Strong Completeness of a Lazy Conditional Narrowing Calculus

Mohamed Hamada
Doctoral Program in Engineering, University of Tsukuba
Tsukuba 305, Japan
E-mail: hamada@score.is.tsukuba.ac.jp

and

Aart Middeldorp
Institute of Information Sciences and Electronics, University of Tsukuba
Tsukuba 305, Japan
E-mail: ami@score.is.tsukuba.ac.jp

ABSTRACT

In this paper we extend the lazy narrowing calculus of Middeldorp, Okui, and Ida [13] to conditional rewrite systems. We show that our calculus is strongly complete whenever basic conditional narrowing is complete. This generalizes one of the three main results of [13] to the conditional case. As a corollary we obtain the strong completeness for three classes of conditional rewrite systems, one of which is the class of decreasing and confluent systems. For this latter class we present a simple direct proof.

1. Introduction

Narrowing (Hullot [8]) is the underlying computational mechanism of many programming languages that integrate the functional and logic programming paradigms (Hanus [5]). The desirable property of narrowing is completeness: for every solution to a given goal a more general solution can be found by narrowing. In this paper we are concerned with calculi in which narrowing is replaced by more elementary inference rules. Completeness issues for the lazy narrowing calculus LNC—which is based on the calculus TRANS of Hölldobler [7]—have been extensively studied in Middeldorp, Okui, and Ida [13] and Middeldorp and Okui [12]. In this paper we extend one of the main results of [13] to conditional rewriting: we prove that LCNC—the conditional variant of LNC—is strongly complete whenever basic conditional narrowing is complete. Strong completeness means that the choice of the equation in goals can be made don't care non-deterministic, resulting in a huge reduction of the search space as well as easing implementations. In future work we plan to address the extension of the other main results of [13]—completeness of LNC for arbitrary confluent systems and completeness of a restricted variant of the eager variable elimination strategy for orthogonal systems—as well as the work of [12]—sufficient conditions for the completeness of a deterministic version of LNC—to conditional systems.

We assume familiarity with the basics of (conditional) term rewriting and narrowing. Surveys can be found in [1, 10, 11]. Below we recall some definitions pertaining to conditional rewriting and narrowing.

A conditional term rewriting system (CTRS for short) over a signature $\mathcal{F}$ is a set $\mathcal{R}$ of (conditional) rewrite rules of the form $l \rightarrow r \Leftarrow c$ where the conditional part c is a (possibly empty) sequence $s_1 \approx t_1, \ldots, s_n \approx t_n$ of equations. All terms $l, r, s_1, \ldots, s_n, t_1, \ldots, t_n$ must belong to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ and we require that l is not a variable. Following [11], CTRSs are classified according to the distribution of variables in rewrite rules. A 1-CTRS contains no extra variables (i.e., $\mathcal{V}\mathrm{ar}(r, c) \subseteq \mathcal{V}\mathrm{ar}(l)$ for all rewrite rules $l \rightarrow r \Leftarrow c$), a 2-CTRS may contain extra variables in the conditions only ($\mathcal{V}\mathrm{ar}(r) \subseteq \mathcal{V}\mathrm{ar}(l)$ for all rewrite rules $l \rightarrow r \Leftarrow c$), and a 3-CTRS may also have extra variables in the right-hand sides provided these occur in the corresponding conditions ($\mathcal{V}\mathrm{ar}(r) \subseteq \mathcal{V}\mathrm{ar}(l, c)$ for all rewrite rules $l \rightarrow r \Leftarrow c$). We assume that every CTRS contains the rewrite rule $x \approx x \rightarrow \mathtt{true}$. Here $\approx$ and $\mathtt{true}$ are function symbols that do not occur in the other rewrite rules. These symbols may only occur at the root position of terms. Let $\mathcal{R}$ be a CTRS. We inductively define unconditional TRSs $\mathcal{R}_n$ for $n \geqslant 0$ as follows:

$$
\begin{array}{lcl}
\mathcal{R}_0 & = & \{\, x \approx x \rightarrow \mathtt{true} \,\}, \\
\mathcal{R}_{n+1} & = & \{\, l\theta \rightarrow r\theta \mid l \rightarrow r \Leftarrow c \in \mathcal{R} \text{ and } c\theta \rightarrow^*_{\mathcal{R}_n} \top \,\}.
\end{array}
$$

Here $\top$ stands for any sequence of $\mathtt{true}$'s. We define $s \rightarrow_{\mathcal{R}} t$ if and only if there exists an $n \geqslant 0$ such that $s \rightarrow_{\mathcal{R}_n} t$. In the following we abbreviate $\rightarrow_{\mathcal{R}_n}$ to $\rightarrow_n$. Our CTRS are known as join CTRSs in the term rewriting literature.

A CTRS $\mathcal{R}$ is level-confluent (Giovannetti and Moiso [4]) if every $\mathcal{R}_n$ is confluent, i.e., ${}_n^*\!\leftarrow \cdot \rightarrow_n^* \subseteq \rightarrow_n^* \cdot {}_n^*\!\leftarrow$ for all $n \geqslant 0$, and shallow-confluent if ${}_m^*\!\leftarrow \cdot \rightarrow_n^* \subseteq \rightarrow_n^* \cdot {}_m^*\!\leftarrow$ for all $m, n \geqslant 0$. Shallow-confluent CTRSs are level-confluent but the reverse is not true. A CTRS $\mathcal{R}$ over a signature $\mathcal{F}$ is decreasing (Dershowitz *et al.* [3]) if there exists a well-founded order $\succ$ on $\mathcal{T}(\mathcal{F}, \mathcal{V})$ with the following three properties: $\succ$ contains $\rightarrow_{\mathcal{R}}$, $\succ$ has the subterm property, and $l\theta \succ s\theta, t\theta$ for every rewrite rule $l \rightarrow r \Leftarrow c$ of $\mathcal{R}$, every equation $s \approx t$ in c, and every substitution θ. Note that according to this definition 2-CTRSs and 3-CTRSs are never decreasing. Decreasing CTRSs are terminating and, when there are finitely many rewrite rules, have a decidable rewrite relation.

An equation is a term of the form $s \approx t$. The constant $\mathtt{true}$ is also viewed as an equation. A goal is a sequence of equations. A substitution θ is a ($\mathcal{R}$-)solution of a goal G if $G\theta \rightarrow^*_{\mathcal{R}} \top$. For confluent $\mathcal{R}$ this is equivalent to validity of the equations in $G\theta$ in all models of the underlying conditional equational system of $\mathcal{R}$ (Kaplan [9]). The conditional narrowing calculus CNC consists of the following inference rule:

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

if there exist a fresh variant $l \rightarrow r \Leftarrow c$ of a rewrite rule in $\mathcal{R}_+$, a non-variable position p in e, and a most general unifier θ of $e_{|p}$ and l.

In the above situation we write $G', e, G'' \rightsquigarrow_\theta (G', e[r]_p, c, G'')\theta$. This is called a CNC-step. A sequence $G_1 \rightsquigarrow_{\theta_1} \cdots \rightsquigarrow_{\theta_{n-1}} G_n$ of CNC-steps is called a CNC-derivation and abbreviated to $G_1 \rightsquigarrow^*_\theta G_n$ where $\theta = \theta_1 \cdots \theta_{n-1}$. We use the symbol Π (and its derivatives) to denote CNC-derivations. For a CNC-derivation $\Pi\colon G \rightsquigarrow^*_\theta G'$, $\Pi\theta$ denotes the corresponding rewrite sequence $G\theta \to^* G'$. A CNC-derivation which ends in $\top$ is called a CNC-refutation. If $G \rightsquigarrow^*_\theta \top$ then θ is a solution of G. This is known as the soundness of conditional narrowing.

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 $\bar{G}$. A CNC-derivation Π:

$$G_1 \quad \rightsquigarrow_{\theta_1, e_1, p_1, l_1 \to r_1 \Leftarrow c_1} \quad \cdots \quad \rightsquigarrow_{\theta_{n-1}, e_{n-1}, p_{n-1}, l_{n-1} \to r_{n-1} \Leftarrow c_{n-1}} \quad G_n$$

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

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

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 $B_i(e_i) \setminus \{q \in B_i(e_i) \mid q \geqslant p_i\} \cup \{p_i{\cdot}q \in \mathcal{P}\mathrm{os}_\mathcal{F}(e) \mid q \in \mathcal{P}\mathrm{os}_\mathcal{F}(r_i)\}$. We say that Π is basic if it is based on $\bar{G}$. So in a basic CNC-derivation no steps take place at subterms introduced by previous narrowing substitutions. Basic CNC has a much smaller search space than CNC. We refer to Middeldorp and Hamoen [11] for a survey of completeness results for CNC and basic CNC.

