

Document downloaded from:

<http://hdl.handle.net/10251/202271>

This paper must be cited as:

Gutiérrez Gil, R.; Lucas Alba, S.; Urbain, X. (2008). Usable Rules for Context-Sensitive Rewrite Systems. Lecture Notes in Computer Science. 5117:126-141.
https://doi.org/10.1007/978-3-540-70590-1_9



The final publication is available at

http://dx.doi.org/10.1007/978-3-540-70590-1_9

Copyright Springer-Verlag

Additional Information

Usable Rules for Context-Sensitive Rewrite Systems^{*}

Raúl Gutiérrez¹, Salvador Lucas¹, and Xavier Urbain²

¹ DSIC, Universidad Politécnica de Valencia, Spain

{rgutierrez, slucas}@dsic.upv.es

² Cédric-CNAM, ENSIIE, France

urbain@ensiie.fr

Abstract. Recently, the dependency pairs (DP) approach has been generalized to context-sensitive rewriting (CSR). Although the *context-sensitive dependency pairs (CS-DP) approach* provides a very good basis for proving termination of CSR, the current developments basically correspond to a ten-years-old DP approach. Thus, the task of adapting all recently introduced dependency pairs techniques to get a more powerful approach becomes an important issue. In this direction, *usable rules* are one of the most interesting and powerful notions. Actually usable rule have been investigated in connection with proofs of *innermost termination* of CSR. However, the existing results apply to a quite restricted class of systems. In this paper, we introduce a notion of usable rules that can be used in proofs of termination of CSR with arbitrary systems. Our benchmarks show that the performance of the CS-DP approach is much better when such usable rules are considered in proofs of termination of CSR.

Keywords. Dependency pairs, term rewriting, termination.

1 Introduction

During the last decade, the impressive advances in techniques for proving termination of rewriting (remarkably the dependency pairs approach [6, 10, 13, 14]) have succeeded in solving termination problems that stood out of reach for a long time. Roughly speaking, given a Term Rewriting System (TRS) \mathcal{R} , the dependency pairs associated to \mathcal{R} give rise to a new TRS $DP(\mathcal{R})$ which (together with \mathcal{R}) determines the so-called *dependency chains* whose finiteness characterizes termination of \mathcal{R} . The dependency pairs can be presented as a *dependency graph*, where the absence of infinite chains can be analyzed by considering the *cycles* in the graph. Basically, given a *cycle* $\mathfrak{C} \subseteq DP(\mathcal{R})$ in the dependency graph, we require $l \succeq r$ for *all* rules in the TRS \mathcal{R} , $u \succeq v$ or $u \sqsupset v$ for all dependency pairs $u \rightarrow v \in \mathfrak{C}$ and $u \sqsupset v$ for *at least one* $u \rightarrow v \in \mathfrak{C}$. Here, \succeq is a stable and monotonic quasi-ordering on terms and \sqsupset is a *well-founded* ordering; both of them can be different for the different cycles in the dependency graph.

^{*} Work partially supported by the EU (FEDER) and the Spanish MEC, under grants TIN 2004-7943-C04-02, TIN 2007-68118-C02 and HA 2006-0007.

Termination problems with many rules require more time for getting an answer. Even worse: since termination proofs are usually constrained to succeed within a given (often short) time-out, the proof could get lost due to a lack of time. For those reasons, techniques leading to increase the efficiency (and also the power) of the dependency pairs method, like *usable rules*, appear like a key issue. Usable rules $\mathcal{U}(\mathcal{R}, \mathcal{C}) \subseteq \mathcal{R}$ are associated to a given *cycle* \mathcal{C} of the dependency graph for \mathcal{R} . For particular (but widely used) classes of quasi-orderings \succeq , we can restrict the comparisons $l \succeq r$ to rules $l \rightarrow r$ in $\mathcal{U}(\mathcal{R}, \mathcal{C})$ instead of using \mathcal{R} . Since $\mathcal{U}(\mathcal{R}, \mathcal{C})$ is (usually) smaller than \mathcal{R} , proofs of termination often become easier in this way. Usable rules were introduced ten years ago by Arts and Giesl for proving termination of innermost rewriting [5]. The adaptation of the idea to (unrestricted) rewriting [14, 17] took some years. A possible reason for that is that the proof of soundness for the innermost and for the unrestricted cases are totally different. The proof of soundness in [14, 17] relies on a transformation in which all infinite (minimal) rewrite sequences can be simulated by using a *restricted* set of rules. This transformation was devised by Gramlich for a completely different purpose [15]. Later, Urbain [24] used it (with some modifications) to prove termination of rewriting modules. Finally, Hirokawa and Middeldorp [17] and (independently) Thiemann *et al.* [14] combined this idea with the idea of *usable rules* leading to an improved framework for proving termination of rewriting.

In this paper, we extend the notion of usable rule to the recently introduced dependency pairs approach for context-sensitive rewriting (CS-DPs [2, 3]). Proving termination of *context-sensitive rewriting* (CSR [18, 20]) is an interesting problem with many applications in the fields of term rewriting and programming languages (see [8, 12, 19, 20, 22] for further motivations). In CSR, a *replacement map* (i.e., a mapping $\mu : \mathcal{F} \rightarrow \wp(\mathbb{N})$ satisfying $\mu(f) \subseteq \{1, \dots, k\}$, for each k -ary symbol f of a signature \mathcal{F}) is used to discriminate the argument positions on which the rewriting steps are allowed; rewriting at the topmost position is always possible. The following example gives a first intuition of CSR and CS-DPs; full details are given below.

Example 1. Consider the following TRS \mathcal{R} borrowed from [7, Example 4.7.37]. The program zips two lists of integers into a single one but instead of pairing the components it rather computes their quotients:

$$\text{sel}(0, \text{cons}(x, xs)) \rightarrow x \quad (1) \quad \text{sel}(\text{s}(n), \text{cons}(x, xs)) \rightarrow \text{sel}(n, xs) \quad (7)$$

$$\text{minus}(x, 0) \rightarrow x \quad (2) \quad \text{minus}(\text{s}(x), \text{s}(y)) \rightarrow \text{minus}(x, y) \quad (8)$$

$$\text{quot}(0, \text{s}(y)) \rightarrow 0 \quad (3) \quad \text{quot}(\text{s}(x), \text{s}(y)) \rightarrow \text{s}(\text{quot}(\text{minus}(x, y), \text{s}(y))) \quad (9)$$

$$\text{zWquot}(\text{nil}, x) \rightarrow \text{nil} \quad (4) \quad \text{from}(x) \rightarrow \text{cons}(x, \text{from}(\text{s}(x))) \quad (10)$$

$$\text{zWquot}(x, \text{nil}) \rightarrow \text{nil} \quad (5) \quad \text{tail}(\text{cons}(x, xs)) \rightarrow xs \quad (11)$$

$$\text{head}(\text{cons}(x, xs)) \rightarrow x \quad (6)$$

$$\text{zWquot}(\text{cons}(x, xs), \text{cons}(y, ys)) \rightarrow \text{cons}(\text{quot}(x, y), \text{zWquot}(xs, ys)) \quad (12)$$

with $\mu(\text{cons}) = \{1\}$ and $\mu(f) = \{1, \dots, ar(f)\}$ for all other symbols $f \in \mathcal{F}$. The set of CS-DPs of \mathcal{R} is:

$$\begin{aligned}
\text{MINUS}(\mathbf{s}(x), \mathbf{s}(y)) &\rightarrow \text{MINUS}(x, y) & \text{SEL}(\mathbf{s}(n), \text{cons}(x, xs)) &\rightarrow \text{SEL}(n, xs) \\
\text{QUOT}(\mathbf{s}(x), \mathbf{s}(y)) &\rightarrow \text{MINUS}(x, y) & \text{ZWQUOT}(\text{cons}(x, xs), \text{cons}(y, ys)) &\rightarrow \text{QUOT}(x, y) \\
\text{QUOT}(\mathbf{s}(x), \mathbf{s}(y)) &\rightarrow \text{QUOT}(\text{minus}(x, y), \mathbf{s}(y)) & \text{SEL}(\mathbf{s}(n), \text{cons}(x, xs)) &\rightarrow xs \\
\text{TAIL}(\text{cons}(x, xs)) &\rightarrow xs
\end{aligned}$$

Note that non- μ -replacing subterms in right-hand sides (e.g., $\text{from}(\mathbf{s}(x))$ in rule (10)) are *not* considered to build the CS-DPs. Also, in sharp contrast with the unrestricted case, *collapsing* dependency pairs like $\text{TAIL}(\text{cons}(x, xs)) \rightarrow xs$ (where the right-hand side is a variable) are introduced.

