

Decidability of Termination and Innermost Termination for Term Rewriting Systems with Right-Shallow Dependency Pairs

Keita UCHIYAMA^{†*}, Nonmember, Masahiko SAKAI^{†a)}, and Toshiki SAKABE[†], Members

SUMMARY In this paper, we show that the termination and the innermost termination properties are decidable for the class of term rewriting systems (TRSs for short) all of whose dependency pairs are right-linear and right-shallow. We also show that the innermost termination is decidable for the class of TRSs all of whose dependency pairs are shallow. The key observation common to these two classes is as follows: for every TRS in the class, we can construct, by using the dependency-pairs information, a finite set of terms such that if the TRS is non-terminating then there is a looping sequence beginning with a term in the finite set. This fact is obtained by modifying the analysis of argument propagation in shallow dependency pairs proposed by Wang and Sakai in 2006. However we gained a great benefit that the resulted procedures do not require any decision procedure of reachability problem used in Wang’s procedure for shallow case, because known decidable classes of reachability problem are not larger than classes discussing in this paper.

key words: looping sequence, argument propagation

1. Introduction

Termination is one of the central properties of term rewriting systems (TRSs for short), where we say a TRS terminates if it does not admit any infinite reduction sequence. The termination property is undecidable not only in general, but also for some classes: TRSs having single rule [1], flat TRSs [2] and length-two string rewriting systems [3]. Thus several classes whose termination is decidable have been studied: right-linear right-shallow TRSs [2], left-linear shallow TRSs, semi-constructor TRSs (TRSs all of whose dependency pairs are right-ground) [4], and so on [5]–[8]. Relationships between these classes are summarized in Fig. 1, where arrows indicate class inclusion, broken lines display the border between decidability and undecidability, and R and L are abbreviation of “right” and “left” respectively.

The innermost reduction strategy, which rewrites innermost redexes, is used for call-by-value computation. The termination property with respect to the innermost reduction is called innermost termination. Since the innermost termination is also undecidable in general, several classes whose innermost termination is decidable have been studied: shallow TRSs, right-linear right-shallow TRSs [2] and semi-constructor TRSs [4]. Relationships between these classes are summarized in Fig. 2.

In this paper, we show the following results.

1. The termination and the innermost termination of a term are decidable properties for TRSs all of whose dependency pairs are right-shallow.
2. The termination and the innermost termination properties are decidable for TRSs all of whose dependency pairs are right-linear and right-shallow.
3. The innermost termination is decidable for TRSs all of whose dependency pairs are shallow.
4. An extension of these results by combining with the result of semi-constructor TRSs and other techniques related to dependency pairs.
5. The termination is undecidable for TRSs all of whose dependency pairs are left-linear and shallow.

The key idea for deciding termination of this paper is the following. As already known, infinite reduction sequences guarantees the existence of infinite dependency chains. Focusing on the fact that all proper subterms occurring in the chains are terminating, it is possible to prove the existence of dependency chains having a looping structure

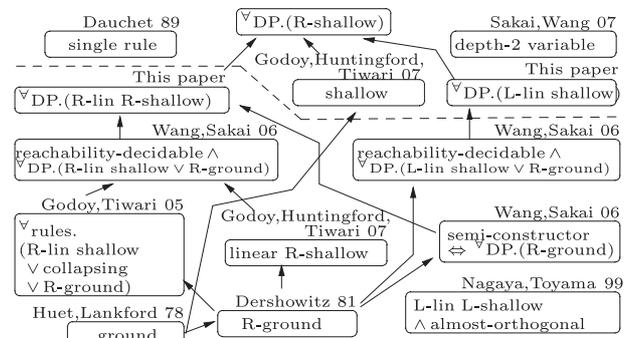


Fig. 1 (Un)-decidable classes on termination.

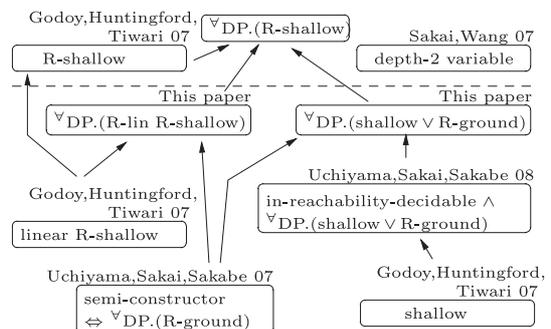


Fig. 2 (Un)-decidable classes on innermost termination.

Manuscript received July 17, 2009.

Manuscript revised November 10, 2009.

[†]The authors are with the Graduate School of Information Science, Nagoya University, Nagoya-shi, 464–8601 Japan.

*Presently, with Suzuki Motor Corporation.

a) E-mail: sakai@is.nagoya-u.ac.jp

DOI: 10.1587/transinf.E93.D.953

in it and starting from a term in a finite set T determined from rewrite rules of R . This is proved by modifying the technique used in the proof of decidability of termination for right-linear shallow TRSs [4]. After all, in order to check termination of R , we generate all derivations from terms in T . Either we detect termination of all such derivations and halt with “termination”, or find a loop and halt with “non-termination”.

One may think that the result 1 is a small extension of those in [4]. However we have a great benefit that the procedures in this paper do not require any decision procedure of reachability problems, because known decision procedures of reachability problem are considerably complex and the classes is not larger than classes discussing in this paper.

2. Preliminary

We assume that readers are familiar with the standard definitions of term rewriting systems [9] and dependency pairs [10].

The followings are basic notations to be used in this paper.

$\text{arity}(f)$: the <i>arity</i> of a function symbol f
$\mathcal{T}(\mathcal{F}, \mathcal{V})$: the set of all <i>terms</i> over signature \mathcal{F} and variable set \mathcal{V}
$\mathcal{T}(\mathcal{F})$: the set $\mathcal{T}(\mathcal{F}, \emptyset)$ of all <i>ground</i> terms over signature \mathcal{F}
$\text{Var}(t)$: the set of all variables occurring in term t
ε	: the <i>empty string</i> , and hence the <i>root</i> position of any term
$\text{root}(t)$: the symbol at the root position in t
$C[t]_p, C[t]$: the term obtained from a context C by replacing the hole \square with a term t (at position p)
$t _p$: the subterm of t at position p
\preceq, \triangleleft	: subterm relation and proper subterm relation
$t\theta$: the term obtained by applying a substitution θ to a term t
$\text{Dom}(\theta)$: the domain of a substitution θ , i.e., $\{x \mid x\theta \neq x\}$
$\text{Ran}(\theta)$: the range of a substitution θ , i.e., $\{x\theta \mid x\theta \neq x\}$
$\rightarrow^+, \rightarrow^*$: the transitive closure and the reflexive transitive closure of a relation \rightarrow
$\rightarrow \circ \rightarrow'$: the composition of relations \rightarrow and \rightarrow'

A *rewrite rule* $l \rightarrow r$ is a pair of terms such that $l \notin \mathcal{V}$ and $\text{Var}(r) \subseteq \text{Var}(l)$. A *term rewriting system (TRS)* R is a finite set of rewrite rules. A *redex* is a term t such that $t = l\theta$ for some rewrite rule $l \rightarrow r$ and substitution θ . An *(R-)normal form* is a term containing no redex. A substitution θ is *normal* if $x\theta$ is a normal form for every x . The *reduction relation* \xrightarrow{R} is defined as $\{(C[l\theta]_p, C[r\theta]_p) \mid l \rightarrow r \in R, \theta \text{ is a substitution, } C \text{ is a context with } \square \text{ at a position } p\}$. We write $s \xrightarrow{R} t$ if $s \xrightarrow{R} t$ and a redex at a position p is contracted.

A term is said to be *linear* if no variable occurs more than once in the term. The *depth* of a position p is the length of p . A term t is *shallow* if variable occurrences in t are at depth 0 or 1. A rewrite rule $l \rightarrow r$ is *right-linear* (*left-linear*, *shallow*, *right-shallow*, *right-ground*) if r is linear (resp. l is linear, l and r are shallow, r is shallow, r is ground). A TRS R is *right-linear* (*left-linear*, *shallow*, *right-shallow*, *right-ground*) if every rule in R is right-linear (resp. left-linear, shallow, right-shallow, right-ground).

