

Decidability of Innermost Termination and Context-Sensitive Termination for Semi-Constructor Term Rewriting Systems

Keita Uchiyama, Masahiko Sakai, Toshiki Sakabe
Graduate School of Information Science Nagoya University

Abstract

Yi and Sakai [9] showed that the termination problem is a decidable property for the class of semi-constructor term rewriting systems, which is a superclass of the class of right ground term rewriting systems. The decidability was shown by the fact that every non-terminating TRS in the class has a loop. In this paper we modify the proof of [9] to show that both innermost termination and μ -termination are decidable properties for the class of semi-constructor TRSs.

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. Since termination is undecidable in general, several decidable classes have been studied [4, 5, 6, 8, 9]. The class of semi-constructor TRSs is one of them [9], where a TRS is in this class if for every right hand sides of rules its all subterms having defined symbol at root position are ground.

Innermost reduction, the strategy which rewrites innermost redexes, is used for call-by-value computation. are indicated by specifying arguments of function symbols. Some non-terminating TRSs are terminating by context-sensitive reduction without loss of computational ability. The termination property with respect to innermost (resp. context-sensitive) reduction is called innermost (resp. context-sensitive) termination. Since innermost termination and context-sensitive termination are also undecidable in general, methods for proving these terminations have been studied [1, 2].

In this paper, we prove that innermost termination and context-sensitive termination for semi-constructor TRSs are decidable properties. The proof is done by using notions of dependency pairs [1, 2].

2 Preliminaries

We assume the reader is familiar with the standard definitions of term rewriting systems [3] and here we just review the main notations used in this paper.

A *signature* \mathcal{F} is a set of function symbols, where every $f \in \mathcal{F}$ is associated with a non-negative integer by an arity function: $\text{arity}: \mathcal{F} \rightarrow \mathbb{N} (= \{0, 1, 2, \dots\})$. The set of all *terms* built from a signature \mathcal{F} and a countable infinite set \mathcal{V} of *variables* such that $\mathcal{F} \cap \mathcal{V} = \emptyset$, is represented by $\mathcal{T}(\mathcal{F}, \mathcal{V})$. The set of *ground terms* is $\mathcal{T}(\mathcal{F}, \emptyset)$ ($\mathcal{T}(\mathcal{F})$ for short). The set of variables occurring in a term t is denoted by $\text{Var}(t)$.

The set of all *positions* in a term t is denoted by $\mathcal{P}\text{os}(t)$ and ε represents the root position. $\mathcal{P}\text{os}(t)$ is: $\mathcal{P}\text{os}(t) = \{\varepsilon\}$ if $t \in \mathcal{V}$, and $\mathcal{P}\text{os}(t) = \{\varepsilon\} \cup \{iu \mid 1 \leq i \leq n, u \in \mathcal{P}\text{os}(t_i)\}$ if $t = f(t_1, \dots, t_n)$. Let C be a *context* with a hole \square . We write $C[t]$ for the term obtained from C by replacing \square with a term t . Especially, we specify an occurrence position p of a hole by $C[\]_p$. We say t is a subterm of s if $s = C[t]$ for some context C . We denote the *subterm relation* by \preceq , that is, $t \preceq s$ if t is a subterm of s , and $t \triangleleft s$ if $t \preceq s$ and $t \neq s$. The *root symbol* of a term t is denoted by $\text{root}(t)$.

A *substitution* θ is a mapping from \mathcal{V} to $\mathcal{T}(\mathcal{F}, \mathcal{V})$ such that the set $\text{Dom}(\theta) = \{x \in \mathcal{V} \mid \theta(x) \neq x\}$ is finite. We usually identify a substitution θ with the set $\{x \mapsto \theta(x) \mid x \in \text{Dom}(\theta)\}$ of variable bindings. In the following, we write $t\theta$ instead of $\theta(t)$.

A *rewrite rule* $l \rightarrow r$ is a directed equation which satisfies $l \notin \mathcal{V}$ and $\text{Var}(r) \subseteq \text{Var}(l)$. A *term rewriting system* TRS is a finite set of rewrite rules. A *redex* is a term $l\theta$ for a rule $l \rightarrow r$ and a substitution θ . A term containing no redex is called a *normal form*. A substitution θ is *normal* if $x\theta$ is in normal forms for every x . The *reduction relation* $\xrightarrow{p}_R \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V}) \times \mathcal{T}(\mathcal{F}, \mathcal{V})$ associated with a TRS R and position p is defined as follows: $s \xrightarrow{p}_R t$ if there exist a rewrite rule $l \rightarrow r \in R$, a substitution θ , and a context $C[\]_p$ such that $s = C[l\theta]_p$ and $t = C[r\theta]_p$, we say that s is reduced to t by contracting redex $l\theta$. We sometimes write \rightarrow_R for \xrightarrow{p}_R by omitting p .

A redex is *innermost*, if its all proper subterms are in normal forms. If s is reduced to t by contracting an innermost redex, then $s \rightarrow_R t$ is said to be an *innermost reduction* denoted by $s \xrightarrow{\text{in}}_R t$.

Proposition 2.1 *For a TRS R , if there is a reduction $s \xrightarrow{\text{in}}_R t$, then $C[s] \xrightarrow{\text{in}}_R C[t]$ for any context C .*

A mapping $\mu : \mathcal{F} \rightarrow \mathcal{P}(\mathbb{N})$ is a *replacement map* (or \mathcal{F} -map) if $\mu(f) \subseteq \{1, \dots, \text{arity}(f)\}$. The set of μ -*replacing positions* $\mathcal{P}\text{os}_\mu(t)$ of a term t is: $\mathcal{P}\text{os}_\mu(t) = \{\varepsilon\}$, if $t \in \mathcal{V}$ and $\mathcal{P}\text{os}_\mu(t) = \{\varepsilon\} \cup \{iu \mid 1 \leq i \leq n, i \in \mu(f), u \in \mathcal{P}\text{os}_\mu(t_i)\}$, if $t = f(t_1, \dots, t_n)$. The set of all μ -*replacing variables* of t is $\text{Var}_\mu(t) = \{x \in \text{Var}(t) \mid \exists p \in \mathcal{P}\text{os}_\mu(t), \exists C, C[x]_p = t\}$. The μ -*replacing subterm relation* \preceq_μ is given by $s \preceq_\mu t$ if there is $p \in \mathcal{P}\text{os}_\mu(t)$ such that $t = C[s]_p$. A context $C[\]_p$ is μ -*replacing* denoted by $C_\mu[\]_p$ if $p \in \mathcal{P}\text{os}_\mu(C[\]_p)$. A *context-sensitive rewriting system* is a TRS with an \mathcal{F} -map. If $s \xrightarrow{p} t$ and $p \in \mathcal{P}\text{os}_\mu(s)$,

then $s \xrightarrow{L} t$ is said to be μ -reduction denoted by $s \xrightarrow{\mu}_R t$.

