On Simulation-Completeness of Unraveling for Conditional Term Rewriting Systems

Naoki NISHIDA†, Masahiko SAKAI†, and Toshiki SAKABE†
† Graduate School of Information Science, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya 464–8603, Japan

Abstract Transformations from conditional term rewriting systems (CTRSs) over (original) signatures into term rewriting systems (TRSs) over the extended signatures, are called unravelings. They are not always simulation-complete for any input CTRS. Here simulation-completeness means that every rewrite sequence of the unraveled CTRSs (the output TRSs), whose initial and final terms are over the original signatures, can be simulated by a rewrite sequence of the original CTRSs. We have proposed an unraveling which generates left-linear, right-linear and non-erasing TRSs from CTRSs satisfying some syntactic conditions, respectively. In this paper, we show two conditions that the unraveling is simulation-complete for CTRSs. One is that the unraveled CTRSs are right-linear and non-erasing, and the other is that they are left-linear. Under the latter condition, we assume that any redex introduced by extra variables is not reduced anywhere in the rewrite sequences of the unraveled CTRSs.

Keywords program transformation, right-linearity, non-erasingness, left-linearity, extra variable

1. Introduction

Transformations from conditional term rewriting systems (CTRSs) into term rewriting systems (TRSs) are called unravelings. The unraveled CTRSs are approximations of the CTRSs, and they are used to analyse properties of the CTRSs, such as syntactic properties, termination and so on. The reason of such uses is that TRSs can be treated more easily than CTRSs. Marchiori has proposed some unravelings for several kinds of CTRSs to analyse ultra-properties and modularity of the CTRSs [4]. Ohlebusch has presented an unraveling for deterministic 3-CTRSs to analyse termination of logic programs [9]. To guarantee effective termination of CTRSs, termination of the unraveled CTRSs are required [4].

On the other hand, the unraveled CTRSs are not used to simulate rewrite sequences of the original CTRSs although reduction of TRSs are more simple than that of CTRSs. The reason is that unravelings are not always simulation-complete for any CTRS. Here simulation-completeness means that every rewrite sequence of the unraveled CTRSs (the output TRSs), whose initial and final terms are over the original signatures, can be simulated by a rewrite sequence of the original CTRSs.

An unraveling based on Ohlebusch’s one has been proposed [6], and it has shown that the unraveling generates left-linear, right-linear and non-erasing TRSs from CTRSs satisfying some conditions, respectively.

In this paper, we show two conditions that the unraveling [6] is simulation-complete for CTRSs. One is that the unraveled CTRSs are right-linear and non-erasing, and the other is that they are left-linear. Under the latter condition, we assume that any redex introduced by extra variables is not reduced anywhere in the rewrite sequences of the unraveled CTRSs. We also show that under the first condition, neither right-linearity nor non-erasingness can be removed.

The unraveling which we treat in this paper, is used in an inverse compiler [6] that is a transformation from constructor TRSs into TRSs with extra variables (EV-TRSs) which compute inverses of functions in the input TRSs. In the first part of the compiler, constructor TRSs are transformed into deterministic 4-CTRSs, and then the CTRSs are transformed into EV-TRSs by using the unraveling. An example of the second transformation will be shown later (Example 2). It has been shown that all products of the inverse compiler are non-erasing, and those from left-linear (right-linear) TRSs are right-linear (left-linear, respectively). Hence, the results in this paper are useful for the inverse compiler. Since the first part of the inverse compiler generates 4-CTRSs and then the 4-CTRSs are unraveled to EV-TRSs, this paper treats 4-CTRSs and EV-TRSs.

2. Preliminaries

This paper follows general standard notations of term rewriting [1], [3], [8].

Let \( V \) be a countably infinite set of variables. The set of all terms over a signature \( \mathcal{F} \) and \( V \) is denoted by \( T(\mathcal{F}, V) \). Especially, we abbreviate the set \( T(\mathcal{F}, \emptyset) \) of all ground terms to \( T(\mathcal{F}) \). The set of variables occurring in either of terms \( t_1, \ldots, t_n \) is represented by \( \text{Var}(t_1, \ldots, t_n) \). Identity of terms is denoted by \( \equiv \). The root symbol of a term \( t \) is represented by \( \text{root}(t) \). A term \( t \) is linear if no variable occurs twice in \( t \).

The set of all positions of a term \( t \) is represented by \( \mathcal{O}(t) \), and the sets of all function symbol positions and all variable positions of \( t \) are denoted by \( \mathcal{O}_F(t) \) and \( \mathcal{O}_V(t) \), respectively. The root position is represented by \( \varepsilon \). For an \( n \)-hole context \( C[\ldots] \) and terms \( t_1, \ldots, t_n \), the notation \( C[t_1, \ldots, t_n]_{\varepsilon_1, \ldots, \varepsilon_n} \) (or simply \( C[t_1, \ldots, t_n] \)) denotes the term which is obtained by replacing the occurrences of \( \square \) at
positions $p_1, \ldots, p_n$ with $t_1, \ldots, t_n$ (from left to right). Given positions $p$ and $q$ of a term, we write $p \leq q$ if there exists $p' \neq e$ such that $pp' = q$. Moreover, we write $p < q$ if $p' \neq e$.

