \ldots, n\}$ let σ_i be the substitution in the i-th step in Π and define $W_i = W - (\mathcal{V}ar(l_1) \cup \ldots \cup \mathcal{V}ar(l_i))$. Without loss of generality we furthermore assume that $W_i \cap (\mathcal{D}(\sigma_i) \cup \mathcal{I}(\sigma_i)) = \varnothing$. These two assumptions simply state that the variables in the rewrite rules are sufficiently fresh. Let the first step of Π be

$$G = (G', e, G'') \quad \leadsto_{\sigma_1, p_1, l_1 \to r_1, e} \quad (G', e[r_1]_{p_1}, G'')\sigma_1 = G_1$$

and let $\Pi_1: G_1 \leadsto_{\theta_1}^* \top$ be the remainder of Π. We have $\sigma_1\theta_1 = \theta$. Since $\gamma \leqslant \theta\ [V]$ there exists a substitution δ such that $\gamma\delta = \theta\ [V]$. Since $\mathcal{V}ar(G) \cup W \subseteq V$ and $\mathcal{D}(\gamma) \cap W = \varnothing$ we have

$$(e\gamma)_{|p_1}\delta = e_{|p_1}(\gamma\delta) = e_{|p_1}\theta = (e_{|p_1}\sigma_1)\theta_1 = (l_1\sigma_1)\theta_1 = l_1\theta = l_1\gamma\delta = l_1\delta,$$

so $(e\gamma)_{|p_1}$ and l_1 are unifiable. Hence there exists an idempotent most general unifier σ_1' of these two terms. We have $\sigma_1'\delta' = \delta$ for some substitution δ'. Now $e_{|p_1}(\gamma\sigma_1') = ((e\gamma)_{|p_1})\sigma_1' = l_1\sigma_1' = l_1(\gamma\sigma_1')$, so $\gamma\sigma_1'$ is a unifier of $e_{|p_1}$ and l_1. (The last equality is due to $\mathcal{D}(\gamma) \cap W = \varnothing$.) Since σ_1 is a most general of $e_{|p_1}$ and l_1 there exists a substitution γ' such that $\sigma_1\gamma' = \gamma\sigma_1'$. Let

$V_1 = V - \mathcal{D}(\sigma_1) \cup \mathcal{I}(\sigma_1 \restriction_V)$ and $\gamma_1 = \gamma' \restriction_{V_1}$. Using Lemma 2.1(2) we obtain $\sigma_1\gamma_1 = \gamma\sigma_1'$ $[V]$. Hence there exists an NC-step

$$G\gamma \quad \rightsquigarrow_{p_1, \sigma_1', l_1 \to r_1, e\gamma} \quad G_1\gamma_1.$$

Using $\sigma_1\gamma_1 = \gamma\sigma_1'$ $[V]$ we obtain $\sigma_1(\gamma_1\delta') = \gamma\sigma_1'\delta' = \gamma\delta = \sigma_1\theta_1$ $[V]$. Lemma 2.1(1) yields $\gamma_1\delta' = \theta_1$ $[V_1]$, so $\gamma_1 \leqslant \theta_1$ $[V_1]$. Before we can apply the induction hypothesis to Π_1, we must verify that $\mathcal{V}ar(G_1) \cup W_1 \subseteq V_1$ and $(\mathcal{D}(\gamma_1) \cup \mathcal{I}(\gamma_1)) \cap W_1 = \varnothing$. First we show that $\mathcal{V}ar(G_1) \cup W_1 \subseteq V_1$. We have

$$\mathcal{V}ar(G_1) = (\mathcal{V}ar(G) - \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1 \restriction_{\mathcal{V}ar(G)}) \subseteq (\mathcal{V}ar(G) - \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1 \restriction_V) \subseteq V_1.$$

From the assumptions $W_1 \cap \mathcal{D}(\sigma_1) = \varnothing$ and $W \subseteq V$ we infer that $W_1 \subseteq V_1$. Hence we obtain $\mathcal{V}ar(G_1) \cup W_1 \subseteq V_1$. Next we prove that $(\mathcal{D}(\gamma_1) \cup \mathcal{I}(\gamma_1)) \cap W_1 = \varnothing$. Idempotency of σ_1' yields

$$\mathcal{D}(\sigma_1') \cup \mathcal{I}(\sigma_1') \subseteq \mathcal{V}ar((e\gamma)_{|p_1}) \cup \mathcal{V}ar(l_1) \subseteq (\mathcal{V}ar(e) - \mathcal{D}(\gamma)) \cup \mathcal{I}(\gamma) \cup \mathcal{V}ar(l_1).$$