Let \rightarrow be a binary relation on terms, the transitive closure of \rightarrow is denoted by \rightarrow^+ . The transitive and reflexive closure of \rightarrow is denoted by \rightarrow^* . If $s \rightarrow^* t$, then we say that there is a \rightarrow -sequence starting from s to t or t is \rightarrow -reachable from s . We write $s \rightarrow^k t$ if t is \rightarrow -reachable from s with k steps. A term t terminates with respect to \rightarrow if there exists no infinite \rightarrow -sequence starting from t .

Example 2.2 Let $R_1 = \{g(x) \rightarrow h(x), h(d) \rightarrow g(c), c \rightarrow d\}$ and $\mu_1(g) = \mu_1(h) = \emptyset$ [10]. A μ_1 -reduction sequence starting from $g(d)$ is $g(d) \xrightarrow{\mu_1}_{R_1} h(d) \xrightarrow{\mu_1}_{R_1} g(c)$. We can not reduce $g(c)$ to $g(d)$ because c is not μ_1 -replacing subterm of $g(c)$.

Proposition 2.3 *For a TRS R and \mathcal{F} -map μ , if there is a reduction $s \xrightarrow{\mu}_R t$, then $C_\mu[s] \xrightarrow{\mu}_R C_\mu[t]$ for any μ -replacing context C_μ .*

For a TRS R (and \mathcal{F} -map μ), we say that R terminates (resp. innermost terminates, μ -terminates) if every term terminates with respect to \rightarrow_R (resp. $\xrightarrow{in}_R, \xrightarrow{\mu}_R$).

For a TRS R , a function symbol $f \in \mathcal{F}$ is *defined* if $f = \text{root}(l)$ for some rule $l \rightarrow r \in R$. The set of all defined symbols of R is denoted by $D_R = \{\text{root}(l) \mid l \rightarrow r \in R\}$. A term t has a *defined root symbol* if $\text{root}(t) \in D_R$.

Let R be a TRS over a signature \mathcal{F} . The signature \mathcal{F}^\sharp denotes the union of \mathcal{F} and $D_R^\sharp = \{f^\sharp \mid f \in D_R\}$ where $\mathcal{F} \cap D_R^\sharp = \emptyset$ and f^\sharp has the same arity as f . We call these fresh symbols *dependency pair symbols*. We define a notation t^\sharp by $t^\sharp = f^\sharp(t_1, \dots, t_n)$ if $t = f(t_1, \dots, t_n)$ and $f \in D_R$, $t^\sharp = x$ if $t = x$ and $x \in \mathcal{V}$. If $l \rightarrow r \in R$ and u is a subterm of r with a defined root symbol and $u \not\leq l$, then the rewrite rule $l^\sharp \rightarrow u^\sharp$ is called a *dependency pair* of R . The set of all dependency pairs of R is denoted by $\text{DP}(R)$.

Definition 2.4 (Semi-Constructor TRS) *A term $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ is a semi-constructor term if every term s such that $s \triangleleft t$ and $\text{root}(s) \in D_R$ is ground. A TRS R is a semi-constructor system if r is a semi-constructor term for every rule $l \rightarrow r \in R$.*

Example 2.5 The TRS $R_2 = \{a \rightarrow g(f(a)), f(g(x)) \rightarrow h(f(a), x)\}$ is a semi-constructor TRS.

A TRS R is called *right-ground* if for every $l \rightarrow r \in R$, r is ground.

Proposition 2.6 *The following statements hold:*

1. *Right-ground TRSs are semi-constructor systems.*
2. *For a semi-constructor TRS R , all rules in $\text{DP}(R)$ are right-ground.*

3 Decidability of Innermost Termination for Semi-Constructor TRSs

Decidability of termination for semi-constructor TRSs is proved based on the observation that there exists an infinite reduction sequence having a loop if it is not terminating [9]. In this section, we prove the decidability of innermost termination in similar way.

Definition 3.1 (loop) A reduction sequence *loops* if it contains $t \rightarrow^+ C[t]$ for some context C , and head-loops if containing $t \rightarrow^+ t$.

Proposition 3.2 *If there exists an innermost sequence that loops, then there exists an infinite innermost sequence.*

Definition 3.3 (Innermost DP-chain) For a TRS R , a sequence of the elements of $DP(R)$ $s_1^\# \rightarrow t_1^\#, s_2^\# \rightarrow t_2^\#, \dots$ is an innermost dependency chain if there exist substitutions τ_1, τ_2, \dots such that $s_i^\# \tau_i$ is in normal forms and $t_i^\# \tau_i \xrightarrow{*}_{in\ R} s_{i+1}^\# \tau_{i+1}$ holds for every i .

Theorem 3.4 ([2]) *For a TRS R , R does not innermost terminate if and only if there exists an infinite innermost dependency chain.*

Let $\mathcal{T}_\infty^{\xrightarrow{in\ R}}$ denote the set of all \geq -minimal non-terminating terms for $\xrightarrow{in\ R}$, here “ \geq -minimal” is used in the sense that all its proper subterms terminate.

