Context-Sensitive Dependency Pairs[☆]

Beatriz Alarcón[a], Raúl Gutiérrez[a], Salvador Lucas[a]

[a] ELP group, DSIC, Universidad Politécnica de Valencia, Spain

---

Abstract

Termination is one of the most interesting problems when dealing with context-sensitive rewrite systems. Although a good number of techniques for proving termination of context-sensitive rewriting (*CSR*) have been proposed so far, the adaptation to *CSR* of the *dependency pair approach*, one of the most powerful techniques for proving termination of rewriting, took some time and was possible only after introducing some new notions like *collapsing dependency pairs*, which are specific for *CSR*. In this paper, we develop the notion of *context-sensitive dependency pair* (CSDP) and show how to use CSDPs in proofs of termination of *CSR*. The implementation and practical use of the developed techniques yield a novel and powerful framework which improves the current state-of-the-art of methods for automatically proving termination of *CSR*.

Keywords: Dependency pairs, term rewriting, program analysis, termination.

---

1. Introduction

Most computational systems whose operational principle is based on reducing expressions can be described and analyzed by using notions and techniques from the abstract framework of Term Rewriting Systems (TRSs [BN98, TeR03]). Such computational systems (e.g., functional, algebraic, and equational programming languages as well as theorem provers based on rewriting techniques) often incorporate a predefined reduction strategy that is used to break down the nondeterminism that is inherent to reduction relations. Eventually, this can raise problems, as each kind of strategy only behaves properly for particular classes of programs (i.e., it is normalizing, optimal, etc.). For this reason, the designers of programming languages have developed mechanisms to give the user more flexible control of the program execution. For instance, *syntactic annotations* (which are associated to arguments of symbols) have been used in programming languages such as Clean [NSEP92], Haskell [HPW92], Lisp [McC60], Maude [CDEL$^+$07], OBJ2 [FGJM85], OBJ3 [GWMF$^+$00], CafeOBJ [FN97], etc. to improve the termination and efficiency of computations. Lazy languages (e.g., Haskell, Clean) interpret them as *strictness annotations*

[☆] This work has been partially supported by the EU (FEDER) and the Spanish MEC/MICINN, under grants TIN 2007-68093-C02 and HA 2006-0007. Beatriz Alarcón was partially supported by the Spanish MEC/MICINN under FPU grant AP2005-3399. Raúl Gutiérrez was partially supported by the Spanish MEC/MICINN grant TIN 2004-7943-C04-02.

Email addresses: `balarcon@dsic.upv.es` (Beatriz Alarcón), `rgutierrez@dsic.upv.es` (Raúl Gutiérrez), `slucas@dsic.upv.es` (Salvador Lucas)

URL: `http://www.dsic.upv.es/~balarcon` (Beatriz Alarcón), `http://www.dsic.upv.es/~rgutierrez` (Raúl Gutiérrez), `http://www.dsic.upv.es/~slucas` (Salvador Lucas)

$$
\begin{array}{rcl}
\mathtt{evenNs} & \to & \mathtt{cons}(\mathtt{0},\mathtt{incr}(\mathtt{oddNs}))\\
\mathtt{oddNs} & \to & \mathtt{incr}(\mathtt{evenNs})\\
\mathtt{incr}(\mathtt{cons}(x,xs)) & \to & \mathtt{cons}(\mathtt{s}(x),\mathtt{incr}(xs))\\
\mathtt{take}(\mathtt{0},xs) & \to & \mathtt{nil}\\
\mathtt{take}(\mathtt{s}(n),\mathtt{cons}(x,xs)) & \to & \mathtt{consF}(x,\mathtt{take}(n,xs))\\
\mathtt{zip}(\mathtt{nil},xs) & \to & \mathtt{nil}\\
\mathtt{zip}(xs,\mathtt{nil}) & \to & \mathtt{nil}\\
\mathtt{zip}(\mathtt{cons}(x,xs),\mathtt{cons}(y,ys)) & \to & \mathtt{cons}(\mathtt{frac}(x,y),\mathtt{zip}(xs,ys))\\
\mathtt{tail}(\mathtt{cons}(x,xs)) & \to & xs\\
\mathtt{rep2}(\mathtt{nil}) & \to & \mathtt{nil}\\
\mathtt{rep2}(\mathtt{cons}(x,xs)) & \to & \mathtt{cons}(x,\mathtt{cons}(x,\mathtt{rep2}(xs)))\\
\mathtt{add}(\mathtt{0},n) & \to & n\\
\mathtt{add}(\mathtt{s}(n),m) & \to & \mathtt{s}(\mathtt{add}(n,m))\\
\mathtt{prod}(\mathtt{0},n) & \to & \mathtt{0}\\
\mathtt{prod}(\mathtt{s}(n),m) & \to & \mathtt{add}(m,\mathtt{prod}(n,m))\\
\mathtt{prodFrac}(\mathtt{frac}(x,y),\mathtt{frac}(z,t)) & \to & \mathtt{frac}(\mathtt{prod}(x,z),\mathtt{prod}(y,t))\\
\mathtt{prodOfFracs}(\mathtt{nil}) & \to & \mathtt{frac}(\mathtt{s}(\mathtt{0}),\mathtt{s}(\mathtt{0}))\\
\mathtt{prodOfFracs}(\mathtt{consF}(p,ps)) & \to & \mathtt{prodFrac}(p,\mathtt{prodOfFracs}(ps))\\
\mathtt{halfPi}(n) & \to & \mathtt{prodOfFracs}(\mathtt{take}(n,\mathtt{zip}(\mathtt{rep2}(\mathtt{tail}(\mathtt{evenNs})),\mathtt{tail}(\mathtt{rep2}(\mathtt{oddNs})))))
\end{array}
$$

Figure 1: Computing Wallis' approximation to $\frac{\pi}{2}$

in order to become 'more eager' and efficient. Eager languages (e.g., Lisp, Maude, OBJ2, OBJ3, CafeOBJ) use them as *replacement restrictions* to become 'more lazy', thus (hopefully) avoiding nontermination. Termination is one of the most interesting practical problems in computation and software engineering. A program or computational system is said to be *terminating* if it does not lead to any infinite computation for any possible call or input data. Ensuring termination is often a prerequisite for essential program properties like correctness. Messages reporting (a never-ending) "processing", "waiting for an answer", or even "abnormal termination" (which are often raised during the execution of software applications) usually correspond to nonterminating computations arising from bugs in the program.

Context-sensitive rewriting (*CSR* [Luc98, Luc02]) is a restriction of rewriting that has proved useful in investigating some of the aforementioned programming languages, see, e.g., [BM06, DLMM$^+$04, DLMM$^+$08, GM04, Luc01, LM09]. In *CSR*, the restriction of the rewriting computations is first imposed on the *arguments* of function symbols f in the signature $\mathcal{F}$. A *signature* is a set of function symbols $f_1,\ldots,f_n,\ldots$ together with an *arity* function $ar : \mathcal{F} \to \mathbb{N}$ that establishes the number of 'arguments' associated to each symbol. A *replacement map* is a mapping $\mu : \mathcal{F} \to \wp(\mathbb{N})$ that satisfies $\mu(f) \subseteq \{1,\ldots,ar(f)\}$, for each symbol f in the signature $\mathcal{F}$ [Luc98]. It specifies the argument positions where rewriting is allowed. In *CSR*, we only rewrite μ-replacing subterms: every term t (as a whole) is μ-replacing by definition; and t_i (as well as all its μ-replacing subterms) is a μ-replacing subterm of $f(t_1,\ldots,t_k)$ if $i \in \mu(f)$.

Example 1. *The TRS $\mathcal{R}$ in Figure 1 can be used to compute approximations to $\frac{\pi}{2}$ by using Wallis' product: $\frac{\pi}{2} = lim_{n\to\infty}\frac{2}{1}\frac{2}{3}\frac{4}{3}\frac{4}{5}\cdots\frac{2n}{2n-1}\frac{2n}{2n+1}$. In $\mathcal{R}$, function symbols* `0` *and* `s` *are used to represent natural numbers in Peano's notation; we also have the usual arithmetic operations* `add`*ition and* `prod`*uct. Symbols* `cons` *and* `nil` *are the standard list constructors which are then used to build (possibly infinite) lists of natural numbers like* `evenNs` *(the infinite list of even numbers) and* `oddNs` *(the infinite list of odd numbers). Function* `incr` *increases all the elements of a list in one unit through the application of* `s`*. The function* `zip` *merges a pair of lists into a list* *of fractions, and* `tail` *returns the elements of a list after removing the first one. The function* `take` *can be used to obtain the components of a finite approximation to* $\frac{\pi}{2}$ *which we multiply with* `prodOfFracs`*. Note the explicit use of* `consF` *for building* finite *lists of fractions of natural numbers by means of* `take`*, thus ensuring that the product of their elements computed by* `prodOfFracs` *is well-defined. A call* $\mathtt{halfPi}(\mathtt{s}^n(\mathtt{0}))$ *for some positive number* $n > 0$ *will return the desired approximation. Since* $\mathcal{R}$ *is* nonterminating *(due to the first two rules), we should be careful when choosing the rewrite steps that will be issued to obtain an approximation.*

With CSR *we can achieve a* terminating behaviour *for this system. Consider the replacement map* μ *given by:*

$$\mu(\mathtt{cons}) = \{1\} \textit{ and } \mu(f) = \{1, \ldots, ar(f)\} \textit{ for all } f \in \mathcal{F} - \{\mathtt{cons}\}$$

where $\mu(\mathtt{cons}) = \{1\}$ *disallows reductions on the* list *part of the list constructor* `cons`*, thus making a kind of* lazy evaluation *of lists possible. Furthermore, the replacement restrictions imposed by the replacement map* μ *are* not *an obstacle to obtaining the desired approximations: the repeated application of context-sensitive rewriting steps to an expression* $\mathtt{halfPi}(\mathtt{s}^n(\mathtt{0}))$ *will obtain (disregarding the particular choice of such steps) an expression* $\mathtt{frac}(\mathtt{s}^p(\mathtt{0}), \mathtt{s}^q(\mathtt{0}))$ *representing the approximation* $\frac{p}{q}$ *to* $\frac{\pi}{2}$*, which is obtained by taking the first n terms in Wallis' formula (this follows from [Luc98, Theorem 11]).*

1.1. Termination of context-sensitive rewriting

Several methods have been developed for proving termination of *CSR* under a replacement map μ for a given TRS $\mathcal{R}$ (i.e., for proving the *μ-termination* of $\mathcal{R}$). Termination of *CSR* is an interesting problem with several applications in the fields of term rewriting and in the analysis of programming languages [AEGL10, DLMM$^+$04, DLMM$^+$08, EH09, Fer05, GM04, Luc02, Luc06, SG08]. The development of methods and techniques for automatically proving termination is, therefore, one of the most interesting and challenging problems when dealing with *CSR*. Furthermore, with *CSR*, we can *achieve* a terminating behavior with nonterminating TRSs by pruning (all) infinite rewrite sequences as shown in Example 1. Examples of tools that are able to automatically prove termination of *CSR* are AProVE [GST06], Jambox [End10], MU-TERM [AGIL07, Luc04a], and VMTL [SG09].

In the nineties, a number of transformations that permit termination of *CSR* to be treated as a standard termination problem were developed (see [GM04, Luc06] for recent surveys). Polynomial orderings and the context-sensitive version of the recursive path ordering were also investigated [BLR02, GL02, Luc04b, Luc05]. In [AGL06], we adapted the *dependency pair method* [AG00, GAO02, HM04], which is a very powerful technique for proving termination of rewriting, to *CSR*. In this paper, we develop and improve the original notions in [AGL06] to incorporate recent improvements introduced by the Dependency Pair Framework [GTS04, GTSF06], and we obtain a powerful and modern framework that improves the current state-of-the-art of methods that can be used to automatically prove termination of *CSR*. Our tool MU-TERM implements the methods and techniques described in this paper.

1.2. Dependency pairs for context-sensitive rewriting

A TRS $\mathcal{R}$ is terminating if there is no infinite rewrite sequence starting from any term. With regard to proofs of termination of rewriting, the dependency pair technique focuses on the following idea: the rules that are really able to produce such infinite sequences are those rules $l \to r$

$$
\begin{array}{rcll}
\texttt{ADD}(\texttt{s}(n), m) & \to & \texttt{ADD}(n, m) & (1)\\
\texttt{EVENNS} & \to & \texttt{INCR}(\texttt{oddNs}) & (2)\\
\texttt{EVENNS} & \to & \texttt{ODDNS} & (3)\\
\texttt{HALFPI}(n) & \to & \texttt{EVENNS} & (4)\\
\texttt{HALFPI}(n) & \to & \texttt{ODDNS} & (5)\\
\texttt{HALFPI}(n) & \to & \texttt{PRODOFFRACS}(\texttt{take}(n, \texttt{zip}(\texttt{rep2}(\texttt{tail}(\texttt{evenNs})), & \\
 & & \qquad\qquad\qquad\qquad\qquad \texttt{tail}(\texttt{rep2}(\texttt{oddNs}))))) & (6)\\
\texttt{HALFPI}(n) & \to & \texttt{REP2}(\texttt{oddNs}) & (7)\\
\texttt{HALFPI}(n) & \to & \texttt{REP2}(\texttt{tail}(\texttt{evenNs})) & (8)\\
\texttt{HALFPI}(n) & \to & \texttt{TAIL}(\texttt{evenNs}) & (9)\\
\texttt{HALFPI}(n) & \to & \texttt{TAIL}(\texttt{rep2}(\texttt{oddNs})) & (10)\\
\texttt{HALFPI}(n) & \to & \texttt{TAKE}(n, \texttt{zip}(\texttt{rep2}(\texttt{tail}(\texttt{evenNs})), \texttt{tail}(\texttt{rep2}(\texttt{oddNs})))) & (11)\\
\texttt{HALFPI}(n) & \to & \texttt{ZIP}(\texttt{rep2}(\texttt{tail}(\texttt{evenNs})), \texttt{tail}(\texttt{rep2}(\texttt{oddNs}))) & (12)\\
\texttt{INCR}(\texttt{cons}(x, xs)) & \to & \texttt{INCR}(xs) & (13)\\
\texttt{ODDNS} & \to & \texttt{EVENNS} & (14)\\
\texttt{ODDNS} & \to & \texttt{INCR}(\texttt{evenNs}) & (15)\\
\texttt{PROD}(\texttt{s}(n), m) & \to & \texttt{ADD}(m, \texttt{prod}(n, m)) & (16)\\
\texttt{PROD}(\texttt{s}(n), m) & \to & \texttt{PROD}(n, m) & (17)\\
\texttt{PRODFRAC}(\texttt{frac}(x, y), \texttt{frac}(z, t)) & \to & \texttt{PROD}(x, z) & (18)\\
\texttt{PRODFRAC}(\texttt{frac}(x, y), \texttt{frac}(z, t)) & \to & \texttt{PROD}(y, t) & (19)\\
\texttt{PRODOFFRACS}(\texttt{consF}(p, ps)) & \to & \texttt{PRODFRAC}(p, \texttt{prodOfFracs}(ps)) & (20)\\
\texttt{PRODOFFRACS}(\texttt{consF}(p, ps)) & \to & \texttt{PRODOFFRACS}(ps) & (21)\\
\texttt{REP2}(\texttt{cons}(x, xs)) & \to & \texttt{REP2}(xs) & (22)\\
\texttt{TAKE}(\texttt{s}(n), \texttt{cons}(x, xs)) & \to & \texttt{TAKE}(n, xs) & (23)\\
\texttt{ZIP}(\texttt{cons}(x, xs), \texttt{cons}(y, ys)) & \to & \texttt{ZIP}(xs, ys) & (24)
\end{array}
$$

Figure 2: Dependency Pairs for the TRS in Example 1

such that r contains some *defined* symbol[1] g. Intuitively, we can think of these rules as representing some possible (direct or indirect) recursive calls. Such recursion paths associated to each rule $l \to r$ are represented as new rules $u \to v$, where $u = f^\sharp(l_1, \ldots, l_k)$ if $l = f(l_1, \ldots, l_k)$, and where $v = g^\sharp(s_1, \ldots, s_m)$ if $s = g(s_1, \ldots, s_m)$ is a subterm of r and g is a defined symbol. The notation $f^\sharp$ for a given symbol f means that f is *marked*. In practice, we often capitalize f and use F instead of $f^\sharp$ in our examples. For this reason, the dependency pair technique starts by considering a new TRS $\mathsf{DP}(\mathcal{R})$ that contains all these new rules for each $l \to r \in \mathcal{R}$. For instance, according to [AG00], the set $\mathsf{DP}(\mathcal{R})$ of dependency pairs for $\mathcal{R}$ in Example 1 consists of the rules in Figure 2. The rules in $\mathcal{R}$ and the rules in $\mathsf{DP}(\mathcal{R})$ determine the so-called *dependency chains* whose finiteness or infiniteness characterize termination or nontermination of $\mathcal{R}$ [AG00]. A chain of dependency pairs is a sequence $u_i \to v_i$ of dependency pairs together with a substitution σ such that $\sigma(v_i)$ rewrites to $\sigma(u_{i+1})$ for all $i \geq 1$. The dependency pairs can be presented as a *dependency graph*, where the infinite chains are represented by the *cycles* in the graph. For

---

[1] A symbol $g \in \mathcal{F}$ is defined in $\mathcal{R}$ if there is a rule in $\mathcal{R}$ whose left-hand side is of the form $g(l_1, \ldots, l_k)$.

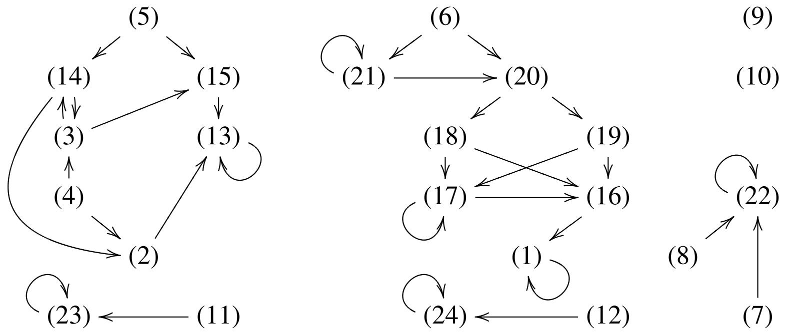

Figure 3: Dependency Graph for the TRS in Example 1

instance, the dependency graph that corresponds to the TRS $\mathcal{R}$ in Example 1 is depicted in Figure 3. The cycle consisting of nodes (3) and (14) witnesses the *nontermination* of $\mathcal{R}$.

In general, these intuitions are valid for *CSR*: the subterms s of the right-hand sides r of the rules $l \to r$ which are considered to build the *context-sensitive dependency pairs* $l^\sharp \to s^\sharp$ must be μ-*replacing* terms now.

Example 2. *Consider* $\mathcal{R}$ *and* μ *as in Example 1. Only the dependency pairs* (1), (4) – (12), (14) – (21), *and* (23) *in Figure 2 are also context-sensitive dependency pairs.*

The following example shows the need for dependency pairs of a *new kind*.

Example 3. *Consider the following TRS* $\mathcal{R}$*:*

$$\mathtt{a} \;\to\; \mathtt{c(f(a))} \qquad\qquad \mathtt{f(c(}x\mathtt{))} \;\to\; x$$

together with $\mu(\mathtt{c}) = \varnothing$ *and* $\mu(\mathtt{f}) = \{1\}$. *No* μ*-replacing subterm* s *in the right-hand sides of the rules is rooted by a defined symbol. Thus, there is no 'regular' dependency pair (in particular* $\mathtt{A} \to \mathtt{A}$ *is* dismissed *due to* $\mu(\mathtt{c}) = \varnothing$*). If no other dependency pair is considered, we could wrongly conclude that* $\mathcal{R}$ *is* μ*-terminating, which is not true:*

$$\mathtt{f(\underline{a})} \hookrightarrow_\mu \underline{\mathtt{f(c(f(a)))}} \hookrightarrow_\mu \mathtt{f(\underline{a})} \hookrightarrow_\mu \cdots$$

Indeed, we must add the following collapsing *dependency pair:*

$$\mathtt{F(c(}x\mathtt{))} \to x.$$

Since the right-hand side is a variable, this would not be allowed in Arts and Giesl's approach [AG00].

Collapsing pairs are essential in our approach. They express that infinite context-sensitive rewrite sequences can involve not only the kind of recursion that is represented by the *usual* dependency pairs but also a new kind of recursion that is *hidden* inside the nonreplacing (or *frozen*) parts of the terms involved in the infinite sequence. The *activation* of such *delayed* recursions is due to the presence of *migrating* variables within a rule $l \to r$ which is used in the sequence. Migrating variables are those that are not replacing in the left-hand side l but that *become* replacing in the right-hand side r.

Example 4. *(Continuing Example 2). The following* collapsing *pairs are* context-sensitive dependency pairs *for the CS-TRS in Example 1:*

$$\mathtt{TAIL}(\mathtt{cons}(x, xs)) \quad \to \quad xs \tag{25}$$

$$\mathtt{TAKE}(\mathtt{s}(n), \mathtt{cons}(x, xs)) \quad \to \quad xs \tag{26}$$

Note that variable xs is μ-replacing in the right-hand sides of the rules $\mathtt{tail}(\mathtt{cons}(x, xs)) \to xs$ and $\mathtt{take}(\mathtt{s}(n), \mathtt{cons}(x, xs)) \to \mathtt{consF}(x, \mathtt{take}(n, xs))$ but it is non-μ-replacing *in the corresponding left-hand sides.*

1.3. Plan of the paper

We have argued that termination of *CSR* is an interesting and challenging topic of research with a good number of practical applications. The results, techniques, and tools that derive from our work can be of interest to a sufficiently wide audience. The material in this paper will be more familiar, however, to those specialists who are interested in termination (in general) and in *how* to prove termination of *CSR* in particular. Throughout the paper, however, we made a serious effort to provide sufficient intuition and informal descriptions for our main definitions and results.

After Section 2 (Preliminaries), the paper is structured in three main parts:

1. Section 3 provides appropriate notions of *minimal* non-μ-terminating terms and introduces the main properties of such terms. We introduce the notion of *hidden term* and investigate the structure of infinite context-sensitive rewrite sequences starting from minimal non-μ-terminating terms. This analysis is essential in order to provide an appropriate definition of context-sensitive dependency pair and the related notions of chains, graphs, etc.
2. We define the notions of *context-sensitive dependency pair* and *context-sensitive chain of pairs* and show how to use them to *characterize* termination of *CSR*. Sections 4 and 5 introduce the general framework to compute and use context-sensitive dependency pairs to prove termination of *CSR*. The introduction of dependency pairs of a new kind (the *collapsing* dependency pairs, as in Example 3) leads to a notion of context-sensitive dependency *chain*, which is quite different from the standard one. In Section 6, we prove that our *context-sensitive dependency pair approach* fully characterizes termination of *CSR*.
3. We describe a suitable *framework* for dealing with proofs of termination of *CSR* by using these results. Section 7 adapts the *dependency pair framework* [GTS04, GTSF06] to *CSR* by defining appropriate notions of *CS problem* and *CS processor* that rely on the results obtained in the second part of the paper. Section 8 introduces the notion of *context-sensitive (dependency) graph* and the associated CS processor. Section 9 describes CS processors for removing or transforming collapsing pairs. Section 10 investigates the use of term orderings in processors. Section 11 adapts Hirokawa and Middeldorp's *subterm criterion* [HM04]. Section 12 adapts *narrowing transformation* of pairs in [GTSF06].

Experiments are reported in Section 13. Sections 14 and 15 discuss related work. Section 16 concludes.

2. Preliminaries

This section collects a number of definitions and notations about term rewriting. More details and missing notions can be found in [BN98, Ohl02, TeR03].

Let A be a set and $\mathsf{R} \subseteq \mathsf{A} \times \mathsf{A}$ be a binary relation on A. We denote the transitive closure of R by R^+ and its reflexive and transitive closure by R^*. We say that R is *terminating* (*strongly normalizing*) if there is no infinite sequence $a_1 \mathrel{\mathsf{R}} a_2 \mathrel{\mathsf{R}} a_3 \cdots$. A reflexive and transitive relation R is a quasi-ordering.

Given relations R and R' over the same set A, we define its *composition* $\mathsf{R} \circ \mathsf{R}'$ as follows: for all $a, b \in \mathsf{A}$, $a \,(\mathsf{R} \circ \mathsf{R}')\, b$ if there is $c \in \mathsf{A}$ such that $a \mathrel{\mathsf{R}} c$ and $c \mathrel{\mathsf{R}'} b$.

2.1. Signatures, Terms, and Positions

Throughout the paper, $\mathcal{X}$ denotes a countable set of variables and $\mathcal{F}$ denotes a signature, i.e., a set of function symbols $\{\mathtt{f}, \mathtt{g}, \ldots\}$, each having a fixed arity given by a mapping $ar : \mathcal{F} \to \mathbb{N}$. The set of terms built from $\mathcal{F}$ and $\mathcal{X}$ is $\mathcal{T}(\mathcal{F}, \mathcal{X})$. $\mathcal{V}ar(t)$ is the set of variables occurring in a term t. A term t is *ground* if it contains no variable (i.e., $\mathcal{V}ar(t) = \varnothing$). A term is said to be *linear* if it has no multiple occurrences of a single variable.

Terms are viewed as labelled trees in the usual way. Positions $p, q, \ldots$ are represented by chains of positive natural numbers used to address subterms of t. We denote the empty chain by Λ. Given positions p, q, we denote their concatenation as $p.q$. Positions are ordered by the standard prefix ordering: $p \leq q$ if $\exists q'$ such that $q = p.q'$. If p is a position, and Q is a set of positions, then $p.Q = \{p.q \mid q \in Q\}$. The set of positions of a term t is $\mathcal{P}os(t)$. Positions of nonvariable symbols in t are denoted as $\mathcal{P}os_{\mathcal{F}}(t)$, and $\mathcal{P}os_{\mathcal{X}}(t)$ are the positions of variables. The subterm at position p of t is denoted as $t|_p$, and $t[s]_p$ is the term t with the subterm at position p replaced by s.

We write $s \trianglerighteq t$, read *t is a subterm of s*, if $t = s|_p$ for some $p \in \mathcal{P}os(s)$ and $s \triangleright t$ if $s \trianglerighteq t$ and $s \neq t$. We write $s \ntrianglerighteq t$ and $s \ntriangleright t$ for the negation of the corresponding properties. The symbol labeling the root of t is denoted as $root(t)$. A *context* is a term $C \in \mathcal{T}(\mathcal{F} \cup \{\Box\}, \mathcal{X})$ with a 'hole' $\Box$ (a fresh constant symbol). We write $C[\;]_p$ to denote that there is a (usually single) hole $\Box$ at position p of C. Generally, we write $C[\;]$ to denote an arbitrary context and make the position of the hole explicit only if necessary. $C[\;] = \Box$ is called the *empty* context.

2.2. Substitutions, renamings, and unifiers

A substitution is a mapping $\sigma : \mathcal{X} \to \mathcal{T}(\mathcal{F}, \mathcal{X})$. Denote as ε the 'identity' substitution: $\varepsilon(x) = x$ for all $x \in \mathcal{X}$. The set $\mathcal{D}om(\sigma) = \{x \in \mathcal{X} \mid \sigma(x) \neq x\}$ is called the *domain* of σ.

Remark 1. *We do* not *impose that the domain of the substitutions be finite. This is usual practice in the dependency pair approach, where a single substitution is used to instantiate an infinite number of variables coming from renamed versions of the dependency pairs (see below).*

A *renaming* is an injective substitution ρ such that $\rho(x) \in \mathcal{X}$ for all $x \in \mathcal{X}$. A substitution σ such that $\sigma(s) = \sigma(t)$ for two terms $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ is called a *unifier* of s and t; we also say that s and t unify (with substitution σ). If two terms s and t unify, then there is a unique *most general unifier* σ (up to renaming of variables) such that for every other unifier τ, there is a substitution θ such that $\theta \circ \sigma = \tau$.

A relation $\mathsf{R} \subseteq \mathcal{T}(\mathcal{F}, \mathcal{X}) \times \mathcal{T}(\mathcal{F}, \mathcal{X})$ on terms is *stable* if, for all terms $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ and substitutions σ, we have $\sigma(s) \mathrel{\mathsf{R}} \sigma(t)$ whenever $s \mathrel{\mathsf{R}} t$.

2.3. Rewrite Systems and Term Rewriting

A rewrite rule is an ordered pair (l, r), written $l \to r$, with $l, r \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, $l \notin \mathcal{X}$ and $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(l)$. The left-hand side (*lhs*) of the rule is l, and the right-hand side (*rhs*) is r. A rewrite rule $l \to r$ is said to be *collapsing* if $r \in \mathcal{X}$. A *Term Rewriting System* (TRS) is a pair $\mathcal{R} = (\mathcal{F}, R)$, where R is a set of rewrite rules. We often use $\varnothing$ to denote TRSs whose set of rules R is empty. Given TRSs $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{R}' = (\mathcal{F}', R')$, we let $\mathcal{R} \cup \mathcal{R}'$ be the TRS $(\mathcal{F} \cup \mathcal{F}', R \cup R')$. An instance $\sigma(l)$ of a *lhs* l of a rule is called a *redex*. Given $\mathcal{R} = (\mathcal{F}, R)$, we consider $\mathcal{F}$ as the disjoint union $\mathcal{F} = \mathcal{C} \uplus \mathcal{D}$ of symbols $c \in \mathcal{C}$ (called *constructors*) and symbols $f \in \mathcal{D}$ (called *defined functions*), where $\mathcal{D} = \{root(l) \mid l \to r \in R\}$ and $\mathcal{C} = \mathcal{F} - \mathcal{D}$.

Example 5. *Consider again the TRS in Example 1. The symbols* `evenNs`, `oddNs`, `incr`, `take`, `zip`, `tail`, `rep2`, `add`, `prod`, `prodFrac`, `prodOfFracs`, *and* `halfPi` *are defined. Symbols* `s`, `0`, `cons`, `consF`, `nil`, *and* `frac` *are constructors.*

We often write $l \to r \in \mathcal{R}$ instead of $l \to r \in R$ to express that the rule $l \to r$ is a rule of $\mathcal{R}$. A term $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ rewrites to t (at position p), written $s \overset{p}{\to}_{\mathcal{R}} t$ (or just $s \to_{\mathcal{R}} t$, or $s \to t$), if $s|_p = \sigma(l)$ and $t = s[\sigma(r)]_p$, for some rule $l \to r \in R$, $p \in \mathcal{P}os(s)$ and substitution σ. We write $s \overset{>p}{\to}_{\mathcal{R}} t$ if $s \overset{q}{\to}_{\mathcal{R}} t$ for some $q > p$. A TRS $\mathcal{R}$ is terminating if its one step rewrite relation $\to_{\mathcal{R}}$ is terminating.

2.4. Context-Sensitive Rewriting

A mapping $\mu : \mathcal{F} \to \wp(\mathbb{N})$ is a *replacement map* (or $\mathcal{F}$-map) if for all symbols $f \in \mathcal{F}$, $\mu(f) \subseteq \{1, \ldots, ar(f)\}$ [Luc98]. Let $M_{\mathcal{F}}$ be the set of all $\mathcal{F}$-maps (or $M_{\mathcal{R}}$ for the $\mathcal{F}$-maps of a TRS $(\mathcal{F}, R)$). Let $\mu_\top$ be the replacement map given by $\mu_\top(f) = \{1, \ldots, ar(f)\}$ for all $f \in \mathcal{F}$ (i.e., no replacement restrictions are specified).

A binary relation R on terms is μ-monotonic if, for all $f \in \mathcal{F}$, $i \in \mu(f)$, and $s, t, t_1, \ldots, t_k \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, $f(t_1, \ldots, t_{i-1}, s, t_{i+1}, \ldots, t_k) \mathrel{\mathsf{R}} f(t_1, \ldots, t_{i-1}, t, t_{i+1}, \ldots, t_k)$ whenever $s \mathrel{\mathsf{R}} t$. If R is $\mu_\top$-monotonic, we just say that R is *monotonic*.

The set of *μ-replacing positions* $\mathcal{P}os^\mu(t)$ of $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ is: $\mathcal{P}os^\mu(t) = \{\Lambda\}$ if $t \in \mathcal{X}$, and $\mathcal{P}os^\mu(t) = \{\Lambda\} \cup \bigcup_{i \in \mu(root(t))} i.\mathcal{P}os^\mu(t|_i)$ if $t \notin \mathcal{X}$. Note that $\mathcal{P}os^\mu(t)$ (as $\mathcal{P}os(t)$) is *prefix closed*. When no replacement map is made explicit, the μ-replacing positions are often called *active*; and the non-μ-replacing ones are often called *frozen*. The following results about *CSR* are often used without any explicit mention.

Proposition 1. [Luc98] *Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ and $p = q.q' \in \mathcal{P}os(t)$. Then $p \in \mathcal{P}os^\mu(t)$ iff $q \in \mathcal{P}os^\mu(t) \wedge q' \in \mathcal{P}os^\mu(t|_q)$*

The chain of symbols lying on positions above/on $p \in \mathcal{P}os(t)$ is $prefix_t(\Lambda) = root(t)$, $prefix_t(i.p) = root(t).prefix_{t|_i}(p)$. The strict prefix *sprefix* is $sprefix_t(\Lambda) = \Lambda$, $sprefix_t(p.i) = prefix_t(p)$, i.e., the last symbol in $prefix_t(p.i)$ is *removed*. Although $sprefix_t(p)$ is a sequence, when the ordering of symbols in $sprefix_t(p)$ does not matter, we often use the standard set-theoretic notation (e.g., inclusion as in $sprefix_t(p) \subseteq \mathcal{F}$) with the obvious meaning.

Proposition 2. [Luc98] *If $p \in \mathcal{P}os(t) \cap \mathcal{P}os(s)$ and $sprefix_t(p) = sprefix_s(p)$, then $p \in \mathcal{P}os^\mu(t) \Leftrightarrow p \in \mathcal{P}os^\mu(s)$.*

The μ-replacing subterm relation $\unrhd_\mu$ is given by $s \unrhd_\mu t$ if there is $p \in \mathcal{P}os^\mu(s)$ such that $t = s|_p$. We write $s \rhd_\mu t$ if $s \unrhd_\mu t$ and $s \neq t$. We write $s \rhd_{\not\mu} t$ to denote that t is a non-μ-replacing (hence strict) subterm of s: $s \rhd_{\not\mu} t$ if there is $p \in \mathcal{P}os(s) - \mathcal{P}os^\mu(s)$ such that $t = s|_p$. The set of *μ-replacing* variables of a term t, i.e., variables occurring at some μ-replacing position in t, is $\mathcal{V}ar^\mu(t) = \{x \in \mathcal{V}ar(t) \mid t \unrhd_\mu x\}$. The set of non-$\mu$-replacing variables of t, i.e., variables occurring at some non-μ-replacing position in t, is $\mathcal{V}ar^{\not\mu}(t) = \{x \in \mathcal{V}ar(t) \mid t \rhd_{\not\mu} x\}$. Note that $\mathcal{V}ar^\mu(t)$ and $\mathcal{V}ar^{\not\mu}(t)$ do not need to be disjoint (when t is not linear).

A pair $(\mathcal{R}, \mu)$ where $\mathcal{R}$ is a TRS and $\mu \in M_\mathcal{R}$ is often called a CS-TRS. In *context-sensitive rewriting*, we (only) contract *μ-replacing* redexes: s μ-rewrites to t, written $s \overset{p}{\hookrightarrow}_{\mathcal{R},\mu} t$ (or $s \hookrightarrow_{\mathcal{R},\mu} t$, $s \hookrightarrow_\mu t$ and even $s \hookrightarrow t$), if $s \overset{p}{\to}_\mathcal{R} t$ and $p \in \mathcal{P}os^\mu(s)$.

Example 6. *Consider $\mathcal{R}$ and μ as in Example 1. Then, we have:*

$$\underline{\texttt{evenNs}} \hookrightarrow_\mu \texttt{cons(0, incr(}\underline{\texttt{oddNs}}\texttt{))} \not\hookrightarrow_\mu \texttt{cons(0, incr(incr(evenNs)))}$$

Since the second argument of `cons` *is not μ-replacing, we have* $2 \notin \mathcal{P}os^\mu(\texttt{cons(0, incr(oddNs))})$. *Thus, redex* `oddNs` *cannot be μ-rewritten.*

A term t is μ-terminating (or $(\mathcal{R}, \mu)$-terminating, if we want an explicit reference to the involved TRS $\mathcal{R}$) if there is no infinite μ-rewrite sequence $t = t_1 \hookrightarrow_{\mathcal{R},\mu} t_2 \hookrightarrow_{\mathcal{R},\mu} \cdots \hookrightarrow_{\mathcal{R},\mu} t_n \hookrightarrow_{\mathcal{R},\mu} \cdots$ starting from t. A TRS $\mathcal{R}$ is μ-terminating if $\hookrightarrow_{\mathcal{R},\mu}$ is terminating.

A term s *μ-narrows* to a term t (written $s \rightsquigarrow_{\mathcal{R},\mu,\theta} t$), if there is a nonvariable μ-replacing position $p \in \mathcal{P}os^\mu_\mathcal{F}(s)$ and a rule $l \to r$ in $\mathcal{R}$ (sharing no variable with s) such that $s|_p$ and l unify with the most general unifier θ and $t = \theta(s[r]_p)$. The following definition is used in Section 10.2 below.