Regarding proofs of termination of *innermost CSR*, the straightforward adaptation of usable rules to the context-sensitive setting only works for the so-called *conservative systems* (see [4]) where collapsing dependency pairs do not occur. In Section 3, we show that the standard adaptation does *not* work when proofs of termination of *CSR* are attempted. In Section 4, we provide a general notion of usable rules for proving termination of *CSR*. Although we follow the same proof style, our proof of soundness differs from those in [14, 15, 17, 24] in several aspects that we clarify below. In Section 5, we prove that it is possible to use the *standard* (simpler) notion of usable rules [14, 17] in proofs of termination of *CSR* for a restricted class of CS-TRSs: the *strongly* conservative systems. Section 6 provides experimental evaluations and Section 7 concludes. Complete proofs are given in [16].

2 Preliminaries

We assume knowledge about standard definitions and notations for term rewriting (including dependency pairs) as given in, e.g., [23]. In the following, we provide some definitions and notation on CSR [18, 20] and CS-DPs [2, 3].

Context-Sensitive Rewriting. Given a TRS $\mathcal{R} = (\mathcal{F}, R)$, we consider the signature \mathcal{F} as the disjoint union $\mathcal{F} = \mathcal{C} \uplus \mathcal{D}$ of *constructors* symbols $c \in \mathcal{C}$ and *defined* symbols $f \in \mathcal{D}$ where $\mathcal{D} = \{\text{root}(l) \mid l \rightarrow r \in R\}$ and $\mathcal{C} = \mathcal{F} - \mathcal{D}$. A mapping $\mu : \mathcal{F} \rightarrow \wp(\mathbb{N})$ is a *replacement map* (or \mathcal{F} -map) if $\forall f \in \mathcal{F}, \mu(f) \subseteq \{1, \dots, \text{ar}(f)\}$ [18]. Let $M_{\mathcal{F}}$ be the set of all \mathcal{F} -maps ($M_{\mathcal{R}}$ for the \mathcal{F} -maps of a TRS $\mathcal{R} = (\mathcal{F}, R)$). A binary relation R on terms in $\mathcal{T}(\mathcal{F}, \mathcal{X})$ is μ -monotonic if tRs implies $f(t_1, \dots, t_{i-1}, t, \dots, t_n)Rf(t_1, \dots, t_{i-1}, s, \dots, t_n)$ for all $f \in \mathcal{F}, i \in \mu(f)$, and $t, s, t_1, \dots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. The set of μ -replacing positions $\text{Pos}^{\mu}(t)$ of $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ is: $\text{Pos}^{\mu}(t) = \{\epsilon\}$, if $t \in \mathcal{X}$ and $\text{Pos}^{\mu}(t) = \{\epsilon\} \cup \bigcup_{i \in \mu(\text{root}(t))} i.\text{Pos}^{\mu}(t|_i)$, if $t \notin \mathcal{X}$. The set of μ -replacing variables of t is $\text{Var}^{\mu}(t) = \{x \in \text{Var}(t) \mid \exists p \in \text{Pos}^{\mu}(t), t|_p = x\}$. The μ -replacing subterm relation \succeq_{μ} is defined by $t \succeq_{\mu} s$ if there is $p \in \text{Pos}^{\mu}(t)$ such that $s = t|_p$. We write $t \triangleright_{\mu} s$ if $t \succeq_{\mu} s$ and $t \neq s$. We write $t \triangleright_{\mu}^{\#} s$ to denote that s is a non- μ -replacing strict subterm of t : $t \triangleright_{\mu}^{\#} s$ if there is $p \in \text{Pos}(t) - \text{Pos}^{\mu}(t)$ such that $s = t|_p$. We say that $f \in \mathcal{F}$ is a *hidden symbol* in

$l \rightarrow r \in R$ if there exists a term $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ s.t. $r \triangleright_{\mu} t$ and $\text{root}(t) = f$. We say that a variable x is *migrating* in $l \rightarrow r \in R$ if $x \in \text{Var}^{\mu}(r) - \text{Var}^{\mu}(l)$. In *context-sensitive rewriting* (CSR [18]), we (only) rewrite terms at μ -replacing positions: t μ -rewrites to s , written $t \hookrightarrow_{\mu} s$ (or $t \hookrightarrow_{\mathcal{R}, \mu} s$), if $t \xrightarrow{p}_{\mathcal{R}} s$ and $p \in \text{Pos}^{\mu}(t)$. A TRS \mathcal{R} is μ -terminating if \hookrightarrow_{μ} is terminating. A term t is μ -terminating if there is no infinite μ -rewrite sequence $t = t_1 \hookrightarrow_{\mu} t_2 \hookrightarrow_{\mu} \dots$. A pair (\mathcal{R}, μ) (or triple (\mathcal{F}, μ, R)) where $\mathcal{R} = (\mathcal{F}, R)$ is a TRS and $\mu \in M_{\mathcal{R}}$ is often called a CS-TRS. We denote $\mathcal{H}(\mathcal{R}, \mu)$ (or just \mathcal{H} , if there is no ambiguity) the set of all hidden symbols in (\mathcal{R}, μ) .

Context-Sensitive Dependency Pairs. Given a TRS $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ and $\mu \in M_{\mathcal{R}}$, the set of context-sensitive dependency pairs (CS-DPs) is $\text{DP}(\mathcal{R}, \mu) = \text{DP}_{\mathcal{F}}(\mathcal{R}, \mu) \cup \text{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$, where $\text{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$ and $\text{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ are obtained as follows: let $f(t_1, \dots, t_m) \rightarrow r \in R$ and $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ be such that $r \triangleright_{\mu} s$. Then (1) if $s = g(s_1, \dots, s_n)$, for some $g \in \mathcal{D}$, $s_1, \dots, s_n \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ and $l \not\triangleright_{\mu} s$, then $f^{\sharp}(t_1, \dots, t_m) \rightarrow g^{\sharp}(s_1, \dots, s_n) \in \text{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$; (2) if $s = x \in \text{Var}^{\mu}(r) - \text{Var}^{\mu}(l)$, then $f^{\sharp}(t_1, \dots, t_m) \rightarrow x \in \text{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$. Here, f^{\sharp} and g^{\sharp} are new fresh symbols (called *tuple symbols*) associated to the symbols f and g respectively. The CS-DPs in $\text{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ are called the *collapsing* CS-DPs. Let $\mathcal{F}^{\sharp} = \mathcal{F} \cup \{f^{\sharp} \mid f \in \mathcal{F}\}$. We extend $\mu \in M_{\mathcal{F}}$ into $\mu^{\sharp} \in M_{\mathcal{F}^{\sharp}}$ by $\mu^{\sharp}(f) = \mu^{\sharp}(f^{\sharp}) = \mu(f)$ for each $f \in \mathcal{F}$. As usual, for $t = f(t_1, \dots, t_n) \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, we write t^{\sharp} to denote the *marked* term $f^{\sharp}(t_1, \dots, t_n)$. Let $\mathcal{T}^{\sharp}(\mathcal{F}, \mathcal{X}) = \{t^{\sharp} \mid t \in \mathcal{T}(\mathcal{F}, \mathcal{X}) - \mathcal{X}\}$ be the set of marked terms. We will also use the set $\mathcal{P}^{\sharp}(\mathcal{F}, \mathcal{X}) = \mathcal{T}^{\sharp}(\mathcal{F}, \mathcal{X}) \times (\mathcal{T}^{\sharp}(\mathcal{F}, \mathcal{X}) \cup \mathcal{X})$. Given $t = f^{\sharp}(t_1, \dots, t_k) \in \mathcal{T}^{\sharp}(\mathcal{F}, \mathcal{X})$, we write t^{\natural} to denote the *unmarked* term $f(t_1, \dots, t_k) \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. As usual, capital letters denote marked symbols in examples. A set of pairs $\mathcal{P} \subseteq \mathcal{P}^{\sharp}(\mathcal{F}, \mathcal{X})$ is decomposed into collapsing and non-collapsing pairs ($\mathcal{P}_{\mathcal{X}}$ and $\mathcal{P}_{\mathcal{F}}$, respectively): $\mathcal{P}_{\mathcal{X}} = \{u \rightarrow v \in \mathcal{P} \mid v \in \mathcal{X}\}$ and $\mathcal{P}_{\mathcal{F}} = \mathcal{P} - \mathcal{P}_{\mathcal{X}}$.

Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mathcal{P} \subseteq \mathcal{P}^{\sharp}(\mathcal{F}, \mathcal{X})$ and $\mu \in M_{\mathcal{F}}$. An $(\mathcal{R}, \mathcal{P}, \mu^{\sharp})$ -chain is a finite or infinite sequence of pairs $u_i \rightarrow v_i \in \mathcal{P}$, for $i \geq 1$ such that there is a substitution σ satisfying both:

1. $\sigma(v_i) \hookrightarrow_{\mathcal{R}, \mu^{\sharp}}^* \sigma(u_{i+1})$, if $u_i \rightarrow v_i \in \mathcal{P}_{\mathcal{F}}$, and
2. if $u_i \rightarrow v_i = u_i \rightarrow x_i \in \mathcal{P}_{\mathcal{X}}$, then there is $s_i \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ such that $\sigma(x_i) \triangleright_{\mu} s_i$ and $s_i^{\sharp} \hookrightarrow_{\mathcal{R}, \mu^{\sharp}}^* \sigma(u_{i+1})$.

where $\text{Var}(v_i) \cap \text{Var}(u_j) = \emptyset$ for all $i \neq j$ (renaming if necessary). Let $\mathcal{M}_{\infty, \mu}$ be the set of minimal non- μ -terminating terms. Then, $t \in \mathcal{M}_{\infty, \mu}$ if t is non- μ -terminating and every strict μ -replacing subterm of t is terminating. We say that an $(\mathcal{R}, \mathcal{P}, \mu^{\sharp})$ -chain is *minimal* if for all $i \geq 1$ $\sigma(v_i)$ (whenever $u_i \rightarrow v_i \in \mathcal{P}_{\mathcal{F}}$), s_i^{\sharp} (whenever $u_i \rightarrow v_i \in \mathcal{P}_{\mathcal{X}}$) are μ -terminating w.r.t. \mathcal{R} . A CS-TRS $\mathcal{R} = (\mathcal{F}, \mu, R)$ is μ -terminating if and only if there is no infinite minimal $(\mathcal{R}, \text{DP}(\mathcal{R}, \mu), \mu^{\sharp})$ -chain. For finite CS-TRSs, the CS-DPs can be presented as a *context-sensitive dependency graph* (CS-DG); there is an arc from $u \rightarrow v \in \text{DP}_{\mathcal{F}}(\mathcal{R}, \mu)$ to $u' \rightarrow v' \in \text{DP}(\mathcal{R}, \mu)$ if there is a substitution σ such that $\sigma(v) \hookrightarrow_{\mathcal{R}, \mu}^* \sigma(u')$; and, there is an arc from $u \rightarrow v \in \text{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ to $u' \rightarrow v' \in \text{DP}(\mathcal{R}, \mu)$ if $\text{root}(u')^{\sharp} \in \mathcal{H}$.

We consider the *strongly connected components* in this graph. A μ -reduction pair (\succeq, \sqsupset) consists of a stable and weakly μ -monotonic quasi-ordering \succeq , and a stable and well-founded ordering \sqsupset satisfying $\succeq \circ \sqsupset \subseteq \sqsupset$ or $\sqsupset \circ \succeq \subseteq \sqsupset$. From now on, we assume that all CS-TRSs are finite.

3 Basic Usable Rules

Consider a set of pairs \mathcal{P} and a CS-TRS (\mathcal{R}, μ) . Then, the set of usable rules is the smallest set of rules from \mathcal{R} which are needed to capture all the infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^\#)$ -chains. The rules that are responsible for generating the chains between pairs are those rules rooted by symbols that appear in the right-hand side of the pairs below the root symbol. This concept is captured by the definition of direct dependency [14, 17, 24]:

Definition 1 (Direct Dependency [14, 17]). *Given a TRS $\mathcal{R} = (\mathcal{F}, R)$, we say that $f \in \mathcal{F}$ directly depends on $g \in \mathcal{F}$, written $f \triangleright_d g$, if there is a rule $l \rightarrow r \in R$ with $f = \text{root}(l)$ and g occurs in r .*

The set of defined symbols in a term t is $\mathcal{DFun}(t) = \{f \mid \exists p \in \mathcal{Pos}(t), f = \text{root}(t|_p) \in \mathcal{D}\}$. Let \triangleright_d^* be the transitive and reflexive closure of \triangleright_d . Then, we have:

Definition 2 (Usable Rules [14, 17]). *For a set \mathcal{G} of symbols we denote by $\mathcal{R} \upharpoonright \mathcal{G}$ the set of rewriting rules $l \rightarrow r \in \mathcal{R}$ with $\text{root}(l) \in \mathcal{G}$. The set $\mathcal{U}(\mathcal{R}, t)$ of usable rules of a term t is defined as $\mathcal{R} \upharpoonright \{g \mid f \triangleright_d^* g \text{ for some } f \in \mathcal{DFun}(t)\}$. If \mathcal{P} is a set of dependency pairs then $\mathcal{U}(\mathcal{R}, \mathcal{P}) = \bigcup_{l \rightarrow r \in \mathcal{P}} \mathcal{U}(\mathcal{R}, r)$.*

The set $\mathcal{U}(\mathcal{R}, \mathcal{P})$ can be used instead of \mathcal{R} when looking for a reduction pair that proves termination of \mathcal{R} [14, 17]. Let us now focus on CS-TRSs.

A first attempt to give a notion of usable rules for CSR is given in [4] (basic usable rules) for proofs of *innermost* termination. The results in [4] show that the straightforward generalization of Definition 2 to *CSR* (see Definition 4 below) only applies to *conservative* CS-TRSs and cycles (of CS-DPs), that is, systems having only conservative rules [22]: a rule $l \rightarrow r \in R$ is *conservative* if $\text{Var}^\mu(r) \subseteq \text{Var}^\mu(l)$. First, we adapt Definition 1 to the CSR setting as follows:

Definition 3 (Basic μ -Dependency). *Given a CS-TRS (\mathcal{F}, μ, R) , we say that $f \in \mathcal{F}$ has a basic μ -dependency on $g \in \mathcal{F}$, written $f \blacktriangleright_{d,\mu} g$, if there is $l \rightarrow r \in R$ with $f = \text{root}(l)$ and g occurs in r at a μ -replacing position.*

This leads to a straightforward extension of Definition 2. The set of μ -replacing defined symbols in a term t is $\mathcal{DFun}^\mu(t) = \{f \mid \exists p \in \mathcal{Pos}^\mu(t), f = \text{root}(t|_p) \in \mathcal{D}\}$. Then, we have³:

Definition 4 (Basic Context-Sensitive Usable Rules). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{R}}$. The set $\mathcal{U}_B(\mathcal{R}, \mu, t)$ of basic context-sensitive usable rules of a term t is defined as $\mathcal{R} \upharpoonright \{g \mid f \blacktriangleright_{d,\mu}^* g \text{ for some } f \in \mathcal{DFun}^\mu(t)\}$,*

³ Note that, due to the focus on innermost CSR, [4, Def. 5] slightly differs from ours.

where $\blacktriangleright_{d,\mu}^*$ is the transitive and reflexive closure of $\blacktriangleright_{d,\mu}$. If $\mathcal{P} \subseteq \mathcal{P}^\sharp(\mathcal{F}, \mathcal{X})$, then $\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) = \bigcup_{l \rightarrow r \in \mathcal{P}} \mathcal{U}_B(\mathcal{R}, \mu^\sharp, r)$.

Example 2. (Continuing Example 1) The cycles in the CS-DG are:

$$\{\text{SEL}(\mathbf{s}(n), \text{cons}(x, xs)) \rightarrow \text{SEL}(n, xs)\} \quad (C_1)$$

$$\{\text{MINUS}(\mathbf{s}(x), \mathbf{s}(y)) \rightarrow \text{MINUS}(x, y)\} \quad (C_2)$$

$$\{\text{QUOT}(\mathbf{s}(x), \mathbf{s}(y)) \rightarrow \text{QUOT}(\text{minus}(x, y), \mathbf{s}(y))\} \quad (C_3)$$

Consider the cycle C_3 ; then, $\mathcal{U}_B(\mathcal{R}, \mu^\sharp, C_3)$ contains the following rules:

$$\text{minus}(x, 0) \rightarrow x \quad \text{minus}(\mathbf{s}(x), \mathbf{s}(y)) \rightarrow \text{minus}(x, y)$$

However, as we are going to see, and in sharp contrast with [4], Definition 4 does not lead to a correct approach for proving termination of *CSR*, even for conservative TRSs.

Example 3. Consider the TRS $\mathcal{R} = \{\mathbf{f}(c(x), x) \rightarrow \mathbf{f}(x, x), \mathbf{b} \rightarrow \mathbf{c}(\mathbf{b})\}$ [4] together with $\mu(\mathbf{f}) = \{1, 2\}$ and $\mu(\mathbf{c}) = \emptyset$. Note that (\mathcal{R}, μ) is conservative (and innermost μ -terminating, see [4]).

We have a single cycle $C = \{\mathbf{F}(c(x), x) \rightarrow \mathbf{F}(x, x)\}$. According to Definition 4, we have no usable rules because $\mathbf{F}(x, x)$ contains no symbol in \mathcal{F} . We could wrongly conclude μ -termination of (\mathcal{R}, μ) , but we have the infinite minimal $(\mathcal{R}, C, \mu^\sharp)$ -chain $\mathbf{F}(c(\mathbf{b}), \mathbf{b}) \rightarrow \mathbf{F}(\mathbf{b}, \mathbf{b}) \hookrightarrow \mathbf{F}(c(\mathbf{b}), \mathbf{b}) \rightarrow \dots$.