Definition 3.5 (C-min) For a TRS R , let $\mathcal{C} \subseteq DP(R)$. An infinite reduction sequence in $R \cup \mathcal{C}$ in the form $t_1^\# \xrightarrow{in\ R \cup \mathcal{C}} t_2^\# \xrightarrow{in\ R \cup \mathcal{C}} t_3^\# \xrightarrow{in\ R \cup \mathcal{C}} \dots$ with $t_i \in \mathcal{T}_\infty^{\xrightarrow{in\ R}}$ for all $i \geq 1$ is called a \mathcal{C} -min innermost reduction sequence. We use $\mathcal{C}_{min}^{in}(t^\#)$ to denote the set of all \mathcal{C} -min innermost reduction sequence starting from $t^\#$.

Proposition 3.6 ([2]) *Given a TRS R , the following statements hold:*

1. *If there exists an infinite innermost dependency chain, then $\mathcal{C}_{min}^{in}(t^\#) \neq \emptyset$ for some $\mathcal{C} \subseteq DP(R)$ and $t \in \mathcal{T}_\infty^{\xrightarrow{in\ R}}$.*
2. *For any sequence in $\mathcal{C}_{min}^{in}(t^\#)$, reduction by rules of R takes place below the root while reduction by rules of \mathcal{C} takes place at the root.*
3. *For any sequence in $\mathcal{C}_{min}^{in}(t^\#)$, there is at least one rule in \mathcal{C} which is applied infinitely often.*

Lemma 3.7 ([2]) *For two terms s and s' , $s^\# \xrightarrow{*}_{in\ R \cup \mathcal{C}} s'^\#$ implies $s \xrightarrow{*}_{in\ R} C[s']$ for some context C .*

Proof. We use induction on the number n of reduction steps in $s^\sharp \xrightarrow[n]{in_{R \cup C}} s'^\sharp$. In the case that $n = 0$, it holds with $C = \square$. Let $n \geq 1$. Then we have $s^\sharp \xrightarrow[n-1]{in_{R \cup C}} s''^\sharp \xrightarrow{in_{R \cup C}} s'^\sharp$ for some s''^\sharp . By the induction hypothesis, $s \xrightarrow{in_R}^* C[s'']$.

- Consider the case that $s''^\sharp \xrightarrow{in_R} s'^\sharp$. Since $s'' \xrightarrow{in_R} s'$, we have $C[s''] \xrightarrow{in_R} C[s']$ by Proposition 2.1. Hence $s \xrightarrow{in_R}^* C[s']$.
- Consider the case that $s''^\sharp \xrightarrow{in_C} s'^\sharp$. Since s'' is a normal form with respect to \rightarrow_R , we have $s'' \xrightarrow{in_R} C'[s']$ by the definition of dependency pairs. $C[s''] \xrightarrow{in_R} C[C'[s']]$, by Proposition 2.1. Hence $s \xrightarrow{in_R}^* C[C'[s']]$. \square

Lemma 3.8 *For a semi-constructor TRS R , the following statements are equivalent:*

1. R does not innermost terminate.
2. There exists $l^\sharp \rightarrow u^\sharp \in DP(R)$ such that sq head-loops for some $C \subseteq DP(R)$ and $sq \in C_{min}^{in}(u^\sharp)$.

Proof. ((ii) \Rightarrow (i)) : It is obvious from Lemma 3.7, and Proposition 3.2. ((i) \Rightarrow (ii)) : By Theorem 3.4 there exists an infinite innermost dependency chain. By Proposition 3.6(1), there exists a sequence $sq \in C_{min}^{in}(t^\sharp)$. By Proposition 3.6(2),(3), there exists some rule $l^\sharp \rightarrow u^\sharp \in C$ which is applied at root position in sq infinitely often. By Proposition 2.6(2), u^\sharp is ground. Thus sq contains a subsequence $u^\sharp \xrightarrow{in_{R \cup DP(R)}}^* \rightarrow_{\{l^\sharp \rightarrow u^\sharp\}} u^\sharp$, which head-loops. \square

Theorem 3.9 *Innermost termination of semi-constructor TRSs is decidable.*

Proof. The decision procedure for the innermost termination of a semi-constructor TRS R is as follows: consider all terms u_1, u_2, \dots, u_n corresponding to the right-hand sides of $DP(R) = \{l_i^\sharp \rightarrow u_i^\sharp \mid 1 \leq i \leq n\}$, and simultaneously generate all innermost reduction sequences with respect to R starting from u_1, u_2, \dots, u_n . It halts if it enumerates all reachable terms exhaustively or it detects a looping reduction sequence $u_i \xrightarrow{in_R}^+ C[u_i]$ for some i .

Suppose R does not innermost-terminate. By Lemma 3.8, 3.7, we have a looping reduction sequence $u_i \xrightarrow{in_R}^+ C[u_i]$ for some i and C , which we eventually detect. If R innermost terminates, then the execution of the reduction sequence generation eventually stops since it is finitely branching. Moreover it does not detect a looping sequence, otherwise it contradicts to Proposition 3.2. Thus the procedure decides innermost termination of R in finitely many steps. \square

4 Decidability of Context-Sensitive Termination for Semi-Constructor TRSs

The proof of decidability for innermost termination is straightforward as above. However, the one for context-sensitive termination is not so straightforward because of the existence of dependency pair with a variable in the right-hand side.

Definition 4.1 (μ -Loop) Let \rightarrow be a relation on terms and μ be an \mathcal{F} -map. A reduction sequence μ -loops if it contains $t' \rightarrow^+ C_\mu[t']$ for some context C_μ .

Example 4.2 Let $\mu_2(f) = \mu_2(h) = \{1\}$ and $\mu_2(g) = \emptyset$. For R_2 (in Example 2.5), the μ_2 -reduction sequence $f(a) \xrightarrow{\mu_2}_{R_2} f(g(f(a))) \xrightarrow{\mu_2}_{R_2} h(f(a), f(a)) \xrightarrow{\mu_2}_{R_2} \dots$ is μ_2 -looping.

Proposition 4.3 If there exists a μ -looping μ -reduction sequence, then there exists an infinite μ -reduction sequence.