We use the followings to represent a set of terms.

$R_{\text{lhs}}, R_{\text{rhs}}$: the set of left-hand (resp. right-hand) sides of a TRS R ; $\{l \mid l \rightarrow r \in R\}$ (resp. $\{r \mid l \rightarrow r \in R\}$)
$\text{Arg}(T)$: the set of immediate subterms of a term in T ; $\{t_i \mid f(t_1, \dots, t_n) \in T\}$
$\text{Inst}(T, T')$: the set of terms obtained from a term in T by instantiation of a term in T' ; $\{t\sigma \mid t \in T, \text{Dom}(\sigma) = \text{Var}(t), \text{Ran}(\sigma) \subseteq T'\}$
$\rightarrow[T]$: the set of terms obtained by reducing a term in T by \rightarrow ; $\{s \mid t \in T, t \rightarrow s\}$

A redex is *innermost* if all of its proper subterms are normal forms. Define the *innermost reduction relation* $\xrightarrow{\text{in}, R}$ as follows: $s \xrightarrow{\text{in}, R} t$ if s is reduced to t by contracting an innermost redex.

For a reduction relation \rightarrow , a sequence s_0, s_1, \dots is an \rightarrow -*reduction sequence* if $s_i \rightarrow s_{i+1}$ for all $i = 0, 1, \dots$. A term t is \rightarrow -*terminating* if there exists no infinite \rightarrow -reduction sequence starting from t . We sometimes write “terminating” (“innermost terminating”) for “ \xrightarrow{R} -terminating” (“ $\xrightarrow{\text{in}, R}$ -terminating”). We say a TRS R is *terminating* (*innermost terminating*) if every term is \xrightarrow{R} -terminating (resp. $\xrightarrow{\text{in}, R}$ -terminating).

Let R be a TRS over a signature \mathcal{F} . We define $\mathcal{F}_D = \{\text{root}(l) \mid l \rightarrow r \in R\}$. We call $f \in \mathcal{F}_D$ a *defined symbol* of R . A term t said to have a *defined root symbol* if $\text{root}(t) \in \mathcal{F}_D$. The signature $\mathcal{F}^\#$ denotes the union of \mathcal{F} and $\mathcal{F}_D^\# = \{f^\# \mid f \in \mathcal{F}_D\}$ where $\mathcal{F} \cap \mathcal{F}_D^\# = \emptyset$ and $f^\#$ has the same arity as f . We use a notation $t^\#$ only if $\text{root}(t) \in \mathcal{F}_D$, and it is defined as $t^\# = f^\#(t_1, \dots, t_n)$ for $t = f(t_1, \dots, t_n)$. If $l \rightarrow r \in R$, u is a subterm of r with a defined root symbol and $u \not\triangleleft l$, then the rewrite rule $l^\# \rightarrow u^\#$ is called a *dependency pair (DP)* for short) of R . The set of all dependency pairs of R is denoted by $\text{DP}(R)$. We use $S^\#$ to represent a TRS consisting of rules in forms of $l^\# \rightarrow r^\#$.

For a reduction relation \rightarrow and a TRS $S^\#$, a (possibly infinite) sequence $s_0^\# \rightarrow t_0^\#, s_1^\# \rightarrow t_1^\#, \dots$ of elements of $S^\#$ is an $(\rightarrow, S^\#)$ -*chain*[†] if there exists a sequence of substitutions τ_0, τ_1, \dots such that $t_i^\# \tau_i$ is \rightarrow -terminating and $t_i^\# \tau_i \rightarrow^* s_{i+1}^\# \tau_{i+1}$ for every $i \geq 0$. We write $(R, S^\#)$ -*chain* for $(\xrightarrow{R}, S^\#)$ -chain. An *innermost* $(R, S^\#)$ -*chain* is an $(\xrightarrow{\text{in}, R}, S^\#)$ -chain such that

[†]To simplify arguments of this paper, we included the condition that $t_i^\# \tau_i$ is \rightarrow -terminating, which means the minimality of chains. However the original proof of Theorem 1 in [10] contains this extended result implicitly.

$s_i^\sharp \tau_i$ is an R -normal form for every $i \geq 0$.

Theorem 1 ([10], [11]): For a TRS R , R is not (innermost) terminating if and only if there exists an infinite (innermost) (R, S^\sharp) -chain for some $S^\sharp \subseteq DP(R)$.

3. Termination and Innermost Termination of a Term for TRSs with Right-Shallow Dependency Pairs

In this section, we show that the termination and the innermost termination of a term are decidable for TRSs all of whose dependency pairs are right-shallow. Note that these properties are different from the termination and the innermost-termination, each of which is undecidable for right-shallow TRSs [2].

In this and next sections, the notation $\xrightarrow{(in),R}$ represents \xrightarrow{R} when arguing termination and $\xrightarrow{in,R}$ when arguing innermost termination. We give proofs only for termination, but they also work for innermost termination by using innermost (R, S^\sharp) -chain and $\xrightarrow{in,R}$ instead of (R, S^\sharp) -chain and \xrightarrow{R} .

The following proposition shows basic properties on reductions and subterms.

Proposition 2: If $t \geq s \xrightarrow{(in),R}^* s'$ then $t \xrightarrow{(in),R}^* t' \geq s'$ for some term t' . Moreover,

1. $t \triangleright s$ implies $t' \triangleright s'$, and
2. $s \xrightarrow{(in),R}^+ s'$ implies $t \xrightarrow{(in),R}^+ t'$.

Proof. Let $t = C[s]$. Then we have $t = C[s] \xrightarrow{(in),R}^* C[s'] \geq s'$. We can take $C[s']$ as t' . Moreover, $C \neq \square$ implies $t' \triangleright s'$, and 2 also holds. \square

For an (innermost-) (R, S^\sharp) -chain $s_0^\sharp \rightarrow t_0^\sharp, \dots, s_n^\sharp \rightarrow t_n^\sharp$, with substitutions τ_0, \dots, τ_n , we say that the chain is *looping* if $s_0^\sharp \rightarrow t_0^\sharp = s_n^\sharp \rightarrow t_n^\sharp$ and $x\tau_0 = x\tau_n$ for all $x \in \text{Var}(s_0^\sharp)$.

We will show that an infinite chain contains a looping chain for right-shallow dependency pairs in Lemma 4. Before this, we prepare a technical lemma that restricts substitutions of chains.

Lemma 3: Let S^\sharp be right-shallow. Let $s_0^\sharp \rightarrow t_0^\sharp, s_1^\sharp \rightarrow t_1^\sharp, \dots$ be an (innermost) (R, S^\sharp) -chain with substitutions τ_0, τ_1, \dots . Then for each $i (\geq 0)$ and $x \in \text{Var}(s_i^\sharp)$ there exists $u \in (\text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})) \cup \{y\tau_0 \mid y \in \text{Var}(s_0^\sharp)\}$ such that $u (\xrightarrow{(in),R}^* \circ \geq) x\tau_i$.

Proof. We use induction on i . Since the case $i = 0$ is trivial, we consider the case $i > 0$. Let $t_{i-1}^\sharp = f^\sharp(u_1, \dots, u_n)$ and $s_i^\sharp = f^\sharp(v_1, \dots, v_n)$. For every $j (1 \leq j \leq n)$, we have $u_j\tau_{i-1} \xrightarrow{(in),R}^* v_j\tau_i$ from the definition of chains.

(Case $u_j \in \mathcal{V}$): Since $u_j \in \text{Var}(t_{i-1}^\sharp) \subseteq \text{Var}(s_{i-1}^\sharp)$, there exists $u \in (\text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})) \cup \{y\tau_0 \mid y \in \text{Var}(s_0^\sharp)\}$ such that $u (\xrightarrow{(in),R}^* \circ \geq) u_j\tau_{i-1}$ by induction hypothesis. Hence $u (\xrightarrow{(in),R}^* \circ \geq) u_j\tau_{i-1} \xrightarrow{(in),R}^* v_j\tau_i \geq x\tau_i$ for all $x \in \text{Var}(v_j)$. Thus we have $u (\xrightarrow{(in),R}^* \circ \geq) x\tau_i$ by Proposition 2.

