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

Naoki Nishida[1], Makishi Yanagisawa[1], and Karl Gmeiner[2]

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 unreachability for 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)reachablity.

* The research in this paper is partly supported by the Austrian Science Fund (FWF) international project I963 and the Japan Society for the Promotion of Science (JSPS).

1st International Workshop on Rewriting Techniques for Program Transformations and Evaluation (WPTE'14).
Editors: Manfred Schmidt-Schauß, Masahiko Sakai, David Sabel, and Yuki Chiba; pp. 39–50
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

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 Şerbănuţă and Roşu [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 notations 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 $\mathcal{V}$ as a countably infinite set of *variables*. Let $\mathcal{F}$ be a *signature*, a finite set of *function symbols* each of which has its own fixed arity, and $\mathtt{arity}_{\mathcal{F}}(\mathsf{f})$ be the arity of function symbol f. We often write $\mathsf{f}/n \in \mathcal{F}$ instead of "$\mathsf{f} \in \mathcal{F}$ and $\mathtt{arity}_{\mathcal{F}}(\mathsf{f}) = n$". The set of *terms* over F ($\subseteq \mathcal{F}$) and V ($\subseteq \mathcal{V}$) 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 $\mathcal{V}ar(t_1, \ldots, t_n)$. A term t is called *ground* if $\mathcal{V}ar(t) = \emptyset$. 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 ε of term t is denoted by $\mathtt{root}(t)$. 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 $C[\ldots]_{p_1,\ldots,p_n}$. 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* σ are denoted by $\mathcal{D}om(\sigma)$ and $\mathcal{R}an(\sigma)$, respectively. We may denote σ by $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$ if $\mathcal{D}om(\sigma) = \{x_1, \ldots, x_n\}$ and $\sigma(x_i) = t_i$ for all $1 \leq i \leq n$. For F ($\subseteq \mathcal{F}$) and V ($\subseteq \mathcal{V}$), the set of *substitutions* that range over F and V is denoted by $\mathcal{S}ub(F, V)$: $\mathcal{S}ub(F, V) = \{\sigma \mid \mathcal{R}an(\sigma) \subseteq T(F, V)\}$. For a substitution σ and a term t, the application $\sigma(t)$ of σ 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 σ w.r.t. X: $\sigma|_X = \{x \mapsto x\sigma \mid x \in \mathcal{D}om(\sigma) \cap X\}$.

An (oriented) *conditional rewrite rule* over a signature $\mathcal{F}$ is a triple (l, r, c), denoted by $l \to r \Leftarrow c$, such that the *left-hand side* l is a non-variable term in $\mathcal{T}(\mathcal{F}, \mathcal{V})$, the *right-hand side* r is a term in $\mathcal{T}(\mathcal{F}, \mathcal{V})$, and the *conditional part* c is a sequence $s_1 \twoheadrightarrow t_1; \ldots; s_k \twoheadrightarrow t_k$ of term pairs ($k \geq 0$) where all of $s_1, t_1, \ldots, s_k, t_k$ are terms in $T(\mathcal{F}, \mathcal{V})$. 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 $l \to r$. We sometimes attach a unique label ρ to the conditional rewrite rule $l \to r \Leftarrow c$ by denoting $\rho : l \to r \Leftarrow c$, and we use the label to refer to the rewrite rule.

An *(oriented) conditional term rewriting system* (CTRS) over a signature $\mathcal{F}$ is a set of conditional rules over $\mathcal{F}$. A CTRS is called an (unconditional) *term rewriting system* (TRS) if every rule $l \to r \Leftarrow c$ in the CTRS is unconditional and satisfies $\mathcal{V}ar(l) \supseteq \mathcal{V}ar(r)$. The *reduction relation* of a CTRS $\mathcal{R}$ is defined as $\to_{\mathcal{R}} = \bigcup_{n\geq 0} \to_{(n),\mathcal{R}}$ where $\to_{(0),\mathcal{R}} = \emptyset$, and $\to_{(i+1),\mathcal{R}} = \{(C[l\sigma]_p, C[r\sigma]_p) \mid \rho : l \to r \Leftarrow s_1 \twoheadrightarrow t_1; \ldots; s_k \twoheadrightarrow t_k \in \mathcal{R},\ s_1\sigma \to^*_{(i),\mathcal{R}} t_1\sigma,\ \ldots,\ s_k\sigma \to^*_{(i),\mathcal{R}} t_k\sigma\}$ for $i \geq 0$. To specify the applied rule ρ and the position p where ρ is applied, we may write $\to_{p,\rho}$ or $\to_{p,\mathcal{R}}$ instead of $\to_{\mathcal{R}}$. Moreover, we may write $\to_{>\varepsilon,\mathcal{R}}$ instead of $\to_{p,\mathcal{R}}$ if $p > \varepsilon$. The *parallel reduction* $\rightrightarrows_{\mathcal{R}}$ is defined as follows: $\rightrightarrows_{\mathcal{R}} = \{(C[s_1, \ldots, s_n]_{p_1,\ldots,p_n}, C[t_1, \ldots, t_n]_{p_1,\ldots,p_n}) \mid s_1 \to_{\mathcal{R}} t_1,\ \ldots,\ s_n \to_{\mathcal{R}} t_n\}$. To specify the positions $p_1, \ldots, p_n$ in the definition, we may write $\rightrightarrows_{\{p_1,\ldots,p_n\},\mathcal{R}}$ instead of $\rightrightarrows_{\mathcal{R}}$, and we may write $\rightrightarrows_{>\varepsilon,\mathcal{R}}$ instead of $\rightrightarrows_{\mathcal{R}}$ if $p_i > \varepsilon$ for all $1 \leq i \leq n$. We denote n-step parallel reduction by $\rightrightarrows^n_{\mathcal{R}}$, and for $m \geq n$, we may write $\rightrightarrows^{\leq m}_{\mathcal{R}}$ instead of $\rightrightarrows^n_{\mathcal{R}}$.

A conditional rewrite rule $l \to r \Leftarrow c$ is called *left-linear* if l is linear, *right-linear* if r is linear, *non-erasing* if $\mathcal{V}ar(l) \subseteq \mathcal{V}ar(r)$, 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 $\rho : l \to r \Leftarrow s_1 \twoheadrightarrow t_1; \ldots ; s_k \twoheadrightarrow t_k$ is called *normal* if $\mathcal{V}ar(s_1, \ldots, s_k) \subseteq \mathcal{V}ar(l)$ and $t_1, \ldots, t_n$ are normal forms w.r.t. the *underlying unconditional system* $\mathcal{R}_u = \{l \to r \mid l \to r \Leftarrow c \in \mathcal{R}\}$. A CTRS is called *normal* (or a *normal CTRS*) if every rewrite rule of the CTRS is normal. Note that we consider *1-CTRSs* (i.e., $\mathcal{V}ar(l) \supseteq \mathcal{V}ar(r)$ for $l \to r \Leftarrow c$). 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 $\mathcal{F}$. The sets of *defined symbols* and *constructors* of $\mathcal{R}$ are denoted by $\mathcal{D}_{\mathcal{R}}$ and $\mathcal{C}_{\mathcal{R}}$, respectively: $\mathcal{D}_{\mathcal{R}} = \{\mathtt{root}(l) \mid l \to r \Leftarrow c \in \mathcal{R}\}$ and $\mathcal{C}_{\mathcal{R}} = \mathcal{F} \setminus \mathcal{D}_{\mathcal{R}}$. Terms in $T(\mathcal{C}_{\mathcal{R}}, \mathcal{V})$ 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 $\mathcal{F}_1$ and $\mathcal{F}_2$, respectively, ϕ be an *initialization* (total) mapping from $T(\mathcal{F}_1, \mathcal{V})$ to $T(\mathcal{F}_2, \mathcal{V})$, and ψ be a partial inverse of ϕ, a *backtranslation* mapping from $T(\mathcal{F}_2, \mathcal{V})$ to $T(\mathcal{F}_1, \mathcal{V})$ such that $\psi(\phi(t_1)) = t_1$ for any term $t_1 \in T(\mathcal{F}_1, \mathcal{V})$. We say that

