On Proving Soundness of the Computationally Equivalent Transformation for Normal Conditional Term Rewriting Systems by Using Unravelings

Naoki Nishida¹, Makishi Yanagisawa¹, and Karl Gmeiner²

¹ Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, 4648603 Nagoya, Japan
nishida@is.nagoya-u.ac.jp, makishi@trs.cm.is.nagoya-u.ac.jp

² Institute of Computer Science, UAS Technikum Wien
gmeiner@technikum-wien.at

Abstract

In this paper, we show that the SR transformation, a computationally equivalent transformation proposed by Şerbănuţă and Roşu, is sound for weakly left-linear normal conditional term rewriting systems (CTRS). Here, soundness for a CTRS means that reduction of the transformed unconditional term rewriting system (TRS) creates no undesired reduction for the CTRS. We first show that every reduction sequence of the transformed TRS starting with a term corresponding to the one considered on the CTRS is simulated by the reduction of the TRS obtained by the simultaneous unraveling. Then, we use the fact that the unraveling is sound for weakly left-linear normal CTRSs.

1998 ACM Subject Classification: F.4.2 Grammars and Other Rewriting Systems

Keywords and phrases: conditional term rewriting, unraveling, condition elimination

Digital Object Identifier: 10.4230/OASIcs.WPTE.2014.39

1 Introduction

Conditional term rewriting is known to be much more complicated than unconditional term rewriting in the sense of analyzing properties, e.g., operational termination [14], confluence [23], reachability [5]. A popular approach to the analysis of conditional term rewriting systems (CTRS) is to transform a CTRS into an unconditional term rewriting system (TRS) that is an overapproximation of the CTRS in terms of reduction. This approach enables us to use techniques for the analysis of TRSs, which are well investigated in the literature. For example, if the transformed TRS is terminating, then the CTRS is operationally terminating [4] — to prove termination of the transformed TRS, we can use many termination proving techniques which have been well investigated for TRSs (cf. [19]). Another interesting application of the approach is the analysis of (un)reachability on CTRSs, especially unreachable TRSs for, e.g., verifying cryptographic protocol [7]. Many techniques to construct tree automata [3] for accepting all the reachable ground terms for given (a recognizable set of) ground terms have been established (see, e.g., [12, 6, 24]), and thus, by transforming CTRSs into TRSs, we can use such techniques for TRSs to analyze (un)reachability.

* The research in this paper is partly supported by the Austrian Science Fund (FWF) international project I963 and the Japanese Society for the Promotion of Science (JSPS).
There are two approaches to transformations of CTRSs into TRSs: unravelings [15, 16] proposed by Marchiori (see, e.g., [8, 17]), and a transformation [25] proposed by Viry (see, e.g., [21, 8]).

Unravelings are transformations from a CTRS into a TRS over an extension of the original signature for the CTRS. They are complete for (reduction of) the CTRS [15], i.e., for every derivation of the CTRS, there exists a corresponding derivation of the unraveled TRS. In this respect, the unraveled TRS is an overapproximation of the CTRS w.r.t. reduction, and is useful for analyzing the properties of the CTRS, such as syntactic properties, modularity, and operational termination, since TRSs are in general much easier to handle than CTRSs.

The latest transformation based on Viry’s approach is a computationally equivalent transformation proposed by Șerbanuță and Roșu [21, 22], called the SR transformation. This converts a left-linear confluent normal CTRS into a TRS which is computationally equivalent to the CTRS. This means that the converted TRS can be used to exactly simulate reduction sequences of the CTRS to normal forms.

This paper aims at investigating sufficient conditions for soundness of the SR transformation w.r.t. reduction. Neither any unraveling nor the SR transformation is sound for (reduction of) all CTRSs. Here, soundness for a CTRS means that reduction of the converted TRS creates no undesired reduction for the CTRS. Since soundness is one of the most important properties for transformations of CTRSs, sufficient conditions for soundness have been well investigated, especially for unravelings (see, e.g., [9, 17, 10]). For example, the simultaneous unraveling [15], which is proposed by Marchiori (and then improved by Ohlebusch [18]), is sound for weakly left-linear, confluent, non-erasing, or ground conditional normal CTRSs [9].

As for unravelings, soundness of the SR transformation plays a very important role for, e.g., computational equivalence. The main purpose of transformations along the Viry’s approach is to use the soundly transformed TRS to simulate the reduction of the original CTRS. The experimental results in [21] indicate that the rewriting engine using the soundly transformed TRS is much more efficient than the one using the original left-linear confluent CTRS. However, unlike unravelings, soundness conditions for the SR transformation have not been investigated well, and the known conditions are left-linearity or confluence of CTRSs [21, 22]. To get an efficient rewriting engine for CTRSs, soundness conditions for the SR transformation are worth investigating.

To clarify the relationship between unravelings and the SR transformation in terms of soundness, it has been shown that if the SR transformation is sound for a CTRS, then so is the corresponding unraveling [17]. This is not so surprising since the SR transformation is more powerful than unravelings in terms of evaluating conditions in parallel. For the same reason, however, it is not so easy to prove the converse of the above claim — as shown later, the converse does not hold for all normal CTRSs.

In this paper, we show that the SR transformation is sound for weakly left-linear normal CTRSs. To this end, we first show that every reduction sequence of the transformed TRS starting with a term corresponding to the one considered on the CTRS is simulated by the reduction of the unraveled TRS obtained by the simultaneous unraveling [15, 18]. Then, we use the fact that the unraveling is sound for weakly left-linear normal CTRSs. One of the reasons why we take this approach is to avoid conditional rewriting in proofs for soundness.

As already described, unravelings are nice tools to analyze properties of CTRSs, and the SR transformation is a nice tool to get a computationally equivalent TRS which provides very efficient computation compared to the one on the original CTRS. For this reason, we do not discuss the usefulness of our results for analyzing properties of CTRSs, and we concentrate on soundness conditions of the SR transformation.
This paper is organized as follows. In Section 2, we briefly recall basic notions of term rewriting. In Section 3, we recall the notion of soundness, the simultaneous unraveling, and the SR transformation for normal CTRSs. We will adopt a slightly different formulation of the SR transformation from the original one [21], while the resulting TRSs are the same. In Section 4, we show that the SR transformation is sound for weakly left-linear normal CTRSs. In Section 5, we conclude this paper and describe future work on this research.

2 Preliminaries

In this section, we recall basic notions and notations of term rewriting [2, 19].