(Otherwise): $u_j \in \text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})$ from right-shalowness of S^\sharp . Hence, $u_j = u_j\tau_{i-1} \xrightarrow{(in),R}^* v_j\tau_i \geq x\tau_i$ for all $x \in \text{Var}(v_j)$. \square

Lemma 4: Let S^\sharp be right-shallow. Let $s_0^\sharp \rightarrow t_0^\sharp, s_1^\sharp \rightarrow t_1^\sharp, \dots$ be an infinite (innermost) (R, S^\sharp) -chain with substitutions τ_0, τ_1, \dots . Then there exist i and $j (0 \leq i < j)$ such that $s_i^\sharp \rightarrow t_i^\sharp = s_j^\sharp \rightarrow t_j^\sharp$ and $x\tau_i = x\tau_j$ for all $x \in \text{Var}(s_i^\sharp)$.

Proof. We can assume that every rule in S^\sharp appears in the given chain without loss of generality. Then all terms in $(\text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})) \cup \{y\tau_0 \mid y \in \text{Var}(s_0^\sharp)\}$ are terminating from the definition of chains. Hence, the union of ranges of all τ_i is finite by Lemma 3. Since the set $S^\sharp \times \{\tau_0, \tau_1, \dots\}$ is finite, the lemma follows. \square

Proposition 5 ([10]): Let t be a $\xrightarrow{(in),R}$ -non-terminating term. Then there exist a term $t' \leq t$ with a defined root symbol and an infinite (innermost) $(R, DP(R))$ -chain $s_0^\sharp \rightarrow t_0^\sharp, s_1^\sharp \rightarrow t_1^\sharp, \dots$ with substitutions τ_0, τ_1, \dots such that $t' \xrightarrow{(in),R}^* s_0^\sharp\tau_0$.

Proposition 6 ([10]): If a term t with defined root symbol is $\xrightarrow{(in),R}$ -terminating, then t^\sharp is $\xrightarrow{(in),R \cup DP(R)}$ -terminating.

We obtain the decidability of (innermost) termination of a term for right-shallow systems.

Theorem 7: Let R be a TRS such that $DP(R)$ is right-shallow. Then $\xrightarrow{(in),R}$ -termination of a term is decidable.

Proof. We can assume that the given term t is ground by regarding each variable as a fresh constant. Consider the following procedure: for every term $s (\leq t)$ with a defined root symbol, simultaneously generate all $\xrightarrow{(in),R \cup DP(R)}$ -reduction sequences starting from s^\sharp . The procedure terminates if it enumerates all reachable terms exhaustively or it detects a sequence $s^\sharp \xrightarrow{(in),R \cup DP(R)}^* u^\sharp \xrightarrow{(in),R \cup DP(R)}^+ u^\sharp$.

Consider the case that t is not $\xrightarrow{(in),R}$ -terminating. Then there exist a term $s \leq t$ with a defined root symbol and an infinite $(R, DP(R))$ -chain $s_0^\sharp \rightarrow t_0^\sharp, s_1^\sharp \rightarrow t_1^\sharp, \dots$ with substitutions τ_0, τ_1, \dots , such that $s^\sharp \xrightarrow{R}^* s_0^\sharp\tau_0$ by Proposition 5. Then by Lemma 4, there exist integers i and $j (i < j)$ such that $s_i^\sharp \rightarrow t_i^\sharp = s_j^\sharp \rightarrow t_j^\sharp$ and $x\tau_i = x\tau_j$ for all $x \in \text{Var}(s_i^\sharp)$. From the definition of chains, there exists a sequence $s^\sharp \xrightarrow{(in),R}^* s_0^\sharp\tau_0 \xrightarrow{(in),R \cup DP(R)}^* s_i^\sharp\tau_i \xrightarrow{(in),R \cup DP(R)}^+ s_j^\sharp\tau_j = s_i^\sharp\tau_i$, and the procedure detects this sequence.

Consider the case that t is $\xrightarrow{(in),R}$ -terminating. Since each s in the procedure is also $\xrightarrow{(in),R}$ -terminating, all $\xrightarrow{(in),R \cup DP(R)}$ -reduction sequences from s^\sharp are finite by Proposition 6. The enumeration of all reachable terms in the procedure eventually stops since $\xrightarrow{(in),R \cup DP(R)}$ is finitely branching. \square

4. Termination and Innermost Termination for TRSs with Right-Linear Right-Shallow Dependency Pairs

In this section, we show that the termination and the innermost termination are decidable properties for TRSs all of whose dependency pairs are right-linear and right-shallow.

We have shown in Lemma 4 that an infinite chain contains a looping structure if dependency pairs in the chain are right-shallow. We have also shown in Lemma 3 that the ranges of the substitutions of the chain are covered by some finite set determined by the dependency pairs and its initial term $s_0^\# \tau_0$. Since we cannot use information on initial terms for deciding termination of TRSs, we have to determine the set only from TRSs. In order to analyze looping chains, we introduce directed graphs called the argument propagation graphs (APGs) [4]. Nodes of an APG indicate immediate subterms of $s_i^\# \tau_i$ or $t_i^\# \tau_i$ of a looping chain, and edges of an APG represent flows of the immediate subterms in the chain.

Definition 8 (Argument Propagation Graph): Let $S^\#$ be right-shallow. For a looping (innermost) $(R, S^\#)$ -chain $s_0^\# \rightarrow t_0^\#, \dots, s_n^\# \rightarrow t_n^\#$ with substitutions τ_0, \dots, τ_n , the *argument propagation graph (APG)* of the chain is a directed graph $G = \langle \mathcal{N}, \mathcal{E} \rangle$ where

$$\begin{aligned} \mathcal{N} &= \{(i, \text{lhs}, j) \mid 1 \leq i < n, 1 \leq j \leq \text{arity}(\text{root}(s_i^\#))\} \\ &\quad \cup \{(i, \text{rhs}, j) \mid 1 \leq i < n, 1 \leq j \leq \text{arity}(\text{root}(t_i^\#))\} \\ &\quad \cup \{(i, x) \mid x \in \text{Var}(s_i^\#) \setminus \text{Var}(t_i^\#)\} \\ \mathcal{E} &= \{((i, \text{lhs}, j), (i, \text{rhs}, j')) \mid t_i^\#|_j \in \text{Var}(s_i^\#|_j)\} \\ &\quad \cup \{((i, \text{rhs}, j), (i', \text{lhs}, j)) \mid \\ &\quad \quad i + 1 = i' \text{ or } i = n - 1 \wedge i' = 0\} \\ &\quad \cup \{((i, \text{lhs}, j), (i, x)) \mid x \in \text{Var}(s_i^\#|_j)\}. \end{aligned}$$

Example 9: Consider the following TRS and its the dependency pairs.

$$R_1 = \begin{cases} f(x, x, z) \rightarrow g(b, c, z), \\ g(i(x, y), y, z) \rightarrow h(x, y, z), \\ h(j(x), z, z) \rightarrow f(x, a, z), \\ b \rightarrow i(j(a), d), c \rightarrow d \end{cases}$$

$$DP(R_1) = \begin{cases} f^\#(x, x, z) \rightarrow g^\#(b, c, z), \\ f^\#(x, x, z) \rightarrow b^\#, f^\#(x, x, z) \rightarrow c^\#, \\ g^\#(i(x, y), y, z) \rightarrow h^\#(x, y, z), \\ h^\#(j(x), z, z) \rightarrow f^\#(x, a, z), \end{cases}$$