In the following, we develop a correct definition of usable rules that can be applied to *arbitrary* CS-TRSs.

4 Termination of CS-TRSs with Usable Rules

As shown in [14, 17], considering the set of *usable rules* instead of all the rules suffices for proving termination of $(\mathcal{R}, \mathcal{P})$ -chains (or \mathcal{P} -minimal sequences in [17]). In [14, 17], an *interpretation* of terms as sequences of their possible reducts is used⁴. The definition of the transformation requires adding new fresh (list constructor) symbols $\perp, \mathbf{g} \notin \mathcal{F}$ and the (projection) rules $\mathbf{g}(x, y) \rightarrow x, \mathbf{g}(x, y) \rightarrow y$ (the π -rules). In this way, infinite minimal $(\mathcal{R}, \mathcal{P})$ -chains can be represented as infinite $(\mathcal{U}(\mathcal{R}, \mathcal{P}) \cup \pi, \mathcal{P})$ -chains. We recall here the interpretation definition.

Definition 5 (Interpretation [14, 17]). Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mathcal{G} \subseteq \mathcal{F}$. Let $>$ be an arbitrary total ordering over $\mathcal{T}(\mathcal{F}^\sharp \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ where \perp is a new constant symbol and \mathbf{g} is a new binary symbol. The interpretation $I_{\mathcal{G}}$ is a mapping from terminating terms in $\mathcal{T}(\mathcal{F}^\sharp, \mathcal{X})$ to terms in $\mathcal{T}(\mathcal{F}^\sharp \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ defined as follows:

$$I_{\mathcal{G}}(t) = \begin{cases} t & \text{if } t \in \mathcal{X} \\ f(I_{\mathcal{G}}(t_1), \dots, I_{\mathcal{G}}(t_n)) & \text{if } t = f(t_1 \dots t_n) \text{ and } f \notin \mathcal{G} \\ \mathbf{g}(f(I_{\mathcal{G}}(t_1), \dots, I_{\mathcal{G}}(t_n)), t') & \text{if } t = f(t_1 \dots t_n) \text{ and } f \in \mathcal{G} \end{cases}$$

⁴ This method goes back to [15].

$$\begin{aligned}
\text{where } \quad t' &= \text{order}(\{I_{\mathcal{G}}(u) \mid t \rightarrow_{\mathcal{R}} u\}) \\
\text{order}(T) &= \begin{cases} \perp, & \text{if } T = \emptyset \\ \mathbf{g}(t, \text{order}(T - \{t\})) & \text{if } t \text{ is minimal in } T \text{ w.r.t. } > \end{cases}
\end{aligned}$$

The set of symbols $\mathcal{G} \subseteq \mathcal{F}$ in Definition 5 is intended to represent the set of ‘non-usable symbols’, i.e., symbols which do not occur in the usable rules of the considered set of pairs \mathcal{P} . In rewriting, when considering infinite minimal $(\mathcal{R}, \mathcal{P})$ -chains, we only deal with terminating terms over \mathcal{R} . The interpretation in Definition 5 is defined only for terminating terms because non-terminating terms would yield an infinite term which, actually, does *not* belong to $\mathcal{T}(\mathcal{F}^{\#} \cup \{\perp, \mathbf{g}\}, \mathcal{X})$.

Similarly, we aim at defining a μ -interpretation $I_{\mathcal{G}, \mu}$ that allows us to associate an infinite $(\mathcal{U}(\mathcal{R}, \mu^{\#}, \mathcal{P}) \cup \pi, \mathcal{P}, \mu^{\#})$ -chain to each infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^{\#})$ -chain. Actually, the main problem is that $(\mathcal{R}, \mathcal{P}, \mu^{\#})$ -chains contain non- μ -terminating terms in non- μ -replacing positions which are potentially able to reach μ -replacing positions: subterms at a μ -replacing position are μ -terminating, but we do not know anything about subterms at non- μ -replacing positions. Hence, we have to define our μ -interpretation $I_{\mathcal{G}, \mu}$ both on μ -terminating and non- μ -terminating terms. In [3], we have investigated the structure of infinite μ -rewriting sequences issued from minimal non- μ -terminating terms. Intuitively, one of the main results in [3] states that terms at non- μ -replacing positions in the right-hand side of the rules are essential to track infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^{\#})$ -chains involving collapsing CS-DPs (see [3, Proposition 3.6]). These terms, by definition, are formed by *hidden symbols*. This observation gives us the key to generalize Definition 5 properly. Following Definition 5, a μ -terminating but non-terminating term generates an infinite list. For this reason, $I_{\mathcal{G}}$ (as a mapping from finite into finite terms) is *not* defined for non-terminating terms.

Regarding our μ -interpretation, if we consider the rules headed by hidden symbols as *usable*, then we are avoiding such infinite μ -interpretations of μ -terminating terms. A non- μ -terminating term t (below a non- μ -replacing position) is treated as if its root symbol does not belong to \mathcal{G} , because if it occurs in the $(\mathcal{R}, \mathcal{P}, \mu^{\#})$ -chain at a μ -replacing position, then $t \triangleright_{\mu} s$ and $s^{\#}$ becomes the next term in the chain. To simulate all possible derivations of the terms over (\mathcal{R}, μ) we also need to add to the system the π -rules. Our new μ -interpretation is:

Definition 6 (μ -Interpretation). *Let $\mathcal{R} = (\mathcal{F}, \mu, R)$ be a CS-TRS, $\mathcal{G} \subseteq \mathcal{F}$ be such that $\mathcal{G} \cap \mathcal{H} = \emptyset$. Let $>$ be an arbitrary total ordering over $\mathcal{T}(\mathcal{F}^{\#} \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ where \perp is a new constant symbol and \mathbf{g} is a new binary symbol (with $\mu(\mathbf{g}) = \{1, 2\}$). The μ -interpretation $I_{\mathcal{G}, \mu}$ is a mapping from arbitrary terms in $\mathcal{T}(\mathcal{F}^{\#}, \mathcal{X})$ to terms in $\mathcal{T}(\mathcal{F}^{\#} \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ defined as follows:*

$$I_{\mathcal{G}, \mu}(t) = \begin{cases} t & \text{if } t \in \mathcal{X} \\ f(I_{\mathcal{G}, \mu}(t_1), \dots, I_{\mathcal{G}, \mu}(t_n)) & \text{if } t = f(t_1 \dots t_n) \text{ and } f \notin \mathcal{G} \\ & \text{or } t \text{ is non-}\mu\text{-terminating} \\ \mathbf{g}(f(I_{\mathcal{G}, \mu}(t_1), \dots, I_{\mathcal{G}, \mu}(t_n)), t') & \text{if } t = f(t_1 \dots t_n) \text{ and } f \in \mathcal{G} \\ & \text{and } t \text{ is } \mu\text{-terminating} \end{cases}$$

$$\begin{aligned}
\text{where } t' &= \text{order}(\{I_{\mathcal{G},\mu}(u) \mid t \hookrightarrow_{(\mathcal{R},\mu)} u\}) \\
\text{order}(T) &= \begin{cases} \perp, & \text{if } T = \emptyset \\ \mathbf{g}(t, \text{order}(T - \{t\})) & \text{if } t \text{ is minimal in } T \text{ w.r.t. } > \end{cases}
\end{aligned}$$

The set $\mathcal{G} \subseteq \mathcal{F}$ in Definition 6 corresponds to the set of non-usable symbols as discussed below. Now, we prove that $I_{\mathcal{G},\mu}$ is well-defined. The most important difference (and essential in our proof) among our μ -interpretation and all previous ones [14, 15, 17, 24] is that $I_{\mathcal{G},\mu}$ is well-defined both for μ -terminating or non- μ -terminating terms.

Lemma 1. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$ and let $\mathcal{G} \subseteq \mathcal{F} - \mathcal{H}$. Then, $I_{\mathcal{G},\mu}$ is well-defined.*

Now, we define an appropriate notion of direct μ -dependency. This is not straightforward as shown in the next example.

Example 4. Consider the following conservative non- μ -terminating CS-TRS $\mathcal{R} = \{\mathbf{a}(x, y) \rightarrow \mathbf{b}(x, x), \mathbf{d}(x, \mathbf{e}) \rightarrow \mathbf{a}(x, x), \mathbf{b}(x, \mathbf{c}) \rightarrow \mathbf{d}(x, x), \mathbf{c} \rightarrow \mathbf{e}\}$ with $\mu(\mathbf{a}) = \mu(\mathbf{d}) = \{1, 2\}$, $\mu(\mathbf{b}) = \{1\}$ and $\mu(\mathbf{c}) = \mu(\mathbf{e}) = \emptyset$. The only cycle consists of the dependency pairs $C = \{\mathbf{A}(x, y) \rightarrow \mathbf{B}(x, x), \mathbf{D}(x, \mathbf{e}) \rightarrow \mathbf{A}(x, x), \mathbf{B}(x, \mathbf{c}) \rightarrow \mathbf{D}(x, x)\}$.

