Lazy Narrowing: Strong Completeness and Eager Variable Elimination

Aart Middeldorp Satoshi Okui Tetsuo Ida

Institute of Information Sciences and Electronics
and
Center for Tsukuba Advanced Research Alliance
University of Tsukuba, Tsukuba 305, Japan

ami@is.tsukuba.ac.jp

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 ([21, 4, 11]) is a general E-unification procedure for equational theories that are presented by confluent term rewriting systems. 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

An extended abstract of this paper appeared in the Proceedings of the 20th Colloquium on Trees in Algebra and Programming, Aarhus, Lecture Notes in Computer Science 915, pp. 394–408,1995.

This work is partially supported by the Grant-in-Aid for Scientific Research (C) 06680300 and the Grant-in-Aid for Encouragement of Young Scientists 06780229 of the Ministry of Education, Science and Culture of Japan.

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.

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, 22, 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, 22]. 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 [22] 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 $f(t_1, \ldots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ 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 $\mathcal{V}\mathrm{ar}(t)$.

A *position* is a sequence of natural numbers identifying a subterm in a term. The set $\mathcal{P}\mathrm{os}(t)$ of positions in a term t is inductively defined as follows: $\mathcal{P}\mathrm{os}(t) = \{\varepsilon\}$ if t is a variable and $\mathcal{P}\mathrm{os}(t) = \{\varepsilon\} \cup \{i \cdot p \mid 1 \leqslant i \leqslant n \text{ and } p \in \mathcal{P}\mathrm{os}(t_i)\}$ if $t = f(t_1, \ldots, t_n)$. Here ε, the *root* position, denotes the empty sequence. If $p \in \mathcal{P}\mathrm{os}(t)$ then $t_{|p}$ denotes the *subterm* of t at position p 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 $\mathcal{P}\mathrm{os}(t)$ is partitioned into $\mathcal{P}\mathrm{os}_{\mathcal{V}}(t)$ and $\mathcal{P}\mathrm{os}_{\mathcal{F}}(t)$ as follows: $\mathcal{P}\mathrm{os}_{\mathcal{V}}(t) = \{p \in \mathcal{P}\mathrm{os}(t) \mid t_{|p} \in \mathcal{V}\}$ and $\mathcal{P}\mathrm{os}_{\mathcal{F}}(t) = \mathcal{P}\mathrm{os}(t) \backslash \mathcal{P}\mathrm{os}_{\mathcal{V}}(t)$. Elements of $\mathcal{P}\mathrm{os}_{\mathcal{V}}(t)$ 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 $\mathcal{P}\mathrm{os}(t)$.

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 $\bigcup_{x \in \mathcal{D}(\theta)} \mathcal{V}\mathrm{ar}(x\theta)$ 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 relation $\leqslant$ is called *subsumption*. The *restriction* $\theta\restriction_V$ of a substitution θ to a set $V (\subseteq \mathcal{V})$ of variables is defined as follows:

$\theta\restriction_V(x) = \theta(x)$ if $x \in \mathcal{V}$ and $\theta\restriction_V(x) = x$ if $x \notin \mathcal{V}$. A *variable substitution* maps variables to variables. A *variable renaming* is a bijective variable substitution. It is well-known that the combination of $\theta_1 \leqslant \theta_2$ and $\theta_2 \leqslant \theta_1$ is equivalent to the existence of a variable renaming δ such that $\theta_1\delta = \theta_2$. We write $\theta_1 = \theta_2$ $[V]$ if $\theta_1\restriction_V = \theta_2\restriction_V$. 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 $\theta\theta = \theta$. It is easy to show that 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 [20] and Martelli and Montanari [15] produce an idempotent most general unifier θ that satisfies $\mathcal{D}(\theta) \cup \mathcal{I}(\theta) \subseteq \mathcal{V}\mathrm{ar}(s) \cup \mathcal{V}\mathrm{ar}(t)$.

A *rewrite rule* is a directed equation $l \to r$ that satisfies $l \notin \mathcal{V}$ and $\mathcal{V}\mathrm{ar}(r) \subseteq \mathcal{V}\mathrm{ar}(l)$. 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 $p \in \mathcal{P}\mathrm{os}(s)$ 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 $l\theta$. Occasionally we write $s \to_{\theta,\,p,\,l\to r} t$ or $s \to_{p,\,l\to r} t$. The transitive and 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 $\to^+_{\mathcal{R}}$. The equational theory induced by $\mathcal{R}$ is denoted by $\leftrightarrow^*_{\mathcal{R}}$. 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 1 *Let θ, θ_1, θ_2 be substitutions and V, V' sets of variables such that $(V' \backslash \mathcal{D}(\theta)) \cup \mathcal{I}(\theta\restriction_{V'}) \subseteq V$. If $\theta_1 \leqslant \theta_2$ $[V]$ then $\theta\theta_1 \leqslant \theta\theta_2$ $[V']$.* □

Lemma 2 *Let θ_1, θ_2 be substitutions and V, V' sets of variables such that $V' \subseteq (V \backslash \mathcal{D}(\theta_1)) \cup \mathcal{I}(\theta_1\restriction_V)$. If $\theta_1\theta_2\restriction_V$ is normalized then $\theta_2\restriction_{V'}$ is normalized.* □

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 $t_1 \to^* t_2$ and $t_1 \to^* t_3$ there exists a term t_4 such that $t_2 \to^* t_4$ and $t_3 \to^* t_4$. Such a term t_4 is called a *common reduct* of t_2 and t_3. Confluence is equivalent to the property that t_1 and t_2 have a common reduct whenever $t_1 \leftrightarrow^* t_2$. If $l \to r$ is a rewrite rule and θ a variable renaming then the rewrite rule $l\theta \to r\theta$ is called a *variant* of $l \to r$. A rewrite rule $l \to r$ 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 $l_1 \to r_1$ and $l_2 \to r_2$ be variants of rewrite rules of a TRS $\mathcal{R}$ such that they have no variables in common. Suppose $l_{1|p}$, for some $p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(l_1)$, 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 $l_1 \to r_1$ and $l_2 \to r_2$ 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 $x \approx x \to \mathtt{true}$ is denoted by $\mathcal{R}_+$. Let $e = s \approx t$ be an equation and θ a substitution. We say that θ is an ($\mathcal{R}$-)*solution* of e and we write $\mathcal{R} \vdash e\theta$ if $s\theta \leftrightarrow^*_{\mathcal{R}} t\theta$. So θ is a solution of e if $e\theta$ belongs to the equational theory generated by $\mathcal{R}$. If $\mathcal{R}$ is confluent, $\mathcal{R} \vdash e\theta$ is equivalent to the existence of a rewrite sequence $e\theta \to^*_{\mathcal{R}_+} \mathtt{true}$. We find it convenient to call a solution θ of e *normalized* if the substitution $\theta\lceil_{\mathcal{V}\mathrm{ar}(e)}$ is normalized. Narrowing is formulated as the following inference rule:

$$\frac{e}{e[r]_p\theta}$$

if there exist a fresh[1]variant $l \to r$ of a rewrite rule in $\mathcal{R}_+$, a position $p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e)$, and a most general unifier θ of $e_{|p}$ and l.

In the above situation we write $e \leadsto_{\theta,p,l\to r} e[r]_p\theta$. 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

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

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 3 *Let $\mathcal{R}$ be a confluent TRS and e an equation. For every normalized[2] solution θ of e there exists an* NC-*refutation $e \leadsto^*_{\theta'}$* `true` *such that $\theta' \leqslant \theta\ [\mathcal{V}\mathrm{ar}(e)]$.* $\square$

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

[1]This means that $l \to r$ has no variables in common with the preceding part of the computation.

[2]Often completeness is stated with respect to normaliz*able* solutions: if $\mathcal{R} \vdash e\theta$ and $\theta\lceil_{\mathcal{V}\mathrm{ar}(e)}$ is normalizable then there exists an nc-refutation $e \leadsto^*_{\theta'}$ `true` such that $\theta' \leqslant_{\mathcal{R}} \theta\ [\mathcal{V}\mathrm{ar}(e)]$. 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.

containing only equations $\mathtt{true}$. So for confluent TRSs $\mathcal{R}$, $\mathcal{R} \vdash G\theta$ if and only if $G\theta \rightarrow^*_{\mathcal{R}_+} \top$. The calculus NC is extended to goals as follows:

$$\frac{G', e, G''}{(G', e[r]_p, G'')\theta}$$

if there exist a fresh variant $l \to r$ of a rewrite rule in $\mathcal{R}_+$, a position $p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e)$, 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. For an NC-derivation $\Pi \colon G \rightsquigarrow^*_\theta G'$, $\Pi\theta$ denotes the corresponding rewrite sequence $G\theta \to^* G'$.

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 equation 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 $\mathtt{true}$. An example of a selection function is $\mathcal{S}_{\mathrm{left}}$ which always returns the leftmost equation different from $\mathtt{true}$. We say that an NC-derivation Π *respects* a selection function $\mathcal{S}$ if the selected equation in every step $G_1 \rightsquigarrow G_2$ of Π coincides with $\mathcal{S}(G_1)$. Now strong completeness of NC is formulated as follows.

Theorem 4 *Let $\mathcal{R}$ be a confluent TRS, $\mathcal{S}$ a selection function, and G a goal. For every normalized solution θ of G there exists an* NC*-refutation $G \rightsquigarrow^*_{\theta'} \top$ respecting $\mathcal{S}$ such that $\theta' \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$.* $\square$

In this paper we make frequent use of the following *lifting lemma* for NC.

Lemma 5 *Let G be a goal and θ a normalized substitution. For every rewrite sequence $G\theta \to^* G'$ there exists an* NC*-derivation $\Pi \colon G \rightsquigarrow^*_{\theta'} G''$ such that $\theta' \leqslant \theta$ $[V]$ and $\Pi\theta'$ subsumes $G\theta \to^* G'$. Here V is any finite set of variables such that $\mathcal{V}\mathrm{ar}(G) \cup \mathcal{D}(\theta) \subseteq V$.* $\square$

The statement "$\Pi\theta'$ subsumes $G\theta \to^* G'$" entails that the constructed NC-derivation Π and the given rewrite sequence $G\theta \to^* G'$ employ the same rewrite rules at the same positions in the corresponding goals.

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

Definition 7 An NC-derivation

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

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

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

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

$$B_i(e_i) \backslash \{q \in B_i(e_i) \mid q \geqslant p_i\} \cup \{p_i{\cdot}q \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e) \mid q \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(r_i)\}.$$

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

So in a basic derivation narrowing is never applied to a subterm introduced by a previous narrowing substitution. 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 8 *Let $\mathcal{R}$ be a confluent TRS and G a goal. For every normalized solution θ of G there exists a basic* NC*-refutation $G \rightsquigarrow^*_{\theta'} \top$ such that $\theta' \leqslant \theta\ [\mathcal{V}\mathrm{ar}(G)]$, 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 [22]. 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 9 Let $\mathcal{R}$ be a TRS. The *lazy narrowing calculus*, LNC for short, consists of the following five inference rules:
[o] *outermost narrowing*

$$\frac{G', f(s_1, \ldots, s_n) \simeq t, G''}{G', s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, G''}$$

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

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

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

[d] *decomposition*

$$\frac{G', f(s_1,\ldots,s_n) \approx f(t_1,\ldots,t_n), G''}{G', s_1 \approx t_1, \ldots, s_n \approx t_n, G''},$$

[v] *variable elimination*

$$\frac{G', s \simeq x, G''}{(G', G'')\theta}$$

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

[t] *removal of trivial equations*

$$\frac{G', x \approx x, G''}{G', G''}.$$

In the rules [o], [i], and [v], $s \simeq t$ 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 [d], [v], and [t] constitute the syntactic unification algorithm of Martelli and Montanari [15]. Because syntactic unification is performed by [d], [v], and [t], the rewrite rule $x \approx x \to \mathtt{true}$ is no longer used in LNC. As a consequence, we assume that the symbol `true` doesn't occur in LNC-goals.

Contrary to usual narrowing, the outermost narrowing rule [o] 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_1 are the upper and lower goal in the inference rule [α] ($\alpha \in \{o, i, d, v, t\}$), we write $G \Rightarrow_{[\alpha]} G_1$. This is called an LNC-step. The applied rewrite rule or substitution may be supplied as subscript, that is, we will write things like $G \Rightarrow_{[o], l \to r} G_1$ and $G \Rightarrow_{[i], \theta} G_1$. LNC-derivations are defined as in the case of NC. To distinguish LNC-derivations from NC-derivations, we use the symbol Ψ (and its derivatives) for the former. An LNC-refutation is an LNC-derivation ending in the *empty goal* $\Box$.

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 10 Consider the TRS

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

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 and some case analysis. The (normalized) empty substitution ϵ is a solution of G because

$$\begin{array}{lllll} f(b) \approx a & \to_{\mathcal{R}} & g(h(b), b) \approx a & \to_{\mathcal{R}} & g(h(b), h(b)) \approx a \\ & \to_{\mathcal{R}} & a \approx a & \to_{\mathcal{R}_+} & \mathtt{true}. \end{array}$$

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

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

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-derivation.) 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 $\mathcal{S}_{\text{left}}$:

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

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 simpler than proving completeness.

4 Restoring Strong Completeness

Observe that the TRS $\mathcal{R}$ of Counterexample 10 satisfies none of the sufficient conditions for the completeness of basic narrowing stated in Theorem 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 11 Let $G \rightsquigarrow_{\theta, p, l\to r} G_1$ 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_1. In this case we say that $e[r]_p\theta$ is the *descendant* of e in G_1.

Otherwise, e is simply instantiated to the equation $e\theta$ in G_1 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 $G \rightsquigarrow^* \top$ every equation $e \in G$ has exactly one descendant $\mathtt{true}$ in $\top$. We now introduce six transformation steps on NC-refutations. The first one states that non-empty NC-refutations are closed under renaming.