Definition 1. [GL02] *Let $\mathcal{F}$ be a signature and $\mu \in M_\mathcal{F}$. The μ-replacing projection TRS $\mathcal{E}mb^\mu(\mathcal{F})$ consists of the following rules:*

$$\{f(x_1, \ldots, x_k) \to x_i \mid f \in \mathcal{F}, i \in \mu(f)\}$$

3. Minimal non-μ-terminating terms and infinite μ-rewrite sequences

Given a TRS $\mathcal{R} = (\mathcal{C} \uplus \mathcal{D}, R)$, the *minimal* nonterminating terms associated to $\mathcal{R}$ are nonterminating terms t whose proper subterms u (i.e., $t \rhd u$) are terminating; $\mathcal{T}_\infty$ is the set of minimal nonterminating terms associated to $\mathcal{R}$ [HM04, HM07]. Minimal nonterminating terms have two important properties:

1. Every nonterminating term s contains a minimal nonterminating term $t \in \mathcal{T}_\infty$ (i.e., $s \unrhd t$), and
2. minimal nonterminating terms t are always rooted by a *defined* symbol $f \in \mathcal{D}$: $\forall t \in \mathcal{T}_\infty, root(t) \in \mathcal{D}$.

As discussed in [HM04], considering the structure of the infinite rewrite sequences starting from a minimal nonterminating term $t \in \mathcal{T}_\infty$ can be helpful to come to the notion of dependency pair [AG00]. Such sequences proceed as follows:

Proposition 3. [HM04, Lemma 1] *Let $\mathcal{R} = (\mathcal{C} \uplus \mathcal{D}, R)$ be a TRS. For all $t \in \mathcal{T}_\infty$, there exist $l \to r \in R$, a substitution σ and a term $u \in \mathcal{T}_\infty$ such that $root(u) \in \mathcal{D}$, $t \xrightarrow{>\Lambda}{}^* \sigma(l) \overset{\Lambda}{\to} \sigma(r) \unrhd u$, and there is a nonvariable subterm v of r, $r \unrhd v$, such that $u = \sigma(v)$.*

In the following, we show how to generalize these notions and results to *CSR*.

3.1. Minimal non-μ-terminating terms

Before starting our discussion about (minimal) non-μ-terminating terms, we provide an obvious auxiliary result about μ-terminating terms[2].

Lemma 1. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, and $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. If s is μ-terminating, then:*

1. *If $s \trianglerighteq_\mu t$, then t is μ-terminating.*
2. *If $s \hookrightarrow^*_{\mathcal{R},\mu} t$, then t is μ-terminating.*

Given a TRS $\mathcal{R} = (\mathcal{F}, R)$ and a replacement map $\mu \in M_{\mathcal{F}}$, maybe the simplest extension to *CSR* of the notion of minimal term for unrestricted rewriting (i.e., $\mathcal{T}_\infty$), is the following: let $\mathcal{T}_{\infty,\mu}$ be a set of minimal non-μ-terminating terms in the following sense: t belongs to $\mathcal{T}_{\infty,\mu}$ if t is non-μ-terminating and every strict subterm u (i.e., $t \rhd u$) is μ-terminating. It is obvious that $root(t) \in \mathcal{D}$ for all $t \in \mathcal{T}_{\infty,\mu}$. We also have the following:

Lemma 2. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, and $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. If s is not μ-terminating, then there is a subterm t of s ($s \trianglerighteq t$) such that $t \in \mathcal{T}_{\infty,\mu}$.*

Unfortunately, there can be non-μ-terminating terms having no *μ-replacing* subterm in $\mathcal{T}_{\infty,\mu}$.

Example 7. *Consider the CS-TRS $(\mathcal{R}, \mu)$ in Example 3 and $s = \mathtt{f(c(f(a)))}$. Note that s is not μ-terminating, but $s \notin \mathcal{T}_{\infty,\mu}$ because $\mathtt{f(c(f(a)))} \rhd \mathtt{f(a)}$ and $\mathtt{f(a)}$ is not μ-terminating. Note that $\mathtt{f(c(f(a)))} \rhd_{\not\mu} \mathtt{f(a)}$. The only μ-replacing strict subterm of s is $\mathtt{c(f(a))}$, which is μ-terminating, i.e., $\mathtt{c(f(a))} \notin \mathcal{T}_{\infty,\mu}$.*

Therefore, minimal non-μ-terminating terms are not the most natural ones because they could occur at non-μ-replacing positions, where no μ-rewriting step is possible. Thus, this simple notion would not lead to an appropriate generalization of Proposition 3 to *CSR*. There is a suitable generalization of Proposition 3 to *CSR* (see Proposition 5 below) based on the following notion.

Definition 2 (Minimal non-μ-terminating term). *Let $\mathcal{M}_{\infty,\mu}$ be a set of minimal non-μ-terminating terms in the following sense: t belongs to $\mathcal{M}_{\infty,\mu}$ if t is non-μ-terminating and every strict* μ-replacing *subterm t' of t (i.e., $t \rhd_\mu t'$) is μ-terminating.*

Note that $\mathcal{T}_{\infty,\mu} \subseteq \mathcal{M}_{\infty,\mu}$. In the following, we often say that terms in $\mathcal{T}_{\infty,\mu}$ are *strongly minimal* non-μ-terminating; we use them in Section 3.4 below. Now, we have the following:

Lemma 3. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, and $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. If s is not μ-terminating, then there is a μ-replacing subterm t of s such that $t \in \mathcal{M}_{\infty,\mu}$.*

Obviously, if $t \in \mathcal{M}_{\infty,\mu}$, then $root(t)$ is a defined symbol. Since μ-terminating terms are preserved under μ-rewriting (Lemma 1), it follows that $\mathcal{M}_{\infty,\mu}$ is preserved under *inner* μ-rewritings in the following sense.

Lemma 4. *Let $\mathcal{R}$ be a TRS, $\mu \in M_{\mathcal{R}}$, and $t \in \mathcal{M}_{\infty,\mu}$. If $t \overset{>\Lambda}{\hookrightarrow}{}^{*}$ u and u is non-μ-terminating, then $u \in \mathcal{M}_{\infty,\mu}$.*

Lemma 4 does *not* hold for $\mathcal{T}_{\infty,\mu}$: consider the CS-TRS $(\mathcal{R}, \mu)$ in Example 3. Note that $\mathtt{f(a)} \in \mathcal{T}_{\infty,\mu}$ and $\mathtt{f(\underline{a})} \overset{>\Lambda}{\hookrightarrow} \mathtt{f(c(f(a)))}$. Although $\mathtt{f(c(f(a)))}$ is not μ-terminating, $\mathtt{f(c(f(a)))} \notin \mathcal{T}_{\infty,\mu}$, as shown in Example 7.

---

[2] For the sake of readability, the missing proofs of the technical results in this section have been moved to the appendix.

3.2. Hidden terms in minimal μ-rewrite sequences

Given a CS-TRS $(\mathcal{R},\mu)$, the *hidden terms* are nonvariable terms occurring on some frozen position in the right-hand side of some rule of $\mathcal{R}$. As we show in the next section, they play an important role in infinite minimal μ-rewrite sequences associated to $\mathcal{R}$.

Definition 3 (Hidden symbols and terms). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. We say that $t \in \mathcal{T}(\mathcal{F},\mathcal{X}) - \mathcal{X}$ is a* hidden term *if there is a rule $l \to r \in R$ such that $r \rhd_{\not\mu} t$. Let $\mathcal{HT}(\mathcal{R},\mu)$ (or just $\mathcal{HT}$, if no confusion arises) be the set of all hidden terms in $(\mathcal{R},\mu)$. We say that $f \in \mathcal{F}$ is a* hidden symbol *if it occurs in a hidden term. Let $\mathcal{H}(\mathcal{R},\mu)$ (or just $\mathcal{H}$) be the set of all hidden symbols in $(\mathcal{R},\mu)$.*

In the following, we also use $\mathcal{DHT}(\mathcal{R},\mu) = \{t \in \mathcal{HT}(\mathcal{R},\mu) \mid root(t) \in \mathcal{D}\}$ for the set of hidden terms which are rooted by a *defined* symbol.

Example 8. *For $\mathcal{R}$ and μ as in Example 1, the maximal hidden terms are* $\mathtt{incr}(\mathtt{oddNs})$, $\mathtt{incr}(x)$, $\mathtt{zip}(xs, ys)$, *and* $\mathtt{cons}(x, \mathtt{rep2}(xs))$. *The hidden symbols are* `incr`, `oddNs`, `incr`, `zip`, `cons`, *and* `rep2`. *Finally,* $\mathcal{DHT}(\mathcal{R},\mu) = \{\mathtt{oddNs}, \mathtt{incr}(\mathtt{oddNs}), \mathtt{incr}(x), \mathtt{zip}(xs, ys), \mathtt{rep2}(xs)\}$.

The following lemma says that frozen subterms t in the contractum $\sigma(r)$ of a redex $\sigma(l)$ that do *not* contain t are (at least partly) 'introduced' by a hidden term in the right-hand side r of the involved rule $l \to r$.

Lemma 5. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $t \in \mathcal{T}(\mathcal{F},\mathcal{X})$ and σ be a substitution. If there is a rule $l \to r \in R$ such that $\sigma(l) \ntrianglerighteq t$ and $\sigma(r) \rhd_{\not\mu} t$, then there is no $x \in \mathcal{V}ar(r)$ such that $\sigma(x) \trianglerighteq t$. Furthermore, there is a term $t' \in \mathcal{HT}$ such that $r \rhd_{\not\mu} t'$ and $\sigma(t') = t$.*

The following lemma establishes that minimal non-μ-terminating and non-μ-replacing subterms that occur in a μ-rewrite sequence involving only minimal terms come directly from the first term in the sequence or are instances of a hidden term.

Lemma 6. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Let A be a μ-rewrite sequence $t_1 \hookrightarrow t_2 \hookrightarrow \cdots \hookrightarrow t_n$ with $t_i \in \mathcal{M}_{\infty,\mu}$ for all i, $1 \leq i \leq n$. If there is a term $t \in \mathcal{M}_{\infty,\mu}$ such that $t_1 \ntrianglerighteq t$ and $t_n \rhd_{\not\mu} t$, then $t = \sigma(s)$ for some $s \in \mathcal{DHT}$ and substitution σ.*

We use the previous results to investigate infinite sequences that combine μ-rewriting steps on minimal non-μ-terminating terms and the extraction of such subterms as μ-replacing subterms of (instances of) right-hand sides of the rules.

Proposition 4. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Consider a finite or infinite sequence of the form $t_1 \overset{\Lambda}{\hookrightarrow} s_1 \trianglerighteq_\mu t'_2 \overset{>\Lambda}{\hookrightarrow}{}^* t_2 \overset{\Lambda}{\hookrightarrow} s_2 \trianglerighteq_\mu t'_3 \overset{>\Lambda}{\hookrightarrow}{}^* t_3 \cdots$ with $t_i, t'_i \in \mathcal{M}_{\infty,\mu}$ for all $i \geq 1$. If there is a term $t \in \mathcal{M}_{\infty,\mu}$ such that $t_i \rhd_{\not\mu} t$ for some $i \geq 1$, then $t_1 \rhd_{\not\mu} t$ or $t = \sigma(s)$ for some $s \in \mathcal{DHT}$ and substitution σ.*

3.3. Infinite μ-rewrite sequences starting from minimal terms

The following proposition establishes that, given a minimal non-μ-terminating term $t \in \mathcal{M}_{\infty,\mu}$, there are only two ways for an infinite μ-rewrite sequence to proceed. The first one is by using 'visible' parts of the rules that correspond to μ-replacing nonvariable subterms in the right-hand sides that are rooted by a defined symbol. The second one is by showing up 'hidden' non-μ-terminating subterms that are activated by *migrating* variables in a rule $l \to r$, i.e., variables $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)$ that are *not* μ-replacing in the left-hand side l but become μ-replacing in the right-hand side r.

Proposition 5. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Then, for all $t \in \mathcal{M}_{\infty,\mu}$, there exist $l \to r \in \mathcal{R}$, a substitution σ, and a term $u \in \mathcal{M}_{\infty,\mu}$ such that $t \overset{>\Lambda}{\hookrightarrow}{}^{*} \sigma(l) \overset{\Lambda}{\hookrightarrow} \sigma(r) \unrhd_\mu u$ and either*

1. *there is a nonvariable μ-replacing subterm s of r, $r \unrhd_\mu s$, such that $u = \sigma(s)$, or*
2. *there is $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)$ such that $\sigma(x) \unrhd_\mu u$.*

Proof. Consider an infinite μ-rewrite sequence starting from t. By definition of $\mathcal{M}_{\infty,\mu}$, all proper μ-replacing subterms of t are μ-terminating. Therefore, t has an inner reduction to an instance $\sigma(l)$ of the left-hand side of a rule $l \to r$ of $\mathcal{R}$: $t \overset{>\Lambda}{\hookrightarrow}{}^{*} \sigma(l) \overset{\Lambda}{\hookrightarrow} \sigma(r)$ and $\sigma(r)$ is not μ-terminating. Thus, we can write $t = f(t_1, \ldots, t_k)$ and $\sigma(l) = f(l_1, \ldots, l_k)$ for some k-ary defined symbol f, and $t_i \hookrightarrow^* \sigma(l_i)$ for all i, $1 \leq i \leq k$. Since all t_i are μ-terminating for $i \in \mu(f)$, by Lemma 1, $\sigma(l_i)$ and all its μ-replacing subterms are also μ-terminating. In particular, $\sigma(y)$ is μ-terminating for all μ-replacing variables y in l: $y \in \mathcal{V}ar^\mu(l)$. Since $\sigma(r)$ is non-μ-terminating, by Lemma 3, it contains a μ-replacing subterm $u \in \mathcal{M}_{\infty,\mu}$: $\sigma(r) \unrhd_\mu u$, i.e., there is a position $p \in \mathcal{P}os^\mu(\sigma(r))$ such that $\sigma(r)|_p = u$. We consider two cases:

1. If $p \in \mathcal{P}os_{\mathcal{F}}(r)$ is a nonvariable position of r, then there is a μ-replacing nonvariable subterm s of r (i.e., $p \in \mathcal{P}os^\mu_{\mathcal{F}}(r)$ and $s = r|_p \notin \mathcal{X}$), such that $u = \sigma(s)$.
2. If $p \notin \mathcal{P}os_{\mathcal{F}}(r)$, then there is a μ-replacing variable position $q \in \mathcal{P}os^\mu(r) \cap \mathcal{P}os_{\mathcal{X}}(r)$ such that $q \leq p$. Let $x \in \mathcal{V}ar^\mu(r)$ be such that $r|_q = x$. Then, $\sigma(x) \unrhd_\mu u$, and $\sigma(x)$ is not μ-terminating: since $u \in \mathcal{M}_{\infty,\mu}$ is not μ-terminating, by Lemma 1, $\sigma(x)$ is not μ-terminating. Since $\sigma(y)$ is μ-terminating for all $y \in \mathcal{V}ar^\mu(l)$, we conclude that $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)$.

□

Proposition 5 entails the following result, which establishes some properties of infinite sequences starting from minimal non-μ-terminating terms.

Corollary 1. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. For all $t \in \mathcal{M}_{\infty,\mu}$, there is an infinite sequence*

$$t \overset{>\Lambda}{\hookrightarrow}{}^{*} \sigma_1(l_1) \overset{\Lambda}{\hookrightarrow} \sigma_1(r_1) \unrhd_\mu t_1 \overset{>\Lambda}{\hookrightarrow}{}^{*} \sigma_2(l_2) \overset{\Lambda}{\hookrightarrow} \sigma_2(r_2) \unrhd_\mu t_2 \overset{>\Lambda}{\hookrightarrow}{}^{*} \cdots$$

where, for all $i \geq 1$, $l_i \to r_i \in \mathcal{R}$ are rewrite rules, σ_i are substitutions, and terms $t_i \in \mathcal{M}_{\infty,\mu}$ are minimal non-μ-terminating terms such that either

1. *$t_i = \sigma_i(s_i)$ for some nonvariable subterm s_i such that $r_i \unrhd_\mu s_i$, or*
2. *$\sigma_i(x_i) \unrhd_\mu t_i$ for some $x_i \in \mathcal{V}ar^\mu(r_i) - \mathcal{V}ar^\mu(l_i)$.*

Remark 2. *The $(\hookrightarrow_\mu \cup \unrhd_\mu)$-sequence in Corollary 1 can be easily viewed as an infinite μ-rewrite sequence by just introducing appropriate contexts $C_i[\,]_{p_i}$ with μ-replacing holes: since $\sigma_i(r_i) \unrhd_\mu t_i$, there is $p_i \in \mathcal{P}os^\mu(\sigma_i(r_i))$ such that $\sigma_i(r_i) = \sigma_i(r_i)[t_i]_{p_i}$; just take $C_i[\;]_{p_i} = \sigma(r_i)[\square]_{p_i}$. Hence:*

$$t \hookrightarrow^* \sigma_1(l_1) \hookrightarrow C_1[t_1]_{p_1} \hookrightarrow^* C_1[\sigma_2(l_2)]_{p_1} \hookrightarrow C_1[C_2[t_2]_{p_2}]_{p_1} \hookrightarrow^* \cdots$$

Note that, e.g., $p_1.p_2 \in \mathcal{P}os^\mu(C_1[C_2[t_2]_{p_2}]_{p_1})$ (use Proposition 1).

3.4. Infinite μ-rewrite sequences starting from strongly minimal terms

In the following, we consider a function $\textsc{Ren}^\mu$ which *independently* renames all *occurrences* of μ-replacing variables within a term t by using new fresh variables that are not in $\mathcal{V}ar(t)$:

- $\textsc{Ren}^\mu(x) = y$ if x is a variable, where y is intended to be a fresh new variable that has not yet been used; and
- $\textsc{Ren}^\mu(f(t_1, \ldots, t_k)) = f([t_1]^f_1, \ldots, [t_k]^f_k)$ for evey k-ary symbol f, where given a term $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, $[s]^f_i = \textsc{Ren}^\mu(s)$ if $i \in \mu(f)$ and $[s]^f_i = s$ if $i \notin \mu(f)$.

Note that $\textsc{Ren}^\mu(t)$ keeps variables at non-μ-replacing positions untouched. Note also that $\textsc{Ren}^\mu$ is *not* a substitution: it replaces the $n(x)$ different μ-replacing occurrences of the *same* variable x by *different* variables $x_1, \ldots, x_{n(x)}$. Clearly, $t = \theta(\textsc{Ren}^\mu(t))$ for some substitution θ which just identifies the variables introduced by $\textsc{Ren}^\mu$ (i.e., $\theta(x_i) = x$ for all $1 \leq i \leq n(x)$). The use of $\textsc{Ren}^\mu$ together with μ-narrowability yields a necessary condition for reducibility of terms under some instantiations which is used in our development.

Proposition 6. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{X}) - \mathcal{X}$ be a nonvariable term and σ be a substitution. If $\sigma(t) \overset{>\Lambda}{\hookrightarrow}{}^* \sigma(l)$ for some (possibly renamed) rule $l \to r \in \mathcal{R}$, then $\textsc{Ren}^\mu(t)$ is μ-narrowable.*

Proof. We can write the sequence from $\sigma(t)$ to $\sigma(l)$ as follows: $\sigma(t) = t_1 \overset{>\Lambda}{\hookrightarrow} t_2 \overset{>\Lambda}{\hookrightarrow} \cdots \overset{>\Lambda}{\hookrightarrow} t_m = \sigma(l)$ for some $m \geq 1$. We proceed by induction on m.

1. If $m = 1$, then $\sigma(t) = \sigma(l)$. Since $t \notin \mathcal{X}$, t is μ-narrowable (at the root position) using the rule $l \to r$. Since $t = \theta(\textsc{Ren}^\mu(t))$ for some substitution θ, we have $\sigma(t) = \sigma(\theta(\textsc{Ren}^\mu(t))) = \sigma(l)$. Since we can assume that the new variables instantiated by θ are not in l, we have $\sigma(\theta(l)) = \sigma(l)$. Thus, $\textsc{Ren}^\mu(t)$ and l unify with mgu $\sigma \circ \theta$. Since $t \notin \mathcal{X}$, implies that $\textsc{Ren}^\mu(t) \notin \mathcal{X}$, $\textsc{Ren}^\mu(t)$ is μ-narrowable at the root position using the same rule $l \to r$.
2. If $m > 1$, then we have $t_1 \overset{>\Lambda}{\hookrightarrow} t_2 \overset{>\Lambda}{\hookrightarrow}{}^* \sigma(l)$. We consider two cases according to the position $p \in \mathcal{P}os^\mu(t_1)$ where the μ-rewrite step $t_1 \overset{>\Lambda}{\hookrightarrow} t_2$ is performed (note that $t_1 = \sigma(t)$ by assumption).
   (a) If $p \in \mathcal{P}os^\mu_{\mathcal{F}}(t)$, then there is a rule $l' \to r'$ and a substitution θ such that $\sigma(t)|_p = \sigma(t|_p) = \theta(l')$. Again, we have $\sigma(t|_p) = \sigma(l')$, i.e., t is μ-narrowable at position p using rule $l' \to r'$ and (reasoning as above), we conclude that $\textsc{Ren}^\mu(t)$ is μ-narrowable.
   (b) If $p \notin \mathcal{P}os^\mu_{\mathcal{F}}(t)$, then there is a μ-replacing variable position $q \in \mathcal{P}os^\mu_{\mathcal{X}}(t)$ of t such that $t|_q = x \in \mathcal{V}ar^\mu(t)$, $q \leq p$ and $\sigma(x) \hookrightarrow_\mu t_2|_q$. Therefore, $t_1 = \sigma(t[x]_q) = \sigma(t)[\sigma(x)]_q$ and $t_2 = \sigma(t)[t_2|_q]_q = \sigma'(t')$ for a term $t' = t[y]_q$ where y is a new fresh variable $y \notin \mathcal{V}ar(t)$ and a substitution σ' given by $\sigma'(y) = t_2|_q$ and $\sigma'(z) = \sigma(z)$ for all $z \in \mathcal{V}ar(t)$ (including x). Clearly,

   $$\sigma'(t') = \sigma'(t[y]_q) = \sigma'(t)[\sigma'(y)]_q = \sigma(t)[t_2|_q]_q = t_2.$$

   By the induction hypothesis, $\textsc{Ren}^\mu(t')$ is μ-narrowable. Since t and t' only differ in a single variable, we can assume that $\textsc{Ren}^\mu(t') = \textsc{Ren}^\mu(t)$. Thus, we conclude that $\textsc{Ren}^\mu(t)$ is μ-narrowable as well.

□

Corollary 2. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{X}) - \mathcal{X}$ be a nonvariable term and σ be a substitution such that $\sigma(t) \in \mathcal{M}_{\infty,\mu}$. Then,* $\textsc{Ren}^{\mu}(t)$ *is μ-narrowable.*

Proof. By Proposition 5, there is a rule $l \to r$ and a substitution σ such that $\sigma(t) \overset{>\Lambda}{\hookrightarrow}^{*}_{\mathcal{R},\mu} \sigma(l)$ (since we can assume that variables in l and variables in t are disjoint, we can apply the same substitution σ to t and l without any problem). By Proposition 6, the conclusion follows. □

In the following, we write $\textsc{Narr}^{\mu}_{\mathcal{R}}(t)$ (or just $\textsc{Narr}^{\mu}(t)$) to indicate that t is μ-narrowable w.r.t. the (intended) TRS $\mathcal{R}$. We also let

$$\mathcal{NHT}(\mathcal{R}, \mu) = \{t \in \mathcal{DHT}(\mathcal{R}, \mu) \mid \textsc{Narr}^{\mu}_{\mathcal{R}}(\textsc{Ren}^{\mu}(t))\}$$

be the set of *hidden terms* that are rooted by a *defined* symbol, and that after applying $\textsc{Ren}^{\mu}$ become *μ-narrowable*.

Example 9. *Since all terms $t \in \mathcal{DHT}(\mathcal{R}, \mu)$ for $\mathcal{R}$ and μ as in Example 8 are μ-narrowable (even without applying* $\textsc{Ren}^{\mu}$*), we have $\mathcal{NHT}(\mathcal{R}, \mu) = \mathcal{DHT}(\mathcal{R}, \mu)$.*

As a consequence of the previous results, we have the following main result, which we use later.

Theorem 1. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. For all $t \in \mathcal{T}_{\infty,\mu}$, there is an infinite sequence*

$$t = t_0 \overset{>\Lambda}{\hookrightarrow}^{*} \sigma_1(l_1) \overset{\Lambda}{\hookrightarrow} \sigma_1(r_1) \trianglerighteq_{\mu} t_1 \overset{>\Lambda}{\hookrightarrow}^{*} \sigma_2(l_2) \overset{\Lambda}{\hookrightarrow} \sigma_2(r_2) \trianglerighteq_{\mu} t_2 \overset{>\Lambda}{\hookrightarrow}^{*} \cdots$$

where, for all $i \geq 1$, $l_i \to r_i \in \mathcal{R}$ are rewrite rules, σ_i are substitutions, and terms $t_i \in \mathcal{M}_{\infty,\mu}$ are minimal non-μ-terminating terms such that either

1. *$t_i = \sigma_i(s_i)$ for some nonvariable term s_i such that $r_i \trianglerighteq_{\mu} s_i$, or*
2. *$\sigma_i(x_i) \trianglerighteq_{\mu} t_i$ for some $x_i \in \mathcal{V}ar^{\mu}(r_i) - \mathcal{V}ar^{\mu}(l_i)$ and $t_i = \theta_i(t'_i)$ for some $t'_i \in \mathcal{NHT}$ and substitution θ_i.*

Proof. Since $\mathcal{T}_{\infty,\mu} \subseteq \mathcal{M}_{\infty,\mu}$, by Corollary 1, we have a sequence

$$t = t_0 \overset{>\Lambda}{\hookrightarrow}^{*} \sigma_1(l_1) \overset{\Lambda}{\hookrightarrow} \sigma_1(r_1) \trianglerighteq_{\mu} t_1 \overset{>\Lambda}{\hookrightarrow}^{*} \sigma_2(l_2) \overset{\Lambda}{\hookrightarrow} \sigma_2(r_2) \trianglerighteq_{\mu} t_2 \overset{>\Lambda}{\hookrightarrow}^{*} \cdots$$

where, for all $i \geq 1$, $l_i \to r_i \in \mathcal{R}$, σ_i are substitutions, $t_i \in \mathcal{M}_{\infty,\mu}$, and either (1) $t_i = \sigma_i(s_i)$ for some nonvariable term s_i such that $r_i \trianglerighteq_{\mu} s_i$ or (2) $\sigma_i(x_i) \trianglerighteq_{\mu} t_i$ for some $x_i \in \mathcal{V}ar^{\mu}(r_i) - \mathcal{V}ar^{\mu}(l_i)$ (and hence $\sigma(l_i) \triangleright_{\not\mu} t_i$ and $\sigma(r_i) \trianglerighteq_{\mu} t_i$ as well). We only need to prove that terms t_i are instances of hidden terms in $\mathcal{NHT}$ whenever (2) holds. By Proposition 4, for all such terms t_i, we have that either (A) $\sigma_1(l_1) \triangleright_{\not\mu} t_i$ or (B) $t_i = \theta_i(t'_i)$ for some $t'_i \in \mathcal{DHT}$ and substitution θ_i. In case (B), we just consider Corollary 2, which ensures that $t'_i \in \mathcal{NHT}$. In case (A), since $t \overset{>\Lambda}{\hookrightarrow}^{*} \sigma_1(l_1)$ and $\sigma_1(l_1)$ is not μ-terminating, by Lemma 4, all terms u_j in the μ-rewrite sequence

$$t = u_1 \overset{>\Lambda}{\hookrightarrow} u_2 \overset{>\Lambda}{\hookrightarrow} \cdots \overset{>\Lambda}{\hookrightarrow} u_m = \sigma_1(l_1)$$

belong to $\mathcal{M}_{\infty,\mu}$: $u_j \in \mathcal{M}_{\infty,\mu}$ for all j, $1 \leq j \leq m$. Since $t \in \mathcal{T}_{\infty,\mu}$, all its strict subterms (disregarding their μ-replacing character) are μ-terminating. Since t_i is not μ-terminating, $t \not\trianglerighteq t_i$. By Lemma 6, $t_i = \theta_i(t'_i)$ for some $t'_i \in \mathcal{DHT}$ and substitution θ_i. By Corollary 2, $t'_i \in \mathcal{NHT}$. □

4. Context-Sensitive Dependency Pairs

By Lemma 2 every non-μ-terminating term s_0 contains a strongly minimal subterm $t \in \mathcal{T}_{\infty,\mu}$ which, by Theorem 1, starts an infinite μ-rewrite sequence. In such a sequence, a number of μ-rewriting steps *below the root* of t are performed. Then a rule $l \to r$ is applied at the *topmost* position of the obtained reduct. According to Proposition 5, the application of such a rule either

1. *introduces* a new minimal non-μ-terminating subterm u having a prefix s which is a non-variable μ-replacing subterm of r. By Corollary 2, $\textsc{Ren}^\mu(s)$ is μ-narrowable. Otherwise,
2. *takes* a minimal non-μ-terminating and non-μ-replacing subterm u and
   (a) brings it up to an *active* position by means of the binding $\sigma(x)$ for some *migrating variable* x in $l \to r$.
   (b) At this point, we know that u, which is rooted by a defined symbol due to $u \in \mathcal{M}_{\infty,\mu}$, is an instance of a hidden term $u' \in \mathcal{NHT}$.

Afterwards, further *inner* μ-rewritings on u lead to a matching with the left-hand-side l' of a new rule $l' \to r'$ and everything starts again. This process is abstracted in the definition of *context-sensitive dependency pairs* and in the definition of chain below.

Given a signature $\mathcal{F}$ and $f \in \mathcal{F}$, we let $f^\sharp$ be a new fresh symbol (often called *tuple* symbol or DP-symbol) associated to a symbol f [AG00]. Let $\mathcal{F}^\sharp$ be the set of tuple symbols associated to symbols in $\mathcal{F}$. As usual, for $t = f(t_1, \ldots, t_k) \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, we write $t^\sharp$ to denote the *marked* term $f^\sharp(t_1, \ldots, t_k)$. Conversely, given a marked term $t = f^\sharp(t_1, \ldots, t_k)$, where $t_1, \ldots, t_k \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, we write $t^\natural$ to denote the term $f(t_1, \ldots, t_k) \in \mathcal{T}(\mathcal{F}, \mathcal{X})$.

Definition 4 (Context-Sensitive Dependency Pairs). *Let $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $\mathsf{DP}(\mathcal{R}, \mu) = \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu) \cup \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ be the set of* context-sensitive *dependency pairs (CSDPs) where:*

$$\begin{array}{rcl} \mathsf{DP}_{\mathcal{F}}(\mathcal{R},\mu) & = & \{l^\sharp \to s^\sharp \mid l \to r \in R, r \trianglerighteq_\mu s, root(s) \in \mathcal{D}, l \ntriangleright_\mu s, \textsc{Narr}^\mu(\textsc{Ren}^\mu(s))\} \\ \mathsf{DP}_{\mathcal{X}}(\mathcal{R},\mu) & = & \{l^\sharp \to x \mid l \to r \in R, x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)\} \end{array}$$

We extend $\mu \in M_{\mathcal{F}}$ into $\mu^\sharp \in M_{\mathcal{F} \cup \mathcal{D}^\sharp}$ by $\mu^\sharp(f) = \mu(f)$ if $f \in \mathcal{F}$, and $\mu^\sharp(f^\sharp) = \mu(f)$ if $f \in \mathcal{D}$.

The CSDPs $u \to v \in \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ in Definition 4, consisting of collapsing rules only, are called the *collapsing* CSDPs.

Remark 3. *The notion of CSDP in Definition 4 differs from the standard definition of dependency pair [AG00, GTSF06] in two additional requirements:*

1. *As in [HM04], which follows Dershowitz's proposal in [Der04], we require that subterms s of the right-hand sides r of the rules $l \to r$ which are considered to build the dependency pairs $l^\sharp \to s^\sharp$ are not subterms of the left-hand side (i.e., $l \ntriangleright_\mu s$).*
2. *As in [LM08], we require μ-narrowability of* $\textsc{Ren}^\mu(s)$*:* $\textsc{Narr}^\mu(\textsc{Ren}^\mu(s))$.

But the crucial difference, which is specific for context-sensitive rewriting, is the introduction and use of collapsing *dependency pairs.*

A rule $l \to r$ of a TRS $\mathcal{R}$ is μ-conservative if $\mathcal{V}ar^\mu(r) \subseteq \mathcal{V}ar^\mu(l)$, i.e., there is no migrating variable; $\mathcal{R}$ is μ-conservative if all its rules are μ-conservative (see [Luc96, Luc06]). The following fact is obvious from Definition 4.

$$\begin{array}{rcll}
\mathtt{ADD}(\mathtt{s}(n), m) & \to & \mathtt{ADD}(n, m) & (1)\\
\mathtt{HALFPI}(n) & \to & \mathtt{EVENNS} & (4)\\
\mathtt{HALFPI}(n) & \to & \mathtt{ODDNS} & (5)\\
\mathtt{HALFPI}(n) & \to & \mathtt{PRODOFFRACS}(\mathtt{take}(n, \mathtt{zip}(\mathtt{rep2}(\mathtt{tail}(\mathtt{evenNs})), \mathtt{tail}(\mathtt{rep2}(\mathtt{oddNs}))))) & (6)\\
\mathtt{HALFPI}(n) & \to & \mathtt{REP2}(\mathtt{oddNs}) & (7)\\
\mathtt{HALFPI}(n) & \to & \mathtt{REP2}(\mathtt{tail}(\mathtt{evenNs})) & (8)\\
\mathtt{HALFPI}(n) & \to & \mathtt{TAIL}(\mathtt{evenNs}) & (9)\\
\mathtt{HALFPI}(n) & \to & \mathtt{TAIL}(\mathtt{rep2}(\mathtt{oddNs})) & (10)\\
\mathtt{HALFPI}(n) & \to & \mathtt{TAKE}(n, \mathtt{zip}(\mathtt{rep2}(\mathtt{tail}(\mathtt{evenNs})), \mathtt{tail}(\mathtt{rep2}(\mathtt{oddNs})))) & (11)\\
\mathtt{HALFPI}(n) & \to & \mathtt{ZIP}(\mathtt{rep2}(\mathtt{tail}(\mathtt{evenNs})), \mathtt{tail}(\mathtt{rep2}(\mathtt{oddNs}))) & (12)\\
\mathtt{ODDNS} & \to & \mathtt{EVENNS} & (14)\\
\mathtt{ODDNS} & \to & \mathtt{INCR}(\mathtt{evenNs}) & (15)\\
\mathtt{PROD}(\mathtt{s}(n), m) & \to & \mathtt{ADD}(m, \mathtt{prod}(n, m)) & (16)\\
\mathtt{PROD}(\mathtt{s}(n), m) & \to & \mathtt{PROD}(n, m) & (17)\\
\mathtt{PRODFRAC}(\mathtt{frac}(x, y), \mathtt{frac}(z, t)) & \to & \mathtt{PROD}(x, z) & (18)\\
\mathtt{PRODFRAC}(\mathtt{frac}(x, y), \mathtt{frac}(z, t)) & \to & \mathtt{PROD}(y, t) & (19)\\
\mathtt{PRODOFFRACS}(\mathtt{consF}(p, ps)) & \to & \mathtt{PRODFRAC}(p, \mathtt{prodOfFracs}(ps)) & (20)\\
\mathtt{PRODOFFRACS}(\mathtt{consF}(p, ps)) & \to & \mathtt{PRODOFFRACS}(ps) & (21)\\
\mathtt{TAKE}(\mathtt{s}(n), \mathtt{cons}(x, xs)) & \to & \mathtt{TAKE}(n, xs) & (23)\\
\mathtt{TAIL}(\mathtt{cons}(x, xs)) & \to & xs & (25)\\
\mathtt{TAKE}(\mathtt{s}(n), \mathtt{cons}(x, xs)) & \to & xs & (26)
\end{array}$$

Figure 4: Context-Sensitive Dependency Pairs for the CS-TRS in Example 1

Proposition 7. *If $\mathcal{R}$ is a μ-conservative TRS, then $\mathsf{DP}(\mathcal{R}, \mu) = \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$.*

Therefore, in order to deal with μ-conservative TRSs $\mathcal{R}$ we only need to consider the 'classical' dependency pairs in $\mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$.

Example 10. *Consider the following TRS $\mathcal{R}$:*

$$\begin{array}{rclcrcl}
\mathtt{g}(x) & \to & \mathtt{h}(x) & \qquad & \mathtt{h}(\mathtt{d}) & \to & \mathtt{g}(\mathtt{c})\\
\mathtt{c} & \to & \mathtt{d} & & & &
\end{array}$$

together with $\mu(\mathtt{g}) = \mu(\mathtt{h}) = \varnothing$ [Zan97, Example 1]. Note that $\mathcal{R}$ is μ-conservative. $\mathsf{DP}(\mathcal{R}, \mu)$ consists of the following (noncollapsing) CSDPs:

$$\begin{array}{rclcrcl}
\mathtt{G}(x) & \to & \mathtt{H}(x) & \qquad & \mathtt{H}(\mathtt{d}) & \to & \mathtt{G}(\mathtt{c})
\end{array}$$

with $\mu^\sharp(\mathtt{G}) = \mu^\sharp(\mathtt{H}) = \varnothing$.

If the TRS $\mathcal{R}$ contains non-μ-conservative rules, then we also need to consider dependency pairs with variables in the right-hand side.

Example 11. *As discussed in Examples 2 and 4, for the CS-TRS $(\mathcal{R}, \mu)$ in Example 1, we have the CSDPs in Figure 4.*

5. Chains of CSDPs

An essential property of the dependency pair method is that it provides a *characterization* of termination of TRSs $\mathcal{R}$ as the absence of infinite (minimal) *chains of dependency pairs* [AG00, GTSF06]. As we prove in Section 6, this is also true for *CSR* when CSDPs are considered. First, we have to introduce a suitable notion of chain that can be used with CSDPs. As in the DP-framework [GTS04, GTSF06], where the origin of *pairs* does not matter, we use another TRS $\mathcal{P}$ together with $\mathcal{R}$ to build the chains. Once this more abstract notion of chain is introduced, it can be particularized to be used with CSDPs, by just taking $\mathcal{P} = \mathsf{DP}(\mathcal{R}, \mu)$.