The domain and range of a substitution $\sigma$ are denoted by $\text{Dom}(\sigma)$ and $\text{Ran}(\sigma)$, respectively. $\text{Sub}(F, V)$ denotes the set of all substitutions whose range is over signature $F$, that is, $\text{Sub}(F, V) = \{ \sigma | \text{Ran}(\sigma) \subseteq \mathcal{T}(F, V) \}$. The application $\sigma(t)$ of $\sigma$ to a term $t$ may be abbreviated to $\sigma t$. The composition $\sigma \sigma'$ of substitutions $\sigma$ and $\sigma'$ is defined as $\sigma \sigma' = \sigma'(\sigma(x))$. When $\text{Dom}(\sigma) = \{ x_1, \ldots, x_n \}$ and $\sigma(x_i) = t_i$ for all $i \in \{1, \ldots, n\}$, we may write $\{ x_1 \rightarrow t_1, \ldots, x_n \rightarrow t_n \}$ for $\sigma$. For substitutions $\sigma$ and $\sigma'$, they are equivalent, written as $\sigma = \sigma'$, if $\text{Dom}(\sigma) = \text{Dom}(\sigma')$ and $\sigma(x) \equiv \sigma'(x)$ for all $x \in \text{Dom}(\sigma)$. The restriction $\sigma|_V$ of a substitution $\sigma$ is defined as $\sigma|_V = \{ x \rightarrow \sigma(x) | x \in \text{Dom}(\sigma) \cap V \}$. For substitutions $\sigma$ and $\sigma'$, we write $\sigma \leq \sigma'$ if there exists a substitution $\theta$ such that $\sigma \theta = \sigma'$.

An oriented conditional rewrite rule over a signature $F$ is a triple $(l, r, c)$, written as $l \rightarrow r < c$, where $l$ is a non-variable term in $\mathcal{T}(F, V)$, $r$ is a term in $\mathcal{T}(F, V)$, and $c$ is a sequence $s_1 \rightarrow t_1, \ldots, s_k \rightarrow t_k$ of oriented conditions with $s_1, t_1, \ldots, s_k, t_k \in \text{Dom}(\sigma)$. The terms $r$ and $t$ are called the left-hand side (LHS) and the right-hand side (RHS) of the rule $l \rightarrow r < c$, respectively. The sequence $c$ is called the conditional part of the rule. We say that the rule $l \rightarrow r < c$ is unconditional (or simply a rewrite rule), written as $l \rightarrow r$, if $k = 0$. The set of all variables in $c$ is denoted by $\text{Var}(c)$. We may write the rule $l \rightarrow r < c$ as $\rho : l \rightarrow r < c$ with a unique label $\rho$. Variables which occur both in the LHS $l$ and in either the RHS $r$ or the conditional part $c$ are called extra variables of the rule $\rho$. The set of all extra variables of $\rho$ is denoted by $\text{EvVar}(\rho)$, that is, $\text{EvVar}(\rho) = \text{Var}(r, c) \setminus \text{Var}(l)$ where $\text{Var}(r, c) = \text{Var}(r) \cup \text{Var}(c)$.

An oriented conditional rewriting system (oriented CTRS) over a signature $F$ is a finite set of oriented conditional rewrite rules over $F$. In this paper, we use the terminology “CTRS” as “oriented CTRS”, and similarly for “conditional rewrite rule”. Let $R$ be a CTRS over signature $F$. The $n$-level rewrite relation $\rightarrow^*_{R}$ of $R$ is recursively defined as follows: $\rightarrow^*_R = \emptyset$, and $C[l]_{\rightarrow^*_{R}} \rightarrow_{\rightarrow^*_{R}} C[r]_{\rightarrow^*_{R}}$ if and only if $\rho : l \rightarrow r < c$. The rewrite relation $\rightarrow_{R}$ of the CTRS $R$ is defined as $\rightarrow_{R} = \bigcup_{n \geq 0} \rightarrow^*_R$. To specify an applied rule $\rho$ and a position $p$ for $\rightarrow_{R}$, we write $\rightarrow^*_{\rho, p}$ or $\rightarrow^*_{\rho}$. We also write $\rightarrow^*_R < c$ if $q < p$, and similarly for $q \leq$. A rewrite sequence of $R$ is either a finite sequence $t_0 \rightarrow_{R} t_1 \rightarrow_{R} \cdots \rightarrow_{R} t_k$, or an infinite sequence $t_0 \rightarrow_{R} t_1 \rightarrow_{R} t_2 \rightarrow_{R} \cdots$, of terms in $\mathcal{T}(F, V)$. The term $t_0$ is called the initial term of the above rewrite sequence, and $t_k$ is the final term when the sequence is finite. The $n$-fold rewrite sequence $t_0 \rightarrow_{R}^{\ast n} t_n$ means that there exists a finite rewrite sequence $t_0 \rightarrow_{R} t_1 \rightarrow_{R} \cdots \rightarrow_{R} t_k$. The binary relation $\rightarrow_{R}$ is the reflexive and transitive closure of $\rightarrow_{R}$. $\rightarrow_{R}$ is the one-step rewrite of $t_0$ under $R$ in the presence of all LHSs in $R$ called reducts of $R$. A term $t$ is called a normal form of $R$ if there is no term $u$ such that $t \rightarrow_{R} u$. The CTRS $R$ is a term rewriting system (TRS) if it is an EV-TRS and every rule $l \rightarrow r \in R$ satisfies $\text{Var}(l) \supseteq \text{Var}(r)$.
Note that $X_t$ is the set of variables that appear in either of $l, t_1, \ldots, t_{i-1}$ and also in either of $r, t_i, s_{i+1}, t_{i+1}, \ldots, s_k, t_k$. Relatively to the above definition of $U$, we define the signature $\mathcal{F}_U$ determined by $F$ and $R$ as $\mathcal{F} \cup \{ u^k \mid k \in \mathbb{N} \}$. It is clear that the unraveled CTRS $U(R)$ is over the signature $\mathcal{F}_U$.