- $\mathcal{R}_2$ is *sound for* (*reduction of*) $\mathcal{R}_1$ *w.r.t.* (ϕ, ψ) if, for any term $t_1 \in T(\mathcal{F}_1, \mathcal{V})$ and for any term $t_2 \in T(\mathcal{F}_2, \mathcal{V})$, $\phi(t_1) \to^*_{\mathcal{R}_2} t_2$ implies $t_1 \to^*_{\mathcal{R}_1} \psi(t_2)$ whenever $\psi(t_2)$ is defined, and
- $\mathcal{R}_2$ is *complete for* (*reduction of*) $\mathcal{R}_1$ *w.r.t.* (ϕ, ψ) if, for all terms $t_1, t_1' \in T(\mathcal{F}_1, \mathcal{V})$, $t_1 \to^*_{\mathcal{R}_1} t_1'$ implies $\phi(t_1) \to^*_{\mathcal{R}_2} \phi(t_1)$.

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 $\boldsymbol{t}_{i..j}$. Moreover, for the application of a mapping f to $\boldsymbol{t}_{i..j}$, we denote $f(t_i), \ldots, f(t_j)$ by $f(\boldsymbol{t}_{i..j})$, e.g., for a substitution θ, we denote $t_i\theta, \ldots, t_j\theta$ by $\theta(\boldsymbol{t}_{i..j})$.

3.1 Simultaneous Unraveling

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

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

$$\mathbb{U}(\rho : l \to r \Leftarrow s \twoheadrightarrow t) = \{\ l \to U_\rho(s, \boldsymbol{x}_{1..n}),\ \ U_\rho(t, \boldsymbol{x}_{1..n}) \to r\ \}$$

where $\{x_1, \ldots, x_n\} = \mathcal{V}ar(l)$ and U_ρ is a fresh $n+1$-ary function symbol, called a *U symbol*. Note that for every unconditional rule $l \to r \in \mathcal{R}$, $\mathbb{U}(l \to r) = \{l \to r\}$. $\mathbb{U}$ is straightforwardly extended to normal CTRSs: $\mathbb{U}(\mathcal{R}) = \bigcup_{\rho\in\mathcal{R}} \mathbb{U}(\rho)$. We abuse $\mathbb{U}$ to represent the extended signature of $\mathcal{F}$: $\mathbb{U}_{\mathcal{R}}(\mathcal{F}) = \mathcal{F} \cup \{U_\rho \mid \rho : l \to r \Leftarrow s \twoheadrightarrow t \in \mathcal{R}\}$. We omit $\mathcal{R}$ from $\mathbb{U}_{\mathcal{R}}(\mathcal{F})$. Note that $\mathbb{U}(\mathcal{R})$ is a TRS over $\mathbb{U}(\mathcal{F})$. We say that $\mathbb{U}$ (and also $\mathbb{U}(\mathcal{R})$) is *sound* (*complete*) *for* $\mathcal{R}$ if $\mathbb{U}(\mathcal{R})$ is sound (complete) for $\mathcal{R}$ w.r.t. $(\mathtt{id}, \mathtt{id})$, where $\mathtt{id}$ is the identity mapping for $T(\mathcal{F},\mathcal{V})$.

Note that $\mathbb{U}$ is complete for all normal CTRSs [15, 18].

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

$$\mathcal{R}_1 = \{\ \mathsf{e}(0) \to \mathsf{true}, \quad \mathsf{e}(\mathsf{s}(x)) \to \mathsf{true} \Leftarrow \mathsf{e}(x) \twoheadrightarrow \mathsf{false}, \quad \mathsf{e}(\mathsf{s}(x)) \to \mathsf{false} \Leftarrow \mathsf{e}(x) \twoheadrightarrow \mathsf{true}\ \}$$

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

$$\mathbb{U}(\mathcal{R}_1) = \left\{ \begin{array}{lll} \mathsf{e}(0) \to \mathsf{true}, & \mathsf{e}(\mathsf{s}(x)) \to \mathsf{u}_1(\mathsf{e}(x), x), & \mathsf{u}_1(\mathsf{false}, x) \to \mathsf{true}, \\ & \mathsf{e}(\mathsf{s}(x)) \to \mathsf{u}_2(\mathsf{e}(x), x), & \mathsf{u}_2(\mathsf{true}, x) \to \mathsf{false} \end{array} \right\}$$

$\mathbb{U}(\mathcal{R}_1)$ is not confluent, while $\mathcal{R}_1$ is confluent. This means that $\mathbb{U}$ does not always preserve confluence of CTRSs. The reason why $\mathbb{U}$ 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 $\mathsf{e}(\mathsf{s}(0)) \to_{\mathbb{U}(\mathcal{R}_1)} \mathsf{u}_1(\mathsf{e}(0), 0) \to_{\mathbb{U}(\mathcal{R}_1)} \mathsf{u}_1(\mathsf{true}, 0)$. To reduce $\mathsf{e}(\mathsf{s}(0))$ to false, we should have applied $\mathsf{e}(\mathsf{s}(x)) \to \mathsf{u}_2(\mathsf{e}(x), x) \in \mathbb{U}(\mathcal{R}_1)$ to the initial term. However, we applied another wrong rule, u_1 expects $\mathsf{e}(0)$ to be reduced to $\mathtt{true}$ (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], $\mathbb{U}$ is not sound for every normal CTRS (see also [19, Example 7.2.14]). For some classes of normal CTRSs, $\mathbb{U}$ is sound (cf. [9, 17]).

▶ **Theorem 3** ([9]). $\mathbb{U}$ *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 $\bar{\mathsf{f}}$ with the arity $n+m$, and
- a fresh constant $\bot$ and a fresh unary symbol $\langle\cdot\rangle$ are introduced.