Figure 3 shows the APG of a looping $(R_1, DP(R_1))$ -chain $f^\#(x, x, z) \rightarrow g^\#(b, c, z)$, $g^\#(i(x, y), y, z) \rightarrow h^\#(x, y, z)$, $h^\#(j(x), z, z) \rightarrow f^\#(x, a, z)$, $f^\#(x, x, z) \rightarrow g^\#(b, c, z)$ with substitutions $\{x \mapsto a, z \mapsto d\}$, $\{x \mapsto j(a), y \mapsto d, z \mapsto d\}$, $\{x \mapsto a, z \mapsto d\}$, $\{x \mapsto a, z \mapsto d\}$. \square

The graph Fig.3 can be represented as in Fig.4 attached some more information related to the chain. In the following, we use this enriched form to represent APGs for readability. In order to handle this augmented information formally in proofs, we define a mapping $t^\square : \mathcal{N} \rightarrow \mathcal{T}(\mathcal{F}, \mathcal{V})$

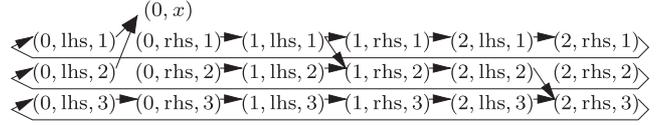


Fig.3 The APG of $(R_1, DP(R_1))$ -chain.

i	0	1	2
$s_i^\# \rightarrow t_i^\#$	$f^\#(x) \rightarrow b$ $g^\#(x, c) \rightarrow y$ $z \rightarrow z$	$g^\#(i(x, y)) \rightarrow y$ $h^\#(x, y) \rightarrow y$ $z \rightarrow z$	$h^\#(j(x)) \rightarrow z$ $f^\#(x) \rightarrow a$ $z \rightarrow z$
$\tau_i(x)$	a	$j(a)$	a
$\tau_i(y)$	-	d	-
$\tau_i(z)$	d	d	d

Fig.4 An enriched representation of the APG in Fig.3.

that returns the term corresponding to a given node:

$$t^N = \begin{cases} (s_i^\#|_j) \cdots N = (i, \text{lhs}, j) \\ (t_i^\#|_j) \cdots N = (i, \text{rhs}, j) \\ x \cdots N = (i, x). \end{cases}$$

Moreover, we also use \tilde{t}^N to denote $t^N \tau_i$. For example, considering $N = (2, \text{lhs}, 1)$, we have $t^N = j(x)$ and $\tilde{t}^N = (j(x))\tau_2 = j(a)$. (See Figs. 3 and 4.)

For a directed graph G , the *in-degree* (*out-degree*) of a node is the number of inward edges (resp. outward edges) of the node. A node is *source* (*sink*) if the in-degree (resp. out-degree) of the node is 0. A *strongly-connected component* of G is a maximal subgraph G' of G such that there is a path from any node to any node. An *undirectional path* is a path regarding edges undirected. An *undirectionally connected component* of G is a maximal subgraph G' of G such that there is an undirectional path between any two nodes. For example, the graph G in Fig.3 has only one strongly-connected component consisting of nodes $\{(i, s, j) \mid i \in \{0, 1, 2\}, s \in \{\text{lhs}, \text{rhs}\}\}$. On the other hand, G itself is the only one undirectionally connected component of G .

In the rest of this section, we analyze properties on substitutions of looping chains by using APGs. Specifically, each term t that belongs to the ranges of the substitutions satisfies the following properties:

- t is a subterm of a term that is reachable from a term in $\text{Arg}(S_{\text{rhs}}^\#) \cap \mathcal{T}(\mathcal{F})$, or
- t is never reduced in the chain,

which correspond to items in the proof of Lemma 13. In the latter case, we can replace t by an arbitrary fixed term (fixed normal form in the innermost case) without destroying the condition of chains. Hence we can cover ranges of substitution of looping chains by a finite set of terms determined by right-hand sides of dependency pairs. This is the key idea of

this section.

The following proposition shows basic properties on APGs.

Proposition 10: Let S^\sharp be right-shallow. Let $s_0^\sharp \rightarrow t_0^\sharp, \dots, s_n^\sharp \rightarrow t_n^\sharp$ be a looping (innermost) (R, S^\sharp) -chain with substitutions τ_0, \dots, τ_n . The APG of the chain satisfies the following properties:

1. If N is a source node, then N is in forms of (i, rhs, j) and $t^N = t_i^\sharp|_j \in \text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})$.
2. For every edge from (i, rhs, j) to (i', lhs, j) , we have $(t_i^\sharp|_j)\tau_i \xrightarrow{(in,R)^*} (s_{i'}^\sharp|_j)\tau_{i'}$.
3. For every edge from (i, lhs, j) to (i, rhs, j') , we have $(s_i^\sharp|_j)\tau_i \triangleright (t_i^\sharp|_{j'})\tau_i$. Moreover
 - a. if $s_i^\sharp|_j$ is not a variable then $(s_i^\sharp|_j)\tau_i \triangleright (t_i^\sharp|_{j'})\tau_i$, and
 - b. if $s_i^\sharp|_j$ is a variable then $(s_i^\sharp|_j)\tau_i = (t_i^\sharp|_{j'})\tau_i$.
4. If S^\sharp is right-linear and the out-degree of N is 2 or more, then $N = (i, \text{lhs}, j)$ and $t^N = s_i^\sharp|_j$ is not a variable.

Proof. For the property 1, N is not in forms of (i, lhs, j) because there exists a node (k, rhs, j) for $k = i - 1$ or $i = 1 \wedge k = n - 1$. The node N is not in forms of (i, x) because there exists a node (i, lhs, j) such that $x \in \text{Var}(s_i^\sharp|_j)$. Let $N = (i, \text{rhs}, j)$. Then $t_i^\sharp|_j$ is not variable since N is a source node. Therefore the claim follows from right-shalowness.

In similar to property 1, the other properties are shown by using the definitions of APGs and chains, and right-shallow property. \square

The following lemma shows properties of APGs on paths and cycles.

Lemma 11: Let S^\sharp be right-shallow. Let $s_0^\sharp \rightarrow t_0^\sharp, \dots, s_n^\sharp \rightarrow t_n^\sharp$ be a looping (innermost) (R, S^\sharp) -chain with substitutions τ_0, \dots, τ_n . The APG of the chain satisfies the following properties:

1. For a path N_0, N_1, \dots, N_m in the APG, we have $\tilde{t}^{N_0} \xrightarrow{(in,R)^*} \tilde{t}^{N_m}$. Moreover,
 - a. if t^{N_k} is not a variable for some $0 \leq k < m$ then $\tilde{t}^{N_0} \xrightarrow{(in,R)^*} \tilde{t}^{N_m}$, and
 - b. if t^{N_k} is a variable for every $0 \leq k < m$ then $\tilde{t}^{N_0} \xrightarrow{(in,R)^*} \tilde{t}^{N_m}$.
2. For every cycle in the APG,
 - a. for all nodes N in the cycle, the terms \tilde{t}^N are equal, and
 - b. the out-degree of each node in the cycle is 1, if S^\sharp is right-linear.
3. Every node that has no path from any source node belongs to a cycle, if S^\sharp is right-linear.

Proof. 1: We prove by induction on m that $\tilde{t}^{N_0} \xrightarrow{(in,R)^*} \tilde{t}^{N_m}$.

Since in the case $m = 0$ the claim trivially holds, we consider the case that $m > 0$.

- In the subcase that $N_{m-1} = (i, \text{rhs}, j)$ for some i and j , the node N_m is (i', lhs, j) for some i' . Then we have $\tilde{t}^{N_{m-1}} = (t_i^\sharp|_j)\tau_i \xrightarrow{(in,R)^*} (s_{i'}^\sharp|_j)\tau_{i'} = \tilde{t}^{N_m}$ by Proposition 10-2. Hence the claim follows from induction hypothesis and Proposition 2.
- In the subcase that $N_{m-1} = (i, \text{lhs}, j)$ for some i and j , the node N_m is either (i, rhs, j') or (i, x) . In the former case, the term $t^{N_m} = s_i^\sharp|_j$ is a variable x that occurs in $t^{N_{m-1}} = s_i^\sharp|_j$ and hence. Since x is a subterm of $t^{N_{m-1}}$, we have $\tilde{t}^{N_{m-1}} \triangleright x\tau_i = \tilde{t}^{N_m}$. Therefore the claim follows from the induction hypothesis and the transitivity of \triangleright .

