Function Calls at Frozen Positions in Termination of Context-Sensitive Rewriting*

Raúl Gutiérrez and Salvador Lucas

DSIC, Universitat Politècnica de València,
Camino de Vera S/N, 46022 Valencia, Spain
{rgutierrez,slucas}@dsic.upv.es

Abstract. *Context-sensitive rewriting* (CSR) is a variant of rewriting where only selected arguments of function symbols can be rewritten. Consequently, the subterm positions of a term are classified as either *active*, i.e., positions of subterms that *can be rewritten*; or *frozen*, i.e., positions that can*not*. Frozen positions can be used to denote subexpressions whose evaluation is *delayed* or just *forbidden*. A typical example is the *if-then-else* operator whose second and third arguments are not evaluated until the evaluation of the first argument yields either *true* or *false*. Imposing replacement restrictions can *improve* the termination behavior of rewriting-based computational systems. Termination of CSR has been investigated by several authors and a number of automatic tools are able to prove it. In this paper, we analyze how frozen subterms affect termination of CSR. This analysis helps us to improve our *Context-Sensitive Dependency Pair (CS-DP) framework* for automatically proving termination of CSR. We have implemented these improvements in our tool MU-TERM. The experiments show the power of the improvements in practice.

Keywords: context-sensitive rewriting, termination, dependency pairs

1 Introduction

During the *4th International Workshop on Rewriting Logic and its Applications, WRLA 2002*, a tutorial by the second author entitled *Context-Sensitive Rewriting Techniques for Programs With Strategy Annotations* was the starting point of a friendly cooperation with José Meseguer leading to multiple exchanges of students and people from the UIUC and the UPV, and to the development of fruitful joint work on Rewriting Logic, Maude, and, in general, the analysis, verification, and optimization of declarative programming languages.

* Partially supported by the EU (FEDER), MINECO project TIN 2013-45732-C4-1-P, and GV project PROMETEO/2011/052. Salvador Lucas' research was developed during a sabbatical year at the CS Dept. of the UIUC and was also partially supported by NSF grant CNS 13-19109. Raúl Gutiérrez is also partially supported by a Juan de la Cierva Fellowship from the Spanish MINECO, ref. JCI-2012-13528.

Actually, the idea of *strategy annotation* (where the list of arguments whose evaluation is *allowed* is explicitly given for each function symbol) originally introduced by José and other colleagues as part of the design of OBJ2 [11] anticipated the main ideas underlying the development of *Context-Sensitive Rewriting* for a rather different purpose[1]. On the basis of previous work in [23, 24], in the aforementioned tutorial *Context-Sensitive Rewriting* (CSR, [22]) was shown useful to model rewriting-based programming languages like CafeOBJ [12], ELAN [8], OBJ [15], and Maude [9] that are able to use such kind of strategies.

In CSR, we start with a pair $(\mathcal{R}, \mu)$ (often called a CS-TRS) consisting of a *Term Rewriting System* (TRS) $\mathcal{R}$ and a *replacement map* μ, i.e., a mapping from a signature $\mathcal{F}$ into natural numbers that satisfies $\mu(f) \subseteq \{1, \ldots, \mathsf{ar}(f)\}$ for each function symbol f in the signature $\mathcal{F}$, where $\mathsf{ar}(f)$ is the arity of f. Here, μ is used to *discriminate* the argument positions on which the rewrite steps are allowed. In this way, we can avoid undesired computations and (in many cases) obtain a terminating behavior for the TRS (with respect to the context-sensitive rewrite relation). Strategy annotations are still used in CafeOBJ and Maude. In Maude, actually, *frozen arguments* have been recently introduced as a powerful mechanism to avoid undesired reductions. Frozen arguments are even closer to CSR, as they are just the complement of the *replacing arguments* specified by a replacement map μ: the i-th argument of f is frozen iff $i \notin \mu(f)$.

Using CSR, we can easily model the evaluation of expressions which *avoid* or *delay* the evaluation of some of their arguments. Paramount examples are *if-then-else* expressions, some boolean operators (*and*/*or*) and *lazy cons* operators for list construction.

Example 1. The following TRS $\mathcal{R}$ [28] provides a definition of factorial

$$
\begin{array}{rlr@{\qquad}rlr}
0{+}x & \to x & (1) & \mathsf{zero}(0) & \to \mathsf{true} & (6)\\
\mathsf{s}(x){+}y & \to \mathsf{s}(x{+}y) & (2) & \mathsf{zero}(\mathsf{s}(x)) & \to \mathsf{false} & (7)\\
\mathsf{p}(\mathsf{s}(x)) & \to x & (3) & \mathsf{fact}(x) & \to \mathsf{if}(\mathsf{zero}(x), \mathsf{s}(0), x{*}\mathsf{fact}(\mathsf{p}(x))) & (8)\\
\mathsf{if}(\mathsf{true}, x, y) & \to x & (4) & 0{*}x & \to 0 & (9)\\
\mathsf{if}(\mathsf{false}, x, y) & \to y & (5) & \mathsf{s}(x){*}y & \to y{+}(x{*}y) & (10)
\end{array}
$$

With $\mu(\mathsf{if}) = \{1\}$ and $\mu(f) = \{1, \ldots, k\}$ for any other k-ary symbol f (i.e., the only *function symbol which is restricted by μ is* if*), we can advantageously use* CSR *for handling the* if-then-else *operator: the second and third arguments of an expression $\mathsf{if}(b, s, t)$ are not evaluated until the guard b is evaluated to* true *or* false*. Without the replacement map, $\mathcal{R}$ is nonterminating because $\mathsf{fact}(x)$ calls $\mathsf{fact}(\mathsf{p}(x))$, which then calls $\mathsf{fact}(\mathsf{p}(\mathsf{p}(x)))$ and so on. Thanks to the replacement restrictions, though, we can evaluate $\mathsf{fact}(\mathsf{s}^n(0))$ to obtain the factorial $\mathsf{s}^{n!}(0)$ of a number n (encoded as $\mathsf{s}^n(0)$) by using* CSR *as follows:*

$$\underline{\mathsf{fact}(\mathsf{s}^n(0))} \hookrightarrow_{(8),\mu} \mathsf{if}(\underline{\mathsf{zero}(\mathsf{s}^n(0))}, \mathsf{s}(0), \mathsf{s}^n(0){*}\mathsf{fact}(\mathsf{p}(\mathsf{s}^n(0)))) \hookrightarrow_{(7),\mu} \cdots$$

[1] The notion context-sensitive rewriting was developed as part of Lucas' Master Thesis (1994) to implement concurrent programming languages that, like the π-calculus, forbid reductions on some arguments of its operations.

This can be formally proved (see [22, 26] and also [19] for an account of the algebraic semantics of context-sensitive specifications). Note that $\mathsf{zero}(\mathsf{s}^n(\mathsf{0}))$ *is forced to be reduced first to either* true *or* false *before evaluating the* 'then' *or* 'else' *expression, thus avoiding undesired reductions until the guard is fully evaluated.*

Direct techniques and frameworks for proving termination of CSR have been developed [1, 3, 17]. But, in practice, proving termination of some CS-TRSs with certain lazy structures as the *if-then-else* in the example can be difficult. In fact, finding an automatic proof of Example 1, and other examples like [13, Example 1] or [10, Example 3.2.14] are open problems since 1997, 2003 or 2008, respectively. The reason why these problems cannot be proved terminating by existing termination tools lies in the lack of sufficiently precise models of how the evaluation of expressions is *delayed* in context-sensitive computations. In this paper, we revisit this problem to obtain easier and mechanizable proofs of termination.

After some preliminaries in Section 2, Section 3 analyzes the role of frozen subterms in infinite μ-rewrite sequences, Section 4 models the activation of delayed subexpressions. Section 5 revises the characterization of the termination of CSR. Sections 6 proposes a new notion of CS usable rules, the extended basic CS usables, that allows us to simplify termination proofs if the application conditions are satisfied, Section 7 shows the experimental evaluation and Section 8 concludes.

2 Preliminaries

See [7] and [22] for basics on term rewriting and CSR, respectively. Throughout the paper, $\mathcal{X}$ denotes a countable set of variables and $\mathcal{F}$ denotes a signature, i.e., a set of function symbols each having a fixed arity given by a mapping $\mathsf{ar} : \mathcal{F} \to \mathbb{N}$. The set of terms built from $\mathcal{F}$ and $\mathcal{X}$ is $\mathcal{T}(\mathcal{F},\mathcal{X})$. Terms are viewed as labeled trees in the usual way. The symbol labeling the root of the term s is denoted as $\mathsf{root}(s)$. Positions $p, q, \ldots$ are represented by chains of positive natural numbers used to address subterms of s. Given positions p, q, we denote their concatenation as $p.q$. We denote the empty chain by Λ. Positions are ordered by the standard prefix ordering: $p \leq q$ if $\exists q'$ such that $q = p.q'$. The set of positions of a term s is $\mathcal{P}os(s)$. If p is a position, and Q is a set of positions, $p.Q = \{p.q \mid q \in Q\}$. For a replacement map μ, the set of *active positions* $\mathcal{P}os^\mu(s)$ of $s \in \mathcal{T}(\mathcal{F},\mathcal{X})$ is: $\mathcal{P}os^\mu(s) = \{\Lambda\}$, if $s \in \mathcal{X}$ and $\mathcal{P}os^\mu(s) = \{\Lambda\} \cup \bigcup_{i \in \mu(\mathsf{root}(s))} i.\mathcal{P}os^\mu(s|_i)$, if $s \notin \mathcal{X}$. We write $s \trianglerighteq t$, t is a subterm of s, if there is $p \in \mathcal{P}os(s)$ such that $t = s|_p$ and $s \triangleright t$, t is a proper subterm of s, if $s \trianglerighteq t$ and $s \neq t$. Given a replacement map μ, we write $s \trianglerighteq_\mu t$, t is a μ-replacing subterm of s, if there is $p \in \mathcal{P}os^\mu(s)$ such that $t = s|_p$ and $s \triangleright_\mu t$, t is a proper μ-replacing subterm of s, if $s \trianglerighteq_\mu t$ and $s \neq t$. Moreover, we write $s \triangleright_{\not\mu} t$, t is a non-μ-replacing subterm of s, if there is a *frozen position* p, i.e. $p \in \mathcal{P}os^{\not\mu}(s)$ where $\mathcal{P}os^{\not\mu}(s) = \mathcal{P}os(s) - \mathcal{P}os^\mu(s)$, such that $t = s|_p$. Let $\mathcal{V}ar(s) = \{x \in \mathcal{X} \mid \exists p \in \mathcal{P}os(s), s|_p = x\}$, $\mathcal{V}ar^\mu(s) = \{x \in \mathcal{V}ar(s) \mid$

$\exists p \in \mathcal{P}os^{\mu}(s), s|_p = x\}$ and $\mathcal{V}ar^{\not\mu}(s) = \{x \in \mathcal{V}ar(s) \mid s \rhd_{\not\mu} x\}$. A *context* is a term $C \in \mathcal{T}(\mathcal{F} \cup \{\Box\}, \mathcal{X})$ with zero or more 'holes' $\Box$ (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 (where the number and location of the holes is clarified 'in situ') and $C[t_1, \ldots, t_n]$ to denote the term obtained by filling the holes of a context $C[\,]$ with terms $t_1, \ldots, t_n$. $C[\,] = \Box$ is called the *empty* context.

A rewrite rule is an ordered pair (ℓ, r), written $\ell \to r$, with $\ell, r \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, $\ell \notin \mathcal{X}$ and $\mathcal{V}ar(r) \subseteq \mathcal{V}ar(\ell)$. A TRS is a pair $\mathcal{R} = (\mathcal{F}, R)$ where R is a set of rewrite rules. 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 $\mathcal{D} = \{\mathsf{root}(\ell) \mid \ell \to r \in R\}$ and $\mathcal{C} = \mathcal{F} - \mathcal{D}$. Given a CS-TRS $(\mathcal{R}, \mu)$, we have $s \hookrightarrow_{\mathcal{R},\mu} t$ (alternatively $s \stackrel{p}{\hookrightarrow}_{\mathcal{R},\mu} t$ if we want to make the position explicit) if there are $\ell \to r \in \mathcal{R}$, $p \in \mathcal{P}os^{\mu}(s)$ and a substitution σ with $s|_p = \ell\sigma$ and $t = s[r\sigma]_p$. A CS-TRS $(\mathcal{R}, \mu)$ is *terminating* if $\hookrightarrow_{\mathcal{R},\mu}$ is well-founded.

3 Minimal Non-μ-Terminating Terms at Frozen Positions

In this section we investigate how frozen subterms affect termination of CSR. Our analysis is used in Section 4 to obtain a more precise model of termination of CSR using Context-Sensitive Dependency Pairs (CS-DPs, [3]). If a TRS $\mathcal{R}$ is nonterminating, then *terms* are either terminating or nonterminating. The subset $\mathcal{T}_\infty$ of *minimal* nonterminating terms consists of nonterminating terms whose proper subterms are all terminating. And the following observations are in order [20, 21]: (1) every nonterminating term s contains a subterm $t \in \mathcal{T}_\infty$, (2) $\mathsf{root}(t)$ is a defined symbol of $\mathcal{R}$, and (3) minimality is preserved under inner rewritings:

Lemma 1. *Let $\mathcal{R}$ be a TRS. For every term $s \in \mathcal{T}_\infty$, if $s \xrightarrow{>\Lambda}_{\mathcal{R}} t$ and t is nonterminating then $t \in \mathcal{T}_\infty$.*

In CSR, if a CS-TRS $(\mathcal{R}, \mu)$ is nonterminating, among non-μ-terminating terms we distinguish the subset $\mathcal{T}_{\infty,\mu}$ of *strongly minimal* non-μ-terminating terms, whose proper subterms are *all* μ-terminating. But unlike minimality for rewriting, strong minimality is *not* preserved under inner μ-rewritings.

Example 2. Consider the following TRS $\mathcal{R}$ [3, Example 3]:

$$\mathsf{a} \to \mathsf{c}(\mathsf{f}(\mathsf{a}))\ (11) \qquad\qquad \mathsf{f}(\mathsf{c}(x)) \to x\ (12)$$

together with $\mu(\mathsf{c}) = \emptyset$ and $\mu(\mathsf{f}) = \{1\}$, and the term $\mathsf{f}(\mathsf{a}) \in \mathcal{T}_{\infty,\mu}$. If we apply (11) to the proper subterm a, we obtain $\mathsf{f}(\mathsf{c}(\mathsf{f}(\mathsf{a}))) \notin \mathcal{T}_{\infty,\mu}$ because $\mathsf{f}(\mathsf{a})$ is a subterm of $\mathsf{f}(\mathsf{c}(\mathsf{f}(\mathsf{a})))$.

Unfortunately, strong minimality does *not* distinguish active and frozen positions and a result as Lemma 1 is not possible for strongly minimal terms. The set of *minimal* non-μ-terminating terms $\mathcal{M}_{\infty,\mu}$ consists of all non-μ-terminating terms whose proper subterms *at active positions* are all μ-terminating. Minimal non-μ-terminating terms are preserved under inner μ-rewritings, as we show in the following lemma.

Lemma 2. [3, Lemma 4] *Let $(\mathcal{R},\mu)$ be a CS-TRS. For all $s \in \mathcal{M}_{\infty,\mu}$, if $s \stackrel{>\Lambda}{\hookrightarrow}_{\mathcal{R},\mu} t$ and t is non-μ-terminating, then $t \in \mathcal{M}_{\infty,\mu}$.*

Furthermore, $\mathcal{T}_{\infty,\mu} \subseteq \mathcal{M}_{\infty,\mu}$. And now, $\mathsf{f}(\mathsf{c}(\mathsf{f}(\mathsf{a})))$ in Example 2 is minimal: $\mathsf{f}(\mathsf{c}(\mathsf{f}(\mathsf{a}))) \in \mathcal{M}_{\infty,\mu}$. The following result establishes that, given a minimal non-μ-terminating term, there are only two ways for an infinite μ-rewrite sequence to proceed.

