数理解析研究所講究録
第 1554 巻 2007 年 166-170

Decidability of Innermost Termination for Semi-Constructor Term Rewriting Systems

Keita Uchiyama †, Masahiko Sakai ‡, Naoki Nishida ‡, Toshiki Sakabe ‡and Keiichirou Kusakari ‡

†School of Engineering, Nagoya University
‡Graduate School of Information Science, Nagoya University

†uchiyama@sakabe.i.is.nagoya-u.ac.jp
‡{sakai,nishida,sakabe,kusakari}@is.nagoya-u.ac.jp

Abstract

Yi and Sakai [7] showed that the termination problem is decidable for a 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 [7] to show that innermost termination is decidable 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 [3, 4, 5, 6, 7]. The class of semi-constructor TRSs showed in [7] is one of them, where a TRS is in the class if every subterm of right hand sides of rules, whose root symbol is defined, is ground.

Innermost reduction, the strategy which rewrites innermost redexes, is used as call-by-value computation semantics. The termination property with respect to innermost reduction is called innermost termination. Since innermost termination is also undecidable in general, methods for proving innermost termination have been studied [1].

In this paper, we prove that innermost termination for semi-constructor TRSs is a decidable property. The proof is done by following the proof [7] for decidability of termination for semi-constructor TRSs.

2 Preliminaries

We assume the reader is familiar with the standard definitions of term rewriting systems [2] 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: *arity*: $\mathcal{F} \to \mathrm{N}(=\{0,1,2,\ldots\})$. Function symbols of arity 0 are called *constants*. 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 denoted by $\mathcal{T}(\mathcal{F},\emptyset)$ ($\mathcal{T}(\mathcal{F})$ for short). The set of variables occurring in a term t is denoted by $\mathrm{Var}(t)$. We write $s = t$ when two terms s and t are identical. The *root symbol* of a term t is denoted by *root*(t).

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. We say t is a subterm of s if $s = C[t]$ for some context C. We denote the *subterm ordering* by $\trianglelefteq$, that is, $t \trianglelefteq s$ if t is a subterm of s, and $t \triangleleft s$ if $t \trianglelefteq s$ and $t \neq s$.

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

A *rewrite rule* $l \to r$ is a directed equation which satisfies $l \notin \mathcal{V}$ and $\mathrm{Var}(r) \subseteq \mathrm{Var}(l)$. A *term rewriting system* TRS is a finite set of rewrite rules. A *redex* is a term $l\theta$ for a rule $l \to r$ and a substitution θ. A term containing no redex is called a *normal form*. A redex is *innermost*, if its all proper subterms are in normal forms. A substitution θ is *normal* if $x\theta$ is in normal forms for every x. The *reduction relation* $\to_R \subseteq \mathcal{T}(\mathcal{F}, \mathcal{V}) \times \mathcal{T}(\mathcal{F}, \mathcal{V})$ associated with a TRS R is defined as follows: $s \to_R t$ if there exist a rewrite rule $l \to r \in R$, a substitution θ, and a context C such that $s = C[l\theta]$ and $t = C[r\theta]$. We say that s is reduced to t by contracting redex $l\theta$. Especially, if s is reduced to t by contracting an innermost redex, then $s \to_R t$ is said to be *innermost reduction* denoted by $s \xrightarrow[in]{}_R t$.

Let $\to$ be a binary relation on terms, the transitive closure of $\to$ is denoted by $\to^+$. The transitive and reflexive closure of $\to$ is denoted by $\to^*$. If $s \to^* t$, then we say that there is a *reduction sequence* starting from s to t or t is *reachable* from s with respect to $\to$. We write $s \to^k t$ if t is reachable from s by reductions with k steps. For a TRS R, a term $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ *terminates* with respect to $\to$ if there is no infinite reduction sequence starting from t with respect to $\to$. We say that R terminates with respect to $\to$ if every term terminates with respect to $\to$. We say that R innermost terminates if it terminates with respect to $\xrightarrow[in]{}_R$.

Proposition 1 For a TRS R, if there is a reduction $s \xrightarrow[in]{}_R t$, then $C[s] \xrightarrow[in]{}_R C[t]$ for any context C.

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

Let R be a TRS over a signature $\mathcal{F}$. $\mathcal{F}^\sharp$ 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*. Given a term $t = f(t_1, \ldots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{V})$ with f defined, we write $t^\sharp$ for the term $f^\sharp(t_1, \ldots, t_n)$. If $l \to r \in R$ and u is a subterm of r with a defined root symbol, then the rewrite rule $l^\sharp \to u^\sharp$ is called a *dependency pair* of R. The set of all dependency pairs of R is denoted by $\mathrm{DP}(R)$.