In similar way, properties 1a and 1b can be shown.

2: Let the cycle be N_0, N_1, \dots, N_m where $N_m = N_0$. Before proving 2a and 2b, we show the following claim:

t^{N_k} is a variable for every k . (*)

We assume that t^{N_k} is not a variable for some $0 \leq k < m$. Then we have $\tilde{t}^{N_0} \xrightarrow{(in,R)^*} \tilde{t}^{N_0}$ from 1a of this lemma and $N_0 = N_m$. Here the displayed reduction $\xrightarrow{(in,R)^*}$ must be $\xrightarrow{(in,R)^+}$ since \triangleright is not reflexive. Now we can construct infinite R -reduction sequence starting from \tilde{t}^{N_0} by Proposition 2. However, \tilde{t}^{N_0} is terminating from the definition of chains. This is a contradiction. Hence the claim (*) has been shown.

2a: From (*), 1b of this lemma and $N_0 = N_k$, we have

$u_0 \xrightarrow{(in,R)^*} u_1 \xrightarrow{(in,R)^*} \dots \xrightarrow{(in,R)^*} u_m = u_0$ where $u_k = \tilde{t}^{N_k}$.

Since u_0 is terminating, we have $u_0 = u_1 = \dots = u_k$.

2b: From (*), it follows from Proposition 10-4.

3: Let N be a node having no path from any source node in the APG. Note that N is not a source node. We have

- N belongs to some cycle, or
- N does not belong to any cycle but it is reachable from a node in some cycle.

However the latter case is impossible from 2b of this lemma. \square

We introduce a symbol \perp for representing some fixed term, and use $\text{Cand}(S^\sharp, \rightarrow)$ to denote the set of all candidate instances of S_{lhs}^\sharp for looping chains. The latter is defined as follows by using substitutions whose ranges are \perp or subterms of terms reachable from immediate ground subterms of right-hand side of dependency pairs in S^\sharp .

$$\begin{aligned} \text{Cand}(S^\sharp, \rightarrow) = \\ \text{Inst}(S_{\text{lhs}}^\sharp, (\rightarrow^* \circ \triangleright)[\text{Arg}(S_{\text{rhs}}^\sharp) \cap \mathcal{T}(\mathcal{F})] \cup \{\perp\}) \end{aligned}$$

Example 12: Consider R_1 in Example 9. We have $\text{Arg}(\text{DP}(R_1)_{\text{rhs}}) \cap \mathcal{T}(\mathcal{F}) = \{a, b, c\}$ and $\text{Cand}(\text{DP}(R_1), \xrightarrow{R_1}) = \{f^\sharp(a, a, a), f^\sharp(a, a, b), f^\sharp(a, a, c), f^\sharp(a, a, d), \dots\}$. \square

By the previous observations on APGs, we obtain the following lemma on an existence of looping sequence starting from a candidate instance.

Lemma 13: Let S^\sharp be right-linear and right-shallow. If there exists a looping (innermost) (R, S^\sharp) -chain then there exists a term $t^\sharp \in \text{Cand}\left(S^\sharp, \xrightarrow{(in,R)}\right)$ such that $t^\sharp \xrightarrow{(in,R \cup S^\sharp)}^+ t^\sharp$.

Proof. Let $s_0^\sharp \rightarrow t_0^\sharp, \dots, s_n^\sharp \rightarrow t_n^\sharp$ be a looping (innermost) (R, S^\sharp) -chain with substitutions τ_0, \dots, τ_n . We construct new substitutions τ'_i for $0 \leq i < n$ such that $\text{Dom}(\tau'_i) = \text{Var}(s_i^\sharp)$ as follows:

- $x\tau'_i = x\tau_i$ for $x \in \text{Var}(s_i^\sharp|_j)$ such that there exists a path from a source node to (i, lhs, j) in the APG of the chain,
- $x\tau'_i = \perp$ for the other $x \in \text{Var}(s_i^\sharp)$.

We also take $\tau'_0 = \tau'_n$.

Now we show that the lemma holds by taking $s_0^\sharp \tau'_0$ as t^\sharp .

Consider each variable $x \in \text{Var}(s_0^\sharp)$. If $x \in \text{Var}(s_0^\sharp|_j)$ for some j and the node has a path from a source node N , we obtain that $x\tau'_0 = x\tau_0$ is reachable by $(\xrightarrow{(in,R)}^* \circ \triangleright)$ from a term $v \in \text{Arg}(S^\sharp_{\text{rhs}}) \cap \mathcal{T}(\mathcal{F})$ by using Proposition 10-1 and Lemma 11-1. If none of nodes corresponding to x has a path from a source node, we have $x\tau'_i = \perp$. In either of two cases, we have $t^\sharp = s_0^\sharp \tau'_0 \in \text{Cand}\left(S^\sharp, \xrightarrow{(in,R)}\right)$.

Moreover, since every node corresponding to the variable x in the second case has no path from a source node, it belongs to a cycle by Lemma 11-3. Hence the term $x\tau_i$ is never reduced by Lemma 11-2a and can be replaced by \perp . Thus the substitutions τ'_0, \dots, τ'_n satisfy the condition of chains, that is, we have a sequence $s_0^\sharp \tau'_0 \xrightarrow{(in,R \cup S^\sharp)}^+ s_n^\sharp \tau'_n = s_0^\sharp \tau'_0$ from the definition of chains. \square

Example 14: For R_1 in Example 9, there exists a looping sequence $f^\sharp(a, a, d) \xrightarrow{\text{DP}(R_1)} g^\sharp(b, c, d) \xrightarrow{R_1}^* g^\sharp(i(j(a), d), d, d) \xrightarrow{\text{DP}(R_1)} h^\sharp(j(a), d, d) \xrightarrow{\text{DP}(R_1)} f^\sharp(a, a, d)$ starting from $f^\sharp(a, a, d) \in \text{Cand}\left(\text{DP}(R_1), \xrightarrow{\text{DP}(R_1)}\right)$. \square

We obtain the decidability of (innermost) termination for right-linear and right-shallow systems.

Theorem 15: The termination and the innermost termination of a TRS R are decidable if $\text{DP}(R)$ is right-linear and right-shallow.

Proof. A decision procedure for (innermost) termination of R is given as follows:

Step 1: Decide $\xrightarrow{(in,R)}$ -termination of each $t \in \text{Arg}(\text{DP}(R)_{\text{rhs}}) \cap \mathcal{T}(\mathcal{F})$. If there exists a non-terminating term then the procedure outputs “ R is non-terminating” and halts.

Step 2: For all term $u^\sharp \in \text{Cand}\left(\text{DP}(R), \xrightarrow{(in,R)}\right)$, simulta-

neously generate all $(R \cup \text{DP}(R))$ -reduction sequences starting from u^\sharp .

- If it enumerates all reachable terms exhaustively then it outputs “ R is terminating” and halts.
- If it detects a looping sequence $u^\sharp \xrightarrow{(in,R \cup \text{DP}(R))}^+ u^\sharp$ then it outputs “ R is non-terminating” and halts.

By Theorem 7, Step 1 is possible. At Step 2, $\text{Cand}\left(\text{DP}(R), \xrightarrow{(in,R)}\right)$ is finite because all terms in $\text{Arg}(\text{DP}(R)_{\text{rhs}}) \cap \mathcal{T}(\mathcal{F})$ are terminating.

If R is not terminating then there exists a looping dependency chain by Theorem 1 and Lemma 4. Hence a non-terminating term is found at Step 1 or else a looping sequence is found at Step 2(b) by Lemma 13.