According to Definition 4, we have no basic usable rules because the right-hand sides of the dependency pairs have no defined symbols. Since we do not consider the rule $\mathbf{c} \rightarrow \mathbf{e}$ as usable, we would assume $\mathcal{G} = \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}\}$. Then, we *cannot* simulate the infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^{\sharp})$ -chain $\underline{\mathbf{A}(\mathbf{c}, \mathbf{c})} \hookrightarrow \underline{\mathbf{B}(\mathbf{c}, \mathbf{c})} \hookrightarrow \underline{\mathbf{D}(\mathbf{c}, \mathbf{c})} \hookrightarrow \underline{\mathbf{D}(\mathbf{c}, \mathbf{e})} \hookrightarrow \underline{\mathbf{A}(\mathbf{c}, \mathbf{c})} \hookrightarrow \dots$ because we have:

$$s = I_{\mathcal{G},\mu}(\mathbf{A}(\mathbf{c}, \mathbf{c})) = \underline{\mathbf{A}(g(\mathbf{c}, g(\mathbf{e}, \perp)), g(\mathbf{c}, g(\mathbf{e}, \perp)))} \hookrightarrow \underline{\mathbf{B}(g(\mathbf{c}, g(\mathbf{e}, \perp)), g(\mathbf{c}, g(\mathbf{e}, \perp)))} = t$$

The interpreted term $g(\mathbf{c}, g(\mathbf{e}, \perp))$ at the μ -replacing position 1 of s is ‘moved’ to a non- μ -replacing position 2 of t . Hence, we cannot reduce t on the second argument of \mathbf{B} to obtain the term $\mathbf{B}(g(\mathbf{c}, g(\mathbf{e}, \perp)), \mathbf{c})$ required for applying the next CS-DP $(\mathbf{B}(x, \mathbf{c}) \rightarrow \mathbf{D}(x, x))$ which continues the previous $(\mathcal{R}, \mathcal{P}, \mu)$ -chain.

In order to avoid this problem, we modify Definition 3 to take into account symbols occurring at non- μ -replacing positions in the *left-hand sides* of the rules.

Definition 7 (μ -Dependency). *Given a CS-TRS $\mathcal{R} = (\mathcal{F}, \mu, R)$, we say that $f \in \mathcal{F}$ directly μ -depends on $g \in \mathcal{F}$, written $f \triangleright_{d,\mu} g$, if there is a rule $l \rightarrow r \in R$ with $f = \text{root}(l)$ and (1) g occurs in r at a μ -replacing position or (2) g occurs in l at a non- μ -replacing position.*

Remarkably, condition (2) in Definition 7 is not very problematic in practice because most programs are *constructor systems*, which means that no defined symbols occur below the root in the left-hand side of the rules.

Now we are ready to define our notion of usable rules. The set of *non- μ -replacing* defined symbols in a term t is $NDFun^{\mu}(t) = \{f \mid \exists p \in \text{Pos}(t) \text{ and } p \notin \text{Pos}^{\mu}(t), f = \text{root}(t|_p) \in \mathcal{D}\}$.

Definition 8 (Context-Sensitive Usable Rules). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{R}}$, and $\mathcal{P} \subseteq \mathcal{P}^{\sharp}(\mathcal{F}, \mathcal{X})$. The set $\mathcal{U}(\mathcal{R}, \mu^{\sharp}, \mathcal{P})$ of context-sensitive usable*

rules for \mathcal{P} is given by $\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) = \mathcal{U}_{\mathcal{H}}(\mathcal{R}, \mu) \cup \bigcup_{l \rightarrow r \in \mathcal{P}} \mathcal{U}_E(\mathcal{R}, \mu^\sharp, l \rightarrow r)$.

where $\mathcal{U}_E(\mathcal{R}, \mu, l \rightarrow r) = \mathcal{R} \mid \{g \mid f \triangleright_{d, \mu}^* g \text{ for some } f \in \mathcal{DFun}^\mu(r) \cup \mathcal{NDFun}^\mu(l)\}$
 $\mathcal{U}_{\mathcal{H}}(\mathcal{R}, \mu) = \mathcal{R} \mid \{g \mid f \triangleright_{d, \mu}^* g \text{ for some } f \in \mathcal{H}\}$

Note that \mathcal{U}_E extends the notion of usable rules in Definition 2, by taking into account not only dependencies with symbols on the right-hand sides of the rules, but also with some symbols in proper subterms of the left-hand sides. We call $\mathcal{U}_E(\mathcal{R}, \mu)$ the set of *extended usable rules*. On the other hand, $\mathcal{U}_{\mathcal{H}}$ is the set of usable rules corresponding to the hidden symbols. Now, we are ready to formulate and prove our main result in this section.

Theorem 1. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mathcal{P} \subseteq \mathcal{P}^\sharp(\mathcal{F}, \mathcal{X})$, and $\mu \in M_{\mathcal{F}}$. If there exists a μ -reduction pair (\succsim, \sqsupset) such that $\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi \subseteq \succsim$, $\mathcal{P} \subseteq \succsim \cup \sqsupset$, and*

1. *If $\mathcal{P}_{\mathcal{X}} = \emptyset$, then $\mathcal{P} \cap \sqsupset \neq \emptyset$*
2. *If $\mathcal{P}_{\mathcal{X}} \neq \emptyset$, then $\sqsupset_\mu \subseteq \succsim$, and*
 - (a) *$\mathcal{P} \cap \sqsupset \neq \emptyset$ and $f(x_1, \dots, x_k) \succsim f^\sharp(x_1, \dots, x_k)$ for all f^\sharp in \mathcal{P} , or*
 - (b) *$f(x_1, \dots, x_k) \sqsupset f^\sharp(x_1, \dots, x_k)$ for all f^\sharp in \mathcal{P} .*

Let $\mathcal{P}_\sqsupset = \{u \rightarrow v \in \mathcal{P} \mid u \sqsupset v\}$. Then there are no infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^\sharp)$ -chains whenever:

1. *there are no infinite minimal $(\mathcal{R}, \mathcal{P} \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains in case (1) and in case (2a).*
2. *there are no infinite minimal $(\mathcal{R}, (\mathcal{P} \setminus \mathcal{P}_{\mathcal{X}}) \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains in case (2b).*

Proof (Sketch). By contradiction. Assume that there exists an infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^\sharp)$ -chain \mathcal{A} but there is no infinite minimal $(\mathcal{R}, \mathcal{P} \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains in case (1) and (2a), or there is no infinite minimal $(\mathcal{R}, (\mathcal{P} \setminus \mathcal{P}_{\mathcal{X}}) \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains in case (2b). We can assume that there is a $\mathcal{P}' \subseteq \mathcal{P}$ such that \mathcal{A} has a tail \mathcal{B} where all pairs are used infinitely often:

$$t_1 \xrightarrow{\mathcal{R}, \mu}^* u_1 \rightarrow_{\mathcal{P}'} \circ \sqsupset_\mu^\sharp t_2 \xrightarrow{\mathcal{R}, \mu}^* u_2 \rightarrow_{\mathcal{P}'} \circ \sqsupset_\mu^\sharp \dots$$

where $s \sqsupset_\mu^\sharp t$ for $s \in \mathcal{T}(\mathcal{F}, \mathcal{X})$ and $t \in \mathcal{T}^\sharp(\mathcal{F}, \mathcal{X})$ means that $s \sqsupset_\mu t^\sharp$.

Let σ be a substitution, we denote by $\sigma_{I_{\mathcal{G}, \mu}}$ the substitution that assigns to each variable x the term $I_{\mathcal{G}, \mu}(\sigma(x))$ and let \mathcal{G} be the set of defined symbols of $\mathcal{R} \setminus \mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P})$. We show that after applying $I_{\mathcal{G}, \mu}$ we get an infinite $(\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi, \mathcal{P}', \mu^\sharp)$ -chain. All terms in the infinite chain are μ -terminating w.r.t. (\mathcal{R}, μ) . We proceed by induction. Let $i \geq 1$.