Definition 5 (Chain of pairs - Minimal chain). *Let* $\mathcal{R} = (\mathcal{F}, R)$ *and* $\mathcal{P} = (\mathcal{G}, P)$ *be TRSs and* $\mu \in M_{\mathcal{F} \cup \mathcal{G}}$. *A* $(\mathcal{P}, \mathcal{R}, \mu)$*-chain is a finite or infinite sequence of pairs* $u_i \to v_i \in \mathcal{P}$*, together with a substitution* $\sigma : \mathcal{X} \to \mathcal{T}(\mathcal{F} \cup \mathcal{G}, \mathcal{X})$ *satisfying that, for all* $i \geq 1$*:*

1. *if* $v_i \notin \mathcal{V}ar(u_i) - \mathcal{V}ar^{\mu}(u_i)$*, then* $\sigma(v_i) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$*, and*
2. *if* $v_i \in \mathcal{V}ar(u_i) - \mathcal{V}ar^{\mu}(u_i)$*, then* $\sigma(v_i) = C_i[s_i]_{p_i}$ *for some* s_i *and* $C_i[\,]_{p_i}$ *such that* $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$*,* $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$*, and* $s_i^{\sharp} \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$*.*

As usual, we assume that different occurrences of pairs do not share any variable (renaming substitutions are used if necessary). A $(\mathcal{P}, \mathcal{R}, \mu)$*-chain is called* minimal *if for all* $i \geq 1$,

1. *if* $v_i \notin \mathcal{V}ar(u_i) - \mathcal{V}ar^{\mu}(u_i)$*, then* $\sigma(v_i)$ *is* $(\mathcal{R}, \mu)$*-terminating, and*
2. *if* $v_i \in \mathcal{V}ar(u_i) - \mathcal{V}ar^{\mu}(u_i)$*, then* $s_i^{\sharp}$ *is* $(\mathcal{R}, \mu)$*-terminating and* $\exists \bar{s}_i \in \mathcal{NHT}(\mathcal{R}, \mu)$ *such that* $s_i = \sigma(\bar{s}_i)$*.*

Note that the condition $v_i \in \mathcal{V}ar(u_i) - \mathcal{V}ar^{\mu}(u_i)$ in Definition 5 implies that v_i is a variable. Furthermore, v_i is a *migrating variable* in the rule $u_i \to v_i$.

Remark 4 (Conventions about $\mathcal{P}$). *The following conventions about the component* $\mathcal{P} = (\mathcal{G}, P)$ *of our chains will be observed during our development:*

1. *According to the usual terminology [GTSF06], we often call* pairs *the rules* $u \to v \in \mathcal{P}$*.*
2. *We have to mark terms* $s_i \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ *before* connecting *them to the instance* $\sigma(u_{i+1})$ *of the left-hand side of the next pair. Since marked symbols* $f^{\sharp}$ *are* fresh *(w.r.t. the signature* $\mathcal{F}$ *of the TRS* $\mathcal{R}$*), we also assume that* $\mathcal{D}^{\sharp} \cap \mathcal{F} = \varnothing$ *and* $\mathcal{D}^{\sharp} \subseteq \mathcal{G}$*.*
3. *We assume that* $\mathcal{P}$ *contains a* finite *set of rules. This is essential in many proofs.*

In the following, the pairs in a CS-TRS $(\mathcal{P}, \mu)$, where $\mathcal{P} = (\mathcal{G}, P)$, are partitioned according to their role in Definition 5 as follows:

$$P_{\mathcal{X}} = \{u \to v \in P \mid v \in \mathcal{V}ar(u) - \mathcal{V}ar^{\mu}(u)\} \text{ and } P_{\mathcal{G}} = P - P_{\mathcal{X}}$$

Remark 5 (Collapsing pairs). *Note that all pairs in* $\mathcal{P}_{\mathcal{X}} = (\mathcal{G}, P_{\mathcal{X}})$ *are* collapsing*. The rules in* $\mathcal{P}_{\mathcal{G}} = (\mathcal{G}, P_{\mathcal{G}})$ can be collapsing *as well: a rewrite rule* $f(x) \to x \in \mathcal{P}$ *with* $\mu(f) = \{1\}$ *does* not *belong to* $\mathcal{P}_{\mathcal{X}}$ *but rather to* $\mathcal{P}_{\mathcal{G}}$ *because* x *is not a migrating variable.*

Despite this fact, we refer to $\mathcal{P}_{\mathcal{X}}$ *as the set of* collapsing *pairs in* $\mathcal{P}$ *because its intended role in Definition 5 is capturing the computational behavior of collapsing CSDPs in* $\mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$*.*

Remark 6 (Notation for chains). *In general, a* $(\mathcal{P}, \mathcal{R}, \mu)$*-chain can be written as follows:*

$$\sigma(u_1) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_1 \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_2 \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

where, for all $i \geq 1$ *and* $u_i \to v_i \in \mathcal{P}$*,*

1. *if* $u_i \to v_i \notin \mathcal{P}_{\mathcal{X}}$*, then* $t_i = \sigma(v_i)$*,*
2. *if* $u_i \to v_i \in \mathcal{P}_{\mathcal{X}}$*, then* $t_i = s_i^{\sharp}$ *for some term* s_i *such that* $\sigma(v_i) = C_i[s_i]_{p_i}$ *for some* $C_i[\,]_{p_i}$ *such that* $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$*, and* $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$*.*

This is denoted in a compact way by $\sigma(u_i) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_i$ *emphasizing that there is a* $\mathcal{P}$*-step followed by either an equality step (as in* (1)*) or by* μ*-replacing projection steps (restricted to symbols in* $\mathcal{F}$*) plus a marking operation (as in* (2)*) depending on the considered pair* $u_i \to v_i$*.*

5.1. Properties of some particular chains

In the following, we let $\mathcal{NHT}_{\mathcal{P}}(\mathcal{R},\mu) \subseteq \mathcal{NHT}(\mathcal{R},\mu)$ (or just $\mathcal{NHT}_{\mathcal{P}}$) be as follows:

$$\mathcal{NHT}_{\mathcal{P}}(\mathcal{R},\mu) = \{t \in \mathcal{NHT}(\mathcal{R},\mu) \mid \exists u \to v \in \mathcal{P}, \exists \theta, \theta', \theta(t^\sharp) \hookrightarrow^*_{\mathcal{R},\mu} \theta'(u)\}$$

This set contains the narrowable hidden terms that 'connect' with pairs in $\mathcal{P}$.

Remark 7. *Note that $\mathcal{NHT}_{\mathcal{P}}(\mathcal{R},\mu)$ is not computable, in general, due to the need for checking the reachability of $\theta'(u)$ from $\theta(t^\sharp)$ using* CSR. *Suitable (over)*approximations *are discussed below (see Remark 10).*

We let $\mathcal{P}^1_{\mathcal{X}}$ denote the subTRS of $\mathcal{P}_{\mathcal{X}}$ containing the rules whose migrating variables occur on non-μ-replacing immediate subterms in the left-hand side:

$$\mathcal{P}^1_{\mathcal{X}} = \{f(u_1,\ldots,u_k) \to x \in \mathcal{P}_{\mathcal{X}} \mid \exists i, 1 \leq i \leq k, i \notin \mu(f), x \in \mathcal{V}ar(u_i)\}$$

Proposition 8. *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$.*

1. *If $\mathcal{NHT}_{\mathcal{P}} = \varnothing$, then every infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain is an infinite minimal $(\mathcal{P}_{\mathcal{G}},\mathcal{R},\mu)$-chain and there is no infinite minimal $(\mathcal{P}_{\mathcal{X}},\mathcal{R},\mu)$-chain.*
2. *If $\mathcal{P} = \mathcal{P}^1_{\mathcal{X}}$, then there is no infinite $(\mathcal{P},\mathcal{R},\mu)$-chain.*

Proof.

1. By contradiction. Assume that there is an infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain containing any $u_i \to v_i \in \mathcal{P}_{\mathcal{X}}$. By Definition 5, such a pair must be followed by a pair $u_{i+1} \to v_{i+1} \in \mathcal{P}$ such that $\theta_i(\bar{s}_i^\sharp) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$ for some $\bar{s}_i \in \mathcal{NHT}$ and substitution θ_i. Therefore, $t'_i \in \mathcal{NHT}_{\mathcal{P}}$, but $\mathcal{NHT}_{\mathcal{P}} = \varnothing$, leading to a contradiction.
2. By contradiction. Assume that there is an infinite chain that only uses dependency pairs $u_i \to x_i \in \mathcal{P}^1_{\mathcal{X}}$ for all $i \geq 1$. Let $f_i = root(u_i)$ for $i \geq 1$. Then, by definition of $\mathcal{P}^1_{\mathcal{X}}$, for all $i \geq 1$, there is $j_i \in \{1,\ldots,ar(f_i)\} - \mu(f_i)$ such that $u_i|_{j_i} \trianglerighteq x_i$. According to Definition 5, we have that $\sigma(u_i)|_{j_i} \trianglerighteq \sigma(x_i) \trianglerighteq_\mu s_i$ for some term s_i such that $s_i^\sharp \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$. Since $root(s_i^\sharp) \in \mathcal{D}^\sharp \subseteq \mathcal{G}$ and $\mathcal{D}^\sharp \cap \mathcal{F} = \varnothing$ (Remark 4), no μ-rewriting step is possible at the root of $s_i^\sharp$. Thus, $root(s_i^\sharp) = root(u_{i+1}) = f_{i+1}$ and $j_{i+1} \notin \mu(f_{i+1})$. Since no μ-rewriting step is possible on the j_{i+1}-th immediate subterm $s_i^\sharp|_{j_{i+1}}$ of $s_i^\sharp$, it follows that $s_i^\sharp|_{j_{i+1}} = \sigma(u_{i+1})|_{j_{i+1}} \trianglerighteq \sigma(x_{i+1})$, i.e., $\sigma(x_i) \triangleright \sigma(x_{i+1})$ for all $i \geq 1$. We get an infinite sequence $\sigma(x_1) \triangleright \sigma(x_2) \triangleright \cdots$ which contradicts well-foundedness of $\triangleright$.

□

The following proposition establishes some important 'basic' cases of (absence of) infinite context-sensitive chains of pairs which are used later.

Proposition 9. *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$.*

1. *If $P = \varnothing$, then every $(\mathcal{P},\mathcal{R},\mu)$-chain is empty.*
2. *If $R = \varnothing$, then there is no infinite $(\mathcal{P}_{\mathcal{X}},\mathcal{R},\mu)$-chain.*
3. *Let $u \to v \in \mathcal{P}_{\mathcal{G}}$ be such that $v = \theta(u)$. Then, there is an infinite $(\mathcal{P},\mathcal{R},\mu)$-chain.*

Proof.

1. Obvious, by Definition 5.
2. By contradiction. If there is an infinite $(\mathcal{P}_\mathcal{X}, \mathcal{R}, \mu)$-chain, then, since there is no rule in $\mathcal{R}$, there is a substitution σ such that

$$\sigma(u_1) \hookrightarrow_{\mathcal{P},\mu} \sigma(x_1) \trianglerighteq^\sharp_\mu t_1 = \sigma(u_2) \hookrightarrow_{\mathcal{P},\mu} \sigma(x_2) \trianglerighteq^\sharp_\mu t_2 = \sigma(u_3) \cdots$$

   where $t_i = s_i^\sharp$ for some terms $s_i \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ such that $\sigma(x_i) = C_i[s_i]_{p_i}$ for some $C_i[\ ]_{p_i}$ and $p_i \in \mathcal{P}os^\mu(C_i[\ ]_{p_i})$ such that $sprefix(p_i) \subseteq \mathcal{F}$ for $i \geq 1$. Since $x_i \in \mathcal{V}ar(u_i)$ and u_i is not a variable, we have $u_i \triangleright x_i$; hence, $\sigma(u_i) \triangleright \sigma(x_i)$ (by stability of $\triangleright$) and also $\sigma(u_i) \triangleright s_i$ for all $i \geq 1$. Since s_i and $\sigma(u_{i+1})$ only differ in the root symbol, we can actually say that $s_i \triangleright s_{i+1}$ for all $i \geq 1$. Thus, we obtain an infinite sequence $s_1 \triangleright s_2 \triangleright \cdots$ that contradicts the well-foundedness of $\triangleright$.
3. Trivial.

□

The following example shows that Proposition 9(2) does not hold for TRSs $\mathcal{P}$ with arbitrary rules.

Example 12. *Consider* $\mathcal{P} = \{\mathtt{F}(x) \to x, \mathtt{G}(x) \to \mathtt{F}(\mathrm{g}(x))\}$ *together with a TRS* $\mathcal{R}$ *with an emtpy set of rules:* $\mathcal{R} = (\{\mathrm{g}\}, \varnothing)$. *Let* μ *be given by* $\mu(f) = \varnothing$ *for all* $f \in \mathcal{F} \cup \mathcal{G}$. *Note that* $\mathcal{P}_\mathcal{X}$ *consists of the pair* $\mathtt{F}(x) \to x$ *because* $x \in \mathcal{V}ar(\mathtt{F}(x)) - \mathcal{V}ar^\mu(\mathtt{F}(x))$. *Then, we have an infinite chain*

$$\mathtt{F}(\mathrm{g}(x)) \hookrightarrow_{\mathcal{P},\mu} \mathrm{g}(x) \trianglerighteq^\sharp_\mu \mathtt{G}(x) \hookrightarrow_{\mathcal{P},\mu} \mathtt{F}(\mathrm{g}(x)) \hookrightarrow_{\mathcal{R},\mu} \cdots$$

Since $\mathcal{NHT} = \varnothing$, $\mathrm{g}(x)$ *is not an instance of any term in* $\mathcal{NHT}$. *Thus, the chain is* not *minimal.*

5.2. Chains of CSDPs vs. chains of DPs

The definition of chain of CSDPs differs from the one for DPs. First, we use $\hookrightarrow^*$ instead of $\to^*$ for connecting pairs. Also, we require μ-termination instead of termination for minimal chains. However, the most important difference concerns the treatment of collapsing pairs. In general (and in sharp contrast with the DP approach), the connection between the right-hand side of a collapsing pair (which is a variable, e.g., x) and the left-hand side u of the next pair in the chain depends on whether a marked *narrowable hidden term* (which is introduced by a *previous* μ-rewriting step) μ-rewrites into $\sigma(u)$. Dealing with collapsing pairs, hidden terms can be thought of as playing the role of *hidden* or *delayed* recursive paths. This fits the guiding idea of the DP approach as an analysis of rewriting-based recursion paths in function calls (as briefly discussed in the introduction).

6. Characterizing termination of *CSR* using chains of CSDPs

The following result establishes the soundness of the CSDP approach.

Theorem 2 (Soundness). *Let* $\mathcal{R}$ *be a TRS and* $\mu \in M_\mathcal{R}$. *Then,* $\mathcal{R}$ *is* μ*-terminating if there is no infinite minimal* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$*-chain.*

Proof. By contradiction. If $\mathcal{R}$ is not μ-terminating, then there is $t \in \mathcal{T}_{\infty,\mu}$ (Lemma 2). By Theorem 1, there are rules $l_i \to r_i \in \mathcal{R}$, substitutions σ_i, and terms $t_i \in \mathcal{M}_{\infty,\mu}$, for $i \geq 1$ such that

$$t = t_0 \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \sigma_1(l_1) \stackrel{\Lambda}{\hookrightarrow} \sigma_1(r_1) \trianglerighteq_\mu t_1 \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \sigma_2(l_2) \stackrel{\Lambda}{\hookrightarrow} \sigma_2(r_2) \trianglerighteq_\mu t_2 \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \cdots$$

where either (D1) $t_i = \sigma_i(s_i)$ for some s_i such that $r_i \trianglerighteq_\mu s_i$ or (D2) $\sigma_i(x_i) \trianglerighteq_\mu t_i$ for some $x_i \in \mathcal{V}ar^\mu(r_i) - \mathcal{V}ar^\mu(l_i)$ and $t_i = \theta_i(t_i')$ for some $t_i' \in \mathcal{NHT}$. Furthermore, since $t_{i-1} \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \sigma_i(l_i)$ and $t_{i-1} \in \mathcal{M}_{\infty,\mu}$ (in particular, $t_0 = t \in \mathcal{T}_{\infty,\mu} \subseteq \mathcal{M}_{\infty,\mu}$), by Lemma 4, $\sigma_i(l_i) \in \mathcal{M}_{\infty,\mu}$ for all $i \geq 1$. Note that, since $t_i \in \mathcal{M}_{\infty,\mu}$, we have that $t_i^\sharp$ is μ-terminating (with respect to $\mathcal{R}$), because all μ-replacing subterms of t_i (hence of $t_i^\sharp$ as well) are μ-terminating and $root(t_i^\sharp)$ is not a defined symbol of $\mathcal{R}$.

First, note that $\mathsf{DP}(\mathcal{R},\mu)$ is a TRS $\mathcal{P}$ over the signature $\mathcal{G} = \mathcal{F} \cup \mathcal{D}^\sharp$ and $\mu^\sharp \in M_{\mathcal{F}\cup\mathcal{G}}$ as required by Definition 5. Furthermore, $\mathcal{P}_\mathcal{G} = \mathsf{DP}_\mathcal{F}(\mathcal{R},\mu)$ and $\mathcal{P}_\mathcal{X} = \mathsf{DP}_\mathcal{X}(\mathcal{R},\mu)$. We can define an infinite minimal $(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mu^\sharp)$-chain using CSDPs $u_i \to v_i$ for $i \geq 1$, where $u_i = l_i^\sharp$ and

1. $v_i = s_i^\sharp$ if (D1) holds. Since $t_i \in \mathcal{M}_{\infty,\mu}$, we have that $root(s_i) \in \mathcal{D}$ and, because $t_i = \sigma_i(s_i)$, by Corollary 2, $\textsc{Ren}^\mu(s_i)$ is μ-narrowable. Furthermore, if we assume that s_i is a μ-replacing subterm of l_i (i.e., $l_i \triangleright_\mu s_i$), then $\sigma_i(l_i) \triangleright_\mu \sigma_i(s_i)$. Since $\sigma_i(s_i) = t_i \in \mathcal{M}_{\infty,\mu}$, this contradicts that $\sigma_i(l_i) \in \mathcal{M}_{\infty,\mu}$. Thus, $l_i \not\trianglerighteq_\mu s_i$. Hence, $u_i \to v_i \in \mathsf{DP}_\mathcal{F}(\mathcal{R},\mu)$. Furthermore, $t_i^\sharp = \sigma_i(v_i)$ is μ-terminating. Finally, since $t_i = \sigma_i(s_i) \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \sigma_{i+1}(l_{i+1})$ and $\mu^\sharp$ extends μ to $\mathcal{F} \cup \mathcal{D}^\sharp$ by $\mu^\sharp(f^\sharp) = \mu(f)$ for all $f \in \mathcal{D}$, we also have that $\sigma_i(v_i) = \sigma_i(s_i^\sharp) \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma_{i+1}(u_{i+1})$.
2. $v_i = x_i$ if (D2) holds. Clearly, $u_i \to v_i \in \mathsf{DP}_\mathcal{X}(\mathcal{R},\mu)$. As discussed above, $t_i^\sharp$ is μ-terminating. Since $\sigma_i(x_i) \in \mathcal{T}(\mathcal{F},\mathcal{X})$ and $\sigma_i(x_i) \trianglerighteq_\mu t_i$, we have that $\sigma(v_i) = C_i[t_i]_{p_i}$ for some $C_i[\;]_{p_i}$ and $p_i \in \mathcal{P}os^\mu(C_i[\;]_{p_i})$ such that $sprefix_{C_i[\;]_{p_i}}(p_i) \subseteq \mathcal{F}$. Finally, since $t_i \stackrel{>\Lambda}{\hookrightarrow}{}^{*} \sigma_{i+1}(l_{i+1})$, again we have that $t_i^\sharp \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma_{i+1}(u_{i+1})$. Furthermore, $t_i = \theta_i(t_i')$ for some $t_i' \in \mathcal{NHT}$ and substitution θ_i.

Regarding σ, w.l.o.g. we can assume that $\mathcal{V}ar(l_i) \cap \mathcal{V}ar(l_j) = \varnothing$ for all $i \neq j$, and therefore $\mathcal{V}ar(u_i) \cap \mathcal{V}ar(u_j) = \varnothing$ as well. Then, σ is given by $\sigma(x) = \sigma_i(x)$ whenever $x \in \mathcal{V}ar(u_i)$ for $i \geq 1$. From the discussion in (1) and (2) above, we conclude that the CSDPs $u_i \to v_i$ together with σ define an infinite minimal $(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mu^\sharp)$-chain. This leads to a contradiction. □

Let $\mathsf{DP}^1_\mathcal{X}(\mathcal{R},\mu) = \mathcal{P}^1_\mathcal{X}$ for $\mathcal{P} = \mathsf{DP}(\mathcal{R},\mu)$. By Theorem 2 and Propositions 8 and 9, we have the following.

Corollary 3 (Basic μ-termination criteria). *Let $\mathcal{R}$ be a TRS and $\mu \in M_\mathcal{R}$.*

1. *If $\mathsf{DP}(\mathcal{R},\mu) = \varnothing$, then $\mathcal{R}$ is μ-terminating.*
2. *If $\mathcal{NHT}_{\mathsf{DP}(\mathcal{R},\mu)}(\mathcal{R},\mu) = \varnothing$ and $\mathsf{DP}_\mathcal{F}(\mathcal{R},\mu) = \varnothing$, then $\mathcal{R}$ is μ-terminating.*
3. *If $\mathsf{DP}(\mathcal{R},\mu) = \mathsf{DP}^1_\mathcal{X}(\mathcal{R},\mu)$, then $\mathcal{R}$ is μ-terminating.*

Example 13. *Consider the following TRS $\mathcal{R}$ [Luc98, Example 15]:*

$$\begin{array}{rcl@{\qquad}rcl}
\mathtt{and}(\mathtt{true},x) & \to & x & \mathtt{add}(\mathtt{0},x) & \to & x \\
\mathtt{and}(\mathtt{false},y) & \to & \mathtt{false} & \mathtt{add}(\mathtt{s}(x),y) & \to & \mathtt{s}(\mathtt{add}(x,y)) \\
\mathtt{if}(\mathtt{true},x,y) & \to & x & \mathtt{from}(x) & \to & \mathtt{cons}(x,\mathtt{from}(\mathtt{s}(x))) \\
\mathtt{if}(\mathtt{false},x,y) & \to & y & \mathtt{first}(\mathtt{0},x) & \to & \mathtt{nil} \\
\mathtt{first}(\mathtt{s}(x),\mathtt{cons}(y,z)) & \to & \mathtt{cons}(y,\mathtt{first}(x,z)) & & &
\end{array}$$

together with the canonical *replacement map* $\mu(\mathtt{cons}) = \mu(\mathtt{s}) = \mu(\mathtt{from}) = \varnothing$, $\mu(\mathtt{add}) = \mu(\mathtt{and}) = \mu(\mathtt{if}) = \{1\}$, *and* $\mu(\mathtt{first}) = \{1, 2\}$, *which ensures completeness of* CSR *for computing head-normal forms*[3] *with* $\mathcal{R}$ *(see [Luc98, Luc02]). Then,* $\mathsf{DP}(\mathcal{R}, \mu) = \mathsf{DP}^1_{\mathcal{X}}(\mathcal{R}, \mu)$ *is:*

$$\begin{array}{rclcrcl}
\mathtt{AND}(\mathtt{true}, x) & \to & x & \qquad & \mathtt{IF}(\mathtt{true}, x, y) & \to & x \\
\mathtt{ADD}(\mathtt{0}, x) & \to & x & & \mathtt{IF}(\mathtt{false}, x, y) & \to & y
\end{array}$$

Note also that $\mathcal{NHT}_{\mathsf{DP}(\mathcal{R},\mu)} = \varnothing$*. Thus, by either of the last two statements of Corollary 3, we conclude the* μ*-termination of* $\mathcal{R}$*.*

The following example shows that Corollary 3(3) does *not* hold for chains consisting of *arbitrary* collapsing CSDPs.

Example 14. *Consider the CS-TRS* $(\mathcal{R}, \mu)$ *in Example 3. Note that* $\mathsf{DP}(\mathcal{R}, \mu) = \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ *(both* $\mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$ *and* $\mathsf{DP}^1_{\mathcal{X}}(\mathcal{R}, \mu)$ *are empty!). We have the following infinite* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$*-chain:*

$$\mathtt{F}(\underline{\mathtt{a}}) \hookrightarrow_{\mathcal{R},\mu^\sharp} \underline{\mathtt{F}(\mathtt{c}(\mathtt{f}(\mathtt{a})))} \hookrightarrow_{\mathsf{DP}(\mathcal{R},\mu),\mu^\sharp} \mathtt{F}(\underline{\mathtt{a}}) \hookrightarrow_{\mathcal{R},\mu^\sharp} \cdots$$

Now, we prove that the previous CS-dependency pair approach is not only correct but also complete for proving termination of *CSR*.

Theorem 3 (Completeness). *Let* $\mathcal{R}$ *be a TRS and* $\mu \in M_{\mathcal{R}}$*. If* $\mathcal{R}$ *is* μ*-terminating, then there is no infinite* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$*-chain.*

Proof. By contradiction. If there is an infinite $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$-chain, then there are a substitution σ and dependency pairs $u_i \to v_i \in \mathsf{DP}(\mathcal{R}, \mu)$ such that

1. $\sigma(v_i) \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_{i+1})$, if $u_i \to v_i \in \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$, and
2. if $u_i \to v_i = u_i \to x_i \in \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$, then there is $s_i \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ such that $\sigma(x_i) \trianglerighteq_\mu s_i$ and $s^\sharp_i \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_{i+1})$.

for $i \geq 1$. Now, consider the first dependency pair $u_1 \to v_1$ in the sequence:

1. If $u_1 \to v_1 \in \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$, then $v^\natural_1$ is a μ-replacing subterm of the right-hand-side r_1 of a rule $l_1 \to r_1$ in $\mathcal{R}$. Therefore, $r_1 = C_1[v^\natural_1]_{p_1}$ for some position $p_1 \in \mathcal{P}os^\mu(r_1)$ and context $C_1[\,]_{p_1}$, and we can perform the μ-rewriting step $t_1 = \sigma(u_1) \hookrightarrow_{\mathcal{R},\mu} \sigma(r_1) = \sigma(C_1)[\sigma(v^\natural_1)]_{p_1} = s_1$, where $\sigma(v^\natural_1)^\sharp = \sigma(v_1) \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_2)$ and $\sigma(u_2)$ initiates an infinite $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$-chain. Note that $p_1 \in \mathcal{P}os^\mu(s_1)$.
2. If $u_1 \to x \in \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$, then there is a rule $l_1 \to r_1$ in $\mathcal{R}$ such that $u_1 = l^\sharp_1$, and $x \in \mathcal{V}ar^\mu(r_1) - \mathcal{V}ar^\mu(l_1)$, i.e., $r_1 = C_1[x]_{q_1}$ for some $q_1 \in \mathcal{P}os^\mu(r_1)$. Furthermore, since $\sigma(x) = C'_1[s]_{p'_1}$ for some term s, $C'_1[\;]_{p'_1}$ and $p'_1 \in \mathcal{P}os^\mu(C'_1[\;]_{p'_1})$ such that $s^\sharp \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_2)$, we can perform the μ-rewriting step $t_1 = \sigma(l_1) \hookrightarrow_{\mathcal{R},\mu} \sigma(r_1) = \sigma(C_1)[C'_1[s]_{p'_1}]_{q_1} = s_1$ where $s^\sharp \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_2)$ (hence $s \xhookrightarrow{>\Lambda}{}^*_{\mathcal{R},\mu} u^\natural_2$) and $\sigma(u_2)$ initiates an infinite $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$-chain. Note that $p_1 = q_1.p'_1 \in \mathcal{P}os^\mu(s_1)$ (use Proposition 1).

---

[3] A head-normal form is a term that cannot be rewritten to a redex.

Since $\mu^\sharp(f^\sharp) = \mu(f)$, and $p_1 \in \mathcal{P}os^\mu(s_1)$, we have that $s_1 \hookrightarrow^*_{\mathcal{R},\mu} t_2[\sigma(u_2)]_{p_1} = t_2$ and $p_1 \in \mathcal{P}os^\mu(t_2)$. Thus, we can build an infinite μ-rewrite sequence $t_1 \hookrightarrow_{\mathcal{R},\mu} s_1 \hookrightarrow^*_{\mathcal{R},\mu} t_2 \hookrightarrow_{\mathcal{R},\mu} \cdots$ which contradicts the μ-termination of $\mathcal{R}$. $\square$

Proposition 9(3) suggests a simple checking of *non-μ-termination*.

Corollary 4 (Non-μ-termination criterion). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. If there is $u \to v \in \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$ such that $v' = \theta(u)$ for some substitution θ and renamed version v' of v, then $\mathcal{R}$ is not μ-terminating.*

As a corollary of Theorems 2 and 3, we have:

Corollary 5 (Characterization of μ-termination). *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Then, $\mathcal{R}$ is μ-terminating if and only if there is no infinite minimal $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$-chain.*

7. Mechanizing proofs of μ-termination using CSDPs

Over the last ten years, the dependency pair method has evolved to a powerful technique for proving termination of TRSs in practice. In the DP-approach [AG00], the starting point is a TRS $\mathcal{R}$ from which a set of dependency pairs $\mathsf{DP}(\mathcal{R})$ is obtained. Then, these dependency pairs are organized in a dependency graph $\mathsf{DG}(\mathcal{R})$ whose nodes are the pairs in $\mathsf{DP}(\mathcal{R})$ and where the arcs are obtained by investigating possible *rewriting connections* between (instances of) the right-hand sides of the pairs and (instances of) the left-hand sides of other (not necessarily distinct) pairs. The cycles of the graph are analyzed to show that no infinite chains of DPs can be obtained from them [GAO02]. In this sense, the treatment of strongly connected components of the graph (SCCs) instead of cycles [HM04, HM05] brought an important improvement to the practical use of this approach.

In the DP-approach, the components $u_i \to v_i$ of the chains (or cycles) are dependency pairs, i.e., $u_i \to v_i \in \mathsf{DP}(\mathcal{R})$ for all $i \geq 1$. Since they only make sense when an underlying TRS $\mathcal{R}$ is given as the source of the dependency pairs, transforming DPs is possible (the *narrowing* transformation is already described in [AG00]) but only as a final step because, afterwards, they are no longer dependency pairs of the original TRS. The dependency pair framework [GTS04, GTSF06] solves this problem in a clear way, leading to a more powerful mechanization of termination proofs. The central notion now is that of *DP problem* [GTSF06, page 158]: given a TRS $\mathcal{R}$ and a set of pairs $\mathcal{P}$, the goal is to verify the absence of infinite (minimal) chains. In this case, the DP problem is called *finite*. Termination of a TRS $\mathcal{R}$ is addressed as a DP problem[4] $(\mathcal{P}, \mathcal{R})$ where $\mathcal{P} = \mathsf{DP}(\mathcal{R})$: $\mathcal{R}$ is terminating if this problem is finite. The most important notion regarding the mechanization of the proofs is the notion of *processor*. Formally, *a DP processor is a function* Proc *that takes a DP problem as input and returns a new set of DP problems that then have to be solved instead. Alternatively, it can also return* "no" [GTSF06, page 159]. In the following we adapt the notions of [GTSF06] to *CSR*.

---

[4] The original definition in [GTSF06] includes an extra parameter e, which specifies two kinds of problems: $e = \mathbf{t}$ for termination problems, and $e = \mathbf{i}$ for *innermost* termination problems.

7.1. CS problems, CS processors, and the CSDP-framework

In our definition of *DP problem* for *CSR*, we prefer to avoid '*DP*' because, as discussed above, dependency pairs (as such) are relevant in the theoretical framework only for investigating a particular problem (termination of TRSs), whereas some transformations can yield sets of pairs which are no longer dependency pairs of the underlying TRS.

Definition 6 (CS problem). *A CS problem τ is a tuple $\tau = (\mathcal{P}, \mathcal{R}, \mu)$, where $\mathcal{R}$ and $\mathcal{P}$ are TRSs and $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$. The CS problem τ is finite if there is no infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain. The CS problem τ is infinite if $\mathcal{R}$ is non-μ-terminating or there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain.*

Remark 8. *As in the standard DP framework (see the discussion and further motivation in [GTSF06, page 159]), the inclusion of the case when $\mathcal{R}$ is nonterminating as part of the definition of infinite problem is essential for dealing with some specific transformations of CS problems (see Theorems 8 and 16 below)*

Definition 7 (CS processor). *A CS processor* Proc *is a mapping from CS problems into sets of CS problems. Alternatively, it can also return* "no"*. A CS processor* Proc *is*

- sound *if for all CS problems τ, we have that (1) τ is finite whenever* $\mathsf{Proc}(\tau) \neq \mathsf{no}$ *and (2)* $\forall\tau' \in \mathsf{Proc}(\tau)$, *$\tau'$ is finite.*
- complete *if for all CS problems τ, we have that (1) τ is infinite whenever* $\mathsf{Proc}(\tau) = \mathsf{no}$ *or (2)* $\exists\tau' \in \mathsf{Proc}(\tau)$ *such that τ' is infinite.*

A (sound) processor transforms DP problems into (hopefully) *simpler* ones, in such a way that the existence of an infinite chain in the original DP problem implies the existence of an infinite chain in the transformed one. Here, 'simpler' usually means that fewer pairs are involved. Soundness is essential for proving *termination*. Completeness is necessary for proving *nontermination*.

Processors are used in a *divide and conquer* scheme to incrementally simplify the original CS problem as much as possible, possibly decomposing it into smaller pieces which are then independently treated in the very same way. The trivial case comes when the set of pairs $\mathcal{P}$ becomes empty. Then, no infinite chain is possible, and we can provide a *positive* answer yes to the CS problem which is propagated upwards to the original problem in the root of the decision tree. In some cases, it is also possible to witness the existence of infinite chains for a given CS problem; then a *negative* answer no can be provided and propagated upwards.

Theorem 4 (CSDP-framework). *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. We construct a tree whose nodes are labeled with CS problems or* "yes" *or* "no"*, and whose root is labeled with* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$. *For every inner node labeled with τ, there is a sound processor* Proc *satisfying one of the following conditions:*

1. $\mathsf{Proc}(\tau) = \mathsf{no}$ *and the node has just one child that is labeled with* "no".
2. $\mathsf{Proc}(\tau) = \varnothing$ *and the node has just one child that is labeled with* "yes".
3. $\mathsf{Proc}(\tau) \neq \mathsf{no}$, $\mathsf{Proc}(\tau) \neq \varnothing$, *and the children of the node are labeled with the CS problems in* $\mathsf{Proc}(\tau)$.

If all leaves of the tree are labeled with "yes"*, then $\mathcal{R}$ is μ-terminating. Otherwise, if there is a leaf labeled with* "no" *and if all processors used on the path from the root to this leaf are complete, then $\mathcal{R}$ is not μ-terminating.*

Propositions 8 and 9 are the basis for the following sound and complete processors, which provide some *base* cases for our proofs of termination of *CSR*.

Theorem 5 (Basic CS processors). *Let* $\mathcal{R} = (\mathcal{F}, R)$ *and* $\mathcal{P} = (\mathcal{G}, P)$ *be TRSs and* $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. *Then, the processors* Proc_{Fin} *and* Proc_{Inf} *given by*[5]

$$\mathsf{Proc}_{Fin}(\mathcal{P},\mathcal{R},\mu) = \begin{cases} \varnothing & \textit{if } P = \varnothing \vee \mathcal{P} = \mathcal{P}^1_{\mathcal{X}} \vee (R = \varnothing \wedge \mathcal{P} = \mathcal{P}_{\mathcal{X}}); \textit{and} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{cases}$$

$$\mathsf{Proc}_{Inf}(\mathcal{P},\mathcal{R},\mu) = \begin{cases} \mathsf{no} & \textit{if } v = \theta(u) \\ & \textit{for some } u \to v \in \mathcal{P}_{\mathcal{G}} \textit{ and substitution } \theta; \textit{and} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{cases}$$

are sound and complete.

In the following sections we describe several sound and (most of them) complete CS processors.

8. Context-Sensitive Dependency Graph

In the dependency pairs approach [AG00], a *dependency graph* $\mathsf{DG}(\mathcal{R})$ is associated to the TRS $\mathcal{R}$. The nodes of $\mathsf{DG}(\mathcal{R})$ are the dependency pairs in $\mathsf{DP}(\mathcal{R})$; there is an arc from a dependency pair $u \to v$ to a dependency pair $u' \to v'$ such that $\mathcal{V}ar(u) \cap \mathcal{V}ar(u') = \varnothing$ if $\theta(v) \to^*_{\mathcal{R}} \theta(u')$ for some substitution θ. In [GTSF06], a more general notion of *graph of pairs* $\mathsf{DG}(\mathcal{P},\mathcal{R})$ associated to a set of pairs $\mathcal{P}$ and a TRS $\mathcal{R}$ is considered. Pairs in $\mathcal{P}$ are now used as the nodes of the graph, but they are connected by $\mathcal{R}$-rewriting in the same way [GTSF06, Definition 7]. The analysis of the cycles in the graph that is built from such pairs is useful for investigating the existence of infinite (minimal) chains of pairs. In the following section we take into account these points to provide an appropriate definition of context-sensitive (dependency) graph.

8.1. Definition of the context-sensitive dependency graph

Given TRSs $\mathcal{R}$ and $\mathcal{P}$ and a replacement map $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$, we want to obtain a notion of graph that is able to represent all infinite *minimal* chains of pairs as given in Definition 5.

Definition 8 (Context-Sensitive Graph of Pairs). *Let* $\mathcal{R}$ *and* $\mathcal{P}$ *be TRSs and* $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$. *The* context-sensitive (CS-)graph $\mathsf{G}(\mathcal{P},\mathcal{R},\mu)$ *has* $\mathcal{P}$ *as the set of nodes. Given* $u \to v, u' \to v' \in \mathcal{P}$, *there is an arc from* $u \to v$ *to* $u' \to v'$ *if* $u \to v, u' \to v'$ *is a minimal* $(\mathcal{P},\mathcal{R},\mu)$*-chain for some substitution* σ.