From this we easily obtain $(\mathcal{D}(\sigma_1') \cup \mathcal{I}(\sigma_1')) \cap W_1 = \varnothing$. Suppose to the contrary that $(\mathcal{D}(\gamma_1) \cup \mathcal{I}(\gamma_1)) \cap W_1 \neq \varnothing$. We distinguish two cases: (1) $\mathcal{D}(\gamma_1) \cap W_1 \neq \varnothing$ and (2) $\mathcal{I}(\gamma_1) \cap W_1 \neq \varnothing$.

(1) In the former case there exists a variable x such that $x \in \mathcal{D}(\gamma_1)$ and $x \in W_1$. From $W_1 \subseteq V_1$ and $W_1 \cap \mathcal{I}(\sigma_1) = \varnothing$ we obtain $W_1 \subseteq V - \mathcal{D}(\sigma_1)$. Hence we see that $x\gamma_1 = x\sigma_1\gamma_1 = x\gamma\sigma_1' = x\sigma_1'$. (The last equality follows from the assumption $\mathcal{D}(\gamma) \cap W = \varnothing$.) Because $x \in \mathcal{D}(\gamma_1)$ we also have $x \in \mathcal{D}(\sigma_1')$. Hence $\mathcal{D}(\sigma_1') \cap W_1 \neq \varnothing$ which yields a contradiction.

(2) Suppose $\mathcal{I}(\gamma_1) \cap W_1 \neq \varnothing$. We have $\mathcal{I}(\gamma) \cap W_1 = \varnothing$ by assumption. Let x be a variable such that $x \in \mathcal{I}(\gamma_1) \cap W_1$. There exists a variable $y \in \mathcal{D}(\gamma_1)$ such that $x \in \mathcal{V}ar(y\gamma_1)$. We distinguish two cases: $y \in V - \mathcal{D}(\sigma_1)$ and $y \in \mathcal{I}(\sigma_1 \restriction_V)$. Suppose $y \in V - \mathcal{D}(\sigma_1)$. Since $y\gamma_1 = y\sigma_1\gamma_1 = y\gamma\sigma_1'$ we obtain $x \in \mathcal{V}ar(y\gamma\sigma_1')$. Because $\mathcal{D}(\gamma_1) \cap W_1 = \varnothing$ we have $y \notin W_1$. Because $(\mathcal{I}(\gamma) \cup \mathcal{I}(\sigma_1')) \cap W_1 = \varnothing$, we have $\mathcal{V}ar(y\gamma\sigma_1') \cap W_1 = \varnothing$. This contradicts $x \in W_1$. In the remaining case we have $y \in \mathcal{I}(\sigma_1 \restriction_V)$. So there exists a variable $z \in \mathcal{D}(\sigma_1) \cap V$ such that $y \in \mathcal{V}ar(z\sigma_1)$. Since $z\sigma_1\gamma_1 = z\gamma\sigma_1'$, we have $x \in \mathcal{V}ar(y\gamma_1) \subseteq \mathcal{V}ar(z\sigma_1\gamma) = \mathcal{V}ar(z\gamma\sigma_1')$. Because $\mathcal{D}(\sigma_1) \cap W_1 = \varnothing$, $z \notin W_1$. From this we obtain a contradiction with $x \in W_1$ as in the previous case.

We conclude that $(\mathcal{D}(\gamma_1) \cup \mathcal{I}(\gamma_1)) \cap W_1 = \varnothing$. Now we are in a position to apply the induction hypothesis to Π_1. This yields an NC-refutation

$$G_1\gamma_1 \quad \rightsquigarrow^*_{\theta_1'} \quad \top$$

such that $\gamma_1\theta_1' = \theta_1$ $[V_1]$. Concatenating this NC-refutation with the NC-step

$$G\gamma \quad \rightsquigarrow_{\sigma_1'} \quad G_1\gamma_1$$

yields the NC-refutation $G\gamma \rightsquigarrow_{\theta'} \top$. Here $\theta' = \sigma_1'\theta_1'$. It remains to show that $\gamma\theta' = \theta$ $[V]$. Lemma 2.1(2) yields $\sigma_1\gamma_1\theta_1' = \sigma_1\theta_1 = \theta$ $[V]$. Hence $\gamma\theta' = \gamma\sigma_1'\theta_1' = \sigma_1\gamma_1\theta_1' = \theta$ $[V]$. □