3 Decidability of Innermost Termination for Semi-Constructor TRSs

Many infinite reduction sequences contain loops, in which an instance of a term re-occurs as a subterm. 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[7]. In this section, we prove decidability of innermost termination in similar way.

Definition 2 (Loop) A *reduction sequence loops* if it contains $t' \to^+ C[t'\theta]$ for some context C, substitution θ. Similarly, a reduction sequence *cycles* if containing $t' \to^+ C[t']$, and *head-cycles* if containing $t' \to^+ t'$.

The following proposition holds from proposition 1.

Proposition 3 If there exists a cycling sequence that loops, then there exists infinite innermost sequence.

Definition 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 $s \trianglelefteq t$ and $root(s) \in D_R$ is ground. A TRS R is a semi-constructor system if r is a semi-constructor term for every rule $l \to r \in R$.

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

Proposition 5 The following statements hold:

1. Right-ground TRSs are semi-constructor systems.
2. For a TRS R, R is semi-constructor TRS if and only if all rules in DP(R) are right-ground.

For a given TRS, let $\mathcal{T}_\infty^{\to}$ denote the set of all *minimal non-terminating terms* for $\to$, here "minimal" is used in the sense that all its proper subterms terminate. For a given TRS, let $\mathcal{NF}^{\to}$ denote the set of all normal forms for $\to$.

Definition 6 (Innermost DP-chain) For a TRS R, a sequence of the elements of DP(R) $s_1^\sharp \to t_1^\sharp, s_2^\sharp \to t_2^\sharp, \ldots$ is an *innermost dependency chain* if there exist substitutions $\tau_1, \tau_2, \ldots$ such that $s_i^\sharp \tau_i \in \mathcal{NF}^{\xrightarrow[in]{}_R}$ and $t_i^\sharp \tau_i \xrightarrow[in]{*}_R s_{i+1}^\sharp \tau_{i+1}$ holds for every i.

Theorem 7 ([1]) For a TRS R, R does not innermost terminate if and only if there exists an infinite innermost dependency chain.

Definition 8 ($\mathcal{C}$-min) For a TRS R, let $\mathcal{C} \subseteq$ DP(R). An infinite reduction sequence in $R \cup \mathcal{C}$ in the form $t_1^\sharp \xrightarrow[in]{}_{R\cup\mathcal{C}} t_2^\sharp \xrightarrow[in]{}_{R\cup\mathcal{C}} t_3^\sharp \xrightarrow[in]{}_{R\cup\mathcal{C}} \cdots$ 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 $\mathcal{C}_{min}^{in}(t^\sharp)$ to denote the set of all $\mathcal{C}$-min innermost reduction sequence starting from $t^\sharp$.

Proposition 9 ([1]) Given a TRS R, we have the following statements:

1. If there exists an infinite innermost dependency chain, then $\mathcal{C}_{min}^{in}(t^\sharp) \neq \emptyset$ for some $\mathcal{C} \subseteq$ DP(R) and $t \in \mathcal{T}_\infty^{\xrightarrow[in]{}_R}$.
2. For any sequence in $\mathcal{C}_{min}^{in}(t^\sharp)$, 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 $\mathcal{C}_{min}^{in}(t^\sharp)$, there is at least one rule in $\mathcal{C}$ which is applied infinitely often.

Lemma 10 For any sequence in $\mathcal{C}_{min}^{in}(t^\sharp)$, subsequence $s^\sharp \xrightarrow[in]{*}_{R\cup\mathcal{C}} s'^\sharp$ implies $s \xrightarrow[in]{*}_R C[s']$ for some context C.

Proof We use induction on number n of reduction steps $s^\sharp \xrightarrow[in]{n}_{R\cup\mathcal{C}} s'^\sharp$. In the case that $n = 0$, it holds with $C = \Box$. Let $n \geq 1$. Then we have $s^\sharp \xrightarrow[in]{n-1}_{R\cup\mathcal{C}} s''^\sharp \xrightarrow[in]{}_{R\cup\mathcal{C}} s'^\sharp$ for some $s''^\sharp$. By induction hypothesis, $s \xrightarrow[in]{*}_R C[s'']$.
(i): 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 1. Hence $s \xrightarrow[in]{*}_R C[s']$.
(ii): Consider the case that $s''^\sharp \xrightarrow[in]{}_{\mathcal{C}} s'^\sharp$. $s'' \xrightarrow[in]{}_R C'[s']$, by definition of dependency pair. $C[s''] \xrightarrow[in]{}_R C[C'[s']]$, by Proposition 1. Hence $s \xrightarrow[in]{*}_R C[C'[s']]$. □

Lemma 11 For a TRS R, if $sq \in \mathcal{C}_{min}^{in}(t^\sharp)$ cycles, then sq head-cycles.