In termination proofs, we are concerned with the so-called *strongly connected components* (SCCs) of the dependency graph, rather than with the cycles themselves (which are exponentially many) [HM05]. A strongly connected component in a graph is a *maximal cycle*, i.e., a cycle that is not contained in any other cycle. The following result justifies the use of SCCs for proving the absence of infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chains.

---

[5] In the following, we often write $\mathsf{Proc}(\mathcal{P},\mathcal{R},\mu)$ instead of $\mathsf{Proc}((\mathcal{P},\mathcal{R},\mu))$ to avoid duplicated parentheses.

Theorem 6 (SCC processor). *Let $\mathcal{R}$ and $\mathcal{P}$ be TRSs and $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$. Then, the processor* Proc_{SCC} *given by*

$$\mathsf{Proc}_{SCC}(\mathcal{P},\mathcal{R},\mu) = \{(\mathcal{Q},\mathcal{R},\mu) \mid \mathcal{Q} \text{ are the pairs of an SCC in } \mathsf{G}(\mathcal{P},\mathcal{R},\mu)\}$$

is sound and complete.

Proof. We prove soundness by contradiction. Assume that Proc_{SCC} is not sound. Then, there is a CS problem $\tau = (\mathcal{P},\mathcal{R},\mu)$ such that, for all $\tau' \in \mathsf{Proc}_{SCC}(\tau)$, τ' is finite but τ is not finite. Thus, there is an infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain A. Since $\mathcal{P}$ contains a finite number of pairs, there is $\mathcal{P}' \subseteq \mathcal{P}$ and a tail B of A, which is an infinite minimal $(\mathcal{P}',\mathcal{R},\mu)$-chain where all pairs in $\mathcal{P}'$ are infinitely often used. According to Definition 8, this means that $\mathcal{P}'$ is a cycle in $\mathsf{G}(\mathcal{P},\mathcal{R},\mu)$. Hence $\mathcal{P}'$ belongs to some SCC with nodes in $\mathcal{Q}$, i.e., $\mathcal{P}' \subseteq \mathcal{Q}$. Thus, B is an infinite minimal $(\mathcal{Q},\mathcal{R},\mu)$-chain, i.e., $\tau' = (\mathcal{Q},\mathcal{R},\mu)$ is not finite. Since $\tau' \in \mathsf{Proc}_{SCC}(\tau)$, we obtain a contradiction.

With regard to completeness, since $\mathcal{Q} \subseteq \mathcal{P}$ for some SCC in $\mathsf{G}(\mathcal{P},\mathcal{R},\mu)$ with nodes in $\mathcal{Q}$, every infinite minimal $(\mathcal{Q},\mathcal{R},\mu)$-chain is an infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain. Hence, the processor is complete as well. □

As a consequence of this theorem, we can *separately* work with the strongly connected components of $\mathsf{G}(\mathcal{P},\mathcal{R},\mu)$, disregarding other parts of the graph. Now we can use these notions to introduce the context-sensitive dependency graph.

Definition 9 (Context-Sensitive Dependency Graph). *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. The* Context-Sensitive Dependency Graph *(CSDG) for $\mathcal{R}$ and μ is* $\mathsf{DG}(\mathcal{R},\mu) = \mathsf{G}(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mu^\sharp)$.

8.2. Estimating the CS-dependency graph

In general, the context-sensitive graph is *not* computable: it involves reachability of $\sigma(u')$ from $\sigma(v)$ (for $u \to v \in \mathcal{P}_{\mathcal{G}}$) or $\sigma(t^\sharp)$ (for $t \in \mathcal{NHT}_{\mathcal{P}}$) using *CSR*. Since the reachability problem for *CSR* is undecidable, we need to use some approximation of it.

Remark 9. *Several estimations of the dependency graph were investigated in [AG00, HM05, GTS05, Mid01, Mid02]. The first one, introduced in [AG00], was adapted to* CSR *in [AGL06].*

Following [GTS05], we describe how to approximate the CS-dependency graph of a CS-TRS. Given a TRS $\mathcal{R}$ and a replacement map μ, we let $\textsc{tcap}^{\mu}_{\mathcal{R}}$ be as follows:

$$\begin{array}{rcl} \textsc{tcap}^{\mu}_{\mathcal{R}}(x) & = & y \text{ if } x \text{ is a variable, and} \\ \textsc{tcap}^{\mu}_{\mathcal{R}}(f(t_1,\ldots,t_k)) & = & \begin{cases} f([t_1]^f_1,\ldots,[t_k]^f_k) & \text{if } f([t_1]^f_1,\ldots,[t_k]^f_k) \text{ does not unify} \\ & \text{with } l \text{ for any } l \to r \text{ in } \mathcal{R} \\ y & \text{otherwise} \end{cases} \end{array}$$

where y is intended to be a new, fresh variable that has not yet been used and given a term s, $[s]^f_i = \textsc{tcap}^{\mu}_{\mathcal{R}}(s)$ if $i \in \mu(f)$ and $[s]^f_i = s$ if $i \notin \mu(f)$. We assume that l shares no variable with $f([t_1]^f_1,\ldots,[t_k]^f_k)$ when the unification is attempted. Function $\textsc{tcap}^{\mu}_{\mathcal{R}}$ is intended to provide a suitable approximation of the aforementioned $(\mathcal{R},\mu)$-reachability problems by means of *unification*. The following result formalizes the correctness of this approach.

Proposition 10. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $t, u \in \mathcal{T}(\mathcal{F},\mathcal{X})$ be such that $\mathcal{V}ar(t) \cap \mathcal{V}ar(u) = \varnothing$. If $\theta(t) \hookrightarrow^* \theta(u)$ for some substitution θ, then $\textsc{tcap}^{\mu}_{\mathcal{R}}(t)$ and u unify.*

Proof. In the following, we let $s = \textsc{tcap}^{\mu}_{\mathcal{R}}(t)$. Note that, since $\mathcal{V}ar(t) \cap \mathcal{V}ar(u) = \varnothing$, we also have $\mathcal{V}ar(s) \cap \mathcal{V}ar(u) = \varnothing$. Clearly, $t = \sigma(s)$ for some substitution σ. We proceed by induction on the length m of the sequence from $\theta(t)$ to $\theta(u)$.

1. If $m = 0$, then $\theta(t) = \theta(\sigma(s)) = \theta(u)$. Since $\mathcal{V}ar(s) \cap \mathcal{V}ar(u) = \varnothing$, we can write $\theta(u) = \theta(\sigma(u))$, i.e., s and u unify.
2. If $m > 0$, then we have $\theta(t) \hookrightarrow t' \hookrightarrow^* \theta(u)$. Let $p \in \mathcal{P}os^{\mu}(\theta(t))$ be the position where the μ-rewrite step $\theta(t) \hookrightarrow t'$ is performed. By definition of $\textsc{tcap}^{\mu}_{\mathcal{R}}$, $s = s[z]_q$ for some fresh variable z and position q such that $q \leq p$. We can write $\theta(t) = \theta(s)$. Furthermore, since z is a fresh variable, we can write $t' = \theta(s)$ if we assume that $\theta(z) = t'|_q$. Thus, $\theta(s) \hookrightarrow^* \theta(u)$ in $m - 1$ steps. By the induction hypothesis, $\textsc{tcap}^{\mu}_{\mathcal{R}}(s)$ and u unify. Since $\textsc{tcap}^{\mu}_{\mathcal{R}}(s) = \textsc{tcap}^{\mu}_{\mathcal{R}}(\textsc{tcap}^{\mu}_{\mathcal{R}}(t))$ and $\textsc{tcap}^{\mu}_{\mathcal{R}}(\textsc{tcap}^{\mu}_{\mathcal{R}}(t))$ is just a renaming of $\textsc{tcap}^{\mu}_{\mathcal{R}}(t)$, the conclusion follows.

□

According to Proposition 10, given terms $t, u \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ that share no variable, and a substitution θ, the reachability of $\theta(u)$ from $\theta(t)$ by μ-rewriting can be *approximated* as unification of $\textsc{tcap}^{\mu}_{\mathcal{R}}(t)$ and u. Thus, taking into account Definitions 5 and 8, we have the following.

Definition 10 (Estimated Context-Sensitive Graph of Pairs). *Let $\mathcal{R}$ and $\mathcal{P}$ be TRSs and $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$. The* estimated *CS-graph associated to $\mathcal{R}$ and $\mathcal{P}$ (denoted $\mathsf{EG}(\mathcal{P}, \mathcal{R}, \mu)$) has $\mathcal{P}$ as the set of nodes and the arcs that connect them as follows:*

1. *There is an arc from $u \to v \in \mathcal{P}_{\mathcal{G}}$ to $u' \to v' \in \mathcal{P}$ if $\textsc{tcap}^{\mu}_{\mathcal{R}}(v)$ and u' unify.*
2. *There is an arc from $u \to v \in \mathcal{P}_{\mathcal{X}}$ to $u' \to v' \in \mathcal{P}$ if there is $t \in \mathcal{NHT}(\mathcal{R}, \mu)$ such that $\textsc{tcap}^{\mu}_{\mathcal{R}}(t^{\sharp})$ and u' unify.*

As a consequence of Proposition 10, we have the following.

Corollary 6 (Approximation of the context-sensitive graph). *Let $\mathcal{R}$ and $\mathcal{P}$ be TRSs and $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$. The* estimated *CS-graph $\mathsf{EG}(\mathcal{P}, \mathcal{R}, \mu)$ contains the CS-graph $\mathsf{G}(\mathcal{P}, \mathcal{R}, \mu)$.*

Therefore, we have the following *estimated* CSDG: $\mathsf{EDG}(\mathcal{R}, \mu) = \mathsf{EG}(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^{\sharp})$.

Remark 10. *Proposition 10 also provides* estimations *for $\mathcal{NHT}_{\mathcal{P}}$: if $t \in \mathcal{NHT}_{\mathcal{P}}$, then $\textsc{tcap}^{\mu}_{\mathcal{R}}(t^{\sharp})$ and u unify for some $u \to v \in \mathcal{P}$. In the following, we compute $\mathcal{NHT}_{\mathcal{P}}$ in this way.*

Example 15. *Consider again the CS-TRS $(\mathcal{R}, \mu)$ in Example 1. Note that*

$$\mathcal{NHT}_{\mathsf{DP}(\mathcal{R},\mu)}(\mathcal{R}, \mu^{\sharp}) = \{\texttt{oddNs}, \texttt{incr}(\texttt{oddNs}), \texttt{incr}(x), \texttt{zip}(xs, ys), \texttt{rep2}(xs)\}$$

The (estimated) CSDG in Figure 5 has four *cycles, each of which contains a single pair. We transform the CS problem $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^{\sharp})$ into a set*

$$\mathsf{Proc}_{SCC}(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^{\sharp}) = \left\{ (\{(1)\}, \mathcal{R}, \mu^{\sharp}),\ (\{(17)\}, \mathcal{R}, \mu^{\sharp}),\ (\{(21)\}, \mathcal{R}, \mu^{\sharp}),\ (\{(23)\}, \mathcal{R}, \mu^{\sharp}) \right\}$$

which contains four new (but very simple) CS problems.

Remark 11 (CSDG vs. DG). *Consider again $\mathcal{R}$ and μ as in Example 1. Pairs* (9) *and* (10) *belong to both $\mathsf{DG}(\mathcal{R})$ (see Figure 3) and $\mathsf{DG}(\mathcal{R}, \mu)$ (see Figure 5). However, they are* not *equally connected in $\mathsf{DG}(\mathcal{R})$ and $\mathsf{DG}(\mathcal{R}, \mu)$. The reason is that the collapsing pair* (25)*, that is* not *a node of $\mathsf{DG}(\mathcal{R})$, originates an* incoming *arc from both* (9) *and* (10).

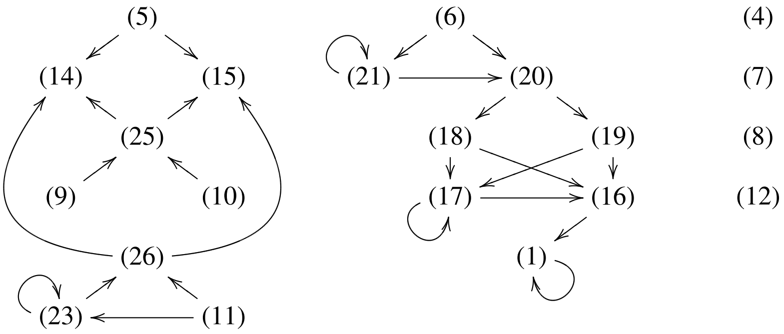

Figure 5: Context-Sensitive Dependency Graph for the CS-TRS in Example 1

9. Treating collapsing pairs

The following result shows how to *safely* transform collapsing pairs into noncollapsing ones in some particular cases.

Theorem 7 (Removing collapsing pairs). *Let* $\mathcal{R} = (\mathcal{F}, R)$ *and* $\mathcal{P} = (\mathcal{G}, P)$ *be TRSs and* $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$*. Let* $\mathcal{P}' = (\mathcal{G}', P')$ *where* $P' = (P - P_{\mathcal{X}}) \cup Q$ *for* $Q = \{u \to t^\sharp \mid u \to x \in \mathcal{P}_{\mathcal{X}}, t \in \mathcal{NHT}_{\mathcal{P}}\}$, $\mathcal{G}' = \mathcal{G}$ *if* $Q = \varnothing$, *and* $\mathcal{G}' = \mathcal{F} \cup \mathcal{G}$ *if* $Q \neq \varnothing$*. Then, the processor* Proc_{gNHT} *given by*

$$\mathsf{Proc}_{gNHT}(\mathcal{P}, \mathcal{R}, \mu) = \begin{cases} \{(\mathcal{P}', \mathcal{R}, \mu)\} & \textit{if } \mathcal{NHT}_{\mathcal{P}}(\mathcal{R}, \mu) \subseteq \mathcal{T}(\mathcal{F}) \\ \{(\mathcal{P}, \mathcal{R}, \mu)\} & \textit{otherwise} \end{cases}$$

is sound.

Proof. First, note that $\mathcal{P}'$ is a TRS: the new rules in Q are of the form $u \to t^\sharp$ for $t \in \mathcal{NHT}_{\mathcal{P}}$. Since $\mathcal{NHT}_{\mathcal{P}} \subseteq \mathcal{T}(\mathcal{F})$, we trivially have $\mathcal{V}ar(t^\sharp) \subseteq \mathcal{V}ar(u)$, i.e., $u \to t^\sharp$ is a rewrite rule. Furthermore, whenever $Q \neq \varnothing$, $\mathcal{G}'$ is the union of $\mathcal{F}$ and $\mathcal{G}$ to reflect the use of symbols in $\mathcal{F}$ coming from terms $t^\sharp$ for $t \in \mathcal{NHT}_{\mathcal{P}}(\mathcal{R}, \mu)$). Since we assume that $\mathcal{D}^\sharp \subseteq \mathcal{G}$ (Remark 4), $root(t^\sharp) \in \mathcal{D}^\sharp \subseteq \mathcal{G} \subseteq \mathcal{G}'$.

We prove that the existence of an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain implies the existence of an infinite minimal $(\mathcal{P}', \mathcal{R}, \mu)$-chain. Consider an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain:

$$\sigma(u_1) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^\sharp_\mu t_1 \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^\sharp_\mu t_2 \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_3) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^\sharp_\mu \cdots$$

for some substitution σ, where, according to Definition 5, for all $i \geq 1$, t_i is μ-terminating and, (1) if $u_i \to v_i \in \mathcal{P}_{\mathcal{G}}$, then $t_i = \sigma(v_i)$ and (2) if $u_i \to v_i = u_i \to x_i \in \mathcal{P}_{\mathcal{X}}$, then $t_i = s_i^\sharp$ for some s_i such that $\sigma(x_i) \trianglerighteq_\mu s_i$ and $s_i = \theta_i(\bar{s}_i)$ for some $\bar{s}_i \in \mathcal{NHT}$ and substitution θ_i. Actually, since $t_i = s_i^\sharp = \theta_i(\bar{s}_i)^\sharp = \theta_i(\bar{s}_i^\sharp)$ and $t_i \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$, we can further say that $\bar{s}_i \in \mathcal{NHT}_{\mathcal{P}}$.

In case (2), since $\mathcal{NHT}_{\mathcal{P}} \subseteq \mathcal{T}(\mathcal{F})$, we have $t_i = s_i^\sharp = \theta_i(\bar{s}_i^\sharp) = \bar{s}_i^\sharp$, i.e., $t_i \in \mathcal{NHT}_{\mathcal{P}}$. Thus, we can use $u_i \to t_i \in Q$ instead of $u_i \to x_i \in \mathcal{P}_{\mathcal{X}}$, because we still have $t_i \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$. In this way, by replacing each $u_i \to x_i \in \mathcal{P}_{\mathcal{X}}$ by the corresponding $u_i \to t_i \in Q$, each step $\sigma(u_i) \hookrightarrow_{\mathcal{P},\mu} \circ \trianglerighteq^\sharp_\mu t_i$

becomes a step $\sigma(u_i) \hookrightarrow_{\mathcal{P}',\mu} t_i$, whereas steps $\sigma(u_i) \hookrightarrow_{\mathcal{P},\mu} \sigma(v_i) = t_i$ for $u_i \to v_i \in \mathcal{P}_{\mathcal{G}}$ remain unchanged. Thus, we obtain an infinite minimal $(\mathcal{P}', \mathcal{R}, \mu)$-chain, as desired. $\square$

Note that no pair in $\mathcal{P}'$ in Theorem 7 is collapsing. Unfortunately, Proc_{gNHT} is *not* complete.

Example 16. *Consider the following TRS:*

$$\begin{array}{rcl} \mathtt{b} & \to & \mathtt{f}(\mathtt{c}(\mathtt{b})) \\ \mathtt{f}(x) & \to & x \end{array}$$

together with the replacement map μ given by $\mu(\mathtt{f}) = \mu(\mathtt{c}) = \varnothing$. $\mathsf{DP}(\mathcal{R}, \mu)$ is:

$$\begin{array}{rcl} \mathtt{B} & \to & \mathtt{F}(\mathtt{c}(\mathtt{b})) \\ \mathtt{F}(x) & \to & x \end{array}$$

and $\mathcal{NHT}_{\mathsf{DP}(\mathcal{R},\mu)} = \{\mathtt{b}\}$. There is no infinite $(\mathcal{P}, \mathcal{R}, \mu^\sharp)$-chain for $\mathcal{P} = \mathsf{DP}(\mathcal{R}, \mu)$, i.e., $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$ is finite and $\mathcal{R}$ μ-terminating. However, with $\mathcal{P}'$ as in Theorem 7:

$$\begin{array}{rcl} \mathtt{B} & \to & \mathtt{F}(\mathtt{c}(\mathtt{b})) \\ \mathtt{F}(x) & \to & \mathtt{B} \end{array}$$

we have an infinite minimal $(\mathcal{P}', \mathcal{R}, \mu^\sharp)$-chain, i.e, $(\mathcal{P}', \mathcal{R}, \mu^\sharp)$ is not *finite.*

The following processor provides a sound and complete *transformation* of collapsing pairs into noncollapsing pairs.

Theorem 8 (Transforming collapsing pairs). *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F} \cup \mathcal{G}}$. Let $u \to x \in \mathcal{P}_{\mathcal{X}}$ and*

$$\begin{array}{rcl} P_u & = & \{u \to U(x)\} \\ & & \cup \, \{U(f(x_1, \ldots, x_k)) \to U(x_i) \mid f \in \mathcal{F}, i \in \mu(f)\} \\ & & \cup \, \{U(t) \to t^\sharp \mid t \in \mathcal{NHT}_{\mathcal{P}}\} \end{array}$$

where U is a fresh symbol. Let $\mathcal{P}' = (\mathcal{G} \cup \{U\}, P')$ where $P' = (P - \{u \to x\}) \cup P_u$, and μ' which extends μ by $\mu'(U) = \varnothing$. The processor Proc_{eColl} given by

$$\mathsf{Proc}_{eColl}(\mathcal{P}, \mathcal{R}, \mu) = \{(\mathcal{P}', \mathcal{R}, \mu')\}$$

is sound and complete.

Proof. With regard to soundness, we proceed by contradiction. If Proc_{eColl} is not sound, then there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain but there is no infinite minimal $(\mathcal{P}', \mathcal{R}, \mu')$-chain A. Since $\mathcal{P}$ is finite, we can assume that there is $\mathcal{Q} \subseteq \mathcal{P}$ such that A has a tail B

$$\sigma(u_1) \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^\sharp_\mu t_1 \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2) \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^\sharp_\mu t_2 \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

for some substitution σ and pairs $u_i \to v_i \in \mathcal{Q}$, and, for all $i \geq 1$,

1. if $v_i \notin \mathcal{X}$, then $t_i = \sigma(v_i)$, and

2. if $v_i = x_i \in \mathcal{X}$, then $x_i \notin \mathcal{V}ar^{\mu}(u_i)$, and $\sigma(x_i) = C_i[s_i]_{p_i}$ for some context $C_i[\,]_{p_i}$, such that $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$, $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$, $s_i = \theta_i(\bar{s}_i)$ for some $\bar{s}_i \in \mathcal{NHT}_{\mathcal{P}}$ and substitution θ_i, and $t_i = s_i^{\sharp}$.

For 'steps' $\sigma(u_i) \overset{\Lambda}{\hookrightarrow}_{Q,\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_i$ such that $u_i \to v_i \neq u \to x$, we have $u_i \to v_i \in \mathcal{P}'$. By minimality of B, t_i is $(\mathcal{R},\mu)$-terminating. Since $t_i \in \mathcal{T}(\mathcal{F} \cup \mathcal{G}, \mathcal{X})$ and $\mu'(f) = \mu(f)$ for all $f \in \mathcal{F} \cup \mathcal{G}$, t_i is $(\mathcal{R},\mu')$-terminating, too. On the other hand, if $u_i \to v_i = u \to x$, then, since $C_i[\,]_{p_i} \subseteq \mathcal{F}$, $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$, and by definition of P_u, we get

$$\sigma(u_i) \hookrightarrow_{P_u,\mu'} U(\sigma(v_i)) = U(C_i[s_i]_{p_i}) \hookrightarrow^{*}_{P_u,\mu'} U(s_i) = U(\theta_i(\bar{s}_i)) = \theta_i(U(\bar{s}_i)) \hookrightarrow_{P_u,\mu'} \theta_i(\bar{s}_i^{\sharp}) = s_i^{\sharp} = t_i$$

where all terms of the form $U(s)$ in the sequence above are $(\mathcal{R},\mu')$-terminating: since $\mu'(U) = \varnothing$ and U does not belong to $\mathcal{F}$, $U(s)$ is in $(\mathcal{R},\mu')$-normal form. Furthermore, by minimality of B, t_i is $(\mathcal{R},\mu)$-terminating and, since $\mu'(f) = \mu(f)$ for all $f \in \mathcal{F} \cup \mathcal{G}$, t_i is $(\mathcal{R},\mu')$-terminating. Therefore, we obtain an infinite minimal $(\mathcal{P}',\mathcal{R},\mu')$-chain, leading to a contradiction.

For completeness, we consider two cases: if $\mathcal{R}$ is not μ-terminating, then all termination problems are infinite (both before and after the application of Proc_{eColl}) and there is no problem. Therefore, assume that $\mathcal{R}$ is μ-terminating and that $(\mathcal{P},\mathcal{R},\mu)$ is finite but there is an infinite $(\mathcal{P}',\mathcal{R},\mu')$-chain. Again, we can assume that there is $Q \subseteq \mathcal{P}'$ such that A has a tail B

$$\sigma(u_1) \overset{\Lambda}{\hookrightarrow}_{Q,\mu'} \circ \trianglerighteq^{\sharp}_{\mu'} t_1 \hookrightarrow^{*}_{\mathcal{R},\mu'} \sigma(u_2) \overset{\Lambda}{\hookrightarrow}_{Q,\mu'} \circ \trianglerighteq^{\sharp}_{\mu'} t_2 \hookrightarrow^{*}_{\mathcal{R},\mu'} \cdots$$

for some substitution σ and pairs $u_i \to v_i \in Q$ where $t_i = \sigma(v_i)$ is $(\mathcal{R},\mu')$-terminating for $i \geq 1$. Without loss of generality, we can assume that $\sigma(x) \in \mathcal{T}(\mathcal{F} \cup \mathcal{G}, \mathcal{X})$ for all $x \in \mathcal{X}$, i.e., σ does *not* introduce any symbol U. It is not difficult to see that, for each $(\mathcal{P}',\mathcal{R},\mu')$-chain which is based on a substitution σ' whose bindings $\sigma'(x)$ contain symbols U, there is a $(\mathcal{P}',\mathcal{R},\mu')$-chain which uses the same pairs in $\mathcal{P}'$ and rules in $\mathcal{R}$ for the rewriting steps, but which is based on a substitution σ where the U's have been just removed from all bindings $\sigma'(x)$ to obtain $\sigma(x)$ instead.

If $u_i \to v_i \in P_u$, then, without loss of generality, we can assume that $u_i = u$ and $v_i = U(x)$. Since $\mu(U) = \varnothing$, there is $n \geq 0$ such that

$$\begin{array}{lll}
\sigma(u_i) & \overset{\Lambda}{\hookrightarrow}_{\mathcal{P}',\mu'} & \sigma(U(x)) = U(\sigma(x)) = \sigma(u_{i+1}) \\
 & \overset{\Lambda}{\hookrightarrow}_{\mathcal{P}',\mu'} & \sigma(v_{i+1}) = \sigma(u_{i+2}) \\
 & \overset{\Lambda}{\hookrightarrow}_{\mathcal{P}',\mu'} & \\
 & \vdots & \\
 & \overset{\Lambda}{\hookrightarrow}_{\mathcal{P}',\mu'} & \sigma(v_{i+n}) = \sigma(U(s_{i+n+1})) \\
 & \overset{\Lambda}{\hookrightarrow}_{\mathcal{P}',\mu'} & \sigma(s^{\sharp}_{i+n+1}) = \sigma(t_i) \\
 & \hookrightarrow^{*}_{\mathcal{R},\mu'} & \sigma(u_{i+n+2})
\end{array}$$

where, for all j, $i+1 \leq j \leq i+n$, $u_j = U(f_j(x_1,\ldots,x_{i_j},\ldots,x_{k_j}))$, $v_j = U(x_{i_j})$, $i_j \in \mu(f_j)$, and $s_{i+n+1} \in \mathcal{NHT}_{\mathcal{P}}$ (by definition of Proc_{eColl}). Therefore, from the n rewriting steps that remove the $f_j \in \mathcal{F}$ for $1 \leq j \leq n$, we know that $\sigma(x) = C_i[t_{i+n+1}]_{p_i}$ with $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$ and $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$. Thus, according to Definition 5, we have: $\sigma(u_i) \overset{\Lambda}{\hookrightarrow}_{\mathcal{P},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_i$ and $t_i \hookrightarrow^{*}_{\mathcal{R},\mu} \sigma(u_{i+n+2})$. Furthermore, t_i is μ-terminating (because $\mathcal{R}$ is μ-terminating). On the other hand, if $u_i \to v_i \in \mathcal{P} - \{u \to x\}$, then we have $\sigma(u_i) \overset{\Lambda}{\hookrightarrow}_{\mathcal{P},\mu} \circ \rhd^\sharp_\mu t_i$ satisfying the conditions in Definition 5. We obtain an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain, leading again to a contradiction. □

Example 17. *The use of* Proc_{eColl} *with* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mu^\sharp)$ *in Example 16 yields* $(\mathcal{P}', \mathcal{R}, \mu')$ *where* $\mathcal{P}'$ *consists of the following pairs:*

$$\begin{array}{rclcrcl} \mathtt{B} & \to & \mathtt{F(c(b))} & \quad & \mathtt{F}(x) & \to & \mathtt{U}(x) \\ \mathtt{U(b)} & \to & \mathtt{B} & & & & \end{array}$$

It is not difficult to see now that there is no infinite minimal $(\mathcal{P}', \mathcal{R}, \mu')$*-chain.*

10. Use of μ-reduction pairs

A reduction pair $(\gtrsim, \sqsupset)$ consists of a stable and monotonic quasi-ordering $\gtrsim$, and a stable and well-founded ordering $\sqsupset$ satisfying either $\gtrsim \circ \sqsupset \subseteq \sqsupset$ or $\sqsupset \circ \gtrsim \subseteq \sqsupset$ [KNT99]. The absence of infinite chains of pairs can be ensured by finding a *reduction pair* $(\gtrsim, \sqsupset)$ that is compatible with the rules and the pairs: $l \gtrsim r$ for all rewrite rules $l \to r$ and $u \gtrsim v$ or $u \sqsupset v$ for all dependency pairs $u \to v$ [AG00]. In the dependency pair framework they are used to obtain *smaller* sets of pairs $\mathcal{P}' \subseteq \mathcal{P}$ by removing the *strict* pairs, i.e., those pairs $u \to v \in \mathcal{P}$ such that $u \sqsupset v$.

Stability is required for both $\gtrsim$ and $\sqsupset$ because, although we only check the left- and right-hand sides of the rewrite rules $l \to r$ (with $\gtrsim$) and pairs $u \to v$ (with $\gtrsim$ or $\sqsupset$), the chains of pairs involve *instances* $\sigma(l)$, $\sigma(r)$, $\sigma(u)$, and $\sigma(v)$ of rules and pairs, and we aim to conclude $\sigma(l) \gtrsim \sigma(r)$ and also $\sigma(u) \gtrsim \sigma(v)$ or $\sigma(u) \sqsupset \sigma(v)$. Monotonicity is required for $\gtrsim$ to deal with the application of rules $l \to r$ to an arbitrary depth in terms. Since the pairs are 'applied' only at the root level, no monotonicity is required for $\sqsupset$ (but, for this reason, we cannot compare the rules in $\mathcal{R}$ using $\sqsupset$). Endrullis et al. noticed that *transitivity* is not necessary for the strict component $\sqsupset$ because it is somehow 'simulated' by the compatibility requirement above [EWZ08].

In our setting, since we are interested in μ-rewriting steps only, we can relax the monotonicity requirements as follows.

Definition 11 (μ-reduction pair). *Let* $\mathcal{F}$ *be a signature and* $\mu \in M_\mathcal{F}$*. A* μ-reduction pair $(\gtrsim, \sqsupset)$ *consists of a stable and* μ*-monotonic quasi-ordering* $\gtrsim$ *and a well-founded stable relation* $\sqsupset$ *on terms in* $\mathcal{T}(\mathcal{F}, \mathcal{X})$ *that are compatible, i.e.,* $\gtrsim \circ \sqsupset \subseteq \sqsupset$ *or* $\sqsupset \circ \gtrsim \subseteq \sqsupset$*. We say that* $(\gtrsim, \sqsupset)$ *is* μ*-monotonic if* $\sqsupset$ *is* μ*-monotonic.*

The following result allows us to use a μ*-monotonic* μ-reduction pair to remove some rewrite rules from the original rewrite system $\mathcal{R}$ before starting a termination proof.

Proposition 11 (Removing strict rewrite rules). *Let* $\mathcal{R}$ *be a TRS and* $\mu \in M_\mathcal{R}$*. Let* $(\gtrsim, \sqsupset)$ *be a* μ*-monotonic* μ*-reduction pair such that* $l\,(\gtrsim \cup \sqsupset)\,r$ *for all* $l \to r \in R$*. Let* $\mathcal{R}_\sqsupset = \{l \to r \in R \mid l \sqsupset r\}$ *and* $\mathcal{S} = \mathcal{R} - \mathcal{R}_\sqsupset$*. Then,* $\mathcal{R}$ *is* μ*-terminating if and only if* $\mathcal{S}$ *is* μ*-terminating.*

Proof. Since $\mathcal{S} \subseteq \mathcal{R}$, the *only if* part is obvious. For the *if* part, we proceed by contradiction. If $\mathcal{R}$ is not μ-terminating, then there is an infinite μ-rewrite sequence A:

$$t_1 \hookrightarrow_{\mathcal{R},\mu} t_2 \hookrightarrow_{\mathcal{R},\mu} \cdots t_n \hookrightarrow_{\mathcal{R},\mu} \cdots$$

where an infinite number of rules in $\mathcal{R}_\sqsupset$ have been used; otherwise, there would be an infinite tail $t_m \hookrightarrow_{\mathcal{S},\mu} t_{m+1} \hookrightarrow_{\mathcal{S},\mu} \cdots$ for some $m \geq 1$ where only rules in $\mathcal{S}$ are applied, contradicting the

μ-termination of $\mathcal{S}$. Let $J = \{j_1, j_2, \ldots\}$ be the infinite set of indices indicating μ-rewrite steps $t_j \hookrightarrow_{\mathcal{R},\mu} t_{j+1}$ in A, for all $j \in J$, where rules in $\mathcal{R}_\sqsupset$ have been used to perform the μ-rewriting step. Since $l \sqsupset r$ for all $l \to r \in \mathcal{R}_\sqsupset$, by stability and μ-monotonicity of $\sqsupset$, we have that $t_{j_i} \sqsupset t_{j_i+1}$. Since $l \gtrsim r$ for all $l \to r \in S$, by stability and μ-monotonicity of $\gtrsim$, we have that $t_{j_i+1} \gtrsim t_{j_{i+1}}$. By compatibility between $\gtrsim$ and $\sqsupset$, we have $t_{j_i} \sqsupset t_{j_{i+1}}$ for all $i \geq 1$. We obtain an infinite sequence $t_{j_1} \sqsupset t_{j_2} \sqsupset \cdots$ which contradicts well-foundedness of $\sqsupset$. □

10.1. Argument filterings for CSR

An argument filtering π for a signature $\mathcal{F}$ is a mapping that assigns to every k-ary function symbol $f \in \mathcal{F}$ an argument position $i \in \{1, \ldots, k\}$ or a (possibly empty) list $[i_1, \ldots, i_m]$ of argument positions with $1 \leq i_1 < \cdots < i_m \leq k$ [KNT99]. The *trivial* argument filtering $\pi_\top$ is given by $\pi_\top(f) = [1, \ldots, k]$ for each k-ary symbol $f \in \mathcal{F}$. It corresponds to the argument filtering which does nothing. In the dependency pair method, argument filterings π provide a simple way to *remove* parts of the syntactic structure of a rule $s \to t$. Argument filterings (recursively) drop immediate subterms of terms and can produce terms from a new signature where the arity of symbols has been decreased if necessary. In this way, we obtain simpler expressions that are (hopefully) easy to compare. In the following, we adapt the argument filtering technique to our CSDP framework. In Section 10.2, we investigate their use together with μ-reduction pairs. We can use an argument filtering π to 'filter' either the signature $\mathcal{F}$ or any replacement map $\mu \in M_\mathcal{F}$. In the following, we assume that:

1. The signature $\mathcal{F}_\pi$ consists of all function symbols f such that $\pi(f)$ is some list $[i_1, \ldots, i_m]$, where, in $\mathcal{F}_\pi$, the arity of f is m. As usual, we give the same name to the version of $f \in \mathcal{F}$ that belongs to $\mathcal{F}_\pi$.
2. The replacement map $\mu_\pi \in M_{\mathcal{F}_\pi}$ is given as follows: for all $f \in \mathcal{F}$ such that $f \in \mathcal{F}_\pi$ and $\pi(f) = [i_1, \ldots, i_m]$:
$$\mu_\pi(f) = \{j \in \{1, \ldots, m\} \mid i_j \in \mu(f)\}$$

An argument filtering π induces a mapping from $\mathcal{T}(\mathcal{F}, \mathcal{X})$ to $\mathcal{T}(\mathcal{F}_\pi, \mathcal{X})$, also denoted by π:

$$\pi(t) = \begin{cases} t & \text{if } t \text{ is a variable} \\ \pi(t_i) & \text{if } t = f(t_1, \ldots, t_k) \text{ and } \pi(f) = i \\ f(\pi(t_{i_1}), \ldots, \pi(t_{i_m})) & \text{if } t = f(t_1, \ldots, t_k) \text{ and } \pi(f) = [i_1, \ldots, i_m] \end{cases}$$

Note that, for the *trivial* argument filtering $\pi_\top$, we have that $\mathcal{F}_{\pi_\top} = \mathcal{F}$ and $\mu_{\pi_\top} = \mu$ for all $\mu \in M_\mathcal{F}$. Furthermore, $\pi_\top(t) = t$ for all $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. In the following, given a substitution σ and an argument filtering π, we let σ_π be the substitution defined by $\sigma_\pi(x) = \pi(\sigma(x))$ for all $x \in \mathcal{X}$. The following auxiliary results are used below.

Lemma 7. *Let $\mathcal{F}$ be a signature, π be an argument filtering for $\mathcal{F}$, and σ be a substitution. If $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, then $\pi(\sigma(t)) = \sigma_\pi(\pi(t))$.*

Proof. By structural induction.

1. Base case: t is a variable or a constant symbol. If $t = x \in \mathcal{X}$, then $\pi(x) = x$ and $\pi(\sigma(x)) = \sigma_\pi(x) = \sigma_\pi(\pi(x))$. If t is a constant symbol, then $\pi(t) = t$ and $\sigma(t) = t = \sigma_\pi(t)$. Hence, $\pi(\sigma(t)) = \pi(t) = t = \sigma_\pi(t) = \sigma_\pi(\pi(t))$.
2. If $t = f(t_1, \ldots, t_k)$, then we consider the two possible cases according to $\pi(f)$:

(a) If $\pi(f) = i$ for some $i \in \{1, \ldots, k\}$, then $\pi(t) = \pi(t_i)$. By the induction hypothesis, $\pi(\sigma(t_i)) = \sigma_\pi(\pi(t_i))$. Therefore, $\pi(\sigma(t)) = \pi(f(\sigma(t_1), \ldots, \sigma(t_k))) = \pi(\sigma(t_i)) = \sigma_\pi(\pi(t_i)) = \sigma_\pi(\pi(f(t_1 \ldots, t_k))) = \sigma_\pi(\pi(t))$.