We denote the extended signature by $\overline{\mathcal{F}}$: $\overline{\mathcal{F}} = \{\mathsf{c} \mid \mathsf{c} \in \mathcal{C}_{\mathcal{R}}\} \cup \{\bar{\mathsf{f}} \mid \mathsf{f} \in \mathcal{D}_{\mathcal{R}}\} \cup \{\bot, \langle\cdot\rangle\}$. We introduce a mapping $\mathtt{ext}(\cdot)$ to extend the arguments of defined symbols in a term as follows: $\mathtt{ext}(x) = x$ for $x \in \mathcal{V}$; $\mathtt{ext}(\mathsf{c}(\boldsymbol{t}_{1..n})) = \mathsf{c}(\mathtt{ext}(\boldsymbol{t}_{1..n}))$ for $\mathsf{c}/n \in \mathcal{C}_{\mathcal{R}}$; $\mathtt{ext}(\mathsf{f}(\boldsymbol{t}_{1..n})) = \bar{\mathsf{f}}(\mathtt{ext}(\boldsymbol{t}_{1..n}), \boldsymbol{z}_{1..m})$ for $\mathsf{f}/n \in \mathcal{D}_{\mathcal{R}}$, where f has m conditional rules in $\mathcal{R}$, $\mathtt{arity}_{\overline{\mathcal{F}}}(\bar{\mathsf{f}}) = n+m$, and $z_1, \ldots, z_m$ are fresh variables. The extended arguments of $\bar{\mathsf{f}}$ 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: $(x)^\bot = x$ for $x \in \mathcal{V}$; $(\mathsf{c}(\boldsymbol{t}_{1..n}))^\bot = \mathsf{c}((\boldsymbol{t}_{1..n})^\bot)$ for $\mathsf{c} \in \mathcal{C}_{\mathcal{R}}$; $(\bar{\mathsf{f}}(\boldsymbol{t}_{1..n}, \boldsymbol{u}_{1..m}))^\bot = \bar{\mathsf{f}}((\boldsymbol{t}_{1..n})^\bot, \bot, \ldots, \bot)$ for $\mathsf{f} \in \mathcal{D}_{\mathcal{R}}$; $(\langle t\rangle)^\bot = \langle (t)^\bot \rangle$; $(\bot)^\bot = \bot$. 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 $\bar{\cdot}$ from $T(\mathcal{F},\mathcal{V})$ to $T(\overline{\mathcal{F}},\mathcal{V})$ as $\bar{t} = (\mathtt{ext}(t))^\bot$.

The SR transformation [21] is defined as follows.

▶ **Definition 4** ($\mathbb{SR}$)**.** Let $\mathsf{f}/n \in \mathcal{D}_\mathcal{R}$ that has m conditional rules in $\mathcal{R}$ (i.e., $\bar{\mathsf{f}}/(n+m) \in \overline{\mathcal{F}}$). Then, $\mathbb{SR}(\mathsf{f}(\boldsymbol{w}_{1..n}) \to r) = \{ \bar{\mathsf{f}}(\mathtt{ext}(\boldsymbol{w}_{1..n}), \boldsymbol{z}_{1..m}) \to \langle \overline{r} \rangle \}$ and, for the i-th conditional rule of f,

$$\mathbb{SR}(\mathsf{f}(\boldsymbol{w}_{1..n}) \to r_i \Leftarrow s_i \twoheadrightarrow t_i) = \left\{ \begin{array}{l} \bar{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1},\ \bot,\ \boldsymbol{z}_{i+1..m}) \to \bar{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \langle s_i \rangle, \boldsymbol{z}_{i+1..m}), \\ \bar{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \langle t_i \rangle, \boldsymbol{z}_{i+1..m}) \to \langle \overline{r_i} \rangle \end{array} \right\}$$

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

$$\begin{array}{rl} \mathcal{R}_{aux} = & \{ \ \langle\langle x \rangle\rangle \to \langle x \rangle \ \} \cup \{ \ \mathsf{c}(\boldsymbol{x}_{1..i-1}, \langle x_i \rangle, \boldsymbol{x}_{i+1..n}) \to \langle \mathsf{c}(\boldsymbol{x}_{1..n}) \rangle \mid \mathsf{c}/n \in \mathcal{C}_\mathcal{R},\ 1 \le i \le n \ \} \\ & \cup \{ \ \bar{\mathsf{f}}(\boldsymbol{x}_{1..i-1}, \langle x_i \rangle, \boldsymbol{x}_{i+1..n}, \boldsymbol{z}_{1..m}) \to \langle \bar{\mathsf{f}}(\boldsymbol{x}_{1..n}, \bot, \ldots, \bot) \rangle \mid \mathsf{f}/n \in \mathcal{D}_\mathcal{R},\ 1 \le i \le n \ \} \end{array}$$

where $z_1, \ldots, z_m$ are fresh variables. The transformation $\mathbb{SR}$ is defined as follows: $\mathbb{SR}(\mathcal{R}) = \bigcup_{\rho \in \mathcal{R}} \mathbb{SR}(\rho) \cup \mathcal{R}_{aux}$. Note that $\mathbb{SR}(\mathcal{R})$ is a TRS over $\overline{\mathcal{F}}$. Note also that $\mathcal{R}_{aux}$ is linear. The backtranslation mapping $\widehat{\cdot}$ for $\bar{\cdot}$ is defined as follows: $\widehat{x} = x$ for $x \in \mathcal{V}$; $\widehat{\mathsf{c}(\boldsymbol{t}_{1..n})} = \mathsf{c}(\widehat{\boldsymbol{t}_{1..n}})$ for $\mathsf{c}/n \in \mathcal{C}_\mathcal{R}$; $\widehat{\bar{\mathsf{f}}(\boldsymbol{t}_{1..n}, \boldsymbol{u}_{1..m})} = \mathsf{f}(\widehat{\boldsymbol{t}_{1..n}})$ for $\mathsf{f}/n \in \mathcal{D}_\mathcal{R}$; $\widehat{\langle t \rangle} = \widehat{t}$; $\widehat{\bot} = \bot$. Note that $\widehat{\cdot}$ is a total function. A term t in $T(\overline{\mathcal{F}}, \mathcal{V})$ is called *reachable* if there exists a term $s \in T(\mathcal{F}, \mathcal{V})$ such that $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$. We say that $\mathbb{SR}$ (and also $\mathbb{SR}(\mathcal{R})$) is *sound* (*complete*) *for* $\mathcal{R}$ if $\mathbb{SR}(\mathcal{R})$ is sound (complete) for $\mathcal{R}$ w.r.t. $(\bar{\cdot}, \widehat{\cdot})$.

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

To evaluate the condition of the i-th conditional rule $\mathsf{f}(\boldsymbol{w}_{1..n}) \to r_i \Leftarrow s_i \twoheadrightarrow t_i$, the i-th conditional rule is transformed into the two unconditional rules: the first one starts to evaluate the condition (an instance of s_i), and the second examines whether the condition holds. The first rule $\langle\langle x \rangle\rangle \to \langle x \rangle$ in $\mathcal{R}_{aux}$ removes the nest of $\langle\cdot\rangle$, the second rule $\mathsf{c}(\boldsymbol{x}_{1..i-1}, \langle x_i \rangle, \boldsymbol{x}_{i+1..n}) \to \langle \mathsf{c}(\boldsymbol{x}_{1..n}) \rangle$ is used for shifting $\langle\cdot\rangle$ upward, and the third rule $\bar{\mathsf{f}}(\boldsymbol{x}_{1..i-1}, \langle x_i \rangle, \boldsymbol{x}_{i+1..n}, \boldsymbol{z}_{1..m}) \to \langle \bar{\mathsf{f}}(\boldsymbol{x}_{1..n}, \bot, \ldots, \bot) \rangle$ is used for both shifting $\langle\cdot\rangle$ upward and resetting the evaluation of conditions at the extended arguments of $\bar{\mathsf{f}}$. The unary symbol $\langle\cdot\rangle$ and its rules in $\mathcal{R}_{aux}$ are introduced to preserve confluence of normal CTRSs on reachable terms (see [21] for the detail of the role of $\langle\cdot\rangle$ and its rules).