Proposition 1. [3, Proposition 5] *Let $(\mathcal{R},\mu)$ be a CS-TRS. For all $s \in \mathcal{M}_{\infty,\mu}$, there exist a rewrite rule $\ell \to r \in \mathcal{R}$, a substitution σ and a term $u \in \mathcal{M}_{\infty,\mu}$ such that $s \stackrel{>\Lambda}{\hookrightarrow}{}^{*}_{\mathcal{R},\mu} \ell\sigma \stackrel{\Lambda}{\hookrightarrow}_{\ell\to r,\mu} r\sigma \trianglerighteq_\mu t$ and either (1) there is a nonvariable subterm u at an active position of r such that $t = u\sigma$, or (2) there is $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(\ell)$ such that $x\sigma \trianglerighteq_\mu t$.*

What Proposition 1 says is that minimal non-μ-terminating terms at frozen positions (as $\mathsf{f}(\mathsf{a})$ in $\mathsf{f}(\mathsf{c}(\mathsf{f}(\mathsf{a})))$) show up at active positions by means of migrating variables (a variable x is migrating in a rule $\ell \to r$ if $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(\ell)$, as x in rule (12)). If (1) happens, information about the shape of t is provided because it is partially introduced by an active subterm of r. This information is crucial to efficiently mechanize proofs of termination. But if (2) happens, information about the shape of t is *hidden* below a binding $x\sigma$ of the matching substitution σ. The frozen occurrence of x in the left-hand side ℓ of the rule is responsible for this information showing up later in the sequence. In the following, we analyze how minimal non-μ-terminating terms appear at frozen positions in infinite μ-rewrite sequences and how they evolve until getting activated by a migrating variable. Without loss of generality, in the following all the considered infinite μ-rewrite sequences start from strongly minimal non-μ-terminating terms.

Example 3. Consider the following non-μ-terminating TRS $\mathcal{R}$ [1, modified (I)]:

$$\mathsf{a} \to \mathsf{f}(\overline{\mathsf{g}(\mathsf{b})}) \quad (13) \qquad \mathsf{h}(\overline{x}) \to x \quad (15)$$
$$\mathsf{f}(\overline{x}) \to \mathsf{h}(\overline{\mathsf{c}(x)}) \quad (14) \qquad \mathsf{b} \to \mathsf{a} \quad (16)$$

with $\mu(\mathsf{g}) = \mu(\mathsf{c}) = \{1\}$ and $\mu(f) = \emptyset$ for all $f \in \mathcal{F} - \{\mathsf{g},\mathsf{c}\}$. Subexpressions at frozen positions are identified using the overbar. And consider the following infinite μ-rewrite sequence (Figure 1 shows it graphically, where shaded triangles are minimal non-μ-terminating terms[2]):

$$\underline{\mathsf{a}} \hookrightarrow_{(13),\mu} \underline{\mathsf{f}(\overline{\mathsf{g}(\mathsf{b})})} \hookrightarrow_{(14),\mu} \underline{\mathsf{h}(\overline{\mathsf{c}(\mathsf{g}(\mathsf{b}))})} \hookrightarrow_{(15),\mu} \mathsf{c}(\mathsf{g}(\underline{\mathsf{b}})) \hookrightarrow_{(16),\mu} \mathsf{c}(\mathsf{g}(\underline{\mathsf{a}})) \hookrightarrow_{(13),\mu} \cdots$$

[2] Note that minimal non-μ-terminating terms may contain minimal non-μ-terminating terms (at frozen positions, though). We use darker shades for such nested minimal non-μ-terminating terms.

As we can see in the sequence, $\mathsf{a} \in \mathcal{T}_{\infty,\mu}$, and the first μ-rewriting step introduces the minimal non-μ-terminating term b at a frozen position by using rule (13) which introduces the context $\mathsf{g}(\Box)$ where b is located. Afterwards, the context $\mathsf{c}(\Box)$ is inserted *above* term $\mathsf{g}(\mathsf{b})$ which is "*pushed down*" by the right-hand side of rule (14). Finally, the migrating variable x in rule (15) is *instantiated* (in the third step) to $\mathsf{c}(\mathsf{g}(\mathsf{b}))$. The application of rule (15) finally *activates* b, which is now *active* inside $\mathsf{c}(\mathsf{g}(\mathsf{b}))$.

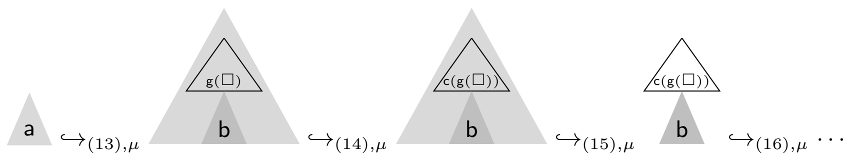

Fig. 1. Infinite μ-rewrite sequence in Example 3

Example 3 shows how minimal non-μ-terminating terms are partially "introduced" in an infinite μ-rewrite sequence: there is a rule $\ell \to r$ (in this case (13)), a subterm u of r at a frozen position (b) and a possible context with a hole at an active position ($\mathsf{g}(\Box)$).

As discussed above, the context surrounding those "hidden" minimal non-μ-terminating terms t can be "increased", i.e., t can be "pushed down" into a bigger context. Furthermore, the context can be "decreased" as well, as we can see in the following example.

Example 4. Consider the following TRS $\mathcal{R}$ [1, modified (II)]:

$$\mathsf{a} \to \mathsf{f}(\overline{\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b})))}) \quad (17) \qquad \mathsf{h}(\overline{\mathsf{c}(x)}) \to x \quad (19)$$

$$\mathsf{f}(\overline{\mathsf{g}(x)}) \to \mathsf{h}(\overline{x}) \quad (18) \qquad \mathsf{b} \to \mathsf{a} \quad (20)$$

with $\mu(\mathsf{g}) = \mu(\mathsf{c}) = \{1\}$ and $\mu(f) = \emptyset$ for all $f \in \mathcal{F} - \{\mathsf{g}, \mathsf{c}\}$, and:

$$\underline{\mathsf{a}} \hookrightarrow_{(17),\mu} \underline{\mathsf{f}(\overline{\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b})))})} \hookrightarrow_{(18),\mu} \underline{\mathsf{h}(\overline{\mathsf{c}(\mathsf{g}(\mathsf{b}))})} \hookrightarrow_{(19),\mu} \mathsf{g}(\underline{\mathsf{b}}) \hookrightarrow_{(20),\mu} \mathsf{g}(\underline{\mathsf{a}}) \hookrightarrow_{(17),\mu} \cdots$$

Figure 2 shows it graphically. Once again, the first μ-rewriting step introduces the minimal non-μ-terminating term b at a frozen position by using rule (17) which introduces the context $\mathsf{g}(\mathsf{c}(\mathsf{g}(\Box)))$. But, in the second μ-rewriting step, part of the active context $\mathsf{g}(\mathsf{c}(\mathsf{g}(\Box)))$ which is frozen at $s_2 = \mathsf{f}(\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b}))))$, i.e. $\mathsf{g}(\Box)$, is *removed* from s_2 due to *pattern matching* with the left-hand side of rule (18). Finally, in the same way, part of the active context $\mathsf{c}(\mathsf{g}(\Box))$ which is frozen at $s_3 = \mathsf{h}(\mathsf{c}(\mathsf{g}(\mathsf{b})))$, i.e. $\mathsf{c}(\Box)$, is removed from s_3 in the third μ-rewriting step by *pattern matching* with rule (19) and, furthermore, the migrating variable x is instantiated to $\mathsf{g}(\mathsf{b})$.

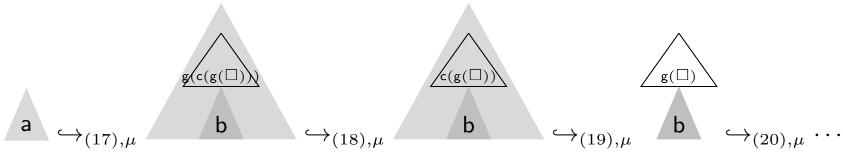

Fig. 2. Infinite μ-rewrite sequence in Example 4

We describe these "incoming" and "outcoming" contexts surrounding frozen subterms. First, we notice that, when examining the rules (14), (18) and (19) (which are responsible for the introduction and removal of contexts discussed in Example 3 and Example 4) they all share the following features:

- if a rule $\ell \to r$ adds a context C_i, then there is a term $s = C_i[x]_p$ such that $r = D[s]_q$, being q a frozen position of r and p an active position of s. Furthermore, if $\ell \to r$ is applied in a minimal non-μ-terminating sequence, the variable x cannot occur at active positions, i.e., $x \in (\mathcal{V}ar^{\not\mu}(\ell) \cap \mathcal{V}ar^{\not\mu}(r)) - (\mathcal{V}ar^{\mu}(\ell) \cup \mathcal{V}ar^{\mu}(r))$ (if not, minimality is violated); and,
- if a rule $\ell \to r$ removes a context C_o, then there is a term $s = C_o[x]_p$ such that $\ell = D[s]_q$, being q a frozen position of ℓ and p an active position of s. Furthermore, if $\ell \to r$ is applied in a minimal non-μ-terminating sequence, the variable x cannot occur at active positions, i.e., $x \in (\mathcal{V}ar^{\not\mu}(\ell) \cap \mathcal{V}ar^{\not\mu}(r)) - (\mathcal{V}ar^{\mu}(\ell) \cup \mathcal{V}ar^{\mu}(r))$ or $\ell|_{q.p}$ is migrating (in this case, we are in the second case of Proposition 1, where the minimal non-μ-terminating term shows up and is the responsible of continuing the sequence).

Rules involving these incoming and outcoming contexts can be applied several times and in different orders.

Example 5. Consider the following TRS $\mathcal{R}$ [1, modified (III)]:

$$\mathsf{a} \to \mathsf{f}(\overline{\mathsf{g}(\mathsf{b})}) \quad (21) \qquad \mathsf{h}(\overline{\mathsf{c}(x)}) \to x \quad (24)$$

$$\mathsf{f}(\overline{x}) \to \mathsf{h}(\overline{\mathsf{g}(\mathsf{c}(\mathsf{c}(x)))}) \quad (22) \qquad \mathsf{b} \to \mathsf{a} \quad (25)$$

$$\mathsf{h}(\overline{\mathsf{g}(x)}) \to \mathsf{h}(\overline{x}) \quad (23)$$

with $\mu(\mathsf{g}) = \mu(\mathsf{c}) = \{1\}$ and $\mu(f) = \emptyset$ for all $f \in \mathcal{F} - \{\mathsf{g}, \mathsf{c}\}$. And consider the following infinite μ-rewrite sequence (graphically in Figure 3):

$$\underline{\mathsf{a}} \hookrightarrow_{(21),\mu} \underline{\mathsf{f}(\overline{\mathsf{g}(\mathsf{b})})} \hookrightarrow_{(22),\mu} \underline{\mathsf{h}(\overline{\mathsf{g}(\mathsf{c}(\mathsf{c}(\mathsf{g}(\mathsf{b}))))})} \hookrightarrow_{(23),\mu} \underline{\mathsf{h}(\overline{\mathsf{c}(\mathsf{c}(\mathsf{g}(\mathsf{b})))})} \hookrightarrow_{(24),\mu} \mathsf{c}(\mathsf{g}(\underline{\mathsf{b}})) \cdots$$

Note that the migrating variable x is instantiated to term $x\sigma = C[u] = \mathsf{c}(\mathsf{g}(\mathsf{b}))$ where $u = \mathsf{b}$ is minimal non-μ-terminating and the context $C[\Box] = \mathsf{c}(\mathsf{g}(\Box))$ with a hole at an active position is a combination of *fragments* of contexts added at frozen positions by rewrite rules.

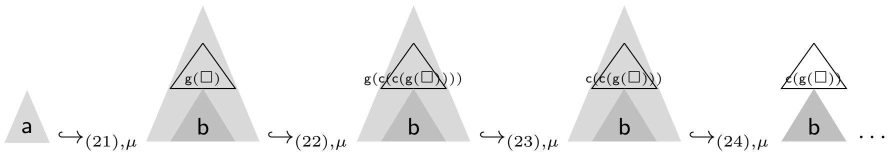

Fig. 3. Infinite μ-rewrite sequence in Example 5

4 Modeling the Unhiding Process using Rules

Recapitulating Section 3, if we consider an infinite sequence starting from $s_1 \in \mathcal{T}_{\infty,\mu}$, following Proposition 1 we extract an infinite sequence of the form:

$$s_1 \overset{>\Lambda *}{\hookrightarrow}_{\mathcal{R},\mu} \ell_1\sigma \overset{\Lambda}{\hookrightarrow}_{\ell_1\to r_1,\mu} r_1\sigma \trianglerighteq_\mu s_2 \overset{>\Lambda *}{\hookrightarrow}_{\mathcal{R},\mu} \ell_2\sigma \overset{\Lambda}{\hookrightarrow}_{\ell_2\to r_2,\mu} r_2\sigma \trianglerighteq_\mu \cdots$$

where $s_i \in \mathcal{M}_{\infty,\mu}$, for all $i > 0$. If Proposition 1(2) is applied on step j, $j > 0$, we know that: (a) previously in the chain there is a rule (like (13), (17) and (21)) that introduces the minimal non-μ-terminating term in the sequence together with an active context, (b) there are rules that modify this active context (like (14), (18), (22) and (23)) and, finally, (c) rule $\ell_j \to r_j$ (like (15), (19) and (24)) shows up the minimal non-μ-terminating by means of a migrating variable x together with part of its active context, $x\sigma = C[u]$. In this section, we use the knowledge of the previous section to define a TRS that can be used to extract u from $C[u]$ by using a minimum set of rules. Furthermore, we introduce the new notion of *unhidable*. All this prepares the introduction of a *new notion of minimality* which is the basis of our new characterization of termination of CSR.

Following the observations in the previous section, we can get the patterns which introduce the minimal non-μ-terminating term at a frozen position in a μ-rewrite sequence together with its active context, as $\mathsf{g}(\mathsf{b})$ in rule (13) in Example 3 and in rule (21) in Example 5 and $\mathsf{g}(\mathsf{c}(\mathsf{b}))$ in rule (17) in Example 4.

Definition 1. *Let $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ be a TRS, $\ell \to r \in R$ and μ a replacement map on $\mathcal{F}$. We say that $s = C[t]_p$ is a* raw hidden term *of $\ell \to r$ if $r = D[C[t]_p]_q$, $q \in \mathcal{P}os^{\not\mu}(r)$, $p \in \mathcal{P}os^{\mu}(C[t]_p)$, $\mathsf{root}(t) \in \mathcal{D}$ and $q.p$ is minimal in r (i.e., there is no $q'.p'$ such that $r = D'[C'[t]_{p'}]_{q'}$, $q' \in \mathcal{P}os^{\not\mu}(r)$, $p' \in \mathcal{P}os^{\mu}(C'[t]_{p'})$ and $p' < p$). Let $\mathsf{H}_{\mathsf{raw}}(\mathcal{R}, \mu)$ be the set of all raw hidden terms from rules in $(\mathcal{R}, \mu)$.*

Example 6. In Example 1, we have $\mathsf{H}_{\mathsf{raw}}(\mathcal{R}, \mu) = \{x * \mathsf{fact}(\mathsf{p}(x))\}$; in Example 3, we have $\mathsf{H}_{\mathsf{raw}}(\mathcal{R}, \mu) = \{\mathsf{g}(\mathsf{b})\}$; in Example 4, we have $\mathsf{H}_{\mathsf{raw}}(\mathcal{R}, \mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b})))\}$; and, in Example 5, we have $\mathsf{H}_{\mathsf{raw}}(\mathcal{R}, \mu) = \{\mathsf{g}(\mathsf{b}))\}$.