(b) If $\pi(f) = [i_1, \ldots, i_m]$, then $\pi(t) = f(\pi(t_{i_1}), \ldots, \pi(t_{i_m}))$. By the induction hypothesis, $\pi(\sigma(t_{i_j})) = \sigma_\pi(\pi(t_{i_j}))$ for all $j \in \{1, \ldots, m\}$. Thus, $\pi(\sigma(t)) = \pi(f(\sigma(t_1), \ldots, \sigma(t_k))) = f(\pi(\sigma(t_{i_1})), \ldots, \pi(\sigma(t_{i_m}))) = f(\sigma_\pi(\pi(t_{i_1})), \ldots, \sigma_\pi(\pi(t_{i_m}))) = \sigma_\pi(f(\pi(t_{i_1}), \ldots, \pi(t_{i_m}))) = \sigma_\pi(\pi(t))$.

□

Proposition 12. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, π be an argument filtering for $\mathcal{F}$, and $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. Let $\gtrsim$ be a μ_π-monotonic quasi-ordering such that $\pi(l) \gtrsim \pi(r)$ for all $l \to r \in R$. If $s \hookrightarrow^* t$, then $\pi(s) \gtrsim \pi(t)$.*

Proof. By induction on the length n of the μ-rewrite sequence.

1. If $n = 0$, then $s = t$ and, trivially, $\pi(s) = \pi(t)$. Since $\gtrsim$ is reflexive, we have $\pi(s) \gtrsim \pi(t)$.
2. If $n > 0$, we can write $s \hookrightarrow s' \hookrightarrow^* t$, where the length of the sequence from s' to t is $n - 1$. Let $p \in \mathcal{P}os^\mu(s)$ be the μ-replacing position where the μ-rewriting step $s \hookrightarrow s'$ is performed. We prove that $s \hookrightarrow s'$ implies $\pi(s) \gtrsim \pi(s')$ by induction on the structure of p.
   (a) If $p = \Lambda$, then $s = \sigma(l)$ and $s' = \sigma(r)$ for some rewrite rule $l \to r$ and matching substitution σ. By Lemma 7, $\pi(s) = \pi(\sigma(l)) = \sigma_\pi(\pi(l))$ and $\pi(s') = \pi(\sigma(r)) = \sigma_\pi(\pi(r))$. Since $\pi(l) \gtrsim \pi(r)$, by stability of $\gtrsim$ we conclude $\pi(s) = \sigma_\pi(\pi(l)) \gtrsim \sigma_\pi(\pi(r)) = \pi(s')$.
   (b) If $p = i.q$, then we can write $s = f(s_1, \ldots, s_i, \ldots, s_k)$ and $s' = f(s'_1, \ldots, s'_i, \ldots, s'_k)$ for some nonconstant symbol f (i.e., $ar(f) > 0$) and we know that $i \in \mu(f)$, $s_i \hookrightarrow s'_i$ at position q, and $s_j = s'_j$ for all $j \neq i$. By the induction hypothesis, $\pi(s_i) \gtrsim \pi(s'_i)$. We consider the two possible cases according to $\pi(f)$:
      i. If $\pi(f) = j$ for some $j \in \{1, \ldots, k\}$, then $\pi(s) = \pi(s_j)$. If $i \neq j$, then $s'_j = s_j$. By reflexivity of $\gtrsim$, we have $\pi(s_j) \gtrsim \pi(s'_j)$. If $i = j$, then we know from above that $\pi(s_i) \gtrsim \pi(s'_i)$. Therefore, $\pi(s) = \pi(s_j) \gtrsim \pi(s'_j) = \pi(s')$.
      ii. If $\pi(f) = [i_1, \ldots, i_m]$, then we have that $\pi(s) = f(\pi(s_{i_1}), \ldots, \pi(s_{i_m}))$ and $\pi(s') = f(\pi(s'_{i_1}), \ldots, \pi(s'_{i_m}))$. Consider i_j for some $j \in \{1, \ldots, m\}$. We have two cases:
         A. If $i_j = i$, then by the induction hypothesis, $\pi(s_{i_j}) \gtrsim \pi(s'_{i_j})$ and, by definition of μ_π, $j \in \mu_\pi(f)$.
         B. If $i_j \neq i$, then $s'_{i_j} = s_{i_j}$ and we have $\pi(s_{i_j}) = \pi(s'_{i_j})$.

      Note that $\pi(s_{i_j})$ is the j-th immediate subterm of $\pi(s)$. By μ_π-monotonicity of $\gtrsim$,

$$\begin{array}{rcl}
\pi(s) & = & \pi(f(s_1, \ldots, s_k)) \\
 & = & f(\pi(s_{i_1}), \ldots, \pi(s_{i_j}), \ldots, \pi(s_{i_m})) \\
 & \gtrsim & f(\pi(s_{i_1}), \ldots, \pi(s'_{i_j}), \ldots, \pi(s_{i_m})) \\
 & = & f(\pi(s'_{i_1}), \ldots, \pi(s'_{i_j}), \ldots, \pi(s'_{i_m})) \\
 & = & \pi(f(s'_1, \ldots, s'_k)) \\
 & = & \pi(s')
\end{array}$$

      where we assume that $i_j = i$ for some $j \in \{1, \ldots, k\}$. If no such j exists, then we would have $\pi(s) = \pi(s')$, which also implies $\pi(s) \gtrsim (s')$ because $\gtrsim$ is reflexive.

   Thus, we have proved that $s \hookrightarrow s'$ implies $\pi(s) \gtrsim \pi(s')$ as desired.

Therefore, $\pi(s) \gtrsim \pi(s')$ and, by the induction hypothesis, $\pi(s') \gtrsim \pi(t)$. By transitivity of $\gtrsim$, we conclude $\pi(s) \gtrsim \pi(t)$.

□

Remark 12. *We often use argument filterings to transform (sets of) rules S as follows: $\pi(s \to t) = \pi(s) \to \pi(t)$ for a rule $s \to t$, and $\pi(S) = \{\pi(s \to t) \mid s \to t \in S\}$. Given a TRS $\mathcal{R} = (\mathcal{F}, R)$, we write $\pi(\mathcal{R})$ to denote the filtered TRS $(\mathcal{F}_\pi, \pi(R))$.*

10.2. Removing pairs using μ-reduction pairs

Given TRSs $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$, and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$, checking the absence of infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chains can often be ‘simplified’ to checking the absence of infinite minimal $(\mathcal{P}', \mathcal{R}, \mu)$-chains for a proper subTRS $\mathcal{P}'$ of $\mathcal{P}$ by finding appropriate *μ-reduction pairs* $(\gtrsim, \sqsupset)$. The presence of *collapsing* pairs $u \to v = u \to x \in \mathcal{P}_\mathcal{X}$ imposes some additional requirements on the μ-reduction pairs:

1. We need to ensure that the quasi-ordering $\gtrsim$ is able to ‘look’ for a μ-replacing subterm s inside the instantiation $\sigma(x)$ of a migrating variable x: since $\sigma(x) = C[s]_p$ for some context $C[\,]_p$ and μ-replacing position $p \in \mathcal{P}os^\mu(C[\,]_p)$ such that $\mathit{sprefix}_{C[\,]_p}(p) \subseteq \mathcal{F}$, we can obtain s out from $C[s]_p$ by applying the projection rules in $\mathcal{E}mb^\mu(\mathcal{F})$ (Definition 1). Hence, we require $\mathcal{E}mb^\mu(\mathcal{F}) \subseteq \gtrsim$.
2. We need to connect the marked version $s^\sharp$ of s (which is an instance of a hidden term $t \in \mathcal{NHT}_\mathcal{P}$, i.e., $s = \theta(t)$ for some substitution θ) with an instance $\sigma(u)$ of the left-hand side u of a pair; hence, we require $t \gtrsim t^\sharp$ or $t \sqsupset t^\sharp$ for all $t \in \mathcal{NHT}_\mathcal{P}$ which, by stability, becomes $s \gtrsim s^\sharp$ or $s \sqsupset s^\sharp$.

The following theorem formalizes a generic processor to remove pairs from $\mathcal{P}$ by using argument filterings and μ-reduction pairs.

Theorem 9 (μ-reduction pair processor). *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. Let π be an argument filtering for $\mathcal{F} \cup \mathcal{G}$ and $(\gtrsim, \sqsupset)$ be a μ_π-reduction pair such that*

1. $\pi(\mathcal{R}) \subseteq \gtrsim$, $\pi(\mathcal{P}) \subseteq \gtrsim \cup \sqsupset$, *and*
2. *whenever $\mathcal{NHT}_\mathcal{P} \neq \varnothing$ and $\mathcal{P}_\mathcal{X} \neq \varnothing$, we have that*
   (a) *for all $f \in \mathcal{F}$, either $\pi(f) = [i_1, \ldots, i_m]$ and $\mu(f) \subseteq \pi(f)$, or $\pi(f) = i$ and $\mu(f) = \{i\}$,*
   (b) *$\mathcal{E}mb^{\mu_\pi}(\mathcal{F}_\pi) \subseteq \gtrsim$, and*
   (c) *$\pi(t)\ (\gtrsim \cup \sqsupset)\ \pi(t^\sharp)$ for all $t \in \mathcal{NHT}_\mathcal{P}$,*

Let $\mathcal{P}_\sqsupset = \{u \to v \in \mathcal{P} \mid \pi(u) \sqsupset \pi(v)\}$. Then, the processor Proc_{RP} *given by*

$$\mathsf{Proc}_{RP}(\mathcal{P}, \mathcal{R}, \mu) = \begin{cases} \{(\mathcal{P} - \mathcal{P}_\sqsupset, \mathcal{R}, \mu)\} & \textit{if (1) and (2) hold} \\ \{(\mathcal{P}, \mathcal{R}, \mu)\} & \textit{otherwise} \end{cases}$$

is sound and complete.

Proof. Completeness is obvious, since $\mathcal{P} - \mathcal{P}_\sqsupset \subseteq \mathcal{P}$. Regarding soundness, we proceed by contradiction. Assume that there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain A, but that there is no infinite minimal $(\mathcal{P} - \mathcal{P}_\sqsupset, \mathcal{R}, \mu)$-chain. Due to the finiteness of $\mathcal{P}$, we assume that there is $Q \subseteq \mathcal{P}$ such that A has a tail B

$$\sigma(u_1) \hookrightarrow_{Q,\mu} \circ \trianglerighteq_\mu^\sharp t_1 \hookrightarrow_{\mathcal{R},\mu}^* \sigma(u_2) \hookrightarrow_{Q,\mu} \circ \trianglerighteq_\mu^\sharp t_2 \hookrightarrow_{\mathcal{R},\mu}^* \sigma(u_3) \hookrightarrow_{Q,\mu} \circ \trianglerighteq_\mu^\sharp \cdots$$

for some substitution σ, where all pairs in $\mathcal{Q}$ are infinitely often used. Also, for all $i \geq 1$, (1) if $u_i \to v_i \in \mathcal{Q}_{\mathcal{G}}$, then $t_i = \sigma(v_i)$ and (2) if $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_{\mathcal{X}}$, then $t_i = s_i^{\sharp}$ for some s_i such that $\sigma(x_i) = C_i[s_i]_{p_i}$ for some $C_i[\ ]_{p_i}$ and $p_i \in \mathcal{P}os^{\mu}(C_i[\ ]_{p_i})$ such that $\mathit{sprefix}_{C_i[\ ]_{p_i}}(p_i) \subseteq \mathcal{F}$ and $s_i = \theta_i(\bar{s}_i)$ for some $\bar{s}_i \in \mathcal{NHT}$ and substitution θ_i. Actually, since $t_i = s_i^{\sharp} = \theta_i(\bar{s}_i)^{\sharp} = \theta_i(\bar{s}_i^{\sharp})$ and $t_i \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$, we can further say that $\bar{s}_i \in \mathcal{NHT}_{\mathcal{Q}}$.

Since $\pi(u_i)\ (\gtrsim \cup \sqsupset)\ \pi(v_i)$ for all $u_i \to v_i \in \mathcal{Q} \subseteq \mathcal{P}$, by stability of $\gtrsim$ and $\sqsupset$, we have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(v_i))$ for all $i \geq 1$.

No pair $u \to v \in \mathcal{Q}$ satisfies that $\pi(u) \sqsupset \pi(v)$. Otherwise, we get a contradiction by considering the following two cases:

1. If $u_i \to v_i \in \mathcal{Q}_{\mathcal{G}}$, then $t_i = \sigma(v_i) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$ and by Proposition 12, $\pi(t_i) \gtrsim \pi(\sigma(u_{i+1}))$. By Lemma 7, $\pi(t_i) \gtrsim \sigma_\pi(\pi(u_{i+1}))$. Since we have $\sigma_\pi(\pi(u_i))(\gtrsim \cup \sqsupset)\sigma_\pi(\pi(v_i)) = \pi(\sigma(v_i)) = \pi(t_i)$ (using Lemma 7), by using transitivity of $\gtrsim$ and compatibility between $\gtrsim$ and $\sqsupset$, we conclude that $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(u_{i+1}))$.
2. If $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_{\mathcal{X}}$, then $\sigma(v_i) = \sigma(x_i) = C_i[s_i]_{p_i}$. Since $i \in \mu(f)$ implies that $i \in \pi(f)$, we can say that $\pi(\sigma(x)) = \sigma_\pi(x) = \pi(C_i)[\pi(s_i)]_{q_i}$ for some $q_i \in \mathcal{P}os^{\mu_\pi}(\pi(C_i))$ and $\mathit{sprefix}_{\pi(C_i)}(q_i) \subseteq \mathcal{F}_\pi$. Since $\mathcal{E}mb^{\mu_\pi}(\mathcal{F}_\pi) \subseteq \gtrsim$, we have $\sigma_\pi(\pi(v_i)) = \sigma_\pi(x_i) \gtrsim \pi(s_i)$. Furthermore, we are assuming that $\pi(t)\ (\gtrsim \cup \sqsupset)\ \pi(t^{\sharp})$ for all $t \in \mathcal{NHT}_{\mathcal{Q}} \subseteq \mathcal{NHT}_{\mathcal{P}}$. Since $s_i = \theta_i(\bar{s}_i)$, we have that $\pi(s_i) = \pi(\theta_i(\bar{s}_i)) = \theta_{i,\pi}(\pi(\bar{s}_i))$ (using Lemma 7 again) and, similarly, $\pi(s_i^{\sharp}) = \theta_{i,\pi}(\pi(\bar{s}_i^{\sharp}))$. By stability we have that $\pi(s_i)\ (\gtrsim \cup \sqsupset)\ \pi(s_i^{\sharp})$. Hence, by transitivity of $\gtrsim$ (and compatibility of $\gtrsim$ and $\sqsupset$), we have $\sigma_\pi(\pi(v_i)) = \sigma_\pi(x_i)\ (\gtrsim \cup \sqsupset)\ \pi(s_i^{\sharp})$. Finally, since $\pi(s_i^{\sharp}) = \pi(t_i)$ and $t_i \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$ for all $i \geq 1$, by Proposition 12 and Lemma 7, $\pi(t_i) \gtrsim \sigma_\pi(\pi(u_{i+1}))$. Therefore, again by transitivity of $\gtrsim$ and compatibility of $\gtrsim$ and $\sqsupset$, we conclude that $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(u_{i+1}))$.

Since $u \to v$ occurs infinitely often in B, there is an infinite set $\mathcal{I} \subseteq \mathbb{N}$ such that $\sigma_\pi(\pi(u_i)) \sqsupset \sigma_\pi(\pi(u_{i+1}))$ for all $i \in \mathcal{I}$. And we have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(u_{i+1}))$ for all other $u_i \to v_i \in \mathcal{Q}$. Thus, by using the compatibility conditions of the μ_π-reduction pair, we obtain an infinite decreasing $\sqsupset$-sequence that contradicts the well-foundedness of $\sqsupset$.

Therefore, $\mathcal{Q} \subseteq \mathcal{P} - \mathcal{P}_{\sqsupset}$, which means that B is an infinite minimal $(\mathcal{P} - \mathcal{P}_{\sqsupset}, \mathcal{R}, \mu)$-chain, thus leading to a contradiction. □

Example 18. *Consider the TRS $\mathcal{R}$ [Zan97, Example 5]:*

$$\begin{array}{rcl@{\qquad\qquad}rcl}
\mathtt{if}(\mathtt{true}, x, y) & \to & x & \mathtt{f}(x) & \to & \mathtt{if}(x, \mathtt{c}, \mathtt{f}(\mathtt{true})) \\
\mathtt{if}(\mathtt{false}, x, y) & \to & y & & &
\end{array}$$

with $\mu(\mathtt{f}) = \{1\}$ and $\mu(\mathtt{if}) = \{1, 2\}$. Then, $\mathsf{DP}(\mathcal{R}, \mu)$ *consists of the following CSDPs:*

$$\begin{array}{rcl@{\qquad\qquad}rcl}
\mathtt{F}(x) & \to & \mathtt{IF}(x, \mathtt{c}, \mathtt{f}(\mathtt{true})) & \mathtt{IF}(\mathtt{false}, x, y) & \to & y
\end{array}$$

with $\mu^{\sharp}(\mathtt{F}) = \{1\}$ and $\mu^{\sharp}(\mathtt{IF}) = \{1, 2\}$. The μ-reduction pair $(\geq, >)$ induced by the polynomial interpretation[6]

$$\begin{array}{rcl@{\qquad}rcl@{\qquad}rcl}
[\mathtt{c}] = [\mathtt{true}] & = & 0 & [\mathtt{f}](x) & = & x & [\mathtt{F}](x) & = & x \\
[\mathtt{false}] & = & 1 & [\mathtt{if}](x, y, z) & = & x + y + z & [\mathtt{IF}](x, y, z) & = & x + z
\end{array}$$

---

[6] See [Luc05] for more information about the automatic generation of polynomial (quasi-)orderings with monotonicity requirements specified by means of replacement maps.

can be used to prove the μ-termination of $\mathcal{R}$. For $\mathcal{P} = \mathsf{DP}(\mathcal{R},\mu)$, we have $\mathcal{NHT}_{\mathcal{P}} = \{\mathtt{f(true)}\}$. First, we can see that the quasi-ordering is compatible with the rules in $\mathcal{E}mb^{\mu}(\mathcal{F})$:

$$\begin{array}{rcccccl}
[\mathtt{f}(x)] & = & x & \geq & x & = & [x] \\
[\mathtt{if}(x,y,z)] & = & x+y+z & \geq & x & = & [x] \\
[\mathtt{if}(x,y,z)] & = & x+y+z & \geq & y & = & [y]
\end{array}$$

Now we can see that the condition on the only hidden term in $\mathcal{NHT}_{\mathcal{P}}$ is also fulfilled:

$$[\mathtt{f(true)}] \;=\; 0 \;\geq\; 0 \;=\; [\mathtt{F(true)}]$$

Finally, for the three rules in $\mathcal{R}$ and the two pairs in $\mathcal{P}$, we have:

$$\begin{array}{rcccccl}
[\mathtt{f}(x)] & = & x & \geq & x & = & [\mathtt{if}(x,\mathtt{c},\mathtt{f(true)})] \\
[\mathtt{if}(\mathtt{true},x,y)] & = & x+y & \geq & x & = & [x] \\
[\mathtt{if}(\mathtt{false},x,y)] & = & x+y+1 & \geq & y & = & [y] \\
[\mathtt{F}(x)] & = & x & \geq & x & = & [\mathtt{IF}(x,\mathtt{c},\mathtt{f(true)})] \\
[\mathtt{IF}(\mathtt{false},x,y)] & = & y+1 & > & y & = & [y]
\end{array}$$

We remove the 'strict' pair $\mathtt{IF}(\mathtt{false},x,y) \to y$ from $\mathcal{P}$ to obtain $\mathcal{P}'$. With $(\mathcal{P}',\mathcal{R},\mu^{\sharp})$, the application of Proc_{SCC} leads to an empty set of CS problems. Thus, the μ-termination of $\mathcal{R}$ is proved.

The 'compatibility' between the replacement map μ and the argument filtering π, which is required when collapsing pairs are present, is necessary in Theorem 9.

Example 19. *Consider the following TRS $\mathcal{R}$:*

$$\begin{array}{rcl}
\mathtt{a} & \to & \mathtt{c(h(f(a),b))} \\
\mathtt{f}(\mathtt{c}(x)) & \to & x
\end{array}$$

together with the replacement map μ given by $\mu(\mathtt{f}) = \mu(\mathtt{h}) = \{1\}$ and $\mu(\mathtt{c}) = \varnothing$. $\mathsf{DP}(\mathcal{R},\mu)$ is:

$$\mathtt{F}(\mathtt{c}(x)) \;\to\; x$$

and $\mathcal{NHT}_{\mathsf{DP}(R,\mu)} = \{\mathtt{f(a)}\}$. Note that $\mathcal{R}$ is not *μ-terminating:*

$$\mathtt{f}(\underline{\mathtt{a}}) \hookrightarrow \underline{\mathtt{f(c(h(f(a),b)))}} \hookrightarrow \mathtt{h}(\mathtt{f}(\underline{\mathtt{a}}),\mathtt{b}) \hookrightarrow \cdots$$

For the argument filtering π given by $\pi(\mathtt{a}) = \pi(\mathtt{h}) = []$, $\pi(\mathtt{F}) = \pi(\mathtt{f}) = [1]$ and $\pi(\mathtt{c}) = 1$, $\mathcal{F}_{\pi}$ consists of the constants $\mathtt{a},\mathtt{h}$ and symbols $\mathtt{f},\mathtt{F}$ of arity 1. Also, $\mu^{\sharp}_{\pi}(\mathtt{f}) = \mu^{\sharp}_{\pi}(\mathtt{F}) = \{1\}$ and $\mu^{\sharp}_{\pi}(\mathtt{a}) = \mu^{\sharp}_{\pi}(\mathtt{h}) = \varnothing$. We get the constraints:

$$\begin{array}{rcccccl}
\pi(\mathtt{a}) & = & \mathtt{a} & \gtrsim & \mathtt{h} & = & \pi(\mathtt{c(h(f(a),b))}) \\
\pi(\mathtt{f}(\mathtt{c}(x))) & = & \mathtt{f}(x) & \gtrsim & x & = & \pi(x) \\
 & & \mathtt{f}(x) & \gtrsim & x & & \\
 & & \mathtt{F}(x) & \gtrsim & x & & \\
\pi(\mathtt{f(a)}) & = & \mathtt{f(a)} & \gtrsim & \mathtt{F(a)} & = & \pi(\mathtt{F(a)}) \\
\pi(\mathtt{F}(\mathtt{c}(x))) & = & \mathtt{F}(x) & \sqsupset & x & = & \pi(x)
\end{array}$$

which are easily satisfiable (by a polynomial interpretation, for instance). We would wrongly conclude μ-termination of $\mathcal{R}$. Note that $\pi(\mathtt{c}) = 1$ but $\mu^{\sharp}(\mathtt{c}) = \varnothing$, and that $\pi(\mathtt{h}) = []$ but $\mu^{\sharp}(\mathtt{h}) = \{1\}$.

The next processor is useful when all (filterings of) terms in $\mathcal{NHT}_{\mathcal{P}}$ are *ground*. The advantage is that the quasi-ordering $\gtrsim$ of the μ-reduction pair does not need to impose compatibility with the rules in $\mathcal{E}mb^{\mu}(\mathcal{F})$.

Theorem 10 (μ-reduction pair processor for ground hidden terms). *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. Let π be an argument filtering for $\mathcal{F} \cup \mathcal{G}$ such that, for all $t \in \mathcal{NHT}_{\mathcal{P}}$, $\pi(t)$ is* ground. *Let $(\gtrsim, \sqsupset)$ be a μ_π-reduction pair such that*

1. $\pi(\mathcal{R}) \subseteq \gtrsim$, $\pi(\mathcal{P}_{\mathcal{G}}) \subseteq \gtrsim \cup \sqsupset$, *and*
2. *for all $u \to v \in \mathcal{P}_{\mathcal{X}}$ and all $t \in \mathcal{NHT}_{\mathcal{P}}$, $\pi(u)\ (\gtrsim \cup \sqsupset)\ \pi(t^\sharp)$*

Let $\mathcal{P}_{\sqsupset} = \{u \to v \in \mathcal{P}_{\mathcal{G}} \mid \pi(u) \sqsupset \pi(v)\} \cup \{u \to v \in \mathcal{P}_{\mathcal{X}} \mid \forall t \in \mathcal{NHT}_{\mathcal{P}}, \pi(u) \sqsupset \pi(t^\sharp)\}$. Then, the processor Proc_{RPg} given by

$$\mathsf{Proc}_{RPg}(\mathcal{P},\mathcal{R},\mu) = \begin{cases} \{(\mathcal{P} - \mathcal{P}_{\sqsupset}, \mathcal{R}, \mu)\} & \textit{if (1) and (2) hold} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{cases}$$

is sound and complete.

Proof. The proof is analogous to that of Theorem 9. Assume the facts and notation in the first paragraph of such a proof. Again, we proceed by contradiction and assume that a pair $u \to v \in \mathcal{Q}$ is in $\mathcal{P}_{\sqsupset}$. Again, we have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(u_{i+1}))$ for all pairs $u_i \to v_i \in \mathcal{Q}_{\mathcal{G}}$.

Now, if $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_{\mathcal{X}}$, then since $\pi(u_i)(\gtrsim \cup \sqsupset)\pi(t^\sharp)$ for all $t \in \mathcal{NHT}_{\mathcal{Q}} \subseteq \mathcal{NHT}_{\mathcal{P}}$, by stability we have that $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(t^\sharp))$. Since $\pi(t)$ is ground, we have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \pi(t^\sharp)$. Therefore, since $s_i \in \mathcal{NHT}_{\mathcal{Q}}$ and $t_i = s_i^\sharp$, we have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \pi(t_i)$. Finally, since $s_i^\sharp = t_i$ and $t_i \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$ for all $i \geq 1$, by Proposition 12 and Lemma 7, we have that $\pi(t_i) \gtrsim \sigma_\pi(\pi(u_{i+1}))$. Thus, we also have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(\pi(u_{i+1}))$.

Since $u \to v$ occurs infinitely often in B, by using the compatibility conditions of the μ_π-reduction pair, we obtain an infinite decreasing $\sqsupset$-sequence that contradicts well-foundedness of $\sqsupset$. In particular, if $u \to v \in \mathcal{Q}_{\mathcal{X}} \cap \mathcal{P}_{\sqsupset}$, then $\pi(u) \sqsupset \pi(t^\sharp)$ for all $t \in \mathcal{NHT}_{\mathcal{Q}}$, so each time that $u \to v$ is used, a strict decrease occurs. □

Theorem 10 can succeed when Theorem 9 fails.

Example 20. *Consider the TRS $\mathcal{R}$:*

$$\mathtt{a} \to \mathtt{f}(\mathtt{d}(\mathtt{c}(\mathtt{a}))) \tag{27}$$

$$\mathtt{f}(\mathtt{c}(x)) \to x \tag{28}$$

$$\mathtt{d}(\mathtt{c}(x)) \to \mathtt{b} \tag{29}$$

and the replacement map μ given by $\mu(\mathtt{c}) = \varnothing$ and $\mu(\mathtt{f}) = \mu(\mathtt{d}) = \{1\}$. There are three CSDPs:

$$\mathtt{A} \to \mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a}))) \tag{30}$$

$$\mathtt{A} \to \mathtt{D}(\mathtt{c}(\mathtt{a})) \tag{31}$$

$$\mathtt{F}(\mathtt{c}(x)) \to x \tag{32}$$

$\mathsf{Proc}_{SCC}(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mu^\sharp)$ *yields a single CS problem $(\mathcal{P},\mathcal{R},\mu)$ with $\mathcal{P} = \{(30),(32)\}$. Since $\mathcal{NHT}_{\mathcal{P}} = \{\mathtt{a}\} \neq \varnothing$ and $\mathtt{F}(\mathtt{c}(x)) \to x$ is a collapsing CSDP, according to Theorem 9 we would require that any μ-reduction ordering used in the theorem satisfy $\mathcal{E}mb^{\mu}(\mathcal{F}) \subseteq \gtrsim$ (assume the trivial filtering $\pi_\top$ here) and that $\mathtt{a}\ (\gtrsim \cup \sqsupset)\ \mathtt{A}$. In this case, though, since $\mathtt{d}(\mathtt{c}(\mathtt{a})) \trianglerighteq_\mu \mathtt{c}(\mathtt{a})$, we must have $\mathtt{d}(\mathtt{c}(\mathtt{a})) \gtrsim \mathtt{c}(\mathtt{a})$; by μ-monotonicity of $\gtrsim$, $\mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a}))) \gtrsim \mathtt{F}(\mathtt{c}(\mathtt{a}))$. Now, one of the following two cases must hold:*

1. $\mathtt{A} \sqsupset \mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a})))$ *and* $\mathtt{F}(\mathtt{c}(x))\,(\gtrsim \cup \sqsupset)\,x$. *By stability of* $\gtrsim$ *and* $\sqsupset$, *we have* $\mathtt{F}(\mathtt{c}(\mathtt{a}))\,(\gtrsim \cup \sqsupset)\,\mathtt{a}$. *Thus,*

   $$\mathtt{A} \sqsupset \mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a}))) \gtrsim \mathtt{F}(\mathtt{c}(\mathtt{a}))\,(\gtrsim \cup \sqsupset)\,\mathtt{a}\,(\gtrsim \cup \sqsupset)\,\mathtt{A}.$$

   By compatibility of $\gtrsim$ *and* $\sqsupset$, *we have* $\mathtt{A} \sqsupset \cdots \sqsupset \mathtt{A}$, *contradicting the well-foundedness of* $\sqsupset$.
2. $\mathtt{A}\,(\gtrsim \cup \sqsupset)\,\mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a})))$ *and* $\mathtt{F}(\mathtt{c}(x)) \sqsupset x$. *Hence,*

   $$\mathtt{A}\,(\gtrsim \cup \sqsupset)\,\mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a}))) \gtrsim \mathtt{F}(\mathtt{c}(\mathtt{a})) \sqsupset \mathtt{a}\,(\gtrsim \cup \sqsupset)\,\mathtt{A}.$$

   Again, by compatibility of $\gtrsim$ *and* $\sqsupset$, *we have* $\mathtt{A} \sqsupset \cdots \sqsupset \mathtt{A}$.

Thus, Theorem 9 cannot be used with this example. Since $\mathcal{NHT}_{\mathcal{P}} \subseteq \mathcal{T}(\mathcal{F})$, *Theorem 10 is applicable here. The* μ*-reduction pair* $(\geq, >)$ *induced by the following polynomial interpretation*[7]*:*

$$\begin{array}{rclrclrclrcl}
[\mathtt{a}] & = & 1 & [\mathtt{b}] & = & 0 & [\mathtt{c}](x) & = & x+1 & [\mathtt{d}](x) & = & \frac{1}{4}x \\
[\mathtt{f}](x) & = & x & [\mathtt{A}] & = & 1 & [\mathtt{F}](x) & = & 0 & & &
\end{array}$$

can be used to remove (30) *from* $\mathcal{P}$. *For the three rules in* $\mathcal{R}$ *and the two pairs in* $\mathcal{P}$, *we have:*

$$\begin{array}{rcccccl}
[\mathtt{a}] & = & 1 & \geq & \frac{1}{2} & = & [\mathtt{f}(\mathtt{d}(\mathtt{c}(\mathtt{a})))] \\
[\mathtt{f}(\mathtt{c}(x))] & = & x+1 & \geq & x & = & [x] \\
[\mathtt{d}(\mathtt{c}(x))] & = & \frac{1}{4}x+\frac{1}{4} & \geq & 0 & = & [\mathtt{b}] \\
[\mathtt{A}] & = & 1 & > & \frac{1}{2} & = & [\mathtt{F}(\mathtt{d}(\mathtt{c}(\mathtt{a})))] \\
[\mathtt{F}(\mathtt{c}(x))] & = & x+1 & \geq & 1 & = & [\mathtt{A}]
\end{array}$$

We remove (30) *from* $\mathcal{P}$ *to obtain* $\mathcal{P}' = \{(32)\}$. *Now,* $\mathsf{Proc}_{SCC}(\mathcal{P}', \mathcal{R}, \mu) = \varnothing$ *because* $\mathcal{NHT}_{\mathcal{P}'} = \varnothing$ *and* $\mathsf{EG}(\mathcal{P}', \mathcal{R}, \mu)$ *contains no cycle. Thus, the* μ*-termination of* $\mathcal{R}$ *is proved.*

Nevertheless, even with $\mathcal{NHT}_{\mathcal{P}} \subseteq \mathcal{T}(\mathcal{F})$, Theorem 9 can be helpful when Theorem 10 fails.

Example 21. *Consider* $\mathcal{R}$ *and* μ *as in Example 16. Theorem 10 cannot be used here because, reasoning as in Example 16, we would obtain constraints that are incompatible with the well-foundedness of* $\sqsupset$ *for any strict component* $\sqsupset$ *of a* μ*-reduction pair* $(\gtrsim, \sqsupset)$. *However, the* μ*-termination of* $\mathcal{R}$ *can be easily proved with Theorem 9. The* μ*-reduction pair* $(\geq, >)$ *generated by the following polynomial interpretation:*

$$\begin{array}{rclrclrcl}
[\mathtt{b}] & = & 1 & [\mathtt{c}](x) & = & 0 & [\mathtt{f}](x) & = & x \\
[\mathtt{B}] & = & 2 & [\mathtt{F}](x) & = & x+1 & & &
\end{array}$$

satisfies the requirements of Theorem 10 and can be used to show a weak decrease of the rules and a strict decrease of the two CSDPs which can both be removed.

Our last result establishes that if we are able to provide a strict comparison between unmarked and marked versions of the (filtered) hidden terms in $\mathcal{NHT}_{\mathcal{P}}$, then we can remove *all* collapsing pairs at the same time.

---

[7] See [Luc05, Luc07] for details about the use of polynomial intepretations with rational coefficients.

Theorem 11 (μ-reduction pair processor for collapsing pairs). *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. Let π be an argument filtering for $\mathcal{F} \cup \mathcal{G}$ and $(\gtrsim, \sqsupset)$ be a μ_π-reduction pair such that*

1. $\pi(\mathcal{R}) \subseteq \gtrsim$, $\pi(\mathcal{P}) \subseteq \gtrsim \cup \sqsupset$, *and*
2. $\pi(t) \sqsupset \pi(t^\sharp)$ *for all* $t \in \mathcal{NHT}_\mathcal{P}$ *and*
   - (a) *for all $f \in \mathcal{F}$, either $\pi(f) = [i_1, \ldots, i_m]$ and $\mu(f) \subseteq \pi(f)$, or $\pi(f) = i$ and $\mu(f) = \{i\}$,*
   - (b) $\mathcal{E}mb^{\mu_\pi}(\mathcal{F}_\pi) \subseteq \gtrsim$.

Then, the processor Proc_{RPc} *given by*

$$\mathsf{Proc}_{RPc}(\mathcal{P}, \mathcal{R}, \mu) = \begin{cases} \{(\mathcal{P}_\mathcal{G}, \mathcal{R}, \mu)\} & \textit{if (1) and (2) hold} \\ \{(\mathcal{P}, \mathcal{R}, \mu)\} & \textit{otherwise} \end{cases}$$

is sound and complete.

Proof. As in the proof of Theorem 9, we proceed by contradiction. We assume that there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain A, but that there is no infinite minimal $(\mathcal{P}_\mathcal{G}, \mathcal{R}, \mu)$-chain. Thus, there is $\mathcal{Q} \subseteq \mathcal{P}$ such that $\mathcal{Q} \cap \mathcal{P}_\mathcal{X} \neq \varnothing$ and A has a tail B as in the proof of Theorem 9. Now, we assume the notation as in the first paragraph of such a proof.

We have $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \pi(t_i)$ and $\pi(t_i) \gtrsim \sigma_\pi(\pi(u_{i+1}))$ for all pairs $u_i \to v_i \in \mathcal{P}_\mathcal{G}$. If $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_\mathcal{X}$, then by applying the considerations in the corresponding item of the proof of Theorem 9 and taking into account that $\pi(t) \sqsupset \pi(t^\sharp)$ for *all* $t \in \mathcal{NHT}_\mathcal{P}$, we now have that $\sigma_\pi(\pi(u_i))\ (\gtrsim \cup \sqsupset)\ \sigma_\pi(x_i) \sqsupset \pi(t_i) \gtrsim \sigma_\pi(\pi(u_{i+1}))$. Since pairs $u_i \to v_i \in \mathcal{Q}_\mathcal{X}$ occur infinitely often in B, by using the compatibility conditions of the μ_π-reduction pair, we obtain an infinite decreasing $\sqsupset$-sequence that contradicts the well-foundedness of $\sqsupset$. □

11. Subterm criterion

In [HM04], Hirokawa and Middeldorp introduce a *subterm criterion* that permits certain cycles of the dependency graph to be ignored *without paying attention to the rules of the TRS*. Their result applies to *cycles* in the *dependency graph*. Thiemann has adapted it to the DP-framework [Thi07, Section 4.6]. In our adaptation to *CSR*, we take ideas from both works. Our first definition is inspired by Thiemann's *head symbols* [Thi07, Definition 4.36].

Definition 12 (Root symbols of a TRS). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS. The set of* root symbols *associated to $\mathcal{R}$ is:*

$$Root(\mathcal{R}) = \{root(l) \mid l \to r \in R\} \cup \{root(r) \mid l \to r \in R, r \notin \mathcal{X}\}$$

The following result relates $Root(\mathcal{P})$ and the set $\mathcal{H}_\mathcal{P}$ of hidden symbols occurring *at the root* of terms in $\mathcal{NHT}_\mathcal{P}(\mathcal{R}, \mu)$. It is silently used in the statements of some theorems below.