If R is terminating, all $\xrightarrow{(in,R \cup \text{DP}(R))}$ -reduction sequences starting from u^\sharp are finite by Proposition 6. Hence the execution of the reduction sequence generation eventually stops since $\xrightarrow{(in,R \cup \text{DP}(R))}$ is finitely branching. Thus the procedure detects the termination of R after finitely many steps. \square

Example 16: The procedure in the proof of Theorem 15 works for R_1 as follows: All terms in $\text{Arg}(\text{DP}(R_1)_{\text{rhs}}) \cap \mathcal{T}(\mathcal{F}) = \{a, b, c\}$ are terminating. By search of a looping sequence from terms in $\text{Cand}\left(\text{DP}(R_1), \xrightarrow{R_1}\right)$, the procedure stops due to detecting the looping sequence $f^\sharp(a, a, d) \xrightarrow{R_1 \cup \text{DP}(R_1)}^+ f^\sharp(a, a, d)$ and outputs “non-terminating”. \square

5. Innermost Termination for TRSs with Shallow Dependency Pairs

In this section, we show that the innermost termination is decidable for TRSs all of whose dependency pairs are shallow. Similarly to the previous section, we use APGs for proving the looping property and use an exhaustive search for the decision procedure.

Example 17: Let $R_2 = \{f(x, y, z) \rightarrow g(x, x, a, z), g(x, b, b, z) \rightarrow f(x, a, z, z), a \rightarrow b\}$. We show the APG of a looping innermost $(R_2, \text{DP}(R_2))$ -chain $f^\sharp(x, y, y, z) \rightarrow g^\sharp(x, x, a, z), g^\sharp(x, b, b, z) \rightarrow f^\sharp(x, a, z, z), f^\sharp(x, y, y, z) \rightarrow g^\sharp(x, x, a, z)$ with substitutions $\{x \mapsto b, y \mapsto b, z \mapsto b\}, \{x \mapsto b, z \mapsto b\}, \{x \mapsto b, y \mapsto b, z \mapsto b\}$ in Fig. 5. \square

Note that proper subterms of $s_i^\sharp \tau_i$ are R -normal forms and hence terms substituted to variables are also R -normal forms in innermost chains. This property plays an important role in this section.

The following shows that all instances of variables corresponding to nodes unidirectionally connected are equal.

Lemma 18: Let S^\sharp be shallow. Let $s_0^\sharp \rightarrow t_0^\sharp, \dots, s_n^\sharp \rightarrow t_n^\sharp$ be a looping innermost (R, S^\sharp) -chain with substitutions τ_0, \dots, τ_n . Let G be an unidirectionally connected component of the APG of the chain. Then terms \tilde{t}^N are equal and in normal forms for all non-source nodes N of G .

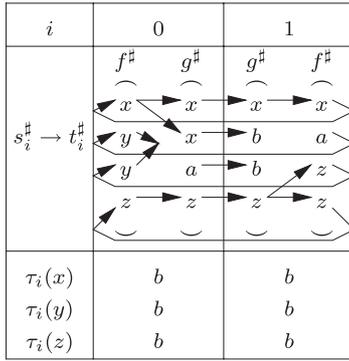


Fig. 5 The APG of innermost- $(R_2, DP(R_2))$ -chain.

Proof. For a non-source node N , it is in forms of (i, lhs, j) or t^N is a variable. In either of cases \tilde{t}^N is a normal form.

Consider non-source nodes N and N' having an edge from N to N' . In the case that $N = (i, \text{rhs}, j)$ and $N' = (i', \text{lhs}, j)$ for some i, i' and j , we have $(t_i^\#|_j)\tau_i \xrightarrow{(in,R)^*} (s_{i'}^\#|_j)\tau_{i'}$ by Proposition 10-2. Since N is a non-source node, the term $(t_i^\#|_j)\tau_i$ is a normal form and hence $\tilde{t}^N = (t_i^\#|_j)\tau_i = (s_{i'}^\#|_j)\tau_{i'} = \tilde{t}^{N'}$. In the case that $N = (i, \text{lhs}, j)$ and $N' = (i, \text{rhs}, j') \vee N' = (i, x)$, terms \tilde{t}^N and $\tilde{t}^{N'}$ are the same variable x since N' is a non-source node. Thus $\tilde{t}^N = x\tau_i = \tilde{t}^{N'}$. \square

We use $\text{Cand}_{in}(S^\#, \rightarrow)$ for candidate terms in innermost case:

$$\begin{aligned} \text{Cand}_{in}(S^\#, \rightarrow) = \\ \text{Inst}(S_{\text{lhs}}^\#, \rightarrow^*[\text{Arg}(S_{\text{rhs}}^\#) \cap \mathcal{T}(\mathcal{F})]) \\ \cup (\text{Arg}(S_{\text{lhs}}^\#) \cap \mathcal{T}(\mathcal{F})) \cup \{\perp\}. \end{aligned}$$

From preceding observations of APGs, we obtain the following lemma on an existence of looping sequence. A difference from Lemma 13 is an existence of a node not reachable from any source node. However, such nodes are never reduced, and equal to a reachable sink node or can be replaced by \perp .

Lemma 19: Let $S^\#$ be shallow. If there exists a looping innermost $(R, S^\#)$ -chain then there exists a term $t^\# \in \text{Cand}_{in}(S^\#, \xrightarrow{in,R})$ such that $t^\# \xrightarrow{in,R \cup S^\#}^+ t^\#$.

Proof. Let $s_0^\# \rightarrow t_0^\#, \dots, s_n^\# \rightarrow t_n^\#$ be a looping innermost $(R, S^\#)$ -chain with substitutions τ_0, \dots, τ_n . We construct new substitutions τ'_i for $0 \leq i < n$ such that $\text{Dom}(\tau'_i) = \text{Var}(s_i^\#)$ as follows:

1. $x\tau'_i = x\tau_i$ for $x = (s_i^\#|_j)$ whose corresponding nodes belong to an unidirectionally connected component G of the APG such that
 - a. G has a sink node in forms of (i', lhs, j') , or
 - b. G has a source node.
2. $x\tau'_i = \perp$ for the other $x \in \text{Var}(s_i^\#)$.

We also take τ'_0 as τ'_n .

Now we show that the lemma holds by taking $s_0^\#\tau'_0$ as $t^\#$.

Consider each variable $x \in \text{Var}(s_0^\#)$. In the case that there exists a node corresponding to x that satisfies 1a, let N be one of the sink nodes. Then $t^N \in \text{Arg}(S_{\text{lhs}}^\#) \cap \mathcal{T}(\mathcal{F})$ from left-shalowness. Thus we have $x\tau'_0 = t^N \in \text{Arg}(S_{\text{lhs}}^\#) \cap \mathcal{T}(\mathcal{F})$ by Lemma 18. In the case 1b, let N be one of the source nodes. Then we obtain that $x\tau'_0$ is reachable by $\xrightarrow{in,R}^*$ from a term $v \in \text{Arg}(S_{\text{rhs}}^\#) \cap \mathcal{T}(\mathcal{F})$ by using Proposition 10-1, Proposition 10-2 and Lemma 18. In the case 2 we have $x\tau'_0 = \perp$. In all of the cases, we have $t^\# = s_0^\#\tau'_0 \in \text{Cand}_{in}(S^\#, \xrightarrow{in,R})$.

Moreover, every node corresponding to the variable x in the case 2 belongs to a unidirectionally connected component having neither source node nor sink node in forms of (i', lhs, j') . Hence the term $x\tau_i$ is never reduced by Lemma 18 and can be replaced by \perp . Thus the substitutions τ'_0, \dots, τ'_n satisfy the condition of innermost chains, that is, we have a sequence $s_0^\#\tau'_0 \xrightarrow{in,R \cup S^\#}^+ s_n^\#\tau'_n = s_0^\#\tau'_0$ from the definition of chains.

Therefore the lemma holds by taking $s_0^\#\tau'_0$ as $t^\#$. \square

We obtain the decidability of innermost termination for shallow systems.

Theorem 20: The innermost termination of a TRS R is decidable if $DP(R)$ is shallow.