Lemma 12 *Let δ be a variable renaming. For every* NC*-refutation $\Pi\colon G \rightsquigarrow^+_\theta \top$ there exists an* NC*-refutation $\phi_\delta(\Pi)\colon G\delta \rightsquigarrow^+_{\delta^{-1}\theta} \top$.*

Proof. Let $G \rightsquigarrow_{\tau_1, e, p, l \to r} G_1$ be the first step of Π and let $\Pi_1\colon G_1 \rightsquigarrow^*_{\tau_2} \top$ be the remainder of Π. We have $\tau_1\tau_2 = \theta$. We show the existence of an NC-step $G\delta \rightsquigarrow_{\delta^{-1}\tau_1, e\delta, p, l\delta \to r\delta} G_1$. First we show that $\delta^{-1}\tau_1$ is a most general unifier of $e\delta_{|p}$ and $l\delta$. We have $e\delta_{|p}\delta^{-1}\tau_1 = e_{|p}\tau_1 = l\tau_1 = l\delta\delta^{-1}\tau_1$, so $\delta^{-1}\tau_1$ is a unifier of $e\delta_{|p}$ and $l\delta$. Let σ be an arbitrary unifier of $e\delta_{|p}$ and $l\delta$. Because $\delta\sigma$ is a unifier of $e_{|p}$ and l, and τ_1 is a most general unifier of these two terms, it follows that $\tau_1 \leqslant \delta\sigma$ and thus $\delta^{-1}\tau_1 \leqslant \sigma$. We conclude that $\delta^{-1}\tau_1$ is a most general unifier of $e\delta_{|p}$ and $l\delta$. Write G as G', e, G''. We obtain $G\delta \rightsquigarrow (G'\delta, e\delta[r\delta]_p, G''\delta)\delta^{-1}\tau_1 = (G', e[r]_p, G'')\tau_1 = G_1$. Concatenating this NC-step with the NC-refutation Π_1 yields the desired NC-refutation $\phi_\delta(\Pi)\colon G\delta \rightsquigarrow^+_{\delta^{-1}\theta} \top$. $\square$

Observe that Lemma 12 doesn't hold for the empty NC-refutation $\Pi\colon \top \rightsquigarrow^*_\epsilon \top$ because $\delta^{-1}\epsilon$ is only equal to ϵ if $\delta = \epsilon$. In the following five lemmata Π denotes an NC-refutation $G \rightsquigarrow^+_\theta \top$ with $G = G', e, G''$ such that $e = s \approx t$ is selected in the first step of Π and V denotes a finite set of variables that includes all variables in the initial goal G of Π.

Lemma 13 *Suppose narrowing is applied to a descendant of e in Π at position 1. If $l \to r$ is the applied rewrite rule in the first such step then there exists an* NC*-refutation $\phi_{[o]}(\Pi)\colon G', s \approx l, r \approx t, G'' \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 = \theta\ [V]$.*

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

$$\begin{array}{lll} G & \rightsquigarrow^*_{\tau_1} & G_1', f(u_1, \ldots, u_n) \approx t', G_1'' \\ & \rightsquigarrow_{\tau_2, 1, l \to r} & (G_1', r \approx t', G_1'')\tau_2 \\ & \rightsquigarrow^*_{\tau_3} & \top \end{array}$$

with $\tau_1\tau_2\tau_3 = \theta$. Let x be a fresh variable (so $x \notin V$) and define the substitution υ_2 as the (disjoint) union of τ_2 and $\{x \mapsto l\tau_2\}$. Because υ_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 $\phi_{[o]}(\Pi)$:

$$\begin{array}{lll} G', s \approx l, r \approx t, G'' & \rightsquigarrow^*_{\tau_1} & G_1', f(u_1, \ldots, u_n) \approx l, r \approx t', G_1'' \\ & \rightsquigarrow_{\upsilon_2} & (G_1', \mathtt{true}, r \approx t', G_1'')\upsilon_2 \\ & = & (G_1', \mathtt{true}, r \approx t', G_1'')\tau_2 \\ & \rightsquigarrow^*_{\tau_3} & \top. \end{array}$$

Let $\theta_1 = \tau_1\upsilon_2\tau_3$. We have $\theta_1 = \theta \cup \{x \mapsto l\tau_2\tau_3\}$ and because $x \notin V$ we obtain $\theta_1 = \theta\ [V]$. $\square$

The tedious (and boring) proof of the next transformation lemma is omitted. It is similar to the proof of Lemma 5, see e.g. [17].

Lemma 14 *Let* $s = f(s_1, \ldots, s_n)$ *and* $t \in \mathcal{V}$. *If* $\mathrm{root}(t\theta) = f$ *then there exists an* NC*-refutation* $\phi_{[i]}(\Pi)\colon G\sigma_1 \rightsquigarrow^*_{\theta_1} \top$ *such that* Π *subsumes* $\phi_{[i]}(\Pi)$, $\Pi\theta = \phi_{[i]}(\Pi)\theta_1$ *and* $\sigma_1\theta_1 = \theta\ [V]$. *Here* $\sigma_1 = \{t \mapsto f(x_1, \ldots, x_n)\}$ *with* $x_1, \ldots, x_n \notin V$. $\square$

Lemma 15 *Let* $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$, *and suppose that narrowing is never applied to a descendant of* e *in* Π *at position* 1 *or* 2. *There exists an* NC*-refutation* $\phi_{[d]}(\Pi)\colon G', s_1 \approx t_1, \ldots, s_n \approx t_n, G'' \rightsquigarrow^*_{\theta_1} \top$ *such that* $\theta_1 \leqslant \theta\ [V]$.

Proof. The given refutation Π must be of the form

$$G \quad \rightsquigarrow^*_{\tau_1} \quad G'_1, s' \approx t', G''_1 \quad \rightsquigarrow_{\tau_2,\varepsilon} \quad (G'_1, \mathtt{true}, G''_1)\tau_2 \quad \rightsquigarrow^*_{\tau_3} \quad \top$$

with $s' = f(s'_1, \ldots, s'_n)$, $t' = f(t'_1, \ldots, t'_n)$, and $\tau_1\tau_2\tau_3 = \theta$. The first part of Π can be transformed into Π_1:

$$G', s_1 \approx t_1, \ldots, s_n \approx t_n, G'' \quad \rightsquigarrow^*_{\tau_1} \quad G'_1, s'_1 \approx t'_1, \ldots, s'_n \approx t'_n, G''_1.$$

Consider the step from $G'_1, s' \approx t', G''_1$ to $(G'_1, \mathtt{true}, G''_1)\tau_2$. Let $x \approx x \to \mathtt{true}$ be the employed rewrite rule, so τ_2 is a most general unifier of $x \approx x$ and $s' \approx t'$. There clearly exists a rewrite sequence

$$(G'_1, s'_1 \approx t'_1, \ldots, s'_n \approx t'_n, G''_1)\tau_2 \quad \to^*_\varepsilon \quad (G'_1, \top, G''_1)\tau_2.$$

Lifting[3] results in an NC-derivation Π_2:

$$G'_1, s'_1 \approx t'_1, \ldots, s'_n \approx t'_n, G''_1 \quad \rightsquigarrow^*_{\upsilon_2,\varepsilon} \quad (G'_1, \top, G''_1)\upsilon_2$$

such that $\upsilon_2 \leqslant \tau_2\ [V \cup \mathcal{I}(\tau_1)]$. We distinguish two cases.

(1) Suppose $G'_1, G''_1 = \square$. In this case $\tau_3 = \epsilon$. We simply define $\phi_{[d]}(\Pi) = \Pi_1; \Pi_2$. Let $\theta_1 = \tau_1\upsilon_2$. From $\upsilon_2 \leqslant \tau_2\ [V \cup \mathcal{I}(\tau_1)]$ we infer, using Lemma 1, that $\theta_1 \leqslant \tau_1\tau_2 = \theta\ [V]$.

(2) The case $G'_1, G''_1 \neq \square$ is more involved. First observe that υ_2 is a unifier of s' and t'. Using the fact that τ_2 is a most general unifier of $s' \approx t'$ and $x \approx x$, it is not difficult to show that $\tau_2 \leqslant \upsilon_2\ [\mathcal{V}\backslash\{x\}]$. Since $x \notin V \cup \mathcal{I}(\tau_1)$ we have in particular $\tau_2 \leqslant \upsilon_2\ [V \cup \mathcal{I}(\tau_1)]$. It follows that there exists a variable renaming δ such that $\upsilon_2 = \tau_2\delta\ [V \cup \mathcal{I}(\tau_1)]$. Clearly $\mathcal{V}\mathrm{ar}(G'_1, G''_1) \subseteq V \cup \mathcal{I}(\tau_1)$. The last part of Π can be trivially transformed (by changing the number of occurrences of $\mathtt{true}$ in each goal) into Π_3: $(G'_1, \top, G''_1)\tau_2 \rightsquigarrow^+_{\tau_3} \top$. An application of Lemma 12 results in the NC-refutation $\phi_\delta(\Pi_3)$: $(G'_1, \top, G''_1)\upsilon_2 \rightsquigarrow^+_{\delta^{-1}\tau_3} \top$. Define $\phi_{[d]}(\Pi) = \Pi_1; \Pi_2; \phi_\delta(\Pi_3)$. Let $\theta_1 = \tau_1\upsilon_2\delta^{-1}\tau_3$. We have $\theta_1 = \tau_1\tau_2\tau_3 = \theta\ [V]$.

$\square$

[3] The lifting lemma for NC—Lemma 5—requires the normalization of the substitution τ_2, which is not necessarily the case here. The reason for requiring normalization is to avoid rewrite sequences in which a term introduced by τ_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 $\theta_1 = \theta\ [V]$ in Lemma 15. Consider for example the NC-refutation $\Pi\colon a \approx a \rightsquigarrow_{\theta, x\approx x \to \mathtt{true}} \mathtt{true}$ 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 $\phi_{[d]}(\Pi)\colon \square$ produces the empty substitution $\theta_1 = \epsilon$. Clearly $\theta_1 \neq \theta\ [V]$ if V contains y or z.

Lemma 16 *Let $t \in \mathcal{V}$, $t \notin \mathcal{V}\mathrm{ar}(s)$, and suppose that the first step of Π takes place at the root position. There exists an* NC*-refutation $\phi_{[v]}(\Pi)\colon (G', G'')\sigma_1 \rightsquigarrow^*_{\theta_1} \top$ with $\sigma_1 = \{t \mapsto s\}$ such that $\sigma_1\theta_1 \leqslant \theta\ [V]$.*

Proof. The given refutation Π is of the form

$$G \quad \rightsquigarrow_{\tau_1, \varepsilon, x\approx x \to \mathtt{true}} \quad (G', \mathtt{true}, G'')\tau_1 \quad \rightsquigarrow^*_{\tau_2} \quad \top$$

with $\tau_1\tau_2 = \theta$. Let υ_1 be the (disjoint) union of $\{x \mapsto s\}$ and σ_1. Clearly υ_1 is a unifier of the equations $s \approx t$ and $x \approx x$. It is not too difficult to show that υ_1 is a most general unifier of these two equations. Since also τ_1 is a most general unifier of $s \approx t$ and $x \approx x$, there exists a variable renaming δ such that $\tau_1\delta = \upsilon_1$. If $G', G'' = \square$ then we let $\phi_{[v]}(\Pi)$ be the empty NC-refutation and thus $\theta_1 = \epsilon$. In this case we have $\sigma_1\theta_1 = \upsilon_1 \leqslant \tau_1 = \theta\ [V]$. If $G', G'' \neq \square$, we reason as follows. From the second part of Π we extract the NC-refutation $\Pi_1\colon (G', G'')\tau_1 \rightsquigarrow^+_{\tau_2} \top$ by simply dropping a single occurrence of $\mathtt{true}$ in every goal of Π. Let $\theta_1 = \delta^{-1}\tau_2$. From Lemma 12 we obtain an NC-refutation $\phi_\delta(\Pi_1)\colon (G', G'')\upsilon_1 \rightsquigarrow^+_{\theta_1} \top$. Because $x \notin V$ we have $\sigma_1\theta_1 = \upsilon_1\theta_1 = \tau_1\delta\delta^{-1}\tau_2 = \tau_1\tau_2 = \theta\ [V]$. Moreover, $(G_1, G_2)\upsilon_1 = (G_1, G_2)\sigma_1$. Hence we can define $\phi_{[v]}(\Pi) = \phi_\delta(\Pi_1)$. $\square$

Lemma 17 *Let $t \in \mathcal{V}$, $s = t$, and suppose that the first step of Π takes place at the root position. There exists an* NC*-refutation $\phi_{[t]}(\Pi)\colon G', G'' \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 \leqslant \theta\ [V]$.*

Proof. The proof is obtained from the previous one by letting σ_1 be the empty substitution ϵ. $\square$

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 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 $\succ_{\mathrm{mul}}$ on the set of finite multisets over A as follows: $M \succ_{\mathrm{mul}} N$ if there exist multisets X and Y such that $\varnothing \neq X \subseteq M$, $N = (M - X) \uplus Y$, and for every $y \in Y$ there exists an $x \in X$ such that $x \succ y$. Here $\uplus$ denotes multiset sum and $-$ denotes multiset difference. Dershowitz and Manna [3] showed that multiset extension preserves well-foundedness.

Definition 18 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 narrowing steps that do not use the rewrite rule $x \approx x \to \mathtt{true}$), M is the multiset $|\mathcal{MV}\mathrm{ar}(G)\theta|$, and s is the number of occurrences of symbols different from $\approx$ and $\mathtt{true}$ in G (which is the same as the total number of symbols in G minus the number of equations in G). Here $\mathcal{MV}\mathrm{ar}(G)$ denotes the multiset of variable occurrences in G, and for any multiset $M = \{t_1, \ldots, t_n\}$ of terms, $M\theta$ and $|M|$ denote the multiset $\{t_1\theta, \ldots, t_n\theta\}$ and $\{|t_1|, \ldots, |t_n|\}$, respectively. We define a (strict) partial order $\gg$ on NC-refutations as follows: $\Pi_1 \gg \Pi_2$ if

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

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

Lemma 19 *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 20 *Let Π be an* NC*-refutation and $\alpha \in \{o, i, d, v, t\}$. We have $\Pi \gg \phi_{[\alpha]}(\Pi)$ whenever $\phi_{[\alpha]}(\Pi)$ is defined.*

Proof. According to the proof of Lemma 13 the number of narrowing steps at non-root positions in the NC-refutation $\phi_{[o]}(\Pi)$ is one less than in Π. Hence $\Pi \gg \phi_{[o]}(\Pi)$.