**Example 2** Consider the following deterministic CTRS \[ R_1 = \begin{cases} \text{add}^{-1}(0) \to \text{tp}_2(0,y), \\ \text{add}^{-1}(s(z)) \to \text{tp}_2(s(x),y), \\ \text{mul}^{-1}(0) \to \text{tp}_2(x,0), \\ \text{mul}^{-1}(s(z)) \to \text{tp}_2(s(x),y), \\ \text{mul}^{-1}(w) \to \text{tp}_2(x,s(y)), \\ \text{mul}^{-1}(\text{mul}(x,y)) \to \text{tp}_2(x,y), \end{cases} \]

$R_1$ is unraveled by $U$ as follows:

\[ U(R_1) = \begin{cases} \text{add}^{-1}(0) \to \text{tp}_2(0,y), \\ \text{add}^{-1}(s(z)) \to u_1(\text{add}^{-1}(z)), \\ u_1(\text{tp}_2(x,y)) \to \text{tp}_2(s(x),y), \\ \text{add}^{-1}(\text{add}(x,y)) \to \text{tp}_2(x,y), \\ \text{mul}^{-1}(0) \to \text{tp}_2(x,0), \\ \text{mul}^{-1}(s(z)) \to u_2(\text{add}^{-1}(z)), \\ u_2(\text{tp}_2(w,y)) \to u_1(\text{mul}^{-1}(w),y), \\ u_3(\text{tp}_2(x,y)) \to \text{tp}_2(s(x),y), \\ \text{mul}^{-1}(\text{mul}(x,y)) \to \text{tp}_2(x,y). \end{cases} \]

The unraveling $U$ defined above is based on an unraveling for deterministic 3-CTRSs which has been proposed by Ohlebusch [9]. The difference between $U$ and Ohlebusch’s one is that $X_{i+1}$ is used instead of $\var{\ell}$, $\var{\ell_1}$, ..., $\var{\ell_k}$ in the definition of $U(\rho)$. The variable list $\var{\ell}$, $\var{\ell_1}$, ..., $\var{\ell_k}$ may contain variables not used in the later part $(s_{i+1}, t_{i+1}, \ldots, s_k, t_k, r)$, and such variables are redundant for simulations of rewrite sequences of CTRSs.

To analyse syntactic relations between a deterministic CTRS $R$ and the EV-TRS $U(R)$, we here define syntactic properties of deterministic CTRSs.

**Definition 3** \([6]\) Let $\rho : \lambda \to r \iff s_1 \to t_1, \ldots, s_k \to t_k$ be a conditional rewrite rule. The rule $\rho$ is said to be

- strictly non-erasing if all variables in $l$ appear in either of $r, s_1, t_1, \ldots, s_k, t_k$, and all variables in $t_i$ appear in either of $r, s_1, t_1, \ldots, s_k, t_k$, that is, $\var{\ell} \subseteq \var{\ell_1}, \var{\ell_2}, \ldots, \var{\ell_k}$ and

- $\var{\ell_i} \subseteq \var{\ell_{i+1}, \ldots, \ell_k}$ for $1 \leq i \leq k$,

- strictly left-linear if $l, t_1, \ldots, t_{i-1}$ are linear and no variable in $s_i$ appears in either of $r, t_i, s_{i+1}, t_{i+1}, \ldots, s_k, t_k$, that is, $\var{\ell_i} \cap \var{\ell_1, \ldots, \ell_{i-1}} = \emptyset$, and

- strictly right-linear if $r, s_1, \ldots, s_k$ are linear and no variable in $s_i$ appears in either of $r, t_i, s_{i+1}, t_{i+1}, \ldots, s_k, t_k$, that is, $\var{\ell_i} \cap \var{\ell_{i+1}, \ldots, \ell_k} = \emptyset$.