For a substitution θ and a set of variables V, we denote $(V \setminus \mathcal{D}(\theta)) \cup \mathcal{I}(\theta\restriction_V)$ by $\mathcal{V}\mathrm{ar}_V(\theta)$. In the sequel we frequently make use of the equivalence of $\theta_1 = \theta_2\ [\mathcal{V}\mathrm{ar}_V(\theta)]$ and $\theta\theta_1 = \theta\theta_2\ [V]$ for all substitutions θ, θ_1, θ_2 and sets of variables V.

In the next section we introduce LCNC. In Section 3 we prove the strong completeness of LCNC for the class of decreasing and confluent CTRSs. We also show that strong completeness fails if decreasingness is relaxed to termination. In Section 4 we extend the result of Section 3 to CTRSs for which basic CNC is complete and we mention the consequences of this result.

2. Lazy Conditional Narrowing

The *lazy conditional narrowing calculus* LCNC consists of the following five inference rules:

[o] *outermost narrowing*

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

if there exists a fresh variant $f(l_1, \ldots, l_n) \to r \Leftarrow c$ 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 $x \notin \mathcal{V}\text{ar}(s)$ 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 only difference between LCNC and the calculus LNC of [13] is in the outermost narrowing rule: in LCNC we add the conditional part of the applied rewrite rule to the new goal. If G and G' are the upper and lower goal in the inference rule $[\alpha]$ ($\alpha \in \{o, i, d, v, t\}$), we write $G \Rightarrow_{[\alpha]} G'$. This is called an LCNC-step. The applied rewrite rule or substitution may be supplied as subscript, that is, we write things like $G \Rightarrow_{[o], l \to r \Leftarrow c} G'$ and $G \Rightarrow_{[i], \theta} G'$. A finite LCNC-derivation $G_1 \Rightarrow_{\theta_1} \cdots \Rightarrow_{\theta_{n-1}} G_n$ may be abbreviated to $G_1 \Rightarrow^*_\theta G_n$ where $\theta = \theta_1 \cdots \theta_{n-1}$. An LCNC-refutation is an LCNC-derivation ending in the empty goal $\square$.

3. Decreasing and Confluent CTRSs

In this section we prove the strong completeness of LCNC for the class of decreasing and confluent CTRSs. We use induction with respect to the well-founded order on rewrite sequences defined below. Strong completeness means completeness independent of selection functions. A selection function is mapping that assigns to every goal G different from $\top$ an equation $e \in G$ different from `true`. An example of a selection function is $\mathcal{S}_{\text{right}}$ which always returns the rightmost equation different from `true`. We say that a CNC-derivation Π respects a selection function $\mathcal{S}$ if the selected equation in every step $G_1 \rightsquigarrow G_2$ of Π coincides with $\mathcal{S}(G_1)$.

Definition 3.1 For every decreasing CTRS $\mathcal{R}$ over the signature $\mathcal{F}$ there exists a well-founded order $\succ$ on $\mathcal{T}(\mathcal{F}, \mathcal{V})$ such that $\rightarrow_{\mathcal{R}} \subseteq \succ$, $\rhd \subseteq \succ$, and $l\theta \succ s\theta, t\theta$ for every rewrite rule $l \rightarrow r \Leftarrow c \in \mathcal{R}$, equation $s \approx t \in c$, and substitution θ. Here $\rhd$ denotes the proper subterm relation, i.e., $s \rhd t$ if t is a proper subterm of s. Let G be a goal and θ a substitution. The multiset $\{s\theta, t\theta \mid s \approx t \in G\}$ is denoted by $(G; \theta)$. Let $\Pi\colon G \rightarrow^* H$ and $\Pi'\colon G' \rightarrow^* H'$ be rewrite sequences. We write $\Pi > \Pi'$ if $(G; \epsilon) \succ_{\mathrm{mul}} (G'; \epsilon)$ where $\succ_{\mathrm{mul}}$ denotes the multiset extension of $\succ$, i.e., $M \succ_{\mathrm{mul}} N$ for finite multisets M, N if and only 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 $x \in X$ with $x \succ y$. Here $-$ and $\uplus$ denote multiset difference and sum.

From [2] we know that $\succ_{\mathrm{mul}}$ inherits well-foundedness from $\succ$. Hence $>$ is a well-founded order on rewrite sequences. Note that this order depends only on the initial goals of rewrite sequences. In the proof below we make use of the well-known equivalence of $M \succ_{\mathrm{mul}} N$ and $M - N \succ_{\mathrm{mul}} N - M$.

Theorem 3.2 *Let $\mathcal{R}$ be a decreasing and confluent CTRS, $\mathcal{S}$ an arbitrary selection function, and G a goal. For every solution θ of G there exists an* LCNC*-refutation $G \Rightarrow^*_\sigma \square$ respecting $\mathcal{S}$ such that $\sigma \leqslant_{\mathcal{R}} \theta$ $[\mathcal{V}\mathrm{ar}(G)]$.*

Proof. We use well-founded induction with respect the order $>$ on rewrite sequences. Let $\Pi\colon G\theta \rightarrow^* \top$ be any rewrite sequence from $G\theta$ to $\top$. In order to make the induction work we prove $\sigma \leqslant_{\mathcal{R}} \theta$ $[V]$ for a finite set of variables V that includes $\mathcal{V}\mathrm{ar}(G)$. The base case is trivial since G must be the empty goal (and thus we can take $\sigma = \epsilon$). For the induction step, whose structure is illustrated in Figure 1, we proceed as follows. Write $G = G', e, G''$ where $e = s \approx t = \mathcal{S}(G)$. We will show the existence of an LCNC-step $\Psi_1\colon G \Rightarrow_{\sigma_1} G_1$ respecting $\mathcal{S}$ and a rewrite sequence $\Pi_1\colon G_1\theta_1 \rightarrow^* \top$ such that $\sigma_1\theta_1 \leqslant_{\mathcal{R}} \theta$ $[V]$ and $\Pi > \Pi_1$. We distinguish the following cases.

(1) Suppose $s, t \notin \mathcal{V}$. We distinguish two further cases, depending on what happens to $e\theta$ in Π.

  (a) Suppose no reduct of $e\theta$ is rewritten at position 1 or 2. 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 $\Psi_1\colon G \Rightarrow_{[d]} G_1$. Let $\sigma_1 = \epsilon$ and $\theta_1 = \theta$. The rewrite sequence Π, which must be of the form

$$G\theta \rightarrow^* H_1, f(u_1, \ldots, u_n) \approx f(u_1, \ldots, u_n), H_2 \rightarrow H_1, \mathtt{true}, H_2 \rightarrow^* \top,$$

  can be transformed into Π_1:

$$G_1\theta_1 \rightarrow^* H_1, u_1 \approx u_1, \ldots, u_n \approx u_n, H_2 \rightarrow^* H_1, \top, H_2 \rightarrow^* \top.$$

  Clearly $\sigma_1\theta_1 = \theta$. For all $1 \leqslant i \leqslant n$ we have $s\theta \rhd s_i\theta_1$ and $t\theta \rhd t_i\theta_1$ and thus also $s\theta \succ s_i\theta_1$ and $t\theta \succ t_i\theta_1$. Hence $(G; \theta) \succ_{\mathrm{mul}} (G_1; \theta_1)$ and therefore $\Pi > \Pi_1$.