Next we consider $\phi_{[i]}$. By construction, Π and $\phi_{[i]}(\Pi)$ have the same number of narrowing steps at non-root positions. Let M_1 and M_2 be the second components of the triples $|\Pi|$ and $|\phi_{[i]}(\Pi)|$. We claim that $M_1 >_{\mathrm{mul}} M_2$. Let X be the multiset of all occurrences of the variable t in G, $Y = \mathcal{MV}\mathrm{ar}(G) - X$, and $X' = \mathcal{MV}\mathrm{ar}(G\sigma_1) - Y$. Observe that X' is the multiset of all occurrences of the variables $x_1, \ldots, x_n$ in $G\sigma_1$. We have $M_1 = |X\theta| \uplus |Y\theta|$ and $M_2 = |X'\theta_1| \uplus |Y\theta_1|$. We have $|Y\theta| = |Y\theta_1|$ because $y\theta_1 = y\sigma_1\theta_1 = y\theta$ for all $y \in Y$. Let $t\theta = f(t_1, \ldots, t_n)$. We have $f(x_1\theta_1, \ldots, x_n\theta_1) = t\sigma_1\theta_1 = t\theta = f(t_1, \ldots, t_n)$ and thus $|x\theta_1| < |t\theta|$ for all $x \in X'$. Therefore $|X\theta| >_{\mathrm{mul}} |X'\theta_1|$ and hence $M_1 >_{\mathrm{mul}} M_2$. We conclude that $\Pi \gg \phi_{[i]}(\Pi)$.

According to the proof of Lemma 15 the number of narrowing steps at non-root positions in $\phi_{[d]}(\Pi)$ is the same as 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 $|\Pi|$. Since 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), we conclude that $\Pi \gg \phi_{[d]}(\Pi)$.

From the proof of Lemma 16 we learn that Π and $\phi_{[v]}(\Pi)$ have the same number of narrowing steps at non-root positions. Let M_1 and M_2 be the second components of the triples $|\Pi|$ and $|\phi_{[v]}(\Pi)|$. We claim that $M_1 >_{\mathrm{mul}} M_2$. We partition $\mathcal{MV}\mathrm{ar}(G', e, G'')$ into the following three multisets: X

the multiset of all variables in $\mathcal{MV}\text{ar}(G', G'')$ that belong to $\mathcal{D}(\sigma_1)$, $Y = \mathcal{MV}\text{ar}(G', G'') - X$, and $Z = \mathcal{MV}\text{ar}(e)$. Let X' be the multiset of all variable occurrences in the initial goal $(G', G'')\sigma_1$ of $\phi_{[v]}(\Pi)$ that are introduced by σ_1, so $X' \uplus Y = \mathcal{MV}\text{ar}((G', G'')\sigma_1)$. We have $M_1 = |X\theta| \uplus |Y\theta| \uplus |Z\theta|$ and $M_2 = |X'\theta_1| \uplus |Y\theta_1|$. Since $y\theta_1 = y\sigma_1\theta_1 \leqslant y\theta$ for all $y \in Y$, we have $|Y\theta| >_{\text{mul}}^{=} |Y\theta_1|$. Using the inequality $\sigma_1\theta_1 \leqslant \theta\ [X]$, it is not difficult to see that $|X\theta| >_{\text{mul}}^{=} |X'\theta_1|$. Since $Z \neq \varnothing$ it follows that $M_1 >_{\text{mul}} M_2$ and therefore $\Pi \gg \phi_{[v]}(\Pi)$.

According to the proof of Lemma 17 the number of narrowing steps at non-root positions in $\phi_{[t]}(\Pi)$ is the same in Π. Because the substitution produced in $\phi_{[t]}(\Pi)$ subsumes the substitution produced in Π for the initial variables, the second component of $|\phi_{[t]}(\Pi)|$ doesn't exceed the second component of $|\Pi|$. Finally, the initial goal of $\phi_{[t]}(\Pi)$ has less symbols different from $\approx$ and `true` than the initial goal of Π (viz. two occurrences of the same variable). $\square$

Lemma 21 explains why we don't need the symmetric versions of Lemmata 13, 14, and 16.

Lemma 21 *For every* NC*-refutation* $\Pi\colon G', s \approx t, G'' \rightsquigarrow_\theta^* \top$ *there exists an* NC*-refutation* $\phi_{\text{swap}}(\Pi)\colon G', t \approx s, G'' \rightsquigarrow_\theta^* \top$ *with the same complexity.*

Proof. Simply swap the two sides in every descendant of $s \approx t$ in Π. This doesn't affect the complexity. $\square$

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

Example 22 Consider the TRS $\mathcal{R} = \{f(g(y)) \to y\}$ and the NC-refutation $\Pi\colon g(f(x)) \approx x \rightsquigarrow_{\{x \mapsto g(y)\}} g(y) \approx g(y) \rightsquigarrow_\epsilon \texttt{true}$. In Π the variable x is bound to $g(y)$, so the complexity of Π is $(1, \{2, 2\}, 4)$. Transformation steps $\phi_{[o]}$, $\phi_{[d]}$, $\phi_{[v]}$, and $\phi_{[t]}$ are not applicable to Π. Hence we try $\phi_{[i]}$. This yields the NC-refutation $\Pi_1 = \phi_{[i]}(\Pi)\colon g(f(g(x_1))) \approx g(x_1) \rightsquigarrow_{\{x_1 \mapsto y\}} g(y) \approx g(y) \rightsquigarrow_\epsilon \texttt{true}$ which has complexity $(1, \{1, 1\}, 6)$. Next we apply $\phi_{[d]}$. This gives the NC-refutation $\Pi_2 = \phi_{[d]}(\Pi_1)\colon f(g(x_1)) \approx x_1 \rightsquigarrow_{\{x_1 \mapsto y\}} y \approx y \rightsquigarrow_\epsilon \texttt{true}$ with complexity $(1, \{1, 1\}, 4)$. Observe that the initial goal of Π is transformed into the initial goal of Π_2 by a single $\Rightarrow_{[i]}$-step. In Π_2 narrowing is applied to the initial equation at position 1. This calls for the transformation step $\phi_{[o]}$, so $\Pi_3 = \phi_{[o]}(\Pi_2)\colon f(g(x_1)) \approx f(g(y)), y \approx x_1 \rightsquigarrow_{\{x_1 \mapsto y\}} \texttt{true}, y \approx y \rightsquigarrow_\epsilon \top$. NC-refutation Π_3 has complexity $(0, \{1, 1, 1, 1\}, 8)$. If we apply $\phi_{[d]}$ to Π_3, we obtain the NC-refutation $\Pi_4 = \phi_{[d]}(\Pi_3)\colon g(x_1) \approx g(y), y \approx x_1 \rightsquigarrow_{\{x_1 \mapsto y\}} \texttt{true}, y \approx y \rightsquigarrow_\epsilon \top$ with complexity $(0, \{1, 1, 1, 1\}, 6)$. The initial goals of Π_2 and Π_4 are connected by an $\Rightarrow_{[o]}$-step. In the first step of Π_4 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 $\Pi_5 = \phi_{[d]}(\Pi_4)\colon x_1 \approx y, y \approx x_1 \rightsquigarrow_{\{x_1 \mapsto y\}} \texttt{true}, y \approx y \rightsquigarrow_\epsilon \top$ with complexity $(0, \{1, 1, 1, 1\}, 4)$. Next we use $\phi_{[v]}$, yielding the NC-refutation $\Pi_6 = \phi_{[v]}(\Pi_5)\colon y \approx y \rightsquigarrow_\epsilon \texttt{true}$ with complexity

$(0, \{1,1\}, 2)$. The initial goals of Π_4, Π_5, and Π_6 are connected by the LNC-derivation $g(x_1) \approx g(y), y \approx x_1 \Rightarrow_{[d]} x_1 \approx y, y \approx x_1 \Rightarrow_{[v],\{x_1 \mapsto y\}} y \approx y$. An application of $\phi_{[t]}$ results in the empty NC-refutation $\Pi_7 = \phi_{[t]}(\Pi_6)\colon \Box$ which has complexity $(0, \varnothing, 0)$. Clearly $y \approx y \Rightarrow_{[t]} \Box$. Concatenating the various LNC-sequences yields an LNC-refutation $g(f(x)) \approx x \Rightarrow^*_\theta \Box$ whose substitution θ satisfies $x\theta = g(y)$. In Figure 1 this transformation process is summarized.

$$
\begin{array}{lcccccc}
\Pi & : & g(f(x)) \approx x & \rightsquigarrow & g(y) \approx g(y) & \rightsquigarrow & \mathtt{true} \\
 & & \Downarrow_{[i],\{x \mapsto g(x_1)\}} & & & & \\
\Pi_2 & : & f(g(x_1)) \approx x_1 & \rightsquigarrow & y \approx y & \rightsquigarrow & \mathtt{true} \\
 & & \Downarrow_{[o]} & & & & \\
\Pi_4 & : & g(x_1) \approx g(y), y \approx x_1 & \rightsquigarrow & \mathtt{true}, y \approx y & \rightsquigarrow & \top \\
 & & \Downarrow_{[d]} & & & & \\
\Pi_5 & : & x_1 \approx y, y \approx x_1 & \rightsquigarrow & \mathtt{true}, y \approx y & \rightsquigarrow & \top \\
 & & \Downarrow_{[v],\{x_1 \mapsto y\}} & & & & \\
\Pi_6 & : & y \approx y & \rightsquigarrow & \mathtt{true} & & \\
 & & \Downarrow_{[t]} & & & & \\
\Pi_7 & : & \Box & & & &
\end{array}
$$

Figure 1: The transformation in Example 22.

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

Example 23 Consider the TRS

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

and the NC-refutation Π_{fail}:

$$
\begin{array}{lllllll}
f(a) \approx g(a) & \rightsquigarrow & f(a) \approx g(b) & \rightsquigarrow & a \approx g(b) & \rightsquigarrow & b \approx g(b) \\
 & \rightsquigarrow & g(b) \approx g(b) & \rightsquigarrow & \mathtt{true}. & &
\end{array}
$$

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 $f(x) \to x$, we transform Π_{fail} using $\phi_{[o]}$ and $\phi_{[d]}$. This yields the NC-refutation $\phi_{[d]}(\phi_{[o]}(\Pi_{\mathrm{fail}}))$:

$$
\begin{array}{lllllll}
a \approx x, x \approx g(a) & \rightsquigarrow & a \approx x, x \approx g(b) & \rightsquigarrow & \mathtt{true}, a \approx g(b) & & \\
 & \rightsquigarrow & \mathtt{true}, b \approx g(b) & \rightsquigarrow & \mathtt{true}, g(b) \approx g(b) & \rightsquigarrow & \top.
\end{array}
$$

Observe that the initial goals of Π_{fail} and $\phi_{[d]}(\phi_{[o]}(\Pi_{\mathrm{fail}}))$ are connected by an $\Rightarrow_{[o]}$-step. Since in the refutation $\phi_{[d]}(\phi_{[o]}(\Pi_{\mathrm{fail}}))$ 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 $\phi_{[o]}$ 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 $\phi_{[d]}(\phi_{[o]}(\Pi_{\text{fail}}))$ 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 $\phi_{[d]}(\phi_{[o]}(\Pi_{\text{fail}}))$ 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 7) is preserved. This is one of the two key observations for the connection between strong completeness of LNC and completeness of basic NC.

Lemma 24 *Let* Π *be a basic* NC*-refutation and* $\alpha \in \{o, i, d, v, t\}$. *The* NC*-refutation* $\phi_{[\alpha]}(\Pi)$ *is basic whenever it is defined.*

Proof (sketch). It is not difficult to see that narrowing is never applied to subterms introduced by previous narrowing substitutions in $\phi_{[o]}(\Pi)$ and $\phi_{[d]}(\Pi)$ whenever this is true for Π. Hence $\phi_{[o]}(\Pi)$ and $\phi_{[d]}(\Pi)$ are basic provided that Π is basic. Since $\phi_{[i]}(\Pi)\theta_1 = \Pi\theta$, $\phi_{[i]}(\Pi)$ inherits basicness from Π. Because $\phi_{[v]}(\Pi)$ and Π_1—Π minus its first step—are the same modulo renaming, the transformation $\phi_{[v]}$ trivially preserves basicness. This reasoning also applies to $\phi_{[t]}$. $\Box$

We are now ready for the main lemma, which can be diagrammatically depicted as follows:

$$\begin{array}{llllll} \forall & \text{basic} & \Pi & : & G & \rightsquigarrow^{+}_{\theta} & \top \\ \exists & & \Psi_1 & : & \Downarrow_{\sigma_1} & & \\ \exists & \text{basic} & \Pi_1 & : & G_1 & \rightsquigarrow^{*}_{\theta_1} & \top \end{array} \quad \text{such that} \quad \left\{ \begin{array}{l} \sigma_1\theta_1 \leqslant \theta \; [V] \\ \Pi \gg \Pi_1 \end{array} \right.$$

Lemma 25 *For every non-empty basic* NC*-refutation* $\Pi\colon G \rightsquigarrow^{+}_{\theta} \top$ *there exist an* LNC*-step* $\Psi_1\colon G \Rightarrow_{\sigma_1} G_1$ *and a basic* NC*-refutation* $\Pi_1\colon G_1 \rightsquigarrow^{*}_{\theta_1} \top$ *such that* $\sigma_1\theta_1 \leqslant \theta\ [V]$, $\Pi \gg \Pi_1$, *and the equation selected in the first step of* Π *is selected in* Ψ_1.

Proof. We distinguish the following cases, depending on what happens to the selected equation $e = s \approx t$ in the first step of Π. Let $G = G', e, G''$.

(1) Suppose narrowing is never applied to a descendant of $s \approx t$ at position 1 or 2. We distinguish four further cases.

  (a) Suppose $s, t \notin \mathcal{V}$. We may write $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$. Let $G_1 = G', s_1 \approx t_1, \ldots, s_n \approx t_n, G''$. We have Ψ_1: $G \Rightarrow_{[d]} G_1$. Lemma 15 yields an NC-refutation $\phi_{[d]}(\Pi)\colon G_1 \rightsquigarrow^{*}_{\theta_1} \top$ such that $\theta_1 \leqslant \theta\ [V]$. Take $\sigma_1 = \epsilon$.