Lemma 8. *Let $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs such that $Root(\mathcal{P}) \cap \mathcal{D} = \varnothing$, and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. For all $f \in \mathcal{H}_\mathcal{P}$, we have $f^\sharp \in Root(\mathcal{P})$.*

Proof. If $f \in \mathcal{H}_\mathcal{P}$, then there is $t \in \mathcal{NHT}_\mathcal{P}$ such that $f = root(t)$. Therefore, there are substitutions θ and θ' such that $\theta(t^\sharp) \hookrightarrow^*_{\mathcal{R},\mu} \theta'(u)$ for some $u \to v \in \mathcal{P}$. Since $f^\sharp \notin \mathcal{F}$, μ-rewritings on $\theta(t^\sharp)$ using $\mathcal{R}$ do not remove it. Thus, $root(u) = f^\sharp$ and $f^\sharp \in Root(\mathcal{P})$. □

Thiemann uses argument filterings (see Section 10.1) instead of *simple projections* [HM04, Definition 10]. We find it more convenient to follow Hirokawa and Middeldorp's style, so we generalize their definition to be used with TRSs rather than cycles in the dependency graph.

Definition 13 (Simple projection). *Let $\mathcal{R}$ be a TRS. A* simple projection *for $\mathcal{R}$ is a mapping π that assigns to every k-ary symbol $f \in Root(\mathcal{R})$ an argument position $i \in \{1, \ldots, k\}$. The mapping that assigns a subterm $\pi(t) = t|_{\pi(f)}$ to every term $t = f(t_1, \ldots, t_k)$ with $f \in Root(\mathcal{R})$ is also denoted by π; we also let $\pi(x) = x$ if $x \in \mathcal{X}$.*

Given a simple projection π for a TRS $\mathcal{R}$, we let $\pi(\mathcal{R}) = \{\pi(l) \to \pi(r) \mid l \to r \in \mathcal{R}\}$.

Theorem 12 (Subterm processor for noncollapsing pairs). *Let $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs such that $\mathcal{P}$ contains no collapsing rule, i.e., for all $u \to v \in \mathcal{P}$, $v \notin \mathcal{X}$, and $Root(\mathcal{P}) \cap \mathcal{D} = \varnothing$. Let $\mu \in M_{\mathcal{F} \cup \mathcal{G}}$ and let π be a simple projection for $\mathcal{P}$. Let $\mathcal{P}_{\pi,\rhd_\mu} = \{u \to v \in \mathcal{P} \mid \pi(u) \rhd_\mu \pi(v)\}$. Then, the processor $\mathsf{Proc}_{subNColl}$ given by*

$$\mathsf{Proc}_{subNColl}(\mathcal{P}, \mathcal{R}, \mu) = \begin{cases} \{(\mathcal{P} - \mathcal{P}_{\pi,\rhd_\mu}, \mathcal{R}, \mu)\} & \text{if } \pi(\mathcal{P}) \subseteq \unrhd_\mu \\ \{(\mathcal{P}, \mathcal{R}, \mu)\} & \text{otherwise} \end{cases}$$

is sound and complete.

Proof. Completeness is obvious because $\mathcal{P} - \mathcal{P}_{\pi,\rhd_\mu} \subseteq \mathcal{P}$. For soundness, we proceed by contradiction. Assume that there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain A but there is no infinite minimal $(\mathcal{P} - \mathcal{P}_{\pi,\rhd_\mu}, \mathcal{R}, \mu)$-chain. Since $\mathcal{P}$ is finite, we can assume that there is $Q \subseteq \mathcal{P}$ such that A has a tail B that is an infinite minimal $(Q, \mathcal{R}, \mu)$-chain where all pairs in Q are infinitely often used. Assume that B is as follows (since $Q_{\mathcal{X}} = \varnothing$, we use a simpler notation):

$$t_0 \hookrightarrow^*_{\mathcal{R},\mu} s_1 \overset{\Lambda}{\hookrightarrow}_{Q,\mu} t_1 \hookrightarrow^*_{\mathcal{R},\mu} s_2 \overset{\Lambda}{\hookrightarrow}_{Q,\mu} t_2 \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

where there is a substitution σ such that, for all $i \geq 1$, $s_i = \sigma(u_i)$ and $t_i = \sigma(v_i)$ for some $u_i \to v_i \in Q$. Furthermore, w.l.o.g. we also assume that $t_0 = \sigma(v_0)$ for some $u_0 \to v_0 \in \mathcal{P}$.

Note that, for all $i \geq 1$, $root(s_i) \in Root(\mathcal{P})$ because $root(u_i) \in Root(\mathcal{P})$. Since $root(v_i) \notin \mathcal{X}$, we have that $root(v_i) \in Root(\mathcal{P})$. Then, for all $i \geq 0$, $root(t_i) \in Root(\mathcal{P})$. Therefore, we can apply π to s_{i+1} and t_i for all $i \geq 0$. Moreover, since $t_i \hookrightarrow^*_{\mathcal{R},\mu} s_{i+1}$ for all $i \geq 0$ and $Root(\mathcal{P}) \cap \mathcal{D} = \varnothing$, we can actually write $t_i \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} s_{i+1}$ because μ-rewritings with $\mathcal{R}$ cannot change $root(t_i)$. Hence, $\pi(t_i) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_{i+1})$ and also $root(t_i) = root(s_{i+1})$ for all $i \geq 0$. Finally, since $\pi(u_i) \unrhd_\mu \pi(v_i)$ for all $i \geq 0$, by stability of $\unrhd_\mu$, we have

$$\pi(s_i) = \pi(\sigma(u_i)) = \sigma(\pi(u_i)) \unrhd_\mu \sigma(\pi(v_i)) = \pi(\sigma(v_i)) = \pi(t_i)$$

for all $i \geq 1$. No pair $u \to v \in Q$ satisfies that $\pi(u) \rhd_\mu \pi(v)$. Otherwise, we get a contradiction in both of the following two complementary cases:

1. if $\pi(f) \notin \mu(f)$ for all $f \in Root(Q)$, then, for all $i \geq 0$, $\pi(t_i) = \pi(s_{i+1})$, because no μ-rewritings are possible on the $\pi(root(t_i))$-th immediate subterm $\pi(t_i)$ of t_i. Since $\pi(s_{i+1}) \unrhd_\mu \pi(t_{i+1})$, we have that $\pi(t_i) \unrhd_\mu \pi(t_{i+1})$ for all $i \geq 0$. Furthermore, since we assume $\pi(u) \rhd_\mu \pi(v)$ for some $u \to v \in Q$ which occurs infinitely often in B, and by stability of $\rhd_\mu$, there is a maximal infinite set $J = \{j_1, j_2, \ldots\} \subseteq \mathbb{N}$ such that $\pi(t_{j_i}) \rhd_\mu \pi(t_{j_i+1})$ for all $i \geq 1$. We obtain an infinite sequence $\pi(t_{j_1}) \rhd_\mu \pi(t_{j_2}) \rhd_\mu \cdots$ which contradicts the well-foundedness of $\rhd_\mu$.
2. if $\pi(f) \in \mu(f)$ for some $f \in Root(Q)$, then, since $root(t_i) = root(s_{i+1})$ and all pairs in Q occur infinitely often in B, we can assume that $root(t_0) = f$. Furthermore, since A is minimal, we can assume that t_0 is μ-terminating (w.r.t. $\mathcal{R}$). Since $\pi(t_i) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_{i+1})$ and

$\pi(s_{i+1}) \trianglerighteq_\mu \pi(t_{i+1})$ for all $i \geq 0$, the sequence B is transformed into an infinite $\hookrightarrow_{\mathcal{R},\mu} \cup \trianglerighteq_\mu$-sequence

$$\pi(t_0) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_1) \trianglerighteq_\mu \pi(t_1) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_2) \trianglerighteq_\mu \pi(t_2) \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

containing infinitely many $\rhd_\mu$-steps, due to $\pi(u) \rhd_\mu \pi(v)$ for some $u \to v \in \mathcal{Q}$ which occurs infinitely often in B. Since $\rhd_\mu$ is well-founded, the infinite sequence must also contain infinitely many $\hookrightarrow_{\mathcal{R},\mu}$-steps. By making repeated use of the fact that $\rhd_\mu \circ \hookrightarrow_{\mathcal{R},\mu} \subseteq \hookrightarrow_{\mathcal{R},\mu} \circ \rhd_\mu$, we obtain an infinite $\hookrightarrow_{\mathcal{R},\mu}$-sequence starting from $\pi(t_0)$. Thus, $\pi(t_0)$ is not μ-terminating with respect to $\mathcal{R}$. Since $\pi(f) \in \mu(f)$ and hence $t_0 \rhd_\mu \pi(t_0)$, this implies that t_0 is not μ-terminating (use Lemma 1(1)). This contradicts μ-termination of t_0.

Hence, $\mathcal{Q} \subseteq \mathcal{P} - \mathcal{P}_{\pi,\rhd}$ and B is an infinite minimal $(\mathcal{P} - \mathcal{P}_{\pi,\rhd_\mu}, \mathcal{R}, \mu)$-chain. This contradicts our initial argument. □

Example 22 (Proof of termination of the main example). *Consider the termination problems obtained in Example 15 for the CS-TRS in Example 1:*

$$\tau_1 = (\{(1)\}, \mathcal{R}, \mu^\sharp), \quad \tau_2 = (\{(17)\}, \mathcal{R}, \mu^\sharp), \quad \tau_3 = (\{(21)\}, \mathcal{R}, \mu^\sharp), \quad \textit{and} \quad \tau_4 = (\{(23)\}, \mathcal{R}, \mu^\sharp)$$

We apply $\mathsf{Proc}_{\mathit{subNColl}}$ *to all such problems. For* τ_1*, with* $\pi(\mathtt{ADD}) = 1$*, we have* $\pi(\mathtt{ADD}(\mathtt{s}(n), m)) = \mathtt{s}(n) \rhd_\mu n = \pi(\mathtt{ADD}(n, m))$*. Now,* $\mathsf{Proc}_{\mathit{subNColl}}(\tau_1) = \{(\varnothing, \mathcal{R}, \mu^\sharp)\}$*. With* $\mathsf{Proc}_{\mathit{Fin}}$ *we conclude that* τ_1 *is finite. Since this can be done for* τ_2*,* τ_3*, and* τ_4*, the* μ*-termination of* $\mathcal{R}$ *is proved.*

The following examples show that if $\mathcal{P}$ contains collapsing rules, then Theorem 12 does not hold.

Example 23. *Consider the two TRSs*

$$\mathcal{R} \;:\; \mathtt{h}(x) \to \mathtt{f}(\mathtt{g}(\mathtt{h}(x))) \quad \textit{and} \quad \mathcal{P} \;:\; \mathtt{f}(\mathtt{g}(x)) \to x$$

Let μ *be given by* $\mu(f) = \{1, \ldots, k\}$ *for all symbols* f*. Note that,* $\mathit{Root}(\mathcal{P}) = \{\mathtt{f}\}$ *and* $\mathcal{D} = \{\mathtt{h}\}$ *are disjoint. By using the projection* $\pi(\mathtt{f}) = 1$*, we get* $\pi(\mathtt{f}(\mathtt{g}(x))) = \mathtt{g}(x) \rhd_\mu x$*. After removing the pair in* $\mathcal{P}$*, a finite CS problem* $(\varnothing, \mathcal{R}, \mu)$ *is obtained. However,* $(\mathcal{P}, \mathcal{R}, \mu)$ *is* not *finite:*

$$\underline{\mathtt{f}(\mathtt{g}(\mathtt{h}(x)))} \hookrightarrow_{\mathcal{P},\mu} \underline{\mathtt{h}(x)} \hookrightarrow_{\mathcal{R},\mu} \underline{\mathtt{f}(\mathtt{g}(\mathtt{h}(x)))} \hookrightarrow_{\mathcal{P},\mu} \cdots$$

In the following theorem, we show how to use the subterm criterion to *remove* all collapsing pairs from $\mathcal{P}$.

Theorem 13 (Subterm processor for collapsing pairs). *Let* $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ *and* $\mathcal{P} = (\mathcal{G}, P)$ *be TRSs such that* $\mathcal{P}_\mathcal{G}$ *contains no collapsing rule,* $\mathit{Root}(\mathcal{P}) \cap \mathcal{D} = \varnothing$*, and* $\mu \in M_{\mathcal{F} \cup \mathcal{G}}$*. Let* π *be a simple projection for* $\mathcal{P}$ *such that*

1. $\pi(\mathcal{P}) \subseteq \trianglerighteq_\mu$*, and*
2. *whenever* $\mathcal{P}_\mathcal{X} \neq \varnothing$*, we have* $\pi(f^\sharp) \in \mu(f^\sharp) \cap \mu(f)$ *for all* $f \in \mathcal{H}_\mathcal{P}$*.*

Then, the processor $\mathsf{Proc}_{\mathit{subColl}}$ *given by*

$$\mathsf{Proc}_{\mathit{subColl}}(\mathcal{P}, \mathcal{R}, \mu) = \begin{cases} \{(\mathcal{P}_\mathcal{G}, \mathcal{R}, \mu)\} & \textit{if (1) and (2) hold} \\ \{(\mathcal{P}, \mathcal{R}, \mu)\} & \textit{otherwise} \end{cases}$$

is sound and complete.

Proof. Completeness is obvious because $\mathcal{P}_\mathcal{G} \subseteq \mathcal{P}$. For soundness, we proceed by contradiction. Assume that there is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain A but there is no infinite minimal $(\mathcal{P}_\mathcal{G}, \mathcal{R}, \mu)$-chain. Since $\mathcal{P}$ is finite, we can assume that there is $\mathcal{Q} \subseteq \mathcal{P}$ such that A has a tail B which is an infinite minimal $(\mathcal{Q}, \mathcal{R}, \mu)$-chain where all pairs in $\mathcal{Q}$ are infinitely often used and $\mathcal{Q}$ contains some collapsing pair $u \to x \in \mathcal{Q}_\mathcal{X}$. Assume that B is

$$t_0 \hookrightarrow^*_{\mathcal{R},\mu} s_1 \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^\sharp_\mu t_1 \hookrightarrow^*_{\mathcal{R},\mu} s_2 \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^\sharp_\mu t_2 \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

where there is a substitution σ such that, for all $i \geq 1$, $s_i = \sigma(u_i)$ for some $u_i \to v_i \in \mathcal{P}$, and

1. if $v_i \notin \mathcal{X}$, then $t_i = \sigma(v_i)$, and
2. if $v_i = x_i \in \mathcal{X}$, then $x_i \notin \mathcal{V}ar^\mu(u_i)$ and $t_i = r_i^\sharp$ for some $r_i \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ such that $\sigma(x_i) = C_i[r_i]_{p_i}$ for some $C_i[\,]_{p_i}$ and $p_i \in \mathcal{P}os^\mu(C_i[\,]_{p_i})$ such that $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$ and $r_i = \theta_i(\bar{r}_i)$ for some $\bar{r}_i \in \mathcal{NHT}_\mathcal{Q}$ and substitution θ_i.

Since we can freely choose the starting term of B, w.l.o.g. we assume that t_0 is a particular case of the second alternative above, i.e., there is a collapsing pair $u_0 \to x_0$ such that $\sigma(x_0) \trianglerighteq_\mu r_0$ and $t_0 = r_0^\sharp$. Note that, for all $i \geq 1$, $\mathit{root}(s_i) \in \mathit{Root}(\mathcal{P})$ because $\mathit{root}(u_i) \in \mathit{Root}(\mathcal{P})$. Furthermore, for all $i \geq 0$, $\mathit{root}(t_i) \in \mathit{Root}(\mathcal{P})$ because:

1. If $u_i \to v_i \in \mathcal{Q}_\mathcal{G}$, then $\mathit{root}(v_i) \in \mathit{Root}(\mathcal{P})$ and $t_i = \sigma(v_i)$.
2. If $u_i \to v_i \in \mathcal{Q}_\mathcal{X}$, then $\mathit{root}(t_i) \in \mathcal{D}^\sharp$; since $t_i \hookrightarrow^*_{\mathcal{R},\mu} s_{i+1}$ and $\mathcal{D}^\sharp \cap \mathcal{F} = \varnothing$ (see Remark 4), rewritings with $\mathcal{R}$ cannot remove the marked root symbol in t_i; hence, we can further conclude $\mathit{root}(t_i) = \mathit{root}(s_{i+1}) \in \mathit{Root}(\mathcal{P})$.

Therefore, we can apply π to s_{i+1} and t_i for all $i \geq 0$. Moreover, since $t_i \hookrightarrow^*_{\mathcal{R},\mu} s_{i+1}$ for all $i \geq 0$ and $\mathit{Root}(\mathcal{P}) \cap \mathcal{D} = \varnothing$, we can actually write $t_i \xhookrightarrow{>\Lambda}{}^*_{\mathcal{R},\mu} s_{i+1}$. Hence, $\pi(t_i) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_{i+1})$ and also $\mathit{root}(t_i) = \mathit{root}(s_{i+1})$ for all $i \geq 0$.

Since $u \to x \in \mathcal{Q}_\mathcal{X}$ and B is infinite, it must be $\mathcal{H}_\mathcal{Q} \neq \varnothing$ (hence $\mathcal{H}_\mathcal{P} \neq \varnothing$). Thus, we have $\pi(f^\sharp) \in \mu(f)$ for all $f \in \mathcal{H}_\mathcal{Q} \subseteq \mathcal{H}_\mathcal{P}$. Then, since $\mathit{root}(t_i) = \mathit{root}(s_{i+1})$ and all pairs in $\mathcal{Q}$ occur infinitely often in B, we can assume that $\mathit{root}(t_0) = f$. Furthermore, since A is minimal, we can assume that t_0 is μ-terminating. We have that $\pi(u_i) \trianglerighteq_\mu \pi(v_i)$ for all $u_i \to v_i \in \mathcal{Q}$. Now we distinguish two cases:

1. If $u_i \to v_i \in \mathcal{Q}_\mathcal{G}$, then $s_i = \sigma(u_i)$ and $t_i = \sigma(v_i)$. By stability of $\trianglerighteq_\mu$ we have $\pi(s_i) \trianglerighteq_\mu \pi(t_i)$.
2. If $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_\mathcal{X}$, then $s_i = \sigma(u_i)$ and there is a term r_i, such that $\sigma(x_i) \trianglerighteq_\mu r_i$ and $r_i^\sharp = t_i$. Since $\pi(u_i) \trianglerighteq_\mu x_i$, by stability of $\trianglerighteq_\mu$ we have

$$\pi(s_i) = \pi(\sigma(u_i)) = \sigma(\pi(u_i)) \trianglerighteq_\mu \sigma(x_i) \trianglerighteq_\mu r_i.$$

   Note that $f_i = \mathit{root}(r_i) = \mathit{root}(\bar{r}_i) \in \mathcal{H}_\mathcal{P}$. Since $\pi(t_{i+1}) = t_i|_{\pi(f_i^\sharp)} = r_i^\sharp|_{\pi(f_i^\sharp)} = r_i|_{\pi(f_i^\sharp)}$ and $\pi(f_i^\sharp) \in \mu(f_i)$, we have that $r_i \vartriangleright_\mu \pi(t_i)$ and thus $\pi(s_i) \vartriangleright_\mu \pi(t_i)$.

Therefore, by applying the simple projection π, the sequence B is transformed into an infinite $\hookrightarrow_{\mathcal{R},\mu} \cup \vartriangleright_\mu$-sequence B'

$$\pi(t_0) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_1) \trianglerighteq_\mu \pi(t_1) \hookrightarrow^*_{\mathcal{R},\mu} \pi(s_2) \trianglerighteq_\mu \pi(t_2) \hookrightarrow^*_{\mathcal{R},\mu} \cdots$$

Since $u \to x$ occurs infinitely often in B, and by case (2) above, B' contains infinitely many $\vartriangleright_\mu$ steps, starting from $\pi(t_0)$. Since $\vartriangleright_\mu$ is well-founded, the infinite sequence must also contain infinitely many $\hookrightarrow_{\mathcal{R},\mu}$-steps. By making repeated use of the fact that $\rhd_\mu \circ \hookrightarrow_{\mathcal{R},\mu} \subseteq \hookrightarrow_{\mathcal{R},\mu} \circ \rhd_\mu$, we obtain an infinite $\hookrightarrow_{\mathcal{R},\mu}$-sequence starting from $\pi(t_0)$. Thus, $\pi(t_0)$ is not μ-terminating with respect to $\mathcal{R}$. Since $\pi(f^\sharp) \in \mu(f^\sharp)$ and hence $t_0 \rhd_\mu \pi(t_0)$, this implies that t_0 is not μ-terminating (use Lemma 1(1)). This contradicts μ-termination of t_0. Therefore, $\mathcal{Q}$ cannot contain collapsing pairs. This contradicts our initial assumption $u \to x \in \mathcal{Q}$. $\square$

Remark 13. *The use of Theorem 13 only makes sense if* $\mathcal{P} \subseteq \mathcal{P}_\mathcal{G} \cup \mathcal{P}^1_\mathcal{X}$. *If* $u \to x \in \mathcal{P}_\mathcal{X} - \mathcal{P}^1_\mathcal{X}$ *for some* $u = f(u_1, \ldots, u_k)$*, then for all* $i \in \{1, \ldots, k\}$*, whenever* $x \in \mathcal{V}ar(u_i)$ *we have* $i \in \mu(f)$ *and* $u_i \rhd_{\not\mu} x$*. Thus, there is no simple projection* π *such that* $\pi(u) \unrhd_\mu x$*.*

Example 24. *Consider the following TRS* $\mathcal{R}$*:*

$$\begin{array}{rcl} \mathtt{g}(x,y) & \to & \mathtt{f}(x,y) \\ \mathtt{f}(\mathtt{c}(x),y) & \to & \mathtt{g}(x,\mathtt{g}(y,y)) \end{array}$$

together with the replacement map μ *given by* $\mu(\mathtt{c}) = \mu(\mathtt{g}) = \{1\}$ *and* $\mu(\mathtt{f}) = \varnothing$*. The CSDPs are:*

$$\mathtt{G}(x,y) \to \mathtt{F}(x,y) \tag{33}$$

$$\mathtt{F}(\mathtt{c}(x),y) \to \mathtt{G}(x,\mathtt{g}(y,y)) \tag{34}$$

$$\mathtt{F}(\mathtt{c}(x),y) \to x \tag{35}$$

and all of them are part of the only SCC $\mathcal{P} = \{(33),(34),(35)\}$ *in the CSDG of* $(\mathcal{R},\mu)$*. Note that* $\mathcal{NHT}_\mathcal{P} = \{\mathtt{g}(y,y)\}$*; hence* $\mathcal{H}_\mathcal{P} = \{\mathtt{g}\}$*. Consider the simple projection* π *given by* $\pi(\mathtt{F}) = \pi(\mathtt{G}) = 1$*. Note that* $\pi(\mathtt{G}) \in \mu^\sharp(\mathtt{G}) \cap \mu^\sharp(\mathtt{g})$ *as required by Theorem 13. We have*

- $\pi(\mathtt{G}(x,y)) = x \unrhd_\mu x = \pi(\mathtt{F}(x,y))$
- $\pi(\mathtt{F}(\mathtt{c}(x),y)) = \mathtt{c}(x) \rhd_\mu x = \pi(\mathtt{G}(x,\mathtt{g}(y,y)))$*, and*
- $\pi(\mathtt{F}(\mathtt{c}(x),y)) = \mathtt{c}(x) \rhd_\mu x = \pi(x)$

We use $\mathsf{Proc}_{subColl}$ *to remove* (35) *from* $\mathcal{P}$ *and obtain a new problem* $(\{(33),(34)\},\mathcal{R},\mu^\sharp)$*. Then,* $\mathsf{Proc}_{subNColl}$ *applies and yields* $(\{(33)\},\mathcal{R},\mu^\sharp)$*. With* Proc_{SCC}*, we conclude the* μ*-termination of* $\mathcal{R}$*.*

The following result provides a kind of generalization of the subterm criterion to simple projections that only take *non-μ-replacing* arguments.

Theorem 14 (Non-μ-replacing projection processor). *Let* $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ *and* $\mathcal{P} = (\mathcal{G}, P)$ *be TRSs such that* $\mathcal{P}_\mathcal{G}$ *contains no collapsing rule,* $Root(\mathcal{P}) \cap \mathcal{D} = \varnothing$*, and* $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$*. Let* $\gtrsim$ *be a stable quasi-ordering on terms whose strict and stable part* $>$ *is well-founded and* π *be a simple projection for* $\mathcal{P}$ *such that*

1. *for all* $f \in Root(\mathcal{P})$, $\pi(f) \notin \mu(f)$,
2. $\pi(\mathcal{P}) \subseteq \gtrsim$*, and,*
3. *whenever* $\mathcal{NHT}_\mathcal{P} \neq \varnothing$ *and* $\mathcal{P}_\mathcal{X} \neq \varnothing$*, we have that* $\mathcal{E}mb^\mu(\mathcal{F}) \subseteq \gtrsim$ *and* $t \gtrsim t|_{\pi(root(t)^\sharp)}$ *for all* $t \in \mathcal{NHT}_\mathcal{P}$*.*

Let $\mathcal{P}_> = \{u \to v \in \mathcal{P} \mid \pi(u) > \pi(v)\}$*. Then, the processor* Proc_{NRP} *given by*

$$\mathsf{Proc}_{NRP}(\mathcal{P},\mathcal{R},\mu) = \left\{ \begin{array}{ll} \{(\mathcal{P} - \mathcal{P}_>, \mathcal{R}, \mu)\} & \textit{if (1), (2), and (3) hold} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{array} \right.$$

is sound and complete.

Proof. Completeness is obvious because $\mathcal{P}-\mathcal{P}_> \subseteq \mathcal{P}$. For soundness, we proceed by contradiction. Assume that there is an infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain A but there is no infinite minimal $(\mathcal{P}-\mathcal{P}_>,\mathcal{R},\mu)$-chain. Since $\mathcal{P}$ is finite, we can assume that there is $\mathcal{Q} \subseteq \mathcal{P}$ such that A has a tail B

$$\sigma(u_1) \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_1 \hookrightarrow^{*}_{\mathcal{R},\mu} \sigma(u_2) \overset{\Lambda}{\hookrightarrow}_{\mathcal{Q},\mu} \circ \trianglerighteq^{\sharp}_{\mu} t_2 \hookrightarrow^{*}_{\mathcal{R},\mu} \cdots$$

for some substitution σ and pairs $u_i \to v_i \in \mathcal{Q}$, and

1. if $v_i \notin \mathcal{X}$, then $t_i = \sigma(v_i)$, and
2. if $v_i = x_i \in \mathcal{X}$, then $x_i \notin \mathcal{V}ar^{\mu}(u_i)$ and $t_i = s_i^{\sharp}$ for some s_i such that $\sigma(x_i) = C_i[s_i]_{p_i}$ for some $C_i[\,]_{p_i}$ and $p_i \in \mathcal{P}os^{\mu}(C_i[\,]_{p_i})$ such that $\mathit{sprefix}_{C_i[\,]_{p_i}}(p_i) \subseteq \mathcal{F}$ and $s_i = \theta_i(\bar{s}_i)$ for some $\bar{s}_i \in \mathcal{NHT}_{\mathcal{P}}$ and substitution θ_i.

Furthermore, *all pairs* in $\mathcal{Q}$ are used infinitely often in B. As discussed in the proof of Theorem 12, for all $i \geq 1$, $\mathit{root}(t_i) \in \mathit{Root}(\mathcal{P})$, $\pi(t_i) \hookrightarrow^{*}_{\mathcal{R},\mu} \pi(\sigma(u_{i+1}))$ and also $\mathit{root}(t_i) = \mathit{root}(u_{i+1})$ for all $i \geq 1$. No pair $u \to v \in \mathcal{Q}$ satisfies that $\pi(u) > \pi(v)$. Otherwise, by applying the simple projection π to the sequence B, we get a contradiction as follows:

1. Since $\pi(f) \notin \mu(f)$ for all $f \in \mathit{Root}(\mathcal{Q})$, no μ-rewritings are possible on the subterm $\pi(t_i)$ of t_i. Therefore, for all $i \geq 1$, $\pi(t_i) = \pi(\sigma(u_{i+1})) = \sigma(\pi(u_{i+1}))$.
2. Due to $\pi(u_i) \gtrsim \pi(v_i)$ and by stability of $\gtrsim$, we have that $\pi(\sigma(u_i)) = \sigma(\pi(u_i)) \gtrsim \sigma(\pi(v_i))$. Now, we distinguish two cases:
    - (a) If $u_i \to v_i \in \mathcal{Q}_{\mathcal{G}}$, then $\pi(t_i) = \pi(\sigma(v_i)) = \sigma(\pi(v_i))$. Thus, $\pi(\sigma(u_i)) \gtrsim \pi(t_i)$.
    - (b) If $u_i \to v_i \in \mathcal{Q}_{\mathcal{X}}$, then $\sigma(\pi(v_i)) = \sigma(x_i)$. We have that $\sigma(x_i) \gtrsim s_i$ (because $\mathcal{E}mb^{\mu}(\mathcal{F}) \subseteq\, \gtrsim$). Let $f = \mathit{root}(u_{i+1}) = \mathit{root}(t_i) = \mathit{root}(\bar{s}_i^{\sharp})$. Since $t \gtrsim t|_{\pi(\mathit{root}(t)^{\sharp})}$ for all $t \in \mathcal{NHT}_{\mathcal{P}}$, by stability, we have $s_i = \theta_i(\bar{s}_i) \gtrsim \theta_i(\bar{s}_i|_{\pi(f)}) = \theta_i(\bar{s}_i)|_{\pi(f)} = s_i|_{\pi(f)}$. Since $s_i|_{\pi(f^{\sharp})} = t_i|_{\pi(f^{\sharp})} = \pi(t_i)$, we have $s_i \gtrsim \pi(t_i)$. Hence, $\pi(\sigma(u_i)) \gtrsim \pi(t_i)$.

Thus, we always have $\pi(\sigma(u_i)) \gtrsim \pi(t_i)$. We obtain an infinite $\gtrsim$ sequence

$$\pi(\sigma(u_1)) \gtrsim \pi(t_1) = \pi(\sigma(u_2)) \gtrsim \pi(t_2) \cdots$$

Since pairs in $\mathcal{Q}$ occur infinitely often, this sequence contains infinitely many $>$ steps starting from $\pi(\sigma(u_1))$. This contradicts the well-foundedness of $>$.

Therefore, $\mathcal{Q} \subseteq \mathcal{P}-\mathcal{P}_>$, i.e., B is an infinite minimal $(\mathcal{P}-\mathcal{P}_>,\mathcal{R},\mu)$-chain. This contradicts our initial assumption. □

Example 25. *Consider the CS-TRS $(\mathcal{R},\mu)$ in Example 10.* $\mathsf{DP}(\mathcal{R},\mu)$ *is:*

$$\mathtt{G}(x) \to \mathtt{H}(x) \qquad\qquad \mathtt{H}(\mathtt{d}) \to \mathtt{G}(\mathtt{c})$$

where $\mu^{\sharp}(\mathtt{G}) = \mu^{\sharp}(\mathtt{H}) = \varnothing$. The dependency graph contains a single cycle that includes both pairs. The only simple projection is $\pi(\mathtt{G}) = \pi(\mathtt{H}) = 1$. Since $\pi(\mathtt{G}(x)) = \pi(\mathtt{H}(x))$, we only need to guarantee that $\pi(\mathtt{H}(\mathtt{d})) = \mathtt{d} > \mathtt{c} = \pi(\mathtt{G}(\mathtt{c}))$ holds for a stable and well-founded ordering $>$ (e.g., an RPO with $\mathtt{d} > \mathtt{c}$).

Theorem 15 (Non-μ-replacing projection processor II). *Let $\mathcal{R} = (\mathcal{F},R) = (\mathcal{C} \uplus \mathcal{D},R)$ and $\mathcal{P} = (\mathcal{G},P)$ be TRSs such that $\mathcal{P}_{\mathcal{G}}$ contains no collapsing rule, $\mathit{Root}(\mathcal{P}) \cap \mathcal{D} = \varnothing$, and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. Let $\gtrsim$ be a stable quasi-ordering on terms whose strict and stable part $>$ is well-founded and let π be a simple projection for $\mathcal{P}$ such that*

1. *for all* $f \in Root(\mathcal{P})$, $\pi(f) \notin \mu(f)$*,*
2. $\pi(\mathcal{P}) \subseteq \gtrsim$*, and,*
3. *whenever* $\mathcal{NHT}_\mathcal{P} \neq \emptyset$ *and* $\mathcal{P}_\mathcal{X} \neq \emptyset$*, we have that* $\mathcal{E}mb^\mu(\mathcal{F}) \subseteq \gtrsim$ *and* $t > t|_{\pi(root(t)^\sharp)}$ *for all* $t \in \mathcal{NHT}_\mathcal{P}$*.*

Then, the processor Proc_{NRP2} *given by*

$$\mathsf{Proc}_{NRP2}(\mathcal{P},\mathcal{R},\mu) = \begin{cases} \{(\mathcal{P}_\mathcal{G},\mathcal{R},\mu)\} & \textit{if (1), (2), and (3) hold} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{cases}$$

is sound and complete.

12. Narrowing Transformation

The starting point of a proof of μ-termination of a TRS $\mathcal{R}$ is the computation of the CSDG $\mathsf{EDG}(\mathcal{R},\mu)$ followed by the use of the SCC processor (Theorem 6). The estimation of the graph can lead to *overestimating* the arcs that connect two CSDPs.

Example 26. *Consider the following example [Luc06, Proposition 7]:*

$$\begin{array}{rcl} \mathtt{f(0)} & \to & \mathtt{cons(0,f(s(0)))} \\ \mathtt{f(s(0))} & \to & \mathtt{f(p(s(0)))} \end{array} \qquad \begin{array}{rcl} \mathtt{p(s(}x\mathtt{))} & \to & x \end{array}$$

together with $\mu(\mathtt{f}) = \mu(\mathtt{p}) = \mu(\mathtt{s}) = \mu(\mathtt{cons}) = \{1\}$ *and* $\mu(\mathtt{0}) = \emptyset$*.* $\mathsf{DP}(\mathcal{R},\mu)$ *consists of the pairs:*

$$\mathtt{F(s(0))} \to \mathtt{F(p(s(0)))} \tag{36}$$

$$\mathtt{F(s(0))} \to \mathtt{P(s(0))} \tag{37}$$

The estimated *CS-dependency graph contains one cycle:* $\{(36)\}$*. However, this cycle does* not *belong to the CS-dependency graph because there is no way to* μ*-rewrite* $\mathtt{F(p(s(0)))}$ *into* $\mathtt{F(s(0))}$*.*

As already observed by Arts and Giesl for the standard case [AG00], in our case, the overestimation comes when a (noncollapsing) pair $u_i \to v_i$ is followed in a chain by a second one $u_{i+1} \to v_{i+1}$ and v_i and u_{i+1} are not directly unifiable, i.e., at least one μ-rewriting step is needed to μ-reduce $\sigma(v_i)$ to $\sigma(u_{i+1})$. Then, we always have $\sigma(v_i) \hookrightarrow_{\mathcal{R},\mu^\sharp} \sigma(v'_i) \hookrightarrow^*_{\mathcal{R},\mu^\sharp} \sigma(u_{i+1})$. Then, v'_i is a one-step μ-narrowing of v_i, and we could require $u_i \sqsupset v'_i$ (which could be easier to prove) instead of $u_i \sqsupset v_i$. Furthermore, we could discover that v_i has *no* μ*-narrowings*. In this case, we know that no chain starts from $\sigma(v_i)$.

We can be more precise when connecting two pairs $u \to v$ and $u' \to v'$ in a chain if we perform all the possible one-step μ-narrowings on v in order to develop the possible reductions from $\sigma(v)$ to $\sigma(u')$. Then, we obtain new terms $v_1, \ldots, v_n$, which are one-step μ-narrowings of v using unifiers θ_i (i.e., $v \leadsto_{\mathcal{R},\mu,\theta_i} v_i$) for $i \in \{1,\ldots,n\}$, respectively. These unifiers are also applied to the left-hand side u of the pair $u \to v$. Therefore, we can replace a pair $u \to v$ by all its (one-step) μ*-narrowed pairs* $\theta_1(u) \to v_1, \ldots, \theta_n(u) \to v_n$.

As in [AG00, GTSF06], a pair $u \to v \in \mathcal{P}$ can only be replaced by its μ-narrowings if the right-hand side v does not unify with any left-hand side u' of a (possibly renamed) pair $u' \to v' \in \mathcal{P}$ (note that this excludes pairs $u \to v$ with $v \in \mathcal{X}$). Moreover, the term v must be *linear*. We need to demand linearity instead of (the apparently more natural) μ-linearity (i.e., something like "no multiple μ*-replacing* occurrences of the same variable are allowed").

Example 27. *The following TRS is used in [AG00] to motivate the requirement of linearity.*

$$\begin{array}{rcl} \mathtt{f}(\mathtt{s}(x)) & \to & \mathtt{f}(\mathtt{g}(x,x)) \\ \mathtt{g}(\mathtt{0},\mathtt{1})) & \to & \mathtt{s}(\mathtt{0}) \\ \mathtt{0} & \to & \mathtt{1} \end{array}$$

We make it a CS-TRS by adding a replacement map μ *given by* $\mu(\mathtt{f}) = \mu(\mathtt{s}) = \{1\}$*, and* $\mu(\mathtt{g}) = \{2\}$*. The only cycle in the CSDG consists of the CSDP*

$$\mathtt{F}(\mathtt{s}(x)) \to \mathtt{F}(\mathtt{g}(x,x)).$$

If linearity of the right-hand sides is not required for μ*-narrowing CSDPs, then this pair will be removed, since* $\mathtt{F}(\mathtt{g}(x,x))$ *and the (renamed version of) the left-hand side* $\mathtt{F}(\mathtt{s}(x'))$ *do not unify. Thus, there are no* μ*-narrowings. However, the system is* not μ-terminating*:*

$$\underline{\mathtt{f}(\mathtt{s}(\mathtt{0}))} \hookrightarrow \mathtt{f}(\mathtt{g}(\mathtt{0},\underline{\mathtt{0}})) \hookrightarrow \mathtt{f}(\underline{\mathtt{g}(\mathtt{0},\mathtt{1})}) \hookrightarrow \underline{\mathtt{f}(\mathtt{s}(\mathtt{0}))}\ldots$$

The problem is that the μ*-reduction from* $\sigma(\mathtt{F}(\mathtt{g}(x,x)))$ *to* $\sigma(\mathtt{F}(\mathtt{s}(x')))$ *takes place 'in* σ*', and, therefore, it cannot be captured by* μ*-narrowing. Note that* $\mathtt{F}(\mathtt{g}(x,x))$ *is "*μ*-linear".*

Another restriction to take into account when μ-narrowing a noncollapsing pair $u \to v$ is that the μ-replacing variables in v have to be μ-replacing in u as well (this corresponds with the notion of conservativeness). Furthermore, they cannot be both μ-replacing and non-μ-replacing at the same time. This corresponds to the following definition.