We identify the shape of the patterns that increase or decrease the active context attached to delayed subexpressions.

Definition 2. *Let* $u \in \mathcal{T}(\mathcal{F},\mathcal{X})$ *and* μ *a replacement map on* $\mathcal{F}$*. We say that* $s = C[\Box]_p$ *is a* maximal active hiding context *in* u *if* $u = D[C[x]_p]_q$, $q \in \mathcal{P}os^{\not\mu}(u)$, $p \in \mathcal{P}os^{\mu}(C[x]_p)$ *and* $q.p$ *is minimal in* u.

Example 7. In rule (14), $\mathsf{c}(\Box)$ is a maximal active hiding contex of the right-hand side, in rule (18), $\mathsf{g}(\Box)$ is an maximal active hiding context of the left-hand side and in rule (19), $\mathsf{c}(\Box)$ is a maximal active hiding context of the left-hand side.

And we clasify the different maximal active hiding contexts existing in a CS-TRS.

Definition 3. *Let* $\mathcal{R} = (\mathcal{F}, R)$ *be a TRS,* $\ell \to r \in R$, μ *a replacement map on* $\mathcal{F}$, $D[\Box]_q$ *a context with a hole at a frozen position* q, $C[\Box]_p$ *a context with a hole at an active position and* $x \in \mathcal{X}$*. We say that* $s = C[\Box]_p$ *is either:*

1. *An* incoming context *of* $\ell \to r$ *if* s *is a maximal active hiding contex of* r, $r = D[C[x]_p]_q$, *and* $x \in (\mathcal{V}ar^{\not\mu}(\ell) \cap \mathcal{V}ar^{\not\mu}(r)) - (\mathcal{V}ar^{\mu}(\ell) \cup \mathcal{V}ar^{\mu}(r))$.
2. *An* outcoming context *of* $\ell \to r$ *if* s *is a maximal active hiding contex of* ℓ, $\ell = D[C[x]_p]_q$, *and* $x \in (\mathcal{V}ar^{\not\mu}(\ell) \cap \mathcal{V}ar^{\not\mu}(r)) - (\mathcal{V}ar^{\mu}(\ell) \cup \mathcal{V}ar^{\mu}(r))$.
3. *A* terminal outcoming context *of* $\ell \to r$ *if* s *is a maximal active hiding context of* ℓ, $\ell = D[C[x]_p]_q$, *and* $x \in \mathcal{V}ar^{\mu}(r) - \mathcal{V}ar^{\mu}(\ell)$.

Let $\mathsf{C_i}(\mathcal{R},\mu)/\mathsf{C_o}(\mathcal{R},\mu)/\mathsf{C_t}(\mathcal{R},\mu)$ *be the set of all incoming / outcoming / terminal outcoming contexts from rules in* $(\mathcal{R},\mu)$.

Example 8. In rule (14), $\mathsf{c}(\Box)$ is an incoming contex of the right-hand side, in rule (18), $\mathsf{g}(\Box)$ is an outcoming context of the left-hand side and in rule (19), $\mathsf{c}(\Box)$ is a terminal outcoming context of the left-hand side.

In Example 1, we have $\mathsf{C_i}(\mathcal{R},\mu) = \mathsf{C_o}(\mathcal{R},\mu) = \mathsf{C_t}(\mathcal{R},\mu) = \emptyset$; in Example 3, we have $\mathsf{C_i}(\mathcal{R},\mu) = \{\mathsf{c}(\Box)\}$, and $\mathsf{C_o}(\mathcal{R},\mu) = \mathsf{C_t}(\mathcal{R},\mu) = \emptyset$; in Example 4, we have $\mathsf{C_i}(\mathcal{R},\mu) = \emptyset$, $\mathsf{C_o}(\mathcal{R},\mu) = \{\mathsf{g}(\Box)\}$, and $\mathsf{C_t}(\mathcal{R},\mu) = \{\mathsf{c}(\Box)\}$; and, in Example 5, we have $\mathsf{C_i}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{c}(\Box)))\}$, $\mathsf{C_o}(\mathcal{R},\mu) = \{\mathsf{g}(\Box)\}$, and $\mathsf{C_t}(\mathcal{R},\mu) = \{\mathsf{c}(\Box)\}$.

Outcoming contexts represent the fragments of active contexts which can be *removed* by a rule. Incoming contexts represent the active contexts that can be *added*. The following fixed-point definition obtains any combination of added/removed contexts (this will allow us to model the contexts that appear in the infinite μ-rewrite sequence in Example 5).

Definition 4. *Let* $\mathcal{R} = (\mathcal{F}, R)$ *be a TRS and* $\mu \in M_{\mathcal{F}}$*. The set* $\mathsf{XC_i}(\mathcal{R},\mu)$ *and* $\mathsf{XC_o}(\mathcal{R},\mu)$ *are the least sets satisfying:*

1. $\mathsf{C_i}(\mathcal{R},\mu) \subseteq \mathsf{XC_i}(\mathcal{R},\mu)$, $\mathsf{C_o}(\mathcal{R},\mu) \subseteq \mathsf{XC_o}(\mathcal{R},\mu)$ *and* $\mathsf{C_t}(\mathcal{R},\mu) \subseteq \mathsf{XC_t}(\mathcal{R},\mu)$.
2. *If* $C_i[\Box] \in \mathsf{XC_i}(\mathcal{R},\mu)$, $C_o[\Box] \in \mathsf{XC_o}(\mathcal{R},\mu)$, *and there exist* $\theta = mgu(C_i[x], C_o[y])$ *(rename variables if necessary) where* x *and* y *are fresh variables, such that* $y\theta \notin \mathcal{X}$ *and* $y\theta = C_i'[x]$, *then* $C_i'[\Box] \in \mathsf{XC_i}(\mathcal{R},\mu)$.
3. *If* $C_o[\Box] \in \mathsf{XC_o}(\mathcal{R},\mu)$, $C_i[\Box] \in \mathsf{XC_i}(\mathcal{R},\mu)$, *and there exist* $\theta = mgu(C_o[x], C_i[y])$ *(rename variables if necessary) where* x *and* y *are fresh variables, such that* $y\theta \notin \mathcal{X}$ *and* $y\theta = C_o'[x]$, *then* $C_o'[\Box] \in \mathsf{XC_o}(\mathcal{R},\mu)$.

4. *If $C_t[\Box] \in \mathsf{XC_t}(\mathcal{R},\mu)$, $C_i[\Box] \in \mathsf{XC_i}(\mathcal{R},\mu)$, and there exist $\theta = mgu(C_t[x], C_i[y])$ (rename variables if necessary) where x and y are fresh variables, such that $y\theta \notin \mathcal{X}$ and $y\theta = C'_t[x]$, then $C'_t[\Box] \in \mathsf{XC_t}(\mathcal{R},\mu)$.*

Note that, when the most general unifier (mgu) is computed, terms do not share variables, so a variable renaming is applied if necessary. The computation of $\mathsf{XC_i}(\mathcal{R},\mu)$, $\mathsf{XC_o}(\mathcal{R},\mu)$ and $\mathsf{XC_t}(\mathcal{R},\mu)$ terminates (in each step, the resulting context is a instantiated fragment of one of the contexts that are unified).

Example 9. In Examples 1, 3 and 4, we have $\mathsf{XC_i}(\mathcal{R},\mu) = \mathsf{C_i}(\mathcal{R},\mu)$, $\mathsf{XC_o}(\mathcal{R},\mu) = \mathsf{XC_o}(\mathcal{R},\mu)$ and $\mathsf{XC_t}(\mathcal{R},\mu) = \mathsf{C_t}(\mathcal{R},\mu)$. In Example 5, we have $\mathsf{C_o}(\mathcal{R},\mu) = \mathsf{XC_o}(\mathcal{R},\mu)$, $\mathsf{C_t}(\mathcal{R},\mu) = \mathsf{XC_t}(\mathcal{R},\mu)$, but $\mathsf{XC_i}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{c}(\Box))), \mathsf{c}(\mathsf{c}(\Box))\}$. The context $\mathsf{c}(\mathsf{c}(\Box))$ represents a fragment of the active incoming context that remains after applying rule (22) and rule (23).

Terminal outcoming contexts can only be applied just before the minimal non-μ-terminating term shows up at an active position. Therefore, $\mathsf{FXC_i}(\mathcal{R},\mu)$ extends $\mathsf{XC_i}(\mathcal{R},\mu)$ obtaining the fragments of contexts obtained after removing the the terminal outcoming context.

Definition 5. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS and $\mu \in M_{\mathcal{F}}$. The set $\mathsf{FXC_i}(\mathcal{R},\mu)$ satisfies:*

1. *$\mathsf{XC_i}(\mathcal{R},\mu) \subseteq \mathsf{FXC_i}(\mathcal{R},\mu)$.*
2. *If $C_i[\Box] \in \mathsf{XC_i}(\mathcal{R},\mu)$, $C_t[\Box] \in \mathsf{XC_t}(\mathcal{R},\mu)$, and there exist $\theta = mgu(C_i[x], C_t[y])$ (rename variables if necessary) where x and y are fresh variables, such that $y\theta \notin \mathcal{X}$ and $y\theta = C[x]$, then $C[\Box] \in \mathsf{FXC_i}(\mathcal{R},\mu)$.*

Example 10. In Examples 1, 3 and 4, we have $\mathsf{FXC_i}(\mathcal{R},\mu) = \mathsf{XC_i}(\mathcal{R},\mu) = \mathsf{C_i}(\mathcal{R},\mu)$. In Example 5, $\mathsf{FXC_i}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{c}(\Box))), \mathsf{c}(\mathsf{c}(\Box)), \mathsf{c}(\Box)\}$. The context $\mathsf{c}(\Box)$ represents a final fragment of the active incoming context that remains after applying rule (24) (when the minimal non-μ-terminating term shows up at an active position).

In the same way, we apply the outcoming contexts to the raw hidden terms to obtain the possible shape of those terms when they show up by means of migrating variables.

Definition 6. *The set $\mathsf{XH_{raw}}$ is the least set satisfying* (1) $\mathsf{H_{raw}} \subseteq \mathsf{XH_{raw}}$, *and* (2) *if $C_i[t] \in \mathsf{XH_{raw}}$, $C_o[\Box] \in \mathsf{XC_o}(\mathcal{R},\mu)$ and there exist $\theta = mgu(C_i[t], C_o[x])$ where x is a fresh variable, such that $x\theta = C[t\theta]$, then $C[t\theta] \in \mathsf{XH_{raw}}$.*

The set $\mathsf{FXH_{raw}}$ satisfies (1) $\mathsf{XH_{raw}} \subseteq \mathsf{FXH_{raw}}$, *and* (1) *if $C_i[t] \in \mathsf{XH_{raw}}$, $C_t[\Box] \in \mathsf{XC_t}(\mathcal{R},\mu)$ and there exist $\theta = mgu(C_i[t], C_t[x])$ where x is a fresh variable, such that $x\theta = C[t\theta]$, then $C[t\theta] \in \mathsf{FXH_{raw}}$.*

Example 11. In Examples 1 and 3, $\mathsf{FXH_{raw}}(\mathcal{R},\mu) = \mathsf{XH_{raw}}(\mathcal{R},\mu) = \mathsf{XH_{raw}}(\mathcal{R},\mu)$; in Example 4, we have $\mathsf{XH_{raw}}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b}))), \mathsf{c}(\mathsf{g}(\mathsf{b}))\}$ and $\mathsf{FXH_{raw}}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{c}(\mathsf{g}(\mathsf{b}))), \mathsf{c}(\mathsf{g}(\mathsf{b})), \mathsf{g}(\mathsf{b})\}$; and, in Example 5, $\mathsf{XH_{raw}}(\mathcal{R},\mu) = \mathsf{FXH_{raw}}(\mathcal{R},\mu) = \{\mathsf{g}(\mathsf{b}), \mathsf{b}\}$.

Previous definitions will be helpful in the next section to obtain a notion of minimality that gives us more information about non-μ-terminating terms at frozen positions.

4.1 A new notion of minimal non-μ-terminating term

The following notion of *unhidable* prepares a notion of *minimality* that provides more information about minimal non-μ-terminating terms at *frozen* positions.

Definition 7. *Let $\mathcal{R} = (\mathcal{F}, R)$ and $\mathcal{S} = (\mathcal{F}, S_0 \uplus S_1)$ be TRSs, and $\mu \in M_{\mathcal{F}}$. Let $s, t \in \mathcal{T}(\mathcal{F}, \mathcal{X})$. We say that s unhides t using $\mathcal{S}$ if $s \xrightarrow{\Lambda}{}^{*}_{S_0} \circ \xrightarrow{\Lambda}_{S_1} t$. We say that a term u is unhidable using $\mathcal{S}$ if for every subterm $v \in \mathcal{M}_{\infty,\mu}$ such that $u = D[C[v]_p]_q$, $q \in \mathcal{P}os^{\not\mu}(u)$, $p \in \mathcal{P}os^{\mu}(C[v]_p)$, $q.p$ minimal, $C[v]_p$ unhides v using $\mathcal{S}$ and v is unhidable using $\mathcal{S}$.*

Setting $S_0 = \mathsf{FXCR_i}(\mathcal{R}, \mu)$ and $S_1 = \mathsf{FXHR_{raw}}(\mathcal{R}, \mu)$ in Definition 7, where we define $\mathsf{FXCR_i}(\mathcal{R}, \mu) = \{C_i[x] \to x \mid C_i[\Box] \in \mathsf{FXC_i}(\mathcal{R}, \mu)\}$ and $\mathsf{FXHR_{raw}}(\mathcal{R}, \mu) = \{C_i[s]_p \to s \mid C_i[s]_p \in \mathsf{FXH_{raw}}(\mathcal{R}, \mu), p \in \mathcal{P}os^{\mu}(C_i[s]_p), s \in \mathcal{D}\}$, we obtain the following properties.