(b) Suppose a reduct of $e\theta$ is rewritten at position 1 or 2. Without loss of generality we assume that the first such reduct is rewritten at position 1, using the fresh variant $l \to r \Leftarrow c$ of a rewrite rule in $\mathcal{R}$ with $l = f(l_1, \ldots, l_n)$. We have $s = f(s_1, \ldots, s_n)$. The rewrite sequence Π is of the form

$$G\theta \to^* H_1, f(s_1', \ldots, s_n') \approx t', H_2 \to H_1, r\tau \approx t', H_2 \to^* \top$$

with $f(s_1', \ldots, s_n') = l\tau$ and $c\tau \to^* \top$. Let $G_1 = G', s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, c, G''$. We have $\Psi_1 \colon G \Rightarrow_{[o]} G_1$. Let $\sigma_1 = \epsilon$ and define θ_1 as the disjoint union of θ and τ. Because

$$G_1\theta_1 = G'\theta, s_1\theta \approx l_1\tau, \ldots, s_n\theta \approx l_n\tau, r\tau \approx t\theta, c\tau, G''\theta$$

we obtain the rewrite sequence Π_1:

$$\begin{aligned} G_1\theta_1 \;&\to^*\; H_1, s_1' \approx l_1\tau, \ldots, s_n' \approx l_n\tau, r\tau \approx t', c\tau, H_2 \\ &\to^*\; H_1, \top, r\tau \approx t', c\tau, H_2 \to^* \top. \end{aligned}$$

Since $V \cap \mathcal{D}(\tau) = \varnothing$ we have $\sigma_1\theta_1 = \theta\ [V]$. In order to conclude that $\Pi > \Pi_1$ we have to show that

$$\begin{aligned} (G;\theta) - (G_1;\theta_1) \;&=\; \{s\theta\} \succ_{\mathrm{mul}} \{s_1\theta, \ldots, s_n\theta, l_1\tau, \ldots, l_n\tau, r\tau\} \uplus (c;\tau) \\ &=\; (G_1;\theta_1) - (G;\theta). \end{aligned}$$

For all $1 \leqslant i \leqslant n$ we have $s\theta \rhd s_i\theta \to^* s_i' = l_i\tau$ and consequently $s\theta \succ s_1\theta, \ldots s_n\theta, l_1\tau, \ldots l_n\tau$. Furthermore, $s\theta \to^* l\tau \to r\tau$ and thus $s\theta \succ r\tau$. Finally, for all $u \approx v \in c$ we have $l\tau \succ u\tau, v\tau$ by definition of decreasingness. Combined with $s\theta \to^* l\tau$ this yields $s\theta \succ u\tau, v\tau$ and therefore $\{s\theta\} \succ_{\mathrm{mul}} (c;\tau)$.

(2) Suppose $s \notin \mathcal{V}$ and $t \in \mathcal{V}$. We distinguish two further cases.

(a) Suppose no reduct of $e\theta$ is rewritten at position 1. Write $s = f(s_1, \ldots, s_n)$. Let $\sigma_1 = \{t \mapsto f(x_1, \ldots, x_n)\}$ and $G_1 = (G', x_1 \approx s_1, \ldots, x_n \approx s_n, G'')\sigma_1$. Here $x_1, \ldots, x_n$ are fresh variables. We have $\Psi_1 \colon G \Rightarrow_{[i],\sigma_1} G_1$. The rewrite sequence Π is of the form

$$G\theta \to^* H_1, f(s_1', \ldots, s_n') \approx f(s_1', \ldots, s_n'), H_2 \to H_1, \mathtt{true}, H_2 \to^* \top.$$

Define θ_1 as the disjoint union of θ and $\{x_1 \mapsto s_1', \ldots, x_n \mapsto s_n'\}$. We have

$$G_1\theta_1 = G'\sigma_1\theta_1, s_1' \approx s_1\sigma_1\theta_1, \ldots s_n' \approx s_n\sigma_1\theta_1, G'\sigma_1\theta_1.$$

Because $t\theta \to^* f(s_1', \ldots, s_n') = f(x_1, \ldots, x_n)\theta_1 = t\sigma_1\theta_1$ and $x\theta = x\sigma_1\theta_1$ for variables x different from $x_1, \ldots, x_n$, we have $x\theta \to^* x\sigma_1\theta_1$ for all variables $x \in V$. Hence $\sigma_1\theta_1 =_{\mathcal{R}} \theta\ [V]$, $G'\theta \to^* G'\sigma_1\theta_1$, $G''\theta \to^* G''\sigma_1\theta_1$, and

$s_i\sigma_1\theta_1 \;{}^*\!\!\leftarrow s_i\theta \to^* s_i'$ for all $1 \leqslant i \leqslant n$. Confluence of $\mathcal{R}$ yields $G'\sigma_1\theta_1 \to^* \top$, $G''\sigma_1\theta_1 \to^* \top$, and terms $u_1, \dots, u_n$ such that $s_i\sigma_1\theta_1 \to^* u_i \;{}^*\!\!\leftarrow s_i'$ for all $1 \leqslant i \leqslant n$. We obtain the rewrite sequence Π_1:

$$G_1\theta_1 \to^* G'\sigma_1\theta_1, u_1 \approx u_1, \dots, u_n \approx u_n, G''\sigma_1\theta_1 \to^* \top.$$

It remains to show that $\Pi > \Pi_1$. We have $(G', G''; \theta) \succ_{\mathrm{mul}}^{=} ((G', G'')\sigma_1; \theta_1)$. Here $\succ_{\mathrm{mul}}^{=}$ denotes the reflexive closure of $\succ_{\mathrm{mul}}$. For all $1 \leqslant i \leqslant n$ we have $s\theta \rhd s_i\theta \to^* s_i'$ and $s\theta \rhd s_i\theta \to^* s_i\sigma_1\theta_1$ and thus also $s\theta \succ s_i'$ and $s\theta \succ s_i\sigma_1\theta_1$. Hence

$$\{s\theta, t\theta\} \succ_{\mathrm{mul}} \{s_1', \dots, s_n', s_1\sigma_1\theta_1, \dots, s_n\sigma_1\theta_1\}$$

and therefore $(G; \theta) \succ_{\mathrm{mul}} (G_1; \theta_1)$.

(b) Suppose a reduct of $e\theta$ is rewritten at position 1. In this case we proceed as in case (1)(b).

(3) Suppose $s \in \mathcal{V}$ and $t \notin \mathcal{V}$. This case is similar to case (2).

(4) Suppose $s, t \in \mathcal{V}$. We distinguish two further cases.

(a) Suppose $s = t$. Let $G_1 = G', G''$. We have $\Psi_1 \colon G \Rightarrow_{[t]} G_1$. Let $\sigma_1 = \epsilon$ and $\theta_1 = \theta$. From Π we extract the rewrite sequence $\Pi_1 \colon G_1\theta_1 \to^* \top$. We clearly have $\sigma_1\theta_1 = \theta$ and $\Pi > \Pi_1$ as $(G_1; \theta_1) \subset (G; \theta)$.

(b) Suppose $s \neq t$. Let $\sigma_1 = \{s \mapsto t\}$ and $G_1 = (G', G'')\sigma_1$. We have $\Psi_1 \colon G \Rightarrow_{[v], \sigma_1} G_1$. Let θ_1 be the normal form of θ, i.e., $\theta_1(x) = x\theta{\downarrow}$ for all variables $x \in \mathcal{V}$. Here $x\theta{\downarrow}$ denotes the unique normal form of $x\theta$ whose existence is guaranteed by confluence and weak normalization (which follows from decreasingness) of $\mathcal{R}$. As θ is a solution of G we have $(s \approx t)\theta \to^* \mathtt{true}$ and thus $s\theta{\downarrow} = t\theta{\downarrow}$. Hence $s\theta \to^* t\theta{\downarrow} = s\sigma_1\theta_1$. Because $x\theta \to^* x\theta{\downarrow} = x\sigma_1\theta_1$ for variables x different from s, we obtain $x\theta \to^* x\sigma_1\theta_1$ for all variables $x \in V$. Therefore $\sigma_1\theta_1 =_{\mathcal{R}} \theta\ [V]$, From Π we extract the rewrite sequence $(G', G'')\theta \to^* \top$. Since $(G', G'')\theta \to^* (G', G'')\sigma_1\theta_1 = G_1\theta_1$, confluence yields a rewrite sequence $\Pi_1 \colon G_1\theta_1 \to^* \top$. We have $(G', G''; \theta) \succ_{\mathrm{mul}}^{=} (G_1; \theta_1)$ and thus $(G; \theta) = (G', G''; \theta) \uplus \{s\theta, t\theta\} \succ_{\mathrm{mul}} (G_1; \theta_1)$. We conclude that $\Pi > \Pi_1$.