Throughout the paper, we use $V$ as a countably infinite set of variables. Let $F$ be a signature, a finite set of function symbols each of which has its own fixed arity, and $\text{arity}_F(f)$ be the arity of function symbol $f$. We often write $f/n \in F$ instead of $f \in F$ and $\text{arity}_F(f) = n$. The set of terms over $F (\subseteq F)$ and $V (\subseteq V)$ is denoted by $T(F, V)$, and the set of variables appearing in any of the terms $t_1, \ldots, t_n$ is denoted by $\text{Var}(t_1, \ldots, t_n)$. A term $t$ is called ground if $\text{Var}(t) = \emptyset$. A term is called linear if any variable occurs in the term at most once, and called linear w.r.t. a variable if the variable appears at most once in $t$. The function symbol at the root position $\varepsilon$ of term $t$ is denoted by $\text{root}(t)$. Given an $n$-hole context $C[\ ]$ with parallel positions $p_1, \ldots, p_n$, the notation $C[t_1, \ldots, t_n]_{p_1, \ldots, p_n}$ represents the term obtained by replacing hole $\Box$ at position $p_i$ with term $t_i$ for all $1 \leq i \leq n$. We may omit the subscript “$p_1, \ldots, p_n$” from $C[\cdots]_{p_1, \ldots, p_n}$. For positions $p$ and $p'$ of a term, we write $p' \geq p$ if $p$ is a prefix of $p'$ (i.e., there exists a sequence $q$ such that $pq = p'$). Moreover, we write $p' > p$ if $p$ is a proper prefix of $p'$.

The domain and range of a substitution $\sigma$ are denoted by $\text{Dom}(\sigma)$ and $\text{Ran}(\sigma)$, respectively. We may denote $\sigma$ by $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ if $\text{Dom}(\sigma) = \{x_1, \ldots, x_n\}$ and $\sigma(x_i) = t_i$ for all $1 \leq i \leq n$. For $F (\subseteq F)$ and $V (\subseteq V)$, the set of substitutions that range over $F$ and $V$ is denoted by $\text{Sub}(F, V)$: $\text{Sub}(F, V) = \{\sigma \mid \text{Ran}(\sigma) \subseteq T(F, V)\}$. For a substitution $\sigma$ and a term $t$, the application $\sigma(t)$ of $\sigma$ to $t$ is abbreviated to $t\sigma$, and $t\sigma$ is called an instance of $t$. Given a set $X$ of variables, $\sigma|_X$ denotes the restricted substitution of $\sigma$ w.r.t. $X$: $\sigma|_X = \{x \mapsto x\sigma \mid x \in \text{Dom}(\sigma) \cap X\}$.

An (oriented) conditional rewrite rule over a signature $F$ is a triple $(l, r, c)$, denoted by $l \rightarrow r \Leftarrow c$, such that the left-hand side $l$ is a non-variable term in $T(F, V)$, the right-hand side $r$ is a term in $T(F, V)$, and the conditional part $c$ is a sequence $s_1 \rightarrow t_1; \ldots; s_k \rightarrow t_k$ of term pairs $(k \geq 0)$ where all of $s_1, t_1, \ldots, s_k, t_k$ are terms in $T(F, V)$. In particular, a conditional rewrite rule is called unconditional if the conditional part is the empty sequence (i.e., $k = 0$), and we may abbreviate it to $l \rightarrow r$. We sometimes attach a unique label $\rho$ to the conditional rewrite rule $l \rightarrow r \Leftarrow c$ by denoting $\rho : l \rightarrow r \Leftarrow c$, and we use the label to refer to the rewrite rule.

An (oriented) conditional term rewriting system (CTRS) over a signature $F$ is a set of conditional rules over $F$. A CTRS is called an (unconditional) term rewriting system (TRS) if every rule $l \rightarrow r \Leftarrow c$ in the CTRS is unconditional and satisfies $\text{Var}(l) \supseteq \text{Var}(r)$. The reduction relation of a CTRS $R$ is defined as $\rightarrow_R = \bigcup_{n \geq 0} \rightarrow_{(n), R}$ where $\rightarrow_{(0), R} = \emptyset$, and $\rightarrow_{(n+1), R} = \{(C[\sigma], r) : \rho : l \rightarrow r \Leftarrow s_1 \rightarrow t_1; \ldots; s_k \rightarrow t_k \in R, s_1\sigma \rightarrow_{(i), R} t_1\sigma, \ldots, s_k\sigma \rightarrow_{(i), R} t_k\sigma\}$ for $i \geq 0$. To specify the applied rule $\rho$ and the position $p$ where $\rho$ is applied, we may write $\rightarrow_{p, \rho} \text{ instead of } \rightarrow_{R}$, and moreover, we may write $\rightarrow_{p, \rho}$ instead of $\rightarrow_{p, \rho, R}$ if $p > \varepsilon$. The parallel reduction $\equiv_R$ is defined as follows: $\equiv_R = \{(C[s_1, \ldots, s_n]_{p_1, \ldots, p_n}, C[t_1, \ldots, t_n]_{p_1, \ldots, p_n}) \mid s_1 \rightarrow_R t_1, \ldots, s_n \rightarrow_R t_n\}$. To specify the positions $p_1, \ldots, p_n$ in the definition, we may write $\equiv_{\{p_1, \ldots, p_n\}, R}$ instead of $\equiv_R$, and we may
write \( \Rightarrow_{\leq m} \) instead of \( \Rightarrow \) if \( p_i > \varepsilon \) for all \( 1 \leq i \leq n \). We denote \( n \)-step parallel reduction by \( \Rightarrow_{\leq n} \), and for \( m > n \), we may write \( \Rightarrow_{\leq m} \) instead of \( \Rightarrow_{\leq n} \).

A conditional rewrite rule \( l \rightarrow r \Leftarrow c \) is called left-linear if \( l \) is linear, right-linear if \( r \) is linear, non-erasing if \( \text{Var}(l) \subseteq \text{Var}(r) \), and ground conditional if \( c \) contains no variable. For a syntactic property \( P \) of conditional rewrite rules, we say that a CTRS has the property \( P \) if all of its rules have the property \( P \), e.g., a CTRS is called left-linear if all of its rules are left-linear.

A conditional rewrite rule \( \rho : l \rightarrow r \Leftarrow s_1 \ldots s_k \rightarrow t_k \) is called normal if \( \text{Var}(s_1, \ldots, s_k) \subseteq \text{Var}(l) \) and \( t_1, \ldots, t_n \) are normal forms w.r.t. the underlying unconditional system \( \mathcal{R}_u = \{ l \rightarrow r \mid l \rightarrow r \Leftarrow c \in \mathcal{R} \} \). A CTRS is called normal (or a normal CTRS) if every rewrite rule of the CTRS is normal. Note that we consider \( \mathcal{R} \)-CTRSs (i.e., \( \text{Var}(l) \supseteq \text{Var}(r) \) for \( l \rightarrow r \Leftarrow c \)). A normal CTRS \( \mathcal{R} \) is called weakly left-linear [9] if every conditional rewrite rule having at least one condition is left-linear, and for every unconditional rule, any non-linear variable in the left-hand side does not occur in the right-hand side.