Proposition 2. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, $S_0 = \mathsf{FXCR_i}(\mathcal{R}, \mu)$, $S_1 = \mathsf{FXHR_{raw}}(\mathcal{R}, \mu)$, $\mathcal{S} = (\mathcal{F}, S_0 \uplus S_1)$, σ be a substitution and u, v be terms such that u unhides v using $\mathcal{S}$. Then,*

1. *$S_0 \cap S_1 = \emptyset$.*
2. *If $C_o[\Box] \in \mathsf{XC_o}(\mathcal{R}, \mu) \cup \mathsf{XC_t}(\mathcal{R}, \mu)$, and $u = C_o\sigma[C[v]]$ then $C[v]$ unhides v using $\mathcal{S}$.*
3. *If $C_i[\Box] \in \mathsf{XC_i}(\mathcal{R}, \mu)$, then $C_i\sigma[u]$ unhides v using $\mathcal{S}$.*

Proof. Item 1 is true by construction, since for all $s \to t \in S_0$, $t \in \mathcal{X}$ and for all $s \to t \in S_1$, $t \notin \mathcal{X}$. Since u unhides v using $\mathcal{S}$, we can write $u \xrightarrow{\Lambda}{}^{*}_{S_0} \circ \xrightarrow{\Lambda}_{S_1} v$. We prove Item 2 by induction on the length i of $\xrightarrow{\Lambda}{}^{i}_{S_0}$, we have:

1. If $i = 0$, we have $u \xrightarrow{\Lambda}_{s \to t} v$, where $s \to t \in S_1$. If $u = C_o\sigma[C[v]]$, there exist $\theta = mgu(s, C_o[x])$, $x\theta \trianglerighteq_{\mu} t\theta$ and $v = t\theta\sigma'$. By Definition 6, $x\theta \to t\theta \in S_1$ and $C[v] \xrightarrow{\Lambda}_{x\theta \to t\theta} v$, i.e. $C[v]$ unhides v using $\mathcal{S}$.
2. If $i > 0$, we have $u \xrightarrow{\Lambda}_{s \to t} w \xrightarrow{\Lambda}{}^{i-1}_{S_0} \circ \xrightarrow{\Lambda}_{S_1} v$, where $s \to t \in S_0$ and $s = C_i[x]$. If $u = C_o\sigma[C[v]]$, there exist $\theta = mgu(C_i[x], C_o[y])$. We consider two cases:
   (a) If $x\theta = C'_o[y]$, then by Definition 4, if $C_o[\Box] \in \mathsf{XC_o}(\mathcal{R}, \mu)$ then $C'_o[\Box] \in \mathsf{XC_o}(\mathcal{R}, \mu)$ and if $C_o[\Box] \in \mathsf{XC_t}(\mathcal{R}, \mu)$ then $C'_o[\Box] \in \mathsf{XC_t}(\mathcal{R}, \mu)$, $w = C'_o\sigma[C[v]]$ and, by the induction hypothesis, $C[v]$ unhides v using $\mathcal{S}$.
   (b) If $y\theta = C'_i[x]$, then by Definition 4, $C'_i[\Box] \in \mathsf{XC_i}(\mathcal{R}, \mu)$, $C'_i[x] \to x \in S_0$, $C[v] = C'_i\sigma'[w]$, and $C'_i\sigma'[w] \xrightarrow{\Lambda}_{C_i[x] \to x} w \xrightarrow{\Lambda}{}^{i-1}_{S_0} \circ \xrightarrow{\Lambda}_{S_1} v$, i.e. $C[v]$ unhides v using $\mathcal{S}$.

Item 3 is trivial.

The following lemma is an auxiliary result to prove Proposition 3.

Lemma 3. *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, $S_0 = \mathsf{FXCR_i}(\mathcal{R}, \mu)$, $S_1 = \mathsf{FXHR_{raw}}(\mathcal{R}, \mu)$, $\mathcal{S} = (\mathcal{F}, S_0 \uplus S_1)$, σ be a substitution and s is μ-terminating. If s is unhidable using $\mathcal{S}$ and $s \hookrightarrow_{\mathcal{R},\mu} t$ and t is μ-terminating then t is unhidable using $\mathcal{S}$.*

Proof. First of all, note that if t has no subterm u such that $t \rhd_{\not\mu} u$ and $u \in \mathcal{M}_{\infty,\mu}$ then the proposition is vacuously true. By induction on the position of $s \overset{p}{\hookrightarrow}_{\ell\to r,\mu} t$. If $p = \Lambda$ then $s = \ell\sigma$ and $t = r\sigma$ and $r\sigma$ is μ-terminating. If $r\sigma = D[C[u]_p]_q$, $q \in \mathcal{P}os^{\not\mu}(r\sigma)$, $p \in \mathcal{P}os^{\mu}(C[u]_p)$, $q.p$ is minimal and $u \in \mathcal{M}_{\infty,\mu}$ we have two possibilities:

- if there is no variable $x \in \mathcal{V}ar(r)$ such that $x\sigma \unrhd u$ then $r = D'[C'[u']_p]_q$ and $C[u]_p = C'\sigma[u'\sigma]_p$. Therefore, $C'[u'] \in \mathsf{H}_{\mathsf{raw}}(\mathcal{R},\mu)$, $C'[u'] \to u' \in \mathsf{FXHR}_{\mathsf{raw}}(\mathcal{R},\mu)$ and $C[u]$ unhides u using $\mathcal{S}$ ($C'[u'] \to u' \in S_1$);
- if there is a variable $x \in \mathcal{V}ar(r)$ such that $x\sigma \unrhd u$, we have $\ell\sigma \rhd_{\not\mu} u$. Let $\ell\sigma = D'[C'[u]_{p'}]_{q'}$, $q \in \mathcal{P}os^{\not\mu}(\ell\sigma)$, $p \in \mathcal{P}os^{\mu}(C'[u]_{p'})$ and $q'.p'$ minimal.
  - If $q' \notin \mathcal{P}os^{\not\mu}(\ell)$ then we have $\ell = D''[x]_{q''}$, $q'' < q'$, $q' = q''.q'''$, $x\sigma = D'''[C'[u]_{p'}]_{q'''}$, $q''' \in \mathcal{P}os^{\not\mu}(x\sigma)$ and, by definition, $C'[u]_{p'}$ unhides u using $\mathcal{S}$. Therefore, $r = D''''[x]_{q''''}$, $r\sigma = D''''[D'''[C'[u]_{p'}]_{q'''}]_{q''''}$, $q''''.q''' = q$, $p' = p$, $C'[\Box] = C[\Box]$ and $C[u]$ unhides u using $\mathcal{S}$.
  - If $q' \in \mathcal{P}os^{\not\mu}(\ell)$ then we have $\ell = D''[C''[x]_{p''}]_{q'}$, $D''\sigma[\Box]_{q'} = D'[\Box]_{q'}$, $C''[\Box]_{p''} \in \mathsf{XC_o}(\mathcal{R},\mu)$ and $C'[u]_{p'} = C''\sigma[C'''[u]_{p'''}]_{p''}$. Applying Proposition 2(2), $C'''[u]_{p'''}$ unhides u using $\mathcal{S}$. Therefore, $r = D'''[C''''[x]_{p''''}]_{q'''}$, $q''' = q$, $D'''\sigma[\Box]_{q'''} = D[\Box]_q$, $C''''[\Box]_{p''''} \in \mathsf{XC_i}(\mathcal{R},\mu)$ and, hence, $C[u]_p = C''''\sigma[C'''[u]_{p'''}]_{p''''}$. By Proposition 2(3), $C''''\sigma[C'''[u]_{p'''}]_{p''''}$ unhides u using $\mathcal{S}$.

If $p \neq \Lambda$ then $s = f(s_1,\ldots,s_i,\ldots,s_n)$, $t = f(s_1,\ldots,t_i,\ldots,t_n$ and $s_i \hookrightarrow_{\ell\to r,\mu} t_i$ where $1 \leq i \leq n$. By definition, s_j is unhidable using $\mathcal{S}$ for all $1 \leq j \leq n$ and applying the induction hypothesis t_i and t are unhidable using $\mathcal{S}$. □

We are ready now to introduce our new notion of minimality.

Definition 8 (Unhidable minimal term). *Let $\mathcal{R} = (\mathcal{F}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, $S_0 = \mathsf{FXCR_i}(\mathcal{R},\mu)$, $S_1 = \mathsf{FXHR}_{\mathsf{raw}}(\mathcal{R},\mu)$ and $\mathcal{S} = (\mathcal{F}, S_0 \uplus S_1)$. We define the set of* unhidable minimal non-μ-terminating *terms $\mathcal{M}^*_{\infty,\mu}$ as follows: $s \in \mathcal{M}^*_{\infty,\mu}$ iff $s \in \mathcal{M}_{\infty,\mu}$ and s is unhidable using $\mathcal{S}$.*

The following result improves Proposition 1 by using then new notion of minimal non-μ-terminating term.

Proposition 3. *Let $\mathcal{R} = (\mathcal{F}, R) = (\mathcal{C} \uplus \mathcal{D}, R)$ be a TRS, $\mu \in M_{\mathcal{F}}$, $S_0 = \mathsf{FXCR_i}(\mathcal{R},\mu)$, $S_1 = \mathsf{FXHR}_{\mathsf{raw}}(\mathcal{R},\mu)$ and $\mathcal{S} = (\mathcal{F}, S_0 \uplus S_1)$. Then for all $t \in \mathcal{M}^*_{\infty,\mu}$, there exist $\ell \to r \in R$, a substitution σ and a term $u \in \mathcal{M}^*_{\infty,\mu}$ such that $t \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \ell\sigma \overset{\Lambda}{\hookrightarrow} r\sigma \unrhd_{\mu} u$ and either:*

1. *There is a nonvariable active subterm s of r, $r \unrhd_{\mu} s$, such that $\mathsf{root}(s) \in \mathcal{D}$ and $u = s\sigma$, or*
2. *There is $x \in \mathcal{V}ar^{\mu}(r) - \mathcal{V}ar^{\mu}(\ell)$ such that $x\sigma = C[u]$ for a possibly empty context $C[\Box]$ with a hole at an active position and $C[u]$ unhides u using $\mathcal{S}$.*

Proof. Consider an infinite μ-rewrite sequence starting from t. By definition of $\mathcal{M}^*_{\infty,\mu}$, all proper active subterms of t are μ-terminating, and for all maximal terms s such that $t \rhd_{\not\mu} s$, $s \trianglerighteq_\mu u$ and $u \in \mathcal{M}^*_{\infty,\mu}$ then s unhides u using $\mathcal{S}$. Therefore, t has an inner reduction to an instance $\ell\sigma$ of the left-hand side of a rule $l \to r$ of $\mathcal{R}$: $t \overset{>\Lambda}{\hookrightarrow}{}^*_{\mathcal{R},\mu} \ell\sigma \overset{\Lambda}{\hookrightarrow} r\sigma$ and $r\sigma$ is not μ-terminating. Thus, we can write $t = f(t_1, \ldots, t_k)$ and $\ell\sigma = f(\ell_1, \ldots, \ell_k)$ for some k-ary defined symbol f, and $t_i \hookrightarrow^* \ell_i\sigma$ for all i, $1 \leq i \leq k$. Since t is unhidable using $\mathcal{S}$ all t_i, where $i \in \mu(f)$, are unhidable using $\mathcal{S}$. Since all t_i are μ-terminating for $i \in \mu(f)$, by [3, Lemma 1] and Lemma 3, $\ell_i\sigma$ and all its active subterms are also μ-terminating, and $\ell_i\sigma$ is unhidable using $\mathcal{S}$. In particular, $y\sigma$ is μ-terminating for all active variables y in ℓ: $y \in \mathcal{V}ar^\mu(\ell)$. Since $r\sigma$ is non-μ-terminating, by [3, Proposition 5], it contains an active subterm $u \in \mathcal{M}_{\infty,\mu}$: $r\sigma \trianglerighteq_\mu u$, i.e., there is a position $p \in \mathcal{P}os^\mu(r\sigma)$ such that $r\sigma|_p = u$. We consider two cases:

1. If $p \in \mathcal{P}os(r)$ is a nonvariable position of r, then there is an active nonvariable subterm s of r (i.e., $p \in \mathcal{P}os^\mu(r)$ and $s = r|_p \notin \mathcal{X}$), such that $u = s\sigma$. If there exist $v \in \mathcal{M}_{\infty,\mu}$ such that $u \rhd_{\not\mu} v$, reasoning in an analogous way to Lemma 3, we obtain that u is unhidable using $\mathcal{S}$, i.e., $u \in \mathcal{M}^*_{\infty,\mu}$.
2. If $p \notin\in \mathcal{P}os(r)$, then there is a variable at an active position $q \in \mathcal{P}os^\mu(r)$ such that $q \leq p$. Let $x \in \mathcal{V}ar^\mu(r)$ be such that $r|_q = x$. Then, $x\sigma = C[u]_p$, and $x\sigma$ is not μ-terminating: since $u \in \mathcal{M}_{\infty,\mu}$ is not μ-terminating, by [3, Lemma 1], $x\sigma$ is not μ-terminating. Since $y\sigma$ is μ-terminating for all $y \in \mathcal{V}ar^\mu(\ell)$, we get that $x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(\ell)$. Let $\ell\sigma = D'[C'[u]_{p'}]_{q'}$, $q' \in \mathcal{P}os^{\not\mu}(\ell\sigma)$, $p' \in \mathcal{P}os^\mu(C'[u]_{p'})$ and $q'.p'$ minimal. We have $\ell = D''[C''[x]_{p''}]_{q'}$, $D''\sigma[\Box]_{q'} = D'[\Box]_{q'}$, $C''[\Box]_{p''} \in \mathsf{XC_t}(\mathcal{R}, \mu)$ and $C'[u]_{p'} = C''\sigma[C[u]_p]_{p''}$. Applying Proposition 2(2), $C[u]_p$ unhides u using $\mathcal{S}$. If there exist $v \in \mathcal{M}_{\infty,\mu}$ such that $u \rhd_{\not\mu} v$, reasoning in an analogous way to Lemma 3, we obtain that u is unhidable using $\mathcal{S}$, i.e., $u \in \mathcal{M}^*_{\infty,\mu}$.

$\Box$

5 From Minimal Terms to the CS-DP Framework

Dependency pairs [6] describe the *propagation* of minimal non-μ-terminating terms in non-terminating rewrite sequences. The notion of CS-DP is a consequence of Proposition 1. The notation $f^\sharp$ for a given symbol f means that f is *marked*. For $s = f(s_1, \ldots, s_n)$, we write $s^\sharp$ to denote the marked term $f^\sharp(s_1, \ldots, s_n)$. We often capitalize f and use F instead of $f^\sharp$ in our examples.

Definition 9 (Context-Sensitive Dependency Pairs [3]). *Given a CS-TRS $(\mathcal{R}, \mu)$, let $\mathsf{DP}(\mathcal{R}, \mu) = \mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu) \cup \mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu)$ the set of CS-DPs where $\mathsf{DP}_{\mathcal{F}}(\mathcal{R}, \mu) = \{\ell^\sharp \to s^\sharp \mid \ell \to r \in \mathcal{R}, r \trianglerighteq_\mu s, \mathsf{root}(s) \in \mathcal{D}, \ell \not\trianglerighteq_\mu s\}$, and $\mathsf{DP}_{\mathcal{X}}(\mathcal{R}, \mu) = \{\ell^\sharp \to x \mid \ell \to r \in \mathcal{R}, x \in \mathcal{V}ar^\mu(r) - \mathcal{V}ar^\mu(\ell)\}$. We extend μ into $\mu^\sharp$ by $\mu^\sharp(f) = \mu(f)$ if $f \in \mathcal{F}$, and $\mu^\sharp(f^\sharp) = \mu(f)$ if $f \in \mathcal{D}$.*

Example 12. *For $(\mathcal{R}, \mu)$ in Example 1, we obtain the following CS-DPs:*

$$\mathsf{s}(x) +^\sharp y \to x +^\sharp y \quad (26) \qquad \mathsf{FACT}(x) \to \mathsf{ZERO}(x) \quad (30)$$
$$\mathsf{s}(x) *^\sharp y \to y +^\sharp (x * y) \quad (27) \qquad \mathsf{IF}(\mathsf{true}, x, y) \to x \quad (31)$$
$$\mathsf{s}(x) *^\sharp y \to x *^\sharp y \quad (28) \qquad \mathsf{IF}(\mathsf{false}, x, y) \to y \quad (32)$$
$$\mathsf{FACT}(x) \to \mathsf{IF}(\mathsf{zero}(x), \mathsf{s}(0), x * \mathsf{fact}(\mathsf{p}(x))) \quad (29)$$

DPs (26)-(30) capture the direct function calls and collapsing DPs (31)-(32) capture the activation of delayed function calls.