$U$ enjoys the following nice properties associated with the above syntactic features.

**Proposition 4** \([6]\) Let $R$ be a deterministic CTRS.

- $R$ is a 3-CTRS if and only if $U(R)$ is a TRS.

- $R$ is strictly NE if and only if $U(R)$ is NE.

- $R$ is strictly LL if and only if $U(R)$ is LL.

- $R$ is strictly RL if and only if $U(R)$ is RL.

**Example 5** The CTRS $R_1$ in Example 2 is strictly RL and strictly NE, and the unraveled CTRS $U(R_1)$ is RL and NE.

### 4. Completeness of Unraveling

Given an unraveling $U$, we say that $U$ is simulation-sound for a CTRS $R$ over a signature $\mathcal{F}$ if for every terms $s$ and $t$ in $\mathcal{F}(\mathcal{V})$, $s \longrightarrow^* R t$ implies $s \longrightarrow^*_U R t$. Conversely, $U$ is simulation-complete for $R$ over $\mathcal{F}$ if for every terms $s$ and $t$ in $\mathcal{F}(\mathcal{V})$, $s \longrightarrow^*_U R t$ implies $s \longrightarrow^* R t$.

Every unraveling is clearly simulation-sound for all CTRSs in its target class (because simulation-soundness is a necessary condition for a transformation from CTRSs to EV-TRSs being an unraveling). However, unravelings are not always simulation-complete for any CTRS in their target classes. The following CTRS [8] \begin{align*} & R_2 = \left\{ f(x) \to x \iff x \to e, \\
& h(x, x) \to g(x, x, f(k)), \quad g(d, x, x) \to A, \\
& a \to c, \\
& b \to c, \quad c \to e, \quad k \to l, \quad d \to m, \\
& a \to d, \quad b \to d, \quad c \to l, \quad k \to m. \right. \end{align*} 
We obtain by $U$ the following TRS:

\[ U(R_2) = \begin{cases} \text{add}^{-1}(0) \to \text{tp}_2(0,y), \\ \text{add}^{-1}(s(z)) \to u_1(\text{add}^{-1}(z)), \\ u_1(\text{tp}_2(x,y)) \to \text{tp}_2(s(x),y), \quad \text{add}^{-1}(\text{add}(x,y)) \to \text{tp}_2(x,y), \\ \text{mul}^{-1}(0) \to \text{tp}_2(x,0), \\ \text{mul}^{-1}(s(z)) \to u_2(\text{add}^{-1}(z)), \\ u_2(\text{tp}_2(w,y)) \to u_1(\text{mul}^{-1}(w),y), \\ u_3(\text{tp}_2(x,y)) \to \text{tp}_2(s(x),y), \quad \text{mul}^{-1}(\text{mul}(x,y)) \to \text{tp}_2(x,y). \end{cases} \]

The unraveling $U$ is not simulation-complete in general.

**Example 6** Consider the following deterministic CTRS:

\[ R_2 = \left\{ f(x) \to u_1(x, x), \\
& u_4(e, x) \to x, \\
& h(x, x) \to g(x, x, f(k)), \quad g(d, x, x) \to A, \\
& a \to c, \\
& b \to e, \quad c \to e, \quad k \to l, \quad d \to m, \\
& a \to d, \quad b \to d, \quad c \to l, \quad k \to m. \right. \]

There exists a rewrite sequence of $U(R_2)$ from $h(f(a), f(b))$ to $A$, but there exists no rewrite sequence of $R_2$ from $h(f(a), f(b))$ to $A$, that is, $h(f(a), f(b)) \not\longrightarrow^* R_2 A$ and $h(f(a), f(b)) \not\longrightarrow^*_R A$. Therefore, $U$ is not simulation-complete for $R_2$.

Now we discuss why $U$ is not simulation-complete for $R_2$. We show a detail of $h(f(a), f(b)) \not\longrightarrow^*_R A$ as follows:

\[ h(f(a), f(b)) \not\longrightarrow^*_R h(u_4(c, d), u_4(d, c)) \]

\[ \longrightarrow^*_R g(u_4(d, c), u_4(c, d), f(k)) \]

\[ \longrightarrow^*_R g(d, u_4(l, m), u_4(l, m)) \longrightarrow^*_R A. \]

To construct the above reduction, the followings are required:

- to apply the rule $g(d, x, x) \to A$, the subterm $f(a)$ in the initial term is reduced to the term $d$,

- to apply the rule $h(x, x) \to g(x, x, f(k))$, the subterm $f(a)$ and $f(b)$ in the initial term are reduced to the same term, and

- to apply the rule $g(d, x, x) \to A$, the subterm $f(b)$ in the initial term and $f(k)$ are reduced to the same term.
The above requirements are summarized as, “$f(a), f(b)$ and $f(k)$ are reduced to the same term $d$”. In this case, it is however impossible on $R_h$. Nevertheless, $h(x, y) \rightarrow g(x, y, f(k))$ is used by reducing $f(a)$ and $f(b)$ to the same term $u(c, d)$. After then, one of $u(c, d)$ which comes from $f(a)$ is reduced to $d$, and the other $u(c, d)$ from $f(b)$ is reduced to the term $u(l, m)$ to be the same with $f(k)$. After then, $g(d, x, x) \rightarrow A$ is used.