▶ **Example 5.** Consider $\mathcal{R}_1$ in Example 2 again. $\mathcal{R}_1$ is transformed by $\mathbb{SR}$ as follows:

$$\mathbb{SR}(\mathcal{R}_1) = \left\{ \begin{array}{ccc} \bar{\mathsf{e}}(0, z_1, z_2) \to \langle \mathsf{true} \rangle, & & \\ \bar{\mathsf{e}}(\mathsf{s}(x), \bot, z_2) \to \bar{\mathsf{e}}(\mathsf{s}(x), \langle \bar{\mathsf{e}}(x, \bot, \bot) \rangle, z_2), & & \bar{\mathsf{e}}(\mathsf{s}(x), \langle \mathsf{false} \rangle, z_2) \to \langle \mathsf{true} \rangle, \\ \bar{\mathsf{e}}(\mathsf{s}(x), z_1, \bot) \to \bar{\mathsf{e}}(\mathsf{s}(x), z_1, \langle \bar{\mathsf{e}}(x, \bot, \bot) \rangle), & & \bar{\mathsf{e}}(\mathsf{s}(x), z_1, \langle \mathsf{true} \rangle) \to \langle \mathsf{false} \rangle, \\ \langle\langle x \rangle\rangle \to \langle x \rangle, & \mathsf{s}(\langle x \rangle) \to \langle \mathsf{s}(x) \rangle, & \bar{\mathsf{e}}(\langle x \rangle, z_1, z_2) \to \langle \bar{\mathsf{e}}(x, \bot, \bot) \rangle \end{array} \right\}$$

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

$$\bar{\mathsf{e}}(\mathsf{s}(0), \bot, \bot) \to_{\mathbb{SR}(\mathcal{R}_1)} \bar{\mathsf{e}}(\mathsf{s}(0), \langle \bar{\mathsf{e}}(0, \bot, \bot) \rangle, \bot) \to^*_{\mathbb{SR}(\mathcal{R}_1)} \bar{\mathsf{e}}(\mathsf{s}(0), \langle \mathsf{true} \rangle, \bot)$$

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

$$\cdots \to_{\mathbb{SR}(\mathcal{R}_1)} \bar{\mathsf{e}}(\mathsf{s}(0), \langle \mathsf{true} \rangle, \langle \bar{\mathsf{e}}(0, \bot, \bot) \rangle) \to^*_{\mathbb{SR}(\mathcal{R}_1)} \bar{\mathsf{e}}(\mathsf{s}(0), \langle \mathsf{true} \rangle, \langle \mathsf{true} \rangle) \to_{\mathbb{SR}(\mathcal{R}_1)} \langle \mathsf{false} \rangle$$

In the case of $\mathbb{U}$, to reach false, we need to backtrack from the undesired normal form $\mathsf{u}_1(\mathsf{true}, 0)$ (see Example 2), but in the case of $\mathbb{SR}$, we do not have to backtrack — choosing an arbitrary redex from reducible terms is sufficient to reach a desired normal form since $\mathbb{SR}(\mathcal{R}_1)$ is confluent on reachable terms.

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

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

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

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

In this section, by using $\mathbb{U}$, we show that $\mathbb{SR}$ 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: $(\theta)^\perp = \{x \mapsto (x\theta)^\perp \mid x \in \mathcal{D}om(\theta)\}$ for a substitution θ such that $\mathcal{R}an(\theta) \subseteq T(\overline{\mathcal{F}}, \mathcal{V})$. The mapping $(\cdot)^\perp$ has the following properties which are trivial by definition.

▶ **Proposition 8.** *Let $\mathcal{R}$ be a normal CTRS. Then, all of the following hold:*

- *For any term $s \in T(\mathcal{F}, \mathcal{V})$, $\overline{s} = (\overline{s})^\perp$.*
- *For any term $t \in T(\overline{\mathcal{F}}, \mathcal{V})$, $(t\theta)^\perp = (t)^\perp(\theta)^\perp$ for any substitution $\theta \in \mathcal{S}ub(\overline{\mathcal{F}}, \mathcal{V})$ 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 $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$, the mappings $\bar{\cdot}$ and $\hat{\cdot}$ often prevent us from using induction because $\hat{\cdot}$ removes all occurrences of $\langle \cdot \rangle$ from terms. For this reason, using the mapping $(\cdot)^\perp$ instead of $\langle \bar{\cdot} \rangle$ is a breakthrough to prove our main theorem.

Next, we observe reduction sequences $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$ with $s \in T(\mathcal{F}, \mathcal{V})$ and $t \in T(\overline{\mathcal{F}}, \mathcal{V})$. The main feature of $\mathbb{SR}$ is to evaluate two or more conditions in parallel. However, to get $\widehat{t}$, it suffices to evaluate successfully at most one condition in each parallel evaluation of conditions. This means that every term appearing in $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$, which is rooted by a defined symbol $\overline{\mathsf{f}}$, is of the form $\overline{\mathsf{f}}(\boldsymbol{t}_{1..n}, \boldsymbol{u}_{1..m})$ where $\mathtt{arity}_{\mathcal{F}}(\mathsf{f}) = n$ and at most one of $u_1, \ldots, u_m$ is rooted by $\langle \cdot \rangle$ (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 $\mathbb{U}(\mathcal{F})$: for a term $\overline{\mathsf{f}}(\boldsymbol{t}_{1..n}, \boldsymbol{u}_{1..m})$, 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 $U_{\mathsf{f},i}$ 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 $T(\overline{\mathcal{F}}, \mathcal{V})$ to a term in $T(\mathbb{U}(\mathcal{F}), \mathcal{V})$.