LEMMA 4.5. *Let $G \rightsquigarrow^*_\theta \top$ be an NC-refutation, V a finite set of variables, and γ a substitution such that $\mathcal{V}ar(G) \subseteq V$, $\gamma \leqslant \theta$ $[V]$, and the variables in $\mathcal{D}(\gamma) \cup \mathcal{I}(\gamma)$ are different from the variables in the employed rewrite rules. There exists an NC-refutation $G\gamma \rightsquigarrow^*_{\theta'} \top$ which employs the same rewrite rules at the same positions in the corresponding equations of the goals in $G \rightsquigarrow^*_\theta \top$ such that $\gamma\theta' = \theta$ $[V]$.*

PROOF. Let W be the set of variables in the employed rewrite rules in the given NC-refutation and define $V' = V \cup W$. There exists a substitution ρ such that $\gamma\rho = \theta\ [V]$. From $\mathcal{D}(\gamma) \cap W = \varnothing$ we obtain $\gamma\theta = \theta\ [W]$. Since $\mathcal{I}(\gamma) \cap W = \varnothing$, the substitution $\delta = \rho\restriction_{\mathcal{I}(\gamma)} \cup \theta\restriction_W$ is well-defined. It is easy to see that $\gamma\delta = \theta\ [V']$ and thus $\gamma \leqslant \theta\ [V']$. From Lemma A.1 we obtain an NC-refutation $G\gamma \leadsto^*_{\theta'} \top$ which employs the same rewrite rules at the same positions in the corresponding equations of the goals in $G \leadsto^*_\theta \top$ such that $\gamma\theta' = \theta\ [V']$. In particular $\gamma\theta' = \theta\ [V]$. □

A.2. Proof of Lemma 4.14

LEMMA 4.14. *For every* NC*-derivation*

$$\begin{array}{lll} G_1, e_1, G_2, e_2, G_3 & \leadsto_{p_1, \sigma_1, l_1 \to r_1, e_1} & (G_1, e_1[r_1]_{p_1}, G_2, e_2, G_3)\sigma_1 \\ & \leadsto_{p_2, \sigma_2, l_2 \to r_2, e_2\sigma_1} & (G_1, e_1[r_1]_{p_1}, G_2, e_2[r_2]_{p_2}, G_3)\sigma_1\sigma_2 \end{array}$$

with $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$ *there exists an* NC*-derivation*

$$\begin{array}{lll} G_1, e_1, G_2, e_2, G_3 & \leadsto_{p_2, \sigma_2', l_2 \to r_2, e_2} & (G_1, e_1, G_2, e_2[r_2]_{p_2}, G_3)\sigma_2' \\ & \leadsto_{p_1, \sigma_1', l_1 \to r_1, e_1\sigma_2'} & (G_1, e_1[r_1]_{p_1}, G_2, e_2[r_2]_{p_2}, G_3)\sigma_2'\sigma_1' \end{array}$$

such that $\sigma_1\sigma_2 = \sigma_2'\sigma_1'$.

PROOF. Since we may assume that the variables in l_2 are fresh, we have $\mathcal{D}(\sigma_1) \cap \mathcal{V}ar(l_2) = \varnothing$. Hence

$$(e_{2|p_2})\sigma_1\sigma_2 = (e_2\sigma_1)_{|p_2}\sigma_2 = l_2\sigma_2 = l_2\sigma_1\sigma_2.$$

So $e_{2|p_2}$ and l_2 are unifiable. Let σ_2' be an idempotent most general unifier of these two terms. There exists a substitution ρ such that $\sigma_2'\rho = \sigma_1\sigma_2$. We have $\mathcal{D}(\sigma_2') \subseteq \mathcal{V}ar(e_{2|p_2}) \cup \mathcal{V}ar(l_2)$. Because we may assume that $\mathcal{V}ar(l_1) \cap \mathcal{V}ar(e_2) = \varnothing$, we obtain $\mathcal{D}(\sigma_2') \cap \mathcal{V}ar(l_1) = \varnothing$. Hence

$$(e_1\sigma_2')_{|p_1}\rho = (e_{1|p_1})\sigma_2'\rho = (e_{1|p_1})\sigma_1\sigma_2 = l_1\sigma_1\sigma_2 = l_1\sigma_2'\rho = l_1\rho.$$