Proof Let $sq \in \mathcal{C}_{min}^{in}(t^\sharp)$ cycles, then there is a subsequence $t_k^\sharp \xrightarrow[in]{+}_{R\cup\mathcal{C}} C[t_k^\sharp]$ in sq. From Proposition 9-(2) and the fact that dependency pair symbols appears only in dependency pairs, we have $C[t_k^\sharp] = t_k^\sharp$, which implies that sq head-cycles. □

Lemma 12 For a TRS R, if $sq \in \mathcal{C}_{min}^{in}(t^\sharp)$ cycles, then there is a term $t_k^\sharp$ in sq such that sequence starting from t_k cycles with respect to $\xrightarrow[in]{}_R$.

Proof From Lemma 11 and Lemma 10. □

Lemma 13 For a semi-constructor TRS R, the following statements are equivalent:

1. R does not innermost terminate.
2. There exists $l^\sharp \to u^\sharp \in$ DP(R) such that sq head-cycles for some $sq \in \mathcal{C}_{min}^{in}(u^\sharp)$.

Proof $(2 \Rightarrow 1)$: It is obvious from Lemma 12, and Proposition 3. $(1 \Rightarrow 2)$: By Theorem 7 there exists an infinite innermost dependency chain. By Proposition 9-(1), there exists a sequence $sq \in \mathcal{C}_{min}^{in}(t^\sharp)$. By Proposition 9-(2,3), there is some rule $l^\sharp \to u^\sharp \in \mathcal{C}$ which is applied at root reduction in sq infinitely often. By Proposition 5-(2), $u^\sharp$ is ground. Thus $u^\sharp$ cycle in the form $u^\sharp \xrightarrow[in]{}^*_{R \cup DP(R)} \cdot \to_{\{l^\sharp \to u^\sharp\}} u^\sharp$ □

Theorem 14 Innermost termination of semi-constructor TRSs is decidable.

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

Suppose R does not innermost terminate. By Lemma 13, 12 and the groundness of u_i's, we have a cycling reduction sequence $u_i \xrightarrow[in]{}^+_R C[u_i]$ for some i and C. Hence we detect innermost non-termination of R. If R innermost terminates, then the execution of the reduction sequence generation stops finally since it is finitely branching. Thus we detect innermost termination of R after finitely many steps. □

4 Some Extension and Example

In this section, we relax the condition that guarantees decidability of innermost termination for semi-constructor TRSs.

Lemma 15 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 above procedure starting with terms $u_1, u_2, \ldots, u_n$, where $u_i^\sharp$'s are all ground right-hand sides of dependency pairs. Suppose R is innermost non-termination, we have an innermost dependency chain with infinite use of a right-ground dependency pair. Similarly to the semi-constructor case, we have a loop $u_i \to_R^+ C[u_i]$, which can be detected by the procedure. □

Definition 16 (Innermost DP-Graph [1]) The *innermost dependency graph* of a TRS R is directed graph whose nodes are the dependency pairs and there is an arc from $s^\sharp \to t^\sharp$ to $u^\sharp \to v^\sharp$ if there exists a normal substitution σ, 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 innermost dependency graph as subgraph. One of computable such graphs are proposed [1].

Theorem 17 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 15. □

Example 18 Let $R = \{f(s(x)) \to g(x), g(s(x)) \to f(s(0))\}$. Then $\mathrm{DP}(R) = \{f^\sharp(s(x)) \to g^\sharp(x),\ g^\sharp(s(x)) \to f^\sharp(s(0))\}$. The dependency graph of R has one cycle, which contains a right ground node. The termination of R is decidable by Theorem 17. Actually we know R is innermost terminating from the procedure in the proof of Theorem 14 since all innermost reduction sequences from $f(s(0))$ terminates.

References

[1] T. Arts and J. Giesl. Proving Innermost Normalisation Automatically. In *the 8th International Conference on Rewriting Techniques and Applications*, volume 1232 of *Lecture Notes in Computer Science*, pages 157–171, 1997.

[2] F. Baader and T. Nipkow. *Term rewriting and all that.* Cambridge University Press, 1998.

[3] N. Dershowitz. Termination of Linear Rewriting Systems. In *Eighth International Colloquium on Automata, Languages and*

Programming, volume 115 of *Lecture Notes in Computer Science*, pages 448–458, 1981.

[4] 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, CADE*, volume 3632 of *Lecture Notes in Computer Science*, pages 164–176, 2005.

[5] G. Huet and D. Lankford. On the Uniform Halting Problem for Term Rewriting Systems. Technical report, INRIA, 1978.

[6] T.Nagaya and Y.Toyama. Decidability for left-linear growing term rewriting systems. In *the 10th International Conference on Rewriting Techniques and Applications*, volume 1631 of *Lecture Notes in Computer Science*, pages 256–270, 1999.

[7] 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.