  (b) Suppose $t \in \mathcal{V}$ and $s = t$. In this case the first step of Π_1 must take place at the root of e. Let $G_1 = G', G''$. We have $\Psi_1\colon G \Rightarrow_{[t]} G_1$. Lemma 17 yields an NC-refutation $\phi_{[t]}(\Pi)\colon G_1 \rightsquigarrow^{*}_{\theta_1} \top$ such that $\theta_1 \leqslant \theta\ [V]$. Take $\sigma_1 = \epsilon$.

  (c) Suppose $t \in \mathcal{V}$ and $s \neq t$. We distinguish two further cases, depending on what happens to e in the first step of Π.

(i) Suppose narrowing is applied to e at the root position. Let $\sigma_1 = \{t \mapsto s\}$ and $G_1 = (G', G'')\sigma_1$. We have $\Psi_1 : G \Rightarrow_{[v], \sigma_1} G_1$. Lemma 16 yields an NC-refutation $\phi_{[v]}(\Pi) : G_1 \rightsquigarrow^*_{\theta_1} \top$ such that $\sigma_1\theta_1 \leqslant \theta\ [V]$.

(ii) Suppose narrowing is not applied to e at the root position. This implies that $s \notin \mathcal{V}$. Hence we may write $s = f(s_1, \ldots, s_n)$. Let $\sigma_1 = \{t \mapsto f(x_1, \ldots, x_n)\}$, $G_1 = (G', s_1 \approx x_1, \ldots, s_n \approx x_n, G'')\sigma_1$, and $G_2 = G\sigma_1$. Here $x_1, \ldots, x_n$ are fresh variables. We have $\Psi_1 : G \Rightarrow_{[i], \sigma_1} G_1$. From Lemma 14 we obtain an NC-refutation $\Pi_2 = \phi_{[i]}(\Pi) : G_2 \rightsquigarrow^*_{\theta_2} \top$ such that $\sigma_1\theta_2 = \theta\ [V]$. Let $V_2 = V \cup \{x_1, \ldots, x_n\}$. Clearly $\mathcal{V}\mathrm{ar}(G_2) \subseteq V_2$. An application of Lemma 15 to Π_2 results in an NC-refutation $\Pi_1 = \phi_{[d]}(\Pi_2) : G_1 \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 \leqslant \theta_2\ [V_2]$. Using the inclusion $(V \setminus \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1\restriction_V) \subseteq V_2$ we obtain $\sigma_1\theta_1 \leqslant \sigma_1\theta_2 = \theta\ [V]$ from Lemma 1.

(d) In the remaining case we have $t \notin \mathcal{V}$ and $s \in \mathcal{V}$. This case reduces to case (1)(c) by an appeal to Lemma 21.

(2) Suppose narrowing is applied to a descendant of e at position 1. Let $l = f(l_1, \ldots, l_n) \to r$ be the used rewrite rule the first time this happens. Because Π is basic, s cannot be a 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 $G_1 = G', s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, G''$ and $G_2 = G', s \approx l, r \approx t, G''$. We have $\Psi_1 : G \Rightarrow_{[o]} G_1$. From Lemma 13 we obtain an NC-refutation $\Pi_2 = \phi_{[o]}(\Pi) : G_2 \rightsquigarrow^*_{\theta_2} \top$ such that $\theta_2 = \theta\ [V]$. Let $V_2 = V \cup \mathcal{V}\mathrm{ar}(l)$. Clearly $\mathcal{V}\mathrm{ar}(G_2) \subseteq V_2$. An application of Lemma 15 to Π_2 results in an NC-refutation $\Pi_1 = \phi_{[d]}(\Pi_2) : G_1 \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 \leqslant \theta_2\ [V_2]$. Using $V \subseteq V_2$ we obtain $\theta_1 \leqslant \theta\ [V]$. Take $\sigma_1 = \epsilon$.

(3) Suppose narrowing is applied to a descendant of e at position 2. This case reduces to the previous one by an appeal to Lemma 21.

The above case analysis is summarized in Table 1. In all cases we obtain Π_1 from Π by applying one or two transformation steps $\phi_{[o]}$, $\phi_{[i]}$, $\phi_{[d]}$, $\phi_{[v]}$, $\phi_{[t]}$ together with an additional application of ϕ_{swap} in case (1)(d) and (3). According to Lemma 24 Π_1 is basic. According to Lemmata 20 and 21 Π_1 has smaller complexity than Π. $\square$

The other key observation for the connection between strong completeness of LNC and completeness of basic NC is the fact 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 26 *Let G_1 be a goal containing distinct equations e_1 and e_2. For every* NC*-derivation*

$$G_1 \quad \rightsquigarrow_{\tau_1, e_1, p_1, l_1 \to r_1} \quad G_2 \quad \rightsquigarrow_{\tau_2, e_2\tau_1, p_2, l_2 \to r_2} \quad G_3$$

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

$$G_1 \quad \rightsquigarrow_{v_2, e_2, p_2, l_2 \to r_2} \quad H_2 \quad \rightsquigarrow_{v_1, e_1 v_2, p_1, l_1 \to r_1} \quad H_3$$

such that $G_3 = H_3$ and $\tau_1\tau_2 = v_2 v_1$. $\square$

case	LNC-step	transformation(s)
(1)(a)	$\Rightarrow_{[d]}$	$\phi_{[d]}$
(1)(b)	$\Rightarrow_{[t]}$	$\phi_{[t]}$
(1)(c)(i)	$\Rightarrow_{[v]}$	$\phi_{[v]}$
(1)(c)(ii)	$\Rightarrow_{[i]}$	$\phi_{[d]} \circ \phi_{[i]}$
(1)(d)	$\Rightarrow_{[v]}$ or $\Rightarrow_{[i]}$	$\phi_{[v]} \circ \phi_{\text{swap}}$ or $\phi_{[d]} \circ \phi_{[i]} \circ \phi_{\text{swap}}$
(2)	$\Rightarrow_{[o]}$	$\phi_{[d]} \circ \phi_{[o]}$
(3)	$\Rightarrow_{[o]}$	$\phi_{[d]} \circ \phi_{[o]} \circ \phi_{\text{swap}}$

Table 1: Case analysis in the proof of Lemma 25.

Observe that the requirement $p_2 \in \mathcal{P}\text{os}_{\mathcal{F}}(e_2)$ in Lemma 26 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 27 *Let $\mathcal{S}$ be an arbitrary selection function. For every basic* NC*-refutation $\Pi\colon G \rightsquigarrow_\theta^* \top$ there exists a basic* NC*-refutation $\phi_{\mathcal{S}}(\Pi)\colon G \rightsquigarrow_\theta^* \top$ 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 $\phi_{\mathcal{S}}(\Pi)\colon G \rightsquigarrow_\theta^* \top$ that respects $\mathcal{S}$ by a finite number of applications of Lemma 26. Since the transformation in Lemma 26 preserves the number of narrowing steps at non-root positions as well as the computed substitution, it follows that the complexities of Π and $\phi_{\mathcal{S}}(\Pi)$ are the same. $\Box$

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

Theorem 28 *Let $\mathcal{R}$ be an arbitrary TRS and $\Pi\colon G \rightsquigarrow_\theta^* \top$ a basic* NC*-refutation. For every selection function $\mathcal{S}$ there exists an* LNC*-refutation $\Psi\colon G \Rightarrow_\sigma^* \Box$ respecting $\mathcal{S}$ such that $\sigma \leqslant \theta\ [\mathcal{V}\text{ar}(G)]$.*

Proof. We use well-founded induction on the complexity of the given basic NC-refutation Π, which is possible because of Lemma 19. In order to make the induction work we prove $\sigma \leqslant \theta\ [V]$ for a finite set of variables V that includes $\mathcal{V}\text{ar}(G)$ instead of $\sigma \leqslant \theta\ [\mathcal{V}\text{ar}(G)]$. The base case is trivial: G must be the empty goal. For the induction step we proceed as follows. First we use Lemma 27 to transform Π into a basic NC-refutation $\phi_{\mathcal{S}}(\Pi)\colon G \rightsquigarrow_\theta^+ \top$ respecting $\mathcal{S}$ with equal complexity. According to Lemma 25 there exist an LNC-step $\Psi_1\colon G \Rightarrow_{\sigma_1} G_1$ respecting $\mathcal{S}$ and a basic NC-refutation $\Pi_1\colon G_1 \rightsquigarrow_{\theta_1}^* \top$ such that $\sigma_1\theta_1 \leqslant \theta\ [V]$ and $\phi_{\mathcal{S}}(\Pi) \gg \Pi_1$. Let $V_1 = (V \setminus \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1\restriction_V) \cup \mathcal{V}\text{ar}(G_1)$. Clearly V_1 is a finite set of variables that includes $\mathcal{V}\text{ar}(G_1)$. The induction hypothesis yields an LNC-refutation $\Psi'\colon G_1 \Rightarrow_{\sigma'}^* \Box$ respecting $\mathcal{S}$ such that $\sigma' \leqslant \theta_1\ [V_1]$. Now define $\sigma = \sigma_1\sigma'$. From $\sigma_1\theta_1 \leqslant \theta\ [V]$, $\sigma' \leqslant \theta_1\ [V_1]$, and $(V \setminus \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1\restriction_V) \subseteq V_1$, we infer—using Lemma 1—that $\sigma \leqslant \theta\ [V]$ and thus also $\sigma \leqslant \theta\ [\mathcal{V}\text{ar}(G)]$. Concatenating the LNC-step Ψ_1 and the LNC-refutation Ψ' yields the desired LNC-refutation Ψ. $\Box$

A related result for lazy paramodulation calculi is given by Moser [19]. 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 28 with Theorem 8 yields the following result.

Corollary 29 *Let $\mathcal{R}$ be a confluent TRS, $\mathcal{S}$ a selection function, and G a goal. For every normalized solution θ of G there exists an* LNC*-refutation $G \Rightarrow_\sigma^* \square$ respecting $\mathcal{S}$ such that $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$, 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.*
$\square$

The converse of Theorem 28 does not hold, as witnessed by the confluent TRS

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

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 $f(a) \approx c \Rightarrow^* \square$ 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 $\mathcal{S}_{\mathrm{left}}$. 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 $\mathcal{S}_{\mathrm{left}}$ that
(1) includes all NC-refutations respecting $\mathcal{S}_{\mathrm{left}}$ that produce normalized solutions, and
(2) which is closed under the transformations $\phi_{[o]}$, $\phi_{[i]}$, $\phi_{[d]}$, $\phi_{[v]}$, $\phi_{[t]}$, and ϕ_{swap},

then completeness with respect to $\mathcal{S}_{\mathrm{left}}$ follows along the lines of the proof of Theorem 28. We didn't succeed in defining such a class, the main problem being the fact that an application of $\phi_{[o]}$ or $\phi_{[d]}$ to an NC-refutation that respects $\mathcal{S}_{\mathrm{left}}$ may result in an NC-refutation that doesn't respect $\mathcal{S}_{\mathrm{left}}$. We found however a class of NC-refutations respecting $\mathcal{S}_{\mathrm{left}}$ that satisfies the first property and which is closed under $\phi_{[o]} \circ \phi_1$, $\phi_{[i]}$, $\phi_{[d]} \circ \phi_2$, $\phi_{[v]}$, $\phi_{[t]}$, and ϕ_{swap}. Here ϕ_1 and ϕ_2 are transformations that preprocess NC-refutations in such a way that a subsequent application of $\phi_{[o]}$ and $\phi_{[d]}$ results in an

NC-refutation respecting $\mathcal{S}_{\text{left}}$. The following definition introduces our class of NC-refutations.

Definition 30 An NC-refutation $\Pi: G \rightsquigarrow_{\theta}^{*} \top$ is called *normal* if it respects $\mathcal{S}_{\text{left}}$ and 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 $\theta_2 \restriction_{\mathcal{V}\text{ar}(s\theta_1)}$ $(\theta_2 \restriction_{\mathcal{V}\text{ar}(t\theta_1)})$ is normalized. Here θ_1 and θ_2 are defined by writing Π as $G = G', s \approx t, G'' \rightsquigarrow_{\theta_1}^{*} \top, (s \approx t, G'')\theta_1 \rightsquigarrow_{\theta_2}^{*} \top$.

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 31 *Let G_1 be the goal e, G'. For every normal* NC*-refutation*

$$G \rightsquigarrow_{\theta_1}^{*} G_1 \rightsquigarrow_{\tau_1, p_1, l_1 \to r_1} G_2 \rightsquigarrow_{\tau_2, p_2, l_2 \to r_2} G_3 \rightsquigarrow_{\theta_2}^{*} \top$$

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

$$G \rightsquigarrow_{\theta_1}^{*} G_1 \rightsquigarrow_{\upsilon_2, p_2, l_2 \to r_2} H_2 \rightsquigarrow_{\upsilon_1, p_1, l_1 \to r_1} H_3 \rightsquigarrow_{\theta_2}^{*} \top$$

with the same complexity such that $G_3 = H_3$ and $\theta_1\tau_1\tau_2\theta_2 = \theta_1\upsilon_2\upsilon_1\theta_2$. $\square$

Lemma 32 *For every normal* NC*-refutation Π:*

$$e, G \rightsquigarrow_{\tau_1}^{*} e_1, G\tau_1 \rightsquigarrow_{\tau_2, 1} e_2, G\tau_1\tau_2 \rightsquigarrow_{\tau_3}^{*} \top$$

with the property that narrowing is not applied to a descendant of e at position 1 in the subderivation that produces substitution τ_1, there exists a normal NC*-refutation $\phi_1(\Pi)$:*

$$e, G \rightsquigarrow_{\upsilon_1}^{*} e_1', G\upsilon_1 \rightsquigarrow_{\upsilon_2, 1} e_2', G\upsilon_1\upsilon_2 \rightsquigarrow_{\upsilon_3}^{*} \top$$

with the same complexity such that $\upsilon_1\upsilon_2\upsilon_3 = \tau_1\tau_2\tau_3$ and narrowing is neither applied at position 1 nor in the right-hand side of a descendant of e in the subderivation that produces the substitution υ_1.