These undesirable reductions may be caused by erasing rules and non-right-linear rules because $h(x, y) \rightarrow g(x, y, f(k))$ remains two deliver terms $u(c, d)$ from $f(a)$ and $f(b)$ (each of which can perform a different role later), and $g(d, x, x) \rightarrow A$ erases two terms by considering them as the same (although it should not be done).

4.1 On Right-Linearity and Non-Erasing

From the discussion before this subsection, U seems to be simulation-complete for a CTRS $R$ if $U(R)$ is RL and NE. In this subsection, we prove this claim.

The fact that every rewrite sequence of right-linear TRSs can be transformed to a basic rewrite sequence, have been proved by Middeldorp and Hanoen [5], and this result has been extended on EV-TRSs [6].

Proposition 7 ( [6] ) Let $R$ be a right-linear EV-TRS, $s, t$ be terms, and $\theta$ be a substitution such that $\theta(x)$ is a normal form of $R$ for all $x \in \text{Dom}(\theta)$. Then, $s \theta \not\rightarrow_R t$ implies $O_R(s) \equiv \theta \not\rightarrow_R t$.

The above proposition says that every rewrite sequence of EV-TRSs can be transformed into a basic one. According to this fact, we can focus on basic rewrite sequences.

To treat reductions of strictly-LL CTRSs easily, we prepare a technical lemma. Let $S, S'$ be sets, $\vec{A}$ be a list $a_1, \ldots, a_n$ on $S$, $\vec{B}$ be a list $b_1, \ldots, b_m$ on $S'$, and $\sigma$ be a binary relation on $S \times S'$. Then, we write $\vec{A} \rightarrow^\sigma \vec{B}$ if $m = n$ and $a_i \rightarrow^\sigma b_i$ holds for all $i \in \{1, \ldots, n\}$. For a substitution $\sigma$, $\vec{A} \sigma$ denotes the unique list $a_1 \sigma, \ldots, a_n \sigma$.

Lemma 8 Let $R$ be a CTRS, $l \rightarrow r \leftarrow s_1 \rightarrow t_1 \ldots \leftarrow s_k \rightarrow t_k \in R$ be strictly RL, and $\sigma_1, \ldots, \sigma_k$ be substitutions. Let $X_i = \text{Var}(l(t_i), l(t_i)) \cap \text{Var}(r(t_i), r(t_i), s_i, s_i, s_i, \ldots, s_i)$ if $s_i \not\rightarrow_R \sigma_i$ and $X_i = \text{Var}(l(t_i), l(t_i))$ for every $i \in \{1, \ldots, k\}$. Then, $l \sigma_1 \not\rightarrow_R r \sigma_1$.

Proof. We only prove the case of $k = 1$. In this case, we have $X_1 = \text{Var}(l) \cap \text{Var}(r, t_1) \subseteq \text{Var}(l)$ from the assumption, and we also have $X_1 \cap \text{Var}(s_i) = \emptyset$ from the strictly-RL.

Let $V = EV \cap \text{Var}(l(t_1), l(t_1), l(t_1))$ and $\sigma = \sigma_1 |_{\text{Var}(l(t_1), l(t_1))} \cup \sigma_2 |_{\text{Var}(t_1)} |_{\text{Var}(t_1)}(l(t_1))$. It follows from the definition of extra variables that $V \cap \text{Var}(t_1) = \emptyset$. Hence $\sigma$ is a substitution.

We have $X_1 \sigma_1 = \text{Var}(l(t_1), l(t_1)) \cap \text{Var}(r(t_1), r(t_1)) \cap \text{Var}(s_i)$ from the assumptions. It follows from $X_1 \sigma_1 = \text{Var}(l(t_1), l(t_1)) \cap \text{Var}(r(t_1), r(t_1)) \cap \text{Var}(s_i) = \emptyset$ that $s_i \not\rightarrow_R \sigma_1$ and hence $l \sigma_1 \not\rightarrow_R r \sigma_1$.

Since $\text{Var}(l(t_1), l(t_1)) \subseteq \text{Var}(l(t_1)) \subseteq \text{Var}(l(t_1), l(t_1))$, we have $s_i \not\rightarrow_R \sigma_1$. Therefore, we have $l \sigma_1 \not\rightarrow_R r \sigma_1$.

To pick up parts of basic rewrite sequences, we show the following property related on rewrite sequences after reductions at the root position.