Definition 4.4 (Context-Sensitive Dependency Pairs [1]) Let R be a TRS and μ be an \mathcal{F} -map. We define $DP(R, \mu) = DP_{\mathcal{F}}(R, \mu) \cup DP_{\mathcal{V}}(R, \mu)$ to be the set of context-sensitive dependency pairs (CS-DPs) where:

$$\begin{aligned} DP_{\mathcal{F}}(R, \mu) &= \{l^\# \rightarrow u^\# \mid l \rightarrow r \in R, u \trianglelefteq_\mu r, \text{root}(u) \in D_R, u \not\triangleleft_\mu l\} \\ DP_{\mathcal{V}}(R, \mu) &= \{l^\# \rightarrow x \mid l \rightarrow r \in R, x \in \text{Var}^\mu(r) \setminus \text{Var}^\mu(l)\} \end{aligned}$$

Example 4.5 Consider TRS R_2 (in Example 2.5) and \mathcal{F} -map μ_2 (in Example 4.2). $DP_{\mathcal{F}}(R_2, \mu_2) = \{f^\#(g(x)) \rightarrow f^\#(a)\}$ and $DP_{\mathcal{V}}(R_2, \mu_2) = \{f^\#(g(x)) \rightarrow x\}$.

Definition 4.6 (Context-Sensitive Semi-Constructor TRS) For an \mathcal{F} -map μ , a TRS R is a μ -semi-constructor TRS if all rules in $DP_{\mathcal{F}}(R, \mu)$ are right-ground.

For a given TRS R and an \mathcal{F} -map μ , we define $\mu^\#$ by $\mu^\#(f) = \mu(f)$ for $f \in \mathcal{F}$, and $\mu^\#(f^\#) = \mu(f)$ for $f \in D_R$. We write $s \triangleright_\mu^\# t$ for $s \triangleright_\mu t$.

Definition 4.7 (Context-Sensitive dependency chain) For a TRS R and \mathcal{F} -map μ , a sequence of the elements of $DP(R, \mu)$ $s_1^\# \rightarrow t_1^\#, s_2^\# \rightarrow t_2^\#, \dots$ is a context-sensitive dependency chain if there exist substitutions τ_1, τ_2, \dots satisfying both:

- $t_i^\# \tau_i \xrightarrow[\mu^\#]{*}_R s_{i+1}^\# \tau_{i+1}$, if $t_i^\# \notin \mathcal{V}$
- $x \tau_i \triangleright_\mu^\# u_i \xrightarrow[\mu^\#]{*}_R s_{i+1}^\# \tau_{i+1}$ for some term u_i , if $t_i^\# = x$.

For a given TRS and an \mathcal{F} -map μ , let $\mathcal{T}_\infty^{\mu^\# R}$ denote the set of all \triangleright_μ -minimal non-terminating terms for $\xrightarrow{\mu^\#}_R$,

Example 4.8 Consider TRS R_2 (in Example 2.5) and \mathcal{F} -map μ_2 (in Example 4.2). $f(a), f(g(f(a))) \in \mathcal{T}_\infty^{\mu_2^\# R_2}$ and $f(f(a)), h(f(a), f(a)) \notin \mathcal{T}_\infty^{\mu_2^\# R_2}$.

Theorem 4.9 ([1]) For a TRS R and an \mathcal{F} -map μ , there exists an infinite context-sensitive dependency chain if and only if R does not μ -terminate.

Let R be a TRS, μ be an \mathcal{F} -map and $\mathcal{C} \subseteq \text{DP}(R, \mu)$. We define $\xrightarrow[\mu]{R, \mathcal{C}}$ as $(\xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{F}}} \cup (\xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{V}}} \cdot \triangleright_{\mu}^\sharp) \cup \xrightarrow[\mu^\sharp]{R})$ where $\mathcal{C}_{\mathcal{F}} = \mathcal{C} \cap \text{DP}_{\mathcal{F}}(R, \mu)$ and $\mathcal{C}_{\mathcal{V}} = \mathcal{C} \cap \text{DP}_{\mathcal{V}}(R, \mu)$.

Definition 4.10 (μ - \mathcal{C} -min) Let R be a TRS, μ be an \mathcal{F} -map, An infinite sequence of terms in the form $t_1^\sharp \xrightarrow[\mu]{R, \mathcal{C}} t_2^\sharp \xrightarrow[\mu]{R, \mathcal{C}} t_3^\sharp \xrightarrow[\mu]{R, \mathcal{C}} \dots$ is called a \mathcal{C} -min μ -sequence if $t_i \in \mathcal{T}_{\infty}^{\mu, R}$ for $i \geq 1$.

We use $\mathcal{C}_{min}^{\mu}(t^\sharp)$ to denote the set of all \mathcal{C} -min μ -sequences starting from t^\sharp .

Example 4.11 The sequence $f^\sharp(a) \xrightarrow[\mu_2]{R_2, \mathcal{C}} f^\sharp(g(f(a))) \xrightarrow[\mu_2]{R_2, \mathcal{C}} f^\sharp(a) \xrightarrow[\mu_2]{R_2, \mathcal{C}} \dots$ is \mathcal{C} -min μ -sequence.

Proposition 4.12 ([1]) Given a TRS R and an \mathcal{F} -map μ , the following statements hold:

1. If there exists an infinite context-sensitive dependency chain, then $\mathcal{C}_{min}^{\mu}(t^\sharp) \neq \emptyset$ for some $\mathcal{C} \subseteq \text{DP}(R, \mu)$ and $t \in \mathcal{T}_{\infty}^{\mu, R}$.
2. For any sequence in $\mathcal{C}_{min}^{\mu}(t^\sharp)$, a reduction with $\xrightarrow[\mu^\sharp]{R}$ takes place below the root while reductions with $\xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{F}}}$ and $\xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{V}}}$ take place at the root.
3. For any sequence in $\mathcal{C}_{min}^{\mu}(t^\sharp)$, there is at least one rule in \mathcal{C} which is applied infinitely often.

Lemma 4.13 For two terms s and t , $s^\sharp \xrightarrow[\mu]{R, \mathcal{C}}^* t^\sharp$ implies $s \xrightarrow[\mu]{R}^* C_{\mu}[t]$ for some context C_{μ} .