Proof. Let Π' be the subderivation $e, G \rightsquigarrow_{\tau_1}^{*} e_1, G\tau_1 \rightsquigarrow_{\tau_2, 1} e_2, G\tau_1\tau_2$ of Π. Because Π respects $\mathcal{S}_{\text{left}}$ all steps in Π' take place in a descendant of e. If there are steps in Π' such that narrowing is applied to the right-hand side of the descendant of e 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 31, 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. $\square$

Lemma 33 *Let* $e = f(s_1, \ldots, s_n) \approx f(t_1, \ldots, t_n)$*. For every normal* NC*-refutation* $\Pi: e, G \rightsquigarrow^*_{\tau_1} \mathtt{true}, G\tau_1 \rightsquigarrow^*_{\tau_2} \top$ *with the property that narrowing is never applied to a descendant of* e *at position* 1 *or* 2*, there exists a normal* NC*-refutation* $\phi_2(\Pi): e, G \rightsquigarrow^*_{\tau_1} \mathtt{true}, G\tau_1 \rightsquigarrow^*_{\tau_2} \top$ *with the same complexity such that in the subderivation producing substitution* τ_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 $\phi_{[o]} \circ \phi_1$, $\phi_{[i]}$, $\phi_{[d]} \circ \phi_2$, $\phi_{[v]}$, $\phi_{[t]}$, and ϕ_{swap} preserve normality.

Lemma 34 *Let* Π *be a normal* NC*-refutation. The* NC*-refutations* $\phi_{[o]}(\phi_1(\Pi))$*,* $\phi_{[i]}(\Pi)$*,* $\phi_{[d]}(\phi_2(\Pi))$*,* $\phi_{[v]}(\Pi)$*,* $\phi_{[t]}(\Pi)$*, and* $\phi_{\mathrm{swap}}(\Pi)$ *are normal whenever they are defined.*

Proof. First we will show the normality of $\phi_{[o]}(\phi_1(\Pi))$. From Lemma 32 it follows that $\phi_1(\Pi)$, which we can write as

$$s \approx t, G \quad \rightsquigarrow^*_{\tau_1} \quad s' \approx t\tau_1, G\tau_1 \quad \rightsquigarrow_{\tau_2, 1, l \to r} \quad (r \approx t\tau_1, G\tau_1)\tau_2 \quad \rightsquigarrow^*_{\tau_3} \quad \top,$$

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

$$\begin{array}{lll} s \approx l, r \approx t, G & \rightsquigarrow^*_{\tau_1} & s' \approx l, r \approx t\tau_1, G\tau_1 \\ & \rightsquigarrow_{\upsilon_2, x \approx x \to \mathtt{true}} & \mathtt{true}, (r \approx t\tau_1, G\tau_1)\upsilon_2 \\ & = & \mathtt{true}, (r \approx t\tau_1, G\tau_1)\tau_2 \\ & \rightsquigarrow^*_{\tau_3} & \top. \end{array}$$

Here υ_2 is the substitution $\tau_2 \cup \{x \mapsto l\tau_2\}$. We have to show that the condition of Definition 30 holds for every equation in the initial goal $s \approx l, r \approx t, G$ of the NC-refutation $\phi_{[o]}(\phi_1(\Pi))$. Consider the equation $s \approx l$. By construction $\phi_{[o]}(\phi_1(\Pi))$ 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 $s \approx l$. (This is equivalent to saying that the derivation from $s \approx l$ to $s' \approx l$ is non-empty.) We have to show that $\tau_1\upsilon_2\tau_3\restriction_{\mathcal{V}\mathrm{ar}(s)}$ 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 $\tau_1\tau_2\tau_3\restriction_{\mathcal{V}\mathrm{ar}(s)}$ from the normality of $\phi_1(\Pi)$. This implies that $\tau_1\upsilon_2\tau_3\restriction_{\mathcal{V}\mathrm{ar}(s)}$ is normalized since $\tau_1\tau_2\restriction_{\mathcal{V}\mathrm{ar}(s)} = \tau_1\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s)}$. By construction, the descendants of the equation $r \approx t$ and the equations in G are only selected in the common subrefutation $(r \approx t\tau_1, G\tau_1)\tau_2 \rightsquigarrow^*_{\tau_3} \top$ of $\phi_1(\Pi)$ and $\phi_{[o]}(\phi_1(\Pi))$. We conclude that $\phi_{[o]}(\phi_1(\Pi))$ is normal.

Next we consider $\phi_{[i]}$. Let $e = s \approx t$ be an arbitrary equation in the initial goal of the normal NC-refutation Π. We may write Π as

$$G', e, G'' \quad \rightsquigarrow^*_{\tau_1} \quad \top, (e, G'')\tau_1 \quad \rightsquigarrow^*_{\tau_2} \quad \top.$$

By construction, $\phi_{[i]}(\Pi)$ can be written as

$$(G', e, G'')\sigma_1 \quad \rightsquigarrow^*_{\upsilon_1} \quad \top, (e, G'')\sigma_1\upsilon_1 \quad \rightsquigarrow^*_{\upsilon_2} \quad \top.$$

Suppose narrowing is applied to the left-hand side of a descendant of $e\sigma_1$ in $\phi_{[i]}(\Pi)$. (If narrowing is applied to the right-hand side of a descendant of $e\sigma_1$ in $\phi_{[i]}(\Pi)$, the desired result follows in exactly the same way.) We have to show that the substitution $\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s\sigma_1\upsilon_1)}$ is normalized. We know that $\tau_2\restriction_{\mathcal{V}\mathrm{ar}(s\tau_1)}$ is normalized since narrowing is applied to the left-hand side of a descendant of e in the normal NC-refutation Π. Because $\Pi\tau_1\tau_2 = \phi_{[i]}(\Pi)\upsilon_1\upsilon_2$, the terms $s\tau_1\tau_2$ and $s\sigma_1\upsilon_1\upsilon_2$ are equal. Since $\phi_{[i]}(\Pi)$ subsumes Π, $s\sigma_1\upsilon_1$ is an instance of $s\tau_1$ and hence the normalization of $\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s\sigma_1\upsilon_1)}$ is a consequence of the normalization of $\tau_2\restriction_{\mathcal{V}\mathrm{ar}(s\tau_1)}$.

Now consider $\phi_{[d]} \circ \phi_2$. According to Lemma 33 the transformation ϕ_2 preserves normality. We may write $\phi_2(\Pi)$ as

$$s \approx t, G \quad \rightsquigarrow^*_{\tau_1} \quad G\tau_1 \quad \rightsquigarrow^*_{\tau_2} \quad \top$$

and $\phi_{[d]}(\phi_2(\Pi))$ as

$$s_1 \approx t_1, \ldots, s_n \approx t_n, G \quad \rightsquigarrow^*_{\upsilon_1} \quad G\upsilon_1 \quad \rightsquigarrow^*_{\upsilon_2} \quad \top$$

with $s = f(s_1, \ldots, s_n)$, $t = f(t_1, \ldots, t_n)$, and $\upsilon_1\upsilon_2 \leqslant \tau_1\tau_2\ [V]$. By construction $\phi_{[d]}(\phi_2(\Pi))$ respects $\mathcal{S}_{\mathrm{left}}$. Suppose narrowing is applied to the left-hand side of a descendant of $s_i \approx t_i$ in $\phi_{[d]}(\phi_2(\Pi))$, for some $1 \leqslant i \leqslant n$. This implies that narrowing is applied to the left-hand side of a descendant of $s \approx t$ in $\phi_2(\Pi)$. Because narrowing is applied to $s \approx t$ in the first step of $\phi_2(\Pi)$, it follows from the normality of $\phi_2(\Pi)$ that $\tau_1\tau_2\restriction_{\mathcal{V}\mathrm{ar}(s)}$ is normalized. From $\upsilon_1\upsilon_2 \leqslant \tau_1\tau_2\ [V]$ we infer the normalization of $\upsilon_1\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s_i)}$. The first part of $\phi_{[d]}(\phi_2(\Pi))$ can be written as

$$\begin{aligned} s_1 \approx t_1, \ldots, s_n \approx t_n, G \quad &\rightsquigarrow^*_{\sigma_1} \quad \top, (s_i \approx t_i, \ldots, s_n \approx t_n, G)\sigma_1 \\ &\rightsquigarrow^*_{\sigma_2} \quad G\sigma_1\sigma_2 \end{aligned}$$

with $\sigma_1\sigma_2 = \upsilon_1$. The normalization of $\sigma_2\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s_i\sigma_1)}$—which is what we have to show—follows from the normalization of $\sigma_1\sigma_2\upsilon_2\restriction_{\mathcal{V}\mathrm{ar}(s_i)}$ by Lemma 2. Narrowing steps applied to the right-hand side of a descendant of $s_i \approx t_i$ with $1 \leqslant i \leqslant n$ in $\phi_{[d]}(\phi_2(\Pi))$ are treated similarly. The equations in G don't pose any problems since the subrefutations $G\tau_1 \rightsquigarrow^*_{\tau_2} \top$ and $G\upsilon_1 \rightsquigarrow^*_{\upsilon_2} \top$ of $\phi_2(\Pi)$ and $\phi_{[d]}(\phi_2(\Pi))$ differ at most a renaming.

The transformations $\phi_{[v]}$ and $\phi_{[t]}$ are easily seen to preserve normality. Finally, ϕ_{swap} trivially preserves normality. $\square$

Example 35 Consider again the NC-refutation Π_{fail} of Example 23. This refutation is easily seen to be normal. An application of $\phi_{[o]}$ results in the NC-refutation $\phi_{[o]}(\Pi_{\mathrm{fail}})$:

$$\begin{aligned} f(a) \approx f(x), x \approx g(a) \quad &\rightsquigarrow \quad f(a) \approx f(x), x \approx g(b) \\ &\rightsquigarrow \quad \mathtt{true}, a \approx g(b) \\ &\rightsquigarrow \quad \mathtt{true}, b \approx g(b) \\ &\rightsquigarrow \quad \mathtt{true}, g(b) \approx g(b) \\ &\rightsquigarrow \quad \top \end{aligned}$$

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

$$\begin{aligned} f(a) \approx g(a) \quad &\rightsquigarrow \quad a \approx g(a) \quad \rightsquigarrow \quad a \approx g(b) \quad \rightsquigarrow \quad b \approx g(b) \\ &\rightsquigarrow \quad g(b) \approx g(b) \quad \rightsquigarrow \quad \mathtt{true}. \end{aligned}$$

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

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

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

Now we are ready to prove the counterpart of Lemma 25 for normal NC-refutations:

$$\begin{array}{llllll} \forall & \text{normal} & \Pi & : & G & \rightsquigarrow_{\theta}^{+} & \top \\ \exists & & \Psi_1 & : & \Downarrow_{\sigma_1} & & \\ \exists & \text{normal} & \Pi_1 & : & G_1 & \rightsquigarrow_{\theta_1}^{*} & \top \end{array} \quad \text{such that} \quad \left\{ \begin{array}{l} \sigma_1\theta_1 \leqslant \theta \; [V] \\ \Psi_1 \text{ respects } \mathcal{S}_{\mathrm{left}} \\ \Pi \gg \Pi_1 \end{array} \right.$$

Lemma 36 *For every non-empty normal* NC*-refutation* $\Pi: G \rightsquigarrow_{\theta}^{+} \top$ *there exist an* LNC*-step* $\Psi_1: G \Rightarrow_{\sigma_1} G_1$ *respecting* $\mathcal{S}_{\mathrm{left}}$ *and a normal* NC*-refutation* $\Pi_1: G_1 \rightsquigarrow_{\theta_1}^{*} \top$ *such that* $\sigma_1\theta_1 \leqslant \theta \; [V]$ *and* $\Pi \gg \Pi_1$.

Proof. The proof is very similar to that of Lemma 25. Since we refer to the proof in the next section, we nevertheless present it in full detail. We distinguish the following cases, depending on what happens to the selected equation $e = s \approx t$ in the first step of Π. Let $G = e, G'$.

(1) Suppose narrowing is never applied to a descendant of e at position 1 or 2. We distinguish four further cases.
  - (a) Suppose $s, t \notin \mathcal{V}$. We may write $s = f(s_1, \ldots, s_n)$ and $t = f(t_1, \ldots, t_n)$. Let $G_1 = s_1 \approx t_1, \ldots, s_n \approx t_n, G'$. We have Ψ_1: $G \Rightarrow_{[d]} G_1$. An application of Lemma 33 followed by Lemma 15 yields an NC-refutation $\phi_{[d]}(\phi_2(\Pi)): G_1 \rightsquigarrow_{\theta_1}^{*} \top$ such that $\theta_1 \leqslant \theta \; [V]$. Take $\sigma_1 = \epsilon$.
  - (b) Suppose $t \in \mathcal{V}$ and $s = t$. In this case the first step of Π_1 must take place at the root of e. Let $G_1 = G'$. We have $\Psi_1: G \Rightarrow_{[t]} G_1$. Lemma 17 yields an NC-refutation $\phi_{[t]}(\Pi): G_1 \rightsquigarrow_{\theta_1}^{*} \top$ such that $\theta_1 \leqslant \theta \; [V]$. Take $\sigma_1 = \epsilon$.
  - (c) Suppose $t \in \mathcal{V}$ and $s \neq t$. We distinguish two further cases, depending on what happens to e in the first step of Π.
    - (i) Suppose narrowing is applied to e at the root position. Let $\sigma_1 = \{t \mapsto s\}$ and $G_1 = G'\sigma_1$. We have $\Psi_1: G \Rightarrow_{[v], \sigma_1} G_1$. Lemma 16 yields an NC-refutation $\phi_{[v]}(\Pi): G_1 \rightsquigarrow_{\theta_1}^{*} \top$ such that $\sigma_1\theta_1 \leqslant \theta \; [V]$.
    - (ii) Suppose narrowing is not applied to e at the root position. This implies that $s \notin \mathcal{V}$. Hence we may write $s = f(s_1, \ldots, s_n)$. Let $\sigma_1 = \{t \mapsto f(x_1, \ldots, x_n)\}$, $G_1 = (s_1 \approx x_1, \ldots, s_n \approx x_n, G')\sigma_1$, and $G_2 = G\sigma_1$. Here $x_1, \ldots, x_n$ are fresh variables. We have $\Psi_1: G \Rightarrow_{[i], \sigma_1} G_1$. From Lemma 14 we obtain an NC-refutation $\Pi_2 = \phi_{[i]}(\Pi): G_2 \rightsquigarrow_{\theta_2}^{*} \top$ such that $\sigma_1\theta_2 = \theta \; [V]$. Let $V_2 = V \cup \{x_1, \ldots, x_n\}$. Clearly $\mathcal{V}\mathrm{ar}(G_2) \subseteq V_2$. An application of Lemma 33 to Π_2 followed by Lemma 15 results in an NC-refutation $\Pi_1 = \phi_{[d]}(\phi_2(\Pi_2)): G_1 \rightsquigarrow_{\theta_1}^{*} \top$ such that

$\theta_1 \leqslant \theta_2\ [V_2]$. Using the inclusion $(V \setminus \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1\restriction_V) \subseteq V_2$ we obtain $\sigma_1\theta_1 \leqslant \sigma_1\theta_2 = \theta\ [V]$ from Lemma 1.