Proposition 9 Let $R$ be an EV-TRS, $l \rightarrow r \in R$, $s, t$ be terms, $B \subseteq O_R(s)$, and $\theta$ be a substitution. Then, $B : s \not\rightarrow_R \theta \not\rightarrow_R t$ implies $B : s \not\rightarrow_R \theta \not\rightarrow_R t$ and $O_R(r) = \theta \not\rightarrow_R t$. For a context $C[l]_{\theta}$ and a set $B' \subseteq O_R(C[s]$ p $)$ such that $B = \{ q \mid p \in B' \}$, $B' : C[s]_{\theta} \not\rightarrow_R t$ implies $B : s \not\rightarrow_R t$.

From the definition of $\not\rightarrow_R t$, this proposition holds clearly.

The following lemma is a generalized one of the claim that $U$ is simulation-complete (with respect to basic sequences) for every strictly-RL and strictly-NE CTRS.

Lemma 10 Let $R$ be a strictly-RL and strictly-NE CTRS over a signature $\mathcal{F}$. Let $s, t \in T(F, V)$, $B \subseteq O_R(s)$, and $\theta \in \text{Sub}(F, V)$. Then, $B : s \not\rightarrow_{U(R)} t$ implies $s \not\rightarrow_R t$.

Proof. For a sequence $B : s \not\rightarrow_{U(R)} t$, we prove this lemma by induction on the lexicographic combination of $n$ and the structure of the term $s$.

The case that $s$ is a variable, holds obviously since $B : s \not\rightarrow_{U(R)} t$ implies $s \not\rightarrow_R t$.

The remaining case is that $s$ is not a variable. We first consider the subcase that the sequence $s \not\rightarrow_{U(R)} t$ does not contain any reduction at the root position, that is, $s \not\rightarrow_{U(R)} t$. This subcase is easily proved by the induction hypothesis because the sequence can be decomposed to rewrite sequences of which numbers of steps are less equal than $n$, and which start from the proper subterms of $s$.

We consider the remaining subcase. From the construction of $U(R)$, $U(R)$ has the unique rule for every $u_i' \notin F$. Now the root symbols of $s$ and $t$ are in $F$ (root(s), root(t)) $\notin F$.

According to these facts, we can assume that $u_i \notin F$.

Proposition 9, we have $O_R(r) = \not\rightarrow_R \sigma_1$. Since $t \in T(F, V)$ and $U(R)$ is NE by Proposition 4, any term introduced by $\sigma_1 |_{\text{Var}(r)}$ does not contain any symbol in $F \cup \mathcal{F}$, that is, $\text{Ran}(\sigma_1 |_{\text{Var}(r)}) \subseteq \mathcal{F}$. We also obtain $\text{Ran}(\sigma_0 |_{\text{Var}(r)}) \subseteq \mathcal{F}$ from the same reason above for the part $u_i'(s_1, X_1, \sigma_0 |_{\text{Var}(r)}, \sigma_1 |_{\text{Var}(r)}(l) \not\rightarrow_R l \sigma_0 |_{\text{Var}(r)}(l) \not\rightarrow_R l \not\rightarrow_R r \sigma_1$.

Hence, we have $\text{Ran}(\sigma_0 |_{\text{Var}(r)}, \sigma_1 |_{\text{Var}(r)}(l) \not\rightarrow_R l \not\rightarrow_R r \sigma_1$ from the basicness of that sequence. By the induction hypothesis, we have $s \not\rightarrow_R l \not\rightarrow_R r \sigma_1$ and $t \not\rightarrow_R r \sigma_1$.

From Lemma 8, we have $l \not\rightarrow_R r \sigma_1$. Therefore, we obtain $s \not\rightarrow_R l \not\rightarrow_R r \sigma_1 \not\rightarrow_R t$ (see Fig. 1).

Finally, we obtain the following result.

Theorem 11 $U$ is simulation-complete for every strictly-RL and strictly-NE CTRS.
\begin{proof}
Let \( s, t \in T(\mathcal{F}, \mathcal{V}) \), and suppose that \( s \xrightarrow[n]{\mathcal{U}(R)} t \). Since \( \mathcal{U}(R) \) is RL by Proposition 4, we have a basic rewrite sequence \( s \xrightarrow[\mathcal{U}(R)]{l} t \) from Proposition 7. Then, from Lemma 10, we obtain \( s \xrightarrow{R} t \).
\end{proof}

The conjecture in [6] that \( U \) is simulation-complete for every strictly-NE CTRS \( R \), does not hold. Consider the CTRS \( R_3 \) obtained from \( R_2 \) by replacing \( A \) with \( x \). Although we have \( h(f(a), f(b)) \notin \xrightarrow[n]{\mathcal{U}(R)} l \), there is no rewrite sequence of \( R_3 \) from \( h(f(a), f(b)) \) to \( l \), that is, \( h(f(a), f(b)) \xrightarrow{R_3} l \). Therefore, there exists a strictly-NE CTRS for which \( U \) is not simulation-complete.