Definition 14 (Strongly Conservative [GLU08]**).** *Let* $\mathcal{R}$ *be a TRS and* $\mu \in M_{\mathcal{R}}$*. A rule* $l \to r$ *is strongly* μ*-conservative if it is* μ*-conservative and* $\mathcal{V}ar^{\mu}(l) \cap \mathcal{V}ar^{\not\mu}(l) = \mathcal{V}ar^{\mu}(r) \cap \mathcal{V}ar^{\not\mu}(r) = \varnothing$*.*

The following result shows that, under these conditions, the set of CSDPs can be safely replaced by their μ-narrowings.

Theorem 16 (Narrowing processor). *Let* $\mathcal{R}$ *and* $\mathcal{P}$ *be TRSs and* $\mu \in M_{\mathcal{R}\cup\mathcal{P}}$*. Let* $u \to v \in \mathcal{P}$ *be such that*

1. v *is linear, and*
2. *for all* $u' \to v' \in \mathcal{P}$ *(with possibly renamed variables),* v *and* u' *do not unify.*

Let $\mathcal{Q} = (\mathcal{P} - \{u \to v\}) \cup \{u' \to v' \mid u' \to v' \text{ is a } \mu\text{-narrowing of } u \to v\}$*. Then, the processor* Proc_{narr} *given by*

$$\mathsf{Proc}_{narr}(\mathcal{P},\mathcal{R},\mu) = \begin{cases} \{(\mathcal{Q},\mathcal{R},\mu)\} & \textit{if (1) and (2) hold} \\ \{(\mathcal{P},\mathcal{R},\mu)\} & \textit{otherwise} \end{cases}$$

is

1. *sound whenever* $u \to v$ *is strongly conservative, and*
2. *complete in all cases.*

Proof. The proof of this theorem is analogous to the proof of [GTSF06, Theorem 31], which we adapt here. For soundness, we prove that given a minimal $(\mathcal{P},\mathcal{R},\mu)$-chain "$\ldots, u_1 \to v_1, u \to v, u_2 \to v_2, \ldots$", there is a μ-narrowing v' of v with the mgu θ such that "$\ldots, u_1 \to v_1, \theta(u) \to v', u_2 \to v_2, \ldots$" is also a minimal $(\mathcal{Q},\mathcal{R},\mu)$-chain. Hence, every infinite minimal $(\mathcal{P},\mathcal{R},\mu)$-chain yields an infinite minimal $(\mathcal{Q},\mathcal{R},\mu)$-chain.

If "$\ldots, u_1 \to v_1, u \to v, u_2 \to v_2, \ldots$" is a minimal $(\mathcal{P},\mathcal{R},\mu)$-chain, then there is a substitution σ such that for all pairs $s \to t$ in the chain,

1. if $s \to t \in \mathcal{P}_{\mathcal{G}}$, then $\sigma(t)$ is μ-terminating and it μ-reduces to the instantiated left-hand side $\sigma(s')$ of the next pair $s' \to t'$ in the chain
2. if $s \to t = s \to x \in \mathcal{P}_{\mathcal{X}}$ then, $\sigma(x)$ has a μ-replacing subterm s_0, $\sigma(x) \trianglerighteq_\mu s_0$ such that $s_0^\sharp$ is μ-terminating and it μ-reduces to the instantiated left-hand side $\sigma(s')$ of the next pair $s' \to t'$ in the chain; furthermore, there is $\bar{s}_0 \in \mathcal{NHT}(\mathcal{R},\mu)$ such that $s_0 = \theta_0(\bar{s}_0)$ for some substitution θ_0.

Assume that σ is a substitution satisfying the above requirements and such that the length of the sequence $\sigma(v) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2)$ is *minimum*. Note that the length of this μ-reduction sequence cannot be zero because v and u_2 do not unify, that is, $\sigma(v) \neq \sigma(u_2)$. Hence, there is a term q such that $\sigma(v) \hookrightarrow_{\mathcal{R},\mu} q \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2)$. We consider two possible cases:

1. The reduction $\sigma(v) \hookrightarrow_{\mathcal{R},\mu} q$ takes place within a binding of σ, i.e., there is a term r, a μ-replacing variable position $p \in \mathcal{P}os^\mu_{\mathcal{X}}(v)$, and a μ-replacing variable $x \in \mathcal{V}ar^\mu(v)$ such that $v|_p = x$, $q = \sigma(v[r]_p)$ and $\sigma(x) \hookrightarrow_{\mathcal{R},\mu} r$. Since v is linear, x occurs only once in v. Thus, $q = \sigma'(v)$ for the substitution σ' with $\sigma'(x) = r$ and $\sigma'(y) = \sigma(y)$ for all variables $y \neq x$. As we assume that all occurrences of pairs in the chain are variable disjoint, $\sigma'(x)$ behaves like σ for all pairs except $u \to v$. We have $\sigma(z) \hookrightarrow^*_{\mathcal{R},\mu} \sigma'(z)$ for all $z \in \mathcal{X}$. Since $u \to v$ is strongly conservative, we also have $\sigma(u) \hookrightarrow^*_{\mathcal{R},\mu} \sigma'(u)$ because all occurrences of x in u must be μ-replacing. Hence, if $u_1 \to v_1 \in \mathcal{P}_{\mathcal{G}}$ we have
$$\sigma'(v_1) = \sigma(v_1) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) \hookrightarrow^*_{\mathcal{R},\mu} \sigma'(u)$$
and if $u_1 \to v_1 \in \mathcal{P}_{\mathcal{X}}$, then there is $s_1 \in \mathcal{T}(\mathcal{F},\mathcal{X})$ such that
$$\sigma'(v_1) = \sigma(v_1) \trianglerighteq_\mu s_1 \text{ and } s_1^\sharp \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) \hookrightarrow^*_{\mathcal{R},\mu} \sigma'(u)$$
and, in both cases,
$$\sigma'(v) = q \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2) = \sigma'(u_2).$$
Note that, by minimality and because $u \to v \in \mathcal{P}_{\mathcal{G}}$, $\sigma(v)$ is $(\mathcal{R},\mu)$-terminating and, since $\sigma(v) \hookrightarrow_{\mathcal{R},\mu} q$, the term q is $(\mathcal{R},\mu)$-terminating as well. Therefore, $\sigma'(x) = q$ is $(\mathcal{R},\mu)$-terminating and σ' satisfies the two conditions above. Since the length of the sequence $\sigma'(v) \hookrightarrow^*_{\mathcal{R},\mu} \sigma'(u_2)$ is shorter than the sequence $\sigma(v) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2)$, we obtain a contradiction and we conclude that the μ-reduction $\sigma(v) \hookrightarrow_{\mathcal{R},\mu} q$ cannot take place in a binding of σ.
2. The reduction $\sigma(v) \hookrightarrow_{\mathcal{R},\mu} q$ 'touches' v, i.e., there is a nonvariable position $p \in \mathcal{P}os^\mu_{\mathcal{F}}(v)$, and a rewrite rule $l \to r \in \mathcal{R}$ such that $\sigma(v|_p) = \rho(l)$, for some substitution ρ and
$$\sigma(v) = \sigma(v)[\sigma(v|_p)]_p = \sigma(v)[\rho(l)]_p \hookrightarrow_{\mathcal{R},\mu} \sigma(v)[\rho(r)]_p = q$$
Since we can assume that variables in l are fresh, we can extend σ to behave like ρ on variables in l. Thus, $\sigma(l) = \sigma(v|_p)$, i.e, l and $v|_p$ unify and there is a mgu θ and a substitution τ satisfying $\sigma(x) = \tau(\theta(x))$ for all variables x. We have that v μ-narrows to $\theta(v)[\theta(r)]_p = v'$ with unifier θ. Again, we can extend σ to behave like τ on the variables of $\theta(u)$ and v'. Therefore, if $u_1 \to v_1 \in \mathcal{P}_{\mathcal{G}}$ we have
$$\sigma(v_1) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) = \tau(\theta(u)) = \sigma(\theta(u))$$
and if $u_1 \to v_1 \in \mathcal{P}_{\mathcal{X}}$, then there is $s_1 \in \mathcal{T}(\mathcal{F},\mathcal{X})$ such that
$$\sigma(v_1) = \sigma(x) \trianglerighteq_\mu s_1 \text{ and } s_1^\sharp \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) = \tau(\theta(u)) = \sigma(\theta(u))$$

and

$$\sigma(v') = \tau(v') = \tau(\theta(v))[\tau(\theta(r))]_p = \sigma(v)[\sigma(r)]_p = \sigma(v)[\rho(r)]_p = q \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2)$$

Hence, "$\ldots, u_1 \to v_1, \theta(u) \to v', u_2 \to v_2, \ldots$" is also a minimal chain.

Completeness is also analogous to the 'completeness' part of [GTSF06, Theorem 31]. If $(\mathcal{Q}, \mathcal{R}, \mu)$ is infinite and $\mathcal{R}$ is non-μ-terminating, then $(\mathcal{P}, \mathcal{R}, \mu)$ is infinite as well. If $\mathcal{R}$ is μ-terminating, then let "$\ldots, u_1 \to v_1, \theta(u) \to v', u_2 \to v_2, \ldots$" be an infinite minimal $(\mathcal{Q}, \mathcal{R}, \mu)$-chain where v' is a one-step μ-narrowing of v using the mgu θ. We prove that "$\ldots, u_1 \to v_1, u \to v, u_2 \to v_2, \ldots$" is an infinite minimal $(\mathcal{P}, \mathcal{R}, \mu)$-chain. There is a substitution σ such that

$$\sigma(v_1) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(\theta(u)) \;\text{ if } u_1 \to v_1 \in \mathcal{P}_{\mathcal{G}}, \text{ and}$$

$$\sigma(v_1) = \sigma(x) \trianglerighteq_\mu s_1 \text{ and } s_1^\sharp \hookrightarrow^*_{\mathcal{R},\mu} \sigma(\theta(u)) \quad \text{if } u_1 \to v_1 \in \mathcal{P}_{\mathcal{X}}$$

Finally, we also have

$$\sigma(v') \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2).$$

Since the variables in the pairs are pairwise disjoint, we may extend σ to behave like $\sigma(\theta(x))$ on $x \in \mathcal{V}ar(u)$ then $\sigma(u) = \sigma(\theta(u))$ and therefore

$$\begin{array}{rl} \sigma(v_1) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) & \text{if } u_1 \to v_1 \in \mathcal{P}_{\mathcal{G}}, \text{ and} \\ \sigma(v_1) \trianglerighteq_\mu s_1 \text{ and } s_1^\sharp \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u) & \text{if } u_1 \to v_1 \in \mathcal{P}_{\mathcal{X}} \end{array}$$

Moreover, by definition of μ-narrowing, we have $\theta(v) \hookrightarrow_{\mathcal{R},\mu} v'$. This implies that $\sigma(\theta(v)) \hookrightarrow_{\mathcal{R},\mu} \sigma(v')$, and since $\sigma(v) = \sigma(\theta(v))$, we obtain

$$\sigma(v) \hookrightarrow_{\mathcal{R},\mu} \sigma(v') \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_2).$$

Since $\mathcal{R}$ is μ-terminating, $\sigma(v)$ is $(\mathcal{R}, \mu)$-terminating. Hence, "$\ldots, u_1 \to v_1, u \to v, u_2 \to v_2, \ldots$" is a minimal infinite $(\mathcal{P}, \mathcal{R}, \mu)$-chain as well. □

Example 28. *Since the right-hand side of pair* (36) *in Example 26 does not unify with any (renamed) left-hand side of a CSDP (including itself) and it can be μ-narrowed at position 1 (notice that* $\mu(\mathtt{f})=\{1\}$*) by using the rule* $\mathtt{p}(\mathtt{s}(x)) \to x$*, we can replace it by its μ-narrowed pair:*

$$\mathtt{F}(\mathtt{s}(\mathtt{0})) \to \mathtt{F}(\mathtt{0}) \tag{38}$$

Now, $\mathsf{Proc}_{SCC}(\{(38)\}, \mathcal{R}, \mu) = \varnothing$ *and the μ-termination of* $\mathcal{R}$ *is proved.*

The following example shows that strong conservativeness cannot be dropped for the pair $u \to v$ to be μ-narrowed. This requirement was not taken into account in [AGL07, Theorem 5.3].

Example 29. *Consider the following*[8] *TRS* $\mathcal{R}$*:*

$$\begin{array}{rcl} \mathtt{c}(\mathtt{e}(x)) & \to & \mathtt{d}(x, x) \\ \mathtt{a} & \to & \mathtt{e}(\mathtt{a}) \end{array}$$

---

[8] We thank Fabian Emmes for providing this example.

and $\mathcal{P}$ consisting of the following pair:

$$\mathtt{F}(\mathtt{d}(x,x)) \to \mathtt{F}(\mathtt{c}(x))$$

together with $\mu(\mathtt{c}) = \mu(\mathtt{d}) = \mu(\mathtt{F}) = \{1\}$ *and* $\mu(\mathtt{e}) = \varnothing$*. There is an infinite* $(\mathcal{P},\mathcal{R},\mu)$*-chain:*

$$\mathtt{F}(\mathtt{c}(\underline{\mathtt{a}})) \hookrightarrow_{\mathcal{R},\mu} \mathtt{F}(\underline{\mathtt{c}(\mathtt{e}(\mathtt{a}))}) \hookrightarrow_{\mathcal{R},\mu} \underline{\mathtt{F}(\mathtt{d}(\mathtt{a},\mathtt{a}))} \hookrightarrow_{\mathcal{P},\mu} \mathtt{F}(\mathtt{c}(\underline{\mathtt{a}})) \hookrightarrow_{\mathcal{R},\mu} \cdots$$

Since $\mathtt{F}(\mathtt{c}(x))$ *does not unify with any left-hand side of another pair, we can* μ*-narrow the pair in* $\mathcal{P}$*. We obtain* $\mathcal{P}'$ *consisting of the* μ*-narrowed pair*

$$\mathtt{F}(\mathtt{d}(\mathtt{e}(x),\mathtt{e}(x))) \to \mathtt{F}(\mathtt{d}(x,x))$$

No infinite $(\mathcal{P}',\mathcal{R},\mu)$*-chain is possible now. Note that* $\mathcal{P}$ *is* μ*-conservative, but it is* not *strongly* μ*-conservative (the variable* x *is both* μ*-replacing and non-*μ*-replacing in* $\mathtt{F}(\mathtt{d}(x,x))$*).*

Remark 14 (Implementing the narrowing processor). *In our current implementation, we apply the narrowing processor only if, after computing the (one-step)* μ*-narrowings of the right-hand side* v *of a pair* $u \to v \in \mathcal{P}$*, the new CS-dependency graph does not increase the number of arcs. More sophisticated strategies like (the corresponding adaptations of) the* safe transformations *in [GTSF06, Definition 33] could be considered in the future.*

13. Experiments

The processors described in the previous sections were implemented as part of the tool MU-TERM. We tested the CSDP-framework in practice on the 90 examples in the Context-Sensitive Rewriting subcategory of the 2007 International Termination Competition:

`http://www.lri.fr/~marche/termination-competition/2007/`

These 90 examples are part of the Termination Problem Data Base (TPDB, version 4.0):

`http://www.lri.fr/~marche/tpdb/`

We addressed this task in three different ways:

1. We compared CSDPs with previously existing techniques for proving termination of *CSR*.
2. We compared the improvements introduced by the different CS processors which have been defined in this paper.
3. We participated in the *CSR* subcategory of the 2007 Termination Competition.

In the following subsections, we provide more details about this experimental evaluation.

13.1. CSDPs vs. other techniques for proving termination of CSR

Several methods have been developed to prove termination of *CSR* for a given CS-TRS $(\mathcal{R},\mu)$. Two main approaches have been investigated so far:

1. *Direct proofs*, which are based on using μ-reduction orderings (see [Zan97]) such as the (context-sensitive) recursive path orderings [BLR02] and polynomial orderings [GL02, Luc04b, Luc05]. These are orderings $>$ on terms that can be used to directly compare the left- and right-hand sides of the rules in order to conclude the μ-termination of the TRS.

2. *Indirect proofs*, which obtain a proof of the μ-termination of $\mathcal{R}$ as a proof of termination of a transformed TRS $\mathcal{R}^\mu_\Theta$ (where Θ represents the transformation). If we are able to prove *termination* of $\mathcal{R}^\mu_\Theta$ (using the standard methods), then the μ-termination of $\mathcal{R}$ is ensured.

We used MU-TERM to compare all these techniques with respect to the aforementioned benchmark examples. The results of this comparison are summarized in Table 1.

Remark 15. *A number of transformations Θ from TRSs $\mathcal{R}$ and replacement maps μ that produce TRSs $\mathcal{R}^\mu_\Theta$ have been investigated by Lucas (transformation L [Luc96]), Zantema (transformation Z [Zan97]), Ferreira and Ribeiro (transformation FR [FR99]), and Giesl and Middeldorp (transformations[9] GM, sGM, and C [GM99, GM04]), see [GM04, Luc06] for recent surveys about these transformations which also include a thorough analysis about their relative power. All these transformations were considered in our experiments, so the item* "Transformations" *in Table 1 concentrates the* joint *impact of all of them.*

Tool Version	Proved	Total Time	Average Time
CSDPs	65/90	0.31 sec.	0.00 sec.
CSRPO	37/90	0.21 sec.	0.00 sec.
Polynomial Orderings	27/90	0.06 sec.	0.00 sec.
Transformations	56/90	5.59 sec.	0.10 sec.

Table 1: Comparison among *CSR* Termination Techniques

From the benchmarks summarized in Table 1, we clearly conclude that the CSDP-framework is the most powerful technique for proving termination of *CSR*. Actually, all the examples that were solved by using CSRPO or polynomial orderings were also solved using CSDPs. With regard to transformations, there is only one example (namely, `Ex9_Luc06`, which can be solved by using transformation GM) that could not be solved with our current implementation.

Example 30. *The following nonterminating TRS $\mathcal{R}$ can be used to compute the list of prime numbers by using the well-known Erathostenes sieve[10] [GM99]:*

$$\begin{array}{rcl}
\mathtt{primes} & \to & \mathtt{sieve}(\mathtt{from}(\mathtt{s}(\mathtt{s}(\mathtt{0})))) \\
\mathtt{from}(x) & \to & \mathtt{cons}(x, \mathtt{from}(\mathtt{s}(x))) \\
\mathtt{head}(\mathtt{cons}(x, y)) & \to & x \\
\mathtt{sieve}(\mathtt{cons}(x, y)) & \to & \mathtt{cons}(x, \mathtt{filt}(x, \mathtt{sieve}(y))) \\
\mathtt{tail}(\mathtt{cons}(x, y)) & \to & y \\
\mathtt{if}(\mathtt{true}, x, y) & \to & x \\
\mathtt{if}(\mathtt{false}, x, y) & \to & y \\
\mathtt{filt}(\mathtt{s}(\mathtt{s}(x)), \mathtt{cons}(y, z)) & \to & \mathtt{if}(\mathtt{div}(\mathtt{s}(\mathtt{s}(x)), y), \mathtt{filt}(\mathtt{s}(\mathtt{s}(x)), z), \mathtt{cons}(y, \mathtt{filt}(\mathtt{s}(\mathtt{s}(x)), z)))
\end{array}$$

Consider the replacement map μ for the signature $\mathcal{F}$ given by:

$$\mu(\mathtt{cons}) = \mu(\mathtt{if}) = \{1\} \textit{ and } \mu(f) = \{1, \ldots, ar(f)\} \textit{ for all } f \in \mathcal{F} - \{\mathtt{cons}, \mathtt{if}\}.$$

---

[9] The labels for these transformations correspond to the ones introduced in [Luc06].

[10] Without appropriate rules for defining symbol `div`, the TRS has no complete computational meaning. However, we take it here as given in [GM99] for the purpose of comparing different techniques for proving termination of *CSR* by transformation.

From the termination point of view, this example is interesting because, since its introduction in Giesl and Middeldorp's paper [GM99], no automatic proof of termination has been reported. In sharp contrast, termination of CSR *for this TRS and replacement map μ is easily proved by using the techniques developed in this paper. In particular, the context-sensitive dependency graph contains no cycle.*

Tool Version	Narrowing	Non-μ-Replacing Projection	Subterm	Proved	Total Time	Average Time
1.	No	No	No	54/90	3.00 sec.	0.05 sec.
2.	No	No	Yes	62/90	0.55 sec.	0.01 sec.
3.	No	Yes	No	57/90	0.82 sec.	0.01 sec.
4.	No	Yes	Yes	65/90	0.49 sec.	0.01 sec.
5.	Yes	No	No	54/90	3.22 sec.	0.06 sec.
6.	Yes	No	Yes	62/90	2.64 sec.	0.04 sec.
7.	Yes	Yes	No	57/90	1.27 sec.	0.02 sec.
8.	Yes	Yes	Yes	65/90	0.31 sec.	0.00 sec.

Table 2: Comparison among CS processors

13.2. Contribution of the different CS processors

In our implementation of the CSDP-framework, besides processor Proc_{SCC}, the subterm processors in Section 11 and the μ-reduction-pair CS processors in Section 10 are the most frequently used (in this order). The impact of the CS processors in Sections 11 and 12 is summarized in Table 2. Our benchmarks show that the CS processors described in Section 11 play an important role in our proofs. The subterm processors $\mathsf{Proc}_{subNColl}$ and $\mathsf{Proc}_{subColl}$ are quite efficient, but the ones that are based on simple projections for non-μ-replacing arguments (Proc_{NRP} and Proc_{NRP2}) also increase the power and the speed of the CSDP technique. Furthermore, these two groups of CS processors are complementary: the extra problems that are specifically solved by them are different. Narrowing is useful for simplifying the graph, but it doesn't play an important role in the benchmarks because it is only applied to solve two examples (which can be solved without narrowing as well). Furthermore, it must be used carefully because recomputing the graph can be expensive in that case. Complete details of our experiments can be found here:

`http://zenon.dsic.upv.es/muterm/benchmarks/csdp/`

13.3. CSDPs at the 2007 International Termination Competition

In 2007, AProVE [GST06] was the only tool (besides mu-term) implementing specific methods for proving termination of *CSR*. Both AProVE and mu-term participated in the *CSR* subcategory of the 2007 International Termination Competition. AProVE participated with a termination expert for *CSR* which, given a CS-TRS $(\mathcal{R},\mu)$, successively tries different transformations Θ for proving termination of *CSR* (which are enumerated in Remark 15, i.e., $\Theta \in \{C, FR, GM, L, sGM, Z\}$). It then uses a huge variety of different and complementary techniques to prove termination of rewriting (according to the DP-framework) on the obtained TRS $\mathcal{R}^{\mu}_{\Theta}$. Actually, AProVE is currently the most powerful tool for proving termination of TRSs and implements most existing results and techniques regarding DPs and related techniques.

However, mu-term's implementation of CSDPs was able to beat AProVE in the *CSR* category (mu-term was able to prove 68 of the 90 examples; AProVE proved 64), thus demonstrating that CSDPs are actually a very powerful technique for proving termination of *CSR*.

14. Related work

The first presentation of the context-sensitive dependency pairs was given in [AGL06]. This paper is an extended and revised version of [AGL06, AGL07]. We provide complete proofs for all results[11], and also present many examples about the use of the theory. The main conceptual differences between [AGL06, AGL07] and this paper are the following:

1. In this paper, we have investigated two different notions of minimal non-μ-terminating terms: the so-called *strongly minimal terms* ($\mathcal{T}_{\infty,\mu}$, which are introduced in this paper) and the *minimal terms* ($\mathcal{M}_{\infty,\mu}$), which were introduced in [AGL06] and further investigated in [AGL07]. The combined use of these notions leads to a better development of the theory. This has brought new essential results, remarkably Theorem 1, which is the basis (at the level of pure context-sensitive rewriting) of the new notions of CSDP and minimal chain.
2. Although most of the ideas in this first part of the paper (Section 3) were present in [AGL07, Section 3], we make some aspects explicit that were only implicit there. For instance, the essential notion of *hidden term* (a consequence of Lemma 5 which is further developed in Lemma 6 and Proposition 4) was implicit in [AGL07, Section 3], but only the notion of *hidden symbol* was made explicit. Actually, the proofs of the aforementioned results in this paper correspond (with minor changes) to those of Lemma 3.4, Lemma 3.5, and Proposition 3.6 in [AGL07], respectively.
3. The notion of context-sensitive dependency pairs was first introduced in [AGL06, Definition 1], but the narrowing condition that we have now included for the noncollapsing CSDPs is new. This condition is inspired in the recent extension of the DP-method to Order-Sorted TRSs [LM08]. In this paper, we have elaborated it in depth to show that it is actually a natural requirement (see Section 3.4). In [LM08], it has already been shown that including 'narrowability' in the usual definition of dependency pair can be useful to automatically prove termination of rewriting. Similar considerations are valid for CSR.
4. In [AGL06], a notion of minimal chain was introduced but not really used in the main results. Actually, the notion of minimal chain in this paper is completely different from the old one and is a consequence of the analysis of infinite μ-rewrite sequences developed in Section 3. Furthermore, in this paper, the notion of minimal chain of pairs is essential for the definition of the context-sensitive dependency graph and the development of the CSDP-framework in Section 7.
5. The notion of context-sensitive dependency graph was first introduced in [AGL06] and further refined in [AGL07] thanks to the introduction of the hidden symbols. The definition in this paper introduces a new refinement through the notion of 'narrowable hidden term' and shows a nice symmetry between the arcs associated to noncollapsing and collapsing pairs. Furthermore, the new definition leads to a great simplification of the computed graph: for the CS-TRS in Example 1, compare the graph in Figure 6 (corresponding to [AGL06]) with the new graph in Figure 5.
6. The estimation of the CSDG in [AGL06, AGL07] was an adaptation of the one by Arts and Giesl [AG00] to the context-sensitive setting. In this paper, we have defined a new estimation of the CSDG on the basis of the most recent proposal by Giesl et al. [GTS05].

---

[11] We report and fix some bugs in previous papers.

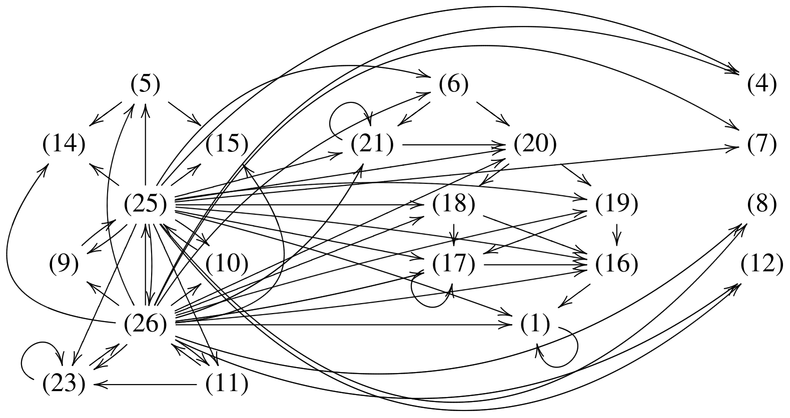

Figure 6: Context-Sensitive Dependency Graph of Example 1 following [AGL06]

7. The definition of a CSDP-framework for the mechanization of proofs of termination of *CSR* using CSDPs is new. A number of processors introduced here had a kind of counterpart in [AGL06] (for instance, the use of μ-reduction orderings was formalized in [AGL06, Theorem 4] and the subterm criterion for noncollapsing pairs was formalized in [AGL06, Theorem 5]) or in [AGL07] (for instance, the narrowing transformation in [AGL07, Theorem 5.3]), but they were not formulated as processors.
8. This paper introduces a number of new processors that can be used for proving termination of *CSR*: the SCC processor[12]; the processors for filtering or transforming collapsing pairs (see Section 9), the use of argument filterings[13], the use of the subterm criterion with collapsing pairs (Theorem 13), etc.
9. Finally, for the first time, we have considered how to *disprove* termination of *CSR* within the CSDP framework (processor $\mathsf{Proc}_{\mathit{Inf}}$ in Theorem 5).

14.1. CSDPs vs. DPs and a piece of history

The first attempt to develop a theory of dependency pairs for *CSR* started more than ten years ago when the third author of this paper asked Thomas Arts (who was preparing the first presentation of the dependency pair method [Art97]) about the possibility of extending the dependency pair approach to *CSR*. Arts immediately noticed that the main problem of extending the existing results for ordinary rewriting to *CSR* was the possibility of having variables that are not replacing in the left-hand sides of the rules but that become replacing in the corresponding right-hand side. This is what we now call *migrating* variables. After this first failed attempt, the focus moved to transformations of CS-TRSs $(\mathcal{R},\mu)$ into ordinary TRSs $\mathcal{R}^\mu_\Theta$ (where Θ represents the transformation) in such a way that termination of $\mathcal{R}^\mu_\Theta$ implies the μ-termination of $\mathcal{R}$ [GM04, Luc06].

During the spring of 2006, MU-TERM was being revised in preparation for its participation in the 2006 International Termination Competition, which was organized by Claude Marché. The idea of adapting DPs to *CSR* came up again. A first correct version of context-sensitive dependency pairs that did *not* at the time consider collapsing pairs was the following:

---

[12] This is mentioned in [AGL06, Section 4.2] but without any formal description.

[13] This was briefly mentioned at the end of [AGL06, Section 4.2] but was never formalized.

Definition 15 (First preliminary version of CSDPs). *Let $\mathcal{R} = (\mathcal{F}, R) = (C \uplus \mathcal{D}, R)$ be a TRS and $\mu \in M_{\mathcal{R}}$. Let*

$$\begin{array}{rcl}\mathsf{DP}_1(\mathcal{R},\mu) & = & \{l^\sharp \to s^\sharp \mid l \to r \in R, r \trianglerighteq_\mu s, root(s) \in \mathcal{D}, l \ntriangleright_\mu s\} \\ & \cup & \{l^\sharp \to \mathtt{MUSUBTERM}(x) \mid l \to r \in R, x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)\} \\ & \cup & \{\mathtt{MUSUBTERM}(f(x_1,\ldots,x_k)) \to \mathtt{MUSUBTERM}(x_i) \mid f \in \mathcal{F}, i \in \mu(f)\} \\ & \cup & \{\mathtt{MUSUBTERM}(f(x_1,\ldots,x_k)) \to f^\sharp(x_1,\ldots,x_k) \mid f \in \mathcal{D}\}\end{array}$$

with $\mu^\sharp(f) = \mu(f)$ if $f \in \mathcal{F}$, $\mu^\sharp(f^\sharp) = \mu(f)$ if $f \in \mathcal{D}$, and $\mu^\sharp(\mathtt{MUSUBTERM}) = \varnothing$.

We handle migrating variables x by enclosing them inside a term `MUSUBTERM(`x`)` which (after instantiating x by means of a substitution σ) would be able to start the search for a μ-replacing subterm $s = f(s_1, \ldots, s_k)$ which (after marking its root symbol f as $f^\sharp$) is able to connect with the left-hand side of the next CSDP in a sequence. The notion of *chain* of CSDPs that was used here was essentially the standard one. All pairs were treated in the very same way and the only difference was that pairs were connected by using *CSR* instead of ordinary rewriting.

The implementation of the CSDPs in Definition 15 did not work very well in practice. The structure of pairs which dealt with migrating variables introduced many arcs in the corresponding graph and, therefore, many cycles. Thus, the following proposal was considered instead.

Definition 16 (Second preliminary version of CSDPs). *Let $\mathcal{R} = (\mathcal{F}, R) = (C \uplus \mathcal{D}, R)$ be a TRS and $\mu \in M_{\mathcal{R}}$. Let $\mathsf{DP}_2(\mathcal{R},\mu) = \mathsf{DP}_{2,\mathcal{F}}(\mathcal{R},\mu) \cup \mathsf{DP}_{2,\mathcal{X}}(\mathcal{R},\mu)$ where:*

$$\begin{array}{rcl}\mathsf{DP}_{2,\mathcal{F}}(\mathcal{R},\mu) & = & \{l^\sharp \to s^\sharp \mid l \to r \in R, r \trianglerighteq_\mu s, root(s) \in \mathcal{D}, l \ntriangleright_\mu s\} \\ \mathsf{DP}_{2,\mathcal{X}}(\mathcal{R},\mu) & = & \{l^\sharp \to U_{l,f,x}(x) \mid l \to r \in R, f \in \mathcal{D}, x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)\} \\ & \cup & \{U_{l,f,x}(f(x_1,\ldots,x_k)) \to f^\sharp(x_1,\ldots,x_k) \\ & & \qquad \mid l \to r \in R, f \in \mathcal{D}, x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(l)\}\end{array}$$

and $\mu^\sharp(f) = \mu(f)$ if $f \in \mathcal{F}$, $\mu^\sharp(f^\sharp) = \mu(f)$ if $f \in \mathcal{D}$, and $\mu^\sharp(U_{l,f,x}) = \{1\}$ for all rules $l \to r$, symbols f, and variables x originating one of these symbols.

Here, migrating variables x are enclosed inside a term $U_{l,f,x}(x)$ which (after instantiating x by means of a substitution σ) would be able to connect any μ-replacing subterm $s = f(s_1, \ldots, s_k)$ (with f a defined symbol) with the left-hand side of the next CSDP in a sequence. Note that no explicit μ-replacing subterm search is possible with this new definition of CSDP. Instead, this requirement *was moved to the definition of chain*. Now, although these dependency pairs still remain as the 'traditional ones', *a clear distinction was made between two kinds of CSDPs*: those that were obtained from the nonvariable parts of the right-hand sides of the rules ($\mathsf{DP}_{2,\mathcal{F}}(\mathcal{R},\mu)$ in Definition 16) and those that were introduced to treat the migrating variables ($\mathsf{DP}_{2,\mathcal{X}}(\mathcal{R},\mu)$ in Definition 16). Both kinds of CSDPs were clearly distinguished in the new definition of chain and the μ-subterm requirement was used to describe how chains of such CSDPs are built.

A version of MU-TERM that implemented the CSDPs in Definition 16 was submitted for participation in the *Context-Sensitive* (sub)category of the 2006 International Termination Competition (June 2006). We are grateful to Claude Marché for providing a copy of the folder where the MU-TERM outcome was stored. It is now available at the following URL:

`zenon.dsic.upv.es/muterm/benchmarks/ic10/muterm-2006/benchmarks.html`

A further evolution led to the definition of CSDP which was finally published in [AGL06]. In sharp contrast to the standard dependency pair approach, where all dependency pairs have tuple symbols $f^\sharp$ both in the left- and right-hand sides, we finally took the definitive step to also consider *collapsing* pairs having a single *variable* in the right-hand side as the most elegant, concise and expressive way to reflect the effect of the *migrating* variables in the termination behavior of *CSR*. This is one of the most important and original contributions of the paper.

15. CSDPs vs. noncollapsing CSDPs

In [AEFG$^+$08], a transformation of collapsing pairs into 'ordinary' (i.e., noncollapsing) pairs is introduced. The transformation uses the following notion.

Definition 17 (Hiding Context [AEFG$^+$08, Definition 7]**).** *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. The function symbol f* hides the argument *i if there is a rule $l \to r \in \mathcal{R}$ with $r \rhd_{\not\mu} f(r_1, \ldots, r_i, \ldots, r_n)$, $i \in \mu(f)$, and r_i contains a defined symbol or a variable at an active position. A context C is* hiding *iff $C = \Box$ or C has the form $f(t_1, \ldots, t_{i-1}, C', t_{i+1}, \ldots, t_n)$ where f hides the argument i and C' is a hiding context.*

The notion of *CSDPs* that is given in [AEFG$^+$08] is the following:

Definition 18. [AEFG$^+$08, Definition 9] *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. If $\mathsf{DP}_{\mathcal{X}}(\mathcal{R},\mu) \neq \varnothing$, we introduce a fresh* unhiding tuple symbol $\mathtt{U}$ *and the following* unhiding DPs*:*

- *$s \to \mathtt{U}(x)$ for every $s \to x \in \mathsf{DP}_{\mathcal{X}}(\mathcal{R},\mu)$,*
- *$\mathtt{U}(f(x_1, \ldots, x_i, \ldots, x_n)) \to \mathtt{U}(x_i)$ for every function symbol f of any arity n and every $1 \leq i \leq n$ where f hides position i, and*
- *$\mathtt{U}(t) \to t^\sharp$ for every hidden term t.*

Let $\mathsf{DP}_u(\mathcal{R},\mu)$ be the set of all unhiding DPs (where $\mathsf{DP}_u(\mathcal{R},\mu) = \varnothing$ whenever $\mathsf{DP}_{\mathcal{X}}(\mathcal{R},\mu) = \varnothing$). Then $\mathsf{DP}'(\mathcal{R},\mu) = \mathsf{DP}_{\mathcal{F}}(\mathcal{R},\mu) \cup \mathsf{DP}_u(\mathcal{R},\mu)$.

The corresponding definition of *chain* is, essentially, the standard one [AG00], but μ-rewriting (with $\mathcal{R}$) is used for connecting pairs.