Proof. We use induction on length n of the sequence. In the case that $n = 0$, it holds trivially. Let $n \geq 1$. Then we have $s^\sharp \xrightarrow[\mu]{R, \mathcal{C}}^* u^\sharp \xrightarrow[\mu]{R, \mathcal{C}} t^\sharp$ for some u .

- In the case that $u^\sharp \xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{F}}} t^\sharp$, we have $u \xrightarrow[\mu]{R} C'_{\mu}[t]$ by the definition of dependency pairs.
- In the case that $u^\sharp \xrightarrow[\mu^\sharp]{\mathcal{C}_{\mathcal{V}}} v \triangleright_{\mu}^\sharp t^\sharp$, we have $u \xrightarrow[\mu]{R} C''_{\mu}[v]$ by the definition of dependency pairs and $v = C'''_{\mu}[t]$. Thus $u \xrightarrow[\mu]{R} C''_{\mu}[C'''_{\mu}[t]] = C'_{\mu}[t]$.
- In the case that $u^\sharp \xrightarrow[\mu^\sharp]{R} t^\sharp$, we have $u \xrightarrow[\mu]{R} C'_{\mu}[t]$ for $C'_{\mu}[\] = \square$.

Therefore $s \xrightarrow[\mu]{R}^* C_{\mu}[u] \xrightarrow[\mu]{R} C_{\mu}[C'_{\mu}[t]]$ by the induction hypothesis and Proposition 2.3. \square

For a TRS R and \mathcal{F} -map μ , we say R is free from infinite variable dependency chain (FFIVDC) if and only if there exists no infinite context-sensitive dependency chain consists of $\text{DP}_{\mathcal{V}}(R, \mu)$. If R is FFIVDC, then $\mathcal{C}_{min}^{\mu}(t^\sharp) = \emptyset$ for any $\mathcal{C} \subseteq \text{DP}_{\mathcal{F}}(R, \mu)$ and a term t .

Lemma 4.14 Let μ be an \mathcal{F} -map. If μ -semi-constructor TRS R is FFIVDC, then the following statements are equivalent:

1. R does not μ -terminate.

2. There exists $l^\sharp \rightarrow u^\sharp \in DP_{\mathcal{F}}(R, \mu)$ such that sq head-loops for some $sq \in C_{min}^\mu(u^\sharp)$.

Proof. ((ii) \Rightarrow (i)) : It is obvious from Lemma 4.13, and Proposition 4.3. ((i) \Rightarrow (ii)) : By Theorem 4.9 there exists an infinite context-sensitive dependency chain. By Proposition 4.12(1), there exists a sequence $sq \in C_{min}^\mu(t^\sharp)$. By Proposition 4.12(2),(3) and the fact that R is FFIVDC, there is some rule in $l^\sharp \rightarrow u^\sharp \in \mathcal{C}_{\mathcal{F}}$ which is applied at root reduction in sq infinitely often. By definition 4.6, u^\sharp is ground. Thus sq contains a subsequence $u^\sharp \xrightarrow[\mu]{R, C}^+ u^\sharp$, which head-loops and is in $C_{min}^\mu(u^\sharp)$. \square

Lemma 4.15 *Let μ be an \mathcal{F} -map. If μ -semi-constructor TRS R is FFIVDC, then μ -termination of R is decidable.*

Proof. The decision procedure for μ -termination of a μ -semi-constructor TRS R is as follows: consider all terms u_1, u_2, \dots, u_n corresponding to the right-hand sides of $DP_{\mathcal{F}}(R, \mu) = \{l_i^\sharp \rightarrow u_i^\sharp \mid 1 \leq i \leq n\}$, and simultaneously generate all μ -reduction sequences with respect to R starting from u_1, u_2, \dots, u_n . It halts if it enumerates all reachable terms exhaustively or it detects a μ -looping reduction sequence $u_i \xrightarrow[\mu]{R}^+ C_\mu[u_i]$ for some i .

Suppose R does not μ -terminate. By Lemma 4.14, 4.13, we have a μ -looping reduction sequence $u_i \xrightarrow[\mu]{R}^+ C_\mu[u_i]$ for some i and C_μ , which we eventually detect. If R μ -terminates, then the execution of the reduction sequence generation eventually stops since it is finitely branching. Moreover it does not detect a μ -looping sequence, otherwise it contradicts to Proposition 4.3. Thus the procedure decides μ -termination of R in finitely many steps. \square

We have to check FFIVDC property in order to use Lemma 4.15. The following Proposition provides its sufficient condition. The set $DP_{\mathcal{V}}^1(R, \mu)$ is a subset of $DP_{\mathcal{V}}(R, \mu)$ defined as follows

$$DP_{\mathcal{V}}^1(R, \mu) = \{f^\sharp(u_1, \dots, u_k) \rightarrow x \in DP_{\mathcal{V}}(R, \mu) \mid \exists i, 1 \leq i \leq k, i \notin \mu(f), x \in Var(u_i)\}$$

Proposition 4.16 ([1]) *Let R be a TRS, μ be an \mathcal{F} -map and $C = DP_{\mathcal{V}}^1(R, \mu)$. $C_{min}^\mu(t^\sharp) = \emptyset$ for any term t .*

Since $DP_{\mathcal{V}}^1(R, \mu) = DP_{\mathcal{V}}(R, \mu)$ implies that R is FFIVDC by Proposition 4.16, the following theorem directly follows from Lemma 4.15.

Theorem 4.17 *Let μ be an \mathcal{F} -map and R be a μ -semi-constructor TRS. μ -termination of R is decidable if $DP_{\mathcal{V}}(R, \mu) = DP_{\mathcal{V}}^1(R, \mu)$.*

In the following, we show that μ -termination of semi-constructor TRSs (not μ -semi-constructor) is decidable.