As usual when dealing with DPs, we *abstract* the notion of chain using generic TRSs $\mathcal{P}$, $\mathcal{R}$ and $\mathcal{S}$. Termination of CS-TRSs is characterized by the absence of infinite chains of CS-DPs [2, 3].

Definition 10 (Chain of Pairs [17]). *Let $\mathcal{P}$, $\mathcal{R}$ and $\mathcal{S}$ be TRSs and μ a replacement map where $\mathcal{S} = \mathcal{S}_{\rhd_\mu} \uplus \mathcal{S}_\sharp$, $\mathcal{S}_{\rhd_\mu}$ are rules of the form $s \to t \in \mathcal{S}$ such that $s \rhd_\mu t$ and $\mathcal{S}_\sharp = \mathcal{S} - \mathcal{S}_{\rhd_\mu}$. A $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain is a finite or infinite sequence of pairs $u_i \to v_i \in \mathcal{P}$, together with a substitution σ satisfying that, for all $i \geq 1$,*

1. *If $v_i \notin \mathcal{V}ar(u_i) - \mathcal{V}ar^\mu(u_i)$, then $v_i\sigma = w_i \hookrightarrow^*_{\mathcal{R},\mu} u_{i+1}\sigma$, and*
2. *If $v_i \in \mathcal{V}ar(u_i) - \mathcal{V}ar^\mu(u_i)$, then $v_i\sigma \stackrel{\Lambda}{\to}^*_{\mathcal{S}_{\rhd_\mu}} \circ \stackrel{\Lambda}{\to}_{\mathcal{S}_\sharp} w_i \hookrightarrow^*_{\mathcal{R},\mu} u_{i+1}\sigma$.*

An infinite $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain is called minimal *if for all $i \geq 1$, w_i is $(\mathcal{R}, \mu)$-terminating.*

In Definition 10, $\mathcal{P}$ plays the role of $\mathsf{DP}(\mathcal{R}, \mu)$ and $\mathcal{S}$ has two components $\mathcal{S}_{\rhd_\mu}$ and $\mathcal{S}_\sharp$ which are useful to model the connection between a collapsing pair to another pair. The connection between the results obtained in the previous section and the notion of chain is straightforward, we only have to introduce the marking in our unhiding rules.

Definition 11 (Unhiding TRS). *Let $\mathcal{R}$ be a TRS and $\mu \in M_\mathcal{R}$. We define $\mathsf{unh}(\mathcal{R}, \mu) = \mathsf{unh}_{\rhd_\mu}(\mathcal{R}, \mu) \uplus \mathsf{unh}_\sharp(\mathcal{R}, \mu)$, where $\mathsf{unh}_{\rhd_\mu}(\mathcal{R}, \mu) = \mathsf{FXCR_i}(\mathcal{R}, \mu)$ and $\mathsf{unh}_\sharp(\mathcal{R}, \mu) = \{s \to t^\sharp \mid s \to t \in \mathsf{FXHR_{raw}}(\mathcal{R}, \mu)\}$.*

Example 13. *The unhiding TRS $\mathsf{unh}(\mathcal{R}, \mu)$ in Example 1 consists of the following rules:*

$$x * \mathsf{fact}(\mathsf{p}(x)) \to x *^\sharp \mathsf{fact}(\mathsf{p}(x)) \quad (33) \qquad x * \mathsf{fact}(\mathsf{p}(x)) \to \mathsf{FACT}(\mathsf{p}(x)) \quad (34)$$
$$x * \mathsf{fact}(\mathsf{p}(x)) \to \mathsf{P}(x) \quad (35)$$

where $\mathsf{FXC_i}(\mathcal{R}, \mu) = \emptyset$. In [17], the definition of the unhiding TRS is different. We would have the following bigger *set of rules:*

$$x * \mathsf{fact}(\mathsf{p}(x)) \to x *^\sharp \mathsf{fact}(\mathsf{p}(x)) \qquad x * y \to y$$
$$\mathsf{fact}(\mathsf{p}(x)) \to \mathsf{FACT}(\mathsf{p}(x)) \qquad \mathsf{fact}(x) \to x$$
$$\mathsf{p}(x) \to \mathsf{P}(x)$$

The following result provides a new characterization of termination of CSR.

Theorem 1. *Let $\mathcal{R}$ be a TRS and $\mu \in M_{\mathcal{R}}$. $\mathcal{R}$ is μ-terminating if and only if there is no infinite minimal* $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$*-chain.*

Proof. **Soundness**.

By contradiction. If $\mathcal{R}$ is not μ-terminating, then there is $t \in \mathcal{T}_{\infty,\mu}$. By Theorem [17, Theorem 1] and Proposition 3, there are rules $\ell_i \to r_i \in \mathcal{R}$, matching substitutions σ_i, and terms $t_i \in \mathcal{M}^*_{\infty,\mu}$, for $i \geq 1$ such that

$$t = t_0 \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \sigma_1(\ell_1) \overset{\Lambda}{\hookrightarrow} \sigma_1(r_1) \trianglerighteq_\mu t_1 \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \sigma_2(\ell_2) \overset{\Lambda}{\hookrightarrow} \sigma_2(r_2) \trianglerighteq_\mu t_2 \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \cdots$$

where either (D1) $t_i = s_i\sigma_i$ for some s_i such that $r_i \trianglerighteq_\mu s_i$ or (D2) $x_i\sigma_i = C_i[t_i]$ for some $x_i \in \mathcal{V}ar^\mu(r_i) - \mathcal{V}ar^\mu(\ell_i)$ and $C_i[t_i]$ unhides t_i using $\mathsf{FXCR_i}(\mathcal{R},\mu) \uplus \mathsf{FXHR_{raw}}(\mathcal{R},\mu)$, $\mathsf{root}(t'_i) \in \mathcal{D}$. Furthermore, since $t_{i-1} \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \ell_i\sigma_i$ and $t_{i-1} \in \mathcal{M}^*_{\infty,\mu}$ (in particular, $t_0 = t \in \mathcal{T}_{\infty,\mu} \subseteq \mathcal{M}^*_{\infty,\mu}$), $\ell_i\sigma_i \in \mathcal{M}^*_{\infty,\mu}$ 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 active subterms of t_i (hence of $t_i^\sharp$ as well) are μ-terminating and $\mathsf{root}(t^\sharp)$ 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 10. 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 define an infinite strongly minimal $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$-chain using CS-DPs $u_i \to v_i$ for $i \geq 1$ (note that if $t_1 \in \mathcal{T}_{\infty,\mu}$, then $t_1^\sharp$ satisfies the conditions of the first element in an infinite strongly minimal $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$-chain), where $u_i = \ell_i^\sharp$ and

1. $v_i = s_i^\sharp$ if (D1) holds. Since $t_i \in \mathcal{M}^*_{\infty,\mu}$, we have that $\mathsf{root}(s_i) \in \mathcal{D}$. Furthermore, if we assume that s_i is an active subterm of ℓ_i (i.e., $\ell_i \triangleright_\mu s_i$), then $\ell_i\sigma_i \triangleright_\mu s_i\sigma_i$ which (since $s_i\sigma_i = t_i \in \mathcal{M}^*_{\infty,\mu}$) contradicts that $\ell_i\sigma_i \in \mathcal{M}^*_{\infty,\mu}$. Thus, $\ell_i \not\triangleright_\mu s_i$. Hence, $u_i \to v_i \in \mathsf{DP}_\mathcal{F}(\mathcal{R},\mu)$. Furthermore, $t_i^\sharp = v_i\sigma_i$ is μ-terminating. Finally, since $t_i = s_i\sigma_i \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \ell_{i+1}\sigma_{i+1}$ 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 $v_i\sigma_i = s_i^\sharp\sigma_i \hookrightarrow^*_{\mathcal{R},\mu^\sharp} u_{i+1}\sigma_{i+1}$.
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 $x_i\sigma_i = C_i[t_i]$, we have that $v_i\sigma_i = C_i[t_i]$ and $C_i[t_i]$ unhides t_i using $\mathsf{unh}(\mathcal{R},\mu)$. By Proposition 3, we know that $\theta_i(C'_i)[t_i] \overset{\Lambda}{\to}^*_{\mathsf{unh}(\mathcal{R},\mu)_{\triangleright_\mu}} \circ \overset{\Lambda}{\to}_{\mathsf{unh}(\mathcal{R},\mu)_\sharp} t_i^\sharp$. Finally, since $t_i \overset{>\Lambda}{\hookrightarrow}^*_{\mathcal{R},\mu} \ell_{i+1}\sigma_{i+1}$, again we have that $t_i^\sharp \hookrightarrow^*_{\mathcal{R},\mu^\sharp} u_{i+1}\sigma_{i+1}$.

Regarding σ, w.l.o.g. we can assume that $\mathcal{V}ar(\ell_i) \cap \mathcal{V}ar(\ell_j) = \emptyset$ for all $i \neq j$, and therefore $\mathcal{V}ar(u_i) \cap \mathcal{V}ar(u_j) = \emptyset$ as well. Then, σ is given by $x\sigma = x\sigma_i$ whenever $x \in \mathcal{V}ar(u_i)$ for $i \geq 1$. From the discussion in points (1) and (2) above, we conclude that the CS-DPs $u_i \to v_i$ for $i \geq 1$ together with σ define an infinite strongly minimal $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$-chain which contradicts our initial assumption.

Completeness.

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

1. $v_i\sigma \hookrightarrow^*_{\mathcal{R},\mu^\sharp} u_{i+1}\sigma$, 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 $v_i\sigma \overset{\Lambda}{\to}^*_{\mathsf{unh}(\mathcal{R},\mu)_{\rhd_\mu}} \circ \overset{\Lambda}{\to}_{\mathsf{unh}(\mathcal{R},\mu)_\sharp} s_i \hookrightarrow^*_{\mathcal{R},\mu} u_{i+1}\sigma$.

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 $v_1^\natural$ is an active subterm of the right-hand-side r_1 of a rule $l_1 \to r_1$ in $\mathcal{R}$. Therefore, $r_1 = C_1[v_1^\natural]_{p_1}$ for some $p_1 \in \mathcal{P}os^\mu(r_1)$ and we can perform the μ-rewriting step $t_1 = u_1\sigma \hookrightarrow_{\mathcal{R},\mu} r_1\sigma = C_1\sigma[v_1^\natural\sigma]_{p_1} = s_1$, where $(v_1^\natural\sigma)^\sharp = v_1\sigma \hookrightarrow^*_{\mathcal{R},\mu^\sharp} u_2\sigma$ and $u_2\sigma$ initiates an infinite $(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mathsf{unh}(\mathcal{R},\mu),\mu^\sharp)$-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 $\ell_1 \to r_1$ in $\mathcal{R}$ such that $u_1 = \ell_1^\sharp$, and $x \in \mathcal{V}ar^\mu(r_1) - \mathcal{V}ar^\mu(\ell_1)$, i.e., $r_1 = C_1[x]_{q_1}$ for some $q_1 \in \mathcal{P}os^\mu(r_1)$. Furthermore, if $v_i\sigma \overset{\Lambda}{\to}^*_{\mathsf{unh}(\mathcal{R},\mu)_{\rhd_\mu}} \circ \overset{\Lambda}{\to}_{\mathsf{unh}(\mathcal{R},\mu)_\sharp} s_i$, this means that $v_i\sigma$ unhides $s_i^\natural$ using $\mathsf{unh}(\mathcal{R},\mu)$. Hence, $v_i\sigma \trianglerighteq_\mu s_1^\natural$, $s_1 \hookrightarrow^*_{\mathcal{R},\mu^\sharp} u_2\sigma$ and $u_2\sigma$ initiates an infinite $(\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mathsf{unh}(\mathcal{R},\mu),\mu^\sharp)$-chain. Note that $p_1 = q_1.p_1' \in \mathcal{P}os^\mu(s_1)$ where p_1' is the position of the hole in $C_1[\Box]_{p_1'}$.

Since $\mu^\sharp(f^\sharp) = \mu(f)$, and $p_1 \in \mathcal{P}os^\mu(s_1)$, we have that $s_1 \hookrightarrow^*_{\mathcal{R},\mu} t_2[u_2\sigma]_{p_1} = t_2$ and $p_1 \in \mathcal{P}os^\mu(t_2)$. Therefore, we can build in that way 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 $\mathcal{R}$. □

Example 14. For $(\mathcal{R},\mu)$ in Example 3, we obtain the following CS-DPs:

$$\mathsf{A} \to \mathsf{F}(\overline{\mathsf{g}(\mathsf{b})}) \quad (36) \qquad \mathsf{F}(\overline{x}) \to x \quad (38)$$

$$\mathsf{F}(\overline{x}) \to \mathsf{F}(\overline{\mathsf{c}(x)}) \quad (37) \qquad \mathsf{B} \to \mathsf{A} \quad (39)$$

The infinite sequence in Example 3 is captured by the following $(\mathcal{P},\mathcal{R},\mathcal{S},\mu^\sharp)$-chain, where $\mathcal{P} = \mathsf{DP}(\mathcal{R},\mu)$ and $\mathcal{S} = \mathsf{unh}(\mathcal{R},\mu)$:

$$\underline{\mathsf{A}} \to_{(36)} \underline{\mathsf{F}(\overline{\mathsf{g}(\mathsf{b})})} \to_{(37)} \underline{\mathsf{F}(\overline{\mathsf{c}(\overline{\mathsf{g}(\mathsf{b})})})} \to_{(38)} \underline{\mathsf{c}(\mathsf{g}(\mathsf{b}))} \overset{\Lambda}{\to}_{\mathcal{S}_{\rhd_\mu}} \underline{\mathsf{g}(\mathsf{b})} \overset{\Lambda}{\to}_{\mathcal{S}_\sharp} \underline{\mathsf{B}} \to_{(39)} \underline{\mathsf{A}} \to_{(36)} \cdots$$

5.1 Context-Sensitive Dependency Pair Framework

In the DP framework [14], the focus is on the so-called *termination problems* involving two TRSs $\mathcal{P}$ and $\mathcal{R}$ instead of just the 'target' TRS $\mathcal{R}$. In our setting we start with the following definition (see also [1, 3]).

Definition 12 (CS problem and processor). *A* CS problem τ *is a tuple* $\tau = (\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$*, where* $\mathcal{P}$*,* $\mathcal{R}$ *and* $\mathcal{S}$ *are TRSs, and* μ *is a replacement map on the signatures of* $\mathcal{R}$*,* $\mathcal{P}$ *and* $\mathcal{S}$*. The* CS problem $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$ *is* finite *if there is no infinite minimal* $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$*-chain.*

A CS processor Proc *is a mapping from CS problems into sets of CS problems. A CS-processor* Proc *is* sound *if for all CS problems* τ*,* τ *is finite whenever* $\forall \tau' \in \mathsf{Proc}(\tau)$*,* τ' *is finite*[3].

In order to prove the μ-termination of a TRS $\mathcal{R}$, we adapt the result from [14] to CSR.