Definition 19. [AEFG$^+$08, Definition 11] *Let $\mathcal{P}$ and $\mathcal{R}$ be TRSs and let μ be a replacement map. We extend μ to tuple symbols by defining $\mu(f^\sharp) = \mu(f)$ for all $f \in \mathcal{D}$ and $\mu(\mathtt{U}) = \varnothing$. A sequence of pairs $u_1 \to v_1, u_2 \to v_2, \ldots$ from $\mathcal{P}$ is a $(\mathcal{P},\mathcal{R},\mu)$-*chain *if there is a substitution σ with $\sigma(v_i) \hookrightarrow^*_{\mathcal{R},\mu} \sigma(u_{i+1})$ and $\sigma(v_i)$ is $(\mathcal{R},\mu)$-terminating for all i.*

Using these definitions, a characterization of termination of *CSR* is given.

Theorem 17. [AEFG$^+$08, Theorem 12] *A TRS $\mathcal{R}$ is μ-terminating if and only if there is no infinite $(\mathsf{DP}'(\mathcal{R},\mu), \mathcal{R}, \mu)$-chain.*

On the basis of these definitions and results, [AEFG$^+$08, Section 4] develops a CSDP framework.

15.1. Comparing CSDPs and noncollapsing CSDPs

As discussed in Section 14.1, the idea of providing a definition of CSDPs that does not use collapsing pairs cannot be considered as the main contribution of [AEFG^{+}08]: in 2006 there was an implementation of CSDPs without collapsing pairs (namely the one which corresponds to Definition 16). Actually, Definition 18 is very close to Definition 15 (i.e., the first correct notion of CSDP developed in 2006) if we write `U` instead of `MUSUBTERM` in Definition 15. The crucial differences between Definition 15 and Definition 18 are the use of *hiding contexts* (Definition 17) and the use of *hidden terms* (Definition 3). As discussed in [AEFG^{+}08, Section 3], the notion of hiding context is a refinement of the notion of hidden term described in this paper (and previously approached in [AGL07]), see Section 3.2.

Indeed, the notion of hiding context is the most important contribution of [AEFG^{+}08] from the theoretical side. The notion of hiding context can be easily integrated in the CSDP framework discussed in this paper. This has been carried out in [Gut08, GL09a, GL09b], where an extension of our CSDP framework was developed to appropriately integrate this notion. Within this new approach, Definition 18 could be incorporated to the CSDP framework by using the following modified version of Theorem 8, which defines the appropriate *CS processor*.

Theorem 18. *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{P} = (\mathcal{G}, P)$ be TRSs and $\mu \in M_{\mathcal{F}\cup\mathcal{G}}$. Let $u \to x \in \mathcal{P}_{\mathcal{X}}$ and*

$$\begin{array}{rcl} P_u & = & \{u \to U(x)\} \\ & & \cup\, \{U(f(x_1,\dots,x_k)) \to U(x_i) \mid f \in \mathcal{F}, 1 \leq i \leq ar(f) \text{ and } f \text{ hides } i\} \\ & & \cup\, \{U(t) \to t^{\sharp} \mid t \in \mathcal{NHT}_{\mathcal{P}}\} \end{array}$$

where U is a fresh symbol. Let $\mathcal{P}' = (\mathcal{G} \cup \{U\}, P')$, where $P' = (P - \{u \to x\}) \cup P_u$, and μ' which extends μ by $\mu'(U) = \varnothing$. Then, the processor Proc_{hCtx} given by

$$\mathsf{Proc}_{hCtx}(\mathcal{P},\mathcal{R},\mu) = \{(\mathcal{P}',\mathcal{R},\mu')\}$$

is sound and complete.

The proof of this result would be analogous to the one for Theorem 8 with the proviso, in our definition of chain of pairs (Definition 5), that the contexts $C_i[\,]_{p_i}$ which are used for handling collapsing pairs are now *hiding contexts*. In contrast to processor Proc_{eColl} in Theorem 8, Proc_{hCtx} has the advantage of introducing fewer rules due to the use of the notion of *hiding* in Definition 17. Obviously, this could lead to simpler proofs when it is used.

In this paper, we have shown that collapsing pairs are an essential part of the theoretical description of termination of *CSR*. Actually, Definition 18 *explicitly* uses them to introduce the new unhiding pairs. This shows that the most basic notion when *modeling* the termination behavior of *CSR* is that of collapsing pair and that unhiding pairs should be better considered as an ingredient for handling collapsing pairs in proofs of termination (as implemented by processor Proc_{hCtx} above).

15.2. Use of CSDPs and noncollapsing CSDPs

The application of Definition 18 at the very beginning of the termination analysis of CS-TRSs (as done in [AEFG^{+}08]) often leads to obtaining a more complex dependency graph. For

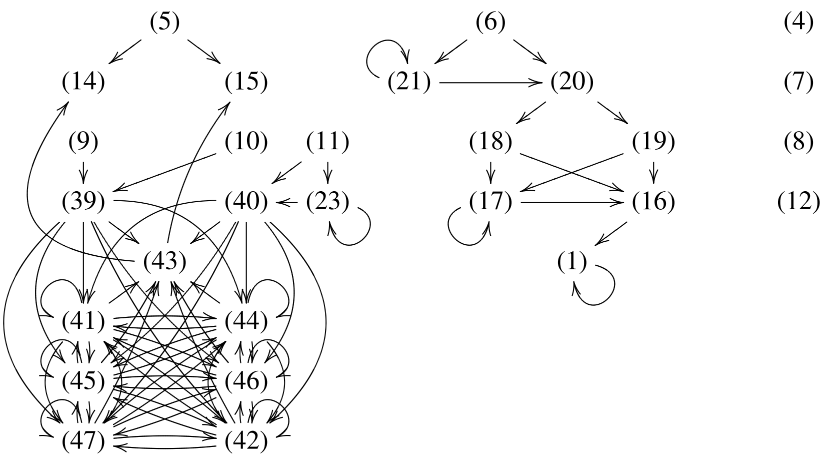

Figure 7: Context-Sensitive Dependency Graph of Example 1 according to [AEFG$^+$08]

instance, we would *replace* the collapsing CSDPs (25) and (26) by the following ones:

$$\begin{array}{rcll}
\mathtt{TAIL}(\mathtt{cons}(x,xs)) & \to & \mathtt{U}(xs) & (39)\\
\mathtt{TAKE}(\mathtt{s}(n),\mathtt{cons}(x,xs)) & \to & \mathtt{U}(xs) & (40)\\
\mathtt{U}(\mathtt{incr}(x)) & \to & \mathtt{U}(x) & (41)\\
\mathtt{U}(\mathtt{incr}(\mathtt{oddNs})) & \to & \mathtt{INCR}(\mathtt{oddNs}) & (42)\\
\mathtt{U}(\mathtt{oddNs}) & \to & \mathtt{ODDNS} & (43)\\
\mathtt{U}(\mathtt{rep2}(x)) & \to & \mathtt{U}(x) & (44)\\
\mathtt{U}(\mathtt{zip}(x,y)) & \to & \mathtt{U}(x) & (45)\\
\mathtt{U}(\mathtt{zip}(x,y)) & \to & \mathtt{U}(y) & (46)\\
\mathtt{U}(\mathtt{cons}(x,y)) & \to & \mathtt{U}(x) & (47)
\end{array}$$

to obtain the graph in Figure 7, which should be compared with the CSDG for the same example in Figure 5. On the other hand, if $\mathcal{P}$ contains no collapsing pairs (as happens if Definition 18 is used to compute the dependency pairs of a CS-TRS), then Definition 19 is subsumed by our notion of chain of pairs (Definition 5). This means that, after using processor Proc_{eColl} in Theorem 8 to remove collapsing pairs in the component $\mathcal{P}$ of a CS problem $(\mathcal{P},\mathcal{R},\mu)$, *we could use all CS processors developed in [AEFG$^+$08]*, some of which have not been discussed in our paper (for instance, the *instantiation processor* [AEFG$^+$08, Theorem 24]). Also, the CS processors that are developed here can be used in any implementation following [AEFG$^+$08].

Remark 16. *Note that, although the definition of chain in [AEFG$^+$08] (see Definition 19) is apparently closer to the standard one [GTSF06, Definition 3], this does* not *mean that we can use or easily 'translate' existing DP-processors (see [GTSF06]) to be used with* CSR.

The narrowing processor provides a striking example. Example 29 shows that the application of the narrowing processor to the TRSs $\mathcal{P}$ and $\mathcal{R}$ in the example is not *correct due to the lack* *of* strong μ-conservativeness *of the μ-narrowed pair in $\mathcal{P}$. Since $\mathcal{P}$ has no collapsing pair, one could think (following a naïve interpretation of [AEFG⁺08]) that the narrowing processor of the DP-framework (see [GTSF06, Theorem 31]), which does not take into account the replacement restrictions, should work with* CSR *without difficulties, which is not the case.*

Thus, a CSDP framework that is based on Definitions 18 and 19 does not boil down to the DP-framework*, and a careful consideration of the replacement restrictions is necessary before being able to use any DP-processor with* CSR.

15.3. Experimental evaluation

We have performed an experimental evaluation of the use of CSDPs vs. the ones in Definition 18 as follows: we prepared two versions of MU-TERM: MU-TERM-LPAR08 and MU-TERM-IC. The tool MU-TERM-LPAR08 first applies Theorem 18 to remove all collapsing pairs (as one would do when working within the approach described in [AEFG$^+$08]) and then uses the CS processors described in both this paper and in [AEFG$^+$08] to achieve termination proofs. On the other hand, MU-TERM-IC implements the CSDP framework that we have described here (with the modifications developed in [Gut08, GL09a, GL09b]).

On a collection of 109 examples, both tools succeeded on the very same ones (94 proofs of termination). However, MU-TERM-IC performed globally faster. Furthermore, we *did not need to use* Proc_{hCtx} in the proofs with MU-TERM-IC. This suggests that (in contrast to what we claimed in [AEFG$^+$08] when the integration of the notion of hiding context into the CSDP framework was pending), collapsing pairs do not represent any drawback for automatically proving termination of *CSR*. Detailed benchmarks are at the following URL:

`zenon.dsic.upv.es/muterm/benchmarks/ic10/muterm-2009/benchmarks.html`

Table 3 shows the use of the different processors in these benchmarks. The interpretation of the frequency of use for the different processors should take into account the following *strategy* for invoking them in MU-TERM-IC when CS problems are treated: first, we try the basic (infinite and finite) processors. If some of them succeed, we are done; otherwise, we continue as follows:

Processors	Applied in
SCC Processor (Section 8)	94/94 problems
Processors based on subterm criteria (Section 11)	56/94 problems
Processors based on reduction pairs (Section 10)	50/94 problems
Basic Processors (Section 7)	28/94 problems
Narrowing processor (Section 12)	3/94 problems

Table 3: Summary of processors used in MU-TERM-IC

1. SCC processor.
2. Subterm criterion processors.
3. Reduction pair (RP) processors with polynomial and matrix interpretations over the reals [ALN09a, ALN09b, Luc05, Luc07].
4. Narrowing processor.

Interestingly, *all* processors are used at least once during the proofs.

16. Conclusions

We have analyzed the structure of infinite context-sensitive rewrite sequences starting from minimal non-μ-terminating terms (Theorem 1). This knowledge is used to provide an appropriate definition of context-sensitive dependency pair (Definition 4), and the related notion of chain (Definition 5). In sharp contrast to the standard dependency pair approach, where all dependency pairs have tuple symbols $f^\sharp$ in both the left- and right-hand sides, we have *collapsing* dependency pairs that have a single *variable* in the right-hand side. These variables reflect the effect of the *migrating* variables on the termination behavior of *CSR*. At the level of *minimal chains*, however, the contrast with the ordinary DP approach is somehow recovered by a nice symmetry arising from the central notion of *hidden term* (Definition 3): a noncollapsing pair $u \to v$ is followed by a pair $u' \to v'$ if $\sigma(v)$ μ-rewrites into $\sigma(u')$ for some substitution σ; a collapsing pair $u \to v$ is followed by a pair $u' \to v'$ if there is a *hidden term* t such that $\sigma(t)^\sharp$ μ-rewrites into $\sigma(u')$ for some substitution σ. We have shown how to use the context-sensitive dependency pairs in proofs of termination of *CSR*. As in Arts and Giesl's approach, the absence of infinite minimal chains of dependency pairs from $\mathsf{DP}(\mathcal{R},\mu)$ characterizes the μ-termination of $\mathcal{R}$ (Theorems 2 and 3).

We have provided a suitable adaptation of the *dependency pair framework* to *CSR* by defining appropriate notions of *CS problem* (Definition 6) and *CS processor* (Definition 7). We have described a number of sound and (most of them) complete CS processors that can be used in any practical implementation of the CSDP-framework. In particular, we have introduced the notion of (estimated) *context-sensitive (dependency) graph* (Definitions 8 and 10) and the associated CS processor (Theorem 6). We have also described some CS processors for removing or transforming collapsing pairs from CS problems (Theorems 7 and 8). We are also able to use μ-reduction pairs (Definition 11) and argument filterings to ensure the absence of infinite chains of pairs (Theorems 9, 10, and 11). We have adapted Hirokawa and Middeldorp's *subterm criterion* which permits concluding the absence of infinite minimal chains by paying attention only to the pairs in the corresponding CS problem (Theorems 12 and 13). Following this appealing idea, we have also introduced two new processors that work in a similar way but use a very basic kind of ordering instead of the subterm relation (Theorems 14 and 15). Narrowing context-sensitive dependency pairs have also been investigated. It is helpful to simplify or restructure the dependency graph and eventually simplify the proof of termination (Theorem 16).

We have implemented these ideas as part of the termination tool MU-TERM [AGIL07, Luc04a]. The implementation and practical use of the developed techniques yield a novel and powerful framework that improves the current state-of-the-art of methods for proving termination of *CSR*. Actually, CSDPs were an essential ingredient for MU-TERM in winning the context-sensitive subcategory of the 2007 competition of termination tools.

For future work, we plan to extend the basic CSDP-framework described in this paper with further CS processors integrating the *usable rules* for *CSR* [GLU08] and proofs of termination of *innermost CSR* using CSDPs [AL07].

Acknowledgements

We thank Jürgen Giesl and his group of the RWTH Aachen (especially Fabian Emmes, Carsten Fuhs, Peter Schneider-Kamp, and René Thiemann) for many fruitful discussions about CSDPs. We also thank the anonymous referees for many useful remarks.

References

[AEFG$^+$08] B. Alarcón, F. Emmes, C. Fuhs, J. Giesl, R. Gutiérrez, S. Lucas, P. Schneider-Kamp and R. Thiemann, *Improving Context-Sensitive Dependency Pairs*, in: I. Cervesato, H. Veith and A. Voronkov, editors, *Proc. of 15th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, LPAR'08*, LNAI 5330:636–651, Springer-Verlag, Berlin, 2008.

[AEGL10] M. Alpuente, S. Escobar, B. Gramlich, and S. Lucas. On-Demand Strategy Annotations Revisited: An Improved On-Demand Evaluation Strategy *Theoretical Computer Science* 411(2):504-541, 2010.

[AG00] T. Arts and J. Giesl. Termination of Term Rewriting Using Dependency Pairs. *Theoretical Computer Science*, 236(1–2):133–178, 2000.

[AGIL07] B. Alarcón, R. Gutiérrez, J. Iborra, and S. Lucas. Proving Termination of Context-Sensitive Rewriting with MU-TERM. *Electronic Notes in Theoretical Computer Science*, 188:105–115, 2007.

[AGL06] B. Alarcón, R. Gutiérrez, and S. Lucas. Context-Sensitive Dependency Pairs. In S. Arun-Kumar and N. Garg, editors, *Proc. of XXVI Conference on Foundations of Software Technology and Theoretical Computer Science, FST&TCS'06*, LNCS 4337:297–308, Springer-Verlag, Berlin, 2006.

[AGL07] B. Alarcón, R. Gutiérrez, and S. Lucas. Improving the Context-Sensitive Dependency Graph. *Electronic Notes in Theoretical Computer Science*, 188:91–103, 2007.

[AL07] B. Alarcón and S. Lucas. Termination of Innermost Context-Sensitive Rewriting Using Dependency Pairs. In B. Konev and F. Wolter, editors, *Proc. of 6th International Symposium on Frontiers of Combining Systems, FroCoS'07*, LNAI 4720: 73–87, Springer-Verlag, Berlin, 2007.

[ALN09a] B. Alarcón, S. Lucas, R. Navarro-Marset. Proving Termination with Matrix Interpretations over the Reals. In A. Geser and J. Waldmann, editors *Proc. of 10th International Workshop on Termination, WST'09*. pages 12-15, June 2009.

[ALN09b] B. Alarcón, S. Lucas, R. Navarro-Marset. Using Matrix Interpretations over the Reals in Proofs of Termination. In F. Lucio, G. Moreno, and R. Peña, editors, *Proc. of the 9th Spanish Conference on Programming and Computer Languages, PROLE'09,* pages 255-264, Universidad del País Vasco, 2009.

[Art97] T. Arts. Automatically Proving Termination and Innermost Normalisation of Term Rewriting Systems. PhD Thesis, Universiteit Utrecht, 1997.

[BLR02] C. Borralleras, S. Lucas, and A. Rubio. Recursive Path Orderings can be Context-Sensitive. In A. Voronkov, editor, *Proc. of XVIII Conference on Automated Deduction, CADE'02*, LNAI 2392:314–331, Springer-Verlag, Berlin, 2002.

[BM06] R. Bruni and J. Meseguer. Semantic foundations for generalized rewrite theories. *Theoretical Computer Science* 351(1):386-414, 2006.

[BN98] F. Baader and T. Nipkow. Term Rewriting and All That. Cambridge University Press, 1998.

[CDEL$^+$07] Clavel, M., F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and C. Talcott. All About Maude – A High-Performance Logical Framework. Lecture Notes in Computer Science 4350, 2007.

[Der04] N. Dershowitz. Termination by Abstraction. In B. Demoen and V. Lifschitz, editors, *Proc. of 20th International Conference on Logic Programming, ICLP'04*, LNCS 3132:1-18, Springer-Verlag, Berlin, 2004.

[DLMM$^+$04] F. Durán, S. Lucas, C. Marché, J. Meseguer, and X. Urbain. Proving Termination of Membership Equational Programs. In *Proc. of 2004 ACM SIGPLAN Symposium on Partial Evaluation and Program Manipulation, PEPM'04*, pages 147–158. ACM Press, 2004.

[DLMM$^+$08] F. Durán, S. Lucas, C. Marché, J. Meseguer, and X. Urbain. Proving Operational Termination of Membership Equational Programs. *Higher-Order and Symbolic Computation*, 21(1-2):59–88, 2008.

[EH09] J. Endrullis and D. Hendriks. From Outermost to Context-Sensitive Rewriting. In R. Treinen, editor, *Proc. of 20th International Conference on Rewriting Techniques and Applications, RTA'09*, LNCS 5595:305-319, Springer-Verlag, Berlin, 2009.

[End10] J. Endrullis. Jambox, Automated Termination Proofs For String and Term Rewriting. Available at `http://joerg.endrullis.de/jambox.html`.

[EWZ08] J. Endrullis, J. Waldmann, and H. Zantema. Matrix Interpretations for Proving Termination of Term Rewriting. *Journal of Automated Reasoning* 40(2-3):195-220, 2008.

[Fer05] M.-L. Fernández. Relaxing monotonicity for innermost termination. *Information Processing Letters*, 93:117-123, 2005.

[FGJM85] K. Futatsugi, J. Goguen, J.-P. Jouannaud, and J. Meseguer. Principles of OBJ2. In *Conference Record of the 12th Annual ACM Symposium on Principles of Programming Languages, POPL'85*, pages 52-66, ACM Press, 1985.

[FN97] K. Futatsugi and A. Nakagawa. An Overview of CAFE Specification Environment – An algebraic approach for creating, verifying, and maintaining formal specification over networks –. In *Proc. of 1st International Conference on Formal Engineering Methods*, 1997.

[FR99] M.C.F. Ferreira and A.L. Ribeiro. Context-Sensitive AC-Rewriting. In P. Narendran and M. Rusinowitch, editors, *Proc. of 10th International Conference on Rewriting Techniques and Applications, RTA'99*, LNCS 1631:286-300, Springer-Verlag, Berlin, 1999.

[GAO02] J. Giesl, T. Arts, and E. Ohlebusch. Modular Termination Proofs for Rewriting Using Dependency Pairs. *Journal of Symbolic Computation*, 34(1):21–58, 2002.

[GL02] B. Gramlich and S. Lucas. Simple Termination of Context-Sensitive Rewriting. In B. Fischer and E. Visser, editors, *Proc. of III ACM SIGPLAN Workshop on Rule-Based Programming, RULE'02*, pages 29–41. ACM Press, 2002.

[GL09a] R. Gutiérrez and S. Lucas. Mechanizing Proofs of Termination in the Context-Sensitive Dependency Pairs Framework. In F. Lucio, G. Moreno, and R. Peña, editors, Proc. of the *9th Spanish Conference on Programming and Computer Languages, PROLE'09,* pages 265-274, Universidad del País Vasco, 2009.

[GL09b] R. Gutiérrez and S. Lucas. Mechanizing Proofs of Termination with Context-Sensitive Dependency Pairs. In A. Geser and J. Waldmann, editors, Proc. of the *10th International Workshop on Termination, WST'09*, pages 43-46, HTWK Leipzig, 2009.

[GLU08] R. Gutiérrez, S. Lucas, and X. Urbain. Usable Rules for Context-Sensitive Rewrite Systems. In A. Voronkov, editor, *Proc. of the 19th International Conference on Rewriting Techniques and Applications, RTA'08*, LNCS, 5117:126-141, Springer-Verlag, Berlin, 2008.

[GM99] J. Giesl and A. Middeldorp. Transforming Context-Sensitive Rewrite Systems. In P. Narendran and M. Rusinowitch, editors, *Proc. of X International Conference on Rewriting Techniques and Applications, RTA'99*, LNCS 1631:271–285, Springer-Verlag, Berlin, 1999.

[GM04] J. Giesl and A. Middeldorp. Transformation techniques for context-sensitive rewrite systems. *Journal of Functional Programming*, 14(4):379–427, 2004.

[GST06] J. Giesl, P. Schneider-Kamp, and R. Thiemann. AProVE 1.2: Automatic Termination Proofs in the Dependency Pair Framework. In U. Furbach and N. Shankar, editors, *Proc. of Third International Joint Conference on Automated Reasoning, IJCAR'06*, LNAI 4130:281-286, Springer-Verlag, Berlin, 2006. Available at `http://aprove.informatik.rwth-aachen.de/`.

[GTS04] J. Giesl, R. Thiemann, and P. Schneider-Kamp. The Dependency Pair Framework: Combining Techniques for Automated Termination Proofs. In F. Baader and A. Voronkov, editors, *Proc. of XI International Conference on Logic for Programming Artificial Intelligence and Reasoning, LPAR'04*, LNAI 3452:301–331, Springer-Verlag, Berlin, 2004.

[GTS05] J. Giesl, R. Thiemann, and P. Schneider-Kamp. Proving and Disproving Termination of Higher-Order Functions. In B. Gramlich, editor, *Proc. of 5th International Workshop on Frontiers of Combining Systems, FroCoS'05*, LNAI 3717:216-231, Springer-Verlag, Berlin, 2005.

[GTSF06] J. Giesl, R. Thiemann, P. Schneider-Kamp, and S. Falke. Mechanizing and Improving Dependency Pairs. *Journal of Automatic Reasoning*, 37(3):155–203, 2006.

[Gut08] R. Gutiérrez. Context-Sensitive Dependency Pairs Framework. Master's thesis, Departamento de Sistemas Informáticos y Computación, Universitat Politècnica de València, València, Spain, December 2008.

[GWMF$^+$00] J.A. Goguen, T. Winkler, J. Meseguer, K. Futatsugi, and J.-P. Jouannaud. Introducing OBJ. In J. Goguen and G. Malcolm, editors, *Software Engineering with OBJ: algebraic specification in action*, Kluwer, 2000.

[HM04] N. Hirokawa and A. Middeldorp. Dependency Pairs Revisited. In V. van Oostrom, editor, *Proc. of XV International Conference on Rewriting Techniques and Applications, RTA'04*, LNCS 3091:249–268, Springer-Verlag, Berlin, 2004.

[HM05] N. Hirokawa and A. Middeldorp. Automating the Dependency Pair Method. *Information and Computation*, 199:172–199, 2005.

[HM07] N. Hirokawa and A. Middeldorp. Tyrolean termination tool: Techniques and features. *Information and Computation*, 205:474-511, 2007.

[HPW92] P. Hudak, S.J. Peyton-Jones, and P. Wadler. Report on the Functional Programming Language Haskell: a non–strict, purely functional language. *Sigplan Notices*, 27(5):1-164, 1992.

[KNT99] K. Kusakari, M. Nakamura, and Y. Toyama. Argument Filtering Transformation. In G. Nadathur, editor, *Proc. of International Conference on Principles and Practice of Declarative Programming, PPDP'99*, LNCS 1702:47–61, 1999.

[LM08] S. Lucas and J. Meseguer. Order-Sorted Dependency Pairs. In *Proc. of the 10th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming, PPDP'08* , pages 108-119, ACM Press, 2008.

[LM09] S. Lucas and J. Meseguer. Operational Termination of Membership Equational Programs: the Order-Sorted Way. *Electronic Notes in Theoretical Computer Science*, 238(3):207-225, 2009.

[Luc96] S. Lucas. Termination of context-sensitive rewriting by rewriting. In F. Meyer auf der Heide and B. Monien, editors, *Proc. of 23rd. International Colloquium on Automata, Languages and Programming, ICALP'96*, LNCS 1099:122-133, Springer-Verlag, Berlin, 1996.

[Luc98] S. Lucas. Context-Sensitive Computations in Functional and Functional Logic Programs. *Journal of Functional and Logic Programming*, 1998(1):1–61, 1998.

[Luc01] S. Lucas. Termination of on-demand rewriting and termination of OBJ programs. In *Proc. of 3rd International Conference on Principles and Practice of Declarative Programming, PPDP'01*, pages 82-93, ACM Press, 2001.

[Luc02] S. Lucas. Context-Sensitive Rewriting Strategies. *Information and Computation*, 178(1):293–343, 2002.

[Luc04a] S. Lucas. MU-TERM: A Tool for Proving Termination of Context-Sensitive Rewriting. In V. van Oostrom, editor, *Proc. of XV International Conference on Rewriting Techniques and Applications, RTA'04*, LNCS 3091:200–209, Springer-Verlag, Berlin, 2004. Available at `http://zenon.dsic.upv.es/muterm/`.

[Luc04b] S. Lucas. Polynomials for Proving Termination of Context-Sensitive Rewriting. In I. Walukiewicz, editor, *Proc. of VII International Conference on Foundations of Software Science and Computation Structures, FOSSACS'04*, LNCS 2987:318–332, Springer-Verlag, Berlin, 2004.

[Luc05] S. Lucas. Polynomials over the Reals in Proofs of Termination: from Theory to Practice. *RAIRO Theoretical Informatics and Applications*, 39(3):547–586, 2005.

[Luc06] S. Lucas. Proving Termination of Context-Sensitive Rewriting by Transformation. *Information and Computation*, 204(12):1782–1846, 2006.

[Luc07] S. Lucas. Practical use of polynomials over the reals in proofs of termination. In *Proc. of 9th International Symposium on Principles and Practice of Declarative Programming, PPDP'07*, pages 39-50, ACM Press, 2007.

[McC60] J. McCarthy. Recursive Functions of Symbolic Expressions and their Computations by Machine, Part I. *Communications of the ACM*, 3(4):184-195, 1960.

[Mid01] A. Middeldorp. Approximating dependency graphs using tree automata techniques. In R. Goré, A. Leitsch, and T. Nipkow, editors, *Proc. of the International Joint Conference on Automated Reasoning, IJCAR'01*, LNAI 2083:593-610, 2001.

[Mid02] A. Middeldorp, Approximations for strategies and termination *Electronic Notes in Computer Science*, volume 70(6), 2002.

[NSEP92] E.G.J.M.H. Nöcker, J.E.W. Smetsers, M.C.J.D. van Eekelen, and M.J. Plasmeijer. Concurrent Clean. In E.H.L. Aarts, J. Leeuwen, and M. Rem, editors, *Proc. of Parallel Architectures and Languages Europe, PARLE'91*, LNCS 506:202-219, Springer-Verlag, Berlin, 1992.

[Ohl02] E. Ohlebusch. *Advanced Topics in Term Rewriting*. Springer-Verlag, 2002.

[SG08] F. Schernhammer and B. Gramlich. Termination of Lazy Rewriting Revisited. *Electronic Notes in Theoretical Computer Science* 204:35-51, 2008.

[SG09] F. Schernhammer and B. Gramlich. VMTL-A Modular Termination Laboratory. In R. Treinen, editor, *Proc. of 20th International Conference on Rewriting Techniques and Applications, RTA'09*, LNCS 5595:285-294, Springer-Verlag, Berlin, 2009.

[TeR03] TeReSe. *Term Rewriting Systems*. Cambridge University Press, 2003.

[Thi07] R. Thiemann. The DP Framework for Proving Termination of Term Rewriting. PhD Thesis. Available as Technical Report AIB-2007-17, RWTH Aachen, Germany, 2007.

[Zan97] H. Zantema. Termination of Context-Sensitive Rewriting. In H. Comon, editor, *Proc. of VII International Conference on Rewriting Techniques and Applications, RTA'97*, LNCS 1232:172–186, Springer-Verlag, Berlin, 1997.

Appendix A. Appendix

Proofs of Section 3

Lemma 2 *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, and $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. If s is not μ-terminating, then there is a subterm t of s ($s \trianglerighteq t$) such that $t \in \mathcal{T}_{\infty,\mu}$.*

Proof. By structural induction. If s is a constant symbol, it is obvious: take $t = s$. If $s = f(s_1, \ldots, s_k)$, then we proceed by contradiction. If there is no subterm t of s such that $t \in \mathcal{T}_{\infty,\mu}$, then $s \notin \mathcal{T}_{\infty,\mu}$. Since s is not μ-terminating, there is a strict subterm t of s ($s \rhd t$) that is not μ-terminating. By the Induction Hypothesis, there is $t' \in \mathcal{T}_{\infty,\mu}$ such that $t \trianglerighteq t'$. Then, we have $s \rhd t'$, thus leading to a contradiction. □

Lemma 3 *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, and $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. If s is not μ-terminating, then there is a μ-replacing subterm t of s such that $t \in \mathcal{M}_{\infty,\mu}$.*

Proof. By structural induction. If s is a constant symbol, it is obvious: take $t = s$. If $s = f(s_1, \ldots, s_k)$, then we proceed by contradiction. If there is no μ-replacing subterm t of s such that $t \in \mathcal{M}_{\infty,\mu}$, then $s \notin \mathcal{M}_{\infty,\mu}$, i.e., there is a strict μ-replacing subterm t of s which is not μ-terminating. We have $t = s|_p$ for some $p \in \mathcal{P}os^{\mu}(s) - \{\Lambda\}$. By the Induction Hypothesis, t contains a μ-replacing subterm t' which belongs to $\mathcal{M}_{\infty,\mu}$, i.e., $t' = t|_q$ for some $q \in \mathcal{P}os^{\mu}(t)$. By Proposition 1, $p.q \in \mathcal{P}os^{\mu}(s)$. Thus, t' is a μ-replacing subterm of s that belongs to $\mathcal{M}_{\infty,\mu}$, thus leading to a contradiction. □

Lemma 4 *Let $\mathcal{R}$ be a TRS, $\mu \in M_{\mathcal{R}}$, and $t \in \mathcal{M}_{\infty,\mu}$. If $t \overset{>\Lambda}{\hookrightarrow}^{*} u$ and u is non-μ-terminating, then $u \in \mathcal{M}_{\infty,\mu}$.*

Proof. All μ-rewritings below of t the root are issued on μ-replacing and μ-terminating terms that remain μ-terminating by Lemma 1. Then, all strict μ-replacing subterms of u (which are the ones that can be originated or transformed by μ-rewritings from t to u) are μ-terminating. Since u is non-μ-terminating, $u \in \mathcal{M}_{\infty,\mu}$. □

Proofs of Section 3.2

Lemma 5 *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ and σ be a substitution. If there is a rule $l \to r \in R$ such that $\sigma(l) \ntrianglerighteq t$ and $\sigma(r) \rhd_{\not\mu} t$, then there is no $x \in \mathcal{V}ar(r)$ such that $\sigma(x) \trianglerighteq t$. Furthermore, there is a term $t' \in \mathcal{HT}$ such that $r \rhd_{\not\mu} t'$ and $\sigma(t') = t$.*

Proof. By contradiction. If there is $x \in \mathcal{V}ar(r)$ such that $\sigma(x) \trianglerighteq t$, then since variables in l are always below some function symbol we have $\sigma(l) \rhd t$, leading to a contradiction.

Since there is no $x \in \mathcal{V}ar(r)$ such that $\sigma(x) \trianglerighteq t$ but we have that $\sigma(r) \rhd_{\not\mu} t$, then there is a nonvariable and non-μ-replacing position $p \in \mathcal{P}os_{\mathcal{F}}(r) - \mathcal{P}os^{\mu}(r)$ of r, such that $\sigma(r|_p) = t$. Then, we let $t' = r|_p$. Note that $t' \in \mathcal{HT}$. □

Lemma 6 *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Let A be a μ-rewrite sequence $t_1 \hookrightarrow t_2 \hookrightarrow \cdots \hookrightarrow t_n$ with $t_i \in \mathcal{M}_{\infty,\mu}$ for all i, $1 \leq i \leq n$. If there is a term $t \in \mathcal{M}_{\infty,\mu}$ such that $t_1 \ntrianglerighteq t$ and $t_n \rhd_{\not\mu} t$, then $t = \sigma(s)$ for some $s \in \mathcal{DHT}$ and substitution σ.*

Proof. By induction on n:

1. If $n = 1$, then it is vacuously true because $t_1 \not\rhd t$ and $t_1 \rhd_{\not\mu} t$ do not simultaneously hold.
2. If $n > 1$, then we assume that $t_1 \not\rhd t$ and $t_n \rhd_{\not\mu} t$. We consider two cases:
   (a) If $t_{n-1} \rhd_{\not\mu} t$, then by the induction hypothesis the conclusion follows.
   (b) If $t_{n-1} \rhd_{\not\mu} t$ does not hold, then, since assuming $t_{n-1} \rhd_{\mu} t$ leads to a contradiction (because $t_{n-1} \in \mathcal{M}_{\infty,\mu}$ in the hypothesis implies that $t \notin \mathcal{M}_{\infty,\mu}$), we have that $t_{n-1} \not\rhd t$. Let $l \to r \in R$ be such that $t_{n-1} = C[\sigma(l)]$ and $t_n = C[\sigma(r)]$ for some context $C[\,]$ and substitution σ. Then, in particular, $\sigma(l) \not\rhd t$ and, since $t_n \rhd_{\not\mu} t$ there must be $\sigma(r) \rhd_{\not\mu} t$. Thus, by Lemma 5 we conclude that $t = \sigma(s)$ for some $s \in \mathcal{HT}$ and substitution σ. Since $t \in \mathcal{M}_{\infty,\mu}$, it follows that $root(t) = root(s) \in \mathcal{D}$. Thus, $s \in \mathcal{DHT}$.

□

Proposition 4 *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. Consider a finite or infinite sequence of the form $t_1 \overset{\Lambda}{\hookrightarrow} s_1 \unrhd_\mu t'_2 \overset{>\Lambda}{\hookrightarrow}{}^* t_2 \overset{\Lambda}{\hookrightarrow} s_2 \unrhd_\mu t'_3 \overset{>\Lambda}{\hookrightarrow}{}^* t_3 \cdots$ with $t_j, t'_j \in \mathcal{M}_{\infty,\mu}$ for all $j \geq 1$. If there is a term $t \in \mathcal{M}_{\infty,\mu}$ such that $t_i \rhd_{\not\mu} t$ for some $i \geq 1$, then $t_1 \rhd_{\not\mu} t$ or $t = \sigma(s)$ for some $s \in \mathcal{DHT}$ and substitution σ.*

Proof. By induction on i:

1. If $i = 1$, it is trivial.
2. If $i > 1$ and $t_i \rhd_{\not\mu} t$, then we consider two cases:
   (a) If $t_{i-1} \rhd_{\not\mu} t$, then by the induction hypothesis, the conclusion follows.
   (b) If $t_{i-1} \rhd_{\not\mu} t$ does not hold, then let $l \to r \in R$ and σ be such that $t_{i-1} = \sigma(l)$ and $s_{i-1} = \sigma(r) \unrhd_\mu t'_i$. Since $t_{i-1} \rhd_\mu t$ leads to a contradiction (because $t_{i-1} \in \mathcal{M}_{\infty,\mu}$ implies that $t \notin \mathcal{M}_{\infty,\mu}$), we have that $t_{i-1} \not\rhd t$. We consider two cases:
      (A) If $t'_i \rhd t$, then, since $t'_i, t \in \mathcal{M}_{\infty,\mu}$, the case $t'_i \rhd_\mu t$ is excluded and the only possibility is that $t'_i \rhd_{\not\mu} t$. Then, since $\sigma(l) = t_{i-1} \not\rhd t$ and $\sigma(r) \unrhd_\mu t'_i \rhd_{\not\mu} t$, i.e. $\sigma(r) \rhd_{\not\mu} t$, by Lemma 5 we conclude that $t = \sigma(s)$ for some $s \in \mathcal{HT}$ and substitution σ. Since $t \in \mathcal{M}_{\infty,\mu}$, it follows that $root(t) = root(s) \in \mathcal{D}$. Thus, $s \in \mathcal{DHT}$.
      (B) If $t'_i \not\rhd t$, then, by applying Lemma 4 and Lemma 6 to the μ-rewrite sequence $t'_i \overset{>\Lambda}{\hookrightarrow}{}^*_{\mathcal{R},\mu} t_i$, the conclusion follows.

□