Lemma 4.18 Consider a reduction $s^\sharp = C_{\mu^\sharp}[l\theta]_p \xrightarrow{\mu^\sharp_R} t^\sharp = C_{\mu^\sharp}[r\theta]_p = C'[u]_q$ where $s, u \in \mathcal{T}_\infty^{\overrightarrow{\mu}^R}$ and $q \in \mathcal{Pos}(t) \setminus \mathcal{Pos}_\mu(t)$. Then one of the following statements hold

1. $s \triangleright u$
2. $v\theta = u$ and $r = C''[v]_{q'}$ for some $\theta, v \notin \mathcal{V}, C''$, and $q' \in \mathcal{Pos}(r) \setminus \mathcal{Pos}_\mu(r)$

Proof. Since $q \in \mathcal{Pos}(t) \setminus \mathcal{Pos}_\mu(t)$, p is not below or equal to q . In the case that p and q are in parallel positions, $s \triangleright u$ trivially holds. In the case that p is above q , it is obvious that $s \triangleright u$ hold or, $v\theta = u$ and $r = C''[v]_{q'}$ for some $\theta, v \notin \mathcal{V}, C''$. Here the fact that $q' \in \mathcal{Pos}(r) \setminus \mathcal{Pos}_\mu(r)$ follows from $p \in \mathcal{Pos}_\mu(t)$ and $q \notin \mathcal{Pos}_\mu(t)$. \square

Lemma 4.19 Let R be a semi-constructor TRS, μ be an \mathcal{F} -map. For a \mathcal{C} -min μ -sequence $s_1^\sharp \xrightarrow{\mu^\sharp_R^*} t_1^\sharp \xrightarrow{\mu^\sharp_{\mathcal{C}_\mathcal{V}}} u_1 \triangleright_{\mu^\sharp} s_2^\sharp \cdots$ with no reduction by rules in $\mathcal{C}_\mathcal{F}$, one of following statements hold for each i :

1. $s_i \triangleright s_{i+1}$
2. There exists $l^\sharp \rightarrow s_{i+1}^\sharp \in \text{DP}(R)$ for some l

Proof. We have $t_i^\sharp = C[s_{i+1}]_q$ for some $q \in \mathcal{Pos}(t_i) \setminus \mathcal{Pos}_\mu(t_i)$. We show (i) or the following (ii') by induction on the number n of steps of $s_i^\sharp \xrightarrow{\mu^\sharp_R^n} t_i^\sharp = C[s_{i+1}]$.

(ii') There exist a reduction by $l \rightarrow r$ in $s_i^\sharp \xrightarrow{\mu^\sharp_R^*} t_i^\sharp$ and $l^\sharp \rightarrow s_{i+1}^\sharp \in \text{DP}(R)$

- In the case that $n = 0$, trivially $s_i = t_i \triangleright s_{i+1}$.
- In the case that $n > 0$, let $s_i^\sharp \xrightarrow{\mu^\sharp_R} s'^\sharp \xrightarrow{\mu^\sharp_R^{n-1}} t_i^\sharp = C[s_{i+1}]_q$. By the induction hypothesis, the condition (ii') or $s' \triangleright s_{i+1}$ follows. In the former case, it is trivial. In the latter case, by Lemma 4.18, we have $s_i \triangleright s_{i+1}$, or, we have $v\theta = s_{i+1}$ and $r = C'[v]_{q'}$ for some $l \rightarrow r \in R, \theta, v \notin \mathcal{V}, C'$ and $q' \in \mathcal{Pos}(r) \setminus \mathcal{Pos}_\mu(r)$ by Lemma 4.18. Hence $v\theta = v$ due to $\text{root}(s_{i+1}) \in D_R$ and Proposition 2.6(2). Therefore (ii') follows. \square

One may think that the Lemma 4.19 would hold even if $\text{DP}(R)$ were replaced with $\text{DP}(R, \mu)$. However, it does not hold from the following counter example.

Example 4.20 Let $R_3 = \{f(g(x)) \rightarrow x, g(b) \rightarrow g(f(g(b)))\}$, $\mu_3(f) = \{1\}$ and $\mu_3(g) = \emptyset$. There exists a \mathcal{C} -min μ_3 -sequence $f^\sharp(g(b)) \xrightarrow{\mu_3^\sharp_{R_3}} f^\sharp(g(f(g(b)))) \xrightarrow{\mu_3^\sharp_{\mathcal{C}_\mathcal{V}}} f(g(b)) \triangleright_{\mu_3^\sharp} f^\sharp(g(b))$ where $\mathcal{C}_\mathcal{V} = \text{DP}_\mathcal{V}(R_3, \mu_3)$.

A dependency pair whose right-hand side is $f^\sharp(g(b))$ is in $\text{DP}(R)$ but not in $\text{DP}(R, \mu)$.

Lemma 4.21 For a semi-constructor TRS R and an \mathcal{F} -map μ , the following statements are equivalent:

1. R does not μ -terminate.

2. There exists $l^\sharp \rightarrow u^\sharp \in DP(R)$ such that sq head-loops for some $sq \in C_{min}^\mu(u^\sharp)$.

Proof. ((ii) \Rightarrow (i)) : It is obvious from Lemma 4.13, and Prop 4.3. ((i) \Rightarrow (ii)) : By Theorem 4.9 there exists a context-sensitive dependency chain. By Proposition 4.12(1), there exists a sequence $sq \in C_{min}^\mu(t^\sharp)$. By Proposition 4.12(2),(3), there exists a rule in \mathcal{C} applied at root position in sq infinitely often.

- Consider the case that there exists a rule $l^\sharp \rightarrow r^\sharp \in \mathcal{C}_{\mathcal{F}}$ with infinite use in sq . Since u is ground by Proposition 4.12(2) and $\mathcal{C}_{\mathcal{F}} \subseteq DP(R)$, sq has a subsequence $u^\sharp \xrightarrow[\mu]{R, \mathcal{C}}^+ u^\sharp$.

- Otherwise, sq has a infinite subsequence without use of rules in $\mathcal{C}_{\mathcal{F}}$. The subsequence is in $C_{min}^\mu(s^\sharp)$ for some s^\sharp . Then the condition (ii) of Lemma 4.19 holds for infinitely many i 's, otherwise we have an infinite sequence $s_k \triangleright s_{k+1} \triangleright \dots$ for some k , which is a contradiction. Hence there exists a $l^\sharp \rightarrow u^\sharp \in DP(R)$ such that u^\sharp occurs more than twice in sq . Thus the sequence $u^\sharp \xrightarrow[\mu]{R, \mathcal{C}}^+ u^\sharp$ appears in sq . \square

Theorem 4.22 μ -termination of semi-constructor TRSs is decidable.