Let \( \mathcal{R} \) be a CTRS over a signature \( F \). The sets of defined symbols and constructors of \( \mathcal{R} \) are denoted by \( D_{\mathcal{R}} \) and \( C_{\mathcal{R}} \), respectively: \( D_{\mathcal{R}} = \{ \text{root}(l) \mid l \rightarrow r \Leftarrow c \in \mathcal{R} \} \) and \( C_{\mathcal{R}} = F \setminus D_{\mathcal{R}} \). Terms in \( T(C_{\mathcal{R}}, V) \) are constructor terms of \( \mathcal{R} \). \( \mathcal{R} \) is called a constructor system if all proper subterms of the left-hand sides in \( \mathcal{R} \) are constructor terms of \( \mathcal{R} \).

### 3 Transformations from Normal CTRSs into TRSs

In this section, we recall the notion of soundness, the simultaneous unraveling [19], the SR transformation [21] for normal CTRSs.

We first show a general notion of soundness of completeness between two (C)TRSs (see [8, 17]). Let \( \mathcal{R}_1 \) and \( \mathcal{R}_2 \) be (C)TRSs over signature \( F_1 \) and \( F_2 \), respectively, \( \phi \) be an initialization (total) mapping from \( T(F_1, V) \) to \( T(F_2, V) \), and \( \psi \) be a partial inverse of \( \phi \), a backtranslation mapping from \( T(F_2, V) \) to \( T(F_1, V) \) such that \( \psi(\phi(t_i)) = t_i \) for any term \( t_i \in T(F_1, V) \). We say that

- \( \mathcal{R}_2 \) is sound (reduction of) \( \mathcal{R}_1 \) w.r.t. \( (\phi, \psi) \) if, for any term \( t_i \in T(F_1, V) \) and for any term \( t_2 \in T(F_2, V) \), \( \psi(t_1) \rightarrow^{\mathcal{R}_2} \) \( t_2 \) implies \( t_1 \rightarrow^{\mathcal{R}_1} \psi(t_2) \) whenever \( \psi(t_2) \) is defined, and
- \( \mathcal{R}_2 \) is complete for (reduction of) \( \mathcal{R}_1 \) w.r.t. \( (\phi, \psi) \) if, for all terms \( t_i, t'_i \in T(F_1, V) \), \( t_1 \rightarrow^{\mathcal{R}_1} t'_1 \) implies \( \phi(t_1) \rightarrow^{\mathcal{R}_2} \phi(t_i) \).

For the sake of readability, we restrict our interest to CTRSs, any rule of which has at most one condition. Note that this is not a restriction on the results in this paper (see [21]). We often denote a sequence \( t_i, t_{i+1}, \ldots, t_j \) of terms by \( t_{i..j} \). Moreover, for the application of a mapping \( f \) to \( t_{i..j} \), we denote \( f(t_i), \ldots, f(t_j) \) by \( f(t_{i..j}) \), e.g., for a substitution \( \theta \), we denote \( t_i \theta, \ldots, t_j \theta \) by \( \theta(t_{i..j}) \).

#### 3.1 Simultaneous Unraveling

The simultaneous unraveling for normal CTRSs, which is reformulated by Ohlebusch, is defined as follows.

\[ \textbf{Definition 1 (U [19])}. \text{ Let } \mathcal{R} \text{ be a normal CTRS over a signature } F. \text{ Then,} \]

\[ U(\rho : l \rightarrow r \Leftarrow s \rightarrow t) = \{ l \rightarrow U_\rho(s, x_{1..n}), \ U_\rho(t, x_{1..n}) \rightarrow r \} \]

where \( \{ x_1, \ldots, x_n \} = \text{Var}(l) \) and \( U_\rho \) is a fresh \( n + 1 \)-ary function symbol, called a \textit{U symbol}. Note that for every unconditional rule \( l \rightarrow r \in \mathcal{R} \), \( U(l \rightarrow r) = \{ l \rightarrow r \} \). \( U \) is straightforwardly
extended to normal CTRSs: \( \mathcal{U}(\mathcal{R}) = \bigcup_{\rho \in \mathcal{R}} \mathcal{U}(\rho) \). We abuse \( \mathcal{U} \) to represent the extended signature of \( \mathcal{F} \): \( \mathcal{U}_\mathcal{F}(\mathcal{F}) = \mathcal{F} \cup \{ \rho \mid \rho : t \rightarrow s \in \mathcal{R} \} \). We omit \( \mathcal{R} \) from \( \mathcal{U}_\mathcal{F}(\mathcal{F}) \).

Note that \( \mathcal{U}(\mathcal{R}) \) is a TRS over \( \mathcal{U}(\mathcal{F}) \). We say that \( \mathcal{U} \) (and also \( \mathcal{U}(\mathcal{R}) \)) is sound (complete) for \( \mathcal{R} \) if \( \mathcal{U}(\mathcal{R}) \) is sound (complete) for \( \mathcal{R} \) w.r.t. \( (\mathcal{id}, \mathcal{id}) \), where \( \mathcal{id} \) is the identity mapping for \( T(\mathcal{F}, \mathcal{V}) \).

Note that \( \mathcal{U} \) is complete for all normal CTRSs [15, 18].

**Example 2.** Consider the following normal CTRS, a simplified variant of the one in [21]:

\[ \mathcal{R}_1 = \{ e(0) \rightarrow \text{true}, \quad e(s(x)) \rightarrow \text{true} \leftarrow e(x) \rightarrow \text{false}, \quad e(s(x)) \rightarrow \text{false} \leftarrow e(x) \rightarrow \text{true} \} \]

\( \mathcal{R}_1 \) is unraveled to the following TRS:

\[ \mathcal{U}(\mathcal{R}_1) = \{ e(0) \rightarrow \text{true}, \quad e(s(x)) \rightarrow u_1(e(x), x), \quad u_1(\text{false}, x) \rightarrow \text{true}, \quad e(s(x)) \rightarrow u_2(e(x), x), \quad u_2(\text{true}, x) \rightarrow \text{false} \} \]

\( \mathcal{U}(\mathcal{R}_1) \) is not confluent, while \( \mathcal{R}_1 \) is confluent. This means that \( \mathcal{U} \) does not always preserve confluence of CTRSs. The reason why \( \mathcal{U} \) loses confluence is that once we start evaluating a condition, the expected goal for the condition is fixed and then we cannot cancel the evaluation. This is illustrated in the derivation \( e(s(0)) \rightarrow_{\mathcal{U}(\mathcal{R}_1)} u_1(e(0), 0) \rightarrow_{\mathcal{U}(\mathcal{R}_1)} u_1(\text{true}, 0) \). To reduce \( e(s(0)) \) to false, we should have applied \( e(s(x)) \rightarrow u_2(e(x), x) \in \mathcal{U}(\mathcal{R}_1) \) to the initial term. However, we applied another wrong rule, \( u_1 \) expects \( e(0) \) to be reduced to true (the expected goal for \( u_1 \) at this point), and we cannot redo applying the desired rule.