Similarly, the other condition “strictly-NE” cannot be also removed. Consider the strictly-RL and non-NE CTRS \( R_4 = \{ f(x) \rightarrow e \leftarrow d \rightarrow l \}, \mathcal{V} \) \( A \rightarrow x \) \( A \). This CTRS is unobservable to \( \mathcal{U}(R_4) \) = \{ \( f(x) \rightarrow u_2(x), u_3(l) \rightarrow x \), \( A \rightarrow x \) \}. Though we have \( h(f(a), f(b)) \notin \xrightarrow[n]{\mathcal{U}(R_4)} A \), there is no rewrite sequence of \( R_4 \) from \( h(f(a), f(b)) \) to \( A \), that is, \( h(f(a), f(b)) \xrightarrow{R_4} A \). Therefore, there exists a strictly-RL CTRS for which \( U \) is not simulation-complete.

\section{On Left-Linearity}
Marchiori has shown that his unraveling [4] for normal CTRSs are simulation-complete for CTRSs if the unraveled CTRSs are left-linear. In addition, the left-linearity seems to be another solution of the problems as we have shown in Example 2. Hence, our unraveling \( U \) seems to be simulation-complete for a strictly-LL CTRS \( R \) (\( \mathcal{U}(R) \) is left-linear from Proposition 4). This claim holds for 3-CTRSs but not for 4-CTRSs. The reason is that extra variables appearing only in RHSs make mischief by introducing garbage terms. We can solve this problem if we restrict rewrite sequences to \( EV-safe \) ones in which any redex introduced by extra variables is not reduced anywhere. In this subsection, we prove the above claim for such sequences.

We first define EV-safe rewrite sequences of EV-TRSs [6], [7]. In these sequences, any redex introduced by extra variables are not reduced anywhere. Let \( R \) be an EV-TRS. EV-safe rewrite sequences are defined similarly to basic rewrite sequences, by adding \( \{ p \rightarrow q \mid p \in B, q \in \mathcal{O}_V(t_1), t_1 \mid \theta_i \in \mathcal{V}_R \} \) to \( B_{i+1} \) in the definition of \( \xrightarrow{\mathcal{U}(R)} \). We say that the above rewrite sequence is EV-safe with respect to \( B_i \) if \( p_i \in B_i \), for \( 1 \leq i < n \). Especially, it is EV-safe if \( B_1 = \mathcal{O}_F(t_1) \). To represent the above rewrite sequence, we may write \( B_i \xrightarrow{\mathcal{U}(R)} t_i \). Especially, we simply write \( t_i \xrightarrow{\mathcal{U}(R)} t_i \) if \( B_i = \mathcal{O}_F(t_i) \). The definition of \( \xrightarrow{\mathcal{U}(R)} \) is different from that in [6], [7], however they are equivalent essentially. A typical instance of EV-safe rewrite sequences is a rewrite sequence obtained by substituting a normal form for each extra variable.

We again prepare a technical lemma to treat reductions of strictly-LL CTRSs easily.

\begin{lemma}
Let \( R \) be a CTRS, \( l \rightarrow r \leftarrow s_1 \rightarrow t_1 \ldots s_k \rightarrow t_k \in R \) be strictly LL, and \( \alpha_1, \ldots, \alpha_{k_1} \) be substitutions. Let \( X_i = \operatorname{Var}(l) \cap \operatorname{Var}(r, t_i) \notin \operatorname{Var}(l) \) from the assumption, and we also have \( X_i \cap \operatorname{Var}(l, t_i) = \emptyset \) from the strictly-LL.
\end{lemma}

\begin{proof}
We only prove the case of \( k = 1 \). In this case, we have \( X_1 \cap \operatorname{Var}(l) \cap \operatorname{Var}(r, t_1) \notin \operatorname{Var}(l) \) from the assumption, and we also have \( X_1 \cap \operatorname{Var}(l, t_1) = \emptyset \) from the strictly-LL.
\end{proof}

\section{Proposition 13}
Let \( R \) be a strictly LL CTRS over a signature \( \mathcal{F} \), and \( s, t \in T(\mathcal{F}, \mathcal{V}) \). If \( s \xrightarrow[n]{\mathcal{U}(R)} t \) is EV-safe, then it is \( EV-arranged \) on \( T(\mathcal{F}, \mathcal{V}) \).

\begin{proof}
(Sketch) We can prove by induction on steps \( n \) for which \( s \in T(\mathcal{F}_1, \mathcal{V}) \), \( t \in T(\mathcal{F}, \mathcal{V}) \), and \( \theta \in \operatorname{Sub}(\mathcal{F}_1, \mathcal{V}) \), if \( \mathcal{O}_F(s) : \theta \xrightarrow{n} \mathcal{O}_F(t) \), then there exists a substitution \( \theta' \in \operatorname{Sub}(\mathcal{F}, \mathcal{V}) \) such that \( \theta' \xrightarrow{n} \mathcal{O}_F(t') \) and \( \theta' \xrightarrow{n} \theta \xrightarrow{n} \theta \). From this lemma, we can obtain the claim of this proposition.