Let $V_1 = \mathcal{V}\mathrm{ar}_V(\sigma_1) \cup \mathcal{V}\mathrm{ar}(G_1)$. Clearly $\mathcal{V}\mathrm{ar}(G_1) \subseteq V_1$. Hence we can apply the induction hypothesis to $\Pi_1 \colon G_1\theta_1 \to^* \top$. This yields an LCNC-refutation $\Psi' \colon G_1 \Rightarrow_{\sigma'}^* \Box$ respecting $\mathcal{S}$ such that $\sigma' \leqslant_{\mathcal{R}} \theta_1\ [V_1]$. Define $\sigma = \sigma_1\sigma'$. From $\sigma_1\theta_1 \leqslant_{\mathcal{R}} \theta\ [V]$, $\sigma' \leqslant_{\mathcal{R}} \theta_1\ [V_1]$, and $\mathcal{V}\mathrm{ar}_V(\sigma_1) \subseteq V_1$ we infer that $\sigma \leqslant_{\mathcal{R}} \theta\ [V]$. Concatenating the LCNC-step Ψ_1 and the LCNC-refutation Ψ' yields the desired LCNC-refutation Ψ. $\Box$

Hanus [6, Theorem 3] gives a similar result for terminating and ground-confluent unconditional TRSs. The structure of his proof and his well-founded ordering are basically the same as ours, but he considers *innermost* rewrite sequences to *normal form* whereas our proof works with arbitrary rewrite sequences. This may be important when extending the (strong) completeness result to possibly non-terminating (non-decreasing) systems (under additional assumptions).

$$
\begin{array}{lllll}
G & \Pi\colon & G\theta & \to^{+} & \top \\
\Downarrow_{\sigma_1} & & & & \sigma_1\theta_1 \leqslant_{\mathcal{R}} \theta \; [V] \quad \text{(case analysis)} \\
G_1 & \Pi_1\colon & G_1\theta_1 & \to^{*} & \top \\
\Downarrow^{*}_{\sigma'} & & & & \sigma' \leqslant_{\mathcal{R}} \theta_1 \; [V_1] \quad \text{(induction hypothesis)} \\
\Box & & & &
\end{array}
$$

Figure 1: The induction step in the proof of Theorem 3.2.

Prehofer [15, Theorem 6.8.11] presents a related result for higher-order CTRSs. He shows the completeness of his conditional lazy narrowing calculus CLNN (CLN in [14]) for decreasing and confluent higher-order *normal* CTRSs with respect to *oriented* goals. Normality means that the right-hand sides of the equations in the conditions of the rewrite rules are ground normal forms with respect to the TRS that is obtained by dropping all conditions. Due to the normality and orientation restrictions, the specialization of Prehofer's result to first-order CTRSs doesn't subsume Theorem 3.2. The reverse also does not hold because the completeness result in [15] allows arbitrary *simplification*, an optimization we don't consider in this paper.

Let us illustrate the above proof on a small example. Consider the CTRS

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

This CTRS is easily seen to be decreasing (take e.g. the recursive path order with precedence $f > b$ and $a > b$) and confluent. Consider the goal $G = g(f(x)) \approx x$. The substitution $\theta = \{x \mapsto g(a)\}$ is a solution of G because we have the rewrite sequence $\Pi\colon G\theta = g(f(g(a))) \approx g(a) \to g(a) \approx g(a) \to \mathtt{true}$. Figure 2 shows the LCNC-refutation Ψ constructed in the above proof. The numbers on the right refer to the various cases in the proof. Note that Ψ computes the solution $\sigma\restriction_{\mathcal{V}\mathrm{ar}(G)} = \{x \mapsto g(b)\}$ which is different from θ but equal to θ modulo (the equational theory induced by) $\mathcal{R}$.

Theorem 3.2 does not hold if we replace decreasingness by termination, as can be seen from the terminating and confluent 1-CTRS

$$
\mathcal{R} = \left\{ \begin{array}{rcccl} f(x) & \to & a & \Leftarrow & x \approx b, x \approx c \\ d & \to & b & & \\ d & \to & c & & \\ b & \to & c & \Leftarrow & f(d) \approx a \end{array} \right.
$$

of Middeldorp and Hamoen [11]. The goal $f(x) \approx a$ admits the solution $\{x \mapsto d\}$ but

$$
\begin{array}{cll}
g(f(x)) \approx x & g(f(g(a))) \approx g(a) \xrightarrow{1\cdot 1} g(a) \approx g(a) \xrightarrow{\varepsilon} \mathtt{true} & \\
\Downarrow_{[i],\,\{x \mapsto g(x_1)\}} & & (2)(\mathrm{a}) \\
f(g(x_1)) \approx x_1 & f(g(a)) \approx a \xrightarrow{1} a \approx a \xrightarrow{\varepsilon} \mathtt{true} & \\
\Downarrow_{[o],\, f(g(x_2)) \to x_2 \Leftarrow x_2 \approx b} & & (2)(\mathrm{b}) \\
g(x_1) \approx g(x_2), x_2 \approx x_1, x_2 \approx b & g(a) \approx g(a), a \approx a, a \approx b \xrightarrow{\varepsilon}{}^{*} \top, a \approx b \xrightarrow{1} \top, b \approx b & \\
\Downarrow_{[d]} & \quad \xrightarrow{\varepsilon} \top & (1)(\mathrm{a}) \\
x_1 \approx x_2, x_2 \approx x_1, x_2 \approx b & a \approx a, a \approx a, a \approx b \xrightarrow{\varepsilon}{}^{*} \top, a \approx b \xrightarrow{1} \top, b \approx b \xrightarrow{\varepsilon} \top & \\
\Downarrow_{[v],\,\{x_1 \to x_2\}} & & (4)(\mathrm{b}) \\
x_2 \approx x_2, x_2 \approx b & b \approx b, b \approx b \xrightarrow{\varepsilon} \mathtt{true}, b \approx b \xrightarrow{\varepsilon} \top & \\
\Downarrow_{[t]} & & (4)(\mathrm{a}) \\
x_2 \approx b & b \approx b \xrightarrow{\varepsilon} \mathtt{true} & \\
\Downarrow_{[i],\,\{x \mapsto b\}} & & (3) \\
\Box & &
\end{array}
$$

Figure 2: An example illustrating the proof of Theorem 3.2.

LCNC with selection function $\mathcal{S}_{\text{right}}$ produces an infinite derivation:

$$
\begin{array}{rll}
f(x) \approx a & \Rightarrow_{[o]} & x \approx x_1, a \approx a, x_1 \approx b, x_1 \approx c \\
& \Rightarrow_{[v]} & x \approx c, a \approx a, c \approx b \\
& \Rightarrow_{[o]} & x \approx c, a \approx a, c \approx c, f(d) \approx a \\
& \Rightarrow_{[o]} & x \approx c, a \approx a, c \approx c, d \approx x_2, a \approx a, x_2 \approx b, x_2 \approx c \\
& \Rightarrow_{[v]} & \cdots
\end{array}
$$

Hence LCNC is not strongly complete for terminating and confluent 1-CTRSs. Given the connection between strong completeness of LCNC and completeness of basic conditional narrowing, to be explained in the next section, this doesn't come as a surprise since in [11] the very same example is used to refute the completeness of basic conditional narrowing for terminating and confluent 1-CTRSs.

4. Strong Completeness

In this section we show that LCNC is strongly complete whenever basic conditional narrowing is complete. To this end we extend the proof of Middeldorp, Okui, and Ida [13] who obtained this result for unconditional TRSs. The structure of the proof is exactly the same as the one in [13, Section 4]. Below we state the relevant lemmata and we present complete proof details of some of the lemmata. The proofs of the other lemmata are straightforward modifications of the corresponding ones in [13]. The numbers between parentheses refer to the corresponding statements in [13].

Definition 4.1 Let $G \rightsquigarrow_{\theta, p, l \to r \Leftarrow c} G_1$ be a CNC-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 CNC-derivations in the obvious way.

Observe that in a CNC-refutation $G \rightsquigarrow^* \top$ every equation $e \in G$ has exactly one descendant `true` in $\top$, but, unlike the unconditional case, not every `true` in $\top$ descends from an equation in G.

Lemma 4.2 (12) *Let δ be a variable renaming. For every* CNC*-refutation* $\Pi\colon G \rightsquigarrow^+_\theta \top$ *there exists a* CNC*-refutation* $\phi_\delta(\Pi)\colon G\delta \rightsquigarrow^+_{\delta^{-1}\theta} \top$. ☐

In the following five lemmata Π denotes a CNC-refutation $G \rightsquigarrow^+_\theta \top$ with $G = G', s \approx t, G''$ such that $s \approx t$ 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.