To simplify the discussion, we do not consider any optimization of unravelings (see e.g. [11]). As shown in [15], \( \mathcal{U} \) is not sound for every normal CTRS (see also [19, Example 7.2.14]). For some classes of normal CTRSs, \( \mathcal{U} \) is sound (cf. [9, 17]).

**Theorem 3 ([9]).** \( \mathcal{U} \) is sound for a normal CTRS satisfying at least one of the following: weak left-linearity, confluence, non-erasingness, or ground conditional.

### 3.2 The SR Transformation

Next, we introduce the SR transformation and its properties. In the following, the word "conditional rule" is used for representing rules having exactly one condition.

Before transforming a CTRS \( \mathcal{R} \), we first extend the signature of \( \mathcal{R} \) as follows:

- we leave constructors of \( \mathcal{R} \) without any change,
- the arity \( n \) of defined symbol \( f \) is extended to \( n + m \) where \( f \) has \( m \) conditional rules in \( \mathcal{R} \), and we replace \( f \) by \( \overline{f} \) with the arity \( n + m \), and
- a fresh constant \( \bot \) and a fresh unary symbol \( (\cdot) \) are introduced.

We denote the extended signature by \( \overline{\mathcal{F}}: \overline{\mathcal{F}} = \{ c \mid c \in \mathcal{C}_\mathcal{R} \} \cup \{ \overline{f} \mid f \in \mathcal{D}_\mathcal{R} \} \cup \{ \bot, (\cdot) \} \).

We introduce a mapping \( \text{ext}(\cdot) \) to extend the arguments of defined symbols in a term as follows: \( \text{ext}(x) = x \) for \( x \in \mathcal{V} \); \( \text{ext}(c(t_{1..n})) = c(\text{ext}(t_{1..n})) \) for \( c/n \in \mathcal{C}_\mathcal{R} \); \( \text{ext}(\overline{f}(t_{1..n})) = \overline{f}(\text{ext}(t_{1..n}), z_{1..m}) \) for \( f/n \in \mathcal{D}_\mathcal{R} \), where \( f \) has \( m \) conditional rules in \( \mathcal{R} \), \( \text{arity}_{\mathcal{R}}(\overline{f}) = n + m \), and \( z_{1..m} \) are fresh variables. The extended arguments of \( \overline{f} \) are used for evaluating the corresponding conditions, and the fresh constant \( \bot \) is introduced to the extended arguments of defined symbols, which does not store any evaluation. To put \( \bot \) into the extended arguments, we define a mapping \( (\cdot)^\bot \) which puts \( \bot \) to all the extended arguments of defined symbols, as follows: \( \langle x \rangle^\bot = x \) for \( x \in \mathcal{V} \); \( \langle c(t_{1..n}) \rangle^\bot = c(\langle t_{1..n} \rangle^\bot) \) for \( c/n \in \mathcal{C}_\mathcal{R} \); \( \langle \overline{f}(t_{1..n}, u_{1..m}) \rangle^\bot = \overline{f}(\langle t_{1..n} \rangle^\bot, \bot, \ldots, \bot) \) for \( f/n \in \mathcal{D}_\mathcal{R} \); \( \langle (t) \rangle^\bot = \langle t \rangle^\bot \); \( \langle \bot \rangle^\bot = \bot \). Note that in applying \( (\cdot)^\bot \) to reachable terms defined later, the case of applying \( (\cdot)^\bot \) to \( \bot \) never happens. Now we define a mapping \( \tau \) from \( T(\mathcal{F}, \mathcal{V}) \) to \( T(\overline{\mathcal{F}}, \mathcal{V}) \) as \( \tau = (\text{ext}(t))^\bot \).

The SR transformation [21] is defined as follows.
Definition 4 (SR). Let \( f/n \in \mathcal{D}_R \) that has \( m \) conditional rules in \( \mathcal{R} \) (i.e., \( f/(n+m) \in \mathcal{T} \)). Then, \( \text{SR}(f(w_{1..n}) \rightarrow r_1) = \{ f(\text{ext}(w_{1..n}), z_{1..m}) \rightarrow (\tau) \} \) and, for the \( i \)-th conditional rule of \( f \),

\[
\text{SR}(f(w_{1..n}) \rightarrow r_i \leftarrow s_i \rightarrow t_i) = \\
\{ \tilde{f}(w'_{1..n}, z_{1..i-1}, \perp, z_{i+1..m}) \rightarrow \tilde{f}(w'_{1..n}, z_{1..i-1}, (s_i), z_{i+1..m}), \\
\tilde{f}(w'_{1..n}, z_{1..i-1}, (t_i), z_{i+1..m}) \rightarrow (\tau) \}
\]

where \( w'_{1..n} = \text{ext}(w_{1..n}) \) and \( z_1, \ldots, z_m \) are fresh variables. The set of auxiliary rules is defined as follows:

\[
\mathcal{R}_{aux} = \{ \langle x \rangle \rightarrow \langle x \rangle \} \cup \{ c(x_{1..i-1}, (x_i), x_{i+1..n}) \rightarrow \langle c(x_1) \rangle \mid c/n \in \mathcal{C}_R, 1 \leq i \leq n \} \\
\cup \{ \tilde{f}(x_{1..i-1}, (x_i), x_{i+1..n}, z_{1..m}) \rightarrow (\tilde{f}(x_{1..n}, \perp, \ldots, \perp)) \mid f/n \in \mathcal{D}_R, 1 \leq i \leq n \}
\]

where \( z_1, \ldots, z_m \) are fresh variables. The transformation \( \text{SR} \) is defined as follows: \( \text{SR}(\mathcal{R}) = \bigcup_{\rho \in \mathcal{R}} \text{SR}(\rho) \cup \mathcal{R}_{aux} \). Note that \( \text{SR}(\mathcal{R}) \) is a TRS over \( \mathcal{T} \). Note also that \( \mathcal{R}_{aux} \) is linear. The backtransformation mapping \( \hat{\cdot} \) for \( \cdot \) is defined as follows: \( \hat{x} = x \) for \( x \in \mathcal{V} \); \( c(t_{1..n}) = c(t_{1..n}) \) for \( c/n \in \mathcal{C}_R \); \( \tilde{f}(t_{1..n}, u_{1..m}) = f(t_{1..n}) \) for \( f/n \in \mathcal{D}_R \); \( \hat{\tau} = \perp \). Note that \( \hat{\cdot} \) is a total function. A term \( t \) in \( T(\mathcal{T}, \mathcal{V}) \) is called reachable if there exists a term \( s \in T(\mathcal{T}, \mathcal{V}) \) such that \( (\hat{x}) \rightarrow_{\text{SR}(\mathcal{R})} t \). We say that \( \text{SR} \) (and also \( \text{SR}(\mathcal{R}) \)) is sound (complete) for \( \mathcal{R} \) if \( \text{SR}(\mathcal{R}) \) is sound (complete) for \( \mathcal{R} \) w.r.t. \( (\hat{\cdot}, \hat{\cdot}) \).