▶ **Definition 9.** Let $\mathcal{R}$ be a normal CTRS. Then, we define a mapping Φ from reachable terms in $T(\overline{\mathcal{F}}, \mathcal{V})$ to $T(\mathbb{U}(\mathcal{F}), \mathcal{V})$ as follows:

- $\Phi(x) = x$ for $x \in \mathcal{V}$,
- $\Phi(\mathsf{c}(\boldsymbol{t}_{1..n})) = \mathsf{c}(\Phi(\boldsymbol{t}_{1..n}))$ for $\mathsf{c}/n \in \mathcal{C}_{\mathcal{R}}$,
- $\Phi(\overline{\mathsf{f}}(\boldsymbol{t}_{1..n}, \perp, \ldots, \perp)) = \mathsf{f}(\Phi(\boldsymbol{t}_{1..n}))$ for $\mathsf{f}/n \in \mathcal{D}_{\mathcal{R}}$,

[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.

$$
\begin{tikzcd}[column sep=small]
\langle \bar{\mathsf{e}}(\mathsf{s}(0),\bot,\bot)\rangle \arrow[r, dotted, no head] \arrow[d] & \langle \bar{\mathsf{e}}(\mathsf{s}(0),\bot,\bot)\rangle \arrow[r, dotted, no head] \arrow[d] & \mathsf{e}(\mathsf{s}(0)) \arrow[d] \\
\langle \bar{\mathsf{e}}(\mathsf{s}(0),\bot,\langle \bar{\mathsf{e}}(0,\bot,\bot)\rangle)\rangle \arrow[r, dotted, no head] \arrow[d] & \langle \bar{\mathsf{e}}(\mathsf{s}(0),\bot,\langle \bar{\mathsf{e}}(0,\bot,\bot)\rangle)\rangle \arrow[r, dotted, no head] \arrow[dd, "2"] & \mathsf{u}_2(\mathsf{e}(0),\mathsf{s}(0)) \arrow[dd] \\
\langle \bar{\mathsf{e}}(\mathsf{s}(0),\langle \bar{\mathsf{e}}(0,\bot,\bot)\rangle,\langle \bar{\mathsf{e}}(0,\bot,\bot)\rangle)\rangle \arrow[ur, dotted, no head] \arrow[d, "4"] & & \\
\langle \bar{\mathsf{e}}(\mathsf{s}(0),\langle \mathsf{true}\rangle,\langle \mathsf{true}\rangle)\rangle \arrow[r, dotted, no head] \arrow[d, "2"] & \langle \bar{\mathsf{e}}(\mathsf{s}(0),\bot,\langle \mathsf{true}\rangle)\rangle \arrow[r, dotted, no head] \arrow[d, "2"] & \mathsf{u}_2(\mathsf{true},\mathsf{s}(0)) \arrow[d] \\
\langle \mathsf{false}\rangle \arrow[r, dotted, no head] & \langle \mathsf{false}\rangle \arrow[r, dotted, no head] & \mathsf{false}
\end{tikzcd}
$$

(a) on $\mathbb{SR}(\mathcal{R}_1)$ (b) on $\mathbb{SR}(\mathcal{R}_1)$ with single evaluation (c) on $\mathbb{U}(\mathcal{R}_1)$

Figure 1 The correspondence of derivations between $\mathbb{SR}(\mathcal{R}_1)$ and $\mathbb{U}(\mathcal{R}_1)$.

- $\Phi(\bar{\mathsf{f}}(\theta(\mathtt{ext}(\boldsymbol{w}_{1..n})),\overbrace{\bot,\ldots,\bot}^{i-1},\langle u_i\rangle,\bot,\ldots,\bot)) = U_{\mathsf{f},i}(\Phi(u_i),\Phi(\theta(\boldsymbol{x}_{1..j})))$ for $\mathsf{f}/n \in \mathcal{D}_\mathcal{R}$, where the i-th conditional rule of f is $\rho : \mathsf{f}(\boldsymbol{w}_{1..n}) \to r_i \Leftarrow s_i \twoheadrightarrow t_i \in \mathcal{R}$, $U_{\mathsf{f},i}(t_i,\boldsymbol{x}_{1..j}) \to r_i \in \mathbb{U}(\rho)$ with $\mathcal{V}ar(\boldsymbol{w}_{1..n}) = \{\boldsymbol{x}_{1..j}\}$, $\Phi(\theta(\mathtt{ext}(w_j)))$ is defined for all $1 \le j \le n$, $\Phi(x\theta)$ is defined for any variable $x \in \mathcal{D}om(\theta)$, and $\Phi((\langle\overline{s_i}\rangle\theta)^\bot) \to^*_{\mathbb{U}(\mathcal{R})} \Phi((\langle u_i\rangle)^\bot)$, and
- $\Phi(\langle t\rangle) = \Phi(t)$.

We say that $\Phi(\theta)$ is *defined* if $\Phi(x\theta)$ is defined for any variable $x \in \mathcal{D}om(\theta)$, and we denote by $\Phi(\theta)$ the substitution $\{x \mapsto \Phi(x\theta) \mid x \in \mathcal{D}om(\theta)\}$.

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 $\Phi((t)^\bot) = \widehat{t}$ for a reachable term $t \in T(\overline{\mathcal{F}},\mathcal{V})$. We will use this property without notice.

▶ **Example 10.** Consider $\mathbb{U}(\mathcal{R}_1)$ and $\mathbb{SR}(\mathcal{R}_1)$ in Examples 2 and 5, respectively, again. We have that $\Phi(\langle\bar{\mathsf{e}}(\mathsf{s}(0),\bot,\bot)\rangle) = \mathsf{e}(\mathsf{s}(0))$, $\Phi(\langle\bar{\mathsf{e}}(\mathsf{s}(0),\bot,\langle\bar{\mathsf{e}}(0,\bot,\bot)\rangle)\rangle) = \mathsf{u}_2(\mathsf{e}(0),\mathsf{s}(0))$, and $\Phi(\langle\bar{\mathsf{e}}(\mathsf{s}(0),\bot,\langle\mathsf{false}\rangle)\rangle) = \mathsf{u}_2(\mathsf{false},\mathsf{s}(0))$. On the other hand, Φ is not defined for the term $\langle\bar{\mathsf{e}}(\mathsf{s}(0),\langle\bar{\mathsf{e}}(0,\bot,\bot)\rangle,\langle\bar{\mathsf{e}}(0,\bot,\bot)\rangle)\rangle$ 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:

$$\mathcal{R}_2 = \left\{ \begin{array}{llll} \mathsf{f}(x) \to x \Leftarrow x \twoheadrightarrow \mathsf{c}, & \mathsf{g}(x,x) \to \mathsf{h}(x,x), & \mathsf{h}(\mathsf{f}(\mathsf{d}),x) \to x, & \\ \mathsf{a} \to \mathsf{c}, & \mathsf{a} \to \mathsf{d}, & \mathsf{b} \to \mathsf{c}, & \mathsf{b} \to \mathsf{d} \end{array} \right\}$$