- If we consider the step $u_i \rightarrow_{\mathcal{P}'} \circ \sqsupset_\mu^\sharp t_{i+1}$, we have two possibilities:
 1. There is $l \rightarrow r \in \mathcal{P}'_{\mathcal{F}}$, then we get:

$$I_{\mathcal{G}, \mu}(u_i) \xrightarrow{\pi}^* \sigma_{I_{\mathcal{G}, \mu}}(l) \rightarrow_{\mathcal{P}'_{\mathcal{F}}} \sigma_{I_{\mathcal{G}, \mu}}(r) = I_{\mathcal{G}, \mu}(r) = I_{\mathcal{G}, \mu}(t_{i+1})$$

2. There is an $l \rightarrow x \in \mathcal{P}'_{\mathcal{X}}$, then we get:

$$I_{\mathcal{G}, \mu}(u_i) \xrightarrow{\pi}^* \sigma_{I_{\mathcal{G}, \mu}}(l) \rightarrow_{\mathcal{P}'_{\mathcal{X}}} \sigma_{I_{\mathcal{G}, \mu}}(x) = I_{\mathcal{G}, \mu}(\sigma(x))$$

$$\text{and } I_{\mathcal{G}, \mu}(\sigma(x)) \sqsupset_\mu I_{\mathcal{G}, \mu}(t_{i+1}^\sharp)$$

– If we consider $t_i \hookrightarrow_{\mathcal{R}, \mu}^* u_i$. We get $I_{\mathcal{G}, \mu}(t_i) \hookrightarrow_{\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* I_{\mathcal{G}, \mu}(u_i)$.

Therefore we get the infinite $(\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}), \mathcal{P}', \mu^\sharp)$ -chain:

$$I_{\mathcal{G}, \mu}(t_1) \hookrightarrow_{\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* I_{\mathcal{G}, \mu}(u_1) \rightarrow_{\mathcal{P}' \circ \succeq_\mu^\sharp} I_{\mathcal{G}, \mu}(t_2) \hookrightarrow_{\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* I_{\mathcal{G}, \mu}(u_2) \rightarrow_{\mathcal{P}'} \dots$$

Using the premises of the theorem, by monotonicity and stability of \succeq , we would have that $I_{\mathcal{G}, \mu}(t_i) \succeq I_{\mathcal{G}, \mu}(u_i)$ for all $i \geq 1$. By stability of \sqsupset (and of \succeq), we have that $I_{\mathcal{G}, \mu}(u_i) (\succeq \cup \sqsupset) I_{\mathcal{G}, \mu}(t_{i+1})$ for all $i \geq 1$ and $I_{\mathcal{G}, \mu}(u_i) \sqsupset I_{\mathcal{G}, \mu}(t_{i+1})$ for all $j \in J$ for an infinite set $J = \{j_1, \dots, j_n, \dots\}$ of natural numbers $j_1 < j_2 < \dots < j_n < \dots$. Now, since $\succeq \circ \sqsupset \subseteq \sqsupset$ or $\sqsupset \circ \succeq \subseteq \sqsupset$, we would obtain an infinite sequence consisting of infinitely many \sqsupset -steps. We obtain a contradiction to the well-foundedness of \sqsupset . \square

Remark 1. Notice that (as expected) $\mathcal{U}(\mathcal{R}, \mathcal{P}, \mu_\top) = \mathcal{U}(\mathcal{R}, \mathcal{P})$, i.e., our usable rules for CS-TRSs (\mathcal{R}, μ) coincide with the standard definition (see Definition 2) when $\mu = \mu_\top$ is considered (here, $\mu_\top(f) = \{1, \dots, ar(f)\}$ for all symbols $f \in \mathcal{F}$, i.e., no replacement restriction is associated to any symbol).

Thanks to Theorem 1, we do not need to make all rules in \mathcal{R} compatible with the weak component $\succeq_{\mathcal{P}}$ of a reduction pair $(\succeq_{\mathcal{P}}, \sqsupset_{\mathcal{P}})$ associated to a given set of pairs \mathcal{P} . We just need to consider $\mathcal{U}(\mathcal{R}, \mu^\sharp, \mathcal{P})$ (together with the π -rules).

Example 5. (Continuing Examples 1 and 2) Since $\mathcal{H} \cap \mathcal{D} = \{\text{from}, \text{zWquot}\}$, we have that $\mathcal{U}(\mathcal{R}, \mu^\sharp, C_1)$ is:

$$\begin{array}{ll} \text{minus}(x, 0) \rightarrow x & \text{minus}(\text{s}(x), \text{s}(y)) \rightarrow \text{minus}(x, y) \\ \text{quot}(0, \text{s}(y)) \rightarrow 0 & \text{quot}(\text{s}(x), \text{s}(y)) \rightarrow \text{s}(\text{quot}(\text{minus}(x, y), \text{s}(y))) \\ \text{zWquot}(\text{nil}, x) \rightarrow \text{nil} & \text{from}(x) \rightarrow \text{cons}(x, \text{from}(\text{s}(x))) \\ \text{zWquot}(x, \text{nil}) \rightarrow \text{nil} & \\ \text{zWquot}(\text{cons}(x, xs), \text{cons}(y, ys)) \rightarrow \text{cons}(\text{quot}(x, y), \text{zWquot}(xs, ys)) & \end{array}$$

According to Theorem 1, the following polynomial interpretation (computed by MU-TERM [1, 21]) shows the absence of infinite $(\mathcal{R}, C_1, \mu^\sharp)$ -chains.

$$\begin{array}{lll} [\text{s}](x) = x + 1 & [\text{quot}](x, y) = x + y & [\text{minus}](x, y) = 0 \\ [\text{from}](x) = 0 & [\text{sel}](x, y) = 0 & [\text{zWquot}](x, y) = x + y \\ [\text{cons}](x, y) = 0 & [0](x, y) = 0 & [\text{nil}](x, y) = 1 \\ [\text{SEL}](x, y) = x & & \end{array}$$

Note that, if the rules for `sel` were present, we could not find a linear polynomial interpretation for solving this cycle.

Remark 2. When considering Definition 8 (usable rules for *CSR*) and Definition 2 (standard usable rules), one can observe that, despite the fact that *CSR* is a *restriction* of rewriting, we can obtain *more* usable rules in the context-sensitive case. Examples 3 and 4 show that this is because rules associated to hidden symbols that do *not* occur in the right-hand sides of the dependency pairs in the considered cycle can play an essential role in capturing infinite μ -rewrite sequences. Thus, for terminating TRSs \mathcal{R} , it could be sometimes easier to find a proof of μ -termination of the CS-TRS (\mathcal{R}, μ) if we ignore the replacement map μ .

5 Improving Usable Rules

According to the discussion in Section 3, the notion of basic usable rules is not correct even for conservative systems. Still, since $\mathcal{U}_B(\mathcal{R}, \mu, \mathcal{P})$ is contained in (and is usually smaller than) $\mathcal{U}(\mathcal{R}, \mu, \mathcal{P})$, it is interesting to identify a class of CS-TRSs where basic usable rules can be safely used. Then, we consider a more restrictive kind of conservative CS-TRSs: the *strongly conservative* CS-TRSs.

Definition 9. Let \mathcal{F} be a signature, $\mu \in M_{\mathcal{F}}$ and $t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. We denote $\mathcal{V}ar^{\sharp}(t)$ the set of variables in t occurring at non- μ -replacing positions, i.e., $\mathcal{V}ar^{\sharp}(t) = \{x \in \mathcal{V}ar(t) \mid t \triangleright_{\mu} x\}$.

Definition 10 (Strongly Conservative). Let \mathcal{R} be a TRS and $\mu \in M_{\mathcal{R}}$. A rule $l \rightarrow r$ is strongly conservative if it is conservative and $\mathcal{V}ar^{\mu}(l) \cap \mathcal{V}ar^{\sharp}(l) = \mathcal{V}ar^{\mu}(r) \cap \mathcal{V}ar^{\sharp}(r) = \emptyset$; and \mathcal{R} is strongly conservative if all rules in \mathcal{R} are strongly conservative.

Linear CS-TRSs trivially satisfy $\mathcal{V}ar^{\mu}(l) \cap \mathcal{V}ar^{\sharp}(l) = \mathcal{V}ar^{\mu}(r) \cap \mathcal{V}ar^{\sharp}(r) = \emptyset$. Hence, linear conservative CS-TRSs are strongly conservative. Note that the CS-TRSs in Examples 1 and 3 are not strongly conservative.

Theorem 2 below is the other main result of this paper. It shows that basic usable rules in Definition 4 can be used to improve proofs of termination of CSR for strongly conservative CS-TRSs. As discussed in Section 4, if we consider minimal $(\mathcal{R}, \mathcal{P}, \mu^{\sharp})$ -chains, then we deal with μ -terminating terms w.r.t. (\mathcal{R}, μ) . We know that any μ -replacing subterm is μ -terminating, but we do not know anything about non- μ -replacing subterms. However, dealing with strongly conservative CS-TRSs, we ensure that non- μ -replacing subterms cannot become μ -replacing after μ -rewriting(s) above them. Hence, we develop a new basic μ -interpretation $I'_{\mathcal{G}, \mu}$ where non- μ -replacing positions are not interpreted. In contrast to $I'_{\mathcal{G}, \mu}$ (but closer to $I_{\mathcal{G}}$) our new basic μ -interpretation is defined now for μ -terminating terms only.