Note that \( \text{SR} \) is complete for all CTRSs [21]. Note also that \( \text{SR} \) is not sound for all normal CTRSs since for any normal CTRS \( \mathcal{R} \), \( \text{SR}(\mathcal{R}) \) can simulate any reduction of \( U(\mathcal{R}) \) [17] — roughly speaking, any undesired derivation on \( U(\mathcal{R}) \) holds on \( \text{SR}(\mathcal{R}) \). It is clear that for any reachable term \( t \in T(\mathcal{T}, \mathcal{V}) \), any term \( t' \in T(\mathcal{T}, \mathcal{V}) \) with \( t \rightarrow_{\text{SR}(\mathcal{R})} t' \) is reachable.

To evaluate the condition (an instance of \( c \)) for \( \text{SR}(\mathcal{R}) \), any substitution \( s \in \mathcal{D}_R \), and its rules in \( \mathcal{R} \), roughly speaking, any undesired derivation on \( \text{SR}(\mathcal{R}) \), any substitution \( s \in \mathcal{D}_R \), and its rules in \( \mathcal{R} \), any substitution \( s \in \mathcal{D}_R \), and its rules in \( \mathcal{R} \) (i.e., \( \text{SR}(\mathcal{R}) \)) is sound (complete) for \( \mathcal{R} \) if \( \text{SR}(\mathcal{R}) \) is sound (complete) for \( \mathcal{R} \) w.r.t. \( (\hat{\cdot}, \hat{\cdot}) \).

Example 5. Consider \( \mathcal{R}_1 \) in Example 2 again. \( \mathcal{R}_1 \) is transformed by \( \text{SR} \) as follows:

\[
\text{SR}(\mathcal{R}_1) = \\
\{ \langle 0, (z_1, z_2) \rangle \rightarrow (true), \\
\langle s(x), (\perp, \perp), (z_1, (x, \perp, \perp)) \rangle \rightarrow (false), \\
\langle s(x), z_1 \rangle \rightarrow \langle s(x), (z_1, (x, \perp, \perp)) \rangle, \\
\langle (\langle x \rangle), s(x) \rangle \rightarrow \langle s(x) \rangle, \\
\langle (\langle x \rangle), z_1, z_2 \rangle \rightarrow \langle s(x), (\perp, \perp) \rangle \}
\]

In contrast to \( U \), \( \text{SR} \) preserves confluence of CTRSs as confluence on reachable terms, e.g., \( \text{SR}(\mathcal{R}_1) \) is confluent on reachable terms, while \( U(\mathcal{R}_1) \) is not. Note that \( \text{SR}(\mathcal{R}_1) \) is not confluent. Let us consider the derivation starting from \( \langle s(0) \rangle \) in Example 2 again. The corresponding derivation on \( \text{SR}(\mathcal{R}_1) \) is illustrated as follows:

\[
\langle s(0), \perp, \perp \rangle \rightarrow_{\text{SR}(\mathcal{R}_1)} \langle s(0), (\langle 0, (\perp, \perp) \rangle, \perp) \rangle \\
\rightarrow_{\text{SR}(\mathcal{R}_1)} \langle s(0), (true), \perp \rangle
\]

Unlike the case of \( U(\mathcal{R}_1) \), we can apply the desired rule to the last term above:

\[
\cdots \rightarrow_{\text{SR}(\mathcal{R}_1)} \langle s(0), (true), (\langle 0, (\perp, \perp) \rangle) \rangle \\
\rightarrow_{\text{SR}(\mathcal{R}_1)} \langle s(0), (true), (true) \rangle \\
\rightarrow_{\text{SR}(\mathcal{R}_1)} \langle false \rangle
\]
In the case of $U$, to reach false, we need to backtrack from the undesired normal form $u_1(\text{true}, 0)$ (see Example 2), but in the case of $SR$, we do not have to backtrack — choosing an arbitrary redex from reducible terms is sufficient to reach a desired normal form since $\text{SR}(R_1)$ is confluent on reachable terms.

Finally, we recall some important properties of $\text{SR}$.

- **Theorem 6 ([21])**. $\text{SR}$ is sound for left-linear or confluent\(^1\) normal CTRSs.

- **Theorem 7 ([17])**. If $\text{SR}$ is sound for a normal CTRS, then so is $U$.

## 4 Soundness of the SR Transformation for Weakly Left-linear CTRSs

In this section, by using $U$, we show that $\text{SR}$ is sound for a weakly left-linear normal CTRS.

Before the discussion, we consider the role of $(\cdot)\|^\perp$ again. The mapping $(\cdot)^\perp$ puts $\perp$ into the extended arguments of defined symbols. We straightforwardly extend $(\cdot)^\perp$ to substitutions: $(\theta)^\perp = \{ x \mapsto (x\theta)^\perp \mid x \in \text{Dom}(\theta) \}$ for a substitution $\theta$ such that $\text{Ran}(\theta) \subseteq T(\mathcal{F}, \mathcal{V})$. The mapping $(\cdot)^\perp$ has the following properties which are trivial by definition.

- **Proposition 8**. Let $R$ be a normal CTRS. Then, all of the following hold:
  - For any term $s \in T(\mathcal{F}, \mathcal{V})$, $\pi = (\pi)^\perp$.
  - For any term $t \in T(\mathcal{F}, \mathcal{V})$, $(t\theta)^\perp = (t)^\perp(\theta)^\perp$ for any substitution $\theta \in \text{Sub}(\mathcal{F}, \mathcal{V})$ such that $t\theta$ is reachable.

We may use Proposition 8 without notice.