As another proof, we can prove this proposition by using the completeness of narrowing of EV-TRSs. As the completeness of narrowing, it holds that for a ground term \( s' \in T(\mathcal{F}) \) and a term \( t' \in T(\mathcal{F}, \mathcal{V}) \), if \( s' \xrightarrow{\mathcal{U}(R)} t' \), then there exist a term \( u \) and substitutions \( \sigma, \tau \) such that \( s' \xrightarrow{n} u \) \( u \) and \( \theta \equiv \theta' \in [6], [7] \). Here we can assume that \( \operatorname{Dom}(\sigma) \) only contains extra variables which are rules of applied at each step. Since \( t' \in T(\mathcal{F}, \mathcal{V}) \), we have \( u \in T(\mathcal{F}, \mathcal{V}) \) and we can suppose that \( \operatorname{Var}(\theta) \in T(\mathcal{F}, \mathcal{V}) \). From the construction of \( U \), every rule \( l \rightarrow r \in \mathcal{U}(R) \) satisfies that all proper subterms of \( l \) are in \( T(\mathcal{F}, \mathcal{V}) \). Since \( \mathcal{U} \) is determined by most general unifiers of terms and LHSs at each step and \( l \) is linear from the LL of \( \mathcal{U}(R) \), we have \( \operatorname{Var}(\theta) \subseteq T(\mathcal{F}, \mathcal{V}) \). Hence, we have \( \operatorname{Var}(\theta') \subseteq T(\mathcal{F}, \mathcal{V}) \). This means that terms in \( T(\mathcal{F}, \mathcal{V}) \) are substituted for each extra variables in every step of \( s' \xrightarrow{\mathcal{U}(R)} t' \). Then, \( s' \xrightarrow{\mathcal{U}(R)} t' \) is \( EV-arranged \) on \( T(\mathcal{F}, \mathcal{V}) \). In every rewrite sequence, variables in redexes can be considered as constants. Therefore, the above claim also holds for \( s \xrightarrow{\mathcal{U}(R)} t \).
\end{proof}

The following lemma is a generalization of the main claim in this subsection.

\begin{lemma}
Let \( R \) be a strictly LL CTRS, \( s \in T(\mathcal{F}, \mathcal{V}) \), \( t \) be a linear term in \( T(\mathcal{F}, \mathcal{V}) \), and \( \theta \in \operatorname{Sub}(\mathcal{F}_1, \mathcal{V}) \). If \( s \xrightarrow[n]{\mathcal{U}(R)} t \) (for some \( n \geq 0 \)) is \( EV-arranged \) on \( T(\mathcal{F}, \mathcal{V}) \), then there exists a substitution \( \theta' \in \operatorname{Sub}(\mathcal{F}, \mathcal{V}) \) such that \( s \xrightarrow{n} \theta' \) and \( \theta' \xrightarrow{n} \mathcal{U}(R) \theta' \) for some \( n' \leq n \). In this case, we have \( s \xrightarrow{n} \mathcal{U}(R) \theta' \).
\end{lemma}

\begin{proof}
We prove this lemma by induction on the lexicographic combination of \( n \) and the structure of the term \( s \).

Since the case of \( n = 0 \) or \( s \in \mathcal{V} \) holds clearly, we concentrate the remaining case, that is, \( n > 0 \) and \( s \notin \mathcal{V} \).

Consider the subcase that \( s \equiv \mathcal{U}(R) \theta' \) does not contain any reduction at the root position, that is, \( s \equiv \mathcal{U}(R) \theta' \) where \( \mathcal{U}(R) \).
It is clear that for every TRS $R$, all rewrite sequences of $R$ are EV-safe. Hence we obtain the following corollary from Theorem 15.

**Corollary 16** $U$ is simulation-complete for every strictly-L1 CTRS.

Here we show a counterexample that the above theorem does not hold for non-EV-safe sequences. Consider the CTRS $R_5 = \{ e \to f(x) \Leftarrow l \Rightarrow d, A \to h(x,x),B \}$ and the EV-TRS $U(R_5) = \{ e \to u(l), u(d) \Rightarrow f(x), A \to h(x,x) \}$. We have $A \overset{R_5}{\Rightarrow} h(f(a), f(b))$ but $A \not \overset{U(R_5)}{\Rightarrow} h(f(a), f(b))$.

Remark that the results in this paper (Theorem 11 and 15) do not hold for Ohlebusch’s unraveling (on which U is based) because Proposition 4 does not hold for his unraveling.

5. Conclusion

We have shown that the unraveling U is simulation-complete for every deterministic CTRS which is either strictly-LL or strictly-RL and strictly-NE, where rewrite sequences of the unravelled CTRSs are EV-safe under the former condition. From the discussion why simulation-completeness does not hold, U seems to be simulation-complete with respect to EV-safe rewrite sequences for a CTRS $R$ if every rule in $U(R)$ is either LL or RL, and NE, in other words, if every rule in $R$ is either strictly LL or strictly RL and strictly NE. As a future work, we will prove this conjecture (or show a counterexample). We are also interested in a relationship between LL, RL, NE, and the existence of extra variables.

**References**