So the terms $(e_1\sigma_2')_{|p_1}$ and l_1 are unifiable. Let σ_1'' be an idempotent most general unifier. We have $\sigma_1'' \leqslant \rho$. It follows that $\sigma_2'\sigma_1'' \leqslant \sigma_1\sigma_2$. Using $\mathcal{D}(\sigma_2') \cap \mathcal{V}ar(l_1) = \varnothing$ we obtain

$$(e_{1|p_1})\sigma_2'\sigma_1'' = (e_1\sigma_2')_{|p_1}\sigma_1'' = l_1\sigma_1'' = l_1\sigma_2'\sigma_1'',$$

so $\sigma_2'\sigma_1''$ is a unifier of $e_{1|p_1}$ and l_1. Because σ_1 is a most general unifier of these two terms, we must have $\sigma_1 \leqslant \sigma_2'\sigma_1''$. Let γ be any substitution satisfying $\sigma_1\gamma = \sigma_2'\sigma_1''$. With help of $\mathcal{D}(\sigma_1) \cap \mathcal{V}ar(l_2) = \varnothing$ we obtain

$$(e_2\sigma_1)_{|p_2}\gamma = (e_{2|p_2})\sigma_1\gamma = (e_{2|p_2})\sigma_2'\sigma_1'' = l_2\sigma_2'\sigma_1'' = l_2\sigma_1\gamma = l_2\gamma.$$

(In the first equality we used the assumption $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$.) Hence we obtain $\sigma_2 \leqslant \gamma$ from the fact that σ_2 is a most general unifier of $(e_2\sigma_1)_{|p_2}$ and l_2. Therefore $\sigma_1\sigma_2 \leqslant \sigma_1\gamma_1 = \sigma_2'\sigma_1''$. Since we also have $\sigma_2'\sigma_1'' \leqslant \sigma_1\sigma_2$, there is a variable renaming δ such that $\sigma_2'\sigma_1''\delta = \sigma_1\sigma_2$. Now define $\sigma_1' = \sigma_1''\delta$. Since most general unifiers are closed under variable renaming, σ_1' is a most general unifier of $(e_2\sigma_2')_{|p_1}$ and l_1. This proves the lemma. □

A.3. Proof of Lemma 5.3

LEMMA 5.3. *For every normal* NC*-refutation*

$$\begin{array}{llll} \Pi\colon & e, G & \rightsquigarrow^*_{\theta_1} & e', G' \\ & & \rightsquigarrow_{\sigma_1, p_1, l_1 \to r_1} & e'[r_1]_{p_1}\sigma_1, G'\sigma_1 \\ & & \rightsquigarrow_{\sigma_2, p_2, l_2 \to r_2} & ((e'[r_1]_{p_1}\sigma_1)[r_2]_{p_2}, G'\sigma_1)\sigma_2 \\ & & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with $p_1 \perp p_2$ *there exists a normal* NC*-refutation*

$$\begin{array}{llll} \Pi'\colon & e, G & \rightsquigarrow^*_{\theta_1} & e', G' \\ & & \rightsquigarrow_{\sigma_2', p_2, l_2 \to r_2} & (e'[r_2]_{p_2}, G')\sigma_2' \\ & & \rightsquigarrow_{\sigma_1', p_1, l_1 \to r_1} & ((e'[r_2]_{p_2})[r_1]_{p_1}, G')\sigma_2'\sigma_1' \\ & & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with the same complexity such that $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma_2'\sigma_1'\theta_2$.

PROOF. First we show that $p_2 \in \mathcal{P}os_{\mathcal{F}}(e')$. Suppose to the contrary that $p_2 \notin \mathcal{P}os_{\mathcal{F}}(e')$. That means that $p_2 \geqslant q$ for some $q \in \mathcal{P}os_{\mathcal{V}}(e')$. Without loss of generality we assume that $q \geqslant 1$. Let $e'_{|q}$ be the variable x. The term $(e'\sigma_1)_{|p_2}$ is a subterm of $x\sigma_1$. Hence $(e'\sigma_1)_{|p_2}\sigma_2$ is a subterm of $x\sigma_1\sigma_2$. Because $p_1 \perp p_2$ we have

$$(e'\sigma_1)_{|p_2}\sigma_2 = ((e'[r_1]_{p_1})\sigma_1)_{|p_2}\sigma_2 = l_2\sigma_2.$$