Theorem 2 (CS-DP Framework [3]). *Let* $\mathcal{R}$ *be a TRS and* μ *a replacement map on the signature of* $\mathcal{R}$*. We construct a tree whose nodes are labeled with CS problems or* "yes"*, and whose root is labeled with* $(\mathsf{DP}(\mathcal{R}, \mu), \mathcal{R}, \mathsf{unh}(\mathcal{R}, \mu), \mu^\sharp)$*. For every inner node labeled with* τ*, there is a sound processor* Proc *satisfying one of the following conditions:*

1. $\mathsf{Proc}(\tau) = \emptyset$ *and the node has just one child, labeled with* "yes"*.*
2. $\mathsf{Proc}(\tau) \neq \mathsf{no}$*,* $\mathsf{Proc}(\tau) \neq \emptyset$*, 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.*

6 Usable Rules in the CS-DP Framework

One of the most powerful CS processors to deal with CS problems is the *μ-reduction pair processor*, a processor that discards pairs that can be strictly oriented using orderings. A *μ-reduction pair* $(\gtrsim, \sqsupset)$ consists of a stable and μ-monotonic[4] quasi-ordering $\gtrsim$, and a well-founded stable relation $\sqsupset$ on terms in $\mathcal{T}(\mathcal{F}, \mathcal{X})$ which are compatible, i.e., $\gtrsim \circ \sqsupset \subseteq \sqsupset$ or $\sqsupset \circ \gtrsim \subseteq \sqsupset$ [2]. Given a CS problem $\tau = (\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$, if there is a μ-reduction pair such that $\mathcal{P} \cup \mathcal{S} \subseteq \gtrsim \cup \sqsupset$ and $\mathcal{R} \subseteq \gtrsim$ then $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$ is finite if $(\mathcal{P} - \mathcal{P}_\sqsupset, \mathcal{R}, \mathcal{S} - \mathcal{S}_\sqsupset, \mu)$ is finite, where $\mathcal{P}_\sqsupset$ and $\mathcal{S}_\sqsupset$ represent the set of rules from $\mathcal{P}$ and $\mathcal{S}$ oriented using $\sqsupset$. The μ-reduction pair processor can be improved using the notion of *usable rule* [5]. Usable rules, initially connected to innermost termination, allow us to discard those rules from $\mathcal{R}$ that are not directly involved in (possible) infinite minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chains. In rewriting (and also in CSR), the notion of usable rule is connected with $\mathcal{C}_\varepsilon$-termination [16, 27]. A TRS $\mathcal{R} = (\mathcal{F}, R)$ is $\mathcal{C}_\varepsilon$-terminating if $\mathcal{R} \uplus \mathcal{C}_\varepsilon$ is terminating, where $\mathcal{C}_\varepsilon = \{\mathsf{c}(x, y) \to x, \, \mathsf{c}(x, y) \to y\}$ (with $\mathsf{c} \notin \mathcal{F}$). The idea behind the usable rules is that for every infinite minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain we can construct an infinite sequence where rewrite steps using $\mathcal{R}$ can be simulated by rewrite steps using $\mathcal{U}_\tau(\mathcal{R})$ and $\mathcal{C}_\varepsilon$, where $\mathcal{U}_\tau(\mathcal{R})$ is the set of usable rules of τ. So, instead of $\mathcal{R} \subseteq \gtrsim$, we only need to satisfy $\mathcal{U}_\tau(\mathcal{R}) \uplus \mathcal{C}_\varepsilon \subseteq \gtrsim$.

[3] In order to keep our presentation simple, we do not introduce here the notions related with completeness of processors, needed for *nontermination* proofs.

[4] A binary relation R on terms is μ-monotonic if for all terms $s, t, t_1, \ldots, t_m$, and m-ary symbols f, whenever $s \,\mathsf{R}\, t$ and $i \in \mu(f)$ we have $f(\ldots, t_{i-1}, s, \ldots) \,\mathsf{R}\, f(\ldots, t_{i-1}, t, \ldots)$.

In [18], the notion of CS usable rule was given for chains of pairs. This notion is different from the one given in unrestricted rewriting. For example, if we consider the following CS problem $\tau_1 = (\{(29), (31), (32)\}, \mathcal{R}, \{(33), (34)\}, \mu^\sharp)$ obtained from Example 13, the set of CS usable rules in τ_1 is $\mathcal{R}$. This is caused by the presence of migrating variables. In the presence of migrating variables, every rule headed by a symbol appeared at a frozen positions in the right-hand side of a rule in $\mathcal{R}$ must be considered usable (in this case $*$, fact and p, and by transitivity $+$, if and zero).

But, if we look closely at the μ-rewrite sequence from Example 1 and its translation into a $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$-chain:

$$\underline{\mathsf{FACT}(\mathsf{s}^n(x))} \to_{(29)} \mathsf{IF}(\underline{\mathsf{zero}(\mathsf{s}^n(x))}, \overline{\mathsf{s}(0)}, \overline{\mathsf{s}^n(x)*\mathsf{fact}(\mathsf{p}(\mathsf{s}^n(x)))}) \hookrightarrow_{(7),\mu}$$

$$\underline{\mathsf{IF}(\mathsf{false}, \overline{\mathsf{s}(0)}, \overline{\mathsf{s}^n(x)*\mathsf{fact}(\mathsf{p}(\mathsf{s}^n(x)))})} \to_{(32)} \underline{\mathsf{s}^n(x)*\mathsf{fact}(\mathsf{p}(\mathsf{s}^n(x)))} \stackrel{\Lambda}{\to}_{(34)}$$

$$\underline{\mathsf{FACT}(\mathsf{p}(\mathsf{s}^n(x)))} \hookrightarrow_{(3),\mu} \cdots$$

we notice that x in $\mathsf{FACT}(\mathsf{p}(\mathsf{s}^n(x)))$ appears at an *active* position, but x comes from the initial term $\mathsf{FACT}(\mathsf{s}^n(x))$ where it was also at an active position, i.e., x does not behave as a migrating variable in the $(\mathsf{DP}(\mathcal{R},\mu), \mathcal{R}, \mathsf{unh}(\mathcal{R},\mu), \mu^\sharp)$-chain. Intuitively, this is equivalent to consider a pair $\mathsf{FACT}(x) \to \mathsf{FACT}(\mathsf{p}(x))$ and remove the intermediate steps. This "*conservative*" behavior allows us to ensure that only the rules defining zero and p are usable and, hence, obtain a smaller set of usable rules. Therefore, we have to find the general conditions that allow us to use this suitable set of usable rules in the μ-reduction pair processor.

6.1 Strongly Minimal Terms

The first stumbling rock in our goal comes when we try to control the shape of infinite terms (minimal non-μ-terminating terms in infinite μ-rewrite sequences) that appear at frozen positions. In the analysis of infinite μ-rewrite sequences, we obtain this control by imposing strong minimality on the initial term of the sequence, but this notion is lost in the notion of chain. Therefore, our first step is to introduce the notion of *strongly minimal* $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain. This notion ensures that the initial term of an infinite $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain does not contain any subterm that can generate an infinite $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain.

Definition 13. *An infinite $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain $u_1 \to v_1, u_2 \to v_2, \ldots$ is* strongly minimal *if it is minimal and there is no rule $s \to t \in \mathcal{S}_\sharp$ and substitutions σ, θ such that $u_1\sigma \rhd s\theta$ and $t\theta$ starts an infinite minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain.*

But the absence of infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain do not characterize the finiteness of CS problems. For example, if $\mathcal{S}_\sharp = \{\mathsf{a} \to \mathsf{F}(\mathsf{a})\}$, $\mathcal{P} = \{\mathsf{F}(x) \to x\}$, $\mathcal{R} = \emptyset$ and $\mu(f) = \emptyset$ for all f in the signature, we have the infinite minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain $\mathsf{F}(\mathsf{a}) \to_{\mathcal{P}} \mathsf{a} \stackrel{\Lambda}{\to}_{\mathcal{S}} \mathsf{F}(\mathsf{a}) \to_{\mathcal{P}} \cdots$ which is not strongly minimal. Furthermore, there is no infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain. The following result allows us to use strongly minimal chains in the CS-DP framework by imposing an structural condition on rules in $\mathcal{S}_\sharp$. Rules in $\mathsf{unh}_\sharp(\mathcal{R},\mu)$ always satisfy the condition imposed on $\mathcal{S}_\sharp$ in Theorem 3.

Theorem 3. *Let $\tau = (\mathcal{P},\mathcal{R},\mathcal{S},\mu)$ be a CS problem such that for every $s \to t \in \mathcal{S}_\sharp$, $s = f(s_1,\ldots,s_m)$ and $t = g(s_1,\ldots,s_m)$. Then, τ is* finite *if there is no infinite strongly minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain.*

Proof. If τ is not finite, then there is an infinite minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain of pairs $u_1 \to v_1, u_2 \to v_2, \ldots$ together with a substitution σ. By structural induction on $u_1\sigma$:

1. If there is no u'_1, rule $s \to t \in \mathcal{S}_\sharp$ and substitution θ such that $u_1\sigma \rhd u'_1$ and $u'_1 = s\theta$, then the infinite minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain is strongly minimal.
2. If there is u'_1, rule $s \to t \in \mathcal{S}_\sharp$ and substitution θ such that $u_1\sigma \rhd u'_1$ and $u'_1 = s\theta$, then without loss of generality we can choose u'_1 to be minimal (i.e., u'_1 has no subterm satisfying the previous conditions) and there is an infinite minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain starting from $t\theta$, by Definition 13. The obtained infinite minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain is strongly minimal, if not there is u''_1 such that $t\theta \rhd u''_1$. But, by hypothesis, $s\theta \rhd u''_1$ contradicting the minimality of u'_1.

6.2 Left-Linearity and μ-Conservativity

The second stumbling rock in our goal comes when we want to ensure that any term occurring at a frozen position does not show up at an active position by means of a variable instantiation after pair or rule applications. We will make use of left-linearity and conservativity conditions. Left-linearity allow us to discard rules which left-hand side variables are at the same time at frozen and active positions, because we impose its unicity. A rule $\ell \to r$ is μ-conservative if $\mathcal{V}ar^\mu(r) \subseteq \mathcal{V}ar^\mu(\ell)$, i.e., there is no migrating variable. Collapsing pairs are not conservative, but if we ensure that when we introduce a possible infinite term at a frozen position in the chain (as $\mathsf{fact}(\mathsf{p}(x))$ in rule (7) or pair rule (29)) it remains unaltered until it shows up by means of a $\mathcal{S}_\sharp$ rule application (in this case, rule (34)), we only need to pay attention to the rule or pair $\ell \to r$ that introduce the possible infinite term u in the chain at a frozen position, $r \rhd_{\not\mu} u$, (i.e., rule (7) or pair rule (29)) and check that $\ell \to u$ (i.e. $\mathsf{fact}(x) \to \mathsf{fact}(\mathsf{p}(x))$ and $\mathsf{FACT}(x) \to \mathsf{fact}(\mathsf{p}(x))$) is conservative. If so, we say that the CS problem is *conservative with respect to $\mathcal{S}$*.

Definition 14 (Conditions for $\mathcal{S}$). *Let $\tau = (\mathcal{P},\mathcal{R},\mathcal{S},\mu)$ be a CS problem. We say that τ is conservative with respect to $\mathcal{S}$ if $\mathcal{S}$ is conservative and the following conditions hold:*

- *for all $s \to t \in \mathcal{S}_\sharp$, $s = f(s_1,\ldots,s_m)$ and $t = g(s_1,\ldots,s_m)$; and,*
- *for each $s \to t \in \mathcal{S}_\sharp$ and for each $u \to v \in \mathcal{P} \cup \mathcal{R}$, if there is a nonvariable subterm v' of v at a frozen position such that $\theta = mgu(v',s)$, then $v' = s$ up to renaming of variables and $u \to v'$ must be conservative.*

These conditions always hold if $\mathcal{S}_\sharp \subseteq \mathsf{unh}_\sharp(\mathcal{R},\mu)$.

6.3 Extended Basic CS Usable Rules

We define our set of usable rule in the usual way. Let $\mathcal{F}un^{\mu}(s)$ be the set of symbols at active positions in a term $s \in \mathcal{T}(\mathcal{F},\mathcal{X})$, $\mathcal{F}un^{\mu}(s) = \{f \mid \exists p \in \mathcal{P}os^{\mu}(s), f = \mathsf{root}(s|_p)\}$. and $\mathcal{F}un^{\not\mu}(s)$ the set of symbols at frozen positions in a term $s \in \mathcal{T}(\mathcal{F},\mathcal{X})$, $\mathcal{F}un^{\not\mu}(s) = \{f \mid \exists p \in \mathcal{P}os(s) - \mathcal{P}os^{\mu}(s), f = \mathsf{root}(s|_p)\}$. Let $Rls_{\mathcal{R}}(f) = \{\ell \to r \in \mathcal{R} \mid \mathsf{root}(\ell) = f\}$.

Definition 15 (Extended Basic μ-Dependency). *Given a TRS $(\mathcal{F}, R)$ and a replacement map μ, we say that $f \in \mathcal{F}$ has an* extended basic μ-dependency *on $h \in \mathcal{F}$, written $f \rhd_{\mathcal{R},\mu} h$, if $f = h$ or there is a function symbol g with $g \rhd_{\mathcal{R},\mu} h$ and a rule $\ell \to r \in Rls_{\mathcal{R}}(f)$ with $g \in \mathcal{F}un^{\not\mu}(\ell) \cup \mathcal{F}un^{\mu}(r)$.*

Definition 16 (Extended Basic CS Usable Rules). *Let $\tau = (\mathcal{P},\mathcal{R},\mathcal{S},\mu)$ be a CS problem. The set $\mathcal{U}^{\rhd}_{\tau}(\mathcal{R})$ of* extended basic context-sensitive usable rules *of τ is*

$$\mathcal{U}^{\rhd}_{\tau}(\mathcal{R}) = \bigcup_{u \to v \in \mathcal{P}\cup\mathcal{S}, f \in \mathcal{F}un^{\not\mu}(u) \cup \mathcal{F}un^{\mu}(v), f \rhd_{\mathcal{R},\mu} g} Rls_{\mathcal{R}}(g)$$

We obtain the processor Proc_{UR}. The pairs $\mathcal{P}$ in a CS problem $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$, where $\mathcal{P}$ is a TRS over the signature $\mathcal{G}$, are partitioned as follows: $\mathcal{P}_{\mathcal{X}} = \{u \to v \in \mathcal{P} \mid v \in \mathcal{V}ar(u) - \mathcal{V}ar^{\mu}(u)\}$ and $\mathcal{P}_{\mathcal{G}} = \mathcal{P} - \mathcal{P}_{\mathcal{X}}$.

Theorem 4. *Let $\tau = (\mathcal{P},\mathcal{R},\mathcal{S},\mu)$ be a CS problem such that (a) $\mathcal{P}_{\mathcal{G}} \cup \mathcal{U}^{\rhd}_{\tau}(\mathcal{R}) \cup \mathcal{S}_{\rhd_{\mu}}$ is left-linear and conservative, and (b) whenever $\mathcal{P}_{\mathcal{X}} \neq \emptyset$ we have that $\mathcal{P}_{\mathcal{X}}$ is left-linear and τ is conservative with respect to $\mathcal{S}$. Let $(\gtrsim, \sqsupset)$ be a μ-reduction pair such that (1) $\mathcal{P} \subseteq \gtrsim \cup \sqsupset$, $\mathcal{U}^{\rhd}_{\tau}(\mathcal{R}) \uplus \mathcal{C}_{\varepsilon} \subseteq \gtrsim$, (2) whenever $\mathcal{P}_{\mathcal{X}} \neq \emptyset$ we have that $\mathcal{S} \subseteq \gtrsim \cup \sqsupset$. Let $\mathcal{P}_{\sqsupset} = \{u \to v \in \mathcal{P} \mid u \sqsupset v\}$ and $\mathcal{S}_{\sqsupset} = \{s \to t \in \mathcal{S} \mid s \sqsupset t\}$. Then, the processor Proc_{UR} given by*

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

is sound.

The proof is at the end of the section, we introduce now the partial results needed to obtain it. We use the interpretation given in [18, Definition 11] to define the new interpretation.