Proof. We can prove the theorem similarly to the proof of Theorem 15 by using Lemma 19 and $\text{Cand}_{in}(S^\#, \xrightarrow{in,R})$, instead of Lemma 13 and $\text{Cand}(S^\#, \xrightarrow{(in,R)})$, respectively. \square

6. Combining Other Dependency Pair Techniques

In this section, we extend the classes shown in Sects. 4 and 5 by combining the preceding results with the technique on right-ground dependency pairs for semi-constructor TRSs [4], [12] and the other techniques for termination proof based on dependency pairs. The observations on right-ground dependency pairs are very similar to those in the preceding sections as shown in the following lemma.

Lemma 21 ([4], [12]): If there exists a (innermost) $(R, S^\#)$ -chain that contains infinite use of right-ground dependency pairs then there exists a sequence $u^\# \xrightarrow{(in,R \cup S^\#)}^+ u^\#$ for some $u^\# \in S_{\text{rhs}}^\# \cap \mathcal{T}(\mathcal{F})$.

We prove the following useful theorem.

Theorem 22: Let R be a TRS and $DP_{\text{nrg}}(R) (\subseteq DP(R))$ be the set of non-right-ground dependency pairs of R . The (innermost) termination property is decidable for the class of TRSs satisfying the following condition:

The existence of an infinite $(R, DP_{\text{nr}}(R))$ -chain if and only if the existence of an infinite $(R, DP_{\text{nr}}(R))$ -chains such that all dependency pairs in the chain are right-linear and right-shallow, (or shallow for innermost case).

Although the condition in the theorem is undecidable, lots of technique in the framework called dependency pair processors [13] are available to show the condition. For example, (approximated) dependency graphs, subterm criterion, argument filtering, reduction pairs and usable rules are known [10], [11], [14].

Before proving this theorem, we need a lemma corresponding to Theorem 7.

Lemma 23: For a TRS R satisfying the condition of Theorem 22, $\xrightarrow{(in),R}$ -termination of a term is decidable.

Proof. We can decide termination of t by exactly the same procedure in the proof of Theorem 7.

If t is not $\xrightarrow{(in),R}$ -terminating, there exist $S^\sharp (\subseteq DP(R))$ and an infinite (innermost) (R, S^\sharp) -chain $s_0^\sharp \rightarrow t_0^\sharp, \dots$ with substitutions τ_0, \dots such that

- all elements in $S^\sharp (\subseteq DP(R))$ appear infinitely in the chain, and
- $t'^\sharp \xrightarrow{(in),R \cup DP(R)}^* s_0^\sharp \tau_0$ for some $t' \leq t$.

by Proposition 5. This is possible because if an element in S^\sharp appears only finitely many then we can take $s_i^\sharp \rightarrow t_i^\sharp, \dots$ so that the element does not appear in the resulted sequence and remove the element from S^\sharp .

If S^\sharp contains right-ground dependency pairs in $DP(R)$, there exists a term $u^\sharp \in S_{\text{rhs}}^\sharp \cap \mathcal{T}(\mathcal{F})$, such that $t'^\sharp \xrightarrow{(in),R \cup DP(R)}^* u^\sharp \xrightarrow{(in),R \cup DP(R)}^+ u^\sharp$ from Lemma 21. Otherwise dependency pairs in S^\sharp are all right-shallow from the condition of Theorem 22. Therefore the rest of the proof can be done in a same way to Theorem 7. \square

Now we give a proof for Theorem 22.

Proof. (Theorem 22): By Lemma 23, we can provide a similar procedure in the proof of Theorem 15 using a set $\text{Cand}\left(DP_{\text{nr}}(R), \xrightarrow{R}\right) \cup (DP(R)_{\text{rhs}} \cap \mathcal{T}(\mathcal{F}))$ instead of $\text{Cand}\left(DP(R), \xrightarrow{R}\right)$. (For innermost case, use $\text{Cand}_{in}\left(DP_{\text{nr}}(R), \xrightarrow{in,R}\right) \cup (DP(R)_{\text{rhs}} \cap \mathcal{T}(\mathcal{F}))$).

If R is not (innermost) terminating then there exists an infinite (innermost) (R, S^\sharp) -chain for some $S^\sharp (\subseteq DP(R))$ such that all elements in S^\sharp appear infinitely in the chain by Theorem 1 and the similar argument of the proof in Lemma 23. If S^\sharp contains a right-ground dependency pair, there exists a term $u^\sharp \in (DP(R)_{\text{rhs}} \cap \mathcal{T}(\mathcal{F}))$ such that $u^\sharp \xrightarrow{(in),R \cup DP(R)}^+ u^\sharp$ from Lemma 21. Otherwise all dependency pairs in S^\sharp are right-linear and right-shallow (or shallow for innermost case). Therefore the rest of the proof can be done in a same way to Theorem 15 (Theorem 20 for innermost case). \square

Example 24: Consider the following TRS that defines the factorial in unary representation of natural numbers having a mistake.

$$R_3 = \begin{cases} \text{sum}(x, 0) \rightarrow x, \\ \text{sum}(x, s(y)) \rightarrow s(\text{sum}(x, y)), \\ \text{mult}(x, 0) \rightarrow 0, \\ \text{mult}(x, s(y)) \rightarrow \text{sum}(\text{mult}(x, y), x), \\ \text{fact}(0) \rightarrow s(0), \\ \text{fact}(x) \rightarrow \text{mult}(s(x), \text{fact}(x)) \end{cases}$$

By simple dependency analysis [10], we know that possible infinite chains may contains the following right-linear right-shallow dependency pairs.

$$S^\sharp = \begin{cases} \text{sum}^\sharp(x, s(y)) \rightarrow \text{sum}^\sharp(x, y), \\ \text{mult}^\sharp(x, s(y)) \rightarrow \text{mult}^\sharp(x, y), \\ \text{fact}^\sharp(x) \rightarrow \text{fact}^\sharp(x) \end{cases}$$

From Theorem 22, we can decide its (innermost) termination. Actually $\text{Cand}\left(S^\sharp, \xrightarrow{(in),R_3}\right) = \{\perp\}$ and we can easily find a looping sequence $\text{fact}(\perp) \xrightarrow{(in),R \cup S^\sharp} \text{fact}(\perp)$. \square

7. Undecidability of Termination for TRSs with Left-Linear Shallow Dependency Pairs

Right-linear right-shallow TRSs [2] and left-linear shallow TRSs [4] are known to be decidable classes of termination. We have extended the former class to TRSs consisting of right-linear right-shallow dependency pairs in Sect. 4. In this section, we show that the extension of the latter class to TRSs consisting of left-linear shallow dependency pairs is impossible.

Definition 25: An *instance* of PCP is a finite set $P \subseteq \mathcal{A}^* \times \mathcal{A}^*$ of finite pairs of non-empty strings over an alphabet \mathcal{A} with at least two symbols. A *solution* of P is a non-empty string w such that $w = u_1 \cdots u_k = v_1 \cdots v_k$ for some $(u_1, v_1), \dots, (u_k, v_k) \in P$. The *Post's correspondence problem* (PCP) is a problem to decide whether such a solution exists or not.

Theorem 26 ([15]): PCP is undecidable.

Theorem 27: Termination is undecidable for TRSs all of whose dependency pairs are left-linear and shallow.

Proof. Let $\{(u_i, v_i) \in \mathcal{A}^* \times \mathcal{A}^* \mid 1 \leq i \leq n\}$ be an instance of PCP. We identify strings in \mathcal{A}^* with terms with unary symbols[†]. We use a notation $g^k(t)$ for k times application of g to t .

A transform of the instance into a TRS is described as follows:

$$R_4 = \begin{cases} \{f(x, b(\varepsilon, \varepsilon), c) \rightarrow f(x, x, x), a(x, y) \rightarrow b(x, y), \\ g(a(x, x)) \rightarrow c, g(c) \rightarrow c\} \\ \cup \{g(b(u_i(x), v_i(y))) \rightarrow b(x, y) \mid 1 \leq i \leq n\} \end{cases}$$

[†]For example $011(x)$ represents $0(1(1(x)))$.