So $x\sigma_1\sigma_2$ is not a normal form. Hence $x\sigma_1\sigma_2\theta_2$ is also not a normal form. There exists a reduction sequence from $e\theta_1$ to e' consisting of non-root reduction steps. Hence $x \in \mathcal{V}ar(e'_{|1}) \subseteq \mathcal{V}ar((e\theta_1)_{|1})$. From the normality of Π we infer that $\theta_1\sigma_1\sigma_2\theta_2\restriction_{\mathcal{V}ar(e_{|1})}$ is normalized. This yields a contradiction with Lemma 2.3. Therefore $p_2 \in \mathcal{P}os_{\mathcal{F}}(e')$.

Since the variables in l_2 are fresh, we have $\mathcal{D}(\sigma_1) \cap \mathcal{V}ar(l_2) = \varnothing$. Hence

$$e'_{|p_2}\sigma_1\sigma_2 = (e'\sigma_1)_{|p_2}\sigma_2 = l_2\sigma_2 = l_2\sigma_1\sigma_2.$$

So $\sigma_1\sigma_2$ is a unifier of $e'_{|p_2}$ and l_2. Hence there exists an idempotent most general unifier σ_2' of $e'_{|p_2}$ and l_2 such that $\sigma_2' \leqslant \sigma_1\sigma_2$. Let ρ be a substitution satisfying $\sigma_2'\rho = \sigma_1\sigma_2$. Since σ_2' is idempotent, $\mathcal{D}(\sigma_2') \cap \mathcal{V}ar(l_1) = \varnothing$. Hence

$$(e'\sigma_2')_{|p_1}\rho = e'_{|p_1}\sigma_2'\rho = (e'_{|p_1}\sigma_1)\sigma_2 = (l_1\sigma_1)\sigma_2 = l_1\sigma_2'\rho = l_1\rho,$$

i.e., ρ is a unifier of $(e'\sigma_2')_{|p_1}$ and l_1. Let σ_1'' be an idempotent most general unifier of these two terms. We have $\sigma_1'' \leqslant \rho$ and thus $\sigma_2'\sigma_1'' \leqslant \sigma_2'\rho = \sigma_1\sigma_2$. Using $\mathcal{D}(\sigma_2') \cap \mathcal{V}ar(l_1) = \varnothing$ we obtain

$$e'_{|p_1}\sigma_2'\sigma_1'' = (e'\sigma_2')_{|p_1}\sigma_1'' = l_1\sigma_1'' = l_1\sigma_2'\sigma_1''.$$

Since σ_1 is a most general of $e'_{|p_1}$ and l_1, we have $\sigma_1 \leqslant \sigma_2'\sigma_1''$, so there exists a substitution γ such that $\sigma_1\gamma = \sigma_2'\sigma_1''$. Using $\mathcal{D}(\sigma_1) \cap \mathcal{V}ar(l_2) = \varnothing$ we obtain

$$(e'\sigma_1)_{|p_2}\gamma = e'_{|p_2}\sigma_1\gamma = (e'_{|p_2}\sigma_2')\sigma_1'' = (l_2\sigma_2')\sigma_1'' = l_2\sigma_1\gamma = l_2\gamma.$$

Because σ_2 is a most general unifier of $(e'\sigma_1)_{|p_2}$ and l_2, we must have $\sigma_2 \leqslant \gamma$ and hence $\sigma_1\sigma_2 \leqslant \sigma_1\gamma = \sigma_2'\sigma_1''$. So $\sigma_1\sigma_2$ and $\sigma_2'\sigma_1''$ are variants. Hence there exists a variable renaming δ such that

$\sigma_2'\sigma_1''\delta = \sigma_1\sigma_2$. Now define $\sigma_1' = \sigma_1''\delta$. Since σ_1'' is a most general unifier of $e'\sigma'_{2|p_1}$ and l_1, and most general unifiers are closed under variable renaming, we infer that also σ_1' is a most general unifier of these two terms. Hence we obtain

$$\begin{array}{lll} e', G' & \rightsquigarrow_{\sigma_2', p_2, l_2 \to r_2} & (e'[r_2]_{p_2}, G')\sigma_2' \\ & \rightsquigarrow_{\sigma_1', p_1, l_1 \to r_1} & ((e'[r_2]_{p_2})[r_1]_{p_1}, G')\sigma_2'\sigma_1' \end{array}$$