$\mathcal{R}_2$ is transformed by $\mathbb{U}$ and $\mathbb{SR}$, respectively, as follows:

$$\mathbb{U}(\mathcal{R}_2) = \{\ \mathsf{f}(x) \to \mathsf{u}_3(x,x),\quad \mathsf{u}_3(\mathsf{c},x) \to x,\quad \mathsf{g}(x,x) \to \mathsf{h}(x,x),\quad \mathsf{h}(\mathsf{f}(\mathsf{d}),x) \to x,\quad \ldots\ \}$$

$$\mathbb{SR}(\mathcal{R}_2) = \left\{ \begin{array}{llll} \bar{\mathsf{f}}(x,\bot) \to \bar{\mathsf{f}}(x,\langle x\rangle), & \bar{\mathsf{f}}(x,\langle\mathsf{c}\rangle) \to \langle x\rangle, & \bar{\mathsf{a}} \to \langle\mathsf{c}\rangle, & \bar{\mathsf{a}} \to \langle\mathsf{d}\rangle, \\ \bar{\mathsf{g}}(x,x) \to \langle\bar{\mathsf{h}}(x,x)\rangle, & \bar{\mathsf{h}}(\bar{\mathsf{f}}(\mathsf{d},z_1),x) \to \langle x\rangle, & \bar{\mathsf{b}} \to \langle\mathsf{c}\rangle, & \bar{\mathsf{b}} \to \langle\mathsf{d}\rangle, \\ \langle\langle x\rangle\rangle \to \langle x\rangle, & \bar{\mathsf{g}}(\langle x\rangle,y) \to \langle\bar{\mathsf{g}}(x,y)\rangle, & \bar{\mathsf{h}}(\langle x\rangle,y) \to \langle\bar{\mathsf{h}}(x,y)\rangle, & \\ \bar{\mathsf{f}}(\langle x\rangle,z_1) \to \langle\bar{\mathsf{f}}(x,\bot)\rangle, & \bar{\mathsf{g}}(x,\langle y\rangle) \to \langle\bar{\mathsf{g}}(x,y)\rangle, & \bar{\mathsf{h}}(x,\langle y\rangle) \to \langle\bar{\mathsf{h}}(x,y)\rangle & \end{array} \right\}$$

We have the following derivations:

$$\begin{array}{ll} \langle\bar{\mathsf{g}}(\bar{\mathsf{f}}(\bar{\mathsf{a}},\bot),\bar{\mathsf{f}}(\bar{\mathsf{b}},\bot))\rangle & \mathsf{g}(\mathsf{f}(\mathsf{a}),\mathsf{f}(\mathsf{b})) \\ \quad \to^*_{\mathbb{SR}(\mathcal{R}_2)} \langle\bar{\mathsf{g}}(\bar{\mathsf{f}}(\langle\mathsf{d}\rangle,\langle\mathsf{c}\rangle),\bar{\mathsf{f}}(\langle\mathsf{d}\rangle,\langle\mathsf{c}\rangle))\rangle & \quad \to^*_{\mathbb{U}(\mathcal{R}_2)} \mathsf{g}(\mathsf{u}_3(\mathsf{c},\mathsf{d}),\mathsf{u}_3(\mathsf{c},\mathsf{d})) \\ \quad \to^*_{\mathbb{SR}(\mathcal{R}_2)} \langle\bar{\mathsf{f}}(\langle\mathsf{d}\rangle,\langle\mathsf{c}\rangle)\rangle \to^*_{\mathbb{SR}(\mathcal{R}_2)} \langle\mathsf{d}\rangle & \quad \not\to_{\mathbb{U}(\mathcal{R}_2)} \mathsf{u}_3(\mathsf{c},\mathsf{d}) \to_{\mathbb{U}(\mathcal{R}_2)} \mathsf{d} \end{array}$$

Neither $\mathbb{U}(\mathcal{R}_2)$ nor $\mathcal{R}_2$ can simulate the derivation $\langle\bar{\mathsf{g}}(\bar{\mathsf{f}}(\bar{\mathsf{a}},\bot),\bar{\mathsf{f}}(\bar{\mathsf{b}},\bot))\rangle \to^*_{\mathbb{SR}(\mathcal{R}_2)} \langle\mathsf{d}\rangle$, and thus, $\mathbb{SR}$ is not sound for $\mathcal{R}_2$.

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 $\mathbb{SR}$, which are trivial by definition.

▶ **Proposition 12.** *If $\mathcal{R}$ is weakly left-linear, then $\mathbb{SR}(\mathcal{R})$ is weakly left-linear, especially, for transformed rewrite rules $\overline{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \bot, \boldsymbol{z}_{i+1..m}) \to \overline{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \langle s_i \rangle, \boldsymbol{z}_{i+1..m})$ and $\overline{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \langle t_i \rangle, \boldsymbol{z}_{i+1..m}) \to \langle \overline{r_i} \rangle$ in $\mathbb{SR}(\mathsf{f}(\boldsymbol{w}_{1..n}) \to r_i \Leftarrow s_i \twoheadrightarrow t_i)$,*
- *$w'_1, \ldots, w'_n$ are linear without any shared variable, and*
- *$\overline{\mathsf{f}}(\boldsymbol{w'}_{1..n}, \boldsymbol{z}_{1..i-1}, \langle t_i \rangle, \boldsymbol{z}_{i+1..m}) \to \langle \overline{r_i} \rangle$ is left-linear.*

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

▶ **Lemma 13.** *Let $\mathcal{R}$ be a weakly left-linear normal CTRS, s be a term in $T(\overline{\mathcal{F}}, \mathcal{V})$, t be a term in $T(\mathcal{C}_{\mathcal{R}} \cup \{\langle \cdot \rangle\}, \mathcal{V})$, and $\theta \in \mathcal{S}ub(\overline{\mathcal{F}}, \mathcal{V})$ with $\mathcal{D}om(\theta) \subseteq \mathcal{V}ar((t)^{\bot})$. If $s \rightrightarrows^k_{\mathbb{SR}(\mathcal{R})} t\theta$ ($k \geq 0$) and $\Phi(s)$ is defined, then there exists a substitution $\theta' \in \mathcal{S}ub(\overline{\mathcal{F}}, \mathcal{V})$ such that*
- *$\mathcal{D}om(\theta') = \mathcal{D}om(\theta)$,*
- *$\Phi(\theta')$ is defined,*
- *for any variable $x \in \mathcal{D}om(\theta')$, if t is linear w.r.t. x, then $x\theta' \rightrightarrows^{\leq k}_{\mathbb{SR}(\mathcal{R})} x\theta$ and $\Phi((x\theta')^{\bot}) \to^*_{\mathbb{U}(\mathcal{R})} \Phi((x\theta)^{\bot})$, and otherwise, $x\theta' = (x\theta)^{\bot}$, and*
- *$\Phi((s)^{\bot}) \to^*_{\mathbb{U}(\mathcal{R})} \Phi((t\theta')^{\bot})$.*

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 [2]). ◀

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

Next, we show that $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$ can be simulated by $\mathbb{U}(\mathcal{R})$.

▶ **Theorem 14.** *Let $\mathcal{R}$ be a weakly left-linear normal CTRS, s be a term in $T(\mathcal{F}, \mathcal{V})$, and t be a term in $T(\overline{\mathcal{F}}, \mathcal{V})$. If $\langle \overline{s} \rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$, then $s \to^*_{\mathbb{U}(\mathcal{R})} \widehat{t}$.*

Proof. By definition, $\Phi((\langle \overline{s} \rangle)^{\bot})$ $(= s)$ is defined, and thus, it follows from Lemma 13 that $\Phi((\langle \overline{s} \rangle)^{\bot}) \to^*_{\mathbb{U}(\mathcal{R})} \Phi((t)^{\bot})$. By definition, $\Phi((t)^{\bot}) = \widehat{t}$. Therefore, $s \to^*_{\mathbb{U}(\mathcal{R})} \widehat{t}$. ◀

Theorem 14 does not hold for all normal CTRSs.