The first transformation lemma depends on the applied rewrite rule. So in the proof we have to take its conditional part into account.

Lemma 4.3 (13) *Suppose narrowing is applied to a descendant of $s \approx t$ in Π at position* 1. *If $l \to r \Leftarrow c$ is the applied rewrite rule in the first such step then there exists a* CNC*-refutation* $\phi_{[o]}(\Pi)\colon G', s \approx l, r \approx t, c, G'' \rightsquigarrow^*_{\theta_1} \top$ *such that* $\theta_1 = \theta\ [V]$.

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

$$\begin{array}{lll} G & \rightsquigarrow^*_{\tau_1} & G'_1, f(u_1, \dots, u_n) \approx t', G''_1 \\ & \rightsquigarrow_{\tau_2, 1, l \to r \Leftarrow c} & (G'_1, r \approx t', c, G''_1)\tau_2 \\ & \rightsquigarrow^*_{\tau_3} & \top \end{array}$$

with $\tau_1\tau_2\tau_3 = \theta$. Write $G''_1 = C, G''_2$ such that $s \approx t \rightsquigarrow^* f(u_1, \dots, u_n) \approx t', C$ and $G'' \rightsquigarrow^* G''_2$. So C consists of all descendants of the (instantiated) equations that appear in the conditional parts of the applied rewrite rules in the derivation from $s \approx t$ to $f(u_1, \dots, u_n) \approx t'$. The first part of Π can be transformed into

$$G', s \approx l, r \approx t, c, G'' \rightsquigarrow^*_{\tau_1} G'_1, f(u_1, \dots, u_n) \approx l, C_1, r \approx t', C_2, c, G''_2$$

where $C_1, C_2 = C$. The last part of Π can be rearranged into

$$(G'_1, C_1, r \approx t', C_2, c, G''_2)\tau_2 \rightsquigarrow^*_{\tau_3} \top.$$

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 $f(u_1, \dots, u_n) \approx l$ and $x \approx x$, the CNC-derivation Π can be transformed into the refutation $\phi_{[o]}(\Pi)$:

$$\begin{array}{lll} G', s \approx l, r \approx t, c, G'' & \rightsquigarrow^*_{\tau_1} & G'_1, f(u_1, \dots, u_n) \approx l, C_1, r \approx t', C_2, c, G''_2 \\ & \rightsquigarrow_{\upsilon_2} & (G'_1, \mathtt{true}, C_1, r \approx t', C_2, c, G''_2)\upsilon_2 \\ & = & (G'_1, \mathtt{true}, C_1, r \approx t', C_2, c, G''_2)\tau_2 \\ & \rightsquigarrow^*_{\tau_3} & \top. \end{array}$$

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 $\theta_1 = \theta\ [V]$. $\square$

For the tedious proof of the next lemma we refer to the appendix.

Lemma 4.4 (14) *Let $s = f(s_1, \ldots, s_n)$ and $t \in \mathcal{V}$. If $\mathrm{root}(t\theta) = f$ then there exists a* CNC*-refutation $\phi_{[i]}(\Pi)$: $G\sigma_1 \rightsquigarrow^*_{\theta_1} \top$ 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 4.5 (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 $s \approx t$ in Π at position 1 or 2. There exists a* CNC*-refutation $\phi_{[d]}(\Pi)$: $G', s_1 \approx t_1, \ldots, s_n \approx t_n, G'' \rightsquigarrow^*_{\theta_1} \top$ such that $\theta_1 \leqslant \theta\ [V]$.*

Proof. The given refutation Π must be of the form

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

with $s' = f(s'_1, \ldots, s'_n)$, $t' = f(t'_1, \ldots, t'_n)$, and $\tau_1\tau_2\tau_3 = \theta$. Write $G''_1 = C, G''_2$ such that $s \approx t \rightsquigarrow^* \mathtt{true}, C$ (and thus $G'' \rightsquigarrow^* G''_2$). The first part of Π can be transformed into Π_1:

$$G', s_1 \approx t_1, \ldots, s_n \approx t_n, G'' \ \rightsquigarrow^*_{\tau_1} \ G'_1, s'_1 \approx t'_1, C_1, \ldots, s'_n \approx t'_n, C_n, G''_2.$$

Here $C_1, \ldots, C_n$ and C consists of the same equations in possibly different order. 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

$$(G'_1, s'_1 \approx t'_1, C_1, \ldots, s'_n \approx t'_n, C_n, G''_2)\tau_2 \ \to^*_\epsilon \ (G'_1, \mathtt{true}, C_1, \ldots, \mathtt{true}, C_n, G''_2)\tau_2.$$

Since all steps in this rewrite sequence take place at root positions using the unconditional rewrite rule $x \approx x \to \mathtt{true}$, lifting can be applied, resulting in a CNC-derivation Π_2:

$$G'_1, s'_1 \approx t'_1, C_1, \ldots, s'_n \approx t'_n, C_n, G''_2 \ \rightsquigarrow^*_{\upsilon_2, \epsilon} \ (G'_1, \mathtt{true}, C_1, \ldots, \mathtt{true}, C_n, G''_2)\upsilon_2$$

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 $\tau_3 = \varepsilon$. We simply define $\phi_{[d]}(\Pi) = \Pi_1; \Pi_2$. Note that $(G'_1, \mathtt{true}, C_1, \ldots, \mathtt{true}, C_n, G''_2)\upsilon_2 = \top$. Let $\theta_1 = \tau_1\upsilon_2$. From $\upsilon_2 \leqslant \tau_2\ [V \cup \mathcal{I}(\tau_1)]$ we infer 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 transformed (by changing the number of occurrences of `true` as well as the order of the equations in each goal) into

$$\Pi_3\colon (G_1', \texttt{true}, C_1, \ldots, \texttt{true}, C_n, G_2'')\tau_2 \rightsquigarrow^{+}_{\tau_3} \top.$$

An application of Lemma 4.2 results in the CNC-refutation

$$\phi_\delta(\Pi_3)\colon (G_1', \texttt{true}, C_1, \ldots, \texttt{true}, C_n, G_2'')\upsilon_2 \rightsquigarrow^{+}_{\delta^{-1}\tau_3} \top.$$

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 $\theta_1 = \tau_1\tau_2\tau_3 = \theta\ [V]$.

□

The proofs of the two remaining transformation lemmata are exactly the same as the unconditional ones.

Lemma 4.6 (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 a* CNC*-refutation $\phi_{[v]}(\Pi)\colon (G', G'')\sigma_1 \rightsquigarrow^{*}_{\theta_1} \top$ with $\sigma_1 = \{t \mapsto s\}$ such that $\sigma_1\theta_1 \leqslant \theta\ [V]$.* □

Lemma 4.7 (17) *Let $t \in \mathcal{V}$, $s = t$, and suppose that the first step of Π takes place at the root position. There exists a* CNC*-refutation $\phi_{[t]}(\Pi)\colon G', G'' \rightsquigarrow^{*}_{\theta_1} \top$ such that $\theta_1 \leqslant \theta\ [V]$.* □

Definition 4.8 The complexity $|\Pi|$ of a CNC-refutation $\Pi\colon G \rightsquigarrow^{*}_{\theta} \top$ is defined as the triple consisting of (1) the number of applications of narrowing in Π at non-root positions, (2) the multiset $|\mathcal{MV}\mathrm{ar}(G)\theta|$, and (3) the number of occurrences of symbols different from $\approx$ and `true` 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 multisets $\{t_1\theta, \ldots, t_n\theta\}$ and $\{|t_1|, \ldots, |t_n|\}$, respectively. The well-founded order $\gg$ on CNC-refutations is defined 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 $>$, $>_{\mathrm{mul}}$, and $>$.

The order defined above is identical to the one in [13].