To prove key claims (e.g., Lemma 13 shown later) related to the derivation $(\pi) \rightarrow^{*}_{\text{SR}(R)} t$, the mappings $\tau$ and $\tau$ often prevent us from using induction because $\tau$ removes all occurrences of $(\cdot)$ from terms. For this reason, using the mapping $(\cdot)^\perp(\cdot)$ instead of $(\cdot)$ is a breakthrough to prove our main theorem.

Next, we observe reduction sequences $(\pi) \rightarrow^{*}_{\text{SR}(R)} t$ with $s \in T(\mathcal{F}, \mathcal{V})$ and $t \in T(\mathcal{F}, \mathcal{V})$. The main feature of $\text{SR}$ is to evaluate two or more conditions in parallel. However, to get $\widehat{\tilde{t}}$, it suffices to evaluate successfully at most one condition in each parallel evaluation of conditions. This means that every term appearing in $(\pi) \rightarrow^{*}_{\text{SR}(R)} t$, which is rooted by a defined symbol $\tilde{f}$, is of the form $\tilde{f}(t_{1..n}, u_{1..m})$ where $\text{arity}_F(f) = n$ and at most one of $u_1, \ldots, u_m$ is rooted by $(\cdot)$ (i.e., others are $\perp$). Such a term is the key idea of this paper, and we say that the term has no parallel evaluation of conditions. For a term having no parallel evaluation of conditions, we can uniquely determine the corresponding term over $U(\mathcal{F})$: for a term $\tilde{f}(t_{1..n}, u_{1..m})$, if all $u_1, \ldots, u_m$ are $\perp$, then the root is $f$, and otherwise, assuming that $u_i$ is not $\perp$ and the others are $\perp$, then the root is $U_{t_i}$ which is introduced for the $i$-th conditional rule of $f$. This correspondence is illustrated in Figure 1. We first show how to convert a reachable term in $T(\mathcal{F}, \mathcal{V})$ to a term in $T(U(\mathcal{F}), \mathcal{V})$.

- **Definition 9**. Let $R$ be a normal CTRS. Then, we define a mapping $\Phi$ from reachable terms in $T(\mathcal{F}, \mathcal{V})$ to $T(U(\mathcal{F}), \mathcal{V})$ as follows:
  - $\Phi(x) = x$ for $x \in \mathcal{V}$,
  - $\Phi(c(t_{1..n})) = c(\Phi(t_{1..n}))$ for $c/n \in \mathcal{C}_R$,
  - $\Phi(\tilde{f}(t_{1..n}, \perp, \ldots, \perp)) = f(\Phi(t_{1..n}))$ for $f/n \in \mathcal{D}_R$.

---

\(^1\) In [21], “ground confluence” is used instead of “confluence” since reduction sequences on ground terms are considered. From the proofs in [21], we can consider “confluence” for the case that arbitrary reduction sequences are considered.
We have that Φ((false)) = 0.

Note that Φ(θ) is a partial mapping, and linearity is not important for Φ while \( w_1, \ldots, w_n \) in the proof of Lemma 13 are linear without any shared variable. By definition, it is clear that Φ(θ) is defined for any variable \( x \in \text{Dom}(θ) \), and Φ((\( \langle x \rangle \)) \( \rightarrow \) \( \text{sr}(\{u_i\}) \)) for \( f/n \in D_R \), where the \( i \)-th conditional rule of \( f \) is \( \rho : f(w_1, \ldots, w_n) \rightarrow s_i \rightarrow t_i \in R, U(t_i, x_1, \ldots, x_m) \rightarrow r_i \in U(\mu) \) with \( \mu(w_1, \ldots, w_n) = x_1 \), \( Φ(θ(t)) \) is defined for all \( 1 \leq j \leq n, Φ(θ) \) is defined for any variable \( x \in \text{Dom}(θ) \), and we denote by Φ(θ) the substitution \( \{ x \rightarrow Φ(θ(x)) \mid x \in \text{Dom}(θ) \} \).

Note that Φ is a partial mapping, and linearity is not important for Φ while \( w_1, \ldots, w_n \) in the proof of Lemma 13 are linear without any shared variable. By definition, it is clear that Φ(θ) is defined for any variable \( x \in \text{Dom}(θ) \), and we denote by Φ(θ) the substitution \( \{ x \rightarrow Φ(θ(x)) \mid x \in \text{Dom}(θ) \} \).

**Example 10.** Consider \( U(R_1) \) and \( \text{sr}(R_1) \) in Examples 2 and 5, respectively, again. We have that Φ((\( \langle s(0), \perp, \perp \rangle \)) = e(s(0)), Φ((\( \langle s(0), \perp, \langle s(0), \perp, \perp \rangle \rangle \)) = u_2(e(s(0))), and Φ((\( \langle s(0), \perp, \langle s(0), \perp, \perp \rangle \rangle \)) = u_2(false, s(0))). On the other hand, Φ is not defined for the term \( \langle \langle s(0), \perp, \langle s(0), \perp, \perp \rangle \rangle \rangle \) which contains two parallel evaluations of conditions.

Unfortunately, the above idea for the proof does not hold for all normal CTRSs.

**Example 11.** Consider the following normal CTRS:

\[
R_2 = \{ f(x) \rightarrow x \leftarrow x \rightarrow c, \quad g(x, x) \rightarrow h(x, x), \quad h(f(d), x) \rightarrow x, \quad a \rightarrow c, \quad a \rightarrow d, \quad b \rightarrow c, \quad b \rightarrow d \}
\]

\( R_2 \) is transformed by \( U \) and \( \text{sr} \), respectively, as follows:

\[
U(R_2) = \{ f(x) \rightarrow u_3(x, x), \quad u_3(c, x) \rightarrow x, \quad g(x, x) \rightarrow h(x, x), \quad h(f(d), x) \rightarrow x, \quad \ldots \}
\]

\[
\text{sr}(R_2) = \begin{cases}
(\langle f(x), \perp \rangle \rightarrow \langle f(x), \perp \rangle), & (\langle f(x), \perp \rangle \rightarrow \langle f(x), \perp \rangle), \quad \pi \rightarrow \langle f(x), \perp \rangle, \quad \pi \rightarrow \langle f(x), \perp \rangle, \\
(\langle f(x), \perp \rangle \rightarrow \langle c, \perp \rangle) \rightarrow \langle f(x), \perp \rangle), & (\langle f(x), \perp \rangle \rightarrow \langle c, \perp \rangle) \rightarrow \langle f(x), \perp \rangle), \quad \pi \rightarrow \langle c, \perp \rangle, \quad \pi \rightarrow \langle c, \perp \rangle, \\
(\langle f(x), \perp \rangle \rightarrow \langle f(x), \perp \rangle), & (\langle f(x), \perp \rangle \rightarrow \langle f(x), \perp \rangle), \quad \pi \rightarrow \langle f(x), \perp \rangle, \quad \pi \rightarrow \langle f(x), \perp \rangle,
\end{cases}
\]