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

$$\mathcal{R}_3 = \left\{ \begin{array}{llll} \mathsf{f}(x) \to \mathsf{c} \Leftarrow x \twoheadrightarrow \mathsf{c}, & \mathsf{f}(x) \to \mathsf{d} \Leftarrow x \twoheadrightarrow \mathsf{d}, & \mathsf{g}(x,x) \to \mathsf{h}(x,x), & \\ \mathsf{a} \to \mathsf{c}, & \mathsf{a} \to \mathsf{d}, & \mathsf{b} \to \mathsf{c}, & \mathsf{b} \to \mathsf{d} \end{array} \right\}$$

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

$$\mathbb{U}(\mathcal{R}_3) = \left\{ \begin{array}{rrr} \mathsf{f}(x) \to \mathsf{u}_5(x,x), & \mathsf{f}(x) \to \mathsf{u}_6(x,x), & \mathsf{g}(x,x) \to \mathsf{h}(x,x), \\ \mathsf{u}_5(\mathsf{c},x) \to \mathsf{c}, & \mathsf{u}_6(\mathsf{d},x) \to \mathsf{d}, & \ldots \end{array} \right\}$$

$$\mathbb{SR}(\mathcal{R}_3) = \left\{ \begin{array}{rrr} \overline{\mathsf{f}}(x,\bot,z_2) \to \overline{\mathsf{f}}(x,\langle x \rangle,z_2), & \overline{\mathsf{f}}(x,z_1,\bot) \to \overline{\mathsf{f}}(x,z_1,\langle x \rangle), & \overline{\mathsf{g}}(x,x) \to \langle \mathsf{h}(x,x) \rangle, \\ \overline{\mathsf{f}}(x,\langle \mathsf{c} \rangle,z_2) \to \langle \mathsf{c} \rangle, & \overline{\mathsf{f}}(x,z_1,\langle \mathsf{d} \rangle) \to \langle \mathsf{d} \rangle, & \overline{\mathsf{a}} \to \langle \mathsf{c} \rangle, \\ \overline{\mathsf{a}} \to \langle \mathsf{d} \rangle, & \overline{\mathsf{b}} \to \langle \mathsf{c} \rangle, & \overline{\mathsf{b}} \to \langle \mathsf{d} \rangle, \\ \langle \langle x \rangle \rangle \to \langle x \rangle, & \mathsf{h}(\langle x \rangle, y) \to \langle \mathsf{h}(x,y) \rangle, & \mathsf{h}(x,\langle y \rangle) \to \langle \mathsf{h}(x,y) \rangle, \\ \overline{\mathsf{f}}(\langle x \rangle,z_1,z_2) \to \langle \overline{\mathsf{f}}(x,\bot,\bot) \rangle, & \overline{\mathsf{g}}(\langle x \rangle, y) \to \langle \overline{\mathsf{g}}(x,y) \rangle, & \overline{\mathsf{g}}(x,\langle y \rangle) \to \langle \overline{\mathsf{g}}(x,y) \rangle \end{array} \right\}$$

[2] Available from `http://www.trs.cm.is.nagoya-u.ac.jp/~nishida/wpte14/`.

We have the following derivations:

$$\begin{array}{ll}
\langle \overline{\mathsf{g}}(\overline{\mathsf{f}}(\overline{\mathsf{a}}, \bot, \bot), \overline{\mathsf{f}}(\overline{\mathsf{b}}, \bot, \bot))\rangle & \mathsf{g}(\mathsf{f}(\mathsf{a}), \mathsf{f}(\mathsf{b})) \\
\quad \to^*_{\mathbb{SR}(\mathcal{R}_3)} \langle \mathsf{h}(\overline{\mathsf{f}}(\langle \mathsf{d}\rangle, \langle \mathsf{c}\rangle, \langle \mathsf{d}\rangle), \overline{\mathsf{f}}(\langle \mathsf{d}\rangle, \langle \mathsf{c}\rangle, \langle \mathsf{d}\rangle))\rangle & \quad \to^*_{\mathbb{U}(\mathcal{R}_3)} \mathsf{h}(\mathsf{u}_5(\mathsf{c}, \mathsf{d}), \mathsf{u}_5(\mathsf{c}, \mathsf{d})) \\
\quad \to_{\mathbb{SR}(\mathcal{R}_3)} \langle \mathsf{h}(\langle \mathsf{c}\rangle, \langle \mathsf{d}\rangle)\rangle \to^*_{\mathbb{SR}(\mathcal{R}_3)} \langle \mathsf{h}(\mathsf{c}, \mathsf{d})\rangle & \quad \not\to_{\mathbb{U}(\mathcal{R}_3)} \mathsf{h}(\mathsf{c}, \mathsf{d})
\end{array}$$

The other normal forms of $\mathsf{g}(\mathsf{f}(\mathsf{a}), \mathsf{f}(\mathsf{b}))$ on $\mathbb{U}(\mathcal{R}_3)$ are $\mathsf{h}(\mathsf{u}_5(\mathsf{c}, \mathsf{c}), \mathsf{u}_5(\mathsf{c}, \mathsf{c}))$, $\mathsf{h}(\mathsf{u}_5(\mathsf{d}, \mathsf{c}), \mathsf{u}_5(\mathsf{d}, \mathsf{c}))$, and $\mathsf{h}(\mathsf{u}_5(\mathsf{d}, \mathsf{d}), \mathsf{u}_5(\mathsf{d}, \mathsf{d}))$, but none of them corresponds to $\mathsf{h}(\mathsf{c}, \mathsf{d})$. For this reason, the derivation on $\mathbb{SR}(\mathcal{R}_3)$ cannot be simulated by $\mathbb{U}(\mathcal{R}_3)$. The derivation $\mathsf{g}(\mathsf{f}(\mathsf{a}), \mathsf{f}(\mathsf{b})) \to^* \mathsf{h}(\mathsf{c}, \mathsf{d})$ does not hold on $\mathcal{R}_3$, either, and thus, $\mathbb{SR}$ is not sound for $\mathcal{R}_3$. In addition, being a constructor system is not sufficient for soundness of $\mathbb{SR}$ since $\mathcal{R}_3$ is a constructor system.

We show the main result obtained by Theorem 14.