Lemma 4.9 (20) *Let Π be a* CNC*-refutation and $\alpha \in \{o, i, d, v, t\}$. We have $\Pi \gg \phi_{[\alpha]}(\Pi)$ whenever $\phi_{[\alpha]}(\Pi)$ is defined.* □

Lemma 4.10 (21) *For every* CNC*-refutation $\Pi\colon G', s \approx t, G'' \rightsquigarrow^{*}_{\theta} \top$ there exists a* CNC*-refutation $\phi_{\mathsf{swap}}(\Pi)\colon G', t \approx s, G'' \rightsquigarrow^{*}_{\theta} \top$ with the same complexity.* □

Lemma 4.11 (24) *Let Π be a basic* CNC*-refutation and $\alpha \in \{o, i, d, v, t\}$. The* CNC*-refutation $\phi_{[\alpha]}(\Pi)$ is basic whenever it is defined.* □

Lemma 4.12 is the key lemma. It can be diagrammatically depicted as follows:

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

Although the proof is very similar to that of Lemma 25 in [13], we present the complete proof in order to show how the previous lemmata are used.

Lemma 4.12 (25) *For every non-empty basic* CNC*-refutation* $\Pi: G \rightsquigarrow_{\theta}^{+} \top$ *there exist an* LCNC*-step* $\Psi_1: G \Rightarrow_{\sigma_1} G_1$ *and a basic* CNC*-refutation* $\Pi_1: G_1 \rightsquigarrow_{\theta_1}^{*} \top$ *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 4.5 yields a CNC-refutation $\phi_{[d]}(\Pi): G_1 \rightsquigarrow_{\theta_1}^{*} \top$ 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 $G_1 = G', G''$. We have $\Psi_1: G \Rightarrow_{[t]} G_1$. Lemma 4.7 yields a CNC-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 $G_1 = (G', G'')\sigma_1$. We have $\Psi_1: G \Rightarrow_{[v],\sigma_1} G_1$. Lemma 4.6 yields a CNC-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 4.4 we obtain a CNC-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 4.5 to Π_2 results in a CNC-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 $\mathcal{V}\mathrm{ar}_V(\sigma_1) \subseteq V_2$ we obtain $\sigma_1\theta_1 \leqslant \sigma_1\theta_2 = \theta$ $[V]$.

  (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 4.10.

(2) Suppose narrowing is applied to a descendant of e at position 1. Let $l = f(l_1, \ldots, l_n) \to r \Leftarrow c$ 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 $G_1 = G', s_1 \approx l_1, \ldots, s_n \approx l_n, r \approx t, c, G''$ and $G_2 = G', s \approx l, r \approx t, c, G''$. We have $\Psi_1\colon G \Rightarrow_{[o]} G_1$. From Lemma 4.3 we obtain a CNC-refutation $\Pi_2 = \phi_{[o]}(\Pi)\colon G_2 \rightsquigarrow^*_{\theta_2} \top$ such that $\theta_2 = \theta\ [V]$. Let $V_2 = V \cup \mathcal{V}\mathrm{ar}(l \to r \Leftarrow c)$. Clearly $\mathcal{V}\mathrm{ar}(G_2) \subseteq V_2$. An application of Lemma 4.5 to Π_2 results in a CNC-refutation $\Pi_1 = \phi_{[d]}(\Pi_2)\colon G_1 \rightsquigarrow^*_{\theta_1} \top$ 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 4.10.

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 4.11 Π_1 is basic. According to Lemmata 4.9 and 4.10 Π_1 has smaller complexity than Π. □

The above proof is similar in structure to that of Theorem 3.2 but the case analysis is more complicated.

Lemma 4.13 (26) *Let G_1 be a goal containing distinct equations e_1 and e_2. For every* CNC*-derivation $G_1 \rightsquigarrow_{\tau_1, e_1, p_1, l_1 \to r_1 \Leftarrow c_1} G_2 \rightsquigarrow_{\tau_2, e_2\tau_1, p_2, l_2 \to r_2 \Leftarrow c_2} G_3$ with $p_2 \in \mathcal{P}\mathrm{os}_{\mathcal{F}}(e_2)$ there exists a* CNC*-derivation $G_1 \rightsquigarrow_{\upsilon_2, e_2, p_2, l_2 \to r_2 \Leftarrow c_2} H_2 \rightsquigarrow_{\upsilon_1, e_1\upsilon_2, p_1, l_1 \to r_1 \Leftarrow c_1} H_3$ such that $G_3 = H_3$ and $\tau_1\tau_2 = \upsilon_2\upsilon_1$.*

Proof. Let $G_1 = G', e_1, G'', e_2, G'''$. Since we may assume that the variables in $l_2 \to r_2 \Leftarrow c_2$ are fresh, we have $\mathcal{D}(\tau_1) \cap \mathcal{V}\mathrm{ar}(l_2 \to r_2 \Leftarrow c_2) = \varnothing$. Hence $r_2\tau_1 = r_2$, $c_2\tau_1 = c_2$, and thus

$$\begin{aligned} G_3 &= ((G', e_1[r_1]_{p_1}, c_1, G'')\tau_1, e_2\tau_1[r_2]_{p_2}, c_2, G'''\tau_1)\tau_2 \\ &= (G', e_1[r_1]_{p_1}, c_1, G'', e_2[r_2]_{p_2}, c_2, G''')\tau_1\tau_2. \end{aligned}$$

From $\mathcal{D}(\tau_1) \cap \mathcal{V}\mathrm{ar}(l_2) = \varnothing$ we also infer that $e_{2|p_2}\tau_1\tau_2 = e_2\tau_{1|p_2}\tau_2 = l_2\tau_2 = l_2\tau_1\tau_2$, so $e_{2|p_2}$ and l_2 are unifiable. Let υ_2 be an idempotent most general unifier of these two terms. By definition of most general unifier there exists a substitution ρ such that $\upsilon_2\rho = \tau_1\tau_2$. We have $H_2 = (G', e_1, G'', e_2[r_2]_{p_2}, c_2, G''')\upsilon_2$ and $\mathcal{D}(\upsilon_2) \subseteq \mathcal{V}\mathrm{ar}(e_{2|p_2}) \cup \mathcal{V}\mathrm{ar}(l_2)$ by idempotency of υ_2. Because we may assume that $\mathcal{V}\mathrm{ar}(l_1 \to r_1 \Leftarrow c_1) \cap (\mathcal{V}\mathrm{ar}(e_2) \cup \mathcal{V}\mathrm{ar}(l_2 \to r_2 \Leftarrow c_2)) = \varnothing$, we obtain $\mathcal{D}(\upsilon_2) \cap \mathcal{V}\mathrm{ar}(l_1 \to r_1 \Leftarrow c_1) = \varnothing$. Hence $e_1\upsilon_{2|p_1}\rho = e_{1|p_1}\upsilon_2\rho = e_{1|p_1}\tau_1\tau_2 = l_1\tau_1\tau_2 = l_1\upsilon_2\rho = l_1\rho$. So the terms $e_1\upsilon_{2|p_1}$ and l_1 are unifiable. Let σ be a 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 $e_{1|p_1}\upsilon_2\sigma = e_1\upsilon_{2|p_1}\sigma = l_1\sigma = l_1\upsilon_2\sigma$, 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 $e_2\tau_{1|p_2}\gamma = e_{2|p_2}\tau_1\gamma = e_{2|p_2}\upsilon_2\sigma = l_2\upsilon_2\sigma = l_2\tau_1\gamma = l_2\gamma$. 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. From $\mathcal{D}(\upsilon_2) \cap \mathcal{V}\mathrm{ar}(l_1 \to r_1 \Leftarrow c_1) = \varnothing$ we infer $r_1\upsilon_2 = r_1$, $c_1\upsilon_2 = c_1$, and thus

$$\begin{aligned} H_3 &= (G'\upsilon_2, e_1\upsilon_2[r_1]_{p_1}, c_1, (G'', e_2[r_2]_{p_2}, c_2, G''')\upsilon_2)\upsilon_1 \\ &= (G', e_1[r_1]_{p_1}, c_1, G'', e_2[r_2]_{p_2}, c_2, G''')\upsilon_2\upsilon_1. \end{aligned}$$

Since $\tau_1\tau_2 = \upsilon_2\upsilon_1$ we conclude that $G_3 = H_3$. $\square$

Lemma 4.14 (27) *Let $\mathcal{S}$ be an arbitrary selection function. For every basic* CNC*-refutation $\Pi\colon G \rightsquigarrow_\theta^* \top$ there exists a basic* CNC*-refutation $\phi_{\mathcal{S}}(\Pi)\colon G \rightsquigarrow_\theta^* \top$ respecting $\mathcal{S}$ with the same complexity.* $\square$