(d) In the remaining case we have $t \notin \mathcal{V}$ and $s \in \mathcal{V}$. This case reduces to case (1)(c) by an appeal to Lemma 21.

(2) Suppose narrowing is applied to a descendant of e at position 1. Let $l = f(l_1, \ldots, l_n) \to r$ be the used rewrite rule the first time this happens. Because Π is normal, s cannot be a variable. Hence we may write $s = f(s_1, \ldots, s_n)$. Let $G_1 = s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, G'$ and $G_2 = s \approx l, r \approx t, G'$. We have $\Psi_1 \colon G \Rightarrow_{[o]} G_1$. An application of Lemma 32 followed by Lemma 13 yields an an NC-refutation $\Pi_2 = \phi_{[o]}(\phi_1(\Pi)) \colon G_2 \rightsquigarrow^*_{\theta_2} \top$ such that $\theta_2 = \theta\ [V]$. Let $V_2 = V \cup \mathcal{V}\mathrm{ar}(l)$. Clearly $\mathcal{V}\mathrm{ar}(G_2) \subseteq V_2$. An application of Lemma 33 followed by Lemma 15 to Π_2 results in an NC-refutation $\Pi_1 = \phi_{[d]}(\phi_2(\Pi_2)) \colon G_1 \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 \leqslant \theta_2\ [V_2]$. Using $V \subseteq V_2$ we obtain $\theta_1 \leqslant \theta\ [V]$. Take $\sigma_1 = \epsilon$.

(3) Suppose narrowing is applied to a descendant of e at position 2. This case reduces to the previous one by an appeal to Lemma 21.

The above case analysis is summarized in Table 2. In all cases we obtain Π_1 from Π by applying one or more transformations $\phi_{[o]} \circ \phi_1$, $\phi_{[i]}$, $\phi_{[d]} \circ \phi_2$, $\phi_{[v]}$, $\phi_{[t]}$ together with an additional application of ϕ_{swap} in case (1)(d) and (3). According to Lemma 34 Π_1 is normal. According to Lemmata 20, 21, 32, and 33 Π_1 has smaller complexity than Π. $\square$

case	LNC-step	transformation(s)
(1)(a)	$\Rightarrow_{[d]}$	$\phi_{[d]} \circ \phi_2$
(1)(b)	$\Rightarrow_{[t]}$	$\phi_{[t]}$
(1)(c)(i)	$\Rightarrow_{[v]}$	$\phi_{[v]}$
(1)(c)(ii)	$\Rightarrow_{[i]}$	$\phi_{[d]} \circ \phi_2 \circ \phi_{[i]}$
(1)(d)	$\Rightarrow_{[v]}$ or $\Rightarrow_{[i]}$	$\phi_{[v]} \circ \phi_{\mathsf{swap}}$ or $\phi_{[d]} \circ \phi_2 \circ \phi_{[i]} \circ \phi_{\mathsf{swap}}$
(2)	$\Rightarrow_{[o]}$	$\phi_{[d]} \circ \phi_2 \circ \phi_{[o]} \circ \phi_1$
(3)	$\Rightarrow_{[o]}$	$\phi_{[d]} \circ \phi_2 \circ \phi_{[o]} \circ \phi_1 \circ \phi_{\mathsf{swap}}$

Table 2: Case analysis in the proof of Lemma 36.

Lemma 38 below is the counterpart of Lemma 27 for normal NC-refutations. The proof is an easy consequence of the following switching lemma, whose proof can be found in the appendix.

Lemma 37 *Let G_1 be a goal containing distinct equations e_1 and e_2. For every* NC*-refutation*

$$G \rightsquigarrow^*_{\theta_1} G_1 \rightsquigarrow_{\tau_1, e_1, p_1, l_1 \to r_1} G_2 \rightsquigarrow_{\tau_2, e_2\tau_1, p_2, l_2 \to r_2} G_3 \rightsquigarrow^*_{\theta_2} \top$$

with $\theta_1\tau_1\tau_2\theta_2\restriction_{\mathcal{V}\mathrm{ar}(G)}$ normalized there exists a NC*-refutation*

$$G \rightsquigarrow^*_{\theta_1} G_1 \rightsquigarrow_{\upsilon_2, e_2, p_2, l_2 \to r_2} H_2 \rightsquigarrow_{\upsilon_1, e_1\upsilon_2, p_1, l_1 \to r_1} H_3 \rightsquigarrow^*_{\theta_2} \top$$

with the same complexity such that $G_3 = H_3$ and $\theta_1\tau_1\tau_2\theta_2 = \theta_1\upsilon_2\upsilon_1\theta_2$. $\square$

Lemma 38 *For every* NC*-refutation* $\Pi\colon G \rightsquigarrow_\theta^* \top$ *that produces a normalized substitution there exists a normal* NC*-refutation* $\phi_{\text{normal}}(\Pi)\colon G \rightsquigarrow_\theta^* \top$ *with the same complexity.*

Proof. Repeated applications of Lemma 37 results in an NC-refutation $\phi_{\text{normal}}(\Pi)\colon G \rightsquigarrow_\theta^* \top$ that respects $\mathcal{S}_{\text{left}}$ with the same complexity as Π. Because $\theta\lceil_{\mathcal{V}\text{ar}(G)}$ is normalized, it follows that $\phi_{\text{normal}}(\Pi)$ is normal. □

Putting all the pieces together, the following result can be proved along the lines of the proof of Theorem 28.

Theorem 39 *For every* NC*-refutation* $\Pi\colon G \rightsquigarrow_\theta^* \top$ *with the property that* $\theta\lceil_{\mathcal{V}\text{ar}(G)}$ *is normalized there exists an* LNC*-refutation* $\Psi\colon G \Rightarrow_\sigma^* \top$ *respecting* $\mathcal{S}_{\text{left}}$ *such that* $\sigma \leqslant \theta\ [\mathcal{V}\text{ar}(G)]$.

Proof. Very similar to the proof of Theorem 28. The only difference is the use of normal rather than basic NC-refutations, and Lemmata 36 and 38 instead of 25 and 27. □

Corollary 40 *Let* $\mathcal{R}$ *be a confluent TRS and* G *a goal. For every normalized solution* θ *of* G *there exists an* LNC*-refutation* $G \Rightarrow_\sigma^* \square$ *respecting* $\mathcal{S}_{\text{left}}$ *such that* $\sigma \leqslant \theta\ [\mathcal{V}\text{ar}(G)]$. □

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 $[o]$). 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 2 shows all LNC-refutations issued from the goal $f(b) \approx a$ that respect the selection function $\mathcal{S}_{\text{left}}$. There are infinitely many such refutations. Because the initial goal is ground, 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 $\mathcal{S}_{\text{left}}$ 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 11. The selected equation $f(s_1, \ldots, s_n) \simeq t$ in the outermost

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

Figure 2: The LNC-refutations starting from $f(b) \approx a$ that respect $\mathcal{S}_{\text{left}}$.

narrowing rule $[o]$ has the body equation $r \approx t$ as only (one-step) descendant. In the imitation rule $[i]$, 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. Observe that every equation in an LNC-derivation descends from either a parameter-passing equation or an equation in the initial goal.

Definition 41 An equation of the form $x \simeq t$, with $x \notin \mathcal{V}\mathrm{ar}(t)$, is called *solved*. An LNC-derivation Ψ 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 Ψ.

Note that the above concept of eager LNC-derivation doesn't cover the full eager variable elimination problem due to the restriction to descendants of parameter-passing equations that we impose. Of the infinitely many LNC-refutations in Figure 2 only the leftmost one is eager since all others apply the outermost narrowing rule $[o]$ to the solved descendant $b \approx x$ of the parameter-passing equation $b \approx g(x)$ introduced in the first $\Rightarrow_{[o]}$-step.

In this section we prove that eager LNC is complete with respect to $\mathcal{S}_{\text{left}}$ 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 ϕ_1, ϕ_2, $\phi_{[o]}$, $\phi_{[i]}$, $\phi_{[d]}$, $\phi_{[v]}$, $\phi_{[t]}$, and ϕ_{swap} 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.

Definition 42 Let $\mathcal{R}$ be an orthogonal TRS. An $\mathcal{R}_+$-rewrite sequence

$$e_1 \quad \to_{p_1, l_1 \to r_1} \quad \cdots \quad \to_{p_{n-1}, l_{n-1} \to r_{n-1}} \quad e_n$$

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 $\varepsilon < p_j < p_i$ then $p_i \backslash p_j \in \mathcal{P}os_{\mathcal{F}}(l_j)$ for the least such j.

In an outside-in sequence every redex contraction p_i contributes either directly to the final result (if there is no j with $i < j < n$ such that $\varepsilon < p_j < p_i$) or to the *creation* of a redex at a position $\varepsilon < p_j < p_i$ with $i < j < n$. (The exclusion of ε in Definition 42 stems from the fact that the only applicable rewrite rule at that position is not left-linear.)

The above 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 43 *Let $\mathcal{R}$ be an orthogonal TRS and e an equation. For every rewrite sequence $e \to^*_{\mathcal{R}_+}$* `true` *there exists an outside-in rewrite sequence $e \to^*_{\mathcal{R}_+}$* `true`*.* $\square$

Definition 44 Let $\mathcal{R}$ be an orthogonal TRS. An NC-refutation $\Pi\colon G \rightsquigarrow^*_\theta \top$ is called *outside-in* if for every equation $e \in G$ the rewrite sequence $e\theta \to^*$ `true` in $\Pi\theta$ is outside-in. In this case we say also that the rewrite sequence $\Pi\theta\colon G\theta \to^* \top$ is outside-in.

Example 45 Consider the orthogonal TRS

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

The NC-refutation Π:

$$\begin{array}{lllll} f(a) \approx y, f(y) \approx b & \rightsquigarrow & a \approx y, f(y) \approx b & \rightsquigarrow & a \approx y, y \approx b \\ & \rightsquigarrow & \mathtt{true}, a \approx b & \rightsquigarrow & \mathtt{true}, b \approx b \quad \rightsquigarrow \quad \top \end{array}$$

is outside-in, because the two rewrite sequences $f(a) \approx a \to a \approx a \to \mathtt{true}$ and $f(a) \approx b \to a \approx b \to b \approx b \to \mathtt{true}$ are outside-in. The NC-refutation Π':

$$\begin{array}{lllll} f(a) \approx y, f(y) \approx b & \rightsquigarrow & a \approx y, f(y) \approx b & \rightsquigarrow & \mathtt{true}, f(a) \approx b \\ & \rightsquigarrow & \mathtt{true}, f(b) \approx b & \rightsquigarrow & \mathtt{true}, b \approx b \quad \rightsquigarrow \quad \top \end{array}$$

is not outside-in since the rewrite sequence $f(a) \approx b \to f(b) \approx b \to b \approx b \to \mathtt{true}$ isn't.

Theorem 46 *Let $\mathcal{R}$ be an orthogonal TRS and G a goal. For every normalized solution θ of G there exists an outside-in* NC*-refutation $G \rightsquigarrow^*_{\theta'} \top$ such that $\theta' \leqslant \theta\ [\mathcal{V}\mathrm{ar}(G)]$.*

Proof. Let $G\theta = e_1, \ldots, e_n$. The rewrite sequence $G\theta \to^*_{\mathcal{R}_+} \top$ can be partitioned into rewrite sequences from e_i to `true` for $1 \leqslant i \leqslant n$. To each of these n rewrite sequences we apply Theorem 43, yielding outside-in rewrite sequences from e_i to `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 $\theta_1 = \theta\lceil_{\mathcal{V}\mathrm{ar}(G)}$. Evidently, $G\theta_1 = G\theta$ and θ_1 is normalized. An application of the lifting lemma for NC—Lemma 5—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 $\theta' \leqslant \theta_1 = \theta\ [\mathcal{V}\mathrm{ar}(G)]$. $\square$

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 46. 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 47 *The transformations ϕ_1, ϕ_2, $\phi_{[o]}$, $\phi_{[i]}$, $\phi_{[d]}$, $\phi_{[v]}$, $\phi_{[t]}$, ϕ_{swap}, and ϕ_{normal} preserve the outside-in property.*

Proof. Straightforward by inspecting the various transformations. $\square$

We define a property $\mathcal{P}$ of equations in the initial goal of NC-refutations in Definition 48. In Lemma 49 we show that parameter-passing equations introduced in the transformation proof of Lemma 36 satisfy this property, provided we start from an NC-refutation Π that is outside-in. In Lemma 51 the property is shown to be preserved by LNC-descendants obtained during the transformation proof. Finally, in Lemma 52 it is shown that Ψ_1 in Lemma 36 consists of a $\Rightarrow_{[v]}$-step whenever the selected (leftmost) equation in Π is solved and satisfies the property.

Definition 48 Let $\Pi\colon G \rightsquigarrow^* \top$ be an NC-refutation and $e \in G$. We say that e has property $\mathcal{P}$ in Π if the following two conditions 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 $2{\cdot}p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e)$.

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

In the following three lemmata, Ψ_1 and Π_1 refer to the LNC-step and the NC-refutation constructed in Lemma 36.

Lemma 49 *Let* $\Pi\colon G \rightsquigarrow^{+} \top$ *be a normal outside-in* NC*-refutation. If* Ψ_1 *consists of an* $\Rightarrow_{[o]}$*-step then every parameter-passing equation has property* $\mathcal{P}$ *in* Π_1.