We have the following derivations:

\[
\begin{align*}
\langle g(\langle f(x), \perp \rangle), \langle f(x), \perp \rangle) \rangle & \rightarrow_{\text{sr}(R_2)} \langle g(\langle f(d), \perp \rangle), \langle f(d), \perp \rangle) \rangle \\
\langle g(\langle f(x), \perp \rangle), \langle f(x), \perp \rangle) \rangle & \rightarrow_{\text{sr}(R_2)} \langle g(\langle f(x), \perp \rangle), \langle f(x), \perp \rangle) \rangle \\
\langle g(f(a), f(b)) \rangle & \rightarrow_{U(R_2)} g(f(a), f(b)) \\
\langle g(u_3(c, d), u_3(c, d)) \rangle & \rightarrow_{\text{sr}(R_2)} \langle g(u_3(c, d), u_3(c, d)) \rangle \\
\langle u_3(c, d) \rangle & \rightarrow_{U(R_2)} u_3(c, d) \\
\langle d \rangle & \rightarrow_{U(R_2)} d
\end{align*}
\]

Neither \( U(R_2) \) nor \( R_2 \) can simulate the derivation \( \langle g(\langle f(x), \perp \rangle), \langle f(x), \perp \rangle) \rangle \rightarrow_{\text{sr}(R_2)} \langle d \rangle \), and thus, \( \text{sr} \) is not sound for \( R_2 \).
As shown below, we only succeed in proving that the idea works for weakly left-linear normal CTRSs. Weakly left-linear normal CTRSs have the following syntactic properties with respect to SR, which are trivial by definition.

**Proposition 12.** If $\mathcal{R}$ is weakly left-linear, then $\overline{\mathcal{R}}(\mathcal{R})$ is weakly left-linear, especially, for transformed rewrite rules $\overline{t}(w'_{1,n},z_{i_{i-1}},\ldots,z_{i_{1+m}}) \rightarrow \overline{t}(w'_{1,n},z_{1_{1-1}},\ldots,z_{1_{i_{1+m}}})$ and $\overline{t}(w'_{1,n},z_{i_{i-1}},\ldots,z_{i_{1+m}}) \rightarrow \overline{t}(t_1)$ in $\mathcal{R}$, then $\overline{t}(w'_{1,n}) \rightarrow t_1$ is left-linear.

The following claim is an auxiliary lemma to show that $U(\mathcal{R})$ can simulate every reduction sequence of the form $(\overline{\sigma}) \rightarrow t$ with $s \in T(\mathcal{F,V})$ and $t \in T(\mathcal{F,V})$.

**Lemma 13.** Let $\mathcal{R}$ be a weakly left-linear normal CTRS, $s$ be a term in $T(\mathcal{F,V})$, and $t$ be a term in $T(\mathcal{C}_R \cup \{\} \cup \{\})$, and $\theta \in \text{Sub}(\mathcal{F,V})$ with $\text{Dom}(\theta) \subseteq \text{Var}(t)$. If $s \rightarrow t \theta$ ($k \geq 0$) and $\Phi(s)$ is defined, then there exists a substitution $\theta' \in \text{Sub}(\mathcal{F,V})$ such that

- $\text{Dom}(\theta') = \text{Dom}(\theta)$,
- $\Phi(\theta')$ is defined,
- for any variable $x \in \text{Dom}(\theta')$, if $t$ is linear w.r.t. $x$, then $x \theta' \rightarrow^*_{\overline{\mathcal{R}}} x \theta$ and $\Phi((x \theta')^-)$
- $\Phi((x \theta')^1)$, and otherwise, $x \theta' = (x \theta')^1$,
- $\Phi((s))^1 \rightarrow (s)^1_{\overline{\mathcal{U}}(\mathcal{R})}$.

**Proof.** This lemma can be proved by induction on the lexicographic product $(k, |s|)$ where $|s|$ denotes the size of $s$ (see the appendix in a full version of this paper). \hfill $\blacksquare$

Weak left-linearity is used skillfully in the proof of Lemma 13.

Next, we show that $(\overline{\sigma}) \rightarrow t$ can be simulated by $U(\mathcal{R})$.

**Theorem 14.** Let $\mathcal{R}$ be a weakly left-linear normal CTRS, $s$ be a term in $T(\mathcal{F,V})$, and $t$ be a term in $T(\mathcal{F,V})$. If $(\overline{\sigma}) \rightarrow t$, then $s \rightarrow t$.

**Proof.** By definition, $\Phi((\overline{\sigma})^1) = s$ is defined, and thus, it follows from Lemma 13 that $\Phi((\overline{\sigma})^1) \rightarrow (\overline{\sigma})^1_{\overline{\mathcal{U}}(\mathcal{R})}$. By definition, $\Phi((t)^1) = t$. Therefore, $s \rightarrow t$. \hfill $\blacksquare$

Theorem 14 does not hold for all normal CTRSs.

**Example 15.** Consider the following normal CTRS which is not weakly left-linear:

$\mathcal{R}_3 = \{ f(x) \rightarrow c \leftarrow x \rightarrow c, \quad f(x) \rightarrow d \leftarrow x \rightarrow d, \quad g(x,x) \rightarrow h(x,x), \quad a \rightarrow c, \quad a \rightarrow d, \quad b \rightarrow c, \quad b \rightarrow d \}.$

$\mathcal{R}_3$ is transformed by $U$ and $\overline{\mathcal{R}}$, respectively, as follows:

$U(\mathcal{R}_3) = \{ f(x) \rightarrow u_5(x,x), \quad f(x) \rightarrow u_6(x,x), \quad g(x,x) \rightarrow h(x,x), \quad u_5(c,x) \rightarrow c, \quad u_6(d,x) \rightarrow d, \}.