Since the only one dependency pair of R is $f^\sharp(x, b(\varepsilon, \varepsilon), c) \rightarrow f^\sharp(x, x, x)$, which is left-linear and shallow. It is enough to show that R is non-terminating if and only if the instance has a solution.

(\Leftarrow): Let $w = u_{i_1} \dots u_{i_k} = v_{i_1} \dots v_{i_k}$ ($k > 0$) be a solution of the instance of PCP and t be a term $g^k(a(w(\varepsilon), w(\varepsilon)))$. Then we have an infinite sequence $f(t, t, t) \xrightarrow{R^+} f(t, g^k(b(w(\varepsilon), w(\varepsilon))), g^{k-1}(c)) \xrightarrow{R^+} f(t, b(\varepsilon, \varepsilon), c) \xrightarrow{R} f(t, t, t) \xrightarrow{R^+} \dots$.

(\Rightarrow): If R is non-terminating then there exists a \xrightarrow{R} -terminating term t such that $f^\sharp(t, t, t) \xrightarrow{R^*} f^\sharp(t, b(\varepsilon, \varepsilon), c) \xrightarrow{DP(R)} f^\sharp(t, t, t) \xrightarrow{R^*} \dots$ by Theorem 1 and the definition of chains. Since t must be reachable to both $b(\varepsilon, \varepsilon)$ and c , the term t must be in forms of $g^k(a(w(\varepsilon), w(\varepsilon)))$ for some integer $k > 0$ and string $w \in \mathcal{A}^+$. Note that $k \neq 0$ because $a(w(\varepsilon), w(\varepsilon))$ is not reachable to c , and that $w \neq \varepsilon$ because $g^k(a(\varepsilon, \varepsilon))$ with $k > 0$ is not reachable to $b(\varepsilon, \varepsilon)$. Moreover t is reachable to $b(\varepsilon, \varepsilon)$. Therefore w is a solution of the instance of PCP. \square

8. Conclusion

In this paper, we have shown the followings.

1. The termination and the innermost termination of a term are decidable properties for TRSs all of whose dependency pairs are right-shallow. (Theorem 7)
2. The termination and the innermost termination properties are decidable for TRSs all of whose dependency pairs are right-linear and right-shallow. (Theorem 15)
3. The innermost termination is decidable for TRSs all of whose dependency pairs are shallow. (Theorem 20)
4. An extension of these results by combining with the result of semi-constructor TRSs and other techniques related to dependency pairs. (Theorem 22)
5. The termination is undecidable for TRSs all of whose dependency pairs are left-linear and shallow. (Theorem 27)

Theorem 22 is useful because it guarantees that decision procedures in this paper can be incorporated into termination provers as a dependency pair processor and may improve their efficiency.

Termination provers such as AProVE [16] have been developed and improved capabilities to prove and disprove (innermost) termination. They look like to work as a decision procedure for classes treated in this paper. Thus it is interesting to clarify properties in this direction. Especially it is interesting topic whether narrowing based method [17] to show non-termination eventually halts for classes of TRSs in this paper or not.

It is also interesting to clarify decidability of context-sensitive termination for TRSs proposed in this paper.

Acknowledgment

We would like to thank Nao Hirokawa for giving a helpful comment, which suggest on the looping property, at an

early stage of this work. This work is partly supported by MEXT.KAKENHI #18500011, #20300010 and #20500008.

References

- [1] M. Dauchet, "Simulation of Turing machines by a regular rewrite rule," *Theore. Comput. Sci.*, vol.103, pp.409–420, 1992.
- [2] G. Godoy, E. Huntingford, and A. Tiwari, "Termination of rewriting with right-flat rules," 18th International Conference on Rewriting Techniques and Applications, *Lect. Notes Comput. Sci.*, vol.4533, pp.200–213, 2007.
- [3] M. Sakai and Y. Wang, "Undecidable properties on length-two string rewriting systems," *Electric Notes in Theor. Comput. Sci.*, vol.204, pp.53–69, 2008.
- [4] Y. Wang and M. Sakai, "Decidability of termination for semi-constructor TRSs, left-linear shallow TRSs and related systems," 17th International Conference on Rewriting Techniques and Applications, *Lect. Notes Comput. Sci.*, vol.4098, pp.343–356, 2006.
- [5] N. Dershowitz, "Termination of linear rewriting systems," 8th International Colloquium on Automata, Languages and Programming, *Lect. Notes Comput. Sci.*, vol.115, pp.448–458, 1981.
- [6] G. Godoy and A. Tiwari, "Termination of rewrite systems with shallow right-linear, collapsing, and right-ground rules," 20th International Conference on Automated Deduction, *Lect. Notes Comput. Sci.*, vol.3632, pp.164–176, 2005.
- [7] G. Huet and D. Lankford, "On the uniform halting problem for term rewriting systems," *Tech. Rep.*, No.283, INRIA, 1978.
- [8] T. Nagaya and Y. Toyama, "Decidability for left-linear growing term rewriting systems," *Inf. Comput.*, vol.178, pp.499–514, 2002.
- [9] F. Baader and T. Nipkow, *Term Rewriting and All That*, Cambridge University Press, 1998.
- [10] T. Arts and J. Giesl, "Termination of term rewriting using dependency pairs," *Theor. Comput. Sci.*, vol.236, pp.133–178, 2000.
- [11] N. Hirokawa and A. Middeldorp, "Dependency pairs revisited," 15th International Conference on Rewriting Techniques and Applications, *Lect. Notes Comput. Sci.*, vol.3091, pp.249–268, 2004.
- [12] K. Uchiyama, M. Sakai, and T. Sakabe, "Decidability of innermost termination and context-sensitive termination for semi-constructor term rewriting systems," *Electric Notes in Theor. Comput. Sci.*, vol.204, pp.21–34, 2008.
- [13] J. Giesl and D. Kapur, "Mechanizing and improving dependency pairs," *J. Automated Reasoning*, vol.37, no.3, pp.155–203, 2006.
- [14] J. Giesl and T. Arts, "Modular termination proofs for rewriting using dependency pairs," *J. Symbolic Computation*, vol.34, pp.21–58, 2002.
- [15] E.L. Post, "A variant of a recursively unsolvable problem," *Bulletin of the American Mathematical Society*, vol.52, pp.264–268, 1946.
- [16] J. Giesl, P. Schneider-Kamp, and R. Thiemann, "AProVE 1.2: Automatic termination proofs in the dependency pair framework," 3rd International Joint Conference on Automated Reasoning, *Lect. Notes Comput. Sci.*, vol.4130, pp.281–286, 2006.
- [17] J. Giesl, R. Thiemann, and P. Schneider-Kamp, "Proving and disproving termination of higher-order functions," 5th International Workshop on Frontiers of Combining Systems, *Lect. Notes Artif. Intell.*, vol.3717, pp.216–231, 2005.



Keita Uchiyama received the B.E. and M.E. degrees from Nagoya University in 2007 and 2009, respectively. He engaged in research on term rewriting systems. He belongs to Suzuki Motor Corporation since April 2009.



Masahiko Sakai completed graduate course of Nagoya University in 1989 and became Assistant Professor, where he obtained a D.E. degree in 1992. From April 1993 to March 1997, he was Associate Professor in JAIST. In 1996 he stayed at SUNY at Stony Brook for six months as Visiting Research Professor. From April 1997, he was Associate Professor in Nagoya University. Since December 2002, he has been Professor. He is interested in term rewriting system, verification of specification and software generation. He received the Best Paper Award from IEICE in 1992. He is a member of JSSST.



Toshiki Sakabe received B.E., M.E. and D.E. degrees from Nagoya University in 1972, 1974 and 1978, respectively. He was a research associate at Nagoya University during 1977–1985, and an associate professor at Mie University and Nagoya University during 1985–1987 and 1987–1993, respectively. He has been a professor of Nagoya University since 1993. His research interests are in the field of theoretical foundations of software including algebraic specifications, rewriting computation, object-oriented computation, program verification, model checking and so on. He is a member of IPSJ, JSAI and JSSST.