Clearly $((e'[r_1]_{p_1}\sigma_1)[r_2]_{p_2}, G'\sigma_1)\sigma_2 = ((e'[r_2]_{p_2})[r_1]_{p_1}, G')\sigma_2'\sigma_1'$. Replacing the two steps that produce $\sigma_1\sigma_2$ in Π by the above two steps yields the desired refutation Π'. We clearly have $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma_2'\sigma_1'\theta_2$. Because the number of narrowing steps at non-root positions is the same in Π and Π', it follows that they have the same complexity. It is also easy to see that Π' inherits normality from Π. □

A.4. Proof of Lemma 5.8

LEMMA 5.8. *For every* NC*-refutation*

$$\begin{array}{lll} \Pi\colon\ G_1, e_1, G_2, e_2, G_3 & \rightsquigarrow^*_{\theta_1} & G_1', e_1', G_2', e_2', G_3' \\ & \rightsquigarrow_{p_1, \sigma_1, l_1 \to r_1, e_1'} & (G_1', e_1'[r_1]_{p_1}, G_2')\sigma_1, e_2'\sigma_1, G_3'\sigma_1 \\ & \rightsquigarrow_{p_2, \sigma_2, l_2 \to r_2, e_2'\sigma_1} & ((G_1', e_1'[r_1]_{p_1}, G_2')\sigma_1, (e_2'\sigma_1)[r_2]_{p_2}, G_3'\sigma_1)\sigma_2 \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

that produces a normalized substitution there exists a NC*-refutation*

$$\begin{array}{lll} \Pi'\colon\ G_1, e_1, G_2, e_2, G_3 & \rightsquigarrow^*_{\theta_1} & G_1', e_1', G_2', e_2', G_3' \\ & \rightsquigarrow_{p_2, \sigma_2', l_2 \to r_2, e_2'} & (G_1', e_1', G_2', e_2'[r_2]_{p_2}, G_3')\sigma_2' \\ & \rightsquigarrow_{p_1, \sigma_1', l_1 \to r_1, e_1'\sigma_2'} & (G_1', e_1'[r_1]_{p_1}, G_2', e_2'[r_2]_{p_2}, G_3')\sigma_2'\sigma_1' \\ & \rightsquigarrow^*_{\theta_2} & \top \end{array}$$

with the same complexity such that $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma_2'\sigma_1'\theta_2$.

PROOF. First we show that $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2')$. Suppose to the contrary that $p_2 \notin \mathcal{P}os_{\mathcal{F}}(e_2')$. That means that $p_2 \geqslant q$ for some $q \in \mathcal{P}os_{\mathcal{V}}(e_2')$. Without loss of generality we assume that $q \geqslant 1$. Let $(e_2')_{|q}$ be the variable x. The term $(e_2'\sigma_1)_{|p_2}$ is a subterm of $x\sigma_1$. Hence $(e_2'\sigma_1)_{|p_2}\sigma_2$ is a subterm of $x\sigma_1\sigma_2$. Since $(e_2'\sigma_1)_{|p_2}\sigma_2 = l_2\sigma_2$, we conclude that $x\sigma_1\sigma_2$ is not a normal form. Hence $x\sigma_1\sigma_2\theta_2$ is also not a normal form. There exists a reduction sequence from $e_2\theta_1$ to e_2' consisting of non-root reduction steps. Hence $x \in \mathcal{V}ar((e_2')_{|1}) \subseteq \mathcal{V}ar((e_2\theta_1)_{|1})$. Because Π produces a normalized solution, $\theta_1\sigma_1\sigma_2\theta_2\lceil_{\mathcal{V}ar((e_2)_{|1})}$ is normalized. This yields a contradiction with Lemma 2.3. Hence we have $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$. This implies that

$$((G_1', e_1'[r_1]_{p_1}, G_2')\sigma_1, (e_2'\sigma_1)[r_2]_{p_2}, G_3'\sigma_1)\sigma_2 = (G_1', e_1'[r_1]_{p_1}, G_2', e_2'[r_2]_{p_2}, G_3')\sigma_1\sigma_2.$$

Now we apply Lemma 4.14, resulting in a refutation Π' of the desired shape with $\theta_1\sigma_1\sigma_2\theta_2 = \theta_1\sigma_2'\sigma_1'\theta_2$. It is easy to see that the transformation of Lemma 4.14 preserves complexity. □