\overline{\mathcal{R}}(\mathcal{R}_3) = \{ \overline{t}(x,\perp_2) \rightarrow \overline{t}(x,\langle x \rangle_2), \quad \overline{t}(x,\perp_2) \rightarrow \overline{t}(x,\langle z_1 \rangle_2), \quad \overline{t}(x,\langle z_1 \rangle_2) \rightarrow \overline{t}(x,\langle x \rangle_1), \quad \overline{g}(x,x) \rightarrow \overline{h}(x,x),} \ \overline{\alpha} \rightarrow \overline{d}, \ \overline{\beta} \rightarrow \overline{c}, \ \overline{\gamma} \rightarrow \overline{d}, \ \overline{\delta} \rightarrow \overline{c}, \ \overline{\epsilon} \rightarrow \overline{d}, \ \overline{\zeta}(\langle x \rangle) \rightarrow x, \ \overline{\eta}(x,y) \rightarrow \overline{h}(x,y), \ \overline{\iota}(x,y) \rightarrow \overline{h}(x,y), \}$

---

2 Available from http://www.trs.cis.is.nagoya-u.ac.jp/~nishida/wp1e4/.
We have the following derivations:

\[
\begin{align*}
(\overline{g}(\overline{f}(\overline{a}, \perp, \perp), \overline{f}(\overline{b}, \perp, \perp))) & \rightarrow_{SR(\mathcal{R}_3)} (h(\overline{f}(\overline{d}, \{c\}), \overline{f}(\overline{d}, \{c\}))) \\
& \rightarrow_{SR(\mathcal{R}_3)} (h(\overline{c}, \{d\}) & \rightarrow_{U(\mathcal{R}_3)} h(u_5(c, d), u_5(c, d)) \\
& \not\rightarrow_{U(\mathcal{R}_3)} h(c, d)
\end{align*}
\]

The other normal forms of \( g(f(a), f(b)) \) on \( U(\mathcal{R}_3) \) are \( h(u_5(c, c), u_5(c, c)) \), \( h(u_5(d, c), u_5(c, c)) \), and \( h(u_5(d, d), u_5(d, d)) \), but none of them corresponds to \( h(c, d) \). For this reason, the derivation on \( SR(\mathcal{R}_3) \) cannot be simulated by \( U(\mathcal{R}_3) \). The derivation \( g(f(a), f(b)) \rightarrow^* h(c, d) \) does not hold on \( \mathcal{R}_3 \), either, and thus, \( SR \) is not sound for \( \mathcal{R}_3 \). In addition, being a constructor system is not sufficient for soundness of \( SR \) since \( \mathcal{R}_3 \) is a constructor system.

We show the main result obtained by Theorem 14.

**Theorem 16.** \( SR \) is sound for weakly left-linear normal CTRSs.

**Proof.** Let \( \mathcal{R} \) be a weakly left-linear CTRS, \( s \in T(\mathcal{F}, \mathcal{V}) \) and \( t \in T(\overline{\mathcal{F}}, \mathcal{V}) \). Suppose that \( (\overline{x}) \rightarrow_{SR(\mathcal{R})} t \). Then, it follows from Theorem 14 that \( s \rightarrow_{U(\mathcal{R})} \overline{t} \). Since \( \overline{t} \in T(\mathcal{F}, \mathcal{V}) \) and \( U \) is sound for \( \mathcal{R} \), we have that \( s \rightarrow_{R} \overline{t} \). Therefore, \( SR \) is sound for \( \mathcal{R} \). □

Theorem 16 does not hold for all normal CTRSs (see Examples 11 and 15).

Finally, we discuss the remaining soundness conditions of \( U \): non-erasingness and groundness of conditions. Non-erasingness of normal CTRSs is not sufficient for soundness since \( \mathcal{R}_2 \) is non-erasing but \( SR \) is not sound for \( \mathcal{R}_2 \). Groundness of conditions is not sufficient for soundness, either.

**Example 17.** Consider the following ground-conditional normal CTRS, a variant of \( \mathcal{R}_3 \):

\[
\mathcal{R}_4 = \left\{ \begin{array}{l}
f(a) \rightarrow c \leftarrow a \rightarrow c, \\
 f(b) \rightarrow d \leftarrow b \rightarrow d, \\
 g(x, x) \rightarrow h(x, x), \\
 a \rightarrow c, \\
 a \rightarrow d, \\
 b \rightarrow c, \\
 b \rightarrow d \end{array} \right\}
\]

We have that \( g(\overline{f}(\overline{a}, \perp, \perp), \overline{f}(\overline{b}, \perp, \perp)) \rightarrow_{SR(\mathcal{R}_4)} (h(c, d)) \), but \( g(f(a), f(b)) \not\rightarrow_{\mathcal{R}_4} h(c, d) \). Therefore, \( SR \) is not sound for \( \mathcal{R}_4 \).

5 Conclusion

In this paper, by using the soundness of \( U \) for weakly left-linear normal CTRSs, we showed that the SR transformation is sound for weakly left-linear normal CTRSs. As far as we know, this paper is the second work on comparing soundness of unravelings and the SR transformation. The first one is a previous work [17] of the first author, in which the converse of Theorem 7 was left as a conjecture. As a negative result, we showed that the converse of Theorem 7 does not hold in general.

One may think that as the first step, we should have started with the transformation proposed by Antoy et al [1], which is a variant of Viry’s transformation. As described in [21], for constructor systems, the unary symbol introduced in the SR transformation to wrap terms evaluating conditions is not necessary and then the SR transformation is the same as the one in [1]. The transformation in [1] is sound for left-linear constructor normal CTRSs, and is extended to the SR transformation in order to adapt it to arbitrary normal CTRSs. This means that any result for the SR transformation can be adapted to the transformation in [1]. Moreover, the SR transformation has been extended to syntactically or strongly deterministic CTRSs [22], which we would like to deal with at the next step of this research. For these reasons, we started with the SR transformation.
Schernhammer and Gramlich showed in [20] that a particular context-sensitive condition [13] is sufficient for soundness of Ohlebusch’s unraveling [18], which is an improved variant of Marchiori’s one. However, the context-sensitive condition is not sufficient for preserving confluence of CTRSs. For this reason, not all the unraveled TRSs with the context-sensitive condition are computationally equivalent to the original CTRSs, and in this sense, the SR transformation is more useful than unravelings with the context-sensitive condition. Moreover, the context-sensitive condition restricts the reduction to the context-sensitive one. Due to this restriction, we did not use the context-sensitive condition in proving the main result in this paper.

As future work, we will extend Theorem 14 to a pair of Ohlebusch’s unraveling [18] and the SR transformation for syntactically or strongly deterministic CTRSs in order to extend Theorem 16 to the SR transformation for the CTRSs. We did not discuss confluence of normal CTRSs as a soundness condition since the SR transformation is known to be sound for confluent normal CTRSs. However, for the extension to deterministic CTRSs, we will adapt the proof technique in this paper to confluent normal CTRSs. Moreover, we will investigate other soundness conditions of unravelings in order to make the SR transformation applicable to more classes of CTRSs as a computationally equivalent transformation.

Acknowledgement. We are deeply grateful to the anonymous referees for their useful comments to improve this paper. We would like to dedicate this paper to the memory of Bernhard Gramlich who encouraged us to further research the soundness of unravelings.

References

9 Karl Gmeiner, Bernhard Gramlich, and Felix Schernhammer. On (un)soundness of unravelings. In Proceedings of the 21st International Conference on Rewriting Techniques and