Proof. According to Table 2 we have to consider cases (2) and (3) in the proof of Lemma 36. We consider here only case (2). Consider a parameter-passing equation $s_i \approx l_i$ in Π_1. The first part of property $\mathcal{P}$ holds by construction. Suppose narrowing is applied to the left-hand side of a descendant of $s_i \approx l_i$ in Π_1. Let $1{\cdot}p$ be a critical position. We have to show that $2{\cdot}p \in \mathcal{P}os_{\mathcal{F}}(s_i \approx l_i)$. The NC-refutation $\phi_1(\Pi)$ can be written as

$$\begin{array}{lll} e, G' & \rightsquigarrow^{*}_{\tau_1} & e_1, G'\tau_1 \\ & \rightsquigarrow_{\tau_2,\,1\cdot i\cdot p} & e_2, G'\tau_1\tau_2 \\ & \rightsquigarrow^{*}_{\tau_3} & e_3, G'\tau_1\tau_2\tau_3 \\ & \rightsquigarrow_{\tau_4,\,1,\,l\to r} & e_4, G'\tau_1\tau_2\tau_3\tau_4 \\ & \rightsquigarrow^{*}_{\tau_5} & \top \end{array}$$

where all narrowing steps in the subderivation $e_2, G'\tau_1\tau_2 \rightsquigarrow^{*}_{\tau_3} e_3, G'\tau_1\tau_2\tau_3$ don't take place at positions above $1{\cdot}i{\cdot}p$. According to Lemma 47 $\phi_1(\Pi)$ is outside-in. Hence, by definition, $1{\cdot}i{\cdot}p\backslash 1 \in \mathcal{P}os_{\mathcal{F}}(l) = \mathcal{P}os_{\mathcal{F}}(f(l_1,\ldots,l_n))$. Therefore $p \in \mathcal{P}os_{\mathcal{F}}(l_i)$ and thus $2{\cdot}p \in \mathcal{P}os_{\mathcal{F}}(s_i \approx l_i)$. $\Box$

The following example shows the necessity of the outside-in property in Lemma 49.

Example 50 Consider the TRS of Example 45. We have

$$\begin{array}{llccccccc} \Pi & : & f(a) \approx b & \rightsquigarrow & f(b) \approx b & \rightsquigarrow & b \approx b & \rightsquigarrow & \mathtt{true} \\ \Psi_1 & : & \Downarrow_{[o]} & & & & & & \\ \Pi_1 & : & a \approx x, x \approx b & \rightsquigarrow & b \approx x, x \approx b & \rightsquigarrow & \mathtt{true}, b \approx b & \rightsquigarrow & \top \end{array}$$

The NC-refutation Π is not outside-in and the parameter-passing equation $a \approx x$ does not satisfy property $\mathcal{P}$ in Π_1 as position 1 is critical while $2 \notin \mathcal{P}os_{\mathcal{F}}(a \approx x)$. Transforming Π into the outside-in NC-refutation Π' results in the following diagram:

$$\begin{array}{llccccccc} \Pi' & : & f(a) \approx b & \rightsquigarrow & a \approx b & \rightsquigarrow & b \approx b & \rightsquigarrow & \mathtt{true} \\ \Psi_1' & : & \Downarrow_{[o]} & & & & & & \\ \Pi_1' & : & a \approx x, x \approx b & \rightsquigarrow & \mathtt{true}, a \approx b & \rightsquigarrow & \mathtt{true}, b \approx b & \rightsquigarrow & \top \end{array}$$

The parameter-passing equation $a \approx x$ does have property $\mathcal{P}$ in Π_1'.

Lemma 51 *Suppose* $\Pi\colon G \rightsquigarrow^{+} \top$ *is a normal* NC*-refutation and let* $e \in G$ *have property* $\mathcal{P}$*. If* e' *is a* Ψ_1*-descendant of* e *then* e' *has property* $\mathcal{P}$ *in* Π_1.

Proof. First we consider the case that $e \in G$ is the selected equation in Ψ_1. Consider the case analysis in the proof of Lemma 36. In cases (1)(b), (1)(c)(i), (1)(d), and (3) there is nothing to show: either e has no Ψ_1-descendants or the first part of the property $\mathcal{P}$ doesn't hold. In case (1)(a) we have $\Pi_1 = \phi_{[d]}(\phi_2(\Pi))$. It is easy to see that equation e has property $\mathcal{P}$ in $\phi_2(\Pi)$. We have $e' = s_i \approx t_i$ for some $1 \leqslant i \leqslant n$. The first part of property $\mathcal{P}$

clearly holds for e' in Π_1. 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 $2{\cdot}i{\cdot}p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e)$ from the fact that e has property $\mathcal{P}$ in $\phi_2(\Pi)$. This implies $2{\cdot}p \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e')$. We conclude that e' has property $\mathcal{P}$ in Π_1. In case (1)(c)(ii) narrowing is applied to the left-hand side s of e in Π. This implies that there is a critical position $1{\cdot}p$. Since the right-hand side t of e is a variable, $2{\cdot}p \notin \mathcal{P}\mathrm{os}_{\mathcal{F}}(e)$. Therefore the equation e doesn't have the property $\mathcal{P}$. It remains to consider case (2). We have $\Pi_1 = \phi_{[d]}(\phi_2(\Pi_2)) = \phi_{[d]}(\phi_2(\phi_{[o]}(\phi_1(\Pi))))$. It is not difficult to see that e has property $\mathcal{P}$ in $\phi_1(\Pi)$. From the construction of $\phi_{[o]}$ we learn that the equation $r \approx t$ in the initial goal G_2 of Π_2 inherits the property $\mathcal{P}$ from e in Π. Since the rewrite sequence $(r \approx t)\theta_1 \to^* \mathtt{true}$ in $\Pi_1\theta_1$ subsumes the rewrite sequence $(r \approx t)\theta_2 \to^* \mathtt{true}$ in $\Pi_2\theta_2$, it follows that the (unique) Ψ_1-descendant $e' = r \approx t$ of e has the property $\mathcal{P}$ in Π_1.

Next suppose that $e \in G$ is not selected in Ψ_1. By comparing the rewrite sequences $e\theta \to^* \mathtt{true}$ in $\Pi\theta$ and $e'\theta_1 \to^* \mathtt{true}$ in $\Pi_1\theta_1$, one easily concludes that in all cases in the proof of Lemma 36 the (unique) Ψ_1-descendent e' of e inherits the property $\mathcal{P}$ of e. $\square$

Lemma 52 *Suppose* $\Pi\colon G \rightsquigarrow^+ \top$ *is a normal* NC*-refutation and let the selected (leftmost) equation* e *have the property* $\mathcal{P}$*. If* e *is solved then* Ψ_1 *consists of a* $\Rightarrow_{[v]}$*-step.*

Proof. Consider the case analysis in the proof of Lemma 36. In cases (1)(a) and (1)(b) the selected equation e is not solved. In case (1)(c)(i) Ψ_1 consists indeed of a $\Rightarrow_{[v]}$-step. In the proof of Lemma 51 we already observed that in case (1)(c)(ii) the equation e doesn't have the property $\mathcal{P}$. In case (1)(d) either Ψ_1 consists of a $\Rightarrow_{[v]}$-step or e doesn't have the property $\mathcal{P}$, just as in case (1)(c). In case (2), the equation e is not solved or doesn't have property $\mathcal{P}$. The latter follows as in case (1)(c)(ii). Finally, in case (3) the selected equation e doesn't have the property $\mathcal{P}$ because narrowing is applied to the right-hand side of a descendant of e. $\square$

Theorem 53 *For every outside-in* NC*-refutation* $G \rightsquigarrow_\theta^* \top$ *with* $\theta\restriction_{\mathcal{V}\mathrm{ar}(G)}$ *normalized there exists an eager* LNC*-refutation* $G \Rightarrow_\sigma^* \top$ *respecting* $\mathcal{S}_{\mathrm{left}}$ *such that* $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$*.*

Proof. Let Π be the given outside-in NC-refutation $G \rightsquigarrow_\theta^* \top$. From Theorem 39 we obtain an LNC-refutation $\Psi\colon G \Rightarrow_\sigma^* \top$ respecting $\mathcal{S}_{\mathrm{left}}$ such that $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$. From Lemmata 49–52 we learn that the variable elimination rule $[v]$ is applied to all selected solved descendants of parameter-passing equations in Ψ, i.e., Ψ is eager. $\square$

The combination of Theorems 46 and 53 yields the final result of this paper.

Corollary 54 *Let* $\mathcal{R}$ *be an orthogonal TRS and* G *a goal. For every normalized solution* θ *of* G *there exists an eager* LNC*-refutation* $G \Rightarrow_\sigma^* \top$ *respecting* $\mathcal{S}_{\mathrm{left}}$ *such that* $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$*.* $\square$

7 Suggestions for Further Research

This paper leaves many questions unanswered. We mention some of them below.

(1) 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 $\mathcal{S}_{\text{left}}$. 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.

(2) 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 $\mathcal{S}_{\text{left}}$ 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 45 and the goal $f(a) \approx b$. There are two different NC-refutations starting from this goal:

$$\Pi_1\colon \quad f(a) \approx b \quad \rightsquigarrow \quad a \approx b \quad \rightsquigarrow \quad b \approx b \quad \rightsquigarrow \quad \texttt{true}$$

and

$$\Pi_2\colon \quad f(a) \approx b \quad \rightsquigarrow \quad f(b) \approx b \quad \rightsquigarrow \quad b \approx b \quad \rightsquigarrow \quad \texttt{true}.$$

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.

(3) 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?

(4) 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 $\{x \mapsto b\}$. Recently Middeldorp and Okui [18] showed that all non-determinism in the choice of the inference rule for descendants of parameter-passing equations can be removed for orthogonal constructor-based TRSs, whereas complete determinism in the choice of the inference rule for descendants of equations in the initial goal can be achieved for arbitrary confluent TRSs if we interpret $\approx$ as *strict equality*, meaning that we only require completeness with respect to solutions θ of G that have the property that $s\theta$ and $t\theta$ have the same ground constructor normal form, for every equation $s \approx t \in G$.

(5) The results reported in this paper should be extended to conditional TRSs. Incorporating conditional rewrite rules into LNC is easy: simply add the conditions of the rewrite rule used in the outermost narrowing rule [o] to the resulting goal. Of the three main results in this paper—Theorems 28, 39, and 53—we expect that the first one can be lifted to conditional TRSs without much difficulties, although extra variables in the right-hand sides of the conditional rewrite rules might prove to be a complication. The second result should also hold in the conditional case, but it is less clear whether the proof technique developed in Section 5 can be extended. We do not know whether Theorem 53 holds for conditional TRSs.

Acknowledgement

We would like to thank Christopher Lynch and the referees for their helpful remarks.

References

[1] L. Bachmair, H. Ganzinger, C. Lynch, and W. Snyder, Basic Paramodulation, *Information and Computation* **121** (1995) 172–192.

[2] N. Dershowitz and J.-P. Jouannaud, Rewrite Systems, in: J. van Leeuwen, ed., *Handbook of Theoretical Computer Science*, Vol. B (North-Holland, Amsterdam, 1990) 243–320.

[3] N. Dershowitz and Z. Manna, Proving Termination with Multiset Orderings, *Communications of the ACM* **22** (1979) 465–476.

[4] M. Fay, First-Order Unification in Equational Theories, in: Proceedings 4th Conference on Automated Deduction, (1979) 161–167.

[5] J. Gallier and W. Snyder, Complete Sets of Transformations for General E-Unification, *Theoretical Computer Science* **67** (1989) 203–260.

[6] M. Hanus, Efficient Implementation of Narrowing and Rewriting, in: Proceedings International Workshop on Processing Declarative Knowledge, *Lecture Notes in Artificial Intelligence*, Vol. 567 (Springer, Berlin, 1991) 344–365.

[7] M. Hanus, The Integration of Functions into Logic Programming: From Theory to Practice, *Journal of Logic Programming* **19 & 20** (1994) 583–628.

[8] S. Hölldobler, A Unification Algorithm for Confluent Theories, in: Proceedings 14th International Colloquium on Automata, Languages and Programming, *Lecture Notes in Computer Science*, Vol. 267 (Springer, Berlin, 1987) 31–41.

[9] S. Hölldobler, Foundations of Equational Logic Programming, *Lecture Notes in Artificial Intelligence*, Vol. 353 (Springer, Berlin, 1989).

[10] G. Huet and J.-J. Lévy, Computations in Orthogonal Rewriting Systems, I and II, in: J.-L. Lassez and G. Plotkin, eds., *Computational Logic, Essays in Honor of Alan Robinson*, (The MIT Press, 1991) 396–443.

[11] J.-M. Hullot, Canonical Forms and Unification, in: Proceedings 5th Conference on Automated Deduction, *Lecture Notes in Computer Science*, Vol. 87 (Springer, Berlin, 1980) 318–334.

[12] J.W. Klop, Term Rewriting Systems, in: S. Abramsky, D. Gabbay and T. Maibaum, eds., *Handbook of Logic in Computer Science*, Vol. II (Oxford University Press, 1992) 1–112.

[13] J.W. Lloyd, *Foundations of Logic Programming*, 2nd edition (Springer, Berlin, 1987).

[14] A. Martelli, C. Moiso, and G.F. Rossi, Lazy Unification Algorithms for Canonical Rewrite Systems, in: H. Aït-Kaci and M. Nivat, eds., *Resolution of Equations in Algebraic Structures, Vol. II, Rewriting Techniques*, (Academic Press, 1989) 245–274.

[15] A. Martelli and U. Montanari, An Efficient Unification Algorithm, *ACM Transactions on Programming Languages and Systems* **4** (1982) 258–282.

[16] A. Martelli, G.F. Rossi, and C. Moiso, An Algorithm for Unification in Equational Theories, in: Proceedings 1986 Symposium on Logic Programming, (1986) 180–186.

[17] A. Middeldorp and E. Hamoen, Completeness Results for Basic Narrowing, *Applicable Algebra in Engineering, Communication and Computing* **5** (1994) 213–253.

[18] A. Middeldorp and S. Okui, A Deterministic Lazy Narrowing Calculus, in: *Proceedings Fuji International Workshop on Functional and Logic Programming*, (World Scientific, Singapore, 1995) 104–118.

[19] M. Moser, Improving Transformation Systems for General E-Unification, in: Proceedings 5th International Conference on Rewriting Techniques and Applications, *Lecture Notes in Computer Science*, Vol. 690 (Springer, Berlin, 1993) 92–105.