Theorem 4.15 (28) *Let $\mathcal{R}$ be an arbitrary CTRS and $\Pi\colon G \rightsquigarrow_\theta^* \top$ a basic* CNC*-refutation. For every selection function $\mathcal{S}$ there exists an* LCNC*-refutation $\Psi\colon G \Rightarrow_\sigma^* \square$ respecting $\mathcal{S}$ such that $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$.*

Proof. We use well-founded induction on the complexity of the given basic CNC-refutation Π. In order to make the induction work we prove $\sigma \leqslant \theta$ $[V]$ for a finite set of variables V that includes $\mathcal{V}\mathrm{ar}(G)$ instead of $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$. The base case is trivial: G must be the empty goal. For the induction step we proceed as follows. First we use Lemma 4.14 to transform Π into a basic CNC-refutation $\phi_{\mathcal{S}}(\Pi)\colon G \rightsquigarrow_\theta^+ \top$ respecting $\mathcal{S}$ with equal complexity. According to Lemma 4.12 there exist an LCNC-step $\Psi_1\colon G \Rightarrow_{\sigma_1} G_1$ respecting $\mathcal{S}$ and a basic CNC-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 $V_1 = \mathcal{V}\mathrm{ar}_V(\sigma_1) \cup \mathcal{V}\mathrm{ar}(G_1)$. Clearly V_1 is a finite set of variables that includes $\mathcal{V}\mathrm{ar}(G_1)$. The induction hypothesis yields an LCNC-refutation $\Psi'\colon G_1 \Rightarrow_{\sigma'}^* \square$ 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 $\mathcal{V}\mathrm{ar}_V(\sigma_1) \subseteq V_1$, we infer that $\sigma \leqslant \theta$ $[V]$ and thus also $\sigma \leqslant \theta$ $[\mathcal{V}\mathrm{ar}(G)]$. Concatenating the LCNC-step Ψ_1 and the LCNC-refutation Ψ' yields the desired LCNC-refutation Ψ. $\square$

Since basic conditional narrowing is known to be complete for decreasing and confluent CTRSs (Middeldorp and Hamoen [11]), terminating and level-confluent 2-CTRSs (Giovannetti and Moiso [4]), and terminating and shallow-confluent 3-CTRSs (Werner [16]), we obtain the strong completeness of LCNC for these three classes of CTRSs. We would like to emphasize that the direct proof of strong completeness for decreasing and confluent CTRSs presented in the previous section is much simpler than the one in this section because in Theorem 3.2 rewrite sequences rather than narrowing derivations are transformed.

The following variant of a well-known example due to Giovannetti and Moiso [4] shows that LCNC is not (strongly) complete for terminating and confluent 2-CTRSs, even under the additional normality restriction:

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

Confluence and termination of $\mathcal{R}$ are easily shown. The goal $b \approx c$ admits the empty substitution as solution but one easily shows that LCNC cannot solve the goal $b \approx c$. This first-order example refutes the completeness of Prehofer's conditional lazy narrowing calculus CLN for weakly normalizing and confluent higher-order normal CTRSs [14, Theorem 3.2].[a]

Acknowledgements

The authors are grateful to an anonymous referee for pointing out a mistake in case (4)(b) of the (previous) proof of Theorem 3.2 as well as relevant literature. The second author is partially supported by the Advanced Information Technology Program (AITP) of the Information Technology Promotion Agency (IPA).

References

[1] N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 243–320. North-Holland, 1990.

[2] N. Dershowitz and Z. Manna. Proving termination with multiset orderings. *Communications of the ACM*, 22:465–476, 1979.

[3] N. Dershowitz, M. Okada, and G. Sivakumar. Canonical conditional rewrite systems. In *Proceedings of the 9th International Conference on Automated Deduction, Argonne*, volume 310 of *Lecture Notes in Computer Science*, pages 538–549, 1988.

[4] E. Giovannetti and C. Moiso. A completeness result for e-unification algorithms based on conditional narrowing. In *Proceedings of the Workshop on Foundations of Logic and Functional Programming, Trento*, volume 306 of *Lecture Notes in Computer Science*, pages 157–167, 1986.

[5] M. Hanus. The integration of functions into logic programming: From theory to practice. *Journal of Logic Programming*, 19 & 20:583–628, 1994.

[a] Cf. also [15, Theorem 6.8.9].

[6] M. Hanus. Lazy unification with simplification. In *Proceedings of the 5th European Symposium on Programming*, volume 788 of *Lecture Notes in Computer Science*, pages 272–286, 1994.

[7] S. Hölldobler. *Foundations of Equational Logic Programming*, volume 353 of *Lecture Notes in Artificial Intelligence*. 1989.

[8] J.-M. Hullot. Canonical forms and unification. In *Proceedings of the 5th Conference on Automated Deduction*, volume 87 of *Lecture Notes in Computer Science*, pages 318–334, 1980.

[9] S. Kaplan. Conditional rewrite rules. *Theoretical Computer Science*, 33:175–193, 1984.

[10] J.W. Klop. Term rewriting systems. In S. Abramsky, D. Gabbay, and T. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 2, pages 1–116. Oxford University Press, 1992.

[11] A. Middeldorp and E. Hamoen. Completeness results for basic narrowing. *Applicable Algebra in Engineering, Communication and Computing*, 5:213–253, 1994.

[12] A. Middeldorp and S. Okui. A deterministic lazy narrowing calculus. In *Proceedings of the Fuji International Workshop on Functional and Logic Programming, Susuno*, pages 104–118. World Scientific, Singapore, 1995.

[13] A. Middeldorp, S. Okui, and T. Ida. Lazy narrowing: Strong completeness and eager variable elimination. *Theoretical Computer Science*, 167(1,2):95–130, 1996.

[14] C. Prehofer. A call-by-need strategy for higher-order functional-logic programming. In *Proceedings of the International Symposium on Logic Programming*, pages 147–161. MIT Press, 1995.

[15] C. Prehofer. *Solving Higher-Order Equations: From Logic to Programming*. PhD thesis, Technische Universität München, 1995. Appeared as Technical Report 19508.

[16] Andreas Werner. Personal communication, January 1995.

Appendix

In this appendix we prove Lemma 4.4. Most of the work is done in the following preliminary lemma.

Lemma A.1 *Let* $\Pi\colon G \leadsto_\theta^* \top$ *be a* CNC*-refutation,* V *a set of variables, and* γ *a substitution such that* $\mathcal{V}\mathrm{ar}(G) \subseteq V$, $\gamma \leqslant \theta$ $[V]$, *and the variables in* $\mathcal{D}(\gamma) \cup \mathcal{I}(\gamma)$ *are different from the variables in the employed rewrite rules. There exists a* CNC*-refutation* $\Pi'\colon G\gamma \leadsto_{\theta'}^* \top$ *such that* Π *subsumes* Π', $\Pi\theta = \Pi'\theta'$, *and* $\gamma\theta' = \theta$ $[V]$.

Proof. The proof is by induction on the length n of Π. The case $n = 0$ is obvious. Suppose $n > 0$. Let the first step of Π be

$$G = (G', e, G'') \quad \leadsto_{\sigma_1, p_1, l_1 \to r_1 \Leftarrow c_1, e} \quad (G', e[r_1]_{p_1}, c_1, G'')\sigma_1 = G_1$$

and let $\Pi_1\colon G_1 \leadsto_{\theta_1}^* \top$ be the remainder of Π, so $\sigma_1\theta_1 = \theta$. Let W be the set of variables in the rewrite rules employed in Π. Without loss of generality we assume that $\mathcal{V}\mathrm{ar}(G) \cap W = \varnothing$ and $W_1 \cap (\mathcal{D}(\sigma_1) \cup \mathcal{I}(\sigma_1)) = \varnothing$. Here W_1 is defined as the set $W \setminus \mathcal{V}\mathrm{ar}(l_1 \to r_1 \Leftarrow c_1)$. These two assumptions simply state that the variables in the rewrite rules are sufficiently fresh.