▶ **Theorem 16.** $\mathbb{SR}$ *is sound for weakly left-linear normal CTRS.*

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 $\langle \overline{s}\rangle \to^*_{\mathbb{SR}(\mathcal{R})} t$. Then, it follows from Theorem 14 that $s \to^*_{\mathbb{U}(\mathcal{R})} \widehat{t}$. Since $\widehat{t} \in T(\mathcal{F}, \mathcal{V})$ and $\mathbb{U}$ is sound for $\mathcal{R}$, we have that $s \to^*_{\mathcal{R}} \widehat{t}$. Therefore, $\mathbb{SR}$ 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 $\mathbb{U}$: non-erasingness and groundness of conditions. Non-erasingness of normal CTRSs is not sufficient for soundness since $\mathcal{R}_2$ is non-erasing but $\mathbb{SR}$ 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 $\mathcal{R}_3$:

$$\mathcal{R}_4 = \left\{ \begin{array}{llll} \mathsf{f}(\mathsf{a}) \to \mathsf{c} \Leftarrow \mathsf{a} \twoheadrightarrow \mathsf{c}, & \mathsf{f}(\mathsf{b}) \to \mathsf{d} \Leftarrow \mathsf{b} \twoheadrightarrow \mathsf{d}, & \mathsf{g}(x, x) \to \mathsf{h}(x, x), & \\ \quad \mathsf{a} \to \mathsf{c}, & \quad \mathsf{a} \to \mathsf{d}, & \qquad\qquad \mathsf{b} \to \mathsf{c}, & \mathsf{b} \to \mathsf{d} \end{array} \right\}$$

We have that $\overline{\mathsf{g}}(\overline{\mathsf{f}}(\overline{\mathsf{a}}, \bot, \bot), \overline{\mathsf{f}}(\overline{\mathsf{b}}, \bot, \bot)) \to^*_{\mathbb{SR}(\mathcal{R}_4)} \langle \mathsf{h}(\mathsf{c}, \mathsf{d})\rangle$, but $\mathsf{g}(\mathsf{f}(\mathsf{a}), \mathsf{f}(\mathsf{b})) \not\to^*_{\mathcal{R}_4} \mathsf{h}(\mathsf{c}, \mathsf{d})$. Therefore, $\mathbb{SR}$ is not sound for $\mathcal{R}_4$.

5 Conclusion

In this paper, by using the soundness of $\mathbb{U}$ 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

1. Sergio Antoy, Bernd Brassel, and Michael Hanus. Conditional narrowing without conditions. In *Proceedings of the 5th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming*, pp. 20–31, ACM, 2003.
2. Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.
3. Hubert Comon, Max Dauchet Reémi Gilleron, Florent Jacquemard, Denis Lugiez Christof, Löding, Sophie Tison, and Marc Tommasi. Tree automata techniques and applications. Available on: `http://www.grappa.univ-lille3.fr/tata`, 2007.
4. Francisco Durán, Salvador Lucas, José Meseguer, Claude Marché, and Xavier Urbain. Proving termination of membership equational programs. In *Proceedings of the 2004 ACM SIGPLAN Workshop on Partial Evaluation and Semantics-based Program Manipulation*, pp. 147–158, ACM, 2004.
5. Guillaume Feuillade and Thomas Genet. Reachability in conditional term rewriting systems. *Electronic Notes in Theoretical Computer Science*, 86(1):133–146, 2003.
6. Thomas Genet. Decidable approximations of sets of descendants and sets of normal forms. In *Proceedings of the 9th International Conference on Rewriting Techniques and Applications*, volume 1379 of *Lecture Notes in Computer Science*, pp. 151–165, Springer, 1998.
7. Thomas Genet and Francis Klay. Rewriting for cryptographic protocol verification. In *Proceedings of the 17th International Conference on Automated Deduction*, volume 1831 of *Lecture Notes in Computer Science*, pp. 271–290, Springer, 2000.
8. Karl Gmeiner and Bernhard Gramlich. Transformations of conditional rewrite systems revisited. In *Proceedings of the 19th International Workshop on Recent Trends in Algebraic Development Techniques*, volume 5486 of *Lecture Notes in Computer Science*, pp. 166–186, Springer, 2009.
9. Karl Gmeiner, Bernhard Gramlich, and Felix Schernhammer. On (un)soundness of unravelings. In *Proceedings of the 21st International Conference on Rewriting Techniques and*

Applications, volume 6 of *LIPIcs*, pp. 119–134, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2010.
10 Karl Gmeiner, Bernhard Gramlich, and Felix Schernhammer. On soundness conditions for unraveling deterministic conditional rewrite systems. In *Proceedings of the 23rd International Conference on Rewriting Techniques and Applications*, volume 15 of *LIPIcs*, pp. 193–208, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2012.
11 Karl Gmeiner, Naoki Nishida, and Bernhard Gramlich. Proving confluence of conditional term rewriting systems via unravelings. In *Proceedings of the 2nd International Workshop on Confluence*, pp. 35–39, 2013.
12 Florent Jacquemard. Decidable approximations of term rewriting systems. In *Proceedings of the 7th International Conference on Rewriting Techniques and Applications*, volume 1103 of *Lecture Notes in Computer Science*, pp. 362–376, Springer, 1996.
13 Salvador Lucas. Context-sensitive computations in functional and functional logic programs. *Journal of Functional and Logic Programming*, 1998(1), 1998.
14 Salvador Lucas, Claude Marché, and José Meseguer. Operational termination of conditional term rewriting systems. *Information Processing Letters*, 95(4):446–453, 2005.
15 Massimo Marchiori. Unravelings and ultra-properties. In *Proceedings of the 5th International Conference on Algebraic and Logic Programming*, volume 1139 of *Lecture Notes in Computer Science*, pp. 107–121, Springer, 1996.
16 Massimo Marchiori. On deterministic conditional rewriting. Computation Structures Group, Memo 405, MIT Laboratory for Computer Science, 1997.
17 Naoki Nishida, Masahiko Sakai, and Toshiki Sakabe. Soundness of unravelings for conditional term rewriting systems via ultra-properties related to linearity. *Logical Methods in Computer Science*, 8(3):1–49, 2012.
18 Enno Ohlebusch. Termination of logic programs: Transformational methods revisited. *Applicable Algebra in Engineering, Communication and Computing*, 12(1/2):73–116, 2001.
19 Enno Ohlebusch. *Advanced Topics in Term Rewriting*. Springer, 2002.
20 Felix Schernhammer and Bernhard Gramlich. Characterizing and proving operational termination of deterministic conditional term rewriting systems. *Journal of Logic and Algebraic Programming*, 79(7):659–688, 2010.
21 Traian-Florin Şerbănuţă and Grigore Roşu. Computationally equivalent elimination of conditions. In *Proceedings of the 17th International Conference on Rewriting Techniques and Applications*, volume 4098 of *Lecture Notes in Computer Science*, pp. 19–34, Springer, 2006.
22 Traian-Florin Şerbănuţă and Grigore Roşu. Computationally equivalent elimination of conditions. Technical Report UIUCDCS-R-2006-2693, Department of Computer Science, University of Illinois at Urbana-Champaign, 2006.
23 Taro Suzuki, Aart Middeldorp, and Tetsuo Ida. Level-confluence of conditional rewrite systems with extra variables in right-hand sides. In *Proceedings of the 6th International Conference on Rewriting Techniques and Applications*, volume 914 of *Lecture Notes in Computer Science*, pp. 179–193, Springer, 1995.
24 Toshinori Takai, Yuichi Kaji, and Hiroyuki Seki. Right-linear finite path overlapping term rewriting systems effectively preserve recognizability. In *Proceedings of the 11th International Conference on Rewriting Techniques and Applications*, volume 1833 of *Lecture Notes in Computer Science*, pp. 246–260, Springer, 2000.
25 Patrick Viry. Elimination of conditions. *Journal of Symbolic Computation*, 28(3):381–401, 1999.