Definition 11 (Basic μ -Interpretation). Let (\mathcal{F}, μ, R) be a CS-TRS and $\mathcal{G} \subseteq \mathcal{F}$. Let $>$ be an arbitrary total ordering over $\mathcal{T}(\mathcal{F}^{\sharp} \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ where \perp is a new constant symbol and \mathbf{g} is a new binary symbol. The basic μ -interpretation $I'_{\mathcal{G}, \mu}$ is a mapping from μ -terminating terms in $\mathcal{T}(\mathcal{F}^{\sharp}, \mathcal{X})$ to terms in $\mathcal{T}(\mathcal{F}^{\sharp} \cup \{\perp, \mathbf{g}\}, \mathcal{X})$ defined as follows:

$$I'_{\mathcal{G}, \mu}(t) = \begin{cases} t & \text{if } t \in \mathcal{X} \\ f(I'_{\mathcal{G}, \mu, f, 1}(t_1), \dots, I'_{\mathcal{G}, \mu, f, n}(t_n)) & \text{if } t = f(t_1 \dots t_n) \text{ and } f \notin \mathcal{G} \\ \mathbf{g}(f(I'_{\mathcal{G}, \mu, f, 1}(t_1), \dots, I'_{\mathcal{G}, \mu, f, n}(t_n)), t') & \text{if } t = f(t_1 \dots t_n) \text{ and } f \in \mathcal{G} \end{cases}$$

$$\text{where } I'_{\mathcal{G}, \mu, f, i}(t) = \begin{cases} I'_{\mathcal{G}, \mu}(t) & \text{if } i \in \mu(f) \\ t & \text{if } i \notin \mu(f) \end{cases}$$

$$t' = \text{order}(\{I'_{\mathcal{G}, \mu}(u) \mid t \hookrightarrow_{\mathcal{R}, \mu} u\})$$

$$\text{order}(T) = \begin{cases} \perp, & \text{if } T = \emptyset \\ \mathbf{g}(t, \text{order}(T - \{t\})) & \text{if } t \text{ is minimal in } T \text{ w.r.t. } > \end{cases}$$

It is easy to prove that the basic μ -interpretation is well-defined (finite) for all μ -terminating terms.

Lemma 2. *For each μ -terminating term t , the term $I'_{\mathcal{G},\mu}(t)$ is finite.*

For the proof of our next theorem, we need some auxiliary definitions and results.

Definition 12. *Let (\mathcal{R}, μ) be a CS-TRS and σ be a substitution and let $\mathcal{G} \subseteq \mathcal{F}$. We denote by $\sigma_{I'_{\mathcal{G},\mu}} : \mathcal{T}(\mathcal{F}, \mathcal{X}) \rightarrow \mathcal{T}(\mathcal{F}, \mathcal{X})$ a function that, given a term t replaces occurrences of $x \in \text{Var}(t)$ at position p in t by either $I'_{\mathcal{G},\mu}(\sigma(x))$ if $p \in \text{Pos}^\mu(t)$, or $\sigma(x)$ if $p \notin \text{Pos}^\mu(t)$.*

Proposition 1. *Let (\mathcal{R}, μ) be a CS-TRS and σ be a substitution and let $\mathcal{G} \subseteq \mathcal{F}$. Let t be a term such that $\text{Var}^\mu(t) \cap \text{Var}^{\neq}(t) = \emptyset$. Let $\bar{\sigma}_{I'_{\mathcal{G},\mu},t}$ be a substitution given by*

$$\bar{\sigma}_{I'_{\mathcal{G},\mu},t}(x) = \begin{cases} I'_{\mathcal{G},\mu}(\sigma(x)) & \text{if } x \in \text{Var}^\mu(t) \\ \sigma(x) & \text{otherwise} \end{cases}$$

Then, $\bar{\sigma}_{I'_{\mathcal{G},\mu},t}(t) = \sigma_{I'_{\mathcal{G},\mu}}(t)$.

The following theorem shows that we can safely consider the basic usable rules (with the π -rules) for proving termination of strongly conservative CS-TRSs.

Theorem 2. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mathcal{P} \subseteq \mathcal{P}^\sharp(\mathcal{F}, \mathcal{X})$, and $\mu \in M_{\mathcal{F}}$. If $\mathcal{P} \cup \mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P})$ is strongly conservative and there exists a μ -reduction pair (\succsim, \sqsupset) such that $\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi \subseteq \succsim$, $\mathcal{P} \subseteq \succsim$, and $\mathcal{P} \cap \sqsupset \neq \emptyset$. Let $\mathcal{P}_\sqsupset = \{u \rightarrow v \in \mathcal{P} \mid u \sqsupset v\}$. Then there are no infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^\sharp)$ -chains whenever there are no infinite minimal $(\mathcal{R}, \mathcal{P} \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains.*

Proof (Sketch). By contradiction. Assume that there exists an infinite minimal $(\mathcal{R}, \mathcal{P}, \mu^\sharp)$ -chain \mathcal{A} but there is no infinite minimal $(\mathcal{R}, \mathcal{P} \setminus \mathcal{P}_\sqsupset, \mu^\sharp)$ -chains. We can assume that there is a $\mathcal{P}' \subseteq \mathcal{P}$ such that \mathcal{A} has a tail \mathcal{B} where all pairs are used infinitely often:

$$t_1 \xrightarrow{\mathcal{R},\mu}^* u_1 \xrightarrow{\mathcal{P}'} t_2 \xrightarrow{\mathcal{R},\mu}^* u_2 \xrightarrow{\mathcal{P}'} \dots$$

After applying the basic μ -interpretation $I'_{\mathcal{G},\mu}$ we obtain an infinite $(\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi, \mathcal{P}', \mu^\sharp)$ -chain. Since all terms in the infinite $(\mathcal{R}, \mathcal{P}', \mu^\sharp)$ -chain are μ -terminating w.r.t. (\mathcal{R}, μ) , we can indeed apply the basic μ -interpretation $I'_{\mathcal{G},\mu}$. Let $i \geq 1$.

– If we consider the pair step $u_i \xrightarrow{\mathcal{P}'} t_{i+1}$ we can obtain the following sequence:

$$I'_{\mathcal{G},\mu}(u_i) \xrightarrow{\pi}^* \sigma_{I'_{\mathcal{G},\mu}}(l) \xrightarrow{\pi}^* \bar{\sigma}_{I'_{\mathcal{G},\mu},r}(l) \xrightarrow{\mathcal{P}'} \bar{\sigma}_{I'_{\mathcal{G},\mu},r}(r) = \sigma_{I'_{\mathcal{G},\mu}}(r) = I'_{\mathcal{G},\mu}(t_{i+1})$$

– If we consider the rewrite sequence $t_i \xrightarrow{\mathcal{R},\mu}^* u_i$. All terms in it are μ -terminating, then we get $I'_{\mathcal{G},\mu}(t_i) \xrightarrow{\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* I'_{\mathcal{G},\mu}(u_i)$.

So we obtain the infinite μ -rewrite sequence:

$$I'_{\mathcal{G},\mu}(t_1) \xrightarrow{\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* I'_{\mathcal{G},\mu}(u_1) \xrightarrow{\pi}^* \circ \xrightarrow{\mathcal{P}'} I'_{\mathcal{G},\mu}(t_2) \xrightarrow{\mathcal{U}_B(\mathcal{R}, \mu^\sharp, \mathcal{P}) \cup \pi}^* \dots$$

Using the premise of the theorem, it is transformed into an infinite sequence consisting of \succsim and infinitely many \sqsupset steps. Using the stability condition, this contradicts the well-foundedness of \sqsupset . \square

Example 6. (Continuing Examples 1, 2 and 5) Cycle C_1 is not strongly conservative, but cycles C_2 and C_3 are strongly conservative. Thus, we can use their basic usable rules. Cycle C_2 has no usable rules and we can easily find a polynomial interpretation to show the absence of infinite minimal $(\mathcal{R}, C_2, \mu^\#)$ -chains:

$$[\mathbf{s}](x) = x + 1 \quad [\mathbf{MINUS}](x, y) = y$$

The basic usable rules $\mathcal{U}_B(\mathcal{R}, \mu^\#, C_3)$ for C_3 are strongly conservative (see Example 2). The following polynomial interpretation proves the absence of infinite $(\mathcal{R}, C_3, \mu^\#)$ -chains:

$$[0] = 0 \quad [\mathbf{s}](x) = x + 1 \quad [\mathbf{minus}](x, y) = x \quad [\mathbf{QUOT}](x, y) = x$$

Since we dealt with cycle C_1 in Example 5, μ -termination of \mathcal{R} is proved. Until now, no tool for proving termination of *CSR* could find a proof for this \mathcal{R} in Example 1. Thanks to the results in this paper, which have been implemented in MU-TERM, we can easily prove μ -termination of \mathcal{R} now.