Proof. The decision procedure for μ -termination of a semi-constructor TRS R is as follows: consider all terms u_1, u_2, \dots, u_n corresponding to the right-hand sides of $DP(R) = \{l_i^\sharp \rightarrow u_i^\sharp \mid 1 \leq i \leq n\}$, and simultaneously generate all μ -reduction sequences with respect to R starting from u_1, u_2, \dots, u_n . It halts if it enumerates all reachable terms exhaustively or it detects a μ -looping reduction sequence $u_i \xrightarrow[\mu]{R}^+ C_\mu[u_i]$ for some i .

Suppose R does not μ -terminate. By Lemma 4.21 and 4.13, we have a μ -looping reduction sequence $u_i \xrightarrow[\mu]{R}^+ C_\mu[u_i]$ for some i and C_μ , which we eventually detect. If R μ -terminates, then the execution of the reduction sequence generation eventually stops since it is finitely branching. Moreover it does not detect a μ -looping sequence, otherwise it contradicts to Proposition 4.3. Thus the procedure decides μ -termination of R in finitely many steps. \square

5 Some Extension and Example

5.1 Innermost

In this subsection, we extend the class for which innermost termination is decidable by using the dependency graph.

Lemma 5.1 *Let R be a TRS whose innermost termination is equivalent to the non-existence of an innermost dependency chain that contains infinite use of right-ground dependency pairs. Then innermost termination of R is decidable.*

Proof. We apply the procedure which is used in proof of Lemma 3.9 starting with terms u_1, u_2, \dots, u_n , where u_i^\sharp 's are all ground right-hand sides of dependency pairs. Suppose R is innermost non-terminating, we have an innermost dependency chain with infinite use of a right-ground dependency pair. Similarly to the semi-constructor case, we have a looping sequence $u_i \xrightarrow{+}_{in R} C[u_i]$, which can be detected by the procedure. \square

Definition 5.2 (Innermost DP-Graph [2]) *The innermost dependency graph of a TRS R is a directed graph whose nodes are the dependency pairs and there is an arc from $s^\sharp \rightarrow t^\sharp$ to $u^\sharp \rightarrow v^\sharp$ if there exist normal substitutions σ and τ such that $t^\sharp\sigma \xrightarrow{*}_{in R} u^\sharp\tau$ and $u^\sharp\tau$ is in normal forms with respect to R .*

An approximated dependency graph is a graph that contains the innermost dependency graph as subgraph. Computable such graphs are proposed in [2], for example.

Theorem 5.3 *Let R be a TRS and G be an approximated dependency graph of R . If at least one node in the cycle is right-ground for every cycle of G , then innermost termination of R is decidable.*

Proof. From Lemma 5.1. \square

Example 5.4 Let $R_3 = \{f(s(x)) \rightarrow g(x), g(s(x)) \rightarrow f(s(0))\}$. Then $DP(R_3) = \{f^\sharp(s(x)) \rightarrow g^\sharp(x), g^\sharp(s(x)) \rightarrow f^\sharp(s(0))\}$. The innermost dependency graph of R_3 has one cycle, which contains a right ground node [Fig. 1]. The innermost termination of R_3 is decidable by Theorem 5.3. Actually we know R_3 is innermost terminating from the procedure in the proof of Theorem 3.9 since all innermost reduction sequences from $f(s(0))$ terminate.

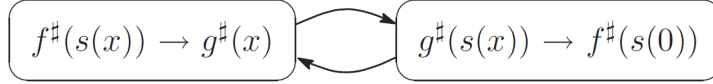


Figure 1: The innermost DP-Graph of R_3

Example 5.5 Let $R_4 = \{a \rightarrow b, f(a, x) \rightarrow x, f(x, b) \rightarrow g(x, x), g(b, x) \rightarrow h(f(a, a), x)\}$. Then $DP(R_4) = \{f^\sharp(x, b) \rightarrow g^\sharp(x, x), g^\sharp(b, x) \rightarrow f^\sharp(a, a), g^\sharp(b, x) \rightarrow a^\sharp\}$. The innermost dependency graph of R_4 has one cycle, which contains a right ground node [Fig. 2]. The innermost termination of R_4 is decidable by Theorem 5.3. Actually we know R_4 is not innermost terminating from the procedure in the proof of Theorem 3.9 by detecting the looping sequence $f(a, a) \xrightarrow{in R_4} f(b, b) \xrightarrow{in R_4} g(b, b) \xrightarrow{in R_4} h(f(a, a), b)$.

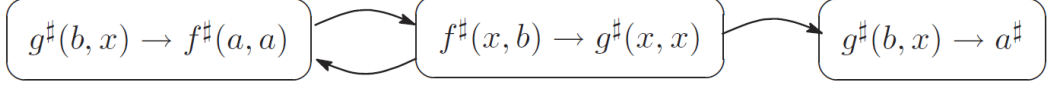


Figure 2: The innermost DP-Graph of R_4

5.2 Context-Sensitive

We extend the class for which μ -termination is decidable by using the dependency graph. The class extended in this subsection is the class that satisfy the condition of Theorem 4.17.