[20] J.A. Robinson, A Machine-Oriented Logic Based on the Resolution Principle, *Journal of the ACM* **12** (1965) 23–41.

[21] J.R. Slagle, Automatic Theorem Proving in Theories with Simplifiers, Commutativity and Associativity, *Journal of the ACM* **21** (1974) 622–642.

[22] W. Snyder, *A Proof Theory for General Unification*, (Birkhäuser, 1991).

[23] Y.H. You, Enumerating Outer Narrowing Derivations for Constructor Based Term Rewriting Systems, *Journal of Symbolic Computation* **7** (1989) 319–343.

Appendix

This appendix contains proofs of Lemmata 26, 31, and 37.

Lemma 26. *Let G_1 be a goal containing distinct equations e_1 and e_2. For every* NC*-derivation*

$$G_1 \quad \leadsto_{\tau_1, e_1, p_1, l_1 \to r_1} \quad G_2 \quad \leadsto_{\tau_2, e_2\tau_1, p_2, l_2 \to r_2} \quad G_3$$

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

$$G_1 \quad \leadsto_{\upsilon_2, e_2, p_2, l_2 \to r_2} \quad H_2 \quad \leadsto_{\upsilon_1, e_1\upsilon_2, p_1, l_1 \to r_1} \quad H_3$$

such that $G_3 = H_3$ and $\tau_1\tau_2 = \upsilon_2\upsilon_1$.

Proof. Let $G_1 = G', e_1, G'', e_2, G'''$. Clearly $G_3 = ((G', e_1[r_1]_{p_1}, G'')\tau_1, e_2\tau_1[r_2]_{p_2}, G'''\tau_1)\tau_2$. Since we may assume that the variables in l_2 are fresh, we have $\mathcal{D}(\tau_1) \cap \mathcal{V}\mathrm{ar}(l_2) = \varnothing$. Hence

$$e_{2|p_2}\tau_1\tau_2 \quad = \quad e_2\tau_{1|p_2}\tau_2 \quad = \quad l_2\tau_2 \quad = \quad l_2\tau_1\tau_2.$$

So $e_{2|p_2}$ and l_2 are unifiable. Let υ_2 be an idempotent most general unifier of these two terms. We have $H_2 = (G', e_1, G'', e_2[r_2]_{p_2}, G''')\upsilon_2$. There exists a substitution ρ such that $\upsilon_2\rho = \tau_1\tau_2$. We have $\mathcal{D}(\upsilon_2) \subseteq \mathcal{V}\mathrm{ar}(e_{2|p_2}) \cup \mathcal{V}\mathrm{ar}(l_2)$. Because we may assume that $\mathcal{V}\mathrm{ar}(l_1) \cap \mathcal{V}\mathrm{ar}(e_2) = \varnothing$, we obtain $\mathcal{D}(\upsilon_2) \cap \mathcal{V}\mathrm{ar}(l_1) = \varnothing$. Hence

$$e_1\upsilon_{2|p_1}\rho \quad = \quad e_{1|p_1}\upsilon_2\rho \quad = \quad e_{1|p_1}\tau_1\tau_2 \quad = \quad l_1\tau_1\tau_2 \quad = \quad l_1\upsilon_2\rho \quad = \quad l_1\rho.$$

So the terms $e_1\upsilon_{2|p_1}$ and l_1 are unifiable. Let σ be an idempotent most general unifier. We have $\sigma \leqslant \rho$. It follows that $\upsilon_2\sigma \leqslant \tau_1\tau_2$. Using $\mathcal{D}(\upsilon_2) \cap \mathcal{V}\mathrm{ar}(l_1) = \varnothing$ we obtain

$$e_{1|p_1}\upsilon_2\sigma \quad = \quad e_1\upsilon_{2|p_1}\sigma \quad = \quad l_1\sigma \quad = \quad l_1\upsilon_2\sigma,$$

so $\upsilon_2\sigma$ is a unifier of $e_{1|p_1}$ and l_1. Because τ_1 is a most general unifier of these two terms, we must have $\tau_1 \leqslant \upsilon_2\sigma$. Let γ be any substitution satisfying $\tau_1\gamma = \upsilon_2\sigma$. With help of $\mathcal{D}(\tau_1) \cap \mathcal{V}\mathrm{ar}(l_2) = \varnothing$ we obtain

$$e_2\tau_{1|p_2}\gamma \quad = \quad e_{2|p_2}\tau_1\gamma \quad = \quad e_{2|p_2}\upsilon_2\sigma \quad = \quad l_2\upsilon_2\sigma \quad = \quad l_2\tau_1\gamma \quad = \quad l_2\gamma.$$

(In the first equality we used the assumption $p_2 \in \mathcal{P}os_{\mathcal{F}}(e_2)$.) Hence we obtain $\tau_2 \leqslant \gamma$ from the fact that τ_2 is a most general unifier of $e_2\tau_{1|p_2}$ and l_2. Therefore $\tau_1\tau_2 \leqslant \tau_1\gamma = \upsilon_2\sigma$. Since we also have $\upsilon_2\sigma \leqslant \tau_1\tau_2$, there is a variable renaming δ such that $\upsilon_2\sigma\delta = \tau_1\tau_2$. Now define $\upsilon_1 = \sigma\delta$. Since most general unifiers are closed under variable renaming, υ_1 is a most general unifier of $e_1\upsilon_{2|p_1}$ and l_1. So $H_3 = (G'\upsilon_2, e_1\upsilon_2[r_1]_{p_1}, (G'', e_2[r_2]_{p_2}, G''')\upsilon_2)\upsilon_1$. From $\tau_1\tau_2 = \upsilon_2\upsilon_1$ we infer that $G_3 = H_3$. This proves the lemma. $\square$

Lemma 31. *Let G_1 be the goal e, G'. For every normal* NC*-refutation*

$$G \quad \rightsquigarrow^*_{\theta_1} \quad G_1 \quad \rightsquigarrow_{\tau_1, p_1, l_1 \to r_1} \quad G_2 \quad \rightsquigarrow_{\tau_2, p_2, l_2 \to r_2} \quad G_3 \quad \rightsquigarrow^*_{\theta_2} \quad \top$$

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

$$G \quad \rightsquigarrow^*_{\theta_1} \quad G_1 \quad \rightsquigarrow_{\upsilon_2, p_2, l_2 \to r_2} \quad H_2 \quad \rightsquigarrow_{\upsilon_1, p_1, l_1 \to r_1} \quad H_3 \quad \rightsquigarrow^*_{\theta_2} \quad \top$$

with the same complexity such that $G_3 = H_3$ and $\theta_1\tau_1\tau_2\theta_2 = \theta_1\upsilon_2\upsilon_1\theta_2$.

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 $e\tau_{1|p_2}$ is a subterm of $x\tau_1$. Hence $e\tau_{1|p_2}\tau_2$ is a subterm of $x\tau_1\tau_2$. Because $p_1 \perp p_2$ we have $e\tau_{1|p_2}\tau_2 = e[r_1]_{p_1}\tau_{1|p_2}\tau_2 = l_2\tau_2$. So $x\tau_1\tau_2$ is not a normal form. Hence $x\tau_1\tau_2\theta_2$ is also not a normal form. There exists a reduction sequence from $e'\theta_1$ to e consisting of non-root reduction steps. Here e' is the leftmost equation in G. Hence $x \in \mathcal{V}ar(e_{|1}) \subseteq \mathcal{V}ar(e'\theta_{1|1})$. From the normality of Π we infer that $\theta_1\tau_1\tau_2\theta_2\restriction_{\mathcal{V}ar(e'_{|1})}$ is normalized. This yields a contradiction with Lemma 2. Therefore $p_2 \in \mathcal{P}os_{\mathcal{F}}(e)$.

The remainder of the proof is similar to the proof of Lemma 26. We have $G_3 = (e[r_1]_{p_1}\tau_1[r_2]_{p_2}, G'\tau_1)\tau_2$. Since the variables in l_2 are fresh, we have $\mathcal{D}(\tau_1) \cap \mathcal{V}ar(l_2) = \varnothing$. Hence

$$e_{|p_2}\tau_1\tau_2 \quad = \quad e\tau_{1|p_2}\tau_2 \quad = \quad l_2\tau_2 \quad = \quad l_2\tau_1\tau_2.$$

So $\tau_1\tau_2$ is a unifier of $e_{|p_2}$ and l_2. Hence there exists an idempotent most general unifier υ_2 of $e_{|p_2}$ and l_2 such that $\upsilon_2 \leqslant \tau_1\tau_2$. Let ρ be a substitution satisfying $\upsilon_2\rho = \tau_1\tau_2$. We obtain $H_2 = (e[r_2]_{p_2}, G')\upsilon_2$. Since υ_2 is idempotent, $\mathcal{D}(\upsilon_2) \cap \mathcal{V}ar(l_1) = \varnothing$. Hence

$$e\upsilon_{2|p_1}\rho \quad = \quad e_{|p_1}\upsilon_2\rho \quad = \quad e_{|p_1}\tau_1\tau_2 \quad = \quad l_1\tau_1\tau_2 \quad = \quad l_1\upsilon_2\rho \quad = \quad l_1\rho,$$

i.e., ρ is a unifier of $e\upsilon_{2|p_1}$ and l_1. Let σ be an idempotent most general unifier of these two terms. We have $\sigma \leqslant \rho$ and thus $\upsilon_2\sigma \leqslant \upsilon_2\rho = \tau_1\tau_2$. Using $\mathcal{D}(\upsilon_2) \cap \mathcal{V}ar(l_1) = \varnothing$ we obtain

$$e_{|p_1}\upsilon_2\sigma \quad = \quad e\upsilon_{2|p_1}\sigma \quad = \quad l_1\sigma \quad = \quad l_1\upsilon_2\sigma.$$

Since τ_1 is a most general of $e_{|p_1}$ and l_1, we have $\tau_1 \leqslant \upsilon_2\sigma$, so there exists a substitution γ such that $\tau_1\gamma = \upsilon_2\sigma$. Using $\mathcal{D}(\tau_1) \cap \mathcal{V}ar(l_2) = \varnothing$ we obtain

$$e\tau_{1|p_2}\gamma \quad = \quad e_{|p_2}\tau_1\gamma \quad = \quad e_{|p_2}\upsilon_2\sigma \quad = \quad l_2\upsilon_2\sigma \quad = \quad l_2\tau_1\gamma \quad = \quad l_2\gamma.$$

Because τ_2 is a most general unifier of $e\tau_{1|p_2}$ and l_2, we must have $\tau_2 \leqslant \gamma$ and hence $\tau_1\tau_2 \leqslant \tau_1\gamma = \upsilon_2\sigma$. So $\tau_1\tau_2$ and $\upsilon_2\sigma$ are variants. Hence there exists a variable renaming δ such that $\upsilon_2\sigma\delta = \tau_1\tau_2$. Now define $\upsilon_1 = \sigma\delta$. Since σ is a most general unifier of $e\upsilon_{2|p_1}$ and l_1, and most general unifiers are closed under variable renaming, we infer that also υ_1 is a most general unifier of these two terms. So $H_3 = (e[r_2]_{p_2}\upsilon_2[r_1]_{p_1}, G'\upsilon_2)\upsilon_1$. We clearly have $\tau_1\tau_2 = \upsilon_2\upsilon_1$. With help of $p_1 \perp p_2$ we infer that $G_3 = H_3$. Because the number of narrowing steps at non-root positions is the same in the two NC-refutations, it follows that they have the same complexity. It is also easy to see that normality is preserved. □

Lemma 37. *Let G_1 be a goal containing distinct equations e_1 and e_2. For every* NC*-derivation*

$$G \rightsquigarrow^*_{\theta_1} G_1 \rightsquigarrow_{\tau_1, e_1, p_1, l_1 \to r_1} G_2 \rightsquigarrow_{\tau_2, e_2\tau_1, p_2, l_2 \to r_2} G_3 \rightsquigarrow^*_{\theta_2} \top$$

with $\theta_1\tau_1\tau_2\theta_2\restriction_{\mathcal{V}\mathrm{ar}(G)}$ normalized there exists a NC*-refutation*

$$G \rightsquigarrow^*_{\theta_1} G_1 \rightsquigarrow_{\upsilon_2, e_2, p_2, l_2 \to r_2} H_2 \rightsquigarrow_{\upsilon_1, e_1\upsilon_2, p_1, l_1 \to r_1} H_3 \rightsquigarrow^*_{\theta_2} \top$$

with the same complexity such that $G_3 = H_3$ and $\theta_1\tau_1\tau_2\theta_2 = \theta_1\upsilon_2\upsilon_1\theta_2$.

Proof. We show that $p_2 \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e_2)$. Suppose to the contrary that $p_2 \notin \mathcal{P}\mathrm{os}_{\mathcal{F}}(e_2)$. That means that $p_2 \geqslant q$ for some $q \in \mathcal{P}\mathrm{os}_{\mathcal{V}}(e_2)$. Without loss of generality we assume that $q \geqslant 1$. Let $e_{2|q}$ be the variable x. The term $e_2\tau_{1|p_2}$ is a subterm of $x\tau_1$. Hence $e_2\tau_{1|p_2}\tau_2$ is a subterm of $x\tau_1\tau_2$. Since $e_2\tau_{1|p_2}\tau_2 = l_2\tau_2$, we conclude that $x\tau_1\tau_2$ is not a normal form. Hence $x\tau_1\tau_2\theta_2$ is also not a normal form. There exists a reduction sequence from $e\theta_1$ to e_2 for some equation $e \in G$ consisting of non-root reduction steps. Hence $x \in \mathcal{V}\mathrm{ar}(e_{2|1}) \subseteq \mathcal{V}\mathrm{ar}(e\theta_{1|1})$. Now the normalization of $\theta_1\tau_1\tau_2\theta_2\restriction_{\mathcal{V}\mathrm{ar}(e_{2|1})}$ yields a contradiction with Lemma 2. Hence we have $p_2 \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e_2)$. Now we apply Lemma 26 to the subderivation $G_1 \rightsquigarrow G_2 \rightsquigarrow G_3$, resulting in a refutation of the desired shape with $\theta_1\tau_1\tau_2\theta_2 = \theta_1\upsilon_2\upsilon_1\theta_2$. We already observed that the transformation of Lemma 26 doesn't affect the complexity. □