Definition 17 (Basic μ-Interpretation [18]). *Let $(\mathcal{R},\mu)$ be a CS-TRS over the signature $\mathcal{F}$ and $\Delta \subseteq \mathcal{F}$. Let $>$ be an arbitrary total ordering on terms in $\mathcal{T}(\mathcal{F}\cup\{\bot,\mathsf{c}\},\mathcal{X})$ where $\bot$ is a fresh constant symbol and c is a fresh binary symbol. The basic μ-interpretation $\mathcal{I}_{0,\Delta,\mu}$ is a mapping from μ-terminating terms in $\mathcal{T}(\mathcal{F},\mathcal{X})$ to terms in $\mathcal{T}(\mathcal{F}\cup\{\bot,\mathsf{c}\},\mathcal{X})$ defined as follows:*

$$\mathcal{I}_{0,\Delta,\mu}(t) = \begin{cases} t & \text{if } t \in \mathcal{X} \\ f(\mathcal{I}_{0,\Delta,\mu,f,1}(t_1),\ldots,\mathcal{I}_{0,\Delta,\mu,f,n}(t_k)) & \text{if } t = f(t_1,\ldots,t_k) \\ & \text{and } f \in \Delta \\ \mathsf{c}(f(\mathcal{I}_{0,\Delta,\mu,f,1}(t_1),\ldots,\mathcal{I}_{0,\Delta,\mu,f,n}(t_k)),t') & \text{if } t = f(t_1,\ldots,t_k) \\ & \text{and } f \notin \Delta \end{cases}$$

$$\textit{where} \quad \mathcal{I}_{0,\Delta,\mu,f,i}(t) = \begin{cases} \mathcal{I}_{0,\Delta,\mu}(t) & \text{if } i \in \mu(f) \\ t & \text{if } i \notin \mu(f) \end{cases}$$
$$t' = \mathit{order}\left(\{\mathcal{I}_{0,\Delta,\mu}(u) \mid t \hookrightarrow_{\mathcal{R},\mu} u\}\right)$$
$$\mathit{order}(T) = \begin{cases} \bot, & \text{if } T = \emptyset \\ \mathsf{c}(t, \mathit{order}(T - \{t\})) & \text{if } t \text{ is minimal in } T \text{ w.r.t. } > \end{cases}$$

Lemma 4. [18] *Let $(\mathcal{R}, \mu)$ be a CS-TRS over the signature $\mathcal{F}$ and t in $\mathcal{T}(\mathcal{F}, \mathcal{X})$. If t is μ-terminating then $\mathcal{I}_{0,\Delta,\mu}$ is well-defined.*

But, in order to deal with collapsing pairs, we allow that at frozen positions we can have terms that are interpreted (or partially interpreted). Then, to have a unique interpretation for each term we have to parametrize it with respect to an infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain.

Definition 18 (Extended Basic μ-Interpretation). *Let $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$ be a CS problem where $\mathcal{R}$ is a TRS over the signature $\mathcal{F}$ and $\Delta \subseteq \mathcal{F}$. Let $\mathcal{A}$ be an infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain of the form $u_1 \to v_1, u_2 \to v_2, \cdots$ Let $\mathcal{I}_{\Delta,\mu,\mathcal{A}}$ be an interpretation that satisfies:*

- *$\mathcal{I}_{\Delta,\mu,\mathcal{A}}(u_1\sigma) = \mathcal{I}_{0,\Delta,\mu}(u_1\sigma)$, and*
- *if there is a pair $u_i \to v_i \in \mathcal{P}$ and a variable $x \in Var(u_i) \cap \mathcal{V}ar(v_i)$, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(x\sigma)$ has the same interpretation in u_i and v_i.*
- *if there is a rule $\ell_i \to r_i \in \mathcal{R}$ and a variable $x \in Var(\ell_i) \cap \mathcal{V}ar(r_i)$, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(x\sigma)$ has the same interpretation in ℓ_i and r_i.*

Definition 19. *Let $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$ be a CS problem where $\mathcal{R}$ is a TRS over the signature $\mathcal{F}$ and $\Delta \subseteq \mathcal{F}$. Let $\mathcal{A}$ be an infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain. We denote by $\sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}$ a substitution that replaces occurrences of $x \in \mathcal{V}ar(t)$ by $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(x\sigma)$.*

Lemma 5. *Let $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$ be a CS problem where $\mathcal{R}$ is a TRS over the signature $\mathcal{F}$ and $\Delta \subseteq \mathcal{F}$. Let $\mathcal{A}$ be an infinite strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain. Let t be a linear term and σ be a substitution. If all subterms t' of t at frozen positions are from $\mathcal{T}(\Delta, \mathcal{X})$ and $t\sigma$ is $(\mathcal{R}, \mu)$-terminating, then $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$. If t only contain Δ-symbols at active positions, then we have $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.*

Proof. By structural induction on t:

- If t is a variable then $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.
- If $t = f(t'_1, \ldots, t'_k)$ then
  - If $f \in \Delta$ then $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = f(t''_1, \ldots, t''_k)$. Terms $t'_i\sigma$ are μ-terminating for $i \in \mu(f)$. By induction hypothesis, for all terms t'_i s.t. $i \in \mu(f)$, we have $t''_i = \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t'_i\sigma) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t'_i)$. And for all t'_i s.t. $i \notin \mu(f)$, we have that t'_i only contains Δ symbols, then $t''_i = \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t'_i\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t'_i)$. This implies $f(t''_1, \ldots, t''_k) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.
  - If $f \notin \Delta$, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = \mathsf{c}(f(t''_1, \ldots, t''_k), t')$ for some t'. Applying a $\mathcal{C}_\varepsilon$ step to this term, we obtain again the term $f(t''_1, \ldots, t''_k)$, and using the previous item result, we get $f(t''_1, \ldots, t''_k) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.

Then we conclude $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.

The second part of the lemma is proved similarly. If t is a variable then $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$. Now let $t = f(t_1,\ldots,t_k)$. Since $f \in \Delta$, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t\sigma) = \mathcal{I}_{\Delta,\mu,\mathcal{A}}(f(t'_1\sigma,\ldots,t'_k\sigma)) = f(t''_1,\ldots,t''_k)$. For $i \in \mu(f)$, we have $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t'_i\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t'_i)$ by the induction hypothesis. For $i \notin \mu(f)$, we have $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(t'_i\sigma) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t'_i)$. This implies that $f(t''_1,\ldots,t''_k) = \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(t)$.

Lemma 6. *Let $\tau = (\mathcal{P},\mathcal{R},\mathcal{S},\mu)$ be a CS problem where $\mathcal{P}$ and $\mathcal{R}$ are TRSs over the signatures $\mathcal{G}$ and $\mathcal{F}$ respectively and $\Delta \subseteq \mathcal{F}$. Let $\mathcal{A}$ be an infinite strongly minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain. Let $\mathcal{P}_\mathcal{G} \cup \mathcal{U}^{\rhd}_\tau(\mathcal{R})$ be conservative and left-linear, $\mathcal{P}_\mathcal{X}$ left-linear and conservative with respect to $\mathcal{S}$, and $\Delta = \{\mathsf{root}(\ell) \mid \ell \to r \in \mathcal{U}^{\rhd}_\tau(\mathcal{R})\}$. If s and t are $(\mathcal{R},\mu)$-terminating and $s \hookrightarrow_{\mathcal{R},\mu} t$ then $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(s) \hookrightarrow^+_{\mathcal{U}^{\rhd}_\tau(\mathcal{R})\uplus\mathcal{C}_\varepsilon,\mu} \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t)$.*

Proof. We proceed by induction on the position p of the redex in $s \hookrightarrow_{\{\ell\to r\},\mu} t$. First assume that $\mathsf{root}(s) \in \Delta$ and $p = \Lambda$ (and therefore $\ell \to r \in \mathcal{U}^{\rhd}_\tau(\mathcal{R})$). So we have $s = \ell\sigma \xrightarrow{\Lambda}_{\{\ell\to r\},\mu} r\sigma = t$ for some substitution σ. Moreover, for all subterms r' at active positions of r, $\mathsf{root}(r') \in \Delta$ and for all subterms ℓ' at frozen positions of ℓ, $\mathsf{root}(\ell') \in \Delta$ by definition of Δ. We know that $\ell \to r$ is conservative and left-linear. We have:

$$\begin{array}{rcll} \mathcal{I}_{\Delta,\mu,\mathcal{A}}(s) & = & \mathcal{I}_{\Delta,\mu,\mathcal{A}}(\ell\sigma) & \\ & \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} & \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(\ell) & \text{by Lemma 5} \\ & \to_{\{\ell\to r\}} & \sigma_{\mathcal{I}_{\Delta,\mu,\mathcal{A}}}(r) = \mathcal{I}_{\Delta,\mu,\mathcal{A}}(r\sigma) = \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t) & \end{array}$$

Note that, by conservativity, every variable at an active position in r is at an active position in ℓ. Now consider the case where $\mathsf{root}(s) \in \Delta$ and $p \neq \Lambda$. Hence, $s = f(s'_1,\ldots,s'_i,\ldots,s'_n)$, $t = f(s'_1,\ldots,t'_i,\ldots,s'_n)$, $i \in \mu(f)$, and $s'_i \hookrightarrow_{\{\ell\to r\},\mu} t'_i$. The induction hypothesis implies $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(s'_i) \hookrightarrow^+_{\{\ell\to r\}\uplus\mathcal{C}_\varepsilon,\mu} \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t'_i)$ and hence, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(s) \hookrightarrow^+_{\{\ell\to r\}\uplus\mathcal{C}_\varepsilon,\mu} \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t)$. Finally, we consider the case $\mathsf{root}(s) \notin \Delta$. In this case, $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(s) \in \mathit{order}\,(\{\mathcal{I}_{\Delta,\mu,\mathcal{A}}(u) \mid s \hookrightarrow_{\mathcal{R},\mu} u\})$ because $s \hookrightarrow_{\mathcal{R},\mu} t$. By applying $\mathcal{C}_\varepsilon$ rules, we get $\mathcal{I}_{\Delta,\mu,\mathcal{A}}(s) \hookrightarrow^+_{\mathcal{C}_\varepsilon,\mu} \mathcal{I}_{\Delta,\mu,\mathcal{A}}(t)$.

Proof (Theorem 4). Regarding soundness, we proceed by contradiction. By Theorem 3, assume that there is an infinite strongly minimal $(\mathcal{P},\mathcal{R},\mathcal{S},\mu)$-chain $\mathcal{A}$, but there is no infinite strongly minimal $(\mathcal{P}-\mathcal{P}_\sqsupset,\mathcal{R},\mathcal{S}-\mathcal{S}_\sqsupset,\mu)$-chain. Due to the finiteness of $\mathcal{P}$ and $\mathcal{S}$, we can assume that there are subsets $\mathcal{Q} \subseteq \mathcal{P}$ and $\mathcal{T} \subseteq \mathcal{S}$ such that A has a tail B

$$u_1\sigma \left\{ \begin{array}{c} \to_{\mathcal{Q}_\mathcal{G}} \\ \to_{\mathcal{Q}_\mathcal{X}} \circ \xrightarrow{\Lambda}{}^*_{\mathcal{T}_{\rhd_\mu}} \circ \xrightarrow{\Lambda}_{\mathcal{T}_\sharp} \end{array} \right\} t'_1 \hookrightarrow^*_{\mathcal{R},\mu} u_2\sigma \left\{ \begin{array}{c} \to_{\mathcal{Q}_\mathcal{G}} \\ \to_{\mathcal{Q}_\mathcal{X}} \circ \xrightarrow{\Lambda}{}^*_{\mathcal{T}_{\rhd_\mu}} \circ \xrightarrow{\Lambda}_{\mathcal{T}_\sharp} \end{array} \right\} \cdots$$

for some substitution σ, where all pairs in $\mathcal{Q}$ and all rules in $\mathcal{T}$ are infinitely often used (note that, if $\mathcal{T} \neq \emptyset$, then $\mathcal{T}_\sharp \neq \emptyset$ and $\mathcal{Q}_\mathcal{X} \neq \emptyset$), and, for all $i \geq 1$, (1) if $u_i \to v_i \in \mathcal{Q}_\mathcal{G}$, then $t'_i = v_i\sigma$ and (2) if $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_\mathcal{X}$, then $x_i\sigma \xrightarrow{\Lambda}{}^*_{\mathcal{T}_{\rhd_\mu}} \circ \xrightarrow{\Lambda}_{\mathcal{T}_\sharp} t'_i$. Moreover, all t'_i are $(\mathcal{R},\mu)$-terminating.

We know that $\mathcal{P}\cup\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})\cup\mathcal{S}_{\triangleright_\mu}$ is conservative and left-linear. We apply $\mathcal{I}_{\Delta,\mu,\mathcal{A}}$ in Definition 18 to the initial term. To ease readability, we let $I = \mathcal{I}_{\Delta,\mu,\mathcal{A}}$. By definition, we know that $I(u_1\sigma) = \mathcal{I}_{0,\Delta,\mu}(u_1\sigma)$. Let $\Delta = \{\mathsf{root}(\ell) \mid \ell \to r \in \mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})\}$. Note that the application of $\mathcal{I}_{0,\Delta,\mu}$ is always possible since active subterms are $(\mathcal{R},\mu)$-terminating due to the minimality of the chain. Using Lemma 6, we obtain $I(v_i\sigma) \hookrightarrow^*_{\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})\uplus\mathcal{C}_\varepsilon,\mu} I(u_{i+1}\sigma)$ for all $i \geq 1$. Moreover, by the definition of $\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})$, for all nonvariable active subterms v'_i of v_i, we have $\mathsf{root}(v'_i) \in \Delta$. By Lemma 6, we have $v_i\sigma_I = I(v_i\sigma)$ and $I(u_{i+1}\sigma) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} u_{i+1}\sigma_I$. Since $u_i\,(\gtrsim\cup\sqsupset)v_i$ for all $u_i \to v_i \in \mathcal{Q} \subseteq \mathcal{P}$, by stability of $\gtrsim$ and $\sqsupset$, we have $u_i\sigma_I\,(\gtrsim\cup\sqsupset)\,v_i\sigma_I$ for all $i \geq 1$.