First we show that we can mimic the first step of Π starting from the goal $G\gamma$. From $\gamma \leqslant \theta$ $[V]$ we obtain a substitution δ such that $\gamma\delta = \theta$ $[V]$. Without loss of generality we may assume that $\mathcal{D}(\delta) \subseteq \mathcal{V}\mathrm{ar}_V(\gamma)$. Hence the substitution δ' defined by

$$\delta'(x) = \begin{cases} x\delta & \text{if } x \in \mathcal{V}\mathrm{ar}_V(\gamma), \\ x\theta & \text{otherwise} \end{cases}$$

satisfies $\gamma\delta' = \theta$ $[V \cup W]$ because $(\mathcal{D}(\gamma) \cup \mathcal{I}(\gamma)) \cap W = \varnothing$. Since $\mathcal{V}\mathrm{ar}(e) \subseteq \mathcal{V}\mathrm{ar}(G) \subseteq V$ and $\mathcal{D}(\gamma) \cap W = \varnothing$ we have $e\gamma_{|p_1}\delta' = e_{|p_1}\gamma\delta' = e_{|p_1}\theta = e_{|p_1}\sigma_1\theta_1 = l_1\sigma_1\theta_1 = l_1\theta = l_1\gamma\delta' = l_1\delta'$, so $e\gamma_{|p_1}$ and l_1 are unifiable. Hence we obtain the CNC-step

$$G\gamma \quad \leadsto_{p_1, \sigma_1', l_1 \to r_1 \Leftarrow c_1, e\gamma} \quad (G'\gamma, e\gamma[r_1]_{p_1}, c_1, G''\gamma)\sigma_1'$$

where σ_1' is any idempotent most general unifier of $e\gamma_{|p_1}$ and l_1.

Next we show that $(G'\gamma, e\gamma[r_1]_{p_1}, c_1, G''\gamma)\sigma_1' = G_1\gamma_1$ for a substitution γ_1 satisfying $\gamma_1 \leqslant \theta_1$ $[V_1]$ for a set of variables V_1 that includes $\mathcal{V}\mathrm{ar}(G_1)$. We have $e_{|p_1}\gamma\sigma_1' = e\gamma_{|p_1}\sigma_1' = l_1\sigma_1' = l_1\gamma\sigma_1'$, so $\gamma\sigma_1'$ is a unifier of $e_{|p_1}$ and l_1. Since σ_1 is a most general unifier of $e_{|p_1}$ and l_1 there exists a substitution γ' such that $\sigma_1\gamma' = \gamma\sigma_1'$. Let $V_1 = \mathcal{V}\mathrm{ar}_{V \cup W}(\sigma_1)$ and $\gamma_1 = \gamma'\restriction_{V_1}$. It is easy to show that $\mathcal{V}\mathrm{ar}(G_1) \subseteq V_1$. We have $\sigma_1\gamma_1 = \gamma\sigma_1'$ $[V \cup W]$ and because $\mathcal{D}(\gamma) \cap W = \varnothing$ we obtain $(G'\gamma, e\gamma[r_1]_{p_1}, c_1, G''\gamma)\sigma_1' = (G'\gamma, e\gamma[r_1\gamma]_{p_1}, c_1\gamma, G''\gamma)\sigma_1' = G_1\gamma_1$. Because δ' is a unifier of $e\gamma_{|p_1}$ and l_1, there exists a substitution δ'' such that $\sigma_1'\delta'' = \delta'$. Using $\sigma_1\gamma_1 = \gamma\sigma_1'$ $[V \cup W]$ we obtain $\sigma_1\gamma_1\delta'' = \gamma\sigma_1'\delta'' = \gamma\delta' = \sigma_1\theta_1$ $[V \cup W]$ and thus $\gamma_1 \leqslant \theta_1$ $[V_1]$.

We still have to show that $(\mathcal{D}(\gamma_1) \cup \mathcal{I}(\gamma_1)) \cap W_1 = \varnothing$ before we can apply the induction hypothesis to Π_1. Because σ_1' is an idempotent most general unifier of $e\gamma_{|p_1}$ and l_1, we have $\mathcal{D}(\sigma_1') \cup \mathcal{I}(\sigma_1') = \mathcal{V}\mathrm{ar}(e\gamma_{|p_1}) \cup \mathcal{V}\mathrm{ar}(l_1)$. Clearly $\mathcal{V}\mathrm{ar}(e\gamma_{|p_1}) \subseteq \mathcal{V}\mathrm{ar}(G) \cup \mathcal{I}(\gamma)$ and thus $\mathcal{V}\mathrm{ar}(e\gamma_{|p_1}) \cap W = \varnothing$ by assumption. Also $\mathcal{V}\mathrm{ar}(l_1) \subseteq W \setminus W_1$. Hence $(\mathcal{D}(\sigma_1') \cup \mathcal{I}(\sigma_1')) \cap W_1 = \varnothing$. Together with $\mathcal{D}(\sigma_1) \cap W_1 = \varnothing$, $\mathcal{D}(\gamma) \cap W = \varnothing$, and $\sigma_1\gamma_1 = \gamma\sigma_1'$ $[V \cup W]$ this implies $\gamma_1 = \varepsilon$ $[W_1]$, i.e., $\mathcal{D}(\gamma_1) \cap W_1 = \varnothing$. It remains to show that $\mathcal{I}(\gamma_1) \cap W_1 = \varnothing$. First observe that from $\mathcal{I}(\gamma\sigma_1') \subseteq \mathcal{I}(\gamma) \cup \mathcal{I}(\sigma_1')$, $\mathcal{I}(\gamma) \cap W = \varnothing$, and $\mathcal{I}(\sigma_1') \cap W_1 = \varnothing$ it follows that $\mathcal{I}(\gamma\sigma_1') \cap W_1 = \varnothing$. Suppose to the contrary that $x \in \mathcal{I}(\gamma_1) \cap W_1$ for some variable x. So there exists a variable $y \in \mathcal{D}(\gamma_1)$ such that $x \in \mathcal{V}\mathrm{ar}(y\gamma_1)$. We have $\mathcal{D}(\gamma_1) \subseteq ((V \cup W) \setminus \mathcal{D}(\sigma_1)) \cup \mathcal{I}(\sigma_1\restriction_{V \cup W})$. If $y \in (V \cup W) \setminus \mathcal{D}(\sigma_1)$ then $y\gamma_1 = y\sigma_1\gamma_1 = y\gamma\sigma_1'$, so $x \in \mathcal{I}(\gamma\sigma_1')$. This contradicts $\mathcal{I}(\gamma\sigma_1') \cap W_1 = \varnothing$. If $y \in \mathcal{I}(\sigma_1\restriction_{V \cup W})$ then there exists a variable $z \in V \cup W$ with $y \in \mathcal{V}\mathrm{ar}(z\sigma_1)$ and thus $x \in \mathcal{V}\mathrm{ar}(z\sigma_1\gamma_1) = \mathcal{V}\mathrm{ar}(z\gamma\sigma_1')$, again contradicting $\mathcal{I}(\gamma\sigma_1') \cap W_1 = \varnothing$.

Now we are in a position to apply the induction hypothesis to Π_1. This yields a CNC-refutation $G_1\gamma_1 \rightsquigarrow^*_{\theta_1'} \top$ such that $\gamma_1\theta_1' = \theta_1\ [V_1]$. Concatenating this CNC-refutation with the CNC-step $G\gamma \rightsquigarrow_{\sigma_1'} G_1\gamma_1$ yields the CNC-refutation $\Pi'\colon G\gamma \rightsquigarrow^*_{\theta'} \top$. Here $\theta' = \sigma_1'\theta_1'$. From $\gamma_1\theta_1' = \theta_1\ [V_1]$ we infer $\sigma_1\gamma_1\theta_1' = \sigma_1\theta_1 = \theta\ [V \cup W]$ and thus $\gamma\theta' = \gamma\sigma_1'\theta_1' = \sigma_1\gamma_1\theta_1' = \theta\ [V \cup W]$. Hence also $\gamma\theta' = \theta\ [V]$. From the construction of Π' it follows that Π subsumes Π' and $\Pi\theta = \Pi'\theta'$. $\square$

Proof of Lemma 4.4. Let $t\theta = f(t_1, \ldots, t_n)$ and define the substitution σ' as follows:

$$\sigma'(x) = \begin{cases} t_i & \text{if } x = x_i \text{ for some } 1 \leqslant i \leqslant n, \\ x\theta & \text{otherwise.} \end{cases}$$

We clearly have $\sigma_1\sigma' = \theta\ [V]$ and thus $\sigma_1 \leqslant \theta\ [V]$. An application of Lemma A.1 yields the desired result. $\square$