Lemma 5.6 *Let R be a TRS and μ be an \mathcal{F} -map. If μ -termination of R is equivalent to the non-existence of a context-sensitive dependency chain that contains infinite use of right-ground rule in $DP_{\mathcal{F}}(R, \mu)$. Then μ -termination of R is decidable.*

Proof. We apply the procedure which used on proof of Lemma 4.22 starting with terms u_1, u_2, \dots, u_n , where u_i^\sharp 's are all ground right-hand sides of rules in $DP_{\mathcal{F}}(R, \mu)$. Suppose R is non- μ -termination, we have an context-sensitive dependency chain with infinite use of right-ground rule in $DP_{\mathcal{F}}(R, \mu)$. Similarly to the μ -semi-constructor case, we have a looping sequence $u_i \xrightarrow{\mu^+}_R C_\mu[u_i]$, which can be detected by the procedure. \square

Definition 5.7 (Context-Sensitive DP-Graph [1]) *The Context-Sensitive dependency graph of a TRS R and an \mathcal{F} -map μ is directed graph whose nodes are CS-dependency pairs:*

1. *There is an arc from $s \rightarrow t \in DP_{\mathcal{F}}(R, \mu)$ to $u \rightarrow v \in DP(R, \mu)$ if there exist substitutions σ and τ such that $t\sigma \xrightarrow{\mu^*}_R u\tau$.*
2. *There is an arc from $s \rightarrow t \in DP_{\mathcal{V}}(R, \mu)$ to each dependency pair $u \rightarrow v \in DP(R, \mu)$.*

Similarly to the innermost case, a computable approximated dependency graph is proposed for context-sensitive DP-graph[1].

Theorem 5.8 *Let R be a TRS, μ be an \mathcal{F} -map and G be an approximated DP-graph of R . μ -termination of R is decidable if one of followings hold for every cycle in G .*

1. *The cycle contains at least one node that is right-ground.*
2. *All nodes of the cycle are elements in $DP_{\mathcal{V}}^1(R, \mu)$.*

Proof. From Lemma 5.6 and Theorem 4.16. \square

Example 5.9 Let $R_5 = \{h(x) \rightarrow g(x, x), g(a, x) \rightarrow f(b, x), f(x, x) \rightarrow h(a), a \rightarrow b\}$, $\mu_5(f) = \mu_5(g) = \mu_5(h) = \{1\}$ and $\mu_5(a) = \mu_5(b) = \emptyset$ [7]. Then $\text{DP}(R_5, \mu_5) = \{h^\sharp(x) \rightarrow g^\sharp(x, x), g^\sharp(a, x) \rightarrow f^\sharp(b, x), f^\sharp(x, x) \rightarrow h^\sharp(a), f^\sharp(x, x) \rightarrow a^\sharp\}$. The context-sensitive dependency graph of R_5 and μ_5 has one cycle, which contains a right ground node [Fig.3]. The μ_5 -termination of R_5 is decidable by Theorem 5.8. Actually we know R_5 is μ_5 -terminating from the procedure in the proof of Theorem 4.15 since all μ_5 -reduction sequences from $h(a)$ terminates.

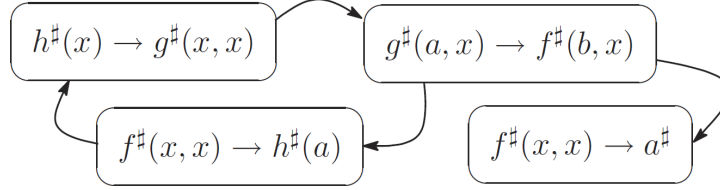


Figure 3: The innermost DP-Graph of R_5 and μ_5

Example 5.10 Let $\mu_6(f) = \{2\}$, $\mu_6(g) = \mu_6(h) = \{1\}$ and $\mu_6(a) = \mu_6(b) = \emptyset$. Consider μ_6 -termination of R_5 . The context-sensitive dependency graph of R_5 and μ_6 is same as one of R_5 and μ_5 [Fig.3]. The μ_6 -termination of R_5 is decidable by Theorem 5.8. By the decision procedure, we can detect the μ_6 -looping sequence $h(a) \xrightarrow{\mu_6}_{R_5} g(a, a) \xrightarrow{\mu_6}_{R_5} f(b, a) \xrightarrow{\mu_6}_{R_5} f(b, b) \xrightarrow{\mu_6}_{R_5} h(a)$. Thus R_5 is non- μ_5 -terminating.

The class of TRS that satisfy the condition of Theorem 5.8 is a superclass of the class of TRS that satisfy the condition of Theorem 4.17. However, the semi-constuctor class and none of these classes are included from each other.

Acknowledgement

We would like to thank the anonymous referees for their helpful comments and remarks. This work is partly supported by MEXT.KAKENHI #18500011 and #16300005.

References

- [1] B. Alarcón, R. Gutiérrez, and S. Lucas. Context-Sensitive Dependency Pairs. In *the 26th Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 4337 of *Lecture Notes in Computer Science*, pages 298–309, 2006.
- [2] T. Arts and J. Giesl. Termination of term rewriting using dependency pairs. *Theoretical Computer Science*, 236:133–178, 2000.

- [3] F. Baader and T. Nipkow. *Term rewriting and all that*. Cambridge University Press, 1998.
- [4] N. Dershowitz. Termination of Linear Rewriting Systems. In *the 8th International Colloquium on Automata, Languages and Programming*, volume 115 of *Lecture Notes in Computer Science*, pages 448–458, 1981.
- [5] G. Godoy and A. Tiwari. Termination of Rewrite Systems with Shallow Right-Linear, Collapsing, and Right-Ground Rules. In *the 20th International Conference on Automated Deduction*, volume 3632 of *Lecture Notes in Computer Science*, pages 164–176, 2005.
- [6] G. Huet and D. Lankford. On the Uniform Halting Problem for Term Rewriting Systems. Technical report, INRIA, 1978.
- [7] S. Lucas. Proving Termination of Context-Sensitive Rewriting by Transformation. *Information and Computation*, 204:1782–1846, 2006.
- [8] T. Nagaya and Y. Toyama. Decidability for left-linear growing term rewriting systems. *Information and Computation*, 178:499–514, 2002.
- [9] Y. Wang and M. Sakai. Decidability of Termination for Semi-Constructor TRSs, Left-Linear Shallow TRSs and Related Systems. In *the 17th International Conference on Rewriting Techniques and Applications*, volume 4098 of *Lecture Notes in Computer Science*, pages 343–356, 2006.
- [10] H. Zantema. Termination of context-sensitive rewriting. In *the 8th Conference on Rewriting Techniques and Applications*, volume 1232 of *Lecture Notes in Computer Science*, pages 172–186, 1997.