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

1. If $u_i \to v_i \in \mathcal{Q}_{\mathcal{F}}$, then $v_i\sigma_I \hookrightarrow^*_{\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})\uplus\mathcal{C}_\varepsilon,\mu} u_{i+1}\sigma$ and (by compatibility of $\gtrsim$ with the rules in $\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})\uplus\mathcal{C}_\varepsilon$) $v_i\sigma_I \gtrsim u_{i+1}\sigma_I$. For all active subterms v'_i of r, $\mathsf{root}(v'_i) \in \Delta$. Then $I(u_i) \hookrightarrow^*_{\mathcal{C}_\varepsilon,\mu} u_i\sigma_I$ and $v_i\sigma_I = I(v_i)$ (we use the fact that $\mathcal{P}$ is conservative and left-linear). Since $u_i\sigma_I\,(\gtrsim\cup\sqsupset)\,v_i\sigma_I$, by using transitivity of $\gtrsim$ and compatibility between $\gtrsim$ and $\sqsupset$, we conclude that $u_i\sigma_I\,(\gtrsim\cup\sqsupset)\,u_{i+1}\sigma_I$.
2. If $u_i \to v_i = u_i \to x_i \in \mathcal{Q}_{\mathcal{X}}$ (which is not empty whenever $\mathcal{T}$ is not empty), then $v_i\sigma_I = x_i\sigma_I \hookrightarrow^*_{\mathcal{T}_{\triangleright_\mu}\uplus\mathcal{C}_\varepsilon,\mu} I(s_i)$, as in Lemma 6. Let $\ell_i \to r_i \in \mathcal{T}_\sharp$. For all subterms r'_i at active positions of r_i, $\mathsf{root}(r'_i) \in \Delta$. If ℓ_i is not linear, we know that $u_1 \not\trianglerighteq \ell_i\sigma$ and there is a rule $u \to v \in \mathcal{P}\cup\mathcal{R}$ such that $u\sigma \not\trianglerighteq \ell_i\sigma$ and $v\sigma \triangleright_{\not\mu} \ell_i\sigma$ and we know that u is left-linear and $v = \ell_i$ up to renaming of variables, so σ_I is unique. Since $\ell(\gtrsim\cup\sqsupset)r$ for all $\ell \to r \in \mathcal{T}$, we have $v_i\sigma_I = x_i\sigma_I(\gtrsim\cup\sqsupset)I(s_i)(\gtrsim\cup\sqsupset)r_i\sigma_I = I(t_i)$. Hence, by transitivity of $\gtrsim$ (and compatibility of $\gtrsim$ and $\sqsupset$), we have $v_i\sigma = x_i\sigma\,(\gtrsim\cup\sqsupset)\,t_i$. Since $I(t_i) \hookrightarrow^*_{\mathcal{R},\mu} u_{i+1}\sigma_I$, we also have that, for all $i \geq 1$, $I(t_i) \gtrsim u_{i+1}\sigma_I$. Therefore, again by transitivity of $\gtrsim$ and compatibility of $\gtrsim$ and $\sqsupset$, we conclude that $u_i\sigma_I(\gtrsim\cup\sqsupset)\,I(t_i) \gtrsim u_{i+1}\sigma_I$ and hence $u_i\sigma_I\,(\gtrsim\cup\sqsupset)\,u_{i+1}\sigma_I$.

Since $u \to v$ occurs infinitely often in B, there is an infinite set $\mathcal{I} \subseteq \mathbb{N}$ of pairs such that $u_i\sigma_I \sqsupset u_{i+1}\sigma_I$ for all $i \in \mathcal{I}$. Thus, by using the compatibility conditions of the μ-reduction pair, we obtain an infinite decreasing $\sqsupset$-sequence which contradicts well-foundedness of $\sqsupset$. Therefore, B is an infinite minimal $(\mathcal{P}-\mathcal{P}_\sqsupset,\mathcal{R},\mathcal{S}-\mathcal{S}_\sqsupset,\mu)$-chain, thus leading to a contradiction.

Example 15. In Example 1, we start with the CS problem $\tau_0 = (\mathsf{DP}(\mathcal{R},\mu),\mathcal{R},\mathsf{unh}(\mathcal{R},\mu),\mu^\sharp)$*. Applying the well-known SCC processor [17] to* τ_0*,* $\mathsf{Proc}_{SCC}(\tau_0)$*, we get the new set of CS problems* $\mathsf{Proc}_{SCC}(\tau_0) = \{\tau_1,\tau_2,\tau_3\}$ *using the computed CS dependency graph from Figure 4, where* $\tau_1 = (\{(26)\},\mathcal{R},\emptyset,\mu)$*,* $\tau_2 = (\{(28)\},\mathcal{R},\emptyset,\mu)$ *and* $\tau_3 = (\{(29),(31),(32)\},\mathcal{R},\{(34)\},\mu)$*. Applying the well-known* μ*-subterm processor [17] to CS problems* τ_1 *and* τ_2 *we get* $\mathsf{Proc}_{sub}(\tau_1) = \tau_4$ *and* $\mathsf{Proc}_{sub}(\tau_2) = \tau_4$*, where* $\tau_4 = (\emptyset,\mathcal{R},\emptyset,\mu)$ *and, hence, we can conclude that* τ_1 *and* τ_2 *are finite.*

But, until now, CS problem τ_3 *could not be handled by any automatic tool. By Definition 16, the set of extended basic CS usable rules* $\mathcal{U}^{\triangleright}_{\mathcal{T}}(\mathcal{R})$ *is:*

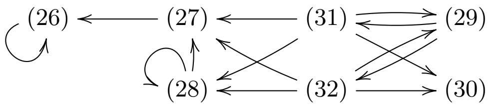

Fig. 4. CS Dependency Graph for Example 1

$$\mathsf{zero}(\mathsf{0}) \to \mathsf{true} \qquad \mathsf{zero}(\mathsf{s}(x)) \to \mathsf{false} \qquad \mathsf{p}(\mathsf{s}(x)) \to x$$

when in the previous approach all the rules are usable. We can use the extended basic CS usable rules instead of $\mathcal{R}$ because the CS problem satisfies the restrictions in Theorem 4 and the following polynomial interpretation [25] allows us to remove pair (32)*:*

$$\begin{array}{ll@{\qquad}ll}
[\mathsf{fact}](x) & = 2x & [*](x,y) & = \frac{1}{2}xy + 2 \\
[\mathsf{p}](x) & = \frac{1}{2}x & [\mathsf{zero}] & = \frac{1}{2}x^2 \\
[\mathsf{0}] & = 2 & [\mathsf{s}](x) & = 2x + 1 \\
[\mathsf{false}] & = \frac{1}{2} & [\mathsf{true}] & = 2 \\
[\mathsf{FACT}](x) & = 2x^2 + 2 & [\mathsf{IF}](x,y,z) & = \frac{1}{2}xy + \frac{1}{2}x + z
\end{array}$$

The new CS problem $\tau_5 = (\{(29), (31)\}, \mathcal{R}, \{(34)\}, \mu)$ can be handled again using Theorem 4. The following polynomial interpretation removes pair (31)*:*

$$\begin{array}{ll@{\qquad}ll}
[\mathsf{fact}](x) & = 1 & [*](x,y) & = 2x + 2 \\
[\mathsf{0}] & = 0 & [\mathsf{s}](x) & = 2x \\
[\mathsf{p}](x) & = 2x + 1 & [\mathsf{zero}] & = 2x + 1 \\
[\mathsf{false}] & = 1 & [\mathsf{true}] & = 1 \\
[\mathsf{FACT}](x) & = 2 & [\mathsf{IF}](x,y,z) & = y + 1
\end{array}$$

and we obtain a finite CS problem by applying Proc_{SCC} *to the resulting CS problem.*

7 Experimental Evaluation

We have performed an experimental evaluation of the new improvements introduced by these new results presented in the paper in our tool for proving termination properties, MU-TERM [4]. We compared our new version, we call it MU-TERM 5.1, with respect to the previous version, MU-TERM 5.08 [17]. The experiments have been performed on an Intel Core 2 Duo at 2.4GHz with 8GB of RAM, running OS X 10.9.1 using a 120 seconds timeout. We used the last version of the termination problem database, TPDB 8.0.7[5], context-sensitive category. Results are in `http://zenon.dsic.upv.es/muterm/benchmarks/lrc15-csr/benchmarks.html` and summarized in Table 1. MU-TERM 5.1 also participated in the CSR category in the 2014 termination competition (`http://termination-portal.org/wiki/Termination_Competition_2014`) and the same results were confirmed.

[5] See `http://termcomp.uibk.ac.at/termcomp/`

Tool Version	Proved	Total time (Av. time)
MU-TERM **5.1**	102/109	1.62s
MU-TERM **5.08**	99/109	2.23s

Table 1. MU-TERM 5.1 vs. MU-TERM 5.08 comparison

The practical improvements revealed by the experimental evaluation are twofold. First, we can prove (now) termination of 102 of the 109 examples, 3 more examples than our previous version, including [28, Example 1], [13, Example 1] and [10, Example 3.2.14], whose automatic proofs were open problems since 1997, 2003 and 2008. To our knowledge, there is no other tool that can prove more than those 99 examples from this collection of problems. Second, the new definitions yield a faster implementation; this is witnessed by a speed-up of 1.37 with respect to our previous version.

8 Conclusions

In this paper, we revisit infinite μ-rewrite sequences to obtain a new notion of minimal non-μ-terminating term and a new set of unhiding rules. Since the introduction of the CS-DPs in 2006, the constraints introduced by the unhiding process have been a headache for constraint solvers. For example, in the original approach for each symbol f in the signature and replacing argument $i \in \mu(f)$, a *projection constraint* $f(x_1, \ldots, x_n) \geq x_i$ should be satisfied in order to find a proof. Subsequent works [1, 17] reduced these projection constraints to a subset of those for the hidden symbols. Now, in many cases, as in the leading example of the paper, we can avoid these projection/embedding constraints and the unhiding rules are a very small set of rules. In the context of the CS-DP framework, we propose a new notion of chain, the notion of strongly minimal $(\mathcal{P}, \mathcal{R}, \mathcal{S}, \mu)$-chain and a new set of CS usable rules, the extended basic CS usable rules, that allows us to simplify termination proofs on CS problems with respect to the set of unhiding rules. The new processor leads us to a faster and more powerful CS-DP framework. We show an example where the technique is successfully applied [28, Example 1] (included in the TPDB), whose automatic proof was an open problem since 1997. An implementation and an experimental evaluation was performed in our tool for proving termination properties, MU-TERM [4]. With these improvements, MU-TERM won the CSR category in the 2014 termination competition.

References

1. Alarcón, B., Emmes, F., Fuhs, C., Giesl, J., Gutiérrez, R., Lucas, S., Schneider-Kamp, P., Thiemann, R.: Improving Context-Sensitive Dependency Pairs. In: Cervesato, I., Veith, H., Voronkov, A. (eds.) Proc. of the 15th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, LPAR'08. LNCS, vol. 5330, pp. 636–651. Springer (2008)
2. Alarcón, B., Gutiérrez, R., Lucas, S.: Context-Sensitive Dependency Pairs. In: Arun-Kumar, S., Garg, N. (eds.) Proc. of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science, FST&TCS'06. LNCS, vol. 4337, pp. 297–308. Springer (2006)

3. Alarcón, B., Gutiérrez, R., Lucas, S.: Context-Sensitive Dependency Pairs. Information and Computation 208, 922–968 (2010)
4. Alarcón, B., Gutiérrez, R., Lucas, S., Navarro-Marset, R.: Proving Termination Properties with MU-TERM. In: Johnson, M., Pavlovic, D. (eds.) Proc. of the 13th International Conference on Algebraic Methodology and Software Technology, AMAST'10. LNCS, vol. 6486, pp. 201–208. Springer (2011)
5. Arts, T., Giesl, J.: Proving Innermost Normalisation Automatically. In: Comon, H. (ed.) Proc. of the 8th International Conference on Rewriting Techniques and Applications, RTA'97. LNCS, vol. 1232, pp. 157–171. Springer (1997)
6. Arts, T., Giesl, J.: Termination of Term Rewriting Using Dependency Pairs. Theoretical Computer Science 236(1–2), 133–178 (2000)
7. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press (1998)
8. Borovanský, P., Kirchner, C., Kirchner, H., Moreau, P., Ringeissen, C.: An overview of ELAN. Electr. Notes Theor. Comput. Sci. 15, 55–70 (1998), `http://dx.doi.org/10.1016/S1571-0661(05)82552-6`
9. Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Talcott, C.: All About Maude – A High-Performance Logical Framework, LNCS, vol. 4350. Springer (2007)
10. Emmes, F.: Automated Termination Analysis of Context-Sensitive Term Rewrite Systems. Master's thesis, Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen, Aachen, Germany (2008)
11. Futatsugi, K., Goguen, J.A., Jouannaud, J.P., Meseguer, J.: Principles of OBJ2. In: Proc. of the 12th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages, POPL'85. pp. 52–66. ACM (1985)
12. Futatsugi, K., Nakagawa, A.: An overview of CAFE Specification Environment–An algebraic approach for creating, verifying, and maintaining formal specifications over networks. In: Proc. of the 1st International Conference on Formal Engineering Methods, ICFEM '97. p. 170. IEEE Computer Society (1997)
13. Giesl, J., Middeldorp, A.: Innermost Termination of Context-Sensitive Rewriting. In: Ito, M., Toyama, M. (eds.) Proc. of the 6th International Conference on Developments in Language Theory, DLT'02. LNAI, vol. 2450, pp. 231–244. Springer (2003)
14. Giesl, J., Thiemann, R., Schneider-Kamp, P., Falke, S.: Mechanizing and Improving Dependency Pairs. Journal of Automatic Reasoning 37(3), 155–203 (2006)
15. Goguen, J.A., Winkler, T., Meseguer, J., Futatsugi, K., Jouannaud, J.P.: Software Engineering with OBJ: Algebraic Specification in Action, chap. Introducing OBJ. Kluwer (2000)
16. Gramlich, B.: Generalized Sufficient Conditions for Modular Termination of Rewriting. Applicable Algebra in Engineering, Comunication and Computing 5, 131–151 (1994)
17. Gutiérrez, R., Lucas, S.: Proving Termination in the Context-Sensitive Dependency Pair Framework. In: Ölveczky, P. (ed.) Proc. of the 8th International Workshop on Rewriting Logic and its Applications, WRLA'10. LNCS, vol. 6381, pp. 19–35. Springer (2010)
18. Gutiérrez, R., Lucas, S., Urbain, X.: Usable Rules for Context-Sensitive Rewrite Systems. In: Voronkov, A. (ed.) Proc. of the 19th International Conference on Rewriting Techniques and Applications, RTA'08. LNCS, vol. 5117, pp. 126–141. Springer (2008)

19. Hendrix, J., Meseguer, J.: On the completeness of context-sensitive order-sorted specifications. In: Baader, F. (ed.) Term Rewriting and Applications, 18th International Conference, RTA 2007, Paris, France, June 26-28, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4533, pp. 229–245. Springer (2007), http://dx.doi.org/10.1007/978-3-540-73449-9_18
20. Hirokawa, N., Middeldorp, A.: Dependency Pairs Revisited. In: van Oostrom, V. (ed.) Proc. of the 15th International Conference on Rewriting Techniques and Applications, RTA'04. LNCS, vol. 3091, pp. 249–268. Springer (2004)
21. Hirokawa, N., Middeldorp, A.: Tyrolean Termination Tool: Techniques and Features. Information and Computation 205(4), 474–511 (2007)
22. Lucas, S.: Context-Sensitive Computations in Functional and Functional Logic Programs. Journal of Functional and Logic Programming 1998(1), 1–61 (1998)
23. Lucas, S.: Termination of On-Demand Rewriting and Termination of OBJ Programs. In: De Nicola, R., Søndergaard, H. (eds.) Proc. of the 3rd Internatinal Conference on Principles and Practice of Declarative Programming, PPDP'01. pp. 82–93. ACM Press (2001)
24. Lucas, S.: Termination of Rewriting With Strategy Annotations. In: Nieuwenhuis, R., Voronkov, A. (eds.) Proc. of the 8th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, LPAR'01. LNAI, vol. 2250, pp. 666–680. Springer (2001)
25. Lucas, S.: Polynomials over the Reals in Proofs of Termination: from Theory to Practice. RAIRO Theoretical Informatics and Applications 39(3), 547–586 (2005)
26. Lucas, S.: Completeness of context-sensitive rewriting. Inf. Process. Lett. 115(2), 87–92 (2015), http://dx.doi.org/10.1016/j.ipl.2014.07.004
27. Ohlebusch, E.: On the Modularity of Termination of Term Rewriting Systems. Theoretical Computer Science 136, 333–360 (1994)
28. Zantema, H.: Termination of Context-Sensitive Rewriting. In: Comon, H. (ed.) Proc. of the 7th International Conference on Rewriting Techniques and Applications, RTA'97. LNCS, vol. 1232, pp. 172–186. Springer (1997)