6 Experiments

The techniques described in the previous sections have been implemented as part of the tool MU-TERM [1, 21]. In order to make clear the real contribution of the new technique to the performance of the tool, we have implemented three different versions of MU-TERM: (1) a basic version without any kind of usable rules, (2) a second version implementing the results about usable rules described in [4], and (3) a final version that implements the usable rules described in this paper (we do not use the notion in [4] even if the TRS is conservative and innermost equivalent). Version (2) of MU-TERM proves termination of *CSR* as termination of innermost *CSR* when the TRS is orthogonal (see [4, 11]), 37 systems, and as termination of *CSR without usable rules* in the rest of cases. In order to keep the set of experiments simple (but still meaningful), we only use linear interpretations with coefficients in $\{0, 1\}$. The usual practice shows that this is already quite powerful (see [9] for recent benchmarks in this sense). The benchmarks have been executed in a completely automatic way with a timeout of 1 minute on each of the 90 examples in the Context-Sensitive Rewriting subcategory of the 2007 Termination Competition⁵. A complete report of our experiments can be found in:

<http://www.dsic.upv.es/~rgutierrez/muterm/rta08/benchmarks.html>

Table 1 summarizes our results. Our notion of usable rules works pretty well: we are able to prove 20 more examples than without any usable rules, and 12 more than with the restricted notion in [4]. Furthermore, a comparison over the 44 examples solved by all the three versions of MU-TERM, we see that version (3) of MU-TERM is 43% faster than (1) and 27% faster than (2) (see Table 2).

⁵ See <http://www.lri.fr/~marche/termination-competition/2007>

Table 1. Comparative among the three MU-TERM versions

Tool Version	Proved	Total Time	Average Time
No Usable Rules	44/90	6.11s	0.14s
Innermost Usable Rules	52/90	11.75s	0.23s
Usable Rules	64/90	8.91s	0.14s

Table 2. Comparative over the 44 examples

Tool Version	Proved	Total Time	Average Time
No Usable Rules	44/90	6.11s	0.14s
Innermost Usable Rules	44/90	5.03s	0.11s
Usable Rules	44/90	3.57s	0.08s

7 Conclusions

We have investigated how *usable rules* can be used to improve termination proofs of CSR when the (context-sensitive) dependency pairs approach is used to achieve the proof. In contrast to [4], the straightforward extension of the standard notion of usable rules (called here *basic* usable rules, see Definition 4) does not work for CSR even for the quite restrictive class of conservative (cycles of) CS-TRSs. We have shown how to adapt the notion of usable rules for their use with arbitrary CS-TRSs (Definition 8). Theorem 1 shows that the new notion of usable rules can be used in proofs of termination of CS-TRSs. Here, although the proof uses a transformation in the very same style than [14, 17], the definition of the transformation is quite different from the usual one in that it applies to arbitrary terms, not only terminating ones. To our knowledge, this is the first time that Gramlich’s transformation [15] is adapted and used in that way. We have also introduced the notion of strongly conservative rule and CS-TRS (Definition 10). Theorem 2 shows that basic usable rules can be used in proofs of termination involving strongly conservative cycles and rules. Although we follow the proof scheme in [14, 17], a number of subtleties have to be carefully addressed before getting a correct adaptation of the proof.

We have implemented our techniques as part of the tool MU-TERM [1, 21]. Our experiments show that usable rules are helpful to improve proofs of termination of CSR. Regarding the previous work on usable rules for innermost CSR [4], this paper provides a fully general definition which is not restricted to conservative systems. Actually, as we show in our experiments, our framework is more powerful in practice than trying to prove termination of CSR as innermost termination of CSR with the restricted notion of usable rules in [4]. Actually, our results provide a basis for refining the notion of usable rules in the *innermost* setting, thus hopefully allowing a generalization of the results in [4].

Finally, usable rules were an essential ingredient for MU-TERM in winning the context-sensitive subcategory of the 2007 competition of termination tools.

References

1. B. Alarcón, R. Gutiérrez, J. Iborra, and S. Lucas. Proving Termination of Context-Sensitive Rewriting with MU-TERM. *ENTCS*, 188:105–115, 2007.

2. B. Alarcón, R. Gutiérrez, and S. Lucas. Context-Sensitive Dependency Pairs. In *Proc. of FST&TCS'06*, volume 4337 of *LNCS*, pages 297–308, 2006. Springer.
3. B. Alarcón, R. Gutiérrez, and S. Lucas. Improving the Context-Sensitive Dependency Graph. *ENTCS*, 188:91–103, 2007.
4. B. Alarcón and S. Lucas. Termination of Innermost Context-Sensitive Rewriting Using Dependency Pairs. In *Proc. of FroCoS'07*, volume 4720 of *LNCS*, pages 73–87, 2007. Springer.
5. T. Arts and J. Giesl. Proving Innermost Normalisation Automatically. In *Proc. of RTA'97*, volume 1232 of *LNCS*, pages 157–171, 1997. Springer.
6. T. Arts and J. Giesl. Termination of Term Rewriting Using Dependency Pairs. *Theoretical Computer Science*, 236(1–2):133–178, 2000.
7. C. Borralleras. *Ordering-Based Methods for Proving Termination Automatically*. PhD thesis, Departament de Llenguatges i Sistemes Informàtics, UPC, 2003.
8. F. Durán, S. Lucas, J. Meseguer, C. Marché, and X. Urbain. Proving Operational Termination of Membership Equational Programs. *Higher-Order and Symbolic Computation*, to appear, 2008.
9. C. Fuhs, J. Giesl, A. Middeldorp, P. Schneider-Kamp, R. Thiemann, and H. Zankl. SAT Solving for Termination Analysis with Polynomial Interpretations. In *Proc. of SAT'07*, volume 4501 of *LNCS*, pages 340–354, 2007. Springer.
10. J. Giesl, T. Arts, and E. Ohlebusch. Modular Termination Proofs for Rewriting Using Dependency Pairs. *Journal of Symbolic Computation*, 34(1):21–58, 2002.
11. J. Giesl and A. Middeldorp. Innermost Termination of Context-Sensitive Rewriting. In *Proc. of DLT'02*, volume 2450 of *LNCS*, pages 231–244, 2003. Springer.
12. J. Giesl and A. Middeldorp. Transformation Techniques for Context-Sensitive Rewrite Systems. *Journal of Functional Programming*, 14(4):379–427, 2004.
13. J. Giesl, R. Thiemann, and P. Schneider-Kamp. The Dependency Pair Framework: Combining Techniques for Automated Termination Proofs. In *Proc. of LPAR'04*, volume 3452 of *LNAI*, pages 301–331, 2004. Springer.
14. J. Giesl, R. Thiemann, P. Schneider-Kamp, and S. Falke. Mechanizing and Improving Dependency Pairs. *Journal of Automatic Reasoning*, 37(3):155–203, 2006.
15. B. Gramlich. Generalized Sufficient Conditions for Modular Termination of Rewriting. *Applicable Algebra in Engineering, Communication and Computing*, 5:131–151, 1994.
16. R. Gutiérrez, S. Lucas, and X. Urbain. Usable Rules for Context-Sensitive Rewrite System, DSIC-II/03/08. Technical report, UPV, 2008.
17. N. Hirokawa and A. Middeldorp. Tyrolean Termination Tool: Techniques and Features. *Information and Computation*, 205(4):474–511, 2007.
18. S. Lucas. Context-Sensitive Computations in Functional and Functional Logic Programs. *Journal of Functional and Logic Programming*, 1998(1):1–61, 1998.
19. S. Lucas. Termination of on-demand rewriting and termination of OBJ programs. In *Proc. of PPDP'01*, pages 82–93, 2001. ACM Press.
20. S. Lucas. Context-Sensitive Rewriting Strategies. *Information and Computation*, 178(1):293–343, 2002.
21. S. Lucas. MU-TERM: A Tool for Proving Termination of Context-Sensitive Rewriting. In *Proc. of RTA'04*, volume 3091 of *LNCS*, pages 200–209, 2004. Springer. Available at <http://zenon.dsic.upv.es/muterm/>.
22. S. Lucas. Proving Termination of Context-Sensitive Rewriting by Transformation. *Information and Computation*, 204(12):1782–1846, 2006.
23. E. Ohlebusch. *Advanced Topics in Term Rewriting*. Springer, 2002.
24. X. Urbain. Modular & Incremental Automated Termination Proofs. *Journal of Automated Reasoning*, 32(